SIOS Protection Suite for Linux

9.5.0 — Last update: 25 September 2020

SIOS TECHNOLOGY CORP.

Table of Contents

1. SIOS Protection Suite for Linux

2. SIOS Protection Suite for Linux Release Notes

Version 9.5.0

Released May 12, 2020

Important!!

Read This Document Before Attempting To Install Or Use This Product!
This document contains last minute information that must be considered before, during and after installation.

Introduction

This release notes document is written for the person who installs, configures and/or administers the SIOS Protection Suite (SPS) for Linux product. The document contains important information not detailed in the formal LifeKeeper and DataKeeper documentation sets such as package versions and last-minute changes to instructions and procedures as well as a link to the Troubleshooting sections for product restrictions and troubleshooting hints and tips that were discovered through final product testing. It is important that you review this document before installing and configuring SPS software.

SIOS Product Descriptions

LifeKeeper for Linux

The LifeKeeper product includes fault detection and recovery software that provides high availability for file systems, network addresses, applications and processes running on Linux. LifeKeeper supports the configuration and switchover of a given application across multiple servers. The servers on which the application is configured are assigned priorities to determine the sequence in which the application will move from server to server in the event of multiple failures.

LifeKeeper for Linux provides switchover protection for a range of system resources. Automatic recovery is supported for the following resource types:

- Processes and Applications
- Shared Storage Devices (Including VMWare virtual hard disks)
- Network Attached Storage Devices
- LVM Volume Groups and Logical Volumes

- File Systems (ext3, ext4, vxfs, xfs and nfs) **Note:** btrfs is not currently supported by the SIOS Protection Suite for Linux. For detailed information see LifeKeeper Core – Known Issues / Restrictions.
- Communication Resources (TCP/IP)
- Database Applications
  - Oracle
  - MySQL
  - DB2
  - SAP MaxDB
  - PostgreSQL
  - EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server
  - Sybase
- Web Server Resources
- Samba Resources
- DataKeeper for Linux
- SAP Application Environment Resources
- (blank)Software RAID (md) Resources
- WebSphere MQ Resources
- Postfix Resources

DataKeeper for Linux

The SIOS DataKeeper product:

- Provides volume-based synchronous and asynchronous data replication.
- Integrates into the LifeKeeper Graphical User Interface for administration and monitoring.
- Automatically resynchronizes data between source and target servers at system recovery.
- Monitors the health of underlying system components and performs local recovery in the event of failure.
- Allows manual resource switchovers and failovers of mirrored volumes.
- Can be easily upgraded to provide high availability clustering and automatic failover and recovery using a license key to enable new functionality.

SPS Components

SPS Core

SPS for Linux is bundled for, and only runs on, 64-bit systems (AMD64 and EM64T systems).

The SPS Core Package Cluster includes the following installable packages:

Package	Package Name	Description
LifeKeeper Core	steeleye-lk-9.5.0-7075.x86_64.rpm	The LifeKeeper package provides recovery software for failures associated with core system components such as memory, CPUs, the operating system, the SCSI disk subsystem and file systems.
DataKeeper Core	steeleye-lkDR-9.5.0-7075.noarch.rpm	The DataKeeper package provides data replication (synchronous or asynchronous mirrors with intent logging).
LifeKeeper GUI	steeleye-lkGUI-9.5.0-7075.x86_64.rpm	The LifeKeeper GUI package provides a graphical user interface for LifeKeeper and DataKeeper administration and status monitoring.
SPS IP Recovery Kit	steeleye-lkIP-9.5.0-7075.noarch.rpm	The SPS IP Recovery Kit provides recovery software for automatic switchover of IP addresses.
SPS Raw I/O Recovery Kit	steeleye-lkRAW-9.5.0-7075.noarch.rpm	The SPS Raw I/O Recovery Kit provides support for applications that use raw i/o to bypass kernel buffering.
LifeKeeper Man Pages	steeleye-lkMAN-9.5.0-7075.noarch.rpm	The LifeKeeper Man Page package provides reference manual pages for the SPS product.

SPS Optional Recovery Software

The following optional software provides resource definition and recovery software for the application versions listed. See the Support Matrix and Recovery Kit Administration Guides for the requirements for each recovery software.

Package	Package Name	Description
SPS Apache Web Server Recovery Kit	steeleye-lkAPA-9.5.0-7075.noarch.rpm	The SPS for Linux Apache Web Server Recovery Kit provides fault resilience for Apache Web Server software in an SPS environment.
SPS SAP Recovery Kit	steeleye-lkSAP-9.5.0-7075.noarch.rpm	The SPS for Linux SAP Recovery Kit provides a mechanism to recover SAP NetWeaver from a failed primary server onto a backup server in an SPS environment working in conjunction with other SPS Recovery Kits to provide comprehensive failover protection.
SPS SAP MaxDB Recovery Kit	steeleye-lkSAPDB-9.5.0-7075.noarch.rpm	The SAP MaxDB Recovery Kit provides fault resilient protection for SAP MaxDB databases in an SPS for Linux environment.
SPS DB2	steeleye-	The SPS for Linux DB2 Recovery Kit provides

Recovery Kit	lkDB2-9.5.0-7075.noarch.rpm	fault resilient protection for DB2 database instances. SPS together with the DB2 Universal Database product family afford increased availability to DB2 operating environments by effectively recovering database server failures without significant down-time or human intervention.
SPS Oracle Recovery Kit	steeleye-lkORA-9.5.0-7075.noarch.rpm	The SPS for Linux Oracle Recovery Kit provides fault resilience for Oracle software in an SPS environment furnishing a mechanism to tie the data integrity of Oracle databases to the increased availability provided by SPS.
SPS MySQL Recovery Kit	steeleye-lkSQL-9.5.0-7075.noarch.rpm	The SPS for Linux MySQL Recovery Kit provides an easy way to add SPS fault-resilient protection for MySQL resources and databases enabling a failure on the primary database server to be recovered on a designated backup server without significant lost time or human intervention.
SPS PostgreSQL Recovery Kit	steeleye-lkPGSQL-9.5.0-7075.noarch.rpm	The SPS for Linux PostgreSQL Recovery Kit is an SQL compliant, object-relational database management system (ORDBMS) based on POSTGRES providing a mechanism for protecting PostgreSQL instances within SPS.
SPS Sybase ASE Recovery Kit	steeleye-lkSYBASE-9.5.0-7075.noarch.rpm	The SPS for Linux Sybase ASE Recovery Kit provides SPS resource protection for the Sybase ASE components Adaptive Server, Monitor Server, and Backup Server.
SPS Postfix Recovery Kit	steeleye-lkPOSTFIX-9.5.0-7075.noarch.rpm	The SPS for Linux Postfix Recovery Kit provides a mechanism to recover Postfix from a failed primary server to a backup server in an SPS environment.
SPS Samba Recovery Kit	steeleye-lkSMB-9.5.0-7075.noarch.rpm	The SPS for Linux Samba Recovery Kit provides fault resilient protection for Samba file and print shares on a Linux server existing in a heterogeneous network enabling a failure on the primary Samba server to be recovered on a designated backup server without significant lost time or human intervention.
SPS NFS Server Recovery Kit	steeleye-lkNFS-9.5.0-7075.noarch.rpm	The SPS for Linux NFS Server Recovery Kit provides fault resilience for Network File System (NFS) software in an SPS environment enabling a failure on the primary NFS server to be recovered on a designated backup server without significant lost time or human intervention.
SPS Network Attached Storage Recovery Kit	steeleye-lkNAS-9.5.0-7075.noarch.rpm	The SPS for Linux Network Attached Storage Recovery Kit provides fault resilience for Network File System (NFS) software in an SPS environment affording SPS users the opportunity to employ an exported NFS file system as the storage basis for SPS

		hierarchies. NFS over UDP is not supported on Red Hat Enterprise Linux 8 and later. Some environments may require additional configurations. Refer to Specific Configuration Considerations for details.
(blank)SPS Logical Volume Manager (LVM) Recovery Kit	steeleye-lkLVM-9.5.0-7075.noarch.rpm	The SPS for Linux Logical Volume Manager (LVM) Recovery Kit provides logical volume support for other SPS recovery kits allowing SPS-protected applications to take advantage of the benefits offered by the Logical Volume Manager, including simplified storage management and the ability to dynamically re-size volumes as needs change.
(blank)SPS Software RAID (md) Recovery Kit	steeleye-lkMD-9.5.0-7075.noarch.rpm	The SPS for Linux Software RAID (md) Recovery Kit provides software RAID support for other SPS recovery kits allowing SPS-protected applications to take advantage of the benefits offered by software RAID, including lower cost data redundancy, data replication over a SAN and simplified storage management.
SPS PowerPath Recovery Kit	steeleye-lkPPATH-9.5.0-7075.noarch.rpm	The SPS PowerPath Recovery Kit protects applications that use EMC PowerPath multipath I/O devices.
SPS Device Mapper Multipath (DMMP) Recovery Kit	steeleye-lkDMMP-9.5.0-7075.noarch.rpm	The SPS Device Mapper Multipath (DMMP Recovery Kit) protects applications and file systems that use DMMP devices allowing SPS to operate with and protect these applications and file systems.
Hitachi Dynamic Link Manager Software (HDLM) Recovery Kit	steeleye-lkHDLM-9.5.0-7075.noarch.rpm	The Hitachi Dynamic Link Manager Software (HDLM) Recovery Kit protects applications that use Hitachi Dynamic Link Manager Software devices.
SPS NEC iStorage StoragePathSavior (NECSPS) Recovery Kit	steeleye-lkSPS-9.5.0-7075.noarch.rpm	The SPS NEC iStorage StoragePathSavior (NECSPS) Recovery Kit protects applications that use NEC iStorage StoragePathSavior v3.3 or later multipath I/O devices.
SIOS DataKeeper	steeleye-lkDR-9.5.0-7075.noarch.rpm	SIOS DataKeeper for Linux provides an integrated data mirroring capability for SPS environments enabling SPS resources to operate in shared and non-shared storage environments.
SPS WebSphere MQ Recovery Kit	steeleye-lkMQS-9.5.0-7075.noarch.rpm	The SPS for Linux WebSphere MQ Recovery Kit provides fault resilient protection for WebSphere MQ queue managers and queue manager storage locations enabling a failure on a primary WebSphere MQ server or queue manager to be recovered on the primary server or a designated

		backup server without significant lost time or human intervention.
Quorum/Witness Package	steeleye-lkQWK-9.5.0-7075.noarch.rpm	The SPS Quorum/Witness Package allows a node to get a “second opinion” on the status of a failing node acting as an intermediary to determine which servers are part of the cluster. When determining when to fail over, the Witness Server allows resources to be brought in service on a backup server only in cases where it verifies the primary server has failed and is no longer part of the cluster.
Quick Service Protection	steeleye-lkQSP-9.5.0-7075.noarch.rpm	SPS Quick Service Protection supplies functionality to easily protect OS services.
Recovery Kit for EC2	steeleye-lkECC-9.5.0-7075.noarch.rpm	The Recovery Kit for EC2 provides a mechanism to recover an Elastic IP from a failed primary server to a backup server. It also provides a mechanism to enable the IP Recovery Kit to work in multiple availability zones.
Route53 Recovery Kit	steeleye-lkROUTE53-9.5.0-7075.noarch.rpm	Route53 Recovery Kit provides a mechanism for updating Amazon Route 53 DNS information corresponding to a virtual IP address and an actual IP address information of IP resources that are in dependency relation when switching to a failed primary server to a backup server.
VMDK as Shared Storage Recovery Kit	steeleye-lkVMDK-9.5.0-7075.noarch.rpm	With the VMDK as Shared Storage Recovery Kit, VMware virtual hard disks and their file systems used as shared disks can be protected as LifeKeeper resources.

New Features of SIOS Protection Suite for Linux Version 9

Product	Feature
New in Version 9.5.0	
SAP HANA Recovery Kit	The SAP HANA Recovery Kit, providing high-availability for SAP HANA 2.0 SPS04 clusters, is now available. See the SAP HANA Recovery Kit Administration Guide for details. • If you are using SIOS Protection Suite for Linux v9.5 or later you must use the new (built-in) SAP HANA Application Recovery Kit. • The existing SAP HANA gen/app based Recovery Kit is **not** supported with v9.5.0. Users who wish to upgrade to the SIOS Protection Suite for Linux v9.5.0 **must** convert their existing SAP HANA gen/app based Recovery Kit to the new SAP HANA Recovery Kit. Refer to Upgrading from the SAP HANA Gen/App to the SAP HANA Recovery Kit for details. • SIOS will continue to support the SAP HANA gen/app based Recovery Kit

	with the 9.4.x releases until **March 31, 2022**. • The SAP HANA Recovery Kit does not support HANA v1.
LifeKeeper Core	Supports Red Hat Enterprise Linux 7.8 (Certified in July 2020) **Note:** If you are using DataKeeper, follow these steps when installing LifeKeeper.
	Supports CentOS 7.8 (Certified in July 2020) **Note:** If you are using DataKeeper, follow these steps when installing LifeKeeper.
	Supports Oracle Linux 7.8 (Certified in July 2020) **Note:** If you are using DataKeeper, follow these steps when installing LifeKeeper.
	Supports SUSE Linux Enterprise Server 12 SP5 (Certified in July 2020)
	Support VMware vSphere 7.0 (Certified in July 2020)
	Supports CentOS 8.0
	Supports Oracle Linux 8.0
	Supports Red Hat Enterprise Linux 8.1
	Supports CentOS 8.1
	Supports Oracle Linux 8.1
	The CLI has been enhanced to allow you to control LifeKeeper through the Command Line Interface. See LKCLI for details.
	Bug Fixes
PostgreSQL	Support PostgreSQL 12
	EDB Postgres Advanced Server v12.0 is supported. (Certified in July 2020)
Oracle	PDBs with Multitenant configurations can now be protected. See Configuring a Pluggable Database with Oracle Multitenant for details.
DataKeeper	DataKeeper online mirrored volumes can now be resized. See Mirror Resize for more information.
	Mirror recovery of data replication resources can now be performed in parallel.
	Added LKDR_CONNECT_NBD_DURING_RESTORE parameter. Refer to DataKeeper parameter list for details.
	Bug Fixes
Filesystem, LVM, NFS, IP, DB2, MaxDB, SAP, Sybase, Sybase ASE, Quorum/	Bug Fixes

Witness	
New in Version 9.4.1	
LifeKeeper Core	OpenJDK included with OS is installed. See Configuring the LifeKeeper GUI for details.
	Supports SUSE Linux Enterprise Server 15 SP1
	Supports Oracle Linux 7.7
	Supports CentOS 7.7
	Supports AWS Nitro system
	Supports AWS Transit Gateway
	Bug Fixes
DataKeeper	Supports NVMe devices
	Bug Fixes
VMDK as Shared Storage	LifeKeeper for Linux VMDK as Shared Storage Recovery Kit is now available. See the VMDK Shared Storage Recovery Kit Management Guide for details.
PostgreSQL	PowerGres Plus (for Linux) v10 and PowerGres on Linux v11 can be protected with the PostgreSQL Recovery Kit
	Support FUJITSU Software Symfoware Server (Postgres) V12.4 (Certified in March 2020) For the details, refer to the SPS Optional Recovery Software Requirements, PostgreSQL Recovery Kit Administration Guide > Administration.
Install, IP, MaxDB, EC2	Bug Fixes
New in Version 9.4.0	
LifeKeeper Core	Standby Node Health Check – allows the user to monitor CPU and memory utilization on the standby node and monitor the health of out-of-service (OSU) resources to detect errors on the standby node.
	Oracle Linux 7 Unbreakable Enterprise Kernel Release 5 (UEK R5) is supported.
	Red Hat Enterprise Linux 8 is supported. **Note:** Upgrading from one kernel version to another major version such as from RHEL7 to RHEL8 is NOT supported. (i.e DataKeeper resource does NOT work when upgrading from RHEL7 to RHEL8.)
	Red Hat Enterprise Linux 7.7 is supported (Certified in November 2019) **Note:** If you are using DataKeeper, follow these steps when installing LifeKeeper.
SAP	SAP-certified support of SAP S/4HANA Platform via SAP High Availability Clustering Certification S/4-HA-CLU-1.0 SAP S/4HANA 1809 Platform is now supported. SAP S/4HANA 1909 Platform is now supported. (Added support in November 2019)
	Support for Standalone Enqueue Server 2 and Enqueue Replication Server 2

	SAP Resource UI Enhancements
	Optimizations for the LifeKeeper SAP ERS Resource
MySQL	MariaDB10.3 is supported.
DB2	DB2 11.5 is supported.
PostgreSQL	FUJITSU Software Enterprise Postgres 11 is supported. (Certified in November 2019)
General maintenance	Bug Fixes
New in Version 9.3.2	
LifeKeeper Core	Red Hat Enterprise Linux 7.6 is supported. **Note:** DataKeeper asynchronous mirrors are not supported because of a Linux kernel bug.
	CentOS 7.6 is supported. **Note:** DataKeeper asynchronous mirrors are not supported because of a Linux kernel bug.
	Oracle Linux Version 7.6 is supported. **Note:** DataKeeper asynchronous mirrors are not supported because of a Linux kernel bug. **Note:** Unbreakable Enterprise Kernel Release 5 (UEK R5) is NOT supported. (i.e. DataKeeper resource does NOT work on UEK R5.)
	Linux Enterprise Server 12 SP4 is supported.
	SUSE Linux Enterprise Server 15 is supported. **Note:** Upgrading from one kernel version to another major version such as from SLES 12 to SLES 15 is NOT supported. (i.e DataKeeper resource does NOT work when upgrading from SLES 12 to SLES 15.)
	OpenJDK v10.0.2 is supported. SIOS has tested OpenJDK v10.0.2 downloaded from https://jdk.java.net/10/ with SIOS Protection Suite for Linux (SPS-L) v9.4.0. OpenJDK is compatible with SPS-L

	v9.4.0, therefore customers may use this version or any compatible version of OpenJDK with SPS-L v9.4.0. If a customer encounters an issue due to the OpenJDK version, SIOS may recommend using a newer version of OpenJDK or the OracleJDK included in SPS-L package.
Install	The -s option for saving the current setup configuration has been added to the setup command.
	RHEL7.6 also does not support DataKeeper's asynchronous mode. The warning message is output by setup.
DataKeeper	Wait For Previous Source for multi-target mirrors. For a multi-target mirror DataKeeper keeps track of the last server, aka previous source, that had the mirror in-service. When there is a failover, the bitmap from the previous source is required to keep all of the targets in-sync. DataKeeper will now automatically wait for the previous source to join the cluster before resuming replication to any target. This allows the bitmap from the previous source to be merged so that only partial resyncs are necessary.
	Unnecessary synchronization is avoided in the environment with three or mode nodes.
	Add updated messages for "wait for source" in GUI and mirror_status.
PostgreSQL	PostgreSQL 11 is supported.
	EDB Postgres Advanced Server v11 is supported.
	FUJITSU Software Enterprise Postgres 10 is supported. For the details, refer to the SPS Optional Recovery Software Requirements.
MQ	SIOS Protection Suite for Linux now supports IBM MQ 9.1
Oracle	Support Oracle 19c (Certified in August 2019).
SAP	Supports new maintenance mode feature available with SAP kernel 7.49 and above.
General Maintenance	Bug Fixes
New in Version 9.3.1	
LifeKeeper Core	Updated the OpenSSL package to 1.0.2p
	Support Red Hat Enterprise Linux 6.10
	Support CentOS 6.10
	Support Oracle Linux 6 Update 10
MySQL	Support MySQL 8.0
Oracle	Support Oracle 18c (Certified in March 2019)
Install, EC2, Route53	Bug Fix
New in Version 9.3	

LifeKeeper Core	Red Hat Enterprise Linux Version 7.5 is supported. However, DataKeeper asynchronous mirrors are not supported because of a Linux kernel bug.
	CentOS7.5 is supported. However, DataKeeper asynchronous mirrors are not supported because of a Linux kernel bug.
	Oracle Linux Version 7.5 is supported. However, DataKeeper asynchronous mirrors are not supported because of a Linux kernel bug.
	Support VMware vSphere 6.7 (Certified in October 2018)
	Bug Fixes
EC2, Route53	EC2 and Route53 RK now support HTTP Proxy.
	Bug Fixes
Quorum/Witness	Storage QWK is now supported. For details, please click here.
	Bug Fixes
Install	The SPS for Linux installation process has been upgraded. For details, please click here.
SAP, Oracle, Samba, MQ, Sybase, Filesystem, Generic Application, QSP, SAP MaxDB, DataKeeper	Bug Fix
New in Version 9.2.2	
EC2,Route53	IAM Role is now supported. • Openswan Recovery Kit does not support IAM Role. You may use v9.2.1 in case of Cross Region configuration.
DataKeeper	Support GUID Partition Table (GPT) to identify protected disks • The supported disk is SCSI Hard Disk or Xen Virtual Disk(xvd) in case of Linux kernel 2.6.27 or earlier.
PostgreSQL	Support PostgreSQL 10
	EDB Postgres Advanced Server v10.0 is now supported. (Certified in April 2018)
SAP, NAS, EC2	Bug Fix
New in Version 9.2.1	
LifeKeeper Core	Support Oracle Linux 7.4
	Support CentOS 7.4
	Support SUSE Linux Enterprise Server 12 SP3 • The kernel should be updated to 4.4.82-6.9.1 for SUSE Linux Enterprise Server 12 SP3
	The Recovery Kit for EC2, Route 53 Recovery Kit, Openswan Recovery Kit can now be

	installed from the setup menu. Openswan Recovery Kit is supported only when using with Cross Region configuration
	Bug Fixes
PostgreSQL	Support EDB Postgres Advanced Server 9.6
MQ	Support IBM MQ 9.0
New in Version 9.2	
LifeKeeper Core	Support Red Hat Enterprise Linux 7.4
	SNMP trap can be sent to multiple targets
	Bug fixes
IP	IP resources using real IP(primary IP address configured for NIC) can be created
PostgreSQL	Support PostgreSQL 9.6
	Support FUJITSU Software Enterprise Postgres 9.6 For the details, refer to the SPS Optional Recovery Software Requirements, PostgreSQL Recovery Kit Administration Guide > Administration.
MQ	Support IBM MQ 9.0 (Certified in December 2017)
MD, SAP, SAP MaxDB, Quorum/ Witness, Route53, Install	Bug fixes
New in Version 9.1.2	
LifeKeeper Core	SUSE Linux Enterprise Server 12 SP2 is supported.
	CentOS7.3 is supported.
	Red Hat Enterprise Linux Version 6.9 is supported.
	kernel of Oracle Linux Version 7.3 is supported.
	Bug fixes
PostgreSQL	Support PostgreSQL 9.6
	Support FUJITSU Software Enterprise Postgres 9.6 For the details, refer to the SPS Optional Recovery Software Requirements, PostgreSQL Recovery Kit Administration Guide > Administration.
Oracle	Oracle 12c R2 is supported.
DB2	DB2 11.1 is supported.
IP, QSP, MySQL, NFS	Bug fixes
New in Version 9.1.1	
LifeKeeper Core	SUSE Linux Enterprise Server 12 SP1 support. • SLES12.0 is not supported. • Btrfs is not supported.
	Red Hat Enterprise Linux Version 7.3 support.
	Oracle Linux Version 7.3 support. • UEK is not supported.

	vSphere 6.5 support.
	Bug Fixes
PostgreSQL	PostgreSQL 9.5 support EDB Postgres Advanced Server v9.5 support FUJITSU Software Symfoware Server (Open Interface) V12.2 support FUJITSU Software Symfoware Server (Postgres) V12.3 support FUJITSU Software Enterprise Postgres 9.5 support For the details, refer to the SPS Optional Recovery Software Requirements, PostgreSQL Recovery Kit Administration Guide > Administration.
Sybase ASE	Sybase ASE 16.0 support.
MySQL	MySQL 5.7 support on RHEL 7.x/CentOS 7.x/OEL 7.x. • MySQL 5.7 on other OS is already supported.
SAP	SAP 7.5 support.
New in Version 9.1.0	
LifeKeeper Core	Red Hat Enterprise Linux 6.8 support (Certified in September 2016). CentOS 6.8, Oracle Linux 6.8 support (Certified in September 2016). • MD Recovery Kit is not supported on these OS.
	LifeKeeper API for Monitoring Added API to supply LifeKeeper status and log information.
	Quick Service Protection support Added functionality to easily protect OS services.
	Bug Fixes
New in Version 9.0.2	
LifeKeeper Core	Support of Red Hat Enterprise Linux Version 7.2. • SQL RK is not supported when running on RHEL 7.x/CentOS 7.x/OEL 7.x.
	Updated OpenSSL package to version to 1.0.1q
	Bug Fixes
MQ	Added support for Multi-version WebSphere MQ. With this support queue managers for 7.1, 7.5, and 8.x can all be protected on the same cluster node.
	Removed the Recovery Kit restriction that only the mqm user could be used for running MQ commands. With this change any user in the mqm group can be used by the Recovery Kit to run MQ commands.
	Bug Fixes
IP, Filesystem, DMMP, DataKeeper, EC2, PostgreSQL, Power Path, SAP, SAP DB/ MaxDB, Oracle	Bug Fixes.
Licensing	Update to a newer version of FlexNet
New in Version 9.0.1	
LifeKeeper Core	Bug Fixes.
DataKeeper	Bug Fixes.

New in Version 9.0	
LifeKeeper Core	Red Hat Enterprise Linux Version 6 Update 7 support. (Certified in October 2015)
	Community ENTerprise Operating System (CentOS) Version 6 Update 7 support. (Certified in October 2015)
	Oracle Linux Version 6 Update 7 support. (Certified in October 2015)
	SUSE LINUX Enterprise Server 11 SP4 support. (Certified in October 2015)
	Chef support
	Added SPS for Linux Parameters List, document detailing tunable values. Added the lkchkconf command.
	vSphere 6 support
	reiserfs filesystem type is no longer supported.
	Arks supported with Red Hat Enterprise Linux Version 7.0/7.1, Community ENTerprise Operating System (CentOS) Version 7.0/7.1, and Oracle Linux Version 7.0/7.1 are the same as LifeKeeper for Linux v8.4.1. (Arks to be supported are: PostgreSQL, MySQL, Oracle, DB2, Apache, Postfix, DMMP, LVM, NFS, NAS, Samba, MD, EC2, Route53, Openswan)
	Bug Fixes.
DataKeeper	The DK rewind feature is no longer supported in version 9. Prior to upgrading to version 9, you will need to deactivate all rewind configuration settings, and perform any necessary archival of data.
	Bug Fixes.
GUI	JRE 8u51 support. (JRE 7 is no longer supported.)
	Chrome Browser is no longer supported.
	Bug Fixes.

Bug Fixes

The following is a list of the latest bug fixes and enhancements.

Bug	Description
PL-114	Added a timer to the restore functionality in the MaxDB ARK
PL-118	NFS quickCheck fails to detect matching export option list if an option is listed more than once
PL-756	Enhanced resync checking to make sure devices are not mounted on the target
PL-1215	Enhanced quickCheck for NFS resources to monitor fsid
PL-1222	Fixed an issue where “Repository named ‘*tmpRPMcache*’ already exists” message was output and setup failed in SLES environment
PL-2300	The LifeKeeper logrotate configuration does not reload rsyslog, which results in lost LK logging after log rotation
PL-2310	Avoid using Sybase “shutdown with nowait”
PL-2315	Fixed LVM ARK cache to automatically update

PL-2320	Repaired DB2 ARK to create DB2 resource if character code is set to IBM-943
PL-3083	Changed LKDR default speed limit from 50k to 500k
PL-3135	Using confirm failover can result in a hierarchy being out of service
PL-3149	Added XFS sanity check
PL-3186	Fixed an issue where quickCheck resulted in an error instead of recovering correctly with local recovery of SAP resources
PL-3232	Fixed an issue where resources could not be created if Sybase was installed outside of /opt/sybase
PL-3292	Mirror create needs to protect against device reordering
PL-3415	Fixed a duplicate message ID
PL-3518	Stopping LifeKeeper needs to clean up in-progress tasks such as machine failover, quorum_isp, and quickCheck daemons
PL-3546	Quorum is inefficient when attempting to restore resources
PL-3760	Use “ip” command to bring up network interfaces.
PL-4051	Fixed an issue where Quorum would not launch resources if there were ISP resources
PL-4164	lkstop fails if the process of removing all the resources takes more than 90 seconds
PL-4355	Fixed an issue where LifeKeeper also stopped if the rsyslog service stopped on systems employing systemd
PL-4575	Fixed an issue where after switching over IP resources on RHEL 8 the virtual IP address could not be pinged from the previous host
PL-4746	When creating a new DataKeeper resource, the configuration using DEVNAME for disk identification is no longer supported. Use GPT partition (GUID Partition Table) configuration. Before upgrading to SPS for Linux 9.5.0 please review the following Known Issue to see if this impacts your cluster.

Discontinued Features

Feature	Description
Discontinued in Version 9.5.0	
LifeKeeper Core	System log management using syslog-ng is no longer supported. Please use rsyslog.
	SUSE Linux Enterprise Server (SLES) 11.0 to SP4 is no longer supported.
DataKeeper	Environments that use DEVNAME for disk identification using DataKeeper for Linux (DK resources) are no longer supported. Please use a GPT partition (GUID Partition Table).
Oracle	Oracle Database Enterprise Edition 11g R2 is no longer supported.
	Oracle Database Standard Edition 11g R2 is no longer supported.
	Oracle Database Standard Edition One 11g R2 is no longer supported.
MySQL	MariaDB 5.5, 10.0 is no longer supported.
PostgreSQL	PostgreSQL 9.4 is no longer supported.
	EnterpriseDB Postgres Plus Advanced Server 9.4 is no longer supported.

Discontinued in Version 9.4.1	
	None
Discontinued in Version 9.4.0	
DataKeeper	Multi-Site Cluster Feature

System Requirements

SPS Product Requirements

SPS for Linux is supported on any Linux platform that satisfies the minimum requirements included in the Linux Configuration Table. Also refer to the SPS Support Matrix for supported operating systems, applications and virtualization.

Note: SPS on a Linux server will not inter-operate with SPS for Windows.

Description	Requirement
Linux Operating System	See the Linux Configuration Table for specific operating system information.
Virtual Environments	The guest operating system running on the virtual machine must be one of the supported versions listed in the Linux Configuration Table. The following virtual environment is an example where SIOS Protection Suite for Linux is deployed. Please refer to the Support Matrix for detailed versions of supported virtualization environments. • KVM • Oracle VM Server for x86 • VMware vSphere v5.5, v6.0, v6.5, v6.7 and v7.0 • Amazon EC2 • Microsoft Azure • Nutanix Acropolis Hypervisor vSAN configuration is supported for vSphere 6.5 or later except RDM which is not supported by VMWare. Fibre channel SAN and shared SCSI cluster configurations are not supported with SPS for Linux running in a KVM and Oracle VM Server for x86 virtual machine.

	Note: Some Amazon EC2 configurations have issues when the Shutdown Strategy is set to “Do not Switchover Resources”. For detailed information, see Troubleshooting > Known Issues and Restrictions. **Note:** On SLES v12 or later running on AWS or Azure, the dynamic change of the virtual IP address by the cloud network plug-in may affect the operation of the LifeKeeper cluster. For detailed information, see LifeKeeper Core – Known Issues / Restrictions.
Memory	The minimum memory requirement for a system supporting SPS is 512 MB. This is the minimum amount required by SPS supported Linux distributions. System memory should be sized for the applications that will be running on the SPS protected system as well. Refer to Application Configuration for further information.
Disk Space	The SPS Package Cluster requires the following disk space: /opt – approx 100MB (depending on kits installed) / – approx 110MB
Java Runtime Environment	• OpenJDK 1.8, 10 or later

SPS Optional Recovery Software Requirements

The following table shows the software requirements for the optional SPS recovery software.

See Application Configuration for additional requirements and/or restrictions that may apply to applications under SPS protection.

Product	Requirement(s)
SPS Apache Web Server Recovery Kit	Apache Web Server v2.4
SAP Recovery Kit	SAP NetWeaver 7.0 including Enhancement Package 1,2 and 3 SAP NetWeaver 7.3 including Enhancement Package 1 SAP NetWeaver 7.4 SAP NetWeaver 7.5 SAP NetWeaver AS for ABAP 7.51 innovation package
SPS SAP MaxDB Recovery Kit	SAP MaxDB v7.9 LifeKeeper v6 or later Core Package Cluster
SPS Postfix Recovery Kit	Postfix software provided with the supported Linux distributions installed and configured on each server. The same version of Postfix should be installed on each server.

	LifeKeeper v6 or later Core Package Cluster
SPS Oracle Recovery Kit	Oracle Database Enterprise Edition v12c, v12c R2, v18c and v19c (excluding ASM and pluggable database) Oracle Database Standard Edition 2 (SE2) v12c, v12c R2, v18c and v19c (excluding ASM and pluggable database)
SPS DB2 Recovery Kit	IBM Db2 Universal Database v10.5, v11.1 IBM Db2 Enterprise Server Edition (ESE) v10.5, v11.1 and v11.5 IBM Db2 Workgroup Server Edition (WSE) v10.5, v11.1 and v11.5 IBM Db2 Express Edition v10.5, v11.1 and v11.5 LifeKeeper v6 or later Core Package Cluster SPS NFS Server Recovery Kit v5.1 or later (for DB2 EEE and DB2 ESE with multiple partitions only)
SPS MySQL Recovery Kit	MySQL and MySQL Enterprise v5.7 and v8.0 MariaDB v10.3 and v10.4
SPS PostgreSQL Recovery Kit	PostgreSQL v9.5, v9.6, v10, v11 and v12 EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server v9.5, v9.6, v10.0, v11.0 and v12.0 PowerGres Plus (for Linux) v10 PowerGres on Linux v11 The following edition of FUJITSU Software Symfoware Server. Symfoware Server V12.2 • Symfoware Server (Open Interface) V12.2 Enterprise Edition • Symfoware Server (Open Interface) V12.2 Standard Edition Symfoware Server V12.3 • Symfoware Server (Postgres) V12.3 Enterprise Edition • Symfoware Server (Postgres) V12.3 Standard Edition • Symfoware Server (Postgres) V12.3 Lite Edition Symfoware Server V12.4 • Symfoware Server (Postgres) V12.4 Enterprise Edition

	• Symfoware Server (Postgres) V12.4 Standard Edition The following edition of FUJITSU Software Enterprise Postgres 9.5 • FUJITSU Software Enterprise Postgres 9.5 Advanced Edition • FUJITSU Software Enterprise Postgres 9.5 Standard Edition The following edition of FUJITSU Software Enterprise Postgres 9.6 • FUJITSU Software Enterprise Postgres 9.6 Standard Edition The following edition of FUJITSU Software Enterprise Postgres 10 • FUJITSU Software Enterprise Postgres 10 Advanced Edition • FUJITSU Software Enterprise Postgres 10 Standard Edition • FUJITSU Software Enterprise Postgres 10 Community Edition The following editions of FUJITSU Software Enterprise Postgres 11 • FUJITSU Software Enterprise Postgres 11 Advanced Edition • FUJITSU Software Enterprise Postgres 11 Standard Edition • FUJITSU Software Enterprise Postgres 11 Community Edition
SPS Sybase ASE Recovery Kit	Sybase ASE 15.7 and 16.0
SPS Samba Recovery Kit	Standard Samba file services provided with the supported Linux distributions
SPS NFS Server Recovery Kit	Linux kernel version 2.6 or later The NFS Server and client packages must be installed on SLES systems. NFSv2 is not supported on Red Hat Enterprise Linux 7 or later, CentOS 7 or later, Oracle Linux 7 or later. NFS over UDP is not supported on Red Hat Enterprise Linux 8 and later. Some environments may require additional configurations. Refer to NFS Specific Configuration Considerations.
SPS Network Attached Storage Recovery Kit	NFS version of Mounted NFS file systems from an NFS server or Network Attached Storage (NAS) device v2, v3 and v4
(blank)SPS Logical Volume Manager (LVM) Recovery Kit	Linux Logical Volume Manager (LVM) Version 1 or 2 volume groups and logical volumes

(blank)SPS Software RAID (md) Recovery Kit	Software RAID devices based on md **Note:** The MD Recovery Kit cannot be used in conjunction with SIOS DataKeeper.
EMC PowerPath	PowerPath for Linux v5.3 or later The sg3_utils Sg3_utils package must be installed.
Device Mapper Multipath (DMMP)	The device-mapper-multipath package attached to the operating system. The sg3_utils package must be installed.
Hitachi Dynamic Link Manager Software (HDLM)	Please see Hitachi Dynamic Link Manager Software Multipath I/O Configurations and Linux Distribution Requirements. The sg3_utils package must be installed.
NEC iStorage Storage Path Savior (NECSPS)	iStorage StoragePathSavior for Linux v3.3 or later For the supported Linux kernel and distribution, please refer to the support information of StoragePathSystem for Linux. The sg3_utils package must be installed on Red Hat and SLES. LifeKeeper v6 or later Core Package Cluster
WebSphere MQ Resources	IBM MQ v8.0, v9.0 and v9.1. See Known Issues and Restrictions > Installation.
Quorum/Witness Package	All nodes which will participate in a quorum/witness mode cluster, including witness-only nodes, should be installed with the Quorum/Witness Server Support Package for SPS.

Open Source Packages

The following open source packages are included in the LifeKeeper installation image.

Name	Version	License Type and Version
curl-7.21.7-3.2	7.21.7	MIT
libcurl-7.212.7-3.2	7.21.7	MIT
gnutls-2.8.6-3.1	2.8.6	GPLv3+ and GPLv2+
gnutls-utils-2.8.6-3.1	2.8.6	GPLv3+
libgcrypt-1.5.0-2.1	1.5.0	LGPv2+
libgpg-error-1.10-2.1	1.1	LGPv2+
libvirt-0.9.3	0.9.3	LGPLv2+
libxml2-2.7.8-7.1	2.7.8	MIT
libxml2-static-2.7.8-7.1	2.7.8	MIT
lighttpd-1.4.41-1	1.4.41	BSD
lighttpd-fastcgi-1.4.41-1	1.4.41	BSD

openjdk-12.0.2	12.0.2	GPLv2+
java-1_8_0-openjdk-1.8.0.222-351.4	1.8.0	GPLv2+
java-1_8_0-openjdk-headless-1.8.0.222-351.4	1.8.0	GPLv2+
openssl-1.0.2p-1.1	1.0.2p	BSDish
openssl-perl-1.0.2p-1.1	1.0.2p	BSDish
pcre-4.5-2.1	4.5	distributable
pdksh-5.2.14	5.2.14	GPL, distributable
perl-5.8.8-8.2	5.8.8	Arrtistic or GPL
perl-addons-5.8.8-24.1	5.8.8	Various(GPL, artistic v2, BSD, Open Market, GPLv2+, MIT)
powercli-11.5.0-1	11.5.0	Various
powershell-6.2.3-1	6.2.3	MIT
readline-4.3-14.1	4.3	GPL
runit-2.0.0-4.11	2.0.0	BSD
util-linux-2.31.1-2	2.31.1	GPLv2 and GPLv2+ and LGPLv2+ and BSD with advertising and Public Domain
xenserver-5.6.100	5.6.100	LGPL
Perl Config::IniFiles (CPAN module)	2.27	GPL/Artistic (Same as Perl)
HADR-CentOS-2.6.32	2.6.32	GPLv2
HADR-CentOS-3.10.0	3.10.0	GPLv2
HADR-CentOS-4.18.0	4.18.0	GPLv2
HADR-RHAS-2.6.32	2.6.32	GPLv2
HADR-RHAS-3.10.0	3.10.0	GPLv2
HADR-RHAS-4.18.0	4.18.0	GPLv2
HADR-OEL-2.6.32	2.6.32	GPLv2
HADR-OEL-3.10.0	3.10.0	GPLv2
HADR-OEL-4.18.0	4.18.0	GPLv2

Client Platforms and Browsers

The SPS web client can run on any platform that provides support for Java Runtime Environment JRE8 update 51. The currently supported browsers are Firefox (Firefox 51 or earlier) and Internet Explorer running on Linux, Windows Server 2008 R2, Windows Server 2012 R2, Windows Server 2016, Windows 7, Windows 8 or Windows 10 with JRE8 update 51. Other recent platforms and browsers will likely work with the SPS web client, but they have not been tested by SIOS Technology Corp. In addition, particular features of each browser have not been tested.

You should specify all the hostnames and addresses in your cluster in the client machine's local hosts file (usually */etc/hosts* or *C:\windows\system32\drivers\etc\hosts*). This minimizes the client connection

time and allows the client to connect even in the event of a Domain Name Server (DNS) failure.

Installation and Configuration

See the SIOS Protection Suite Installation Guide for complete installation and configuration information.

Upgrades

LifeKeeper can be upgraded to Version 9.4.x from either LifeKeeper Version 9.2.x or Version 9.3.x. If upgrading from a version other than 9.2.x or 9.3.x, the older version will need to be uninstalled and SIOS Protection Suite for Linux will have to be reinstalled. An alternative to uninstalling the older version would be to upgrade from the older version to 9.2.x or 9.3.x, then perform the upgrade to 9.4.x.

Storage and Adapter Options

For a list of the disk array storage models and adapters currently supported by SPS in shared storage configurations as well as their type of certification, see the Storage and Adapter Options topic. Details about driver versions and other configuration requirements for these arrays and adapters are listed in the Storage and Adapter Configuration topic.

Technical Notes

We strongly recommend that you read the Technical Notes section concerning configuration and operational issues related to your SPS environment.

Known Issues

See Known Issues and Restrictions in the Troubleshooting section of SIOS Protection Suite for Linux Technical Documentation and the DataKeeper Troubleshooting section..

3. SIOS Protection Suite for Linux Getting Started Guide

This document will guide you through the installation of the SIOS Protection Suite for Linux (SPS) and assumes the user has basic knowledge of the Linux operating system. Please refer to the SIOS Protection Suite for Linux product documentation for more information.

Pre-Installation Requirements

Before installing SPS for Linux, please check the following:

- **SPS for Linux Release Notes** -The Release Notes include supported platforms, operating systems, applications, and storage. They also include the latest features and Bug Fixes.
- **TCP/IP Connection and Name Resolution** – In order to use the GUI function, both cluster nodes need to be able to resolve the name. Use the DNS service or */etc/hosts* for name resolution. Also, localhost needs to be resolved to 127.0.0.1.
- **Firewall** – The following ports are used:
  - Communication Path (TCP): 7365/tcp
  - Communication of a GUI Server: 81/tcp、82/tcp
  - RMI Communication between the GUI Server and Client: all the ports after 1024/tcp
  - Synchronization of DataKeeper (when using DataKeeper): “10001+<mirror number>+<256 * i>”

More Firewall Information

- The port used for communication with the GUI server and a client needs to be open on the cluster node where SPS is installed and on all systems where the GUI client runs.
- The ports used by DataKeeper can be calculated using the formula above. The value of i starts at 0 and uses an unused port when found. For example, in an environment where a DataKeeper resource with mirror number 0 exists, if port 10001 is being used by another application, port 10257 will be used.
- For communication between the GUI server and a client, Java RMI (Remote Method Invocation) randomly uses ports 1024 and above. When applying access control etc. to a cluster system, packet filtering needs to be performed considering these ports. If this specification is an issue from a security standpoint, you can use ssh X forwarding. Please refer to the Technical Documentation for the setting details.

- ◦ **Check the SELinux Setting** – When the SELinux setting is enabled, SPS for Linux cannot be installed. Please refer to the OS distribution documentation on how to disable SELinux. It is not recommended to use SELinux permissive mode unless it is required in an SAP environment. Please make sure that the application to be run on the cluster supports a permissive mode. SELinux permissive mode has been tested for following ARKs: SAP / SAP MaxDB / Sybase / Oracle / DB2 / NFS / DataKeeper / NAS / EC2 / IP / FileSystem / MQ
  Refer to Linux Dependencies for required packages.
  - Install the appropriate package provided by your distribution.
  - The sg3_utils package is required for environments using recovery kits for Multipath such

as the DMMP Recovery Kit and the PowerPath Recovery Kit. This is not required for environments where recovery kits for Multipath are not used.

- ◦ **Check Known Issues** – Please make sure that there are no known issues for your environment.

Installing SPS for Linux

Note: These installation instructions assume that you are familiar with the Linux operating system installed on your servers.

Install the SPS software on each server in the SPS configuration.

Packages that LifeKeeper is dependent on are installed automatically because the LifeKeeper installation setup script uses package manager tools (yum or zypper) to ensure installation of all dependent packages.

IMPORTANT: A functional yum or zypper configuration is required for the successful installation of LifeKeeper. A non-functional configuration can result in an installation failure (see Installation Known Issues for more information). Additionally, the package manager repo or rpm database must not be locked as that could cause the install to hang. If the dependent packages cannot be installed automatically via the package manager, refer to Linux Dependencies and install the necessary packages in advance.

The SPS for Linux image file (sps.img) provides a set of installation scripts designed to perform the user interactive system setup tasks that are necessary when installing SPS on your system.

A licensing utilities package is also installed providing utilities for obtaining and displaying the Host ID or Entitlement ID of your server. Host IDs and/or Entitlement IDs are used to obtain valid licenses for running SPS.

IMPORTANT:
• Installing SPS on your shared storage is not supported. Each server should have its own copy installed on its local disk.
• All SPS packages are installed in the directory /opt/LifeKeeper.

Obtaining and Installing the License

SPS for Linux requires a unique license for each server. The license is a run-time license, which means that you can install SPS without it, but the license must be installed before you can successfully start and run the product.

Note: If using newer hardware with RHEL 6.1, please see the IP Licensing Known Issues in the SPS for Linux Troubleshooting Section.

The Installation script installs the Licensing Utilities package which obtains and displays all of the available Host IDs for your server during the initial install of your SIOS Protection Suite Software. Once your licenses have been installed the utility will return the Entitlement ID if it is available or the Host IDs if it is not.

Note: Host IDs, if displayed will always be based on the MAC address of the NICs.

The new licenses obtained from the SIOS Technology Corp. Licensing Operations Portal will contain your Entitlement ID and will be locked to a specific node or IP address in the cluster. The Entitlement ID (Authorization Code) which was provided with your SIOS Protection Suite Software, is used to obtain the permanent license required to run the SIOS Protection Suite Software. The process is illustrated below.

Note: Each software package requires a license for each server.

Perform the following steps to obtain and install your license(s) for each server in the SPS cluster:

1. **Ensure you have your LifeKeeper Entitlement ID** (Authorization Code). You should have received an email with your software containing the Entitlement ID needed to obtain the license.
2. **Obtain your licenses from the SIOS Technology Corp. Licensing Operations Portal**.
   - a. Using the system that has internet access, log in to the SIOS Technology Corp. Licensing Operations Portal.
   b. From the **Activation & Entitlements** dropdown list select **List Entitlements**.

Note: If changing password, use the **Profile** button in the upper right corner of the display.

- c. Find your

Entitlement ID and select each **Activation ID** associated with that Entitlement ID by checking the box to the left of the line item.
d. From the **Action** dropdown list select **Activate**.
e. Define the required fields and select **Next**.
f. Click on the **Green Plus Sign** to add a new host.
g. Select and Define the required fields and click **Okay**. (**Note:** Internet = IP address, Ethernet = MAC address)
h. Check the box to the left of the **Host ID** or **IP address** and select **Generate**. The **Fulfillment ID** will display on the **License Summary** screen.
i. Select **Complete**.
j. Check the box to the left of the **Fulfillment ID** and select the **Email** from the View dropdown list.
k. Enter a valid email address to send the license to and select **Send**.
l. Retrieve the email(s).
m. Copy the file(s) to a temporary directory on each node. **Make sure that the licenses match the MAC address**. This path and filename(s) will be used during the 'Install License Key' portion of the setup script.

NOTE: To install the license outside of the 'Setup' script, copy the license file(s) to `/var/LifeKeeper/license` on each system, or run `/opt/LifeKeeper/bin/lkkeyins` and specify the filename (including full path) to the file.

How to Install / Upgrade SPS Using the Setup Script

To install or upgrade SPS, follow the steps below.

Interactive Mode

1. After logging in as the root user, use the following command to mount the sps.img file:

   ```
   mount <PATH/IMAGE_NAME> <MOUNT_POINT> -t iso9660 -o loop
   ```

   Where PATH is the path to the image
   IMAGE_NAME is the name of the image
   MOUNT_POINT is the path to mount location

2. Change to the directory where sps.img is mounted and enter the following:

   ```
   ./setup
   ```

3. The script collects information about the system environment and determines what you need to do to install SPS.

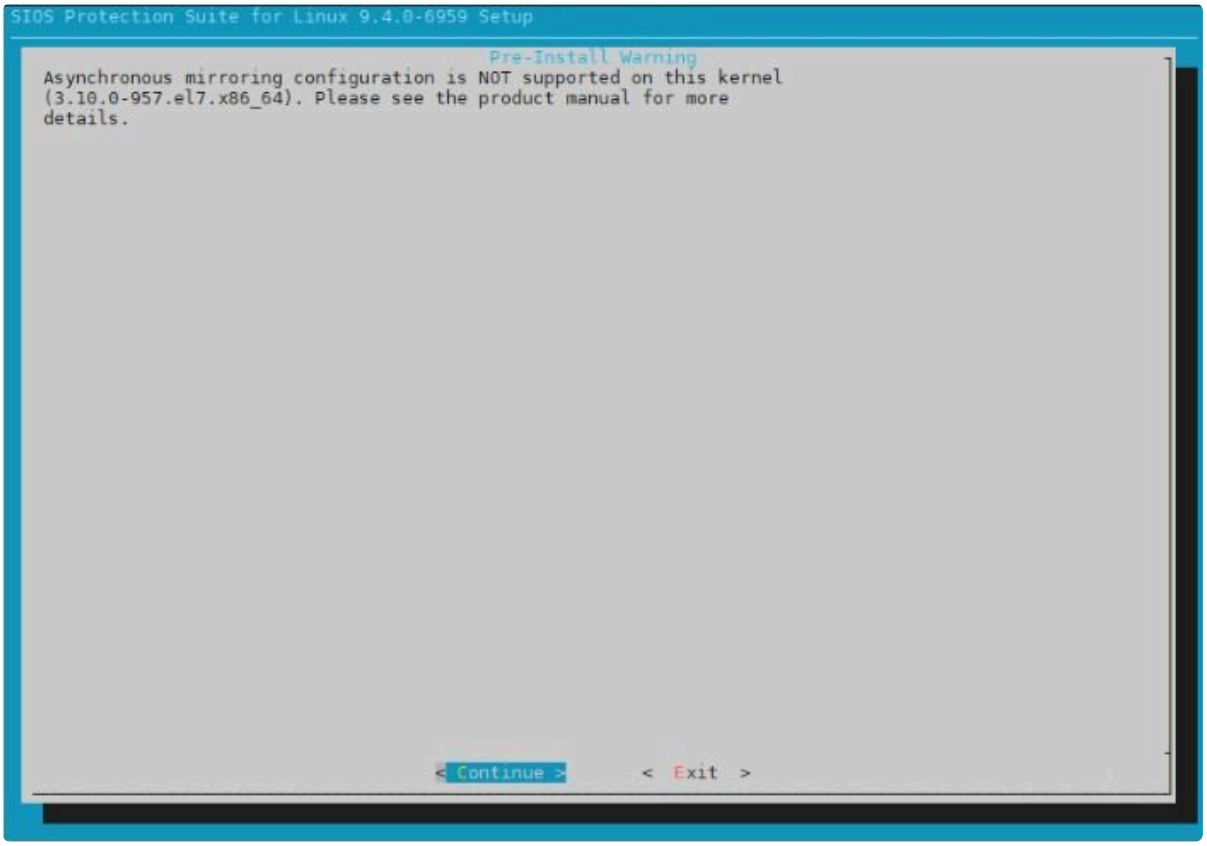
If the system requirements for installation or upgrade are not satisfied, then an error message is displayed and the installation / upgrade is cancelled.

Also, if some restrictions arise or a configuration change is required, a warning message will be displayed requiring the user to decide whether to continue or abort the installation.

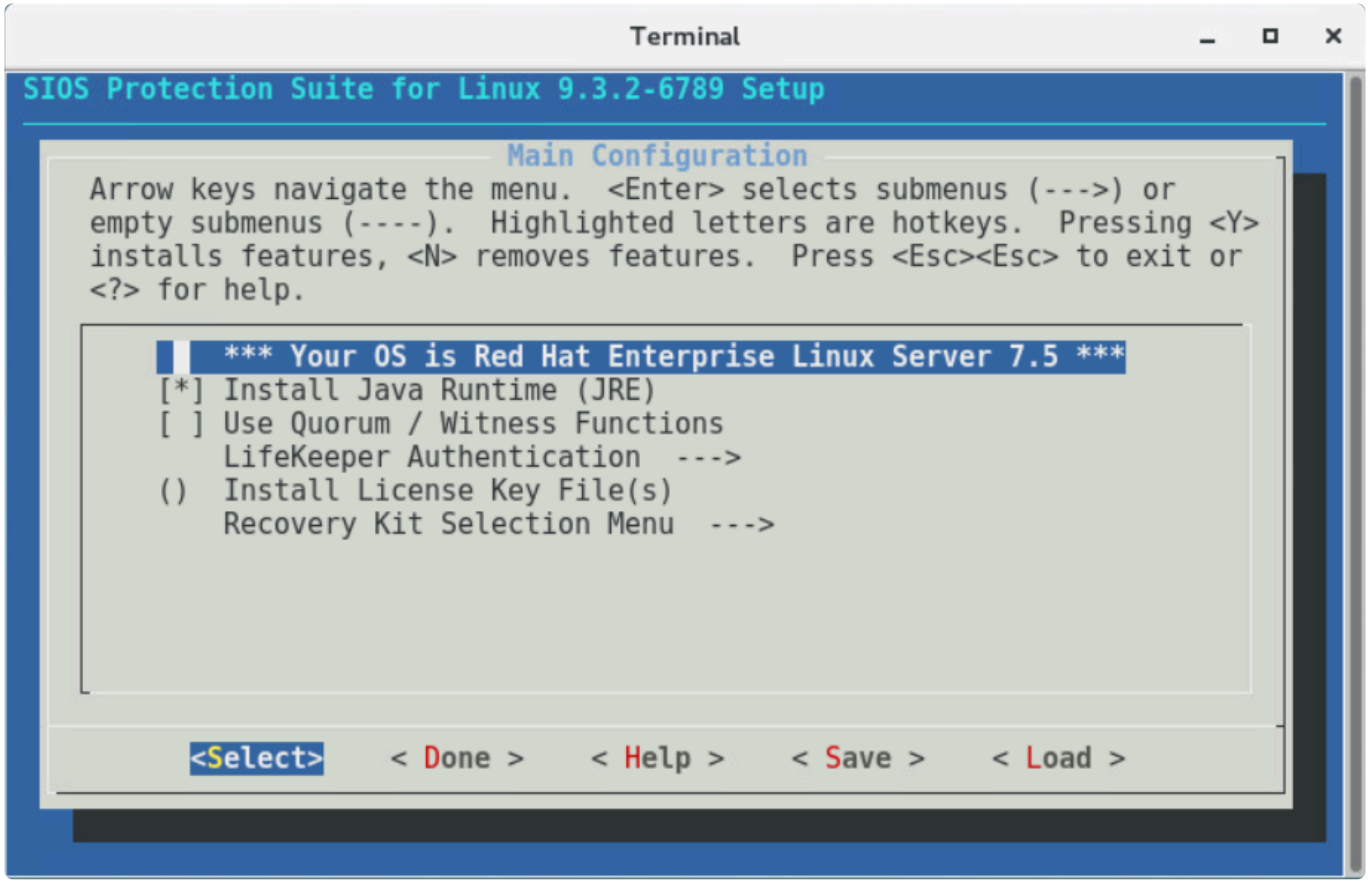
4. Select the SPS features and Application Recovery Kits (ARKs) to install via the main dialog screen.

How to Use the Dialog Screen

If the kernel version is not supported for asynchronous mirroring the following dialog will appear.



The dialog screen is displayed below.



Use the following keys to navigate the menu.

↑ ↓	Navigate between menu items
← →	Navigate between the menu buttons at the bottom of the screen
ENTER	Open the selected sub menu
Y / N / SPACE	Turn on, turn off or invert the selected item

The menu buttons at the bottom of the screen are used for the following operations.

Select	Opens a sub menu dialog screen
Done	Closes the current screen and returns to the previous screen. Selecting this button on the main screen completes the configuration.
Help	Displays help for the highlighted item
Save	Saves the current settings in a configuration file. The saved configuration file can be used for non-interactive installations.
Load	Loads a saved configuration file

The “Save” and “Load” menu buttons display a dialog screen asking for a configuration file name for use in saving the current configuration or for loading a saved configuration. If you want to change the default file name provided, move to the file name field using the [TAB] key, and enter a new name. **Note:** The “Save” operation will prompt for confirmation before overwriting a file with the same name.

The items listed below are configurable during installation. During an upgrade only items that can be

configured are listed. Using the hotkey <Z> will show those items that will remain unchanged during the upgrade.

- **Install Java Runtime (JRE)**

  Install the Java runtime environment used by the LifeKeeper GUI.

- **Restart NFS Service**

  When configuring High Availability NFS, restarting the NFS services is required. When this is selected, the services are restarted automatically after the configuration is completed.

  Note: If you do not want to restart the NFS services automatically, a restart will need to be done to pick up the configuration changes before using the NFS Recovery Kit.

- **Use Quorum / Witness Functions**

  Use Quorum / Witness for I/O fencing. For details, please refer to Quorum/Witness in the technical documentation.

  The Quorum/Witness Server Support Package for LifeKeeper will need to be installed on every node in the cluster that uses quorum/witness functionality, including a witness-only node. The only configuration requirement for the witness node is to create appropriate comm paths. When using a quorum mode with tcp_remote, LifeKeeper does not need to be installed on the host which was set as `QUORUM_MODE in /etc/default/LifeKeeper` configuration file.

  The general process for setting up quorum/witness functionality will involve the following steps:

-

1. Set up the server and make sure that it can communicate with other servers.
2. Install LifeKeeper on the server. During the installation, enable “Use Quorum / Witness functions” with the setup command and install the quorum/witness package as well.
3. Create appropriate communication paths between the nodes including witness-only nodes.
4. Configure quorum/witness.

   When the above steps are completed, the quorum/witness functions will be activated in the cluster and quorum checking and witness checking will be performed before failovers are allowed.

- **LifeKeeper Authentication**

  Specify the users allowed to log in to the SPS for Linux GUI along with their privilege levels. Multiple user accounts can be specified by separating them with blanks. For details, refer to GUI User Settings.

- **Install License Key File(s)**

Install the licenses required to start SPS for Linux by entering the path name of the license file to install. Multiple files can be specified by separating them with spaces.

- **Recovery Kit Selection**

  Select the Application Recovery Kits to install.

  Application Recovery Kits are broken into several categories based on common functionality.

  Please refer to Categories for Application Recovery Kits for details.

- **LifeKeeper Startup After Install**

  When selected, SPS for Linux will be started when the installation is completed.

> **Note:** Because the SPS-L Data Replication package may install kernel modules for some of the supported OS distributions, a re-install of SPS-L may be required when the kernel is upgraded. This applies to OS distributions for RedHat, CentOS and Oracle Enterprise Linux (non-UEK kernels only) running kernel versions 3.10.0-514 or later.

Categories for Application Recovery Kits

Category	Description
Application Suite	A group of recovery kits that protect applications such as SAP and IBM MQ.
Networking	A group of recovery kits that protect network services in the cloud such as EC2 and Route53.
Database	A group of recovery kits that protect database applications, including, but not limited to, Oracle, PostgreSQL, and MaxDB.
File Sharing	A group of recovery kits that protect file sharing services such as NFS and Samba.
Mail Server	A group of recovery kits that protect email services such as Postfix.
Storage	A group of recovery kits that protect data storage methods, including, but not limited to, DataKeeper (replication), Device Mapper (DM) Multipath (DMMP), and Network Attached Storage (NAS).
Web Server	A group of recovery kits that protect web services such as Apache.

5. Once all the required SPS features and ARKs have been selected, select <Done> to begin the installation.

   If any notifications are output when the installation completes, please take the necessary actions to correct them.

Creating a Cluster

LifeKeeper must be installed on all systems before creating a cluster.

To create a cluster system first you need set up a “communication path” between the nodes that make up a HA cluster. Then create “resources” to define what to protect.

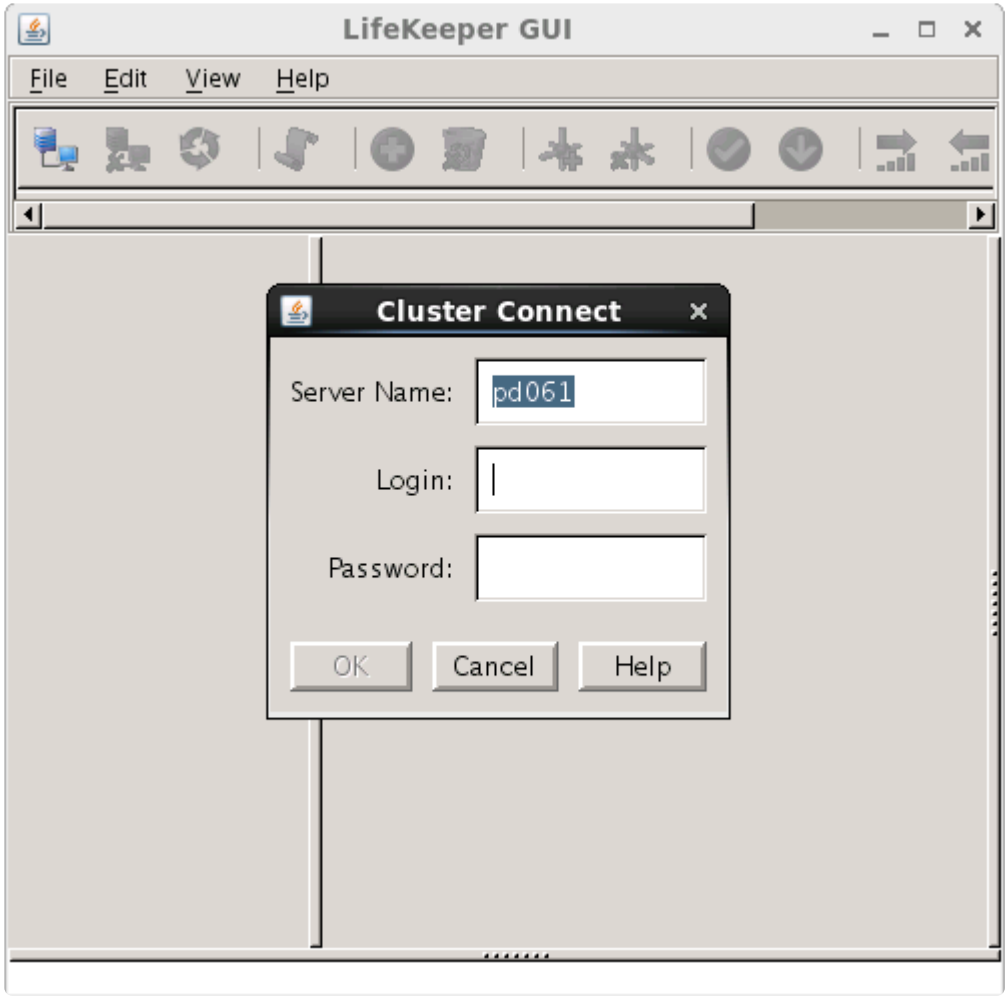
Connecting with LifeKeeper GUI Client

Configure LifeKeeper using the GUIs.

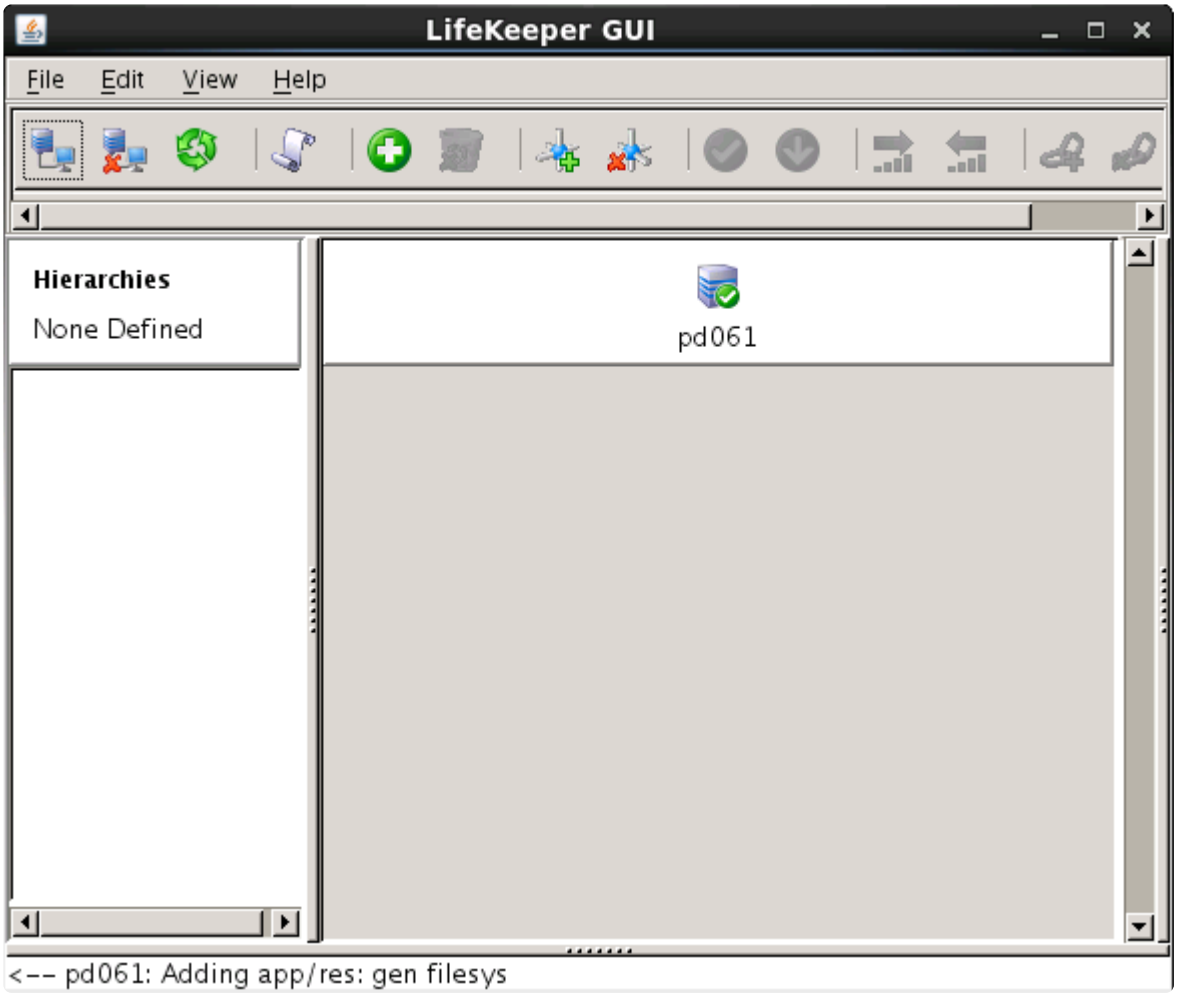
The GUI client is started by lkGUIapp command. After starting LifeKeeper, start the LifeKeeper GUI client with the following command.

```
# lkGUIapp
java version "1.8.0_51"
Java(TM) SE Runtime Environment (build 1.8.0_51-b16)
Java HotSpot(TM) 64-Bit Server VM (build 25.51-b03, mixed mode)
Setting up secure random number generator
Random number setup completed
```

After executing the command, the GUI client is started and the login screen is launched. Server Name is the name of the server you are running. For login username and password, enter the LifeKeeper admin user name and password. By default, the operating system super user (root) and its password are used for the admin user.



After successfully logging in the following screen is displayed.



Note: You can also access the GUI from a remote host via a web browser. When using a web browser, names should be resolved between the remote host you want to access and the cluster server. Use Port 81 for a web browser. Access from the remote host by entering http://host name:81 or http://IP address:81. Please note that there are several requirements for using the LifeKeeper GUI with a browser. Refer to the Release Notes and the Technical Documentation for details. Due to this requirement, browser operation may be difficult depending on the network environment. In such cases, consider using remote desktop and SSH X forwarding.

Creating a Communication Path

To create a communication path between a pair of servers, you must define the path individually on both servers. LifeKeeper allows you to create both TCP (TCP/IP) and TTY communication paths between a pair of servers. Only one TTY path can be created between a given pair. However, you can create multiple TCP communication paths between a pair of servers by specifying the local and remote addresses that are to be the end-points of the path. A priority value is used to tell LifeKeeper the order in which TCP paths to a given remote server should be used.

IMPORTANT: Using a single communication path can potentially compromise the ability of servers in a cluster to communicate with one another. If a single comm path is used and the comm path fails, LifeKeeper hierarchies may come in service on multiple servers simultaneously. This is known as “false failover”. Additionally, heavy network traffic on a

> TCP comm path can result in unexpected behavior, including false failovers and LifeKeeper initialization problems.

1. On the global toolbar, click the **Create Comm Path** button.
2. A dialog entitled **Create Comm Path** will appear. For each of the options that follow, click **Help** for an explanation of each choice.
3. Select the **Local Server** from the list box and click **Next**.
4. Select one or more **Remote Servers** in the list box. If a remote server is not listed in the list box (i.e. it is not yet connected to the cluster), you may enter it using **Add**. You must make sure that the network addresses for both the local and remote servers are resolvable (for example, with DNS or added to the */etc/hosts* file). Click **Next**.
5. Select either **TCP** or **TTY** for **Device Type** and click **Next**.
6. Select one or more **Local IP Addresses** if the **Device Type** was set for **TCP**. Select the **Local TTY Device** if the **Device Type** was set to **TTY**. Click **Next**.
7. Select the **Remote IP Address** if the **Device Type** was set for **TCP**. Select the **Remote TTY Device** if the **Device Type** was set to **TTY**. Click **Next**.
8. Enter or select the **Priority** for this comm path if the **Device Type** was set for **TCP**. Enter or select the **Baud Rate** for this Comm Path if the **Device Type** was set to **TTY**. Click **Next**.
9. Click **Create**. A message should be displayed indicating the network connection is successfully created. Click **Next**.
10. If you selected multiple Local IP Addresses or multiple Remote Servers and the **Device Type** was set for **TCP**, then you will be taken back to Step 6 to continue with the next Comm Path. If you selected multiple Remote Servers and the **Device Type** was set for **TTY**, then you will be taken back to Step 5 to continue with the next Comm Path.
11. Click **Done** when presented with the concluding message.

You can verify the comm path by viewing the Server Properties Dialog or by entering the command **`lcdstatus -q`**. See the LCD man page for information on using **lcdstatus**. You should see an **ALIVE** status.

In addition, check the server icon in the right pane of the GUI. If this is the first comm path that has been created, the server icon shows a yellow heartbeat, indicating that one comm path is **ALIVE**, but there is no redundant comm path. 

The server icon will display a green heartbeat when there are at least two comm paths **ALIVE**. 

Creating Resource Hierarchies

Create resources for the services and applications you want to protect.

1. On the global toolbar, click on the **Create Resource Hierarchy** button.
2. A dialog entitled Create Resource Hierarchy will appear with a list of all recognized recovery kits installed within the cluster. Select the **Recovery Kit** that builds resource hierarchies to protect your application and click **Next**.
3. Select the **Switchback Type** and click **Next**.
4. Select the **Server** and click **Next**. **Note:** If you began from the server context menu, the server will be determined automatically from the server icon that you clicked on, and this step will be skipped.
5. Continue through the succeeding dialogs, entering whatever data is needed for the type of resource hierarchy that you are creating.

Recovery Kit Options

Each optional recovery kit that you install adds entries to the Select Recovery Kit list; for example, you may see Oracle, Apache, and NFS Recovery Kits. Refer to the Administration Guide that accompanies each recovery kit for directions on creating the required resource hierarchies.

> ✻ Please refer to the procedure for creating each resource for the Recovery Kits in the Application Recovery Kit Documentation. There you will find setup requirements for each Recovery Kit.

Creating a File System Resource Hierarchy

1. On the global toolbar, click on the **Create Resource Hierarchy** button.
2. A dialog entitled *Create Resource Wizard* will appear with a **Recovery Kit** list. Select *File System Resource* and click **Next**.
3. Select the **Switchback Type** and click **Next**.
4. Select the **Server** and click **Next**. **Note:** *If you began from the server context menu, the server will be determined automatically from the server icon that you clicked on, and this step will be skipped.*
5. The Create *gen/filesys Resource* dialog will now appear. Select the **Mount Point** for the file system resource hierarchy and click **Next**. The selected mount point will be checked to see that it is shared with another server in the cluster by checking each storage kit to see if it recognizes the mounted device as shared. If no storage kit recognizes the mounted device, then an error dialog will be presented:

<file system> is not a shared file system

Selecting **OK** will return to the *Create gen/filesys Resource* dialog.

Notes:

- In order for a mount point to appear in the choice list, the mount point must be currently mounted. If an entry for the mount point exists in the `/etc/fstab` file, LifeKeeper will remove this entry during the creation and extension of the hierarchy. It is advisable to make a backup of `/etc/fstab` prior to using the NAS Recovery Kit, especially if you have complex mount settings. You can direct that entries are re-populated back into `/etc/fstab` on deletion by setting the `/etc/default/LifeKeeper` tunable `REPLACEFSTAB=true|TRUE`.
- Many of these resources (SIOS DataKeeper, LVM, Device Mapper Multipath, etc.) require LifeKeeper recovery kits on each server in the cluster in order for the file system resource to be created. If these kits are not properly installed, then the file system will not appear to be shared in the cluster.

6. LifeKeeper creates a default **Root Tag** for the file system resource hierarchy. (This is the label used for this resource in the status display). You can select this root tag or create your own, then click **Next**.

7. Click **Create Instance**. A window will display a message indicating the status of the instance creation.

8. Click **Next**. A window will display a message that the file system hierarchy has been created successfully.

9. At this point, you can click **Continue** to move on to extending the file system resource hierarchy, or you can click **Cancel** to return to the GUI. If you click **Cancel**, you will receive a warning message that your hierarchy exists on only one server, and it is not protected at this point.

Frequently Used Commands

- Starting the LifeKeeper GUI client
  ```
  # /opt/LifeKeeper/bin/lkGUIapp
  ```
- Starting LifeKeeper
  ```
  # /opt/LifeKeeper/bin/lkstart
  ```
- Stopping LifeKeeper (stopping resources)
  ```
  # /opt/LifeKeeper/bin/lkstop
  ```
- Stopping LifeKeeper (without stopping resources)
  ```
  # /opt/LifeKeeper/bin/lkstop -f
  ```

- Checking a status of LifeKeeper
  Specify “-e” option to display the simple status
  `# /opt/LifeKeeper/bin/lcdstatus (or lcdstatus -e)`

- Checking a LifeKeeper log
  Refer to `/var/log/lifekeeper.log`. If you want to check the log output in real time, you can also use the tail command as follows.
  `# tail -f /var/log/lifekeeper.log`

- Collect LifeKeeper Configuration Information and Logs together
  `# /opt/LifeKeeper/bin/lksupport`

- Backup/Restore of LifeKeeper Configuration Information
  Taking a backup of the LifeKeeper configuration information
  `# /opt/LifeKeeper/bin/lkbackup -c`

- Restoring the LifeKeeper configuration information
  `# /opt/LifeKeeper/bin/lkbackup -x -f archive..tar.gz`

Support for SPS for Linux

Contact SIOS Technology Corp. Support at support@us.sios.com

You can also contact SIOS Technology Corp. Support at:

- 1-877-457-5113 (Toll Free)

- 1-803-808-4270 (International)

Email: support@us.sios.com

> In order to begin our investigation, we will need the ‘lksupport’ logs. These are critical in diagnosing the issue/status of the cluster and should be included whenever you contact Support. Run: ***/opt/LifeKeeper/bin>lksupport*** to create a .tar file for each node under the directory: */tmp/lksupport*

4. SIOS Protection Suite for Linux Installation Guide

The SIOS Protection Suite (SPS) Installation Guide contains information on how to plan and install your SPS environment. In addition to providing the necessary steps for setting up your server, storage device and network components, it includes details for configuring your LifeKeeper graphical user interface (GUI).

Once you have completed the steps in this guide, you will be ready to configure your LifeKeeper and DataKeeper resources. The SPS for Linux Technical Documentation provides the information needed to complete your SPS configuration.

System Requirements

For a complete list of hardware and software requirements and versions, see the SPS for Linux Release Notes.

Also, before installing SPS, be sure that you have completed the planning and hardware configuration tasks described in this document.

Technical Notes

Refer to the Technical Notes and Troubleshooting sections of the SPS for Linux Technical Documentation for information detailing troubleshooting issues, restrictions, etc., pertaining to this software.

4.1. Software Packaging

The SIOS Protection Suite (SPS) for Linux software, including Optional SPS Recovery Kits, is contained within a single image file (sps.img).

SPS for Linux Installation Image File

The SPS for Linux image file (sps.img) provides a set of installation scripts designed to perform the user interactive system setup tasks that are necessary when installing SPS on your system (see Interactive Way for more information). A non-user interactive install can be performed as well (see Non-interactive Way for more information).

The SPS installation process is broken down into 3 steps:

- Collection
- Selection (user interactive only)
- Installation and Configuration

The first step of the process is the Collection phase and is responsible for collecting information about the system, such as the Linux distribution being used, to ensure the system meets the requirements for a successful install. Step 2 of the process is the Selection phase and is responsible for interacting with the user via a menu based selection process to determine what SPS packages to install and the configurations required to support those selections. The third and final step is the Installation and Configuration phase. This step is responsible for installing the SPS Core Package Cluster and Optional Recovery Software, and configuring the system for SPS. This step also installs any required OS supporting packages that are not already on the system.

The SPS for Linux image file includes a core package cluster containing the following software packages:

SPS Core Package Cluster

- LifeKeeper (**steeleye-lk**). The LifeKeeper core packages provide recovery software for core system components, such as memory, CPUs, the operating system, the SCSI disk subsystem and file systems.

- LifeKeeper GUI (**steeleye-lkGUI**). The LifeKeeper GUI package provides a graphical user interface for LifeKeeper administration and monitoring.

- DataKeeper (**steeleye-lkDR**). The DataKeeper package provides data replication (synchronous or asynchronous mirrors) with intent logging.

- IP Recovery Kit (**steeleye-lkIP**). The LifeKeeper IP Recovery Kit provides switchover software for automatic recovery of IP addresses.

- Raw I/O Recovery Kit (**steeleye-lkRAW**). The LifeKeeper Raw I/O Recovery Kit provides support for applications that use raw i/o to bypass kernel buffering.

- Man Pages (**steeleye-lkMAN**). The LifeKeeper Man Page package provides reference manual pages for the LifeKeeper product.

Optional Recovery Software

Recovery kits are also released with the SPS Core software. During the installation, you will be presented with a complete, up-to-date, selectable list of available recovery kits. For information regarding these recovery kits, see the Application Recovery Kits section of the SPS Technical Documentation.

4.2. Planning Your SPS Environment

The following topics will assist in defining the SPS for Linux cluster environment.

4.2.1. Mapping Server Configurations

Document your server configuration using the following guidelines:

1. Determine the server names, processor types, memory and other I/O devices for your configuration. When you specify a backup server, you should ensure that the server you select has the capacity to perform the processing should a failure occur on the primary server.

2. Determine your communications connection requirements.

Important: Potentially, clustered configurations have two types of communications requirements: cluster requirements and user requirements.

- **Cluster** – A LifeKeeper cluster requires at least two communication paths (also called “comm paths” or “heartbeats”) between servers. This redundancy helps avoid “split-brain” scenarios due to communication failures. Two separate LAN-based (TCP) comm paths using dual independent subnets are recommended, and at least one of these should be configured as a private network. Using a combination of TCP and TTY is also supported. A TTY comm path uses an RS-232 null-modem connection between the servers’ serial ports.

  Note that using only one comm path can potentially compromise the ability of systems in a LifeKeeper cluster to communicate with each other. If a single comm path is used and the comm path fails, then LifeKeeper hierarchies may try to come into service on multiple systems simultaneously. This is known as a false failover or a “split-brain” scenario. In the “split-brain” scenario, each server believes it is in control of the application and thus may try to access and write data to the shared storage device. To resolve the split-brain scenario, LifeKeeper may cause servers to be powered off or rebooted or leave hierarchies out-of-service to assure data integrity on all shared data. Additionally, heavy network traffic on a TCP comm path can result in unexpected behavior, including false failovers and the failure of LifeKeeper to initialize properly.

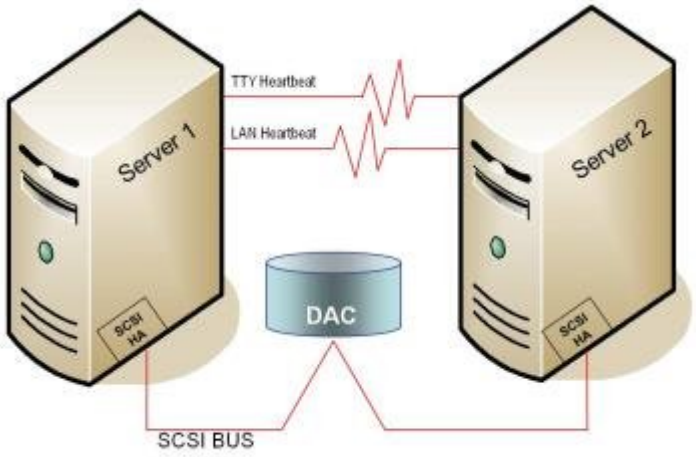
- **User** – We recommend that you provide alternate LAN connections for user traffic – that is, a separate LAN connection than the one used for the cluster heartbeat. However, if two TCP comm paths are configured (as recommended), one of those comm paths can share the network address with other incoming and outgoing traffic to the server.

Note: To help ensure that resources are brought into service only when necessary, you may elect to utilize the Quorum/Witness Server Support Package for LifeKeeper.

3. Identify and understand your shared resource access requirements. Clusters that use shared storage can utilize either shared SCSI buses or Fibre Channel loops. Because LifeKeeper locks resources to one server, you must ensure that only one server requires access to all locked resources at any given time. LifeKeeper device locking is done at the Logical Unit (LUN) level. For active/active configurations, each hierarchy must access its own unique LUN. All hierarchies accessing a common LUN must be active (in-service) on the same server.

4. Determine your shared memory requirements. Remember to take into account the shared memory requirements of third-party applications as well as those of LifeKeeper when configuring shared memory and semaphore parameters. See Tuning in Technical Notes for LifeKeeper's shared memory requirements.

Sample Configuration Map for LifeKeeper Pair



This sample configuration map depicts a pair of LifeKeeper servers sharing a disk array subsystem where, normally, Server 1 runs the application(s) and Server 2 is the backup or secondary server. In this case, there is no contention for disk resources because one server at a time reserves the entire disk storage space of the disk array. The disk array controller is labeled "DAC," and the SCSI host adapters (parallel SCSI, Fibre Channel, etc.) are labeled "SCSI HA."

A pair of servers is the simplest LifeKeeper configuration. When you plan a cluster consisting of more than two servers, your map is even more critical to ensure that you have the appropriate connections between and among servers. For example, in a multi-directional failover configuration, it is possible to define communications paths within LifeKeeper when the physical connections do not exist. Each server must have a physical communication path to every other server in the cluster in order to provide cascading failover capability.

4.2.2. Storage and Adapter Requirements

Determine your storage and host adapter requirements using the following guidelines:

Storage Devices – Based on your application's data storage requirements, you will need to determine the type and number of data storage devices required by your configuration. Your shared files should reside on a disk array subsystem (Redundant Array of Inexpensive Disks, or RAID). LifeKeeper supports a number of hardware RAID peripherals for use in LifeKeeper configurations. See Supported Storage for a list of the supported peripherals.

Consider the following issues when planning the configuration of your storage devices:

- LifeKeeper manages resources at the physical disk or Logical Unit (LUN) level, making the resources on each physical disk or LUN available to only one server in the configuration at a time. As a result, it is a good idea to plan disk allocations before you begin to configure LifeKeeper. For example, each hierarchy in active/active configurations must access its own unique LUN, so a minimum of two LUNs is required for a two-node active/active configuration.

- Some model-specific issues and hardware configuration details are listed in Supported Storage.

Adapters – Based upon the type of configuration and the number of peripherals, determine the types and number of SCSI or Fibre Channel Host Adapters required. It is important that any adapter you choose be supported by LifeKeeper, as well as by your Linux distribution so that there is a driver available. Refer to Supported Adapter Models for a list of supported host adapters. For reference purposes, you should add the host adapter specifications to your configuration map.

4.2.3. Storage and Adapter Options

For a list of the disk array storage models currently supported by LifeKeeper in shared storage configurations, see the Supported Storage. Refer to Storage and Adapter Configuration for details about driver versions and other configuration requirements for these arrays and adapters.

Note that a supported disk array and adapter are not required in LifeKeeper configurations involving non-shared storage with IP failover only or when using SIOS Data Replication or Network Attached Storage.

SIOS Technology Corp. does not specifically certify fibre channel hubs and switches, because there are no known LifeKeeper-specific restrictions or requirements on these devices. Unless otherwise noted for a given array in Storage and Adapter Configuration, LifeKeeper recommends the hubs and switches that the disk array vendor supports.

4.3. Setting Up Your SPS Environment

Now that the requirements have been determined and LifeKeeper configuration has been mapped, components of this SPS environment can be set up.

> ✻ Although it is possible to perform some setup tasks in a different sequence, this list is provided in the recommended sequence.

Installing the Linux OS and Associated Communications Packages

Linux Dependencies

Connecting Servers and Shared Storage

Configuring Shared Storage

Verifying Network Configuration

Creating Switchable IP Address

Installing and Setting Up Database Applications

Configuring GUI Users

4.3.1. Installing the Linux OS and Associated Communication Packages

Before attempting to install the SPS for Linux software, you must first ensure that your Linux operating system is successfully installed and operational. Please see the Linux installation instructions provided with your distribution of Linux for complete installation details.

Notes:

- Refer to the Linux Dependencies topic for further dependencies that may be necessary for the required packages.
- It is possible to install Linux *after* connecting and configuring your shared storage, but it may be simpler to have Linux installed and running before introducing new peripheral devices.
- The SPS for Linux Installation Image File provides a set of installation scripts designed to perform user-interactive system setup tasks and installation tasks for installing SPS on your system.

4.3.2. Linux Dependencies

Successful completion of the installation of SPS for Linux requires the installation of a number of prerequisite packages. To prevent script failures, these packages should be installed prior to attempting to run the installation setup script.

The prerequisite packages are broken down into the following three groups:

- General Package Dependencies
- Optional Recovery Kit Package Dependencies

Depending on the operating system version and the packages installed based on the operating system type selected (minimal, default, etc.), additional dependent packages may be required.

> The dependencies are based on the versions of the OS supported by LifeKeeper. Refer to the Support Matrix for details.

Note: You may want to consider using a repository-based package manager such as **yum** or **zypper** that is designed to automatically resolve dependencies by searching in predefined software repositories thereby easing the installation of these required packages. To facilitate the installation of dependent packages, the SPS for Linux installer uses **yum** or **zypper** to install the SPS for Linux packages. Therefore, it is highly recommended that an OS package repository be setup and configured. This will negate the need to install the OS dependent packages listed below before attempting to install SPS for Linux.

rpm Install Example

```
rpm -ivh <package(s)>
```

yum Install Example

```
yum install <package(s)>
```

Zypper Install Example

```
zypper install <package(s)>
```

yum/Zypper Package Lists

The following list of rpm packages, for each distribution listed and installed with the corresponding package installer, is the minimum list of packages that will resolve all the required dependencies for SIOS Protection Suite for Linux:

Red Hat Enterprise Linux, CentOS and Oracle Linux

```
yum install libXtst libstdc++ bzip2-libs pam zlib patch redhat-lsb
ncurses-libs
```

SLES

```
zypper install libstdc++ bzip2 pam pam-modules zlib lsb libncurses5
```

General Package Dependencies

The following packages are always required to successfully install SPS for Linux. The package architecture version of the installed package should always match the operating system architecture (x86 or x86_64):

Red Hat Enterprise Linux, CentOS and Oracle Linux

- bzip2
- iproute
- iputils
- patch (version 2.5 or later)
- redhat-lsb
- ncurses-libs

Note: Some or all of these packages may already be installed depending on the selections made during the install of the operating system.

SLES

- bzip2
- iproute2
- iptables
- iputils
- insserv
- patch (version 2.5 or later)
- lsb-release
- libncurses5
- libXtst6 (SLES15 only)
- libXi6 (SLES15 only)

Note: Some or all of these packages may already be installed depending on the selections made during the install of the operating system.

Dependency with syslog daemon

LifeKeeper logs use the syslog daemon. LifeKeeper supports rsyslog. Before installing LifeKeeper the

rsyslog daemon must be installed and activated.

Note: In the distributions using systemd such as RHEL7 or SLES12, journald administrates the log collectively. Because LifeKeeper does log output using the syslog daemon, the syslog daemon also must be operating in these environments. Therefore, set up syslog daemon to operate when using LifeKeeper.

Note: journald records the log output to a temporary file system (tmpfs) mount on /run/log/journal by default. Thus, the system log is not saved at the time of the OS shutdown. Change the setup to let the journald log perpetuate.

Note: To let the journald log perpetuate, set up “Storage=persistent” in /etc/systemd/journald.conf, or, create the /var/log/journal directory with the set up “Storage=auto” (default). After changing the set up, restart systemd-journald.service.

Optional Recovery Kit Package Dependencies

Additionally, some of the SIOS Protection Suite for Linux optional Application Recovery Kits (ARKs) require supporting packages to be installed.

If NFS exports are to be protected via the SIOS Protection Suite for Linux NFS Application Recovery Kit, then the following dependent packages are required:

- nfs-utils (Red Hat Enterprise Linux, CentOS, Oracle Linux)
- nfs-client (SLES)
- nfs-kernel-server (SLES)

If multipath devices are to be protected via Device Mapper Multipath (DMMP), Hitachi Dynamic Link Manager Software (HDLM), Power Path or NEC iStorage StoragePathSavior (NECSPS), then the following dependent packages are required:

- sg3_utils (All multipath kits)
- sg3_utils-libs (All multipath kits)
- HDLM (Hitachi Dynamic Link Manager Software Kit)
- EMCpower.LINUX (Power Path Kit)
- sps (NEC iStorage StoragePathSavior Kit **4.2.0** or prior)
- sps-utils and sps-driver (NEC iStorage StoragePathSavior Kit **4.2.1** or later)

If Websphere MQSeries queue managers are to be protected via the SIOS Protection Suite for Linux Websphere MQ/MQSeries Application Recovery Kit, then the following dependent Websphere MQ packages are required:

- MQSeriesServer
- MQSeriesSamples
- MQSeriesClient
- MQSeriesRuntime
- MQSeriesSDK

If Software RAID devices are to be protected via the SIOS Protection Suite for Linux Software RAID (md) Recovery Kit, then the following dependent package is required:

- mdadm

4.3.3. Connecting Servers and Shared Storage

If you are planning to use LifeKeeper in a non-shared storage environment, then you may skip this information. If you are using LifeKeeper in a data replication (mirroring) environment, see the DataKeeper section of this documentation. If you are using LifeKeeper in a network attached storage environment, see LifeKeeper Network Attached Storage Recovery Kit Administration Guide.

Once Linux is installed, you should set the host adapter and shared peripheral addressing. Refer to the documentation accompanying your adapter and storage device for specific details.

4.3.4. Configuring Shared Storage

LifeKeeper configurations may use the facilities of shared Small Computer System Interface (SCSI) host adapters and shared disk hardware to switch resources from a failed server to a designated backup server. A Fibre Channel Storage Area Network (SAN) may also be used to switch resources from a failed server to a designated backup server.

Perform the following tasks before creating disk-based application resource hierarchies that enable LifeKeeper to provide failover protection.

1. Partition disks and LUNs. Because all disks placed under LifeKeeper protection must be partitioned, your shared disk arrays must now be configured into logical units, or LUNs. Use your disk array management software to perform this configuration. You should refer to your disk array software documentation for detailed instructions.

   Note: Remember that LifeKeeper locks **its disks** at the LUN level. Therefore, one LUN may be adequate in an Active/Standby configuration. But, if you are using an Active/Active configuration, then you must configure at least two separate LUNs, so that each hierarchy can access its **own unique** LUN.

2. Verify that both servers recognize the shared disks (for example, using the **gdisk** command). If Linux does not recognize the LUNs you have created, then LifeKeeper will not either.

3. Create file systems on your shared disks from the system you plan to use as the primary server in your LifeKeeper hierarchy. Refer to the Linux documentation for complete instructions on the administration of file systems.

4.3.5. Verfying Network Configuration

It is important to ensure that your network is configured and working properly before you install LifeKeeper. There are several tasks you should do at this point to verify your network operation:

1. If your server installation has a firewall enabled, you will either need to accommodate the LifeKeeper ports or disable the firewall. Please refer to Running LifeKeeper With a Firewall.

2. From each server, ping the local server, and ping the other server(s) in the cluster. If the ping fails, then do the necessary troubleshooting and perform corrective actions before continuing.

3. If your server has more than one network adapter, you should configure the adapters to be on different subnets. If the adapters are on the same subnet, TCP/IP cannot effectively utilize the second adapter.

4. Ensure that *localhost* is resolvable by each server in the cluster. If DNS is not implemented, edit the */etc/hosts* file and add an entry for the *localhost* name. This entry can list either the IP address for the local server, or it can list the default entry (127.0.0.1). If *localhost* is not resolvable, the LifeKeeper GUI may not work.

5. If DNS is implemented, verify the configuration to ensure the servers in your LifeKeeper cluster can be resolved using DNS.

6. Ensure each server's hostname is correct and will not change after LifeKeeper is installed. If you later decide to change the hostname of a LifeKeeper system, you should follow these steps *on all servers in the cluster*.

   a. Stop LifeKeeper on all servers in the cluster using the command:

   ```
   /opt/LifeKeeper/bin/lkstop -f
   ```

   b. Change the server's hostname using the Linux **hostname** command.

   c. Before continuing, you should ensure that the new hostname is resolvable by each server in the cluster (see the previous bullets).

   d. Run the following command on every server in the cluster to update LifeKeeper's hostname. (Refer to lk_chg_value(1M) for details.)

   ```
   opt/LifeKeeper/bin/lk_chg_value -o oldhostname -n newhostname
   ```

   e. Start LifeKeeper using the command:

   ```
   /opt/LifeKeeper/bin/lkstart
   ```

LifeKeeper for Linux v7.x supports VLAN interface for Communication Paths and IP resources. The type

of VLAN interface can be chosen as described below.

VLAN Interface Support Matrix

- not supported \ x supported

LK Linux v7.1 or Prior Version

VLAN_NAME_TYPE	CommPath	IP Resource
DEV_PLUS_VID (eth0.0100)	-	x
DEV_PLUS_VID_NO_PAD (eth0.100)	-	x
VLAN_PLUS_VID (vlan0100)	x	x
VLAN_PLUS_VID_NO_PAD (vlan100)	x	x

LK Linux v7.2 or Later Version

VLAN_NAME_TYPE	CommPath	IP Resource
DEV_PLUS_VID (eth0.0100)	x	x
DEV_PLUS_VID_NO_PAD (eth0.100)	x	x
VLAN_PLUS_VID (vlan0100)	x	x
VLAN_PLUS_VID_NO_PAD (vlan100)	x	x

4.3.6. Creating Switchable IP Address

A switchable IP address is a “virtual” IP address that can be switched between servers. It is separate from the IP address associated with the network interface card of each server. Applications under LifeKeeper protection are associated with the switchable IP address. Then, if there is a failure on the primary server, that IP address “switches” to the backup server.

If you plan to configure resource hierarchies for switchable IP addresses, you must do the following on each server in the cluster:

- Verify that the computer name is correct and will not be changed.
- Verify that the switchable IP addresses are unique using the ping command.
- Edit the */etc/hosts* file to add an entry for each switchable IP address.

Refer to the LifeKeeper for Linux IP Recovery Kit Technical Documentation for additional information.

4.3.7. Installing and Setting Up Database Applications

If your environment includes a protected database application such as Oracle, DB2 or MySQL, you should install the application using the documentation provided with the database. Ensure that the database is on a shared file system and that the configuration files are on a shared file system. The executables may either be on each local or a shared file system.

Although it is possible to install your application *after* LifeKeeper is installed, you should test the application to ensure it is configured and operating properly before placing it under LifeKeeper protection. Please reference the specific LifeKeeper database recovery kit documentation for additional installation and setup considerations.

4.3.8. Configuring GUI Users

GUI Authentication with PAM

SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB). SPS no longer uses its private password file once located in /opt/LifeKeeper/website/passwd. Instead, users are identified and authenticated against the system's PAM configuration. Privilege levels are determined from group membership as provided through PAM.

In order to access the GUI, a user must be a member in one of the three LifeKeeper groups: lkadmin, lkoper or lkguest. Membership in these groups should be set by the system administrator using whatever technique is appropriate for the type of user account database that is being used throughout the cluster.

These three LifeKeeper groups provide three different sets of permissions (see Permissions Table).

1. Users with ***Administrator*** permission `(lkadmin)` throughout a cluster can perform all possible actions through the GUI.

2. Users with ***Operator*** permission `(lkoper)` on a server can view LifeKeeper configuration and status information and can bring resources into service and take them out of service on that server.

3. Users with ***Guest*** permission `(lkguest)` on a server can view LifeKeeper configuration and status information on that server.

During installation of the GUI package, the *root user* on the system is automatically added to the lkadmin group in the system's local group database allowing *root* to perform all LifeKeeper tasks on that server via the GUI application or web client. If you plan to allow users other than *root* to use LifeKeeper GUI clients, then these LifeKeeper GUI users will need to be configured by adding them to the appropriate group.

If PAM is configured to use a non-local database such as NIS, LDAP or AD, then the system administrator must ensure that the accounts are correctly configured in those databases. The groups listed above must exist and users who are allowed to log into the LifeKeeper GUI must be a member of one of these groups. These groups should be created in the remote database only and they should be removed from the local */etc/group* file.

If any system in the cluster is using an LK GUI password other than the system's 'root' password, the LK GUI login will fail. Once the root passwords are the same on each system in the cluster, the LK GUI login for 'root' will succeed.

Note: To avoid confusion and maintain consistency if leveraging more complex PAM configurations such as LDAP, NIS or AD, it is recommended that all user and

LifeKeeper group accounts exist prior to installing or upgrading SPS.

The best practice is to always grant permissions on a cluster-wide basis. It is possible to grant permissions on a single-server basis, but that is confusing to users and makes it impossible to perform administrative tasks.

4.4. Licensing

Obtaining and Installing the License

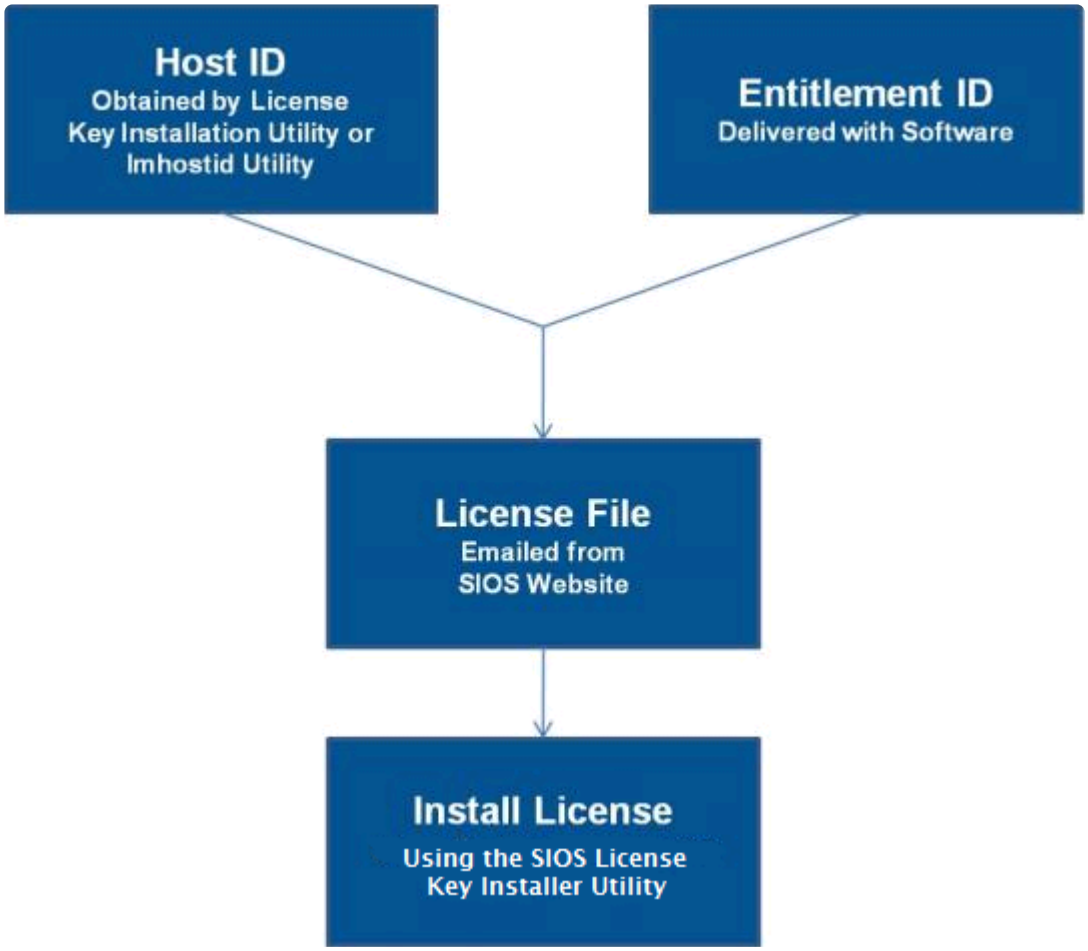
SPS for Linux requires a unique license for each server. The license is a run-time license, which means that you can install SPS without it, but the license must be installed before you can successfully start and run the product.

> **Note:** If using newer hardware with RHEL 6.1, please see the IP Licensing Known Issues in the SPS for Linux Troubleshooting Section.

The Installation script installs the Licensing Utilities package which obtains and displays all of the available Host IDs for your server during the initial install of your SIOS Protection Suite Software. Once your licenses have been installed the utility will return the Entitlement ID if it is available or the Host IDs if it is not.

> **Note:** Host IDs, if displayed will always be based on the MAC address of the NICs.

Starting with v8.2.0 any new licenses obtained from the SIOS Technology Corp. Licensing Operations Portal will contain your Entitlement ID and will be locked to a specific node or IP address in the cluster. The Entitlement ID (Authorization Code) which was provided with your SIOS Protection Suite Software, is used to obtain the permanent license required to run the SIOS Protection Suite Software. The process is illustrated below.



Note: Each software package requires a license for each server.

License Key Manager

In addition to installing SIOS Protection Suite for Linux product licenses, the **License Key Manager** allows you to perform the following functions:

- View all licenses currently installed on your system.
- View all expiration notifications (days remaining) for each time-expiring license.
- Identify invalid licenses that are currently installed.
- Delete any installed licenses (right-click on the license and select **Delete**).
- Delete all expired licenses as a group (press the **Delete Expired License** button).
- **Refresh** the Installed License list when installing software or upgrades.

Perform the following steps to obtain and install your licenses for each server:

1. Get your **Host ID**. At the end of the SPS installation, make note of the **Host ID** displayed by the

License Key Installer utility.

2. Write the **Host IDs** in a notebook or save them in a file. If saved in a file, copy that file to a system with internet access. Otherwise, take your notebook with you to the system with internet access.

3. Ensure you have your SIOS Protection Suite for Linux **Entitlement ID** (Authorization Code). You should have received an email with your software containing the Entitlement ID needed to obtain the license.

4. Obtain your licenses from the SIOS Technology Corp. Licensing Operations Portal.

   a. Using the system that has internet access, navigate to the SIOS Technology Corp. Licensing Operations Portal and log in entering your **User Name** and **Password** (or register if you do not already have an account).
   Note: New users must enter the Entitlement ID that is included in the delivery email..

   b. From the **Activation and Entitlements** dropdown select **List Entitlements**.

   c. Check the box to the left of the product line item(s) that you wish to license.

   d. From the **Action** dropdown select **Activate** and enter the requested information (including your system HOSTNAME) then select **Next**.

   e Click on the **Gray Plus Sign** to choose an already defined host or create a new host by selecting the **Green Plus Sign**.

   f. Select **ANY** for the Node Locked Host choice if it is available, otherwise select **ETHERNET MAC ADDRESS** and enter the Host ID (MAC address), click **OK** then click **Generate**.

> **Note:** The Host ID is 12 characters with no spaces, no colons, no dashes, and no separators.

   g. Check the box to the left of the **Fulfillment ID** and select **Complete**.

   h. From the **License Support** dropdown select **List Licenses**. Check the box to the left of the **Fulfillment ID** and select **Email** from the **View** dropdown.

   i. Enter a valid email address to send the license to and select **Send**.

   j. Retrieve the email(s).

   k. Copy the file(s) to the appropriate system(s).

5. Install your license(s).

- On each system, copy the license file(s) to /var/LifeKeeper/license. Run /opt/LifeKeeper/bin/

lkkeyins and specify the filename (including full path) to the file.

6. Repeat on all additional servers. You must install a license on the other SPS server(s) using the unique Host ID for each server.

7. Restart SIOS Protection Suite for Linux.

Primary Network Interface Change May Require a License Rehost

The Host ID used by the License Key Installer utility is obtained from the SPS for Linux server's primary network interface card (NIC). SPS for Linux will check for a valid license each time it starts. If your SPS server should require a NIC replacement in the future that would cause the Host ID to change, then the next time SPS for Linux is stopped, a License Rehost must be performed before starting either again. Log in to the SIOS Technology Corp. Licensing Operations Portal and select **License Support**, **List Licenses**, **Action**, **Rehost**. (**Note:** A rehost can be performed one time per six-month period without contacting support.)

Troubleshooting

If errors are encountered, please try the following before contacting Support:

- Verify credentials by logging in to the SIOS Technology Corp. Licensing Operations Portal. Enter **User ID** and **Password**. Run `%ExtMirrBase%\lmSubscribe.exe` again using the correct **User ID** and **Password**.

- To force a manual check for a license renewal, stop and restart the service. (**Note:** To find the service, bring up the view for all of the Linux services and search for "**SIOS Subscription Licensing**".)

- If ownership of the license certificate has changed, please contact SIOS Technology Corp. Support personnel to have the certificate moved to the new owner. Once ownership has been moved, the automatic license renewal service will need to be updated with these new credentials by running the above command again using the new **User ID** and **Password**.

4.4.1. Obtaining an Internet HOST ID

Use `lmutil` to obtain your machine's Internet Host ID. The Internet Host ID is normally the primary IP address of the primary network interface in the system. Internet Host IDs can be used as an alternative to Ethernet (or MAC) Host IDs and may be preferable in virtual environments where MAC addresses can change due to VM cloning.

1. Type the following command:

```
# /opt/LifeKeeper/bin/lmutil lmhostid -internet -n
```

2. Record the ID returned by the program.

Example:

```
# /opt/LifeKeeper/bin/lmutil lmhostid -internet -n

"INTERNET=172.17.100.161"
```

Note: This info must match the information contained in the permanent license key obtained from SIOS Technology Corp.

4.5. Installing the Software

This document will guide you through the installation of the SIOS Protection Suite for Linux (SPS) and assumes the user has basic knowledge of the Linux operating system. Please refer to the SIOS Protection Suite for Linux product documentation for more information.

Pre-Installation Requirements

Before installing SPS for Linux, please check the following:

- **SPS for Linux Release Notes** -The Release Notes include supported platforms, operating systems, applications, and storage. They also include the latest features and Bug Fixes.
- **TCP/IP Connection and Name Resolution** – In order to use the GUI function, both cluster nodes need to be able to resolve the name. Use the DNS service or */etc/hosts* for name resolution. Also, localhost needs to be resolved to 127.0.0.1.
- **Firewall** – The following ports are used:
  - Communication Path (TCP): 7365/tcp
  - Communication of a GUI Server: 81/tcp、82/tcp
  - RMI Communication between the GUI Server and Client: all the ports after 1024/tcp
  - Synchronization of DataKeeper (when using DataKeeper): “10001+<mirror number>+<256 * i>”

More Firewall Information

- The port used for communication with the GUI server and a client needs to be open on the cluster node where SPS is installed and on all systems where the GUI client runs.
- The ports used by DataKeeper can be calculated using the formula above. The value of i starts at 0 and uses an unused port when found. For example, in an environment where a DataKeeper resource with mirror number 0 exists, if port 10001 is being used by another application, port 10257 will be used.
- For communication between the GUI server and a client, Java RMI (Remote Method Invocation) randomly uses ports 1024 and above. When applying access control etc. to a cluster system, packet filtering needs to be performed considering these ports. If this specification is an issue from a security standpoint, you can use ssh X forwarding. Please refer to the Technical Documentation for the setting details.
- Add the following to the port numbers you are using: *WebGUI server process and policy setting with the lkpolicy command : 778(SSL) /tcp*

- ◦ **Check the SELinux Setting** – When the SELinux setting is enabled, SPS for Linux may not be able to be installed depending on the mode.
  - enforcing mode – SPS for Linux cannot be installed
  - permissive mode – SPS for Linux can be installed (not recommended except in some ARK environments)
    - It is not recommended to use SELinux permissive mode unless it is required in an SAP environment. Please make sure that the application to be run on the cluster supports permissive mode. SELinux permissive mode has been tested for following

ARKs: SAP / SAP MaxDB / Sybase / Oracle / DB2 / NFS / DataKeeper / NAS / EC2 / IP / FileSystem / MQ. Refer to Linux Dependencies for required packages.

- disabled mode – SPS for Linux can be installed
  - Please refer to the OS distribution documentation on how to disable SELinux.
- Install the appropriate package provided by your distribution.
- The sg3_utils package is required for environments using recovery kits for Multipath such as the DMMP Recovery Kit and the PowerPath Recovery Kit. This is not required for environments where recovery kits for Multipath are not used.

- **Check Known Issues** – Please make sure that there are no known issues for your environment.

Installing SPS for Linux

Install the SPS software on each server in the SPS configuration.

Packages that LifeKeeper is dependent on are installed automatically because the LifeKeeper installation setup script uses package manager tools (`yum` or `zypper`) to ensure installation of all dependent packages.

> **IMPORTANT:** A functional yum or zypper configuration is required for the successful installation of LifeKeeper. A non-functional configuration can result in an installation failure (see Installation Known Issues for more information). Additionally, the package manager repo or rpm database must not be locked as that could cause the install to hang. If the dependent packages cannot be installed automatically via the package manager, refer to Linux Dependencies and install the necessary packages in advance.

The SPS for Linux image file (sps.img) provides a set of installation scripts designed to perform the user interactive system setup tasks that are necessary when installing SPS on your system (see Interactive Mode for more information). A non-user interactive install can be performed as well (see Non-interactive Mode for more information).

A licensing utilities package is also installed providing utilities for obtaining and displaying the Host ID or Entitlement ID of your server. Host IDs and/or Entitlement IDs are used to obtain valid licenses for running SPS. Refer to Licensing for information on how to obtain and install your licenses.

> **Note:** These installation instructions assume that you are familiar with the Linux operating system installed on your servers.

> **IMPORTANT:**
> • Installing SPS on your shared storage is not supported. Each server should have its own copy installed on its local disk.
> • All SPS packages are installed in the directory /opt/LifeKeeper.

SPS will be installed through the command line regardless of the Linux distribution you are operating under.

- Please refer to How to Use Setup Scripts for the installation activities.
- For upgrade installations, see Upgrading SPS.

4.6. How to Use Setup Scripts

How to Install / Upgrade SPS Using the Setup Script

To install or upgrade SPS, follow the steps below.

Interactive Mode

1. After logging in as the root user, use the following command to mount the sps.img file:

   ```
   mount <PATH/IMAGE_NAME> <MOUNT_POINT> -t iso9660 -o loop
   ```

   Where PATH is the path to the image
   IMAGE_NAME is the name of the image
   MOUNT_POINT is the path to mount location

2. Change to the directory where sps.img is mounted and enter the following:

   ```
   ./setup
   ```

3. The script collects information about the system environment and determines what you need to do to install SPS.

   If the system requirements for installation or upgrade are not satisfied, then an error message is displayed and the installation / upgrade is cancelled.

   Also, if some restrictions arise or a configuration change is required, a warning message will be displayed requiring the user to decide whether to continue or abort the installation.

4. Select the SPS features and Application Recovery Kits (ARKs) to install via the main dialog screen.

   Please refer to How to Use the Dialog Screen.

5. Once all the required SPS features and ARKs have been selected, select <Done> to begin the installation.

   If any notifications are output when the installation completes, please take the necessary actions to correct them.

Non-interactive Mode

1. After logging in as root user, use the following command to mount the sps.img file:

   ```
   mount <PATH/IMAGE_NAME> <MOUNT_POINT> -t iso9660 -o loop
   ```

PATH is the path to the image
IMAGE_NAME is the name of the image
MOUNT_POINT is the path to mount location

2. Change to the directory where sps.img is mounted and enter the following command. First you will need to run setup in Interactive Mode, with the “Save Configuration” (-s) option:

   `./setup -s` <response_file>

   Select the necessary packages and options and complete setup. A configuration file will be saved in the location you specified. This configuration file can be copied to other systems and used as follows.

3. On the system where you wish to perform a non-interactive install, enter the following command:

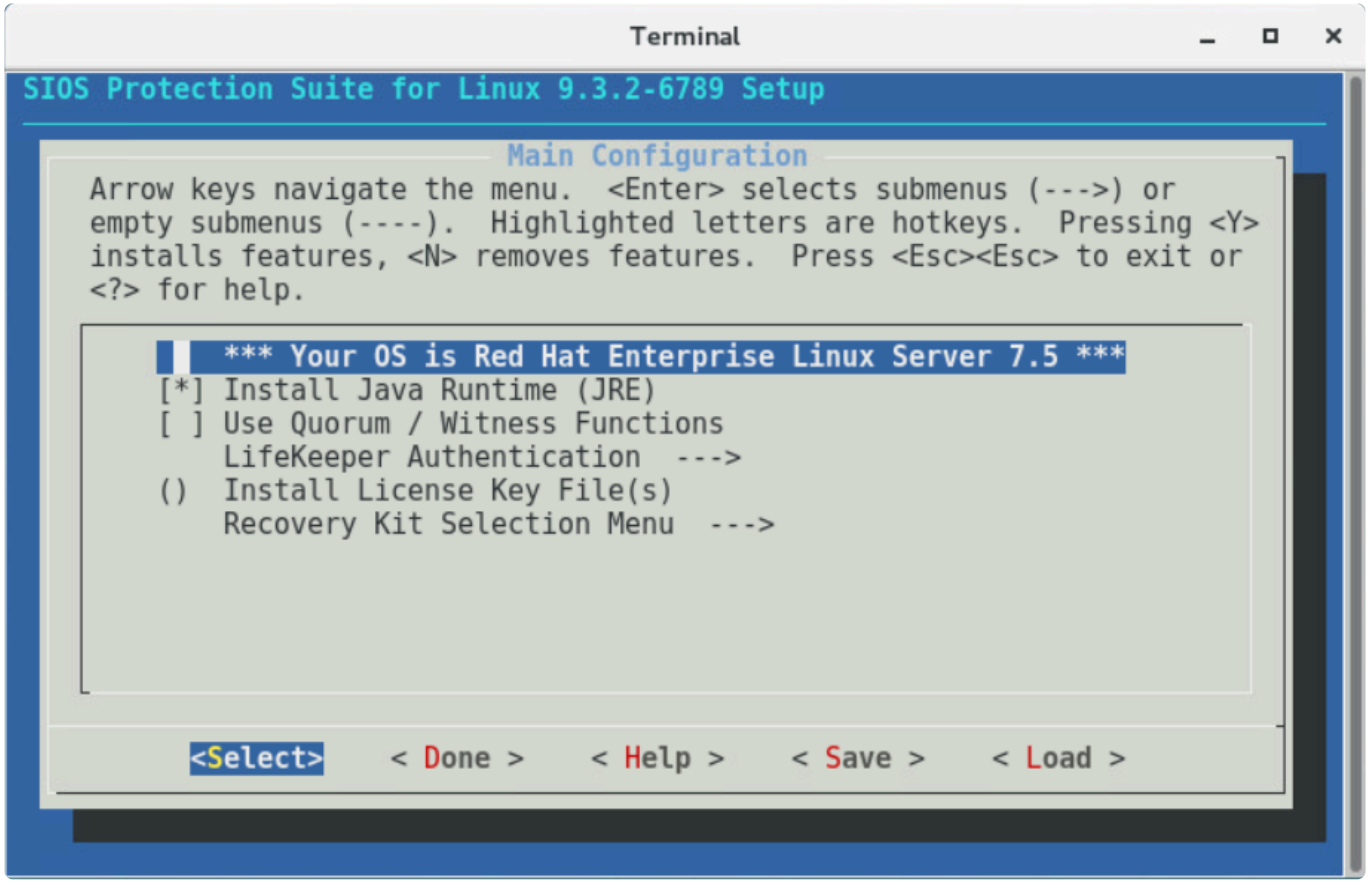
   `./setup -f` <response_file> -q y

   The “-q y” option ensures that user prompts are skipped, and the default answers given.

> **Note:** When using a configuration file for non-interactive installations the system on which the file is used must be configured the same as the system on which the file was generated. If the systems have too many differences the non-interactive installation may fail.

How to Use the Dialog Screen

The dialog screen is displayed below.



Use the following keys to navigate the menu.

↑ ↓	Navigate between menu items
← →	Navigate between the menu buttons at the bottom of the screen
ENTER	Open the selected sub menu
Y / N / SPACE	Turn on, turn off or invert the selected item

The menu buttons at the bottom of the screen are used for the following operations.

Select	Opens a sub menu dialog screen
Done	Closes the current screen and returns to the previous screen. Selecting this button on the main screen completes the configuration.
Help	Displays help for the highlighted item
Save	Saves the current settings in a configuration file. The saved configuration file can be used for non-interactive installations.
Load	Loads a saved configuration file

The “Save” and “Load” menu buttons display a dialog screen asking for a configuration file name for use in saving the current configuration or for loading a saved configuration. If you want to change the default file name provided, move to the file name field using the [TAB] key, and enter a new name. **Note:** The “Save” operation will prompt for confirmation before overwriting a file with the same name.

The items listed below are configurable during installation. During an upgrade only items that can be

configured are listed. Using the hotkey <Z> will show those items that will remain unchanged during the upgrade.

- **Install Java Runtime (JRE)**

  Install the Java runtime environment used by the LifeKeeper GUI.

- **Restart NFS Service**

  When configuring High Availability NFS, restarting the NFS services is required. When this is selected, the services are restarted automatically after the configuration is completed.

  Note: If you do not want to restart the NFS services automatically, a restart will need to be done to pick up the configuration changes before using the NFS Recovery Kit.

- **Use Quorum / Witness Functions**

  Use Quorum / Witness for I/O fencing. For details, please refer to Quorum/Witness in the technical documentation.

- **LifeKeeper Authentication**

  Specify the users allowed to log in to the SPS for Linux GUI along with their privilege levels. Multiple user accounts can be specified by separating them with blanks. For details, refer to GUI User Settings.

- **Install License Key File(s)**

  Install the licenses required to start SPS for Linux by entering the path name of the license file to install. Multiple files can be specified by separating them with spaces.

  Please refer to Licensing for details.

- **Recovery Kit Selection**

  Select the Application Recovery Kits to install.

  Application Recovery Kits are broken into several categories based on common functionality.

  Please refer to Categories for Application Recovery Kits for details.

- **LifeKeeper Startup After Install**

  When selected, SPS for Linux will be started when the installation is completed.

Note: Because the SPS-L Data Replication package may install kernel modules for

> some of the supported OS distributions, a re-install of SPS-L may be required when the kernel is upgraded. This applies to OS distributions for RedHat, CentOS and Oracle Enterprise Linux (non-UEK kernels only) running kernel versions 3.10.0-514 or later.

Adding / Removing Application Recovery Kits

To add Application Recovery Kits after completing an installation, simply execute setup, select the Recovery Kit in the Recovery Kit Selection, followed by the Application Recovery Kit Category and then select the desired kit.If you deselect an Application Recovery Kit which is no longer necessary, that kit will be removed.

Repair Installation

To repair an SPS for Linux installation run setup with the “—force” option. A repair installation will update the installation replacing any lost or corrupted files.

setup Script Options

The setup script can be executed with the following options:

- `-f <file>`

  Install non-interactively. <file> contains the configuration information to use during the installation.

- `-s <file>`

  Save a configuration file containing your menu selections. This file can then be used with the “-f” option to install the same LifeKeeper configuration to another system. For example, run:

  ```
  setup -s <file>
  ```

  Select the necessary packages and options and complete setup.

  Then run:

  ```
  setup -f <file> -q y
  ```

  to run a silent installation of LifeKeeper (on another system) with the same options that were selected the first time setup was run.

- `—force`

  Forcibly reinstall SPS for Linux.

- `-q <y/n>`

Specifies the response to any confirmation questions that may arise during non-interactive installation.

Categories for Application Recovery Kits

Category	Description
Application Suite	A group of recovery kits that protect applications such as SAP and IBM MQ.
Networking	A group of recovery kits that protect network services in the cloud such as EC2 and Route53.
Database	A group of recovery kits that protect database applications, including, but not limited to, Oracle, PostgreSQL, and MaxDB.
File Sharing	A group of recovery kits that protect file sharing services such as NFS and Samba.
Mail Server	A group of recovery kits that protect email services such as Postfix.
Storage	A group of recovery kits that protect data storage methods, including, but not limited to, DataKeeper (replication), Device Mapper (DM) Multipath (DMMP), and Network Attached Storage (NAS).
Web Server	A group of recovery kits that protect web services such as Apache.

4.7. Verfying the SPS Installation

You can verify that the SPS packages were installed correctly by entering the following at the command line:

```
rpm -V <package name>
```

Note: If the package is installed correctly, no output will be displayed by this command.

To perform a query from the command line, type

```
rpm -qi <package name>
```

Note: The expected output for this command is the package information.

4.8. Upgrading SPS

Beginning in v9.5.0 SIOS has released the new SAP HANA Application Recovery Kit. SIOS will continue to support the SAP HANA gen/app based Recovery Kit with the 9.4.x releases until **March 31, 2022**.
• The existing SAP HANA gen/app based Recovery Kit is **not** supported with v9.5.0.
• If you are using SIOS Protection Suite for Linux v9.5 or later you must use the new (built-in) SAP HANA Application Recovery Kit.
• Users who wish to upgrade to the SIOS Protection Suite for Linux v9.5.0 **must** convert their existing SAP HANA gen/app based Recovery Kit to the new SAP HANA Recovery Kit. Refer to Upgrading from the SAP HANA Gen/App to the SAP HANA Recovery Kit for details.

SPS for Linux may be upgraded to future releases while preserving existing resource hierarchies. Review this information carefully to ensure that you minimize application downtime.

Note: LifeKeeper can be upgraded to the current version from up to two versions back. If upgrading from a version previous to that, the older version will need to be uninstalled, and SIOS Protection Suite for Linux will have to be reinstalled. An alternative to uninstalling the older version would be to upgrade from the older version to one of the two acceptable versions, then perform the upgrade to the current version.

Note: If using `lkbackup` during your upgrade, see the lkbackup Known Issue for further information.

1. While upgrading a cluster, switch all applications away from the server to be upgraded now. Do this manually or by setting the LifeKeeper shutdown strategy to **"Switchover"** which causes the applications to be switched when LifeKeeper is stopped or the server is shut down.

2. If necessary, upgrade the Linux operating system before upgrading SPS. It is recommended that you unextend all resources from a server that is to be upgraded prior to performing the operating system upgrade.

3. Upgrade LifeKeeper by referring to How to Use Setup Scripts.

4. Switch all applications back to the upgraded server.

5. Repeat this procedure for each server in the SPS cluster to be upgraded.

CAUTION: The same version and release of SPS must be installed on all systems in a cluster. In general, different versions and/or releases of SPS are not compatible. For situations other than rolling upgrades, LifeKeeper should not be started when a different version or release is resident and running on another system in the cluster.

NOTES:

When upgrading the OS, make sure the currently installed version of LifeKeeper supports the upgraded version of the OS. If it is not supported, LifeKeeper will need to be upgraded as well provided a version of LifeKeeper has been released that supports the new OS version. If no version of LifeKeeper has been released that supports the new OS version you may not be able to upgrade the OS. Refer to the Supported Operating Systems.

Before upgrading the OS, it is recommended that the LifeKeeper configuration be backed up via the lkbackup command.

> **Note:** When using lkbackup, refer to the known issues of lkbackup.

1. When upgrading the cluster, all the resource hierarchies and thus the applications they protect, must be switched from the server to be upgraded to a standby node in the cluster. This can be done manually, or, by setting the LifeKeeper Shutdown Strategy to “Switchover”. By setting the Shutdown Strategy to “Switchover”, the resource hierarchies are switched over to a standby node when LifeKeeper stops or the servers are shutdown.
2. Stop LifeKeeper.
3. Upgrade and reboot the OS.
4. Upgrade LifeKeeper when required. If you do not upgrade, run setup again and update the settings corresponding to the new OS..
5. Start up LifeKeeper.
6. Switch all the resource hierarchies to the upgraded server.
7. Execute these steps for all the nodes in the SPS cluster.

> **Note:** All nodes in the cluster must be running the same version of the OS and the same version of LifeKeeper to be considered supported. Only during the upgrade process can the nodes differ in the OS and LIfeKeeper versions as this would be considered a temporary condition.

5. SIOS Protection Suite for Linux Technical Documentation

SIOS Protection Suite (SPS) for Linux integrates high availability clustering with innovative data replication functionality in a single, enterprise-class solution.

SPS for Linux Integrated Components

SIOS LifeKeeper provides a complete fault-resilient software solution to provide high availability for your servers' file systems, applications, and processes. LifeKeeper does not require any customized, fault-tolerant hardware. LifeKeeper simply requires two or more systems to be grouped in a network, and site-specific configuration data is then created to provide automatic fault detection and recovery.

In the case of a failure, LifeKeeper migrates protected resources from the failed server to a designated back-up server. Users experience a brief interruption during the actual switchover; however, LifeKeeper restores operations on the back-up server without operator intervention.

SIOS DataKeeper provides an integrated data mirroring capability for LifeKeeper environments. This feature enables LifeKeeper resources to operate in shared and non-shared storage environments.

Documentation and Training

5.1. Introduction

About SIOS Protection Suite for Linux

SIOS Protection Suite (SPS) for Linux integrates high availability clustering with innovative data replication functionality in a single, enterprise-class solution.

SPS for Linux Integrated Components

SIOS LifeKeeper provides a complete fault-resilient software solution to provide high availability for your servers' file systems, applications,and processes. LifeKeeper does not require any customized, fault-tolerant hardware. LifeKeeper simply requires two or more systems to be grouped in a network, and site-specific configuration data is then created to provide automatic fault detection and recovery.

In the case of a failure, LifeKeeper migrates protected resources from the failed server to a designated back-up server. Users experience a brief interruption during the actual switchover; however, LifeKeeper restores operations on the back-up server without operator intervention.

SIOS DataKeeper provides an integrated data mirroring capability for LifeKeeper environments. This feature enables LifeKeeper resources to operate in shared and non-shared storage environments.

Documentation and Training

5.2. Documentation and Training

Documentation

A complete reference providing instructions for installing, configuring, administering and troubleshooting SIOS Protection Suite for Linux.The following sections cover every aspect of SPS for Linux:

Section	Description
Introduction	Provides an introduction to the SIOS Protection Suite for Linux product, including software packaging and configuration concepts.
SPS for Linux Installation Guide	Provides useful information for planning and setting up your SPS environment, installing and licensing SPS and configuring the LifeKeeper graphical user interface (GUI).
Configuration	Contains detailed information and instructions for configuring the LifeKeeper software on each server in your cluster.
Administration	Discusses server-level tasks such as editing server properties and creating resources and resource-level tasks such as editing, extending or deleting resources.
User's Guide	Contains detailed information on the LifeKeeper GUI, including the many tasks that can be performed within the LifeKeeper GUI. Also includes a Technical Notes section along with many more Advanced Tasks.
DataKeeper	Contains planning and installation instructions as well as administration, configuration and user information for SIOS DataKeeper for Linux.
Troubleshooting	Describes known issues and restrictions and suggests solutions to problems that may be encountered during installation, configuration and/or use of SIOS LifeKeeper for Linux.
Recovery Kits	Contains planning and installation instructions as well as administration, configuration and user information for the Optional Recovery Kits that allow LifeKeeper to manage and control specific applications.
Error Code Search	Provides a listing of all messages that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. This full listing may be searched for any error code received.

Training

SPS training is available through SIOS Technology Corp. or through your reseller. Contact your sales representative for more information.

Technical Support

As a SIOS Technology Corp. customer with a valid Support contract, you are entitled to access the SIOS Technology Corp. Support Self-Service Portal.

The SIOS Technology Corp. Support Self-Service Portal offers you the following capabilities:

- Search our **Solution Knowledge Base** to find solutions to problems and answers to questions

- Always on 24/7 service with the SIOS Technology Corp. Support team to:
  - **Log a Case** to report new incidents.
  - **View Cases** to see all of your open and closed incidents.
  - **Review Top Solutions** providing information on the most popular problem resolutions being viewed by our customers.

Contact SIOS Technology Corp. Support at support@us.sios.com to set up and activate your Self-Service Portal account.

You can also contact SIOS Technology Corp. Support at:

1-877-457-5113 (Toll Free)

1-803-808-4270 (International)

Email: support@us.sios.com

5.3. LifeKeeper

The following SPS product documentation is available from the SIOS Technology Corp. website:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- Optional Recovery Kit Documentation

5.3.1. SIOS LifeKeeper for Linux Introduction

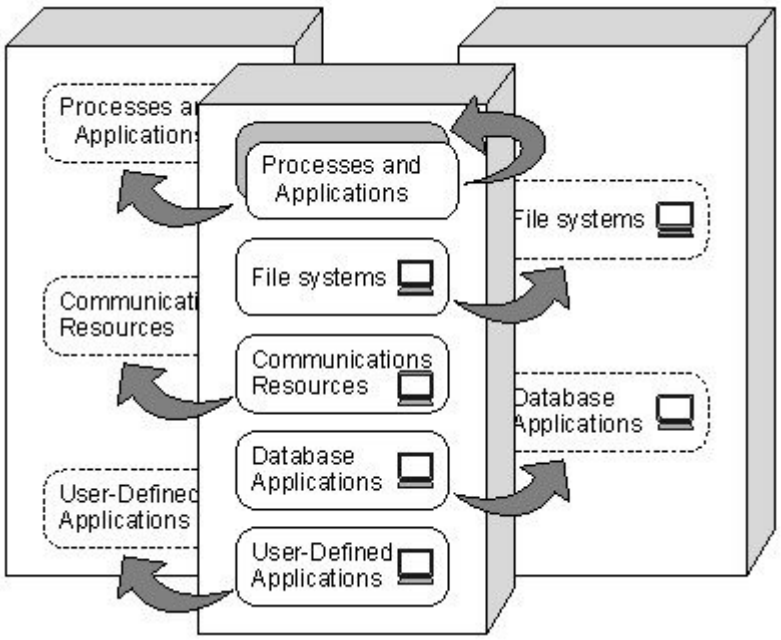
SIOS LifeKeeper for Linux provides high availability clustering for up to 32 nodes with many supported storage configurations, including shared storage (Fiber Channel SAN, iSCSI), network attached storage (NAS), host-based replication, integration with array-based SAN replication including HP Continuous Access and VMware virtual hard disk (VMDK).

5.3.1.1. Protected Resources

The LifeKeeper family of products includes software that allows you to provide failover protection for a range of system resources. The following figure demonstrates LifeKeeper's flexibility and identifies the resource types you can specify for automatic recovery:

- **File systems.** LifeKeeper allows for the definition and failover of file systems, such as ext3, ext4, NFS, vxfs or xfs.
- **Communications resources.** LifeKeeper provides communications Recovery Kits for communications resources, such as TCP/IP.
- **Infrastructure resources.** LifeKeeper provides optional Recovery Kits for Linux infrastructure services, such as NFS, Samba, LVM, WebSphere MQ, and software RAID (md).
- **Web Server resources.** LifeKeeper provides an optional Recovery Kit for Apache Web Server resources.
- **Databases and other applications.** LifeKeeper provides optional Recovery Kits for major RDBMS products such as Oracle, MySQL and PostgreSQL, Sybase, SAP DB/MaxDB, SAP HANA DB, and for enterprise applications such as SAP.
- **Cloud resources.** LifeKeeper provides communications Recovery Kits for communications resources, such as EC2 EIP, and Route53.

LifeKeeper supports N-Way Recovery for a range of resource types.



5.3.1.2. LifeKeeper Core

LifeKeeper Core is composed of four major components:

- LifeKeeper Core Software
- File System, Generic Application, Raw I/O and IP Recovery Kit Software
- LifeKeeper GUI Software
- LifeKeeper Man Pages

LifeKeeper Core Software

- The LifeKeeper Core Software consists of the following components:
- LifeKeeper Configuration Database (LCD) – The LCD stores information about the LifeKeeper-protected resources. This includes information on resource instances, dependencies, shared equivalencies, recovery direction, and LifeKeeper operational flags. The data is cached in shared memory and stored in files so that the data can be remembered over system boots.
- LCD Interface (LCDI) – The LCDI queries the configuration database (LCD) to satisfy requests for data or modifications to data stored in the LCD. The LCDI may also be used by the Application Recovery Kit to obtain resource state or description information.
- LifeKeeper Communications Manager (LCM) – The LCM is used to determine the status of servers in the cluster and for LifeKeeper inter-process communication (local and remote). Loss of LCM communication across all communication paths on a server in the cluster indicates the server has failed.
- LifeKeeper Alarm Interface – The LifeKeeper Alarm Interface provides the infrastructure for triggering an event. The sendevent program is called by application daemons when a failure is detected in a LifeKeeper-protected resource. The sendevent program communicates with the LCD to determine if recovery scripts are available.
- LifeKeeper Recovery Action and Control Interface (LRACI) – The LRACI determines the appropriate recovery script to execute for a resource and invokes the appropriate restore / remove scripts for the resource.

File System, Generic Application, IP and RAW I/O Recovery Kit Software

The LifeKeeper Core provides protection of specific resources on a server. These resources are:

- File Systems – LifeKeeper allows for the definition and failover of file systems on shared storage

devices. A file system can be created on a disk that is accessible by two servers via a shared SCSI bus. A LifeKeeper file system resource is created on the first server and then extended to the second server. File System Health Monitoring detects disk full and improperly mounted (or unmounted) file system conditions. Depending on the condition detected, the Recovery Kit may log a warning message, attempt a local recovery, or failover the file system resource to the backup server.

Specific help topics related to the File System Recovery Kit include Creating and Extending a File System Resource Hierarchy and File System Health Monitoring.

- Generic Applications – The Generic Application Recovery Kit allows protection of a generic or user-defined application that has no predefined Recovery Kit to define the resource type. This kit allows a user to define monitoring and recovery scripts that are customized for a specific application.

  Specific help topics related to the Generic Application Recovery Kit include Creating and Extending a Generic Application Resource Hierarchy.

- IP Addresses – The IP Recovery Kit provides a mechanism to recover a “switchable” IP address from a failed primary server to one or more backup servers in a LifeKeeper environment. A switchable IP address is a virtual IP address that can switch between servers and is separate from the IP address associated with the network interface card of each server. Applications under LifeKeeper protection are associated with the switchable IP address, so if there is a failure on the primary server, the switchable IP address becomes associated with the backup server. The resource under LifeKeeper protection is the switchable IP address.

  Refer to the IP Recovery Kit Technical Documentation included with the Recovery Kit for specific product, configuration and administration information.

- RAW I/O – The RAW I/O Recovery Kit provides support for raw I/O devices for applications that prefer to bypass kernel buffering. The RAW I/O Recovery Kit allows for the definition and failover of raw devices bound to shared storage devices. The raw device must be configured on the primary node prior to resource creation. Once the raw resource hierarchy is created, it can be extended to additional servers.

- Quick Service Protection (QSP) – QSP Recovery Kit provides a mechanism to simply protect OS services. Resources can be created for services that can be started/stopped with OS service commands. Generic Applications can provide the same protection, but QSP doesn’t require code development. Also, you can create dependencies to start/stop services with applications protected by other resources.

  However, QuickCheck of QSP only performs a simple check (using service command’s “status”) and does not ensure the provision of the services and running of the processes. If complicated start/stop processing or robust check is required, please consider the use of Generic Applications.

  For other topics regarding QSP, please see Creating/extending QSP resources.

LifeKeeper GUI Software

The LifeKeeper GUI is a client / server application developed using Java technology that provides a graphical administration interface to LifeKeeper and its configuration data. The LifeKeeper GUI client is implemented as both a stand-alone Java application and as a Java applet invoked from a web browser.

LifeKeeper Man Pages

The LifeKeeper Core reference manual pages for the LifeKeeper product.

5.3.1.3. Configuration Concepts

LifeKeeper works on the basis of resource hierarchies you define for groups of two or more servers. The following topics introduce the LifeKeeper failover configuration concepts:

5.3.1.3.1. Common Hardware Components

All LifeKeeper configurations share these common components:

1. **Server groups.** The basis for the fault resilience provided by LifeKeeper is the grouping of two or more servers into a cluster. The servers can be any supported platform running a supported distribution of Linux. LifeKeeper gives you the flexibility to configure servers in multiple overlapping groups, but, for any given recoverable resource, the critical factor is the linking of a group of servers with defined roles or priorities for that resource. The priority of a server for a given resource is used to determine which server will recover that resource should there be a failure on the server where it is currently running. The highest possible priority value is one (1). The server with the highest priority value (normally 1) for a given resource is typically referred to as the primary server for that resource; any other servers are defined as backup servers for that resource.

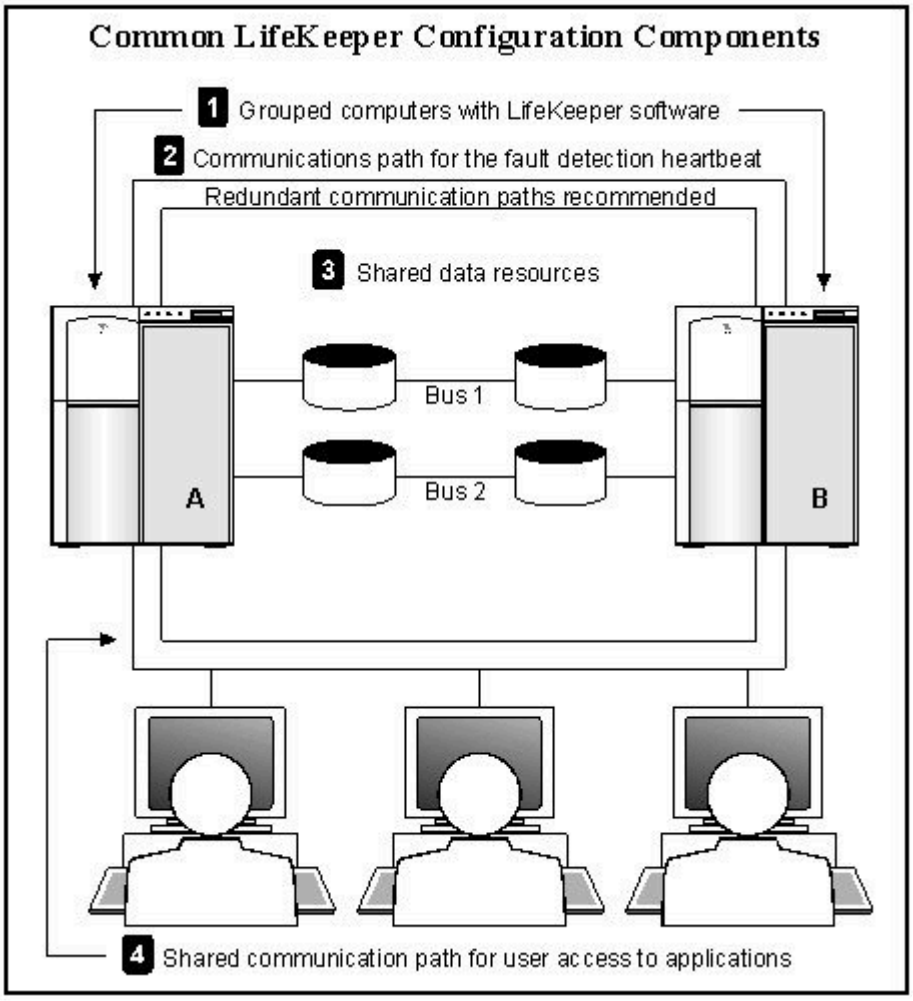
2. **Communications paths.** The LifeKeeper heartbeat, a periodic message between servers in a LifeKeeper cluster, is a key fault detection facility. All servers within the cluster require redundant heartbeat communications paths (or, comm paths) to avoid system panics due to simple communications failures. Two separate LAN-based (TCP) comm paths using dual independent subnets are recommended (at least one of these should be configured as a private network); however, using a combination of TCP and TTY comm paths is supported. A TCP comm path can also be used for other system communications.

   Note: A TTY comm path is used by LifeKeeper only for detecting whether other servers in the cluster are alive. The LifeKeeper GUI uses TCP/IP for communicating status information about protected resources; if there are two TCP comm paths configured, LifeKeeper uses the comm path on the public network for communicating resource status. Therefore if the network used by the LifeKeeper GUI is down, the GUI will show hierarchies on other servers in an UNKNOWN state, even if the TTY (or other TCP) comm path is operational.

3. **Shared data resources.** In shared storage configurations, servers in the LifeKeeper cluster share access to the same set of disks. In the case of a failure of the primary server, LifeKeeper automatically manages the unlocking of the disks from the failed server and the locking of the disks to the next available back-up server.

4. **Shared communication.** LifeKeeper can automatically manage switching of communications resources, such as TCP/IP addresses, allowing users to connect to the application regardless of where the application is currently active.

Components Common to All LifeKeeper Configurations



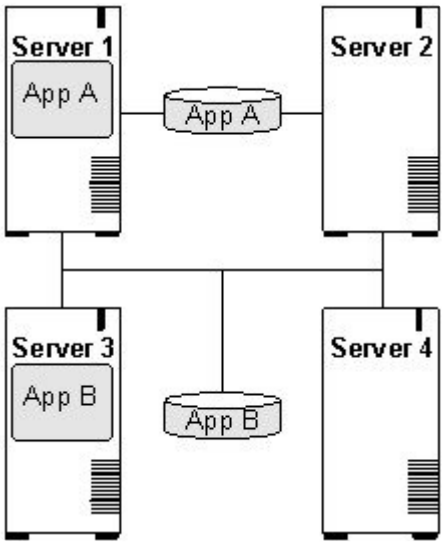
5.3.1.3.2. System Grouping Arrangements

A resource hierarchy is defined on a cluster of LifeKeeper servers. For a given hierarchy, each server is assigned a priority, with one (1) being the highest possible priority. The primary, or highest priority, server is the computer you want to use for the normal operation of those resources. The server having the second highest priority is the backup server to which you want LifeKeeper to switch those resources should the primary server fail.

In an active/active group, all servers are active processors, but they also serve as the backup server for resource hierarchies on other servers. In an active/standby group, the primary server is processing and any one of the backup servers can be configured to stand by in case of a failure on the primary server. The standby systems can be smaller, lower-performance systems, but they must have the processing capability to assure resource availability should the primary server fail.

Your physical connections and access to the shared resources determine your grouping options. To be grouped, servers must have communications and heartbeat paths installed and operational, and all servers must have access to the disk resources through a shared SCSI or Fibre Channel interface. For example, in the following diagram, there is only one grouping option for the resource *AppA* on Server 1. Server 2 is the only other server in the configuration that has shared access to the *AppA* database.

The resource *AppB* on Server 3, however, could be configured for a group including any one of the other three servers, because the shared SCSI bus in this example provides all four servers in the configuration access to the *AppB* database.



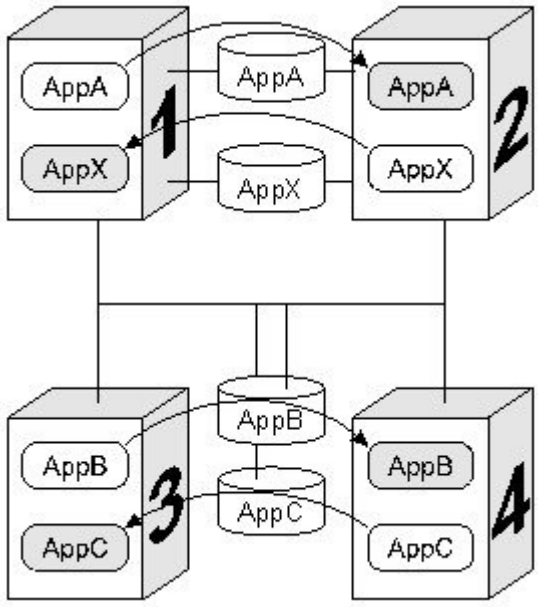
5.3.1.3.3. Active – Active Grouping

In an active/active pair configuration, all servers are active processors; they also serve as the backup server for resource hierarchies on other servers.

For example, the configuration example below shows two active/active pairs of servers. Server 1 is processing *AppA*, but also serves as the backup server for *AppX* running on Server 2. The reverse is also true. Server 2 is processing *AppX*, but also serves as the backup server for *AppA* running on Server 1. Servers 3 and 4 have the same type of active/active relationships.

Although the configurations on Servers 1 and 2 and the configurations on Servers 3 and 4 are similar, there is a critical difference. For the *AppA* and *AppX* applications, Servers 1 and 2 are the only servers available for grouping. They are the only servers that have access to the shared resources.

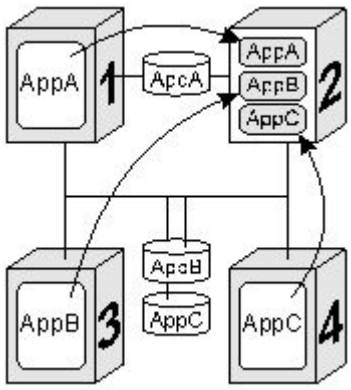
AppB and *AppC*, however, have several grouping options because all four servers have access to the *AppB* and *AppC* shared resources. *AppB* and *AppC* could also be configured to failover to Server1 and/or Server2 as a third or even fourth backup system.



Note: Because LifeKeeper applies locks at the disk level, only one of the four systems connected to the *AppB* and *AppC* disk resources can have access to them at any time. Therefore, when Server 3 is actively processing *AppB*, those disk resources are no longer available to Servers 1, 2, and 4, even though they have physical connections.

5.3.1.3.4. Active – Standby Grouping

In an active/standby pair configuration, the primary server is processing, and the back-up servers are standing by in case of a failure on the primary server. The standby systems can be smaller, lower-performance systems, but they must have the processing capability to assure resource availability should the primary server fail.



A standby server can provide backup for more than one active server. For example in the figure above, Server 2 is the standby server in three active/standby resource pairs. The LifeKeeper resource definitions specify the following active/standby paired relationships:

- *AppA* on *Server1* fails over to *Server2*.
- *AppB* on *Server3* fails over to *Server2*.
- *AppC* on *Server4* fails over to *Server2*.

Be aware of these three critical configuration concepts when you are considering configurations with multiple active/standby groups:

- **Disk ownership.** Different active applications cannot use disk partitions on the same shared disk or LUN from different servers. LifeKeeper applies locks at the disk or LUN level. When the SCSI locks are applied, only one system on the shared SCSI bus can access partitions on the disk or LUN. This requires that applications accessing different partitions on the same disk be active on the same server. In the example, Server 3 has ownership of the *AppB* disk resources and Server 4 owns the *AppC* resources.
- **Processing capacity.** Although it is unlikely that Servers 1, 3 and 4 would fail at the same time, you must take care when designating a standby server to support multiple resource relationships so that the standby server can handle all critical processing should multiple faults occur.
- **LifeKeeper administration.** In the example, Server 2 provides backup for three other servers. In general it is not desirable to administer the LifeKeeper database on the different logical groups simultaneously. You should first create the resources between the spare and one active system, then between the spare and another active system, and so on.

5.3.1.3.5. Intelligent Versus Automatic Switchback

By default, the switchback setting of a resource is *intelligent*. This means that once the failover occurs for that resource from *Server A* to *Server B*, the resource remains on *Server B* until another failure or until an administrator *intelligently* switches the resource to another server. Thus, the resource continues to run on *Server B* even after *Server A* returns to service. *Server A* now serves as a backup for the resource.

In some situations, it may be desirable for a resource to switch back automatically to the original failed server when that server recovers. LifeKeeper offers an *automatic switchback* option as an alternative to the default *intelligent switchback* behavior described above. This option can be configured for individual resource hierarchies on individual servers. If *automatic switchback* is selected for a resource hierarchy on a given server and that server fails, the resource hierarchy is failed over to a backup system; when the failed server recovers, the hierarchy is automatically switched back to the original server.

Notes:

- For automatic switchback, switch back will take place automatically after the primary server comes back online and LifeKeeper communications path is re-established.
- LifeKeeper never performs an *automatic switchback* from a higher priority server to a lower priority server.

5.3.1.3.6. Logging With syslog

Beginning with LifeKeeper v8.0, logging is done through the standard syslog facility. LifeKeeper supports rsyslog, which is an extension of the original syslog protocol. During package installation, `syslog` will be configured to use the “local6” facility for all LifeKeeper log messages (if “local6” is already in use another local should be used). The `syslog` configuration file */etc/rsyslog.conf* is modified to include LifeKeeper-specific routing sending all LifeKeeper log messages to */var/log/lifekeeper.log* (the original configuration file will be backed up with the same name ending in “~”).

> **Important:** DO NOT edit LifeKeeper’s unique setup steps manually or upgrading and uninstalling may not function correctly.

The facility can be changed after installation by using the `lklogconfig` tool located in */opt/LifeKeeper/bin*. For example, changing the facility to local5, run the following command.

```
lkstop -f
lklogconfig --action=update --facility=local5
lkstart
```

See the lklogconfig(8) manpage on a system with LifeKeeper installed for more details on this tool.

If a generic resource script puts a message into */opt/LifeKeeper/out/log* directly, LifeKeeper will send a log message with the error severity level as the default into */var/log/lifekeeper.log*. The severity level can be changed to information level by adding the following parameter into */etc/default/LifeKeeper*, **LOGMGR_LOGLEVEL=LK_INFO**.

> **Note:** When LifeKeeper is removed from a server, the LifeKeeper-specific `syslog` configuration will be removed.

5.3.1.3.7. Resource Hierarchies

The LifeKeeper GUI enables you to create a resource hierarchy on one server, then extend that hierarchy to one or more backup servers. LifeKeeper then automatically builds the designated hierarchies on all servers specified. LifeKeeper maintains hierarchy information in a database on each server. If you use the command line interface, you must explicitly define the hierarchy on each server.

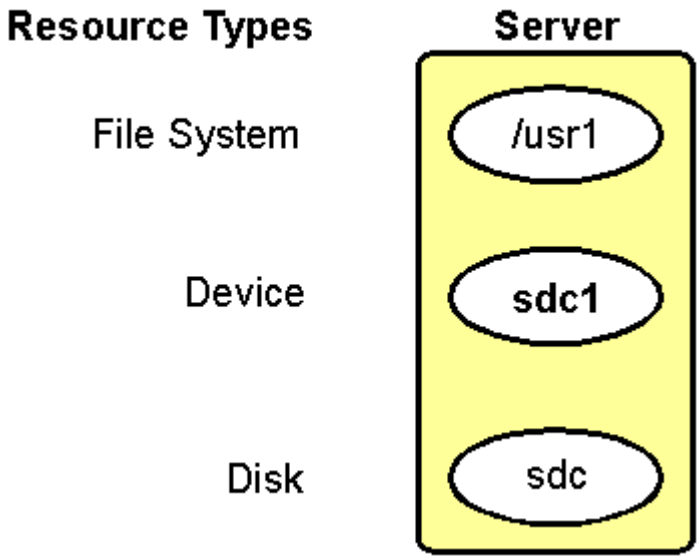
After you create the resource hierarchy, LifeKeeper manages the stopping and starting of the resources within the hierarchy. The related topics below provide background for hierarchy definition tasks.

5.3.1.3.7.1. Resource Types

A resource can be either a hardware or software entity, categorized by resource type. LifeKeeper supplies file system and SCSI resource types, and the recovery kits provide communications, RDBMS and other application resource types.

For example, a hierarchy for a protected file system includes instances for resources of the following types:

- **filesys**. Linux file system resource objects identified by their mount point.
- **device**. SCSI disk partitions and virtual disks, identified by their device file names, for example *sdc1*.
- **disk**. SCSI disks or RAID system logical units, identified by SCSI device name, for example *sd*.

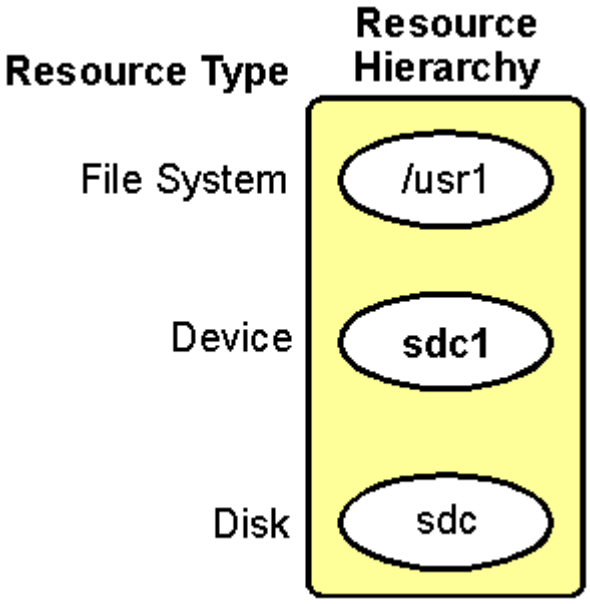


5.3.1.3.7.2. Resource States

State	Meaning
In-Service, Protected (ISP)	Resource is operational. LifeKeeper local recovery operates normally. LifeKeeper inter-server recovery and failure recovery is operational.
In-Service, Unprotected (ISU)	Resource is operational. However, no local recovery or failure recovery will occur because the LifeKeeper protection is not operational. **Note:** When the file system protected by the file system resource(filesys) has reached at least 90% (the default threshold) of its capacity, the resource status will be changed to ISU to alert the user. In this case, the monitoring process is continued. For the monitoring of file system resources and its capacity, the ISU state is used differently from other resource types. Once the file system capacity drops below the threshold, the resource state will return to ISP.
Out-of-Service, Failed (OSF)	Resource has gone out-of-service because of a failure in the resource. Recovery has not been completed or has failed. LifeKeeper alarming is not operational for this resource.
Out-of-Service, Unimpaired (OSU)	Resource is out-of-service but available to take over a resource from another server.
Illegal (Undefined) State (ILLSTATE)	This state appears in situations where no state has been set for a resource instance. Under normal circumstances, this invalid state does not last long: a transition into one of the other states is expected. This state will occur if switchover occurs before all LifeKeeper information tables have been updated (for example, when LifeKeeper is first started up).

5.3.1.3.7.3. Hierarchy Relationships

LifeKeeper allows you to create relationships between resource instances. The primary relationship is a dependency, for example one resource instance depends on another resource instance for its operation . The combination of resource instances and dependencies is the resource hierarchy.



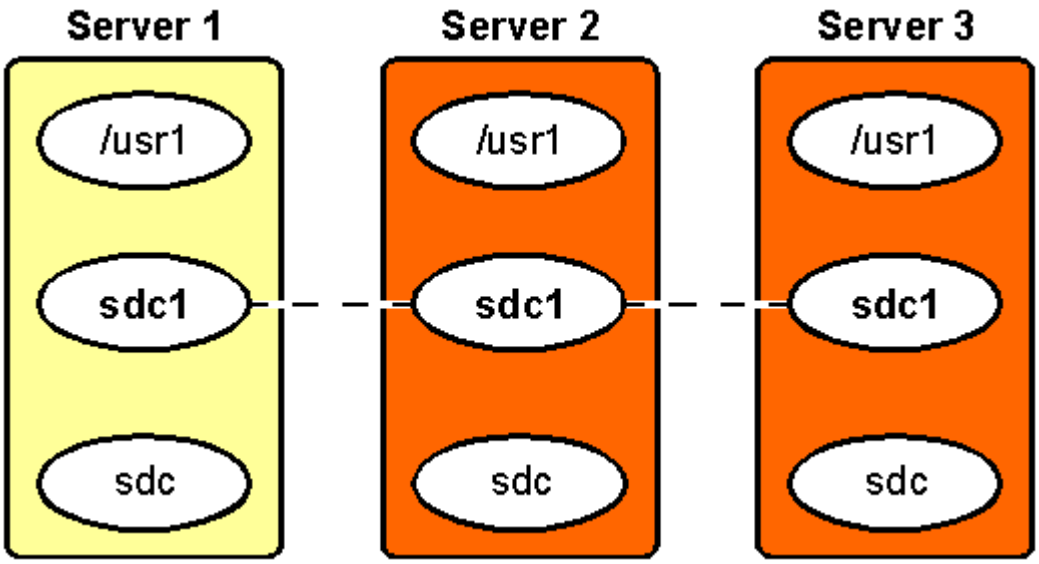
For example, since */usr1* depends on its operation upon the disk subsystem, you can create an ordered hierarchy relationship between */usr1* and those instances representing the disk subsystem.

The dependency relationships specified by the resource hierarchy tell LifeKeeper the appropriate order for bringing resource instances in service and out-of-service. In the example resource hierarchy, LifeKeeper cannot bring the */usr1* resource into service until it successfully brings into service first the *disk* and *device* instances.

5.3.1.3.7.4. Shared Equivalencies

When you create and extend a LifeKeeper resource hierarchy, the hierarchy exists on *both* the primary and the secondary servers. Most resource instances can be active on only one server at a time. For such resources, LifeKeeper defines a second kind of relationship called a shared equivalency that ensures that when the resource is *in-service* on one server, it is *out-of-service* on the other servers on which it is defined.

In the example below, the shared equivalency between the disk partition resource instances on each server is represented. Each resource instance will have a similar equivalency in this example.



5.3.1.3.7.5. Resource Hierarchy Information

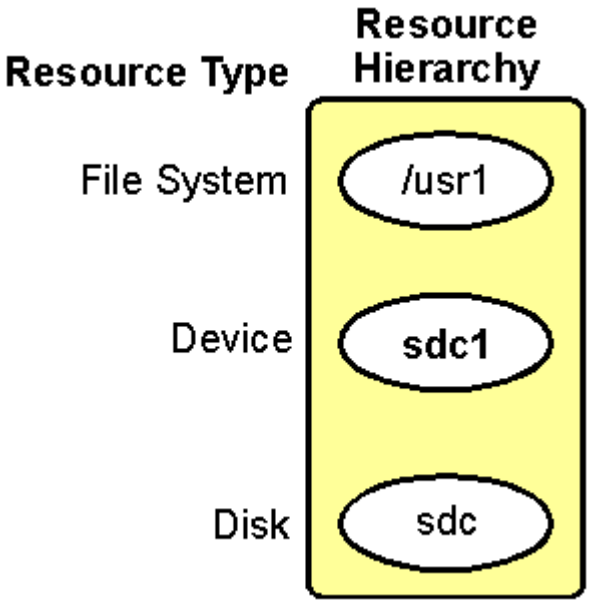
The resource status of each resource is displayed in the Detailed Status Display and the Short Status Display. The LifeKeeper tag names of root resources are displayed beginning in the left-most position of the TAG column, with tag names of resources within the hierarchy indented appropriately to indicate dependency relationships between resources.

The following sample is from the resource hierarchy section of a short status display (the device and disk ID's are shortened to fit in the display area):

LOCAL	TAG	ID	STATE	PRIO	PRIMARY
svr1	app3910-on-svr1	app4238	ISP	1	svr2
svr1	filesys4083	/jrl1	ISP	1	svr2
svr1	device2126	000…300-1	ISP	1	svr2
svr1	disk2083	000…300	ISP	1	svr2

See Resource Hierarchy Example for an illustration of a hierarchy. For more information, see the Resource Hierarchy Information section of Detailed Status Display and Short Status Display.

5.3.1.3.7.6. Resource Hierarchy Example



5.3.1.3.7.7. Detailed Status Display

This topic describes the categories of information provided in the detailed status display as shown in the following example of output from the **lcdstatus** command. For information on how to display this information, see the LCD(1M) man page. At the command line, you can enter either **man lcdstatus** or **man LCD**. For status information available in the LifeKeeper GUI, see Viewing the Status of a Server or Viewing the Status of Resources.

Example of detailed status display:

Resource Hierarchy Information

```
Resource hierarchies for machine "wileecoyote":

ROOT of RESOURCE HIERARCHY

apache-home.fred: id=apache-home.fred app=webserver type=apache
state=ISP

initialize=(AUTORES_ISP) automatic restore to IN-SERVICE by LifeKeeper

info=/home/fred /usr/sbin/httpd

reason=restore action has succeeded

depends on resources:
ipeth0-172.17.104.25,ipeth0-172.17.106.10,ipeth0-172.17.106.105

Local priority = 1

SHARED equivalency with "apache-home.fred" on "roadrunner", priority =
10

FAILOVER ALLOWED

ipeth0-172.17.104.25: id=IP-172.17.104.25 app=comm type=ip state=ISP

initialize=(AUTORES_ISP) automatic restore to IN-SERVICE by LifeKeeper

info=wileecoyote eth0 172.17.104.25 fffffc00

reason=restore action has succeeded
```

```
these resources are dependent: apache-home.fred

Local priority = 1

SHARED equivalency with “ipeth0-172.17.104.25” on “roadrunner”,
priority = 10

FAILOVER ALLOWED

ipeth0-172.17.106.10: id=IP-172.17.106.10 app=comm type=ip state=ISP

initialize=(AUTORES_ISP) automatic restore to IN-SERVICE by LifeKeeper

info=wileecoyote eth0 172.17.106.10 fffffc00

reason=restore action has succeeded

these resources are dependent: apache-home.fred

Local priority = 1

SHARED equivalency with “ipeth0-172.17.106.10” on “roadrunner”,
priority = 10

FAILOVER ALLOWED

ipeth0-172.17.106.105: id=IP-172.17.106.105 app=comm type=ip state=ISP

initialize=(AUTORES_ISP) automatic restore to IN-SERVICE by LifeKeeper

info=wileecoyote eth0 172.17.106.105 fffffc00

reason=restore action has succeeded

These resources are dependent: apache-home.fred

Local priority = 1

SHARED equivalency with “ipeth0-172.17.106.105” on “roadrunner”,
priority = 10

FAILOVER ALLOWED
```

Communication Status Information

```
The following LifeKeeper servers are known:

machine=wileecoyote state=ALIVE

machine=roadrunner state=DEAD (eventslcm detected failure at Wed Jun 7
15:45:14 EDT 2000)

The following LifeKeeper network connections exist:

to machine=roadrunner type=TCP addresses=192.168.1.1/192.168.105.19

state="DEAD" priority=2 #comm_downs=0
```

LifeKeeper Flags

```
The following LifeKeeper flags are on:

shutdown_switchover
```

Shutdown Strategy

```
The shutdown strategy is set to: switchover.
```

Resource Hierarchy Information

LifeKeeper displays the resource status beginning with the root resource. The display includes information about all resource dependencies.

Elements common to multiple resources appear only once under the first root resource. The first line for each resource description displays the resource tag name followed by a colon (:), for example: `device13557:`. These are the information elements that may be used to describe the resources in the hierarchy:

- **id**. Unique resource identifier string used by LifeKeeper.
- **app**. Identifies the type of application, for example the sample resource is a *webserver* application.
- **type**. Indicates the resource class type, for example the sample resource is an *Apache* application.
- **state**. Current state of the resource:
  - ISP—In-service locally and protected.
  - ISU—In-service, unprotected.
  - OSF—Out-of-service, failed.

    - OSU—Out-of-service, unimpaired.

- **initialize**. Specifies the way the resource is to be initialized, for example LifeKeeper restores the application resource, but the host adapter initializes without LifeKeeper.

- **info**. Contains object-specific information used by the object's `remove` and `restore` scripts.

- **reason**. If present, describes the reason the resource is in its current state. For example, an application might be in the OSU state because it is in-service (ISP or ISU) on another server. Shared resources can be active on only one of the grouped servers at a time.

- **depends on resources**. If present, lists the tag names of the resources on which this resource depends.

- **these resources are dependent**. If present, indicates the tag names of all parent resources that are directly dependent on this object.

- **Local priority**. Indicates the failover priority value of the targeted server, for this resource.

- **SHARED equivalency**. Indicates the resource tag and server name of any remote resources with which this resource has a defined equivalency, along with the failover priority value of the remote server, for that resource.

- **FAILOVER ALLOWED**. If present, indicates that LifeKeeper is operational on the remote server identified in the equivalency on the line above, and the application is protected against failure. `FAILOVER INHIBITED` means that the application is not protected due to either the shutting down of LifeKeeper or the stopping of the remote server.

Communication Status Information

This section of the status display lists the servers known to LifeKeeper and their current state, followed by information about each communications path.

These are the communications information elements you can see on the status display:

- **State**. Status of communications path. These are the possible communications state values:
    - ALIVE — Functioning normally
    - DEAD — No longer functioning normally

- **priority**. The assigned priority value for the communications path.This item is displayed only for TCP paths.

- **#comm_downs**. The number of times the port has failed and caused a failover. The path failure causes a failover only if no other communications paths are marked "ALIVE" at the time of the failure.

In addition, the status display can provide any of the following statistics maintained only for TTY communications paths:

- **wrpid**. Each TTY communications path has unique reader and writer processes. The `wrpid` field contains the process ID for the writer process. The writer process sleeps until one of two conditions occurs:
  - Heartbeat timer expires, causing the writer process to send a message.
  - Local process requests the writer process to transmit a LifeKeeper maintenance message to the other server. The writer process transmits the message, using its associated TTY port, to the reader process on that port on the other system.

- **rdpid**. Each TTY communications path has unique reader and writer processes. The `rdpid` field contains the process ID for the reader process. The reader process sleeps until one of two conditions occurs:
  - Heartbeat timer expires and the reader process must determine whether the predefined heartbeat intervals have expired. If so, the reader process marks the communications path in the DEAD state, which initiates a failover event if there are no other communications paths marked ALIVE.
  - Remote system writer process transmits a LifeKeeper maintenance message, causing the reader process to perform the protocol necessary to receive the message.

- **#NAKs**. Number of times the writer process received a negative acknowledgment (NAK). A `NAK` message means that the reader process on the other system did not accept a message packet sent by the writer process, and the writer process had to re-transmit the message packet. The #NAKs statistic can accumulate over a long period of time due to line noise. If, however, you see the numbers increasing rapidly, you should perform diagnostic procedures on the communications subsystem.

- **#chksumerr**. Number of mismatches in the check sum message between the servers. This statistic can accumulate over a long period of time due to line noise. If, however, you see the numbers increasing rapidly, you should perform diagnostic procedures on the communications subsystem.

- **#incmpltmes**. Number of times the incoming message packet did not match the expected size. A high number of mismatches may indicate that you should perform diagnostic procedures on the hardware port associated with the communications path.

- **#noreply**. Number of times the writer process timed out while waiting for an acknowledgment and had to re-transmit the message. Lack of acknowledgment may indicate an overloaded server or it can signal a server failure.

- **#pacresent**. Number of times the reader process received the same packet. This can happen when the writer process on the sending server times out and resends the same message.

- **#pacoutseq**. Number of times the reader received packets out of sequence. High numbers in this field can indicate lost message packets and may indicate that you should perform diagnostic

procedures on the communications subsystem.

- **#maxretrys**. Metric that increments for a particular message when the maximum retransmission count is exceeded (for `NAK` and `noreply` messages). If you see a high number in the `#maxretrys` field, you should perform diagnostic procedures on the communications subsystem.

LifeKeeper Flags

Near the end of the detailed status display, LifeKeeper provides a list of the flags set for the system. A common type is a Lock LCD flag used to ensure that other processes wait until the process lock completes its action. The following is the standard LCD lock format:

```
!action!processID!time!machine:id.
```

These are examples of general LCD lock flags:

- `!action!02833!701236710!server1:`*filesys* – The creation of a file system hierarchy produces a flag in this format in the status display. The *filesys* designation can be a different resource type for other application resource hierarchies, or `app` for generic or user-defined applications.

- Other typical flags include `!nofailover!`*machine*, `!notarmode!`*machine*, and `shutdown_switchover`. The `!nofailover!`*machine* and `!notarmode!`*machine* flags are internal, transient flags created and deleted by LifeKeeper, which control aspects of server failover. The `shutdown_switchover` flag indicates that the shutdown strategy for this server has been set to *switchover* such that a shutdown of the server will cause a switchover to occur. See the LCDI-flag(1M) for more detailed information on the possible flags.

Shutdown Strategy

The last item on the detailed status display identifies the LifeKeeper shutdown strategy selected for this system. See Setting Server Shutdown Strategy for more information.

5.3.1.3.7.8. Short Status Display

This topic describes the categories of information provided in the short status display as shown in the following example of output from the **lcdstatus -e** command. For information on how to display this information, see the LCD man page. At the command line, you can enter either **man lcdstatus** or **man LCD**. For status information available in the LifeKeeper GUI, see Viewing the Status of a Server or Viewing the Status of Resources.

Example of Short Status Display:

Resource Hierarchy Information

BACKUP	TAG	ID	STATE	PRIO	PRIMARY
svr1	appfs3910-on-svr1	appfs4238	ISP	1	svr2
svr1	filesys4083	/jrl	ISP	1	svr2
svr1	device2126	1000…300-1	ISP	1	svr2
svr1	disk2083	000…300	ISP	1	svr2

Communication Status Information

MACHINE	NETWORK	ADDRESSES/DEVICE	STATE	PRIO
svr1	TCP	100.10.1.20/100.11.1.21	ALIVE	1
svr1	TTY	/dev/ttyS0	ALIVE	--

Resource Hierarchy Information

LifeKeeper displays the resource status of each resource. The LifeKeeper tag names of root resources are displayed beginning in the left-most position of the **TAG** column, with tag names of resources within the hierarchy indented appropriately to indicate dependency relationships between resources.

The **BACKUP** column indicates the next system in the failover priority order, after the system for which the status display pertains. If the target system is the lowest priority system for a given resource, the **BACKUP** column for that resource contains dashes (for example, ------).

- **TAG column**. Contains the root tag for the resource.
- **ID column**. Contains each resource's identifier string.
- **STATE column**. Contains the current state of each resource, as described in Resource States.
- **PRIO column**. Contains the failover priority value of the local server, for each resource.
- **PRIMARY column**. Contains the name of the server with the highest priority, for each resource.

Communication Status Information

This section of the display lists each communications path that has been defined on the target system. For each path, the following information is provided.

- **MACHINE**. Remote server name for the communications path.
- **NETWORK**. The type of communications path (TCP or TTY)
- **ADDRESSES/DEVICE**. The pair of IP addresses or device name for the communications path
- **STATE**. The state of the communications path (ALIVE or DEAD)
- **PRIO**. For TCP paths, the assigned priority of the path. For TTY paths, this column will contain dashes
  (----), since TTY paths do not have an assigned priority.

5.3.1.4. Fault Detection and Recovery Scenarios

To demonstrate how the various LifeKeeper components work together to provide fault detection and recovery, see the following topics that illustrate and describe three types of recovery scenarios:

5.3.1.4.1. IP Local Recovery

SIOS recommends the use of bonded interfaces via the standard Linux NIC bonding mechanism in any LifeKeeper release where a backup interface is required. Beginning with LifeKeeper Release 7.4.0, bonded interfacing is the only supported method. For releases prior to 7.4.0, the backup interface feature in the IP kit, described below, can be used.

The IP local recovery feature allows LifeKeeper to move a protected IP address from the interface on which it is currently configured to another interface in the same server when a failure has been detected by the IP Recovery Kit. Local recovery provides you an optional backup mechanism so that when a particular interface fails on a server, the protected IP address can be made to function on the backup interface, therefore avoiding an entire application/resource hierarchy failing over to a backup server.

Local Recovery Scenario

IP local recovery allows you to specify a single backup network interface for each LifeKeeper-protected IP address on a server. In order for the backup interface to work properly, it must be attached to the same physical network as the primary interface. The system administrator is expected to insure that a valid interface is being chosen. Note that it is completely reasonable and valid to specify a backup interface on one server but not on another within the cluster (i.e. the chosen backup interface on one server has no impact on the choice of a backup on any other server).

When a failure of an IP address is detected by the IP Recovery Kit, the resulting failure triggers the execution of the IP local recovery script. LifeKeeper first attempts to bring the IP address back in service on the current network interface. If that fails, LifeKeeper checks the resource instance to determine if there is a backup interface available. If so, it will then attempt to move the IP address to the backup interface. If all local recovery attempts fail, LifeKeeper will perform a failover of the IP address and all dependent resources to a backup server.

The backup interface name can be identified in the Information field of the IP resource instance. The Information field values are space-separated and are, in order, the primary server name, the network interface name, the IP address, the netmask and the backup interface name. Here is an example:

ServerA eth0 172.17.106.10 fffffc00 eth1

If no backup interface is configured, the 5th field value will be set to **none**.

When the protected IP address is moved to the backup interface, the 2nd and 5th field values are swapped so that the original backup interface becomes the primary and vice versa. The result is that during LifeKeeper startups, switchovers and failovers, LifeKeeper always attempts to bring the IP address in service on the interface on which it was last configured.

Command Line Operations

In LifeKeeper for Linux v3.01 or later, the mechanism for adding or removing a backup interface from an

existing IP resource instance is provided as a command line utility. This capability is provided by the `lkipbu` utility. The command and syntax are:

```
lkipbu [-d machine] -{a|r} -t tag -f interface
```

The `add` operation (specified via the -a option) will fail if a backup interface has already been defined for this instance or if an invalid interface name is provided. The remove operation (specified via the -r option) will fail if the specified interface is not the current backup interface for this instance.

A command line mechanism is also provided for manually moving an IP address to its backup interface. This capability is specified via the -m option using the following syntax:

```
lkipbu [-d machine] -m -t tag
```

This operation will fail if there is no backup interface configured for this instance. If the specified resource instance is currently in service, the move will be implemented by using the `ipaction remove` operation to un-configure the IP address on the current interface, and `ipaction restore` to configure it on the backup interface. Following the move, the `execute_broadcast_ping` function will be used to verify the operation of the address on the new interface, and if successful, the interface values will be swapped in the IP resource instance *INFO* field. If the specified IP resource instance is out-of-service when this command is executed, the primary and backup interface values will simply be swapped in the *INFO* field.

The `lkipbu` utility also provides an option for retrieving the currently defined primary and backup interfaces for a specified IP resource instance along with the state of the resource on the primary interface (up or down). This capability is specified via the -s option using the following syntax:

```
lkipbu [-d machine] -s -t tag
```

The output will be similar to the following:

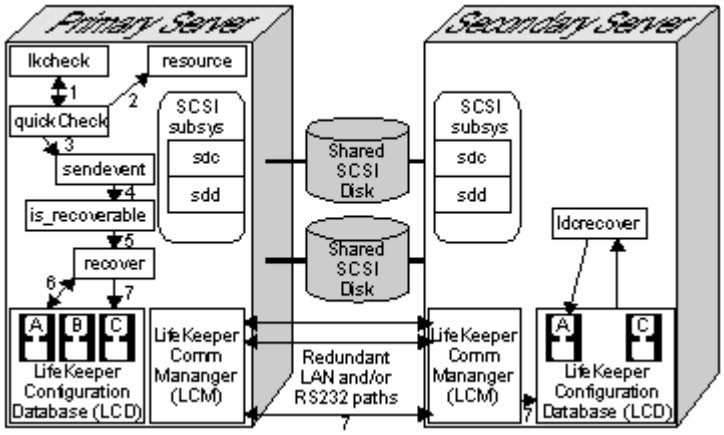
```
IP address: 172.17.106.10

Netmask: 255.255.252.0

Primary interface: eth0 (up)

Backup interface: eth1
```

Refer to the lkipbu(8) man page for further detail.

5.3.1.4.2. Resource Error Recovery Scenario

LifeKeeper provides a real-time daemon monitor, **lkcheck**, to check the status and health of LifeKeeper-protected resources. For each in-service resource, **lkcheck** periodically calls the quickCheck script for that resource type. The **quickCheck** script performs a quick health check of the resource, and if the resource is determined to be in a failed state, the **quickCheck** script calls the event notification mechanism, sendevent.

The following figure illustrates the recovery process tasks when **lkcheck** initiates the process:



1. **lkcheck** runs. By default, the **lkcheck** process runs once every two minutes. When **lkcheck** runs, it invokes the appropriate **quickCheck** script for each in-service resource on the system.

2. **quickCheck** script checks resource. The nature of the checks performed by the **quickCheck** script is unique to each resource type. Typically, the script simply verifies that the resource is available to perform its intended task by imitating a client of the resource and verifying that it receives the expected response.

3. **quickCheck** script invokes **sendevent**. If the **quickCheck** script determines that the resource is in a failed state, it initiates an event of the appropriate class and type by calling **sendevent**.

4. Recovery instruction search. The system event notification mechanism, **sendevent**, first attempts to determine if the LCD has a resource and/or recovery for the event type or component. To make this determination, the is_recoverable process scans the resource hierarchy in LCD for a resource instance that corresponds to the event (in this example, the filesys name).

   The action in the next step depends upon whether the scan finds resource-level recovery instructions:

   - Not found. If resource recovery instructions are not found, is_recoverable returns to **sendevent** and **sendevent** continues with basic event notification.

   - Found. If the scan finds the resource, is_recoverable forks the recover process into the background. The is_recoverable process returns and **sendevent** continues with basic event

notification, passing an advisory flag "*-A*" to the basic alarming event response scripts, indicating that LifeKeeper is performing recovery.

5. Recover process initiated. Assuming that recovery continues, is_recoverable initiates the recover process which first attempts local recovery.

6. Local recovery attempt. If the instance was found, the recover process attempts local recovery by accessing the resource hierarchy in LCD to search the hierarchy tree for a resource that knows how to respond to the event. For each resource type, it looks for a recovery subdirectory containing a subdirectory named for the event class, which in turn contains a recovery script for the event type.

   The recover process runs the recovery script associated with the resource that is farthest above the failing resource in the resource hierarchy. If the recovery script succeeds, recovery halts. If the script fails, recover runs the script associated with the next resource, continuing until a recovery script succeeds or until recover attempts the recovery script associated with the failed instance.

   If local recovery succeeds, the recovery process halts.

7. Inter-server recovery begins. If local recovery fails, the event then escalates to inter-server recovery.

8. Recovery continues. Since local recovery fails, the recover process marks the failed instance to the *Out-of-Service-FAILED* (OSF) state and marks all resources that depend upon the failed resource to the *Out-of-Service-UNIMPAIRED* (OSU) state. The recover process then determines whether the failing resource or a resource that depends upon the failing resource has any shared equivalencies with a resource on any other systems,and selects the one to the highest priority alive server. Only one equivalent resource can be active at a time.

   If no equivalency exists, the recover process halts.

   If a shared equivalency is found and selected, LifeKeeper initiates inter-server recovery. The recover process sends a message through the LCM to the LCD process on the selected backup system containing the shared equivalent resource. This means that LifeKeeper would attempt inter-server recovery.

9. **lcdrecover** process coordinates transfer. The LCD process on the backup server forks the process **lcdrecover** to coordinate the transfer of the equivalent resource.

10. Activation on backup server. The **lcdrecover** process finds the equivalent resource and determines whether it depends upon any resources that are not in-service. **lcdrecover** runs the restore script (part of the resource recovery action scripts) for each required resource, placing the resources in-service.

    The act of restoring a resource on a backup server may result in the need for more shared resources to be transferred from the primary system. Messages pass to and from the primary

system, indicating resources that need to be removed from service on the primary server and then brought into service on the selected backup server to provide full functionality of the critical applications. This activity continues, until no new shared resources are needed and all necessary resource instances on the backup are restored.

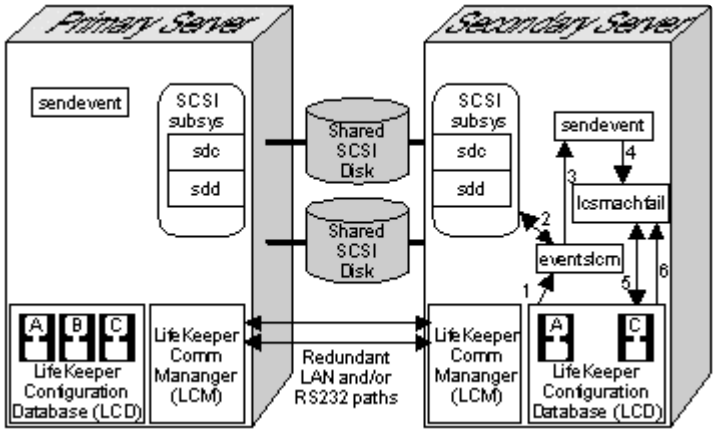
5.3.1.4.3. Server Failure Recovery Scenario

The LifeKeeper Communications Manager (LCM) has two functions:

- Messaging. The LCM serves as a conduit through which LifeKeeper sends messages during recovery, configuration, or when running an audit.
- Failure detection. The LCM also plays a role in detecting whether or not a server has failed.

LifeKeeper has a built-in heartbeat signal that periodically notifies each server in the configuration that its paired server is operating. If a server fails to receive the heartbeat message through one of the communications paths, LifeKeeper marks that path DEAD.

The following figure illustrates the recovery tasks when the LCM heartbeat mechanism detects a server failure.



The following steps describe the recovery scenario, illustrated above, if LifeKeeper marks all communications connections to a server DEAD.

1. LCM activates **eventslcm**. When LifeKeeper marks all communications paths dead, the LCM initiates the **eventslcm** process.

   Only one activity stops the **eventslcm** process:

   - Communication path alive. If one of the communications paths begins sending the heartbeat signal again, the LCM stops the **eventslcm** process.

   It is critical that you configure two or more physically independent, redundant communication paths between each pair of servers to prevent failovers and possible system panics due to communication failures.

2. Message to sendevent. **eventslcm** sends the system failure alarm by calling **sendevent** with the event type *machfail*.

3. sendevent initiates failover recovery. The sendevent program determines that LifeKeeper can

handle the system failure event and executes the LifeKeeper failover recovery process **lcdmachfail**.

4. **lcdmachfail** checks. The **lcdmachfail** process first checks to ensure that the non-responding server was not shut down. Failovers are inhibited if the other system was shut down gracefully before system failure. Then **lcdmachfail** determines all resources that have a shared equivalency with the failed system. This is the commit point for the recovery.

5. **lcdmachfail** restores resources. **lcdmachfail** determines all resources on the backup server that have shared equivalencies with the failed primary server. It also determines whether the backup server is the highest priority alive server for which a given resource is configured. All backup servers perform this check, so that only one server will attempt to recover a given hierarchy. For each equivalent resource that passes this check, **lcdmachfail** invokes the associated restore program. Then, **lcdmachfail** also restores each resource dependent on a restored resource, until it brings the entire hierarchy into service on the backup server.

5.3.2. Installation and Configuration

SPS for Linux Installation

For complete installation instructions on installing the SPS for Linux software, see the SPS for Linux Installation Guide. Refer to the SPS for Linux Release Notes for additional information.

SPS for Linux Configuration

Once the SPS environment has been installed, the SPS software can be configured on each server in the cluster. Follow the steps in ***SPS Configuration Steps*** below which contains links to topics with additional details.

Configuration Steps

Event Forwarding via SNMP

Event Email Notification

Optional Configuration Tasks

Linux Configuration

Data Replication Configuration

Network Configuration

Application Configuration

Storage and Adapter Configuration

Fencing

Resource Policy Management

Configuring Credentials

5.3.2.1. SPS Configuration Steps

If you have installed your SPS environment as described in the SPS Installation Guide, you should be ready to start and configure the SPS software on each server in your cluster.

Follow the steps below which contain links to topics with additional details. Perform these tasks on each server in the cluster.

1. Start LifeKeeper by typing the following command as root:

   ```
   /etc/init.d/lifekeeper start
   ```

   or

   ```
   systemctl start lifekeeper
   ```

   This command starts all LifeKeeper daemon processes on the server being administered if they are not currently running.

   For additional information on starting and stopping LifeKeeper, see Starting LifeKeeper and Stopping LifeKeeper.

2. Set Up TTY Communications Connections. If you plan to use a TTY communications (comm) path for a LifeKeeper heartbeat, you need to set up the physical connection for that heartbeat.

3. Configure the GUI. There are multiple tasks involved with configuring the GUI. Start with the LifeKeeper GUI – Overview topic within Preparing to Run the GUI. Then for detailed instructions, follow the browse sequence throughout Preparing to Run the GUI.

   Note: The first time you run the LifeKeeper GUI, you will see a QuickStart button which opens a window with instructions and links to help you step through the configuration of your LifeKeeper resources. Subsequently, you can access this QuickStart Configuration Assistant under the Help menu.

4. Create Communication Paths. Before you can activate LifeKeeper protection, you must create the communications path (heartbeat) definitions within LifeKeeper.

5. Perform any of the following optional configuration tasks:

   - Set the Server Shutdown Strategy
   - Configure the manual failover confirmation option
   - Tune the LifeKeeper heartbeat
   - Configure SNMP Event Forwarding via SNMP

- ◦ Configure Event Email Notification

- ◦ If you plan to use STONITH devices in your cluster, create the scripts to control the STONITH devices and place them in the appropriate LifeKeeper events directory.

6. SPS is now ready to protect your applications. The next step depends on whether you will be using one of the optional SPS Recovery Kits:

- ◦ If you are using an SPS Recovery Kit, refer to the Documentation associated with the kit for instructions on creating and extending your resource hierarchies.

- ◦ If you are using an application that does not have an associated Recovery Kit, then you have two options:

    - ◦ If it is a simple application, you should carefully plan how to create an interface between your application and LifeKeeper. You may decide to protect it using the Generic Application Recovery Kit included with the LifeKeeper core.

    - ◦ Services provided by the operating system can easily be protected by using the Quick Service Protection (QSP) Recovery Kit include with the LifeKeeper Core. However, please be aware that quickCheck will only perform a simple check of the service state.

5.3.2.1.1. Set Up TTY Connections

If you plan to use a TTY communications (comm) path for a LifeKeeper heartbeat, you need to set up the physical connection for that heartbeat. Remember that multiple communication paths are required to avoid false failover due to a simple communications failure. Two or more LAN-based (TCP) comm paths should also be used.

Connect the TTY cable to the serial ports of each server to be used for the serial heartbeat.

1. Test the serial path using the following command:

   /opt/LifeKeeper/bin/portio -r -p port -b baud

   where:

   - **baud** is the baud rate selected for the path (normally 9600)
   - **port** is the serial port being tested on Server 1, for example */dev/ttyS0*.
   - Server 1 is now waiting for input from Server 2.

2. Run command **portio** on Server 2. On the second system in the pair, type the following command:

   echo Helloworld | /opt/LifeKeeper/bin/portio -p port -b baud

   where:

   - **baud** is the same baud rate selected for Server 1.
   - **port** is the serial port being tested on Server 2, for example */dev/ttyS0*.

3. View the console. If the communications path is operational, the software writes “Helloworld” on the console on Server 1. If you do not see that information, perform diagnostic and correction operations before continuing with LifeKeeper configuration.

5.3.2.2. LifeKeeper Event Forwarding via SNMP

5.3.2.2.1. Overview of LifeKeeper Event Forwarding via SNMP

The Simple Network Management Protocol (SNMP) defines a device-independent framework for managing networks. Devices on the network are described by MIB (Management Information Base) variables that are supplied by the vendor of the device. An SNMP agent runs on each node of the network, and interacts with a Network Manager node. The Network Manager can query the agent to get or set the values of its MIB variables, there by monitoring or controlling the agent's node. The agent can also asynchronously generate messages called traps to notify the manager of exceptional events. There are a number of applications available for monitoring and managing networks using the Simple Network Management Protocol (SNMP).

LifeKeeper has an event notification mechanism for registering applications that wish to be notified of specific events or alarms (see the sendevent(5) man page). LifeKeeper can be easily enabled to send SNMP trap notification of key LifeKeeper events to a third party network management console wishing to monitor LifeKeeper activity.

The remote management console receiving SNMP traps must first be configured through the administration software of that system; LifeKeeper provides no external SNMP configuration. The remote management server is typically located outside of the LifeKeeper cluster (i.e., it is not a LifeKeeper node).

LifeKeeper Events Table

The following table contains the list of LifeKeeper events and associated trap numbers. The entire Object ID (OID) consists of a prefix followed by a specific trap number in the following format:

```
prefix.0.specific trap number
```

The prefix is **.1.3.6.1.4.1.7359**, which expands to **iso.org.dod.internet.private.enterprises.7359** in the MIB tree. (7359 is SteelEye's [SIOS Technology] enterprise number, followed by 1 for LifeKeeper.) For example, the LifeKeeper Startup Complete event generates the OID: **.1.3.6.1.4.1.7359.1.0.100**.

LifeKeeper Event/Description	Trap #	Object ID
LifeKeeper Startup Complete Sent from a node when LifeKeeper is started on that node	100	.1.3.6.1.4.1.7359.1.0.100
LifeKeeper Shutdown Initiated	101	.1.3.6.1.4.1.7359.1.0.101

Sent from a node beginning LifeKeeper shutdown		
LifeKeeper Shutdown Complete Sent from a node completing LifeKeeper shutdown	102	.1.3.6.1.4.1.7359.1.0.102
LifeKeeper Manual Switchover Initiated on Server Sent from the node from which a manual switchover was requested	110	.1.3.6.1.4.1.7359.1.0.110
LifeKeeper Manual Switchover Complete – recovered list Sent from the node where the manual switchover was completed	111	.1.3.6.1.4.1.7359.1.0.111
LifeKeeper Manual Switchover Complete – failed list Sent from each node within the cluster where the manual switchover failed	112	.1.3.6.1.4.1.7359.1.0.112
LifeKeeper Node Failure Detected for Server Sent from each node within the cluster when a node in that cluster fails	120	.1.3.6.1.4.1.7359.1.0.120
LifeKeeper Node Recovery Complete for Server – recovered list Sent from each node within the cluster that has recovered resources from the failed node	121	.1.3.6.1.4.1.7359.1.0.121
LifeKeeper Node Recovery Complete for Server – failed list Sent from each node within the cluster that has failed to recover resources from the failed node	122	.1.3.6.1.4.1.7359.1.0.122
LifeKeeper Resource Recovery Initiated	130	.1.3.6.1.4.1.7359.1.0.130

Sent from a node recovering a resource; a 131 or 132 trap always follows to indicate whether the recovery was completed or failed.		
LifeKeeper Resource Recovery Failed Sent from the node in trap 130 when the resource being recovered fails to come into service	131*	.1.3.6.1.4.1.7359.1.0.131
LifeKeeper Resource Recovery Complete Sent from the node in trap 130 when the recovery of the resource is completed	132	.1.3.6.1.4.1.7359.1.0.132
LifeKeeper Communications Path Up A communications path to a node has become operational	140	.1.3.6.1.4.1.7359.1.0.140
LifeKeeper Communications Path Down A communications path to a node has gone down	141	.1.3.6.1.4.1.7359.1.0.141
LifeKeeper <Node Monitoring> Failure Sent from a node where a failure was detected with Node Monitoring of the Standby Node Health Check. Detected failure is described in <Node Monitoring>.	190	.1.3.6.1.4.1.7359.1.0.190
LifeKeeper <OSUquickCheck> Failure Sent from a node where a failure was detected with OSU Resource Monitoring of the Standby Node Health Check. Tag name of the resource where the failure was detected is described in <OSUquickCheck>.	200	.1.3.6.1.4.1.7359.1.0.200
The following variables are used to “carry” additional information in the trap PDU:		
Trap message	all	.1.3.6.1.4.1.7359.1.1
Resource Tag	130	.1.3.6.1.4.1.7359.1.2
Resource Tag	131	.1.3.6.1.4.1.7359.1.2

Resource Tag	132	.1.3.6.1.4.1.7359.1.2
List of recovered resources	111	.1.3.6.1.4.1.7359.1.3
List of recovered resources	121	.1.3.6.1.4.1.7359.1.3
List of failed resources	112	.1.3.6.1.4.1.7359.1.4
List of failed resources	122	.1.3.6.1.4.1.7359.1.4

* This trap may appear multiple times if recovery fails on multiple backup servers.

5.3.2.2.2. Configuring LifeKeeper Event Forwarding

Prerequisites

The SNMP event forwarding feature is included as part of the LifeKeeper Core functionality and does not require additional LifeKeeper packages to be installed. Since LifeKeeper uses the snmptrap utility to generate the traps, the snmptrap command is required to be installed on the node that will generate SNMP notification.
The snmptrap utility is provided by the following packages:
RHEL 5 or later and supported operating systems – net-snmp-utils
SLES 11 or later – net-snmp

In older versions of the snmp implementation (prior to 4.1) where the `defCommunity` directive is not supported, the traps will be sent using the “public” community string.

It is not necessary to have an SNMP agent snmpd running on the LifeKeeper node.

The configuration of a trap handler on the network management console and its response to trap messages is beyond the scope of this LifeKeeper feature. See the documentation associated with your system management tool for related instructions.

Configuration Tasks

The following tasks must be performed to set up LifeKeeper SNMP Event Forwarding. All but the last task must be repeated on each node in the LifeKeeper cluster that will be generating SNMP trap messages.

1. Ensure that the `snmptrap` utility is available as noted above.

2. Specify the network management node to which the SNMP traps will be sent. This can be done either by command line or by editing the `/etc/default/LifeKeeper` file. You must specify the IP address rather than domain name to avoid DNS issues.

   - ◦ By command line, use the `lk_configsnmp` (see the `lk_configsnmp(1m)` man page for details). This utility will only accept IP addresses.

   - ◦ Or, edit the defaults file `/etc/default/LifeKeeper` to add the IP address. Find the entry `LK_TRAP_MGR=` and insert one or more IP addresses (separated by commas) to the right of “=” (no white space before or after “=” or commas).

3. If you are using an older version of the snmp implementation that does not support the `defCommunity` directive, skip this step. Traps will be sent using the “public” community string. Otherwise, do the following:

Specify a default community in `/usr/share/snmp/snmp.conf`. If this file does not exist, create it using sufficiently secure permissions. Add the directive "**`defCommunity`**" with a value. This specifies the SNMP version 2c community string to use when sending traps. For example, add a line like this:

```
defCommunity myCommunityString
```

Refer to the `snmp.conf` man page (delivered with the snmp package) for more information about this configuration file.

4. Perform whatever configuration steps are needed on the remote management console to detect and respond to the incoming trap OIDs from LifeKeeper events. If the management node is a Linux server, the minimum that you would need to do to begin verification of this feature would be to start the `snmptrapd` daemon with the -f -Lo option (print the messages to `stdout`).

Verifying the Configuration

To verify that the configuration is working, initiate a LifeKeeper action (for example, start or stop LifeKeeper, or bring a resource in-service manually using the LifeKeeper GUI). Verify that the trap message was received at the management console. If a trap is not received, inspect the appropriate log files on the management system, and follow the normal troubleshooting practices provided with the management software. The LifeKeeper log can be inspected to determine if there was a problem sending the trap message. See SNMP Troubleshooting for more information.

Disabling SNMP Event Forwarding

To disable the generation of SNMP traps by LifeKeeper, simply remove the assignment of an IP address from the `LK_TRAP_MGR` environment variable in the file `/etc/default/LifeKeeper`. This can be accomplished using the `lk_configsnmp` utility from the command line with the "disable"option (see the `lk_configsnmp(1M)` page for an example). Or, edit `/etc/default/LifeKeeper` and change the entry for **`LK_TRAP_MGR`** to `LK_TRAP_MGR=` (or remove the line entirely). This must be done on each node that should be disabled from sending trap messages.

5.3.2.2.3. SNMP Troubleshooting

Following are some possible problems and solutions related to SNMP Event Forwarding. For specific error messages, see the LifeKeeper Message Catalog.

Problem: No SNMP trap messages are sent from LifeKeeper.

Solution: Verify that the snmptrap utility is installed on the system (it is usually located in /usr/bin). If it is not installed, install the appropriate snmp package (see Prerequisites). If it is installed in someother location, edit the PATH variable in the file /etc/default/LifeKeeper and add the appropriate path.

Problem: No SNMP error messages are logged and SNMP trap messages do not appear to be sent from a LifeKeeper server.

Solution: Check to see if `LK_TRAP_MGR` is set to the IP address of the network management server that will receive the traps. From the command line, use the **`lk_configsnmp`** utility with the “query” option to verify the setting (See the `lk_configsnmp(1M)` man page for an example.) Or, search for the entry for `LK_TRAP_MGR` in the file `/etc/default/LifeKeeper`. This variable must be set on each LifeKeeper node that will generate SNMP trap messages.

5.3.2.3. LifeKeeper Event Email Notification

5.3.2.3.1. Overview of LifeKeeper Event Email Notification

LifeKeeper Event Email Notification is a mechanism by which one or more users may receive email notices when certain events occur in a LifeKeeper cluster. LifeKeeper has an event notification mechanism for registering applications that wish to be notified of specific events or alarms (see the sendevent(5) man page). LifeKeeper can be easily enabled to send email notification of key LifeKeeper events to a selected set of users wishing to monitor LifeKeeper activity. Additionally, a log of each email notice issued is available via `/var/log/lifekeeper.log` or by using the Viewing Server Log Files facility in the LifeKeeper GUI. The messages may be found in the **`NOTIFY`** log.

By default, LifeKeeper Event Email Notification is disabled. Enabling this feature requires setting the `LK_NOTIFY_ALIAS` environment variable defined in `/etc/default/LifeKeeper`. The `LK_NOTIFY_ALIAS` environment variable can be set to a single email address or alias, or it can contain multiple addresses or aliases separated by commas. To set `LK_NOTIFY_ALIAS` either run `lk_confignotify alias` (See the `lk_confignotifyalias(1M)` man page for an example) from the command line and supply the address or list of addresses that should receive email when an event occurs or edit the defaults file `/etc/default/LifeKeeper` to add the email address or address list. Search for the entry `LK_NOTIFY_ALIAS=` and insert the address or address list separated by commas. Repeat this action on all nodes in the cluster that need to send email for the selected LifeKeeper events.

To disable Email Notification, either run **`lk_confignotifyalias`** (See the `lk_confignotifyalias(1M)` man page for an example) with the *—disable* argument or edit the defaults file `/etc/default/LifeKeeper` and remove the setting of `LK_NOTIFY_ALIAS` (change the line to `LK_NOTIFY_ALIAS=`).

LifeKeeper Events Generating Email

The following LifeKeeper events will generate email notices when `LK_NOTIFY_ALIAS` is set.

LifeKeeper Event	Event Description
LifeKeeper Startup Complete	Sent from a node when LifeKeeper is started on that node.
LifeKeeper Shutdown Initiated	Sent from a node beginning LifeKeeper shutdown.
LifeKeeper Shutdown Complete	Sent from a node completing LifeKeeper shutdown.
LifeKeeper Manual Switchover Initiated on Server	Sent from the node from which a manual switchover was requested.
LifeKeeper Manual Switchover Complete – recovered list	Sent from the node where the manual switchover was completed listing the resource successfully recovered.

LifeKeeper Manual Switchover Complete – failed list	Sent from the node where the manual switchover was completed listing the resource that failed to successfully switchover.
LifeKeeper Node Failure Detected	Sent from each node within the cluster when a node in that cluster fails.
LifeKeeper Node Recovery Complete for Server – recovered list	Sent from each node within the cluster that has recovered resources from the failed node listing the resource successfully recovered.
LifeKeeper Node Recovery Complete for Server – failed list	Sent from each node within the cluster that has failed to recover resources from the failed node listing the resource that failed to successfully recover.
LifeKeeper Resource Recovery Initiated	Sent from a node recovering a resource; a “Resource Recovery Complete” or “Resource Recovery Failed” message always follows to indicate whether the recovery was completed or failed.
LifeKeeper Resource Recovery Complete	Sent from the node that issued a “Resource Recovery Initiated” message when the recovery of the resource is completed listing the resource successfully recovered.
LifeKeeper Resource Recovery Failed	Sent from the node that issued a “Resource Recovery Initiated” message if the resource fails to come into service listing the resource that failed to successfully recover.
LifeKeeper Communications Path Up	A communications path to a node has become operational.
LifeKeeper Communications Path Down	A communications path to a node has gone down.
LifeKeeper <Node Monitoring> Failure Detected	Sent from a node where a failure was detected with Node Monitoring of the Standby Node Health Check. Detected failure is described in <Node Monitoring>.
LifeKeeper <OSUquickCheck> Failure Detected	Sent from a node where a failure was detected with OSU resource monitoring of the Standby Node Health Check. Tag name of the resource where the failure was detected is described in <OSUquickCheck>.

5.3.2.3.2. Configuring LifeKeeper Event Email Notification

Prerequisites

The Event Email Notification feature is included as part of the LifeKeeper core functionality and does not require additional LifeKeeper packages to be installed. It does require that email software be installed and configured on each LifeKeeper node that will generate email notification of LifeKeeper events. LifeKeeper uses the mail utility, usually installed by the mailx package to send notifications.

The configuration of email is beyond the scope of this LifeKeeper feature. By default, LifeKeeper Event Email Notification is disabled.

Configuration Tasks

The following tasks must be performed to set up LifeKeeper Event Email Notification.

1. Ensure that the mail utility is available as noted above.

2. Identify the user or users that will receive email notices of LifeKeeper events and set `LK_NOTIFY_ALIAS` in the LifeKeeper defaults file `/etc/default/LifeKeeper`. This can be done either from the command line or by editing the file `/etc/default/LifeKeeper` and specifying the email address or alias or the list of email addresses or aliases that should receive notification.

- ◦ From the command line, use the `lk_confignotifyalias` utility (see the `lk_confignotifyalias(1M)` man page for details). This utility will only accept email addresses or aliases separated by commas.

- ◦ Or, edit the defaults file `/etc/default/LifeKeeper` to add the email address or alias. Search for the entry `LK_NOTIFY_ALIAS`= and insert the email address or alias (single or list separated by commas) to the right of the = (no white space around the =).

Verifying the Configuration

To verify that the configuration is working, initiate a LifeKeeper action (for example, start or stop LifeKeeper or bring a resource in-service manually using the LifeKeeper GUI). Verify that an email message was received by the users specified in `LK_NOTIFY_ALIAS` in the file `/etc/default/LifeKeeper` and a message was logged in the LifeKeeper log file. If an email message has not been received, follow your normal debugging procedures for email failures. The LifeKeeper log can be inspected to determine if there was a problem sending the email message. See Email Notification Troubleshooting for more information.

Disabling Event Email Notification

To disable the generation of email notices by LifeKeeper, simply remove the assignment of an email address or alias from the `LK_NOTIFY_ALIAS` environment variable in the file `/etc/default/LifeKeeper`. This can be accomplished using the `lk_confignotifyalias` utility from the command line with the *"—disable"* option (see the `lk_confignotifyalias(1M)` page for an example). Or, edit `/etc/default/LifeKeeper` and change the entry for `LK_NOTIFY_ALIAS` to `LK_NOTIFY_ALIAS=`. This must be done on each node that should be disabled from sending email messages.

5.3.2.3.3. Email Notification Troubleshooting

Following are some possible problems and solutions related to email notification of LifeKeeper events. For specific error messages, see the LifeKeeper Message Catalog.

Problem: No email messages are received from LifeKeeper.

Solution: Verify that the mail utility is installed on the system (it is usually located in */bin/mail*). If it is not installed, install the mailx package. If it is installed in some other location, edit the PATH variable in the file `/etc/default/LifeKeeper` and add the path to the mail utility.

Problem: No email messages are received from LifeKeeper.

Solution: Check the email configuration and ensure email messages have not be queued for delivery indicating a possible email configuration problem. Also ensure that the email address or addresses specified in `LK_NOTIFY_ALIAS` are valid and are separated by a comma.

Problem: The log file has a “mail returned” error message.

Solution: There was some problem invoking or sending mail for a LifeKeeper event, such as a “node failure”, as the mail command return the error X. Verify the mail configuration and that `LK_NOTIFY_ALIAS` contains a valid email address or list of addresses separated by a comma and ensure that email can be sent to those addresses by sending email from the command line using the email recipient format defined in `LK_NOTIFY_ALIAS`.

Problem: No messages, success or failure, are logged and the user or users designated to receive email have not received any mail when a LifeKeeper Event has occurred, such as a node failure.

Solution: Check to see if `LK_NOTIFY_ALIAS` is, in fact, set to an email address or list of addresses separated by commas. From the command line, use the `lk_confignotifyalias` utility with the *“—query”* option to verify the setting (See the `lk_configotifyalias(1M)` man page for an example.) Or, search for the entry `LK_NOTIFY_ALIAS` in the file `/etc/default/LifeKeeper`. This variable must be set on each LifeKeeper node that will generate email notification messages. Also, see the Overview of LifeKeeper Event Email Notification to see if the LifeKeeper event generates an email message (not all events generate email messages).

5.3.2.4. Optional Configuration Tasks

5.3.2.4.1. Confirm Failover and Block Resource Failover Settings

Normally, LifeKeeper will automatically switch operations to a backup node when a node failure or a resource failure occurs. However, depending on the environment, requiring manual confirmation by a system administrator may be desirable, instead of an automatic failover recovery initiated by LifeKeeper. In these cases, the **Confirm Failover** or **Block Resource Failover** settings are available. By using these functions, automatic failover can be blocked and a time to wait for failover can be set when a resource failure or a node failure occurs.

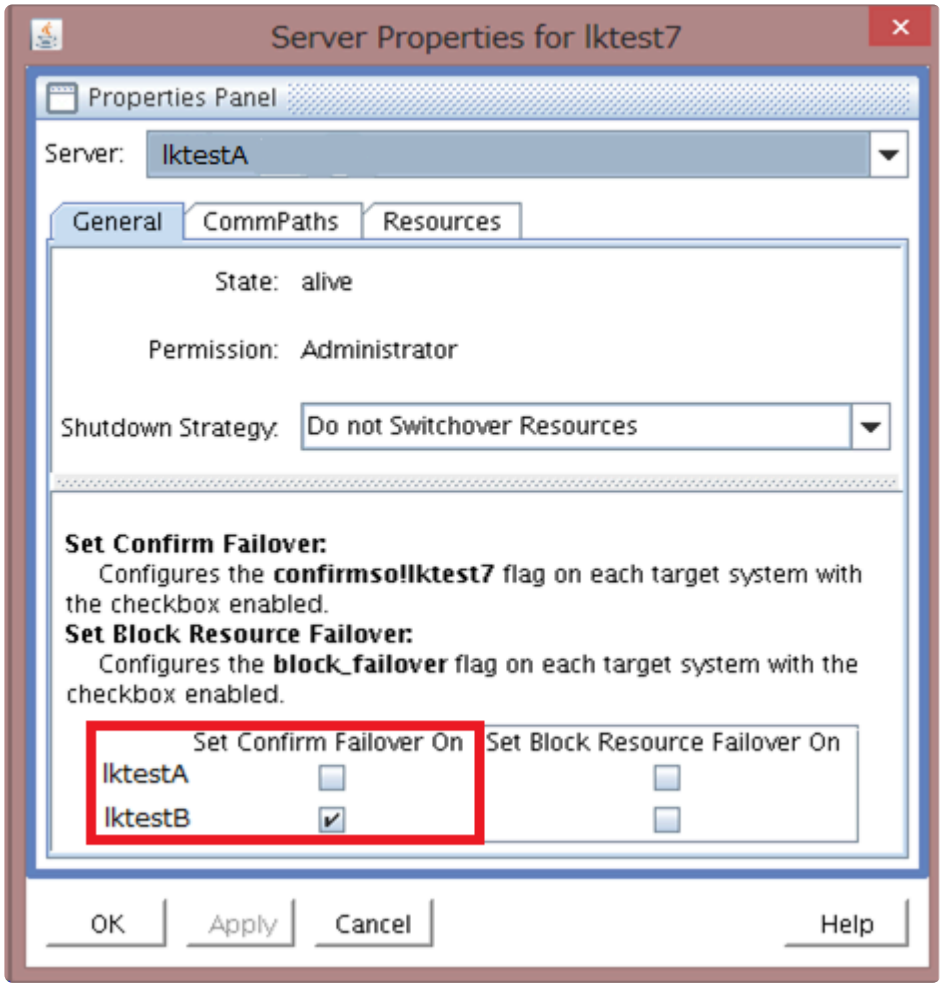
Set **Confirm Failover** or **Block Resource Failover** in your SPS environment after carefully reading the descriptions, examples, and considerations below. These settings are available from the **Server Properties** dialog of the GUI and via the command line of LifeKeeper.

Set Confirm Failover On

When a failover occurs because a node in the LifeKeeper cluster fails (**Note:** a node failure is identified by a failure of all LifeKeeper communication paths to that system), the time to wait before LifeKeeper switches resources to a backup node can be set with the **Confirm Failover** setting (see the discussion on the CONFIRMSOTO variable later in this document). Also, a user can decide whether to automatically switch to the backup node or not after the time to wait expires (see the discussion of the CONFIRMSODEF variable later in this document).

> **Note: Set Confirm Failover On** actions are only available when a node failure occurs. It is not available for resource failures where one or more communications paths are still active.

To enable the **Confirm Failover** setting via the GUI, use the General tab of the **Server Properties** dialog. An example of the General tab for Server Properties is shown below. The part outlined in red on the screen addresses the **Confirm Failover** setting.



Note: This setting is only available for users with Administrator permission for SPS.

In this example, the setting is seen from the host named lktestA. The part outlined in red on the screen is used for this setting **Confirm Failover**. The node names for the HA cluster are displayed vertically. In this example, the standby node for lktestA is lktestB.

The screen shows the configuration status for server lktestA, with the checkbox for lktestB set. In this case, the confirm failover flag is created on lktestB. When a failover from lktestA to lktestB is executed, the confirmation process for executing a failover occurs on lktestB. This process includes checking the default action to take based on the CONFIRMSODEF variable setting (see the discussion later in this document) and how long to wait before taking that action based on the CONFIRMSOTO variable setting.

The **Confirm Failover** flag creation status can be checked via the command line. When the checkbox for lktestB is set on the host named lktestA, the **Confirm Failover** flag is created on lktestB. (**NOTE:** In this example, the flag is not created on lktestA, only on lktestB.) An example of the command line output is below.

[root@lktestB~]# /opt/LifeKeeper/bin/flg_list

confirmso!lktestA

The “confirmso!lktestA” output is the result of the flg_list command, and indicates that the **Confirm Failover** flag is set on node lktestB to confirm lktestA failures.

When failover occurs with the confirmso flag, the following messages are recorded in the LifeKeeper log file.

INFO:lcd.recover:::004113:

chk_man_interv: Flag confirmso!hostname is set, issuing confirmso event and waiting for switchover instruction.

NOTIFY:event.confirmso:::010464:

LifeKeeper: FAILOVER RECOVERY OF MACHINE lktestA requires manual confirmation! Execute ‘/opt/LifeKeeper/bin/lk_confirmso -y -s lktestA ‘ to allow this failover, or execute ‘/opt/LifeKeeper/bin/lk_confirmso -n -s lktestA’ to prevent it. If no instruction is provided, LifeKeeper will timeout in 600 seconds and the failover will be allowed to proceed.

Execute one of the following commands to confirm the failover:

To proceed with the failover:

/opt/LifeKeeper/bin/lk_confirmso -y -s hostname

To block the failover:

/opt/LifeKeeper/bin/lk_confirmso -n -s hostname

The host name that is specified when executing the command is the host name listed in **Confirm Failover** flag which for this example would be lktestA. Execute the command by referring to the example commands provided in the Log output.

In the case where the set time to wait is exceeded, the default failover action is executed (allow failover or block failover). The default failover action is determined by the CONFIRMSODEF variable (see discussion later in this document).

The following message is output to the LifeKeeper log when the timeout expires.

lcdrecover[xxxx]: INFO:lcd.recover:::004408:chk_man_interv: Timed out waiting for instruction, using default CONFIRMSODEF value 0.

The LifeKeeper operation when the time to wait is exceeded is controlled by the setting of the variable “CONFIRMSODEF” which is set in the /etc/default/LifeKeeper file with a value of “1” or “0”. A value of “0” is set by default, and this indicates that the failover will proceed when the time to wait is exceeded. If the value is set to a “1”, the failover is blocked when the time to wait is exceeded.

The time to wait for confirmation of a failover can be changed by adjusting the value of the

CONFIRMSOTO variable in the /etc/default/LifeKeeper file. The value of the variable specifies the number of seconds to wait for a manual confirmation from the user before proceeding or blocking the failover as determined by the value of the “CONFIRMSODEF” variable (see above).

Restarting LifeKeeper or rebooting the OS **is not** required for changes to these variables to take effect. If the value of CONFIRMSOTO is set to 0 seconds, then the operation based on the CONFIRMSODEF setting will occur immediately.

When to Select [Confirm Failover] Setting

This setting is used for Disaster Recovery or WAN configurations in the environment which the communication paths are not redundant.

- Open the **Properties** page from one server and then select the server that you want the **Confirm Failover flag** to be set on.

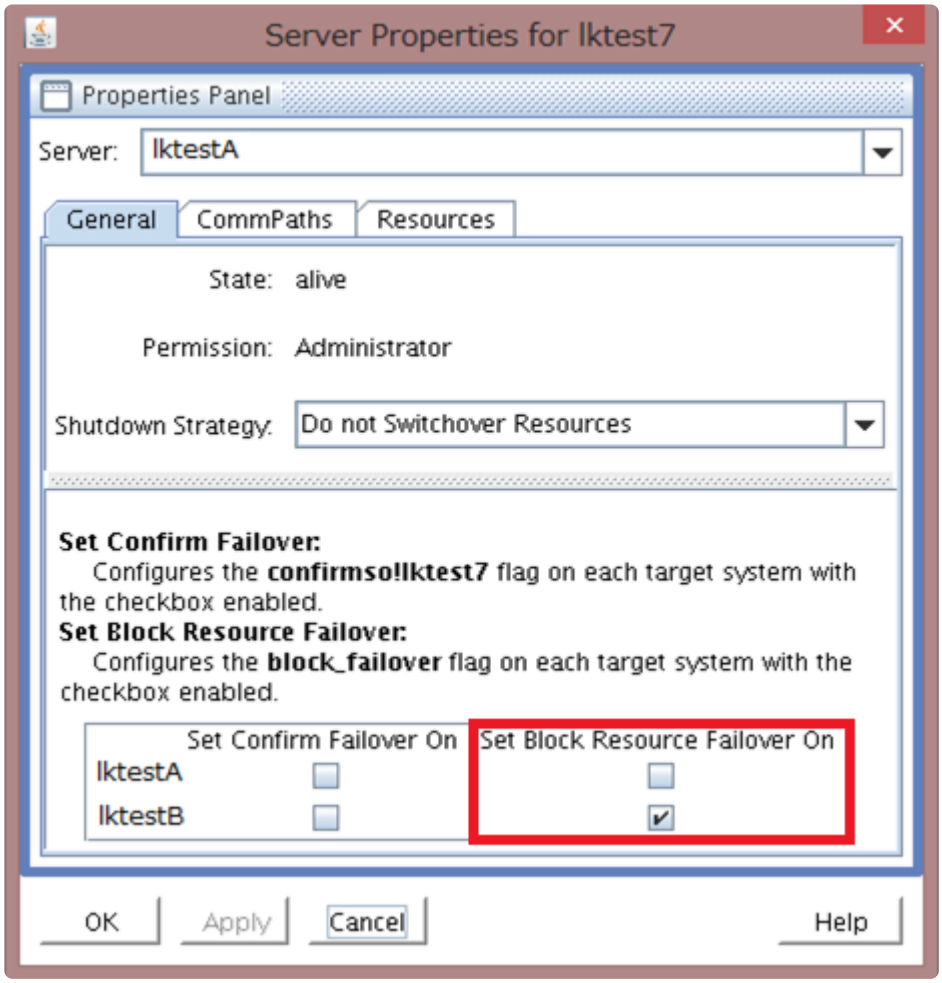
Block Resource Failover On

The **Block Resource Failover On** setting blocks all resource transfers due to a resource failure from the given system.

> **Note:** The **Block Resource Failover On** setting has no effect on the failover processing when a node failure occurs. This setting only blocks a failover attempt when a local resource recovery fails and attempts to transfer the resource to another node in the cluster.

By default, the recovery of resource failures in a local system (local recovery) is performed when a resource failure is detected. When the local recovery has failed or is not enabled, a failover is initiated to the next highest priority standby node defined for the resource. The Block Resource Failover On setting will prevent this failover attempt.

To enable the **Block Resource Failover On** setting by the GUI, use the General tab of the Server Properties. An example of the General tab for Server Properties is below. The part outlined in red on the screen addresses the setting **Block Resource Failover On**.



Note: This setting is only available for users with Administrator permission for SPS.

In this example, the setting is seen from the host named lktestA. The part outlined in red on the screen is used for setting **Block Resource Failover**. The node names for the HA cluster are displayed vertically here. In this example, the standby node for lktestA is lktestB.

In this case, the Block Resource Failover flag is created on lktestB. The “block_failover” flag can be verified on the command line by executing the flg_list command. An example of the output is below.

[root@lktestB~]# /opt/LifeKeeper/bin/flg_list

block_failover

When the block_failover flag is set, the failover to other node (lktestA) is blocked when a resource failure occurs on lkestB.

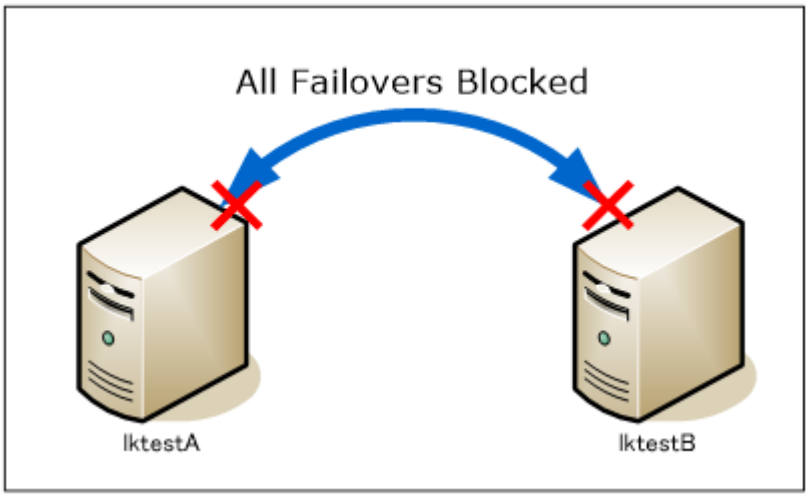
The block_failover flag prevents resource failovers from occurring on the node where the flag is set. The following log message is output to the LifeKeeper log when the failover is blocked by this setting.

ERROR:lcd.recover:::004787:Failover is blocked by current settings. MANUAL INTERVENTION IS REQUIRED

Configuration examples

Some configuration examples are described below.

Block All Automatic Failovers



In this example, the failover is blocked when a node failure or a resource failure is detected on either lktestA or lktestB. Use the Confirm failover and Block Resource failover settings for this. The configuration example is below.

1. Select lktestA and view **Server Properties**.On the General tab, check the **“Set Confirm Failover On”** box for lktestB and the **“Set Block Resource Failover On”** box for both lktestA and lktestB. The setting status in the GUI is below.

   The configuration of Server Properties for lktestA as displayed in the GUI once set.

	Set Confirm Failover On	Set Block Resource Failover On
lktestA	(Not checked)	✔
lktestB	✔	✔

   *When viewing the Server Properties in the GUI the node name can be found near the top of the properties panel display.

2. Select lktestB and view **Server Properties**.

   On General tab, check **“Set Confirm Failover On”** box for lktestA. The **“Set Block Resource Failover On”** property will already be set based on the actions taken in step 1.

   The configuration of Server Properties for lktestB as displayed in the GUI once set.

	Set Confirm Failover On	Set Block Resource Failover On
lktestB	(Not checked)	✔
lktestA	✔	✔

*When viewing the Server Properties in the GUI the node name can be found near the top of the properties panel display.

After completing these steps, confirm that the “confirmso!hostname” and “ block_failover” flags are set for each node using the flg_list command. For the confirmso flag, verify that the host name for which failover confirmation is to be performed is listed as part of the contents of the flag name (to block failover from lktestB on lktestA, lktestA should list lktestB in the contents of the confirmso flag name on lktestA, see the table below).

	Confirm Failover Flag	Block Resource Failover Flag
lktestA	confirmso!lktestB	block_failover
lktestB	confirmso!lktestA	block_failover

3. Set the values for “CONFIRMSOTO” and “CONFIRMSODEF” in /etc/default/LifeKeeper on each node. (Restarting LifeKeeper or rebooting the OS is not required.)

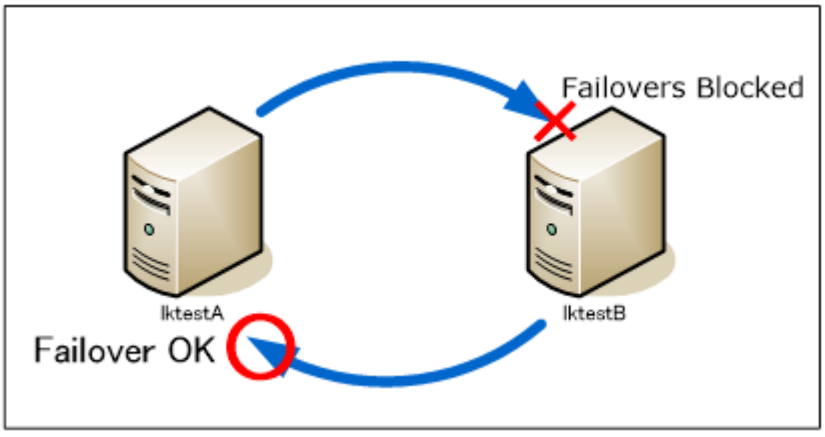
   CONFIRMSODEF=1

   CONFIRMSOTO=0

   When setting the time to wait value, it is specified in seconds via CONFIRMSOTO. For the default action to be taken on failover, the CONFIRMSODEF setting must be either 0 (failover is executed) or 1 (failover is blocked).

With the above settings any node failure will be immediately blocked without any operator intervention.

Block Failovers in One Direction



In this example, the failover to lktestB is blocked when a node failure or a resource failure is detected on

lktestA. On the contrary, the failover to lktestA is allowed when a node failure or a resource failure is detected on lktestB.

1. Select lktestA and view **Server Properties**.

2. On the **General** tab, check **“Set Confirm Failover On”** box for lktestB and the **“Set Block Resource Failover On”** for lktestA.

   The configuration of Server Properties for lktestA as displayed in the GUI once set.

	Set Confirm Failover On	Set Block Resource Failover On
lktestA	(Not checked)	✔
lktestB	✔	(Not checked)

3. Select lktestB and view **Server Properties**.

   In the **General** tab, the **“Set Block Resource Failover On”** box for lktestA should already be set (from the action taken on lktestA).

   The configuration of Server Properties for lktestB as displayed in the GUI once set.

	Set Confirm Failover On	Set Block Resource Failover On
lktestB	(Not checked)	(Not checked)
lktestA	(Not checked)	✔

   *In the GUI, the local host name is listed first.

   For this configuration, verify that the “confirmso!lktestA” flag is set on lktestB (no confirmso flag should be set on lktestA) and the block failover flag is set on lktestA.

	Confirm Failover Flag	Block Resource Failover Flag
lktestA	N/A	block_failover
lktestB	confirmso!lktestA	N/A

4. Set the values for “CONFIRMSODEF” and “CONFIRMSOTO” in /etc/default/LifeKeeper on lktestB.

   CONFIRMSODEF=1

   CONFIRMSOTO=0

For this configuration resource and machine failovers from lktestA to lktestB are blocked. Resource and machine failovers from lktestB to lktestA are allowed.

5.3.2.4.2. Setting Server Shutdown Strategy

The Shutdown Strategy is a LifeKeeper configuration option that governs whether or not resources are switched over to a backup server when a server is shut down. The options are:

Do Not Switch Over Resources (default)	LifeKeeper will not bring resources in service on a backup server during an orderly shutdown.
Switch Over Resources	LifeKeeper will bring resources in service on a backup server during an orderly shutdown.

The Shutdown Strategy is set by default to “Do Not Switch Over Resources.” You should decide which strategy you want to use on each server in the cluster, and if you wish, change the Shutdown Strategy to“Switch Over Resources”.

For each server in the cluster:

1. On the Edit Menu, point to **Server** and then click **Properties**.
2. Select the server to be modified.
3. On the General Tab of the **Server Properties** dialog, select the **Shutdown Strategy**.

Note: The LifeKeeper process must be running during an orderly shutdown for the Shutdown Strategy to have an effect.

Note: Some Amazon EC2 configurations have issues when the Shutdown Strategy is set to “Do not Switchover Resources”. Refer to Known Issues and Restrictions for more information.

5.3.2.4.3. Tuning the LifeKeeper Heartbeat

Overview of the Tunable Heartbeat

The LifeKeeper heartbeat is the signal sent between LifeKeeper servers over the communications path(s) to ensure each server is “alive”. There are two aspects of the heartbeat that determine how quickly LifeKeeper detects a failure:

- **Interval:** the time interval between heartbeats signal sent (unit is second). Failing to receive the LCM signal, which includes heartbeat signal, from another server within the interval time is determined as a missed heartbeat.
- **Number of Heartbeats:** the consecutive number of heartbeats by which the communications path is determined as dead, triggering a failover.

The heartbeat values are specified by two tunables in the LifeKeeper defaults file /etc/default/LifeKeeper. These tunables can be changed if you wish LifeKeeper to detect a server failure sooner than it would using the default values:

- LCMHBEATTIME (interval)
- LCMNUMHBEATS (number of heartbeats)

The following table summarizes the defaults and minimum values for the tunables over both TCP and TTY heartbeats. The interval for a TTY communications path cannot be set below 2 seconds because of the slower nature of the medium.

Tunable	Default Value	Minimum Value
LCMHBEATTIME	5	1 (TCP) 2 (TTY)
LCMNUMHBEATS	3	2 (TCP or TTY)

! The values for both tunables MUST be the SAME on all servers in the cluster.

Example

Consider a LifeKeeper cluster in which both intervals are set to the default values. LifeKeeper sends a heartbeat between servers every 5 seconds. If a communications problem causes the heartbeat to skip two beats, but it resumes on third heartbeat, LifeKeeper takes no action. However, if the communications path remains dead for 3 beats, LifeKeeper will label that communications path as dead, but will initiate a failover only if the redundant communications path is also dead.

Configuring the Heartbeat

You must manually edit file /etc/default/LifeKeeper to add the tunable and its associated value. Normally, the defaults file contains no entry for these tunables; you simply append the following lines with the desired value as follows:

```
LCMHBEATTIME=x

LCMNUMHBEATS=y
```

If you assign the value to a number below the minimum value, LifeKeeper will ignore that value and use the minimum value instead.

Configuration Considerations

- If you wish to set the interval at less than 5 seconds, then you should ensure that the communications path is configured on a private network, since values lower than 5 seconds create a high risk of false failovers due to network interruptions.
- Testing has shown that setting the number of heartbeats to less than 2 creates a high risk of false failovers. This is why the value has been restricted to 2 or higher.
- In order to avoid false failovers, both the interval and heartbeat count values must be the same on all servers in the cluster. For this reason, LifeKeeper must be stopped on both servers before modifying these values. After starting LifeKeeper, you can use the command `/opt/LifeKeeper/bin/lkstop -f` to edit the heartbeat settings while the application is protected. This command stops LifeKeeper but does not stop the protected application.
- LifeKeeper does not impose an upper limit for the LCMHBEATTIME and LCMNUMHBEATS values. But setting these values at a very high number can effectively disable LifeKeeper's ability to detect a failure. For instance, setting both values to 25 would instruct LifeKeeper to wait 625 seconds (over 10 minutes) to detect a server failure, which may be enough time for the server to re-boot and re-join the cluster.

Note: If you are using both TTY and TCP communications paths, the value for each tunable applies to both communications paths. The only exception is if the interval value is below 2, which is the minimum for a TTY communications path.

For example, suppose you specify the lowest values allowed by LifeKeeper in order to detect failure as quickly as possible:

```
LCMHBEATTIME=1

LCMNUMHBEATS=2
```

LifeKeeper will use a 1 second interval for the TCP communications path, and a 2 second interval for TTY. In the case of a server failure, LifeKeeper will detect the TCP failure first because its interval is shorter (2 heartbeats that are 1 second apart), but then will do nothing until it detects the TTY failure, which will be after 2 heartbeats that are 2 seconds apart.

5.3.2.4.4. Using Custom Certificates with the SPS API

Beginning with Release 7.5, the SIOS Protection Suite (SPS) API uses SSL/TLS to communicate between different systems. Currently, this API is only partially used and is reserved for internal use only but may be opened up to customer and third party usage in a future release. By default, the product is installed with default certificates that provide some assurance of identity between nodes. This document explains how to replace these default certificates with certificates created by your own Certificate Authority (CA).

Note: Normal SPS communication does not use these certificates.

How Certificates Are Used

In cases where SSL/TLS is used for communications between SPS servers to protect the data being transferred, a certificate is provided by systems to identify themselves. The systems also use a CA certificate to verify the certificate that is presented to them over the SSL connection.

Three certificates are involved:

- `/opt/LifeKeeper/etc/certs/LK4LinuxValidNode.pem` (server certificate)
- `/opt/LifeKeeper/etc/certs/LK4LinuxValidClient.pem` (client certificate)
- `/opt/LifeKeeper/etc/certs/LKCA.pem` (certificate authority)

The first two certificates must be signed by the CA certificate to satisfy the verification performed by the servers. Note that the common name of the certificates is not verified, only that the certificates are signed by the CA.

Using Your Own Certificates

In some installations, it may be necessary to replace the default certificates with certificates that are created by an organization's internal or commercial CA. If this is necessary, replace the three certificates listed above with new certificates *using the same certificate file names*. These certificates are of the PEM type. The `LK4LinuxValidNode.pem` and `LK4LinuxValidClient.pem` each contain both their respective key and certificate. The `LK4LinuxValidNode.pem` certificate is a *server* type certificate. `LK4LinuxValidClient.pem` is a *client* type certificate.

If the default certificates are replaced, SPS will need to be restarted to reflect the changes. If the certificates are misconfigured, `SIOS-lighttpd` daemon will not start successfully and errors will be received in the LifeKeeper log file. If problems arise, refer to this log file to see the full command that

should be run.

5.3.2.5. Linux Configuration

<table>
<tr><td>Operating System</td><td>The default operating system must be installed to ensure that all required packages are installed. The minimal operating system install does not contain all of the required packages, and therefore, cannot be used with LifeKeeper.</td></tr>
<tr><td>Kernel updates</td><td>
In order to provide the highest level of availability for a LifeKeeper cluster, the kernel version used on a system is very important. The table below lists each supported distribution and version with the kernel that has passed LifeKeeper certification testing.

Note: Beginning with SPS v8.1, when performing a kernel upgrade on Red Hat Enterprise Linux and supported Red Hat Enterprise Linux derivatives (CentOS and OEL), it is no longer a requirement that the setup script (./setup) from the installation image be rerun. Modules should be automatically available to the upgraded kernel without any intervention as long as the kernel was installed from a proper Red Hat package (rpm file).
However, for Red Hat Enterprise Linux 7.3 or later and its compatible distributions, it may be necessary to update the kernel module. If DataKeeper fails to start after upgrading the kernel, check to see if the nbd kernel module is loaded. If it is not loaded run the SPS for Linux setup script to install the appropriate kernel module. There are no SPS kernel module requirements for SUSE Linux Enterprise Server.
<table>
<tr><th>Distribution/Version</th><th>Supported Version</th><th>Supported Kernels</th></tr>
<tr><td>Red Hat Enterprise Linux for AMD64/EM64T

(*6.0 is Not Recommended)</td><td>6.0*
6.1
6.2
6.3
6.4
6.5
6.6
6.7
6.8
6.9
6.10</td><td>2.6.32-71.el6
2.6.32-131.17.1.el6
2.6.32-220.el6
2.6.32-279.el6
2.6.32-358.el6
2.6.32-431.el6
2.6.32-504.el6
2.6.32-573.el6
2.6.32-642.el6
2.6.32-696.el6
2.6.32-754.el6</td></tr>
<tr><td>Red Hat Enterprise Linux for AMD64/EM64T</td><td>7
7.1</td><td>3.10.0-123.el7
3.10.0-229.el7</td></tr>
</table>
</td></tr>
</table>

Distribution/Version	Supported Version	Supported Kernels
(*DataKeeper asynchronous mirrors are not supported on some kernels on RHEL 7.4-7.6. Click here for full details.) (**Note:** If you are using DataKeeper with RHEL 7.8 follow these steps when installing LifeKeeper) (Some kernel versions do not support asynchronous mode. Please see Known Issues and Restrictions for details)	7.2 7.3 7.4* 7.5* 7.6* 7.7 7.8	3.10.0-327.el7 3.10.0-514.el7 3.10.0-693.el7 3.10.0-862.el7 3.10.0-957.el7 3.10.0-1062.el7 3.10.0-1127.el7
Red Hat Enterprise Linux for AMD64/EM64T (Upgrading from RHEL 7 to RHEL 8 is not supported)	8.0 8.1	4.18.0-80.el8.x86_64 4.18.0-147.el8.x86_64
SUSE Linux Enterprise Server 12 for x86_64 (The kernel should be updated to 4.4.82-6.9.1 for SP3.)	12 SP1 12 SP2 12 SP3* 12 SP4 12 SP5 * SLES12.0 is not supported.	3.12.49-11.1 4.4.21-69.1 4.4.82-6.9.1 4.12.14-94.41.1 4.12.14-120.1
SUSE Linux Enterprise Server 15 for x86_64 (*DataKeeper cannot use disks with an odd sector size.) (*Upgrading from version SLES12 to SLES15 is not supported.)	15*/>15 SP1	4.12.14-23.1 4.12.14-195.1

Distribution/Version	Supported Version	Supported Kernels
Oracle Linux	6.3 6.4 6.5 6.6 6.7 6.8 6.9 6.10 UEK R3 UEK R4	2.6.32-279.el6 2.6.32-358.el6 2.6.32-431.el6 2.6.32-504.el6 2.6.32-573.el6 2.6.32-642.el6 2.6.32-696.el6 2.6.32-754.el6 3.8.13-16.2.1.el6uek 4.1.12-37.3.1.el6uek
Oracle Linux (*DataKeeper asynchronous mirrors are not supported on some kernels on OEL 7.4-7.6. Click here for full details.)	7 7.1 7.2 7.3 7.4* 7.5* 7.6* 7.7 7.8 UEK R3 UEK R4 UEK R5	3.10.0-123.el7 3.10.0-229.el7 3.10.0-327.el7 3.10.0-514.el7 3.10.0-693.el7 3.10.0-862.el7 3.10.0-957.el7 3.10.0-1062.el7 3.10.0-1127.el7 3.8.13-16.2.1.el7uek 4.1.12-37.3.1.el7uek 4.14.35-1818.3.3.el7uek
Oracle Linux (Upgrading from OEL 7 to OEL 8 is not supported.)	8.0 8.1	4.18.0-80.el8 4.18.0-147.el8.x86_64
CentOS *(*6.0 – DataKeeper Configuration is Not Supported)*	6.0* 6.1 6.2 6.3 6.4 6.5	2.6.32-71.el6 2.6.32-131.el6 2.6.32-220.el6 2.6.32-279.2.1.el6 2.6.32-358.el6 2.6.32-431.el6

<table>
<tr><td></td><td>
<table>
<tr><th>Distribution/Version</th><th>Supported Version</th><th>Supported Kernels</th></tr>
<tr><td></td><td>6.6
6.7
6.8
6.9
6.10</td><td>2.6.32-504.el6
2.6.32-573.el6
2.6.32-642.el6
2.6.32-696.el6
2.6.32-754.el6</td></tr>
<tr><td>CentOS

(*DataKeeper asynchronous mirrors are not supported on some kernels on CentOS 7.4-7.6. Click here for full details.)</td><td>7
7.1
7.2
7.3
7.4*
7.5*
7.6*
7.7
7.8</td><td>3.10.0-123.el7
3.10.0-229.el7
3.10.0-327.el7
3.10.0-514.el7
3.10.0-693.el7
3.10.0-862.el7
3.10.0-957.el7
3.10.0-1062.el7
3.10.0-1127.el7</td></tr>
<tr><td>CentOS

(Upgrading from CentOS 7 to CentOS 8 is not supported.)</td><td>8.0
8.1</td><td>4.18.0-80.el8
4.18.0-147.el8.x86_64</td></tr>
</table>

Note: This list of supported distributions and kernels is for LifeKeeper only. You should also determine and adhere to the supported distributions and kernels for your server and storage hardware as specified by the manufacturer.
</td></tr>
<tr><td>LUN support</td><td>The Linux SCSI driver has several parameters that control which devices will be probed for Logical Units (LUNs):

• List of devices that do not support LUNs – this list of devices are known to NOT support LUNs, so the SCSI driver will not allow the probing of these devices for LUNs.

• List of devices that do support LUNs – this list of devices is known to support LUNs well, so always probe for LUNs.

• Probe all LUNs on each SCSI device – if a device is not found on either list,</td></tr>
</table>

	whether to probe or not. This parameter is configured by make config in the SCSI module section. While most distributions (including SUSE) have the Probe all LUNs setting enabled by default, Red Hat has the setting disabled by default. External RAID controllers that are typically used in LifeKeeper configurations to protect data are frequently configured with multiple LUNs (Logical Units). To enable LUN support, this field must be selected and the kernel remade. To enable Probe all LUNs without rebuilding the kernel or modules, set the variable max_scsi_luns to 255 (which will cause the scan for up to 255 LUNs). To set the max_scsi_luns on a kernel where the scsi driver is a module (e.g. Red Hat), add the following entry to */etc/modules.conf*, rebuild the initial ramdisk and reboot loading that ramdisk: `options scsi_mod max_scsi_luns=255` To set the max_scsi_luns on a kernel where the scsi driver is compiled into the kernel (e.g. SUSE), add the following entry to */etc/lilo.conf*: `append="max_scsi_luns=255"` **Note**: For some devices, scanning for 255 LUNs can have an adverse effect on boot performance (in particular devices with the BLIST_SPARSELUN defined). The Dell PV650F is an array where this has been experienced. To avoid this performance problem, set the max_scsi_luns to the maximum number of LUNs you have configured on your arrays such as 16 or 32. For example, `append="max_scsi_luns=16"`
Testing environment of channel bonding, network teaming	In LifeKeeper, we performed tests in the environment using the channel bonding or network teaming with the following settings: • Bonding policy in channel bonding + balance-rr + active-backup • Runner in network teaming + round-robin

	+ active-backup

5.3.2.6. Data Replication Configuration

Item	Description
SIOS DataKeeper Feature/ Distribution Matrix	SIOS DataKeeper supports Linux kernel versions 2.6 and higher.
SIOS DataKeeper Documentation	The documentation for SIOS DataKeeper is located within the SIOS Protection Suite Technical Documentation on the SIOS Technology Corp. Website.

5.3.2.7. Network Configuration

Item	Description
IP Recovery Kit impact on routing table	LifeKeeper-protected IP addresses are implemented on Linux as logical interfaces. When a logical interface is configured on Linux, a route to the subnet associated with the logical interface is automatically added to the routing table, even if a route to that subnet already exists (for example, through the physical interface). This additional route to the subnet could possibly result in multiple routing-table entries to the same subnet. If an application is inspecting and attempting to verify the address from which incoming connections are made, the multiple routing-table entries could cause problems for such applications on other systems (non-LifeKeeper installed) to which the LifeKeeper system may be connecting. The multiple routing table entries can make it appear that the connection was made from the IP address associated with the logical interface rather than the physical interface.
IP subnet mask	For IP configurations under LifeKeeper protection, if the LifeKeeper-protected IP address is intended to be on the same subnet as the IP address of the physical interface on which it is aliased, the subnet mask of the two addresses must be the same. Incorrect settings of the subnet mask may result in connection delays and failures between the LifeKeeper GUI client and server.
EEpro100 driver initialization	The Intel e100 driver should be installed to resolve initialization problems with the eepro100 driver on systems with Intel Ethernet Interfaces. With the eepro100 driver, the following errors may occur when the interface is started at boot time and repeat continuously until the interface is shut down. eth0: card reports no Rx buffers eth0: card reports no resources

5.3.2.8. Application Configuration

Item	Description
Database Initialization Files	The initialization files for databases need to be either on a shared device and symbolically linked to specified locations in the local file system or kept on separate systems and manually updated on both systems when changes need to be implemented.
Localized Oracle Mount Points	Localized Oracle environments are different depending on whether you connect as *internal* or as *sysdba*. A database on a localized mount point must be created with "`connect / as sysdba`" if it is to be put under LifeKeeper protection.
Apache Updates	Upgrading an SPS protected Apache application as part of upgrading the Linux operating system requires that the default server instance be disabled on start up. If your configuration file *(httpd.conf)* is in the default directory *(/etc/httpd/conf)*, the Red Hat upgrade will overwrite the config file. Therefore, you should make a copy of the file before upgrading and restore the file after upgrading. Also, see the Specific Configuration Considerations for Apache Web Server section in the *Apache Web Server Recovery Kit Administration Guide*.

5.3.2.9. Storage and Adapter Configuration

Item	Description
Multipath I/O and Redundant Controllers	There are several multipath I/O solutions either already available or currently being developed for the Linux environment. SIOS Technology Corp. is actively working with a number of server vendors, storage vendors, adapter vendors and driver maintainers to enable LifeKeeper to work with their multipath I/O solutions. LifeKeeper's use of SCSI reservations to protect data integrity presents some special requirements that frequently are not met by the initial implementation of these solutions. Refer to the technical notes below for supported disk arrays to determine if a given array is supported with multiple paths and with a particular multipath solution. Unless an array is specifically listed as being supported by LifeKeeper with multiple paths and with a particular multipath solution, it must be assumed that it is not.
Heavy I/O in Multipath Configurations	In multipath configurations, performing heavy I/O while paths are being manipulated can cause a system to temporarily appear to be unresponsive. When the multipath software moves the access of a LUN from one path to another, it must also move any outstanding I/Os to the new path. The rerouting of the I/Os can cause a delay in the response times for these I/Os. If additional I/Os continue to be issued during this time, they will be queued in the system and can cause a system to run out of memory available to any process. Under very heavy I/O loads, these delays and low memory conditions can cause the system to be unresponsive such that LifeKeeper may detect a server as down and initiate a failover. There are many factors that will affect the frequency at which this issue may be seen. • The speed of the processor will affect how fast I/Os can be queued. A faster processor may cause the failure to be seen more frequently. • The amount of system memory will affect how many I/Os can be queued before the system becomes unresponsive. A system with more memory may cause the failure to be seen less frequently. • The number of LUNs in use will affect the amount of I/O that can be queued. • Characteristics of the I/O activity will affect the volume of I/O queued. In test cases where the problem has been seen, the test was writing an unlimited amount of data to the disk. Most applications will both read and write data. As the reads are blocked waiting on the failover, writes will also

	be throttled, decreasing the I/O rate such that the failure may be seen less frequently. For example, during testing of the IBM DS4000 multipath configuration with RDAC, when the I/O throughput to the DS4000 was greater than 190 MB per second and path failures were simulated, LifeKeeper would (falsely) detect a failed server approximately one time out of twelve. The servers used in this test were IBM x345 servers with dual Xeon 2.8GHz processors and 2 GB of memory connected to a DS4400 with 8 volumes (LUNs) per server in use. To avoid the failovers, the LifeKeeper parameter LCMNUMHBEATS (in `/etc/default/LifeKeeper`) was increased to 16. The change to this parameter results in LifeKeeper waiting approximately 80 seconds before determining that an unresponsive system is dead, rather than the default wait time of approximately 15 seconds.
Special Considerations for Switchovers with Large Storage Configurations	With some large storage configurations (for example, multiple logical volume groups with 10 or more LUNs in each volume group), LifeKeeper may not be able to complete a sendevent within the default timeout of 300 seconds when a failure is detected. This results in the switchover to the backup system failing. All resources are not brought in-service and an error message is logged in the LifeKeeper log. The recommendation with large storage configurations is to change SCSIERROR from “event” to “halt” in the ***`/etc/default/LifeKeeper`*** file. This will cause LifeKeeper to perform a “halt” on a SCSI error. LifeKeeper will then perform a successful failover to the backup system.
HP 3PAR StoreServ 7200 FC	The HP 3PAR StoreServ 7200 was tested by a SIOS Technology Corp. partner with the following configurations: HP 3PAR StoreServ 7200 (Firmware (HP 3PAR OS) version 3.1.2) using QLogic QMH2572 8Gb FC HBA for HP BladeSystem c-Class (Firmware version 5.06.02 (90d5)), driver version 8.03.07.05.06.2-k (RHEL bundled) with DMMP (device-mapper-1.02.66-6, device-mapper-multipath-0.4.9-46.el6). The test was performed with SPS for Linux v8.1.1 using RHEL 6.2 (x86_64). **Note:** 3PAR StoreServ 7200 returns a reservation conflict with the default path checker. To avoid this conflict, set the following parameter in “*/etc/default/LifeKeeper*”: **DMMP_REGISTRATION_TYPE=hba**

HP 3PAR StoreServ 7400 FC	The HP 3PAR StoreServ 7400 was tested by a SIOS Technology Corp. partner with the following configurations: HP 3PAR StoreServ 7400 (Firmware (HP 3PAR OS) version 3.1.2) with HP DL380p Gen8 with Emulex LightPulse Fibre Channel SCSI HBA (driver version 8.3.5.45.4p) with DMMP (device-mapper-1.02.66-6, device-mapper-multipath-0.4.9-46.el6). The test was performed with LifeKeeper for Linux v8.1.1 using RHEL 6.2 (x86_64). **Note:** 3PAR StoreServ 7400 returns a reservation conflict with the default path checker. To avoid this conflict, set the following parameter in "**/etc/default/LifeKeeper**": **DMMP_REGISTRATION_TYPE=hba** And user friendly device mapping are not supported. Set the following parameter in "multipath.conf" **"user_friendly_names no"**
HP 3PAR StoreServ 7400 iSCSI (multipath configuration using the DMMP Recovery Kit)	The HP 3PAR StoreServ 7400 was tested by a SIOS Technology Corp. partner with the following configurations: HP 3PAR StoreServ 7400 (Firmware (HP 3PAR OS) version 3.1.3) using HP Ethernet 10Gb 2-port 560SFP+ Adapter (Networkdriver ixgbe-3.22.0.2) with iSCSI (iscsi-initiator-utils-6.2.0.873-10.el6.x86_64), DMMP (device-mapper-1.02.79-8.el6,device-mapper-multipath-0.4.9-72.el6). **Note:** 3PAR StoreServ 7400 iSCSI returns a reservation conflict. To avoid this conflict, set the following parameter in "**/etc/default/LifeKeeper**": **DMMP_REGISTER_IGNORE=TRUE**
HP 3PAR StoreServ 7400 iSCSI(using Quorum/Witness Kit)	The HP 3PAR StoreServ 7400 was tested by a SIOS Technology Corp. partner with the following configurations: iSCSI (iscsi-initiator-utils-6.2.0.872-21.el6.x86_64), DMMP (device-mapper-multipath-0.4.9-41, device-mapper-1.02.62-3), nx_nic v4.0.588. DMMP with the DMMP Recovery Kit on RHEL 6.1 — must be used with the

	combination of Quorum/Witness Server Kit and STONITH. To disable SCSI reservation, set RESERVATIONS=none in"**/etc/default/LifeKeeper**". Server must have interface based on IPMI 2.0.
HP 3PAR StoreServ 10800 FC	The HP 3PAR StoreServ 10800 FC was tested by a SIOS Technology Corp. partner with the following configurations: Firmware (HP 3PAR OS) version 3.1.2 with HP DL380p Gen8 with Emulex LightPulse Fibre Channel HBA (driver version 8.3.5.45.4p) with DMMP (device-mapper-1.02.66-6, device-mapper-multipath-0.4.9-46 el6). The test was performed with SPS for Linux v8.1.2 using RHEL 6.2 (x86_64). **Note:** 3PAR StoreServ 10800 FC returns a reservation conflict with the default path checker. To avoid this conflict, set the following parameter in "**/etc/default/LifeKeeper**": **DMMP_REGISTRATION_TYPE=hba** And user friendly device mapping are not supported. Set the following parameter in "multipath.conf" **"user_friendly_names no"**
HP MSA1040/2040fc	The HP MSA 2040 Storage FC was tested by a SIOS Technology Corp. partner with the following configurations: HP MSA 2040 Storage FC (Firmware GL101R002) using HP SN1000Q 16Gb 2P FC HBA QW972A (Firmware version 6.07.02, driver version 8.04.00.12.06.0-k2 (RHEL bundled)) with DMMP (device-mapper-1.02.74-10, device-mapper-multipath-0.4.9-56). The test was performed with LifeKeeper for Linux v8.1.2 using RHEL 6.3 (X86_64).
HP P9500/XP	Certified by Hewlett-Packard Company using SIOS LifeKeeper for Linux v7.2 or later. Model tested was the HP P9500/XP and has been qualified to work with LifeKeeper on the following: • Red Hat Enterprise for 32-bit, x64 (64-bit; Opteron and Intel EMT64) RHEL 5.3, RHEL 5.4, RHEL 5.5 • SuSE Enterprise Server for 32-bit, x64 (64-bit; Opteron and Intel EMT64)

	SLES 10 SP3, SLES 11, SLES 11 SP1 • Native or Inbox Clustering Solutions RHCS and SLE HA
HP StoreVirtual 4330 iSCSI (multipath configuration using the DMMP Recovery Kit)	The HP StoreVirtual 4330 was tested by a SIOS Technology Corp. partner with the following configurations: HP StoreVirtual 4330 (Firmware HP LeftHand OS 10.5) using HP Ethernet 1Gb 4-port 331FLR (Networkdriver tg3-3.125g) with iSCSI (iscsi-initiator-utils-6.2.0.872-41.el6.x86_64), DMMP (device-mapper-1.02.74-10.el6,device-mapper-multipath-0.4.9-56.el6).
StoreVirtual (LeftHand) series OS (SAN/iQ) version 11.00 iSCSI (multipath configuration using the DMMP Recovery Kit)	OS (SAN/iQ) version 11.00 is supported in HP StoreVirtual (LeftHand) storage. All StoreVirtual series are supported, including StoreVirtual VSA as the virtual storage appliance. This storage was tested with the following configurations: StoreVirtual VSA + RHEL 6.4(x86_64) ＋ DMMP
HP StoreVirtual 4730 iSCSI (multipath configuration using the DMMP Recovery Kit)	The HP StoreVirtual 4730 was tested by a SIOS Technology Corp. partner with the following configurations: HP StoreVirtual 4730 (Firmware HP LeftHand OS 11.5) using HP FlexFabric 10Gb 2-port 536FLB Adapter (Networkdriver bnx2×-1.710.40) with iSCSI (iscsi-initiator-utils-6.2.0.873-10.el6.x86_64), DMMP (device-mapper-1.02.79-8.el6,device-mapper-multipath-0.4.9-72.el6).
HP StoreVirtual LeftHand OS version 11.5 iSCSI (multipath configuration using the DMMP Recovery Kit)	LeftHand OS version 11.5 is supported in HP StoreVirtual (LeftHand) storage. All StoreVirtual series are supported, including StoreVirtual VSA as the virtual storage appliance. This storage was tested with the following configurations: StoreVirtual 4730(11.5.00.0673.0) + RHEL 6.5(x86_64) ＋ DMMP (device-mapper-1.02.79-8.el6.x86_64, device-mapper-multipath-0.4.9-72.el6.x86_64)
HP StoreVirtual 4330 iSCSI (using Quorum/Witness Kit)	The HP StoreVirtual 4330 was tested by a SIOS Technology Corp. partner with the following configurations: HP StoreVirtual 4330 (Firmware HP LeftHand OS 10.5) using iSCSI (iscsi-initiator-utils-6.2.0.872-41.el6.x86_64), bonding(version: 3.6.0),tg3(version: 3.125g) To disable SCSI reservation, set RESERVATIONS=none in “*/etc/default/*

	LifeKeeper".
IBM San Volume Controller (SVC)	Certified by partner testing in a single path configuration. Certified by SIOS Technology Corp. in multipath configurations using the Device Mapper Multipath Recovery Kit.
IBM Storwize V7000 iSCSI	The IBM Storwize V7000 (Firmware Version 6.3.0.1) has been certified by partner testing using iSCSI (iscsi-initiator-utils-6.2.0.872-34.el6.x86_64) with DMMP (device-mapper-1.02.66-6.el6, device-mapper-multipath-0.4.9-46.el6). The test was performed with LifeKeeper for Linux v7.5 using RHEL 6.2. **Restriction:** IBM Storwize V7000 must be used in combination with the Quorum/ Witness Server Kit and STONITH. Disable SCSI reservation by setting the following in */etc/default/LifeKeeper*: • RESERVATIONS=none
IBM Storwize V7000 FC	The IBM Storwize V7000 FC has been certified by partner testing in multipath configurations on Red Hat Enterprise Linux Server Release 6.2 (Tikanga), HBA: QLE2562 DMMP: 0.4.9-46.
IBM Storwize V3700 FC	The IBM Storwize V3700 FC has been certified by partner testing in multipath configurations on Red Hat Enterprise Linux Server Release 6.5 (Santiago), HBA: QLE2560 DMMP: 0.4.9-72.
IBM XIV Storage System	Certified by partner testing in only multipath configuration on Red Hat Enterprise Linux Release 5.6, HBA: NEC N8190-127 Single CH 4Gbps (Emulex LPe1150 equivalent), XIV Host Attachement Kit: Version 1.7.0. **Note:** If you have to create over 32 LUNs on IBM XIV Storage System with LifeKeeper, please contact your IBM sales representative for details.
Dell EqualLogic PS4000/4100/4110/ 6000/6010/6100/ 6110/6500/6510	The Dell EqualLogic was tested by a SIOS Technology Corp. partner with the following configurations: Dell EqualLogic PS4000/4100/4110/6000/6010/6100/6110/ 6500/6510 using DMMP with the DMMP Recovery Kit with RHEL 5.3 with iscsi-initiator-utils-6.2.0.868-0.18.el5. With a large number of luns (over 20), change the REMOTETIMEOUT setting in */etc/default/LifeKeeper* to REMOTETIMEOUT=600.
Fujitsu ETERNUS DX60 S2 / DX80 S2 / DX90 S2 iSCSI	When using LifeKeeper DMMP ARK for multipath configuration it is necessary to set the following parameters to /etc/multipath.conf. `prio alua`

ETERNUS DX410 S2 / DX440 S2 iSCSI ETERNUS DX8100 S2/DX8700 S2 iSCSI ETERNUS DX100 S3/DX200 S3 iSCSI ETERNUS DX500 S3/DX600 S3 iSCSI ETERNUS DX200F iSCSI ETERNUS DX60 S3 iSCSI ETERNUS AF250 / AF650 iSCSI ETERNUS DX60 S4 / DX100 S4 / DX200 S4 iSCSI ETERNUS DX500 S4 / DX600 S4 / DX8900S4 iSCSI ETERNUS AF250 S2 / AF650 S2 iSCSI ETERNUS DX60 S5 / DX100 S5 / DX200 S5 iSCSI ETERNUS DX500 S5 / DX600 S5 / DX900S5 iSCSI ETERNUS AF150	`path_grouping_policy group_by_prio` `failback immediate` `no_path_retry 10` `Path_checker tur`

S3 / AF250 S3 / AF650 S3 iSCSI	
Fujitsu ETERNUS DX60 S2 / DX80 S2 / DX90 S2 • FC, single path and multipath configurations ETERNUS DX410 S2 / DX440 S2 • FC, single path and multipath configurations ETERNUS DX8100 S2/DX8700 S2 • FC, single path and multipath configurations ETERNUS DX100 S3/DX200 S3/ DX500 S3/ DX600S3 • FC, single path and multipath configurations ETERNUS DX200F • FC, single path and multipath	When using LifeKeeper DMMP ARK for multipath configuration it is necessary to set the following parameters to /etc/multipath.conf. `prio alua` `path_grouping_policy group_by_prio` `failback immediate` `no_path_retry 10` `Path_checker tur` When using ETERNUS Multipath Driver for multipath configuration, it is no need to set parameters to any configure file.

configurations ETERNUS DX60 S3 • FC, single path and multipath configurations ETERNUS DX8700 S3 / DX8900 S3 • FC, single path and multipath configurations ETERNUS AF250 / AF650 ETERNUS DX60 S4 / DX100 S4 / DX200 S4 ETERNUS DX500 S4 / DX600 S4 / DX8900S4 ETERNUS AF250 S2 / AF650 S2 ETERNUS DX60 S5 / DX100 S5 / DX200 S5 ETERNUS DX500 S5 / DX600 S5 / DX900S5 ETERNUS AF150 S3 / AF250 S3 / AF650 S3 • iSCSI, single path and multipath configurations	

NEC iStorage M10e iSCSI (Multipath configuration using the SPS Recovery Kit)	The NEC iStorage M10e iSCSI was tested by a SIOS Technology Corp. partner with the following configurations: NEC iStorage M10e iSCSI + 1GbE NIC + iSCSI (iscsi-initiator-utils-6.2.0.873-10.el6.x86_64),SPS (sps-utils-5.3.0-0.el6,sps-driver-E-5.3.0-2.6.32.431.el6)
NEC iStorage Storage Path Savior Multipath I/O	**Protecting Applications and File Systems That Use Multipath Devices:** In order for SPS to configure and protect applications or file systems that use SPS devices, the SPS recovery kit must be installed. Once the SPS kit is installed, simply creating an application hierarchy that uses one or more of the multipath device nodes will automatically incorporate the new resource types provided by the SPS kit. **Multipath Device Nodes:** To use the SPS kit, any file systems and raw devices must be mounted or configured on the multipath device nodes (/dev/dd*) rather than on the native /dev/sd* device nodes. **Use of SCSI-3 Persistent Reservations:** The SPS kit uses SCSI-3 persistent reservations with a “Write Exclusive” reservation type. This means that devices reserved by one node in the cluster will remain read-accessible to other nodes in the cluster, but those other nodes will be unable to write to the device. Note that this does not mean that you can expect to be able to mount file systems on those other nodes for ongoing read-only access. LifeKeeper uses the sg_persist utility to issue and monitor persistent reservations. If necessary, LifeKeeper will install the sg_persist(8) utility. **Tested Environment:** The SPS kit has been tested and certified with the NEC iStorage disk array using Emulex HBAs and Emulex lpfc driver. This kit is expected to work equally well with other NEC iStorage D, S and M supported by SPS. **[Tested Emulex HBA]** iStorage D-10 ========== LP952 LP9802 LP1050 LP1150 ==========

	iStorage M100 ========== LPe1150 LPe11002 LPe1250 LPe12002 LPe1105 LPe1205 ========== **Multipath Software Requirements:** The SPS kit has been tested with SPS for Linux 3.3.001. There are no known dependencies on the version of the SPS package installed. **Installation Requirements:** SPS software must be installed prior to installing the SPS recovery kit. **Adding or Repairing SPS Paths:** When LifeKeeper brings an SPS resource into service, it establishes a persistent reservation registered to each path that was active at that time. If new paths are added after the initial reservation, or if failed paths are repaired and SPS automatically reactivates them, those paths will not be registered as a part of the reservation until the next LifeKeeper quickCheck execution for the SPS resource. If SPS allows any writes to that path prior to that point in time, reservation conflicts that occur will be logged to the system message file. The SPS driver will retry these IOs on the registered path resulting in no observable failures to the application. Once quickCheck registers the path, subsequent writes will be successful.
Pure Storage FA-400 Series FC (Multipath configuration using the DMMP Recovery Kit)	By partner testing in multipath configuration of FC connection using the DMMP Recovery Kit.
QLogic Drivers	For other supported fibre channel arrays with QLogic adapters, use the qla2200 or qla2300 driver, version 6.03.00 or later.
Emulex Drivers	For the supported Emulex fibre channel HBAs, you must use the lpfc driver v8.0.16 or later.
Adaptec 29xx Drivers	For supported SCSI arrays with Adaptec 29xx adapters, use the aic7xxx driver, version 6.2.0 or later, provided with the OS distribution.

HP Multipath I/O Configurations

Item	Description

Multipath Cluster Installation Using Secure Path	For a fresh installation of a multiple path cluster that uses Secure Path, perform these steps: 1. Install the OS of choice on each server. 2. Install the clustering hardware: FCA2214 adapters, storage, switches and cables. 3. Install the HP Platform Kit. 4. Install the HP Secure Path software. This will require a reboot of the system. Verify that Secure Path has properly configured both paths to the storage. See Secure Path documentation for further details. 5. Install LifeKeeper.
Secure Path Persistent Device Nodes	Secure Path supports “persistent” device nodes that are in the form of /dev/spdev/spXX where XX is the device name. These nodes are symbolic links to the specific SCSI device nodes /dev/sdXX. LifeKeeper v4.3.0 or later will recognize these devices as if they were the “normal” SCSI device nodes /dev/sdXX. LifeKeeper maintains its own device name persistence, both across reboots and across cluster nodes, by directly detecting if a device is /dev/sda1 or /dev/sdq1, and then directly using the correct device node. **Note:** Support for symbolic links to SCSI device nodes was added in LifeKeeper v4.3.0.
Active/ Passive Controllers and Controller Switchovers	The MSA1000 implements multipathing by having one controller active with the other controller in standby mode. When there is a problem with either the active controller or the path to the active controller, the standby controller is activated to take over operations. When a controller is activated, it takes some time for the controller to become ready. Depending on the number of LUNs configured on the array, this can take 30 to 90 seconds. During this time, IOs to the storage will be blocked until they can be rerouted to the newly activated controller.
Single Path on Boot Up Does Not Cause Notification	If a server can access only a single path to the storage when the system is loaded, there will be no notification of this problem. This can happen if a system is rebooted where there is a physical path failure as noted above, but transient path failures have also been observed. It is advised that any time a system is loaded, the administrator should check that all paths to the storage are properly configured, and if not, take actions to either repair any hardware problems or reload the system to resolve a transient problem.

Hitachi Multipath I/O Configurations

Item	Description
Protecting Applications and File Systems That Use Multipath Devices	In order for LifeKeeper to configure and protect applications or file systems that use HDLM device must be installed. Once the HDLM Kit is installed, simply creating an application hierarchy that uses one or more of automatically incorporate the new resource types provided by the HDLM Kit.
Multipath Device Nodes	To use the HDLM Kit, any file systems and raw devices must be mounted or configured on the multipa rather than on the native /dev/sd* device nodes.

Use of SCSI-3 Persistent Reservations	The HDLM Kit uses SCSI-3 persistent reservations with a “Write Exclusive” reservation type. This by one node in the cluster will remain read-accessible to other nodes in the cluster, but those othe to the device. Note that this does not mean that you can expect to be able to mount file systems o ongoing read-only access. LifeKeeper uses the sg_persist utility to issue and monitor persistent reservations. If necessary, Li sg_persist(8) utility.
Hardware Requirements	The HDLM Kit has been tested and certified with the Hitachi SANRISE AMS1000 disk array using 8.02.00-k5-rhel5.2-04 driver and Silkworm3800 FC switch. This kit is expected to work equally wel arrays. The HDLM Kit has also been certified with the SANRISE AMS series, SANRISE USP and HBA driver must be supported by HDLM. BR1200 is certified by Hitachi Data Systems. Both single path and multipath configuration require BR1200 configuration using the RDAC driver is supported, and the BR1200 configuration using HD supported.
Multipath Software Requirements	The HDLM kit has been tested with HDLM for Linux as follows: 05-80, 05-81, 05-90, 05-91, 05-92, 05-93, 05-94, 6.0.0, 6.0.1, 6.1.0, 6.1.1, 6.1.2, 6.2.0, 6.2.1, 6.3.0 6.5.2, 6.6.0, 6.6.2, 7.2.0, 7.2.1, 7.3.0, 7.3.1, 7.4.0, 7.4.1, 7.5.0, 7.6.0, 7.6.1, 8.0.0, 8.0.1, 8.1.0, 8.1 8.2.1, 8.4.0, 8.5.0, 8.5.1, 8.5.2, 8.5.3, 8.6.0, 8.6.1, 8.6.2, 8.6.4, 8.6.5, 8.7.0, 8.7.1, 8.7.2, 8.7.3 There are no known dependencies on the version of the HDLM package installed. **Note:** The product name changed to “Hitachi Dynamic Link Manager Software (HDLM)” for HDLM than 6.0.0 (05-9x) are named “Hitachi HiCommand Dynamic Link Manager (HDLM)”. **Note:** HDLM version 6.2.1 or later is not supported by HDLM Recovery Kit v6.4.0-2. If you need to you can use HDLM Recovery Kit v7.2.0-1 or later with LK Core v7.3 or later. **Note:** If using LVM with HDLM, the version supported by HDLM is necessary. Also, a filter setting lvm.conf to ensure that the system does not detect the /dev/sd* corresponding to the /dev/sddlm*. please see “LVM Configuration” in the HDLM manual.
Linux Distribution Requirements	Linux Distribution Requirements HDLM is supported in the following distributions: RHEL 4 (AS/ES) (x86 or x86_64) Update 1, 2, 3, 4, Update 4 Security Fix (*2), 4.5,4.5 Security Fix Fix(*8),4.7,4.7 Security Fix(*9), 4.8,4.8 Security Fix(*12) (x86/x86_64)(*1)

	RHEL 5, 5.1, 5.1 Security Fix(*5), 5.2, 5.2 Security Fix(*6), 5.3, 5.3 Security Fix(*10),5.4 , 5.4 Sec Fix(*13), 5.6, 5.6 Security Fix(*14), 5.7 (x86/x86_64)(*1), 5.8 (x86/x86_64)(*1) RHEL 6, 6.1, 6.2 , 6.3, 6.4, 6.5, 6.6, 6.7, 6.8, 6.9, 6.10 (x86/x86_64)(*1)(*15) RHEL 7, 7.1, 7.2, 7.3, 7.3, 7.4, 7.5, 7.6, 7.7 (x86_64)(*1) (*1) AMD Opteron(Single Core, Dual Core) or Intel EM64T architecture CPU with x86_64 kernel. (*2) The following kernels are supported x86:2.6.9-42.0.3.EL, 2.6.9-42.0.3.ELsmp, 2.6.9-42.0.3.ELhugemem x86_64:2.6.9-42.0.3.EL, 2.6.9-42.0.3.ELsmp, 2.6.9-42.0.3.Ellargesmp (*3) Hitachi does not support RHEL4 U2 environment (*4) The following kernels are supported x86:2.6.9-55.0.12.EL, 2.6.9-55.0.12.ELsmp, 2.6.9-55.0.12.Elhugememx 86_64:2.6.9-55.0.12.EL, 2.6.9-55.0.12.ELsmp, 2.6.9-55.0.12.Ellargesmp (*5) The following kernels are supported x86:2.6.18-53.1.13.el5, 2.6.18-53.1.13.el5PAE, 2.6.18-53.1.21.el5, 2.6.18-53.1.21.el5PAE x86_64:2.6.18-53.1.13.el5, 2.6.18-53.1.21.el5 (*6) The following kernels are supported x86:2.6.18-92.1.6.el5, 2.6.18-92.1.6.el5PAE, 2.6.18-92.1.13.el5, 2.6.18- 92.1.13.el5PAE, 2.6.18-9 2.6.18-92.1.22.el5PAE x86_64:2.6.18-92.1.6.el5, 2.6.18-92.1.13.el5, 2.6.18-92.1.22.el5 (*7) The following kernels are supported x86:2.6.9-34.0.2.EL, 2.6.9-34.0.2.ELsmp, 2.6.9-34.0.2.ELhugemem x86_64:2.6.9-34.0.2.EL, 2.6.9-34.0.2.ELsmp, 2.6.9-34.0.2.Ellargesmp (*8) The following kernels are supported x86:2.6.9-67.0.7.EL, 2.6.9-67.0.7.ELsmp, 2.6.9-67.0.7.ELhugemem, 2.6.9- 67.0.22.EL, 2.6.9-67.0. 2.6.9-67.0.22.ELhugemem x86_64:2.6.9-67.0.7.EL, 2.6.9-67.0.7.ELsmp, 2.6.9-67.0.7.ELlargesmp 2.6.9-67.0.22.EL, 2.6.9-67.0.22.ELsmp, 2.6.9-67.0.22.Ellargesmp (*9) The following kernels are supported x86:2.6.9-78.0.1.EL, 2.6.9-78.0.1.ELsmp, 2.6.9-78.0.1.ELhugemem, 2.6.9-78.0.5.EL, 2.6.9-78.0.5. 2.6.9-78.0.5.ELhugemem, 2.6.9-78.0.8.EL, 2.6.9-78.0.8.ELsmp, 2.6.9-78.0.8.ELhugemem, 2.6.9-7 2.6.9-78.0.17.ELsmp, 2.6.9-78.0.17.ELhugemem, 2.6.9-78.0.22.EL, 2.6.9-78.0.22.ELsmp, 2.6.9-78 x86_64:2.6.9-78.0.1.EL, 2.6.9-78.0.1.ELsmp, 2.6.9-78.0.1.ELlargesmp, 2.6.9-78.0.5.EL, 2.6.9-78. 2.6.9-78.0.5.ELlargesmp, 2.6.9-78.0.8.EL, 2.6.9-78.0.8.ELsmp, 2.6.9-78.0.8.ELlargesmp, 2.6.9-78.

	2.6.9-78.0.17.ELsmp, 2.6.9-78.0.17.ELlargesmp, 2.6.9-78.0.22.EL, 2.6.9-78.0.22.ELsmp, 2.6.9-78. (*10) The following kernels are supported x86:2.6.18-128.1.10.el5, 2.6.18-128.1.10.el5PAE, 2.6.18-128.1.14.el5, 2.6.18-128.1.14.el5PAE, 2 2.6.18-128.7.1.el5PAE x86_64:2.6.18-128.1.10.el5, 2.6.18-128.1.14.el5 (*11) The following kernels are supported x86:2.6.18-164.9.1.el5, 2.6.18-164.9.1.el5PAE, 2.6.18-164.11.1.el5, 2.6.18-164.11.1.el5PAE x86_64:2.6.18-164.9.1.el5, 2.6.18-164.11.1.el5 (*12) The following kernels are supported x86:2.6.9-89.0.20.EL, 2.6.9-89.0.20.ELsmp, 2.6.9-89.0.20.Elhugemem x86_64:2.6.9-89.0.20.EL, 2.6.9-89.0.20.ELsmp, 2.6.9-89.0.20.Ellargesmp (*13) The following kernels are supported x86:2.6.18-194.11.1.el5, 2.6.18-194.11.1.el5PAE, 2.6.18-194.11.3.el5, 2.6.18-194.11.3.el5PAE, 2 2.6.18-194.17.1.el5PAE, 2.6.18-194.32.1.el5, 2.6.18-194.32.1.el5PAE x86_64:2.6.18-194.11.1.el5, 2.6.18-194.11.3.el5, 2.6.18-194.17.1.el5, 2.6.18-194.32.1.el5 (*14) The following kernels are supported x86: 2.6.18-238.1.1.el5,2.6.18-238.1.1.el5PAE,2.6.18-238.9.1.el5,2.6.18-238.9.1.el5PAE,2.6.18-238.19 x86_64:2.6.18-238.1.1.el5,2.6.18-238.9.1.el5,2.6.18-238.19.1.el5 (*15) The following kernels are supported x86:2.6.32-71.el6.i686, 2.6.32-131.0.15.el6.i686, 2.6.32-220.el6.i686, 2.6.32-279.el6.i686 x86_64:2.6.32-71.el6.x86_64, 2.6.32-131.0.15.el6.x86_64, 2.6.32-220.el6.x86_64, 2.6.32-279.el6 (*16) The following kernels are supported x86:2.6.18-274.12.1.el5, 2.6.18-274.12.1.el5PAE, 2.6.18-274.18.1.el5, 2.6.18-274.18.1.el5PAE x86_64:2.6.18-274.12.1.el5, 2.6.18-274.18.1.el5 (*17) The following kernels are supported x86:2.6.18-308.8.2.el5, 2.6.18-308.8.2.el5PAE x86_64:2.6.18-308.8.2.el5 (*18) The following kernels are supported x86:2.6.32-220.4.2.el6.i686, 2.6.32-220.17.1.el6.i686, 2.6.32-220.23.1.el6.i686, 2.6.32-220.31.1.e 2.6.32-220.45.1.el6.i686, 2.6.32-220.77.1.el6.x86_64 x86_64:2.6.32-220.4.2.el6.x86_64, 2.6.32-220.17.1.el6.x86_64, 2.6.32-220.23.1.el6.x86_64, 2.6.3 2.6.32-220.45.1.el6.x86_64, 2.6.32-220.48.1.el6.x86_64, 2.6.32-220.64.1.el6.x86_64, 2.6.32-220. 2.6.32-220.72.2.el6.x86_64, 2.6.32-220.73.1.el6.x86_64, 2.6.32-220.75.1.el6.x86_64, 2.6.32-220.

	(*19) The following kernels are supported x86：2.6.32-279.19.1.el6.i686 x86_64：2.6.32-279.19.1.el6.x86_64 (*20) The following kernels are supported x86：2.6.32-358.6.2.el6.i686, 2.6.32-358.11.1.el6.i686, 2.6.32-358.14.1.el6.i686, 2.6.32-358.23.2.e x86_64：2.6.32-358.6.2.el6.x86_64, 2.6.32-358.11.1.el6. x86_64, 2.6.32-358.14.1.el6. x86_64, 2.6 2.6.32-358.28.1.el6. x86_64, 2.6.32-358.87.1.el6. x86_64 (*21) The following kernels are supported x86：2.6.32-431.1.2.el6.i686, 2.6.32-431.3.1.el6.i686, 2.6.32-431.5.1.el6.i686, 2.6.32-431.17.1.el6 2.6.32-431.20.3.el6.i686, 2.6.32-431.23.3.el6.i686, 2.6.32-431.29.2.el6.i686, 2.6.32-431.72.1.el6.i x86_64：2.6.32-431.1.2.el6.x86_64, 2.6.32-431.3.1.el6. x86_64, 2.6.32-431.5.1.el6. x86_64, 2.6.3 2.6.32-431.20.3.el6. x86_64, 2.6.32-431.23.3.el6. x86_64, 2.6.32-431.29.2.el6. x86_64, 2.6.32-43 2.6.32-431.77.1.el6.x86_64, 2.6.32-431.87.1.el6.x86_64 (*22) The following kernels are supported x86：2.6.32-504.3.3.el6.i686，2.6.32-504.12.2.el6.i686，2.6.32-504.30.3.el6.i686 x86_64：2.6.32-504.3.3.el6.x86_64，2.6.32-504.12.2.el6.x86_64，2.6.32-504.16.2.el6.x86_64，2.6.3 2.6.32-504.40.1.el6.x86_64，2.6.32-504.43.1.el6.x86_64, 2.6.32-504.66.1.el6.x86_64 (*23) The following kernels are supported x86：2.6.18-348.1.1.el5, 2.6.18-348.1.1.el5PAE, 2.6.18-348.6.1.el5, 2.6.18-348.6.1.el5PAE, 2.6.18 2.6.18-348.18.1.el5PAE x86_64：2.6.18-348.1.1.el5, 2.6.18-348.6.1.el5, 2.6.18-348.18.1.el5 (*24) The following kernels are supported x86_64:3.10.0-123.13.2.el7.x86_64, 3.10.0-123.20.1.el7.x86_64 (*25) The following kernels are supported x86_64:3.10.0-229.4.2.el7.x86_64,3.10.0-229.20.1.el7.x86_64, 3.10.0-229.34.1.el7.x86_64 (*26) The following kernels are supported x86_64:3.10.0-327.4.4.el7.x86_64, 3.10.0-327.4.5.el7.x86_64, 3.10.0-327.10.1.el7.x86_64, 3.10.0 3.10.0-327.22.2.el7.x86_64, 3.10.0-327.36.1.el7.x86_64, 3.10.0-327.36.3.el7.x86_64, 3.10.0-327. 3.10.0-327.46.1.el7.x86_64, 3.10.0-327.49.2.el7.x86_64，3.10.0-327.55.2.el7.x86_64，3.10.0-327. 3.10.0-327.58.1.el7.x86_64, 3.10.0-327.62.1.el7.x86_64，3.10.0-327.62.4.el7.x86_64, 3.10.0-327. (*27) The following kernels are supported x86: 2.6.32-573.8.1.el6.i686, 2.6.32-573.12.1.el6.i686, 2.6.32-573.18.1.el6.i686, 2.6.32-573.53.1.e x86_64: 2.6.32-573.8.1.el6.x86_64，2.6.32-573.12.1.el6.x86_64, 2.6.32-573.18.1.el6.x86_64, 2.6.3 (*28) The following kernels are supported x86:2.6.32-642.1.1.el6.i686, 2.6.32-642.6.2.el6.i686, 2.6.32-642.13.1.el6.i686

	x86_64:2.6.32-642.1.1.el6.x86_64, 2.6.32-642.6.1.el6.x86_642.6.32-642.6.2.el6.x86_64, 2.6.32-64 2.6.32-642.15.1.el6.x86_64 (*29) The following kernels are supported x86:2.6.18-416.el5, 2.6.18-416.el5PAE, 2.6.18-419.el5, 2.6.18-419.el5PAE, 2.6.18-426.el5, 2.6.18 x86_64:2.6.18-416.el5, 2.6.18-419.el5, 2.6.18-426.el5 (*30)The following kernels are supported x86_64:3.10.0-514.6.1.el7.x86_64, 3.10.0-514.10.2.el7.x86_64, 3.10.0-514.16.1.el7.x86_64, 3.10 3.10.0-514.26.2.el7.x86_64, 3.10.0-514.36.5.el7.x86_64, 3.10.0-514.44.1.el7.x86_64, 3.10.0-514. (*31)The following kernels are supported x86:2.6.32-696.3.2.el6.i686 2.6.32-696.6.3.el6.i686, 2.6.32-696.10.3.el6.i686, 2.6.32-696.18.7.el6 2.6.32-696.20.1.el6.i686, 2.6.32-696.23.1.el6.i686 x86_64:2.6.32-696.3.2.el6.x86_64, 2.6.32-696.10.3.el6.x86_64, 2.6.32-696.18.7.el6.x86_64, 2.6.3 2.6.32-696.23.1.el6.x86_64 2.6.32-696.23.1.el6.x86_64 (*32)The following kernels are supported x86_64:3.10.0-693.1.1.el7.x86_64, 3.10.0-693.5.2.el7.x86_64, 3.10.0-693.11.1.el7.x86_64, 3.10.0 3.10.0-693.21.1.el7.x86_64, 3.10.0-693.43.1.el7.x86_64 (*33)The following kernels are supported x86_64 3.10.0-862.3.2.el7.x86_64, 3.10.0-862.14.4.el7.x86_64 (*34)The following kernels are supported x86:2.6.32-754.3.5.el6.i686, 2.6.32-754.15.3.el6.i686 ×86_64: 2.6.32-754.3.5.el6.x86_64, 2.6.32-754.15.3.el6.x86_64 (*35)The following kernels are supported x86_64:3.10.0-957.10.1.el7.x86_64, 3.10.0-957.12.2.el7.x86_64 (*36)The following kernels are supported x86_64 3.10.0 1062 .1 .1 .el 7.x86_64
Installation Requirements	HDLM software must be installed prior to installing the HDLM recovery kit. Also, customers wanting to SCSI devices to HDLM devices must run the Installation setup script after configuring the HDLM enviro not be installed.
Adding or Repairing HDLM Paths	When LifeKeeper brings an HDLM resource into service, it establishes a persistent reservation register at that time. If new paths are added after the initial reservation, or if failed paths are repaired and HDLM them, those paths will not be registered as a part of the reservation until the next LifeKeeper quickChec resource. If HDLM allows any writes to that path prior to that point in time, reservation conflicts that occ message file. The HDLM driver will retry these IOs on the registered path resulting in no observable fai quickCheck registers the path, subsequent writes will be successful. The status will be changed to “Off reservation conflict. If the status is “Offline(E)”, customers will need to manually change the status to “O command.
Additional settings for	If Red Hat Enterprise Linux 7.0 or later are used with HDLM Recovery Kit, you must add “HDLM_DLMI default/LifeKeeper. HDLM_DLMMGR=.dlmmgr_exe

RHEL7.x	

		OS version / Architecture										
		RHEL4										
		U1-U4	U3 Security Fix(*7)	U4 Security Fix(*2)	4.5	4.5 Security Fix(*4)	4.6	4.6 Security Fix(*8)	4.7	4.7 Security Fix(*9)	4.8	4.8 Security Fix(*12)
		x86/x86_64										
HDLM	05-80 05-81 05-90	X										
	05-91 05-92	X		X								
	05-93	X(*3)		X	X							
	05-94	X(*3)		X	X	X	X	X				
	6.0.0	X(*3)		X	X	X	X	X	X	X	X	X
	6.0.1	X(*3)		X	X	X	X	X	X	X	X	X
	6.1.0	X(*3)		X	X	X	X	X	X	X	X	X
	6.1.1	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.1.2	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.2.0	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.2.1	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.3.0	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.4.0	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.4.1	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.5.0	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.5.1	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.5.2	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.6.0	X(*3)	X	X	X	X	X	X	X	X	X	X
	6.6.2	X(*3)	X	X	X	X	X	X	X	X	X	X
	7.2.0	X(*3)	X	X	X	X	X	X	X	X	X	X
	7.2.1	X(*3)	X	X	X	X	X	X	X	X	X	X
	7.3.0 or later	X(*3)	X	X	X	X	X	X	X	X	X	X
LifeKeeper	v6.0 v6.0(v6.0.1-2 or later)	X	X	X								
	v6.1	X	X	X								

	(v6.1.0-5 or later)										
	v6.2 (v6.2.0-5 or later)	X	X	X	X	X	X	X			
	v6.2 (v6.2.2-1or later)	X	X	X	X	X	X	X			
	v6.3 (v6.3.2-1or later)	X	X	X	X	X	X	X			
	v6.4 (v6.4.0-10 or later)	X	X	X	X	X	X	X	X	X	
	v7.0 (v7.0.0-5 or later)	X	X	X	X	X	X	X	X	X	X	X
	V 7.1 (v7.1.0-8 or later)	X	X	X	X	X	X	X	X	X	X	X
	V7.2 (v7.2.0-10 or later)	X	X	X	X	X	X	X	X	X	X	X
	V 7.3 (v7.3.0-21 or later)	X	X	X	X	X	X	X	X	X	X	X
	V 7.4 (v7.4.0-63 or later)	X	X	X	X	X	X	X	X	X	X	X
	V 7.5 (v7.5.0-3640 or later)	RHEL4 is not supported in v7.5 or later version of LK.										
HDLM ARK	6.0.1-2	X	X	X	X	X	X	X				
	6.1.0-4	X	X	X	X	X	X	X				
	6.2.2-3	X	X	X	X	X	X	X				
	6.2.3-1	X	X	X	X	X	X	X	X	X	X	X
	6.4.0-2	X	X	X	X	X	X	X	X	X	X	X
	7.0.0-1	X	X	X	X	X	X	X	X	X	X	X
	7.2.0-1	X	X	X	X	X	X	X	X	X	X	X
	X = supported blank = not supported											

		OS version / Architecture											
		RHEL5											
		No Updates	5.1	5.1 Security Fix (*5)	5.2	5.2 Security Fix (*6)	5.3	5.3 Security Fix(*10)	5.4	5.4 Security Fix(*11)	5.5	5.5 Security Fix(*13)	5.6
		x86/x86_64											
HDLM	05-80 05-81 05-90												
	05-91 05-92												
	05-93	X											
	05-94	X	X										
	6.0.0	X	X	X	X	X							
	6.0.1	X	X	X	X	X							
	6.1.0	X	X	X	X	X							
	6.1.1	X	X	X	X	X							
	6.1.2	X	X	X	X	X	X	X	X	X	X	X	X
	6.2.0	X	X	X	X	X	X	X	X	X	X	X	X
	6.2.1	X	X	X	X	X	X	X	X	X	X	X	X
	6.3.0	X	X	X	X	X	X	X	X	X	X	X	X
	6.4.0	X	X	X	X	X	X	X	X	X	X	X	X
	6.4.1	X	X	X	X	X	X	X	X	X	X	X	X
	6.5.0	X	X	X	X	X	X	X	X	X	X	X	X
	6.5.1	X	X	X	X	X	X	X	X	X	X	X	X
	6.5.2	X	X	X	X	X	X	X	X	X	X	X	X
	6.6.0	X	X	X	X	X	X	X	X	X	X	X	X
	6.6.2	X	X	X	X	X	X	X	X	X	X	X	X
	7.2.0	X	X	X	X	X	X	X	X	X	X	X	X
	7.2.1	X	X	X	X	X	X	X	X	X	X	X	X
	7.3.0	X	X	X	X	X	X	X	X	X	X	X	X
	7.3.1	X	X	X	X	X	X	X	X	X	X	X	X
	7.4.0	X	X	X	X	X	X	X	X	X	X	X	X
	7.4.1	X	X	X	X	X	X	X	X	X	X	X	X
	7.5.0	X	X	X	X	X	X	X	X	X	X	X	X
	7.6.0	X	X	X	X	X	X	X	X	X	X	X	X
	7.6.1	X	X	X	X	X	X	X	X	X	X	X	X
	8.0.0	X	X	X	X	X	X	X	X	X	X	X	X

	8.0.1	X	X	X	X	X	X	X	X	X	X	X
	8.1.0	X	X	X	X	X	X	X	X	X	X	X
	8.1.1	X	X	X	X	X	X	X	X	X	X	X
	8.1.2	X	X	X	X	X	X	X	X	X	X	X
	8.1.3	X	X	X	X	X	X	X	X	X	X	X
	8.1.4	X	X	X	X	X	X	X	X	X	X	X
	8.2.0	X	X	X	X	X	X	X	X	X	X	X
	8.4.0	X	X	X	X	X	X	X	X	X	X	X
	8.5.0	X	X	X	X	X	X	X	X	X	X	X
	8.5.1	X	X	X	X	X	X	X	X	X	X	X
	8.5.2	X	X	X	X	X	X	X	X	X	X	X
	8.5.3	X	X	X	X	X	X	X	X	X	X	X
	8.5.4	X	X	X	X	X	X	X	X	X	X	X
	8.6.0	X	X	X	X	X	X	X	X	X	X	X
	8.6.1	X	X	X	X	X	X	X	X	X	X	X
	8.6.2	X	X	X	X	X	X	X	X	X	X	X
	8.6.4	X	X	X	X	X	X	X	X	X	X	X
	8.6.5	X	X	X	X	X	X	X	X	X	X	X
	8.7.0	X	X	X	X	X	X	X	X	X	X	X
	8.7.1	X	X	X	X	X	X	X	X	X	X	X
	8.7.2	X	X	X	X	X	X	X	X	X	X	X
	8.7.3	X	X	X	X	X	X	X	X	X	X	X
LifeKeeper	v6.0 (v6.0.1-2 or later)											
	v6.1 (v6.1.0-5 or later)	X	X									
	v6.2 (v6.2.0-5 or later)	X	X									
	v6.2 (v6.2.2-1 or later)	X	X	X								
	v6.3 (v6.3.2-1 or later)	X	X	X	X	X						
	v6.4 (v6.4.0-10 or later)	X	X	X	X	X	X	X				
	v7.0	X	X	X	X	X	X	X	X	X		

	(v7.0.0-5 or later)											
	v7.1 (v7.1.0-8 or later)	X	X	X	X	X	X	X	X	X	X	X	
	v7.2 (v7.2.0-10 or later)	X	X	X	X	X	X	X	X	X	X	X	
	v7.3 (v7.3.0-21 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v7.4 (v7.4.0-63 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v7.5 (v7.5.0-3640 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.0 (v8.0.0-510 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.1 (v8.1.1-5620 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.2 (v8.2.0-6213 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.2.1 (v8.2.1-6353 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.3.0 (v8.3.0-6389 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.3.1 (v8.3.1-6397 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.3.2 (v8.3.2-6405 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.4.0 (v8.4.0-6427 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v8.4.1 (v8.4.1-6449 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v9.0.0 (v9.0.0.0-6488 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v9.0.1	X	X	X	X	X	X	X	X	X	X	X	X

	(v9.0.1-6492 or later)												
	v9.0.2 (v9.0.2-6213 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v9.1.0 (v9.1.0-6538 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v9.1.1 (v9.1.1-6594 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v9.2.0 (v9.2.0-6629 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v9.2.1 (v9.2.1.0-6653 or later)	X	X	X	X	X	X	X	X	X	X	X	X
	v9.2.2 (v9.2.2-6679 or later)	X	X	X	X	X	X	X	X	X	X	X	X
HDLM ARK	6.0.1-2												
	6.1.0-4	X	X										
	6.2.2-3	X	X	X									
	6.2.3-1	X	X	X	X	X							
	6.4.0-2	X	X	X	X	X	X	X					
	7.0.0-1	X	X	X	X	X	X	X	X	X	X	X	
	7.2.0-1	X	X	X	X	X	X	X	X	X	X	X	X
	8.1.1-5620	X	X	X	X	X	X	X	X	X	X	X	X
	8.2.0-6213	X	X	X	X	X	X	X	X	X	X	X	X
	8.2.1-6213	X	X	X	X	X	X	X	X	X	X	X	X
	8.3.0-6213	X	X	X	X	X	X	X	X	X	X	X	X
	8.4.0-6213	X	X	X	X	X	X	X	X	X	X	X	X
	8.4.1-6213	X	X	X	X	X	X	X	X	X	X	X	X
		X = supported blank = not supported											

Note: RHEL5 is not supported in v9.3 or later version of LK.

	OS version / Architecture										
	RHEL6										
	6	6.1	6.2 Security Fix(*18)	6.3 Security Fix(*19)	6.4 Security Fix(*20)	6.5 Security Fix(*21)	6.6 Security Fix(*22)	6.7 Security Fix(*27)	6.8 Security Fix(*28)	6.9 Security Fix(*31)	Se Fi
	x86/x86_64										

HDLM	6.5.0											
	6.5.1											
	6.5.2	X										
	6.6.0	X										
	6.6.2	X										
	6.6.2-01	X	X									
	7.2.0	X	X	X								
	7.2.1	X	X	X								
	7.3.0	X	X	X								
	7.3.1	X	X	X								
	7.4.0	X	X	X	X	X	X	X	X	X		
	7.4.1	X	X	X	X	X	X	X	X	X		
	7.5.0	X	X	X	X	X	X	X	X	X		
	7.6.0	X	X	X	X	X	X	X	X	X		
	7.6.1	X	X	X	X	X	X	X	X	X		
	8.0.0	X	X	X	X	X	X	X	X	X	X	
	8.0.1	X	X	X	X	X	X	X	X	X	X	
	8.1.0	X	X	X	X	X	X	X	X	X	X	
	8.1.1	X	X	X	X	X	X	X	X	X	X	
	8.1.2	X	X	X	X	X	X	X	X	X	X	
	8.1.3	X	X	X	X	X	X	X	X	X	X	
	8.1.4	X	X	X	X	X	X	X	X	X	X	
	8.2.0	X	X	X	X	X	X	X	X	X	X	
	8.2.1	X	X	X	X	X	X	X	X	X	X	
	8.4.0	X	X	X	X	X	X	X	X	X	X	
	8.5.0	X	X	X	X	X	X	X	X	X	X	
	8.5.1	X	X	X	X	X	X	X	X	X	X	
	8.5.2	X	X	X	X	X	X	X	X	X	X	
	8.5.3	X	X	X	X	X	X	X	X	X	X	
	8.5.4	X	X	X	X	X	X	X	X	X	X	
	8.6.0	X	X	X	X	X	X	X	X	X	X	
	8.6.1	X	X	X	X	X	X	X	X	X	X	
	8.6.2	X	X	X	X	X	X	X	X	X	X	
	8.6.4	X	X	X	X	X	X	X	X	X	X	
	8.6.5	X	X	X	X	X	X	X	X	X	X	
	8.7.0	X	X	X	X	X	X	X	X	X	X	

	8.7.1	X	X	X	X	X	X	X	X	X	X	
	8.7.2	X	X	X	X	X	X	X	X	X	X	
	8.7.3	X	X	X	X	X	X	X	X	X	X	
LifeKeeper	v7.0 (v7.0.0-5 or later)											
	V 7.1 (v7.1.0-8 or later)											
	V7.2 (v7.2.0-10 or later)											
	V 7.3 (v7.3.0-21 or later)	X										
	V 7.4 (v7.4.0-63 or later)	X										
	V 7.5 (v7.5.0-3640 or later)	X	X	X	X							
	v8.0 (v8.0.0-510 or later)	X	X	X	X							
	v8.1 (v8.1.1-5620 or later)	X	X	X	X							
	v8.1.2 (v8.1.2-5795 or later)	X	X	X	X	X						
	v8.2.0 (v8.2.0-6213 or later)	X	X	X	X	X						
	v8.2.1 (v8.2.1-6353 or later)	X	X	X	X	X	X					
	v8.3.0 (v8.3.0-6389 or later)	X	X	X	X	X	X					
	v8.3.1 (v8.3.1-6397	X	X	X	X	X	X					

	or later)											
	v8.3.2 (v8.3.2-6405 or later)	X	X	X	X	X	X	X	X	X		
	v8.4.0 (v8.4.0-6427 or later)	X	X	X	X	X	X	X	X	X		
	v8.4.1 (v8.4.1-6449 or later)	X	X	X	X	X	X	X	X	X		
	v9.0.0 (v9.0.0-6488 or later)	X	X	X	X	X	X	X	X	X		
	v9.0.1 (v9.0.1-6492 or later)	X	X	X	X	X	X	X	X	X		
	v9.0.2 (v9.0.2-6513 or later)	X	X	X	X	X	X	X	X	X		
	v9.1.0 (v9.1.0-6538 or later)	X	X	X	X	X	X	X	X	X		
	v9.1.1 (v9.1.1-6594 or later)	X	X	X	X	X	X	X	X	X		
	v9.1.2 (v9.1.2-6609 or later)	X	X	X	X	X	X	X	X	X	X	
	v9.2.0 (v9.2.0-6629 or later)	X	X	X	X	X	X	X	X	X	X	
	v9.2.1 (v9.2.1-6653 or later)	X	X	X	X	X	X	X	X	X	X	
	v9.2.2 (v9.2.2-6679 or later)	X	X	X	X	X	X	X	X	X	X	
	v9.3 (v9.3.0-6738 or later)	X	X	X	X	X	X	X	X	X	X	
	v9.3.1	X	X	X	X	X	X	X	X	X	X	

	(v9.3.1-6750 or later)											
	v9.3.2 (v9.3.2-6863 or later)	X	X	X	X	X	X	X	X	X	X	
	v9.4.0 (v9.4.0-6959 or later)	X	X	X	X	X	X	X	X	X	X	
	v9.4.1 (v9.4.1-6983 or later)	X	X	X	X	X	X	X	X	X	X	
HDLM ARK	7.0.0-1											
	7.2.0-1	X	X	X	X							
	8.1.1-5620	X	X	X	X							
	8.1.2-5795	X	X	X	X	X						
	8.2.0-6213	X	X	X	X	X						
	8.2.1-6353	X	X	X	X	X	X	X	X	X		
	8.3.0-6389	X	X	X	X	X	X	X	X	X		
	8.3.1-6397	X	X	X	X	X	X	X	X	X		
	8.3.2-6405	X	X	X	X	X	X	X	X	X		
	8.4.0-6427	X	X	X	X	X	X	X	X	X		
	8.4.1-6449	X	X	X	X	X	X	X	X	X		
	9.0.0-6488	X	X	X	X	X	X	X	X	X		
	9.0.1-6492	X	X	X	X	X	X	X	X	X		
	9.0.2-6513	X	X	X	X	X	X	X	X	X		
	9.1.0-6538	X	X	X	X	X	X	X	X	X		
	9.1.1-6594	X	X	X	X	X	X	X	X	X		
	9.1.2-6609	X	X	X	X	X	X	X	X	X	X	
	9.2.0-6629	X	X	X	X	X	X	X	X	X	X	
	9.2.1-6653	X	X	X	X	X	X	X	X	X	X	
	9.2.2-6679	X	X	X	X	X	X	X	X	X	X	
	9.3.0-6738	X	X	X	X	X	X	X	X	X	X	
	9.3.1-6750	X	X	X	X	X	X	X	X	X	X	
	9.3.2-6863	X	X	X	X	X	X	X	X	X	X	
	9.4.0-6959	X	X	X	X	X	X	X	X	X	X	
	9.4.1-6983	X	X	X	X	X	X	X	X	X	X	
	X = supported blank = not supported											

		RHEL7							
		7.0 Security Fix(*24)	7.1 Security Fix(*25)	7.2 Security Fix(*26)	7.3 Security Fix(*30)	7.4 Security Fix(*32)	7.5 Security Fix(*33)	7.6 Security Fix(*35)	7.7 Security Fix(*36)
	x86/x86_64								
HDLM	8.0.1	X	X						
	8.1.0	X	X						
	8.1.1	X	X						
	8.1.2	X	X						
	8.1.3	X	X						
	8.1.4	X	X						
	8.2.0	X	X						
	8.2.1	X	X						
	8.4.0	X	X	X					
	8.5.0	X	X	X					
	8.5.1	X	X	X	X	X			
	8.5.2	X	X	X	X	X			
	8.5.3	X	X	X	X	X			
	8.5.4	X	X	X	X	X			
	8.6.0	X	X	X	X	X			
	8.6.1	X	X	X	X	X	X		
	8.6.2	X	X	X	X	X	X	X (Note: 3)	
	8.6.4	X	X	X	X	X	X	X	
	8.6.5	X	X	X	X	X	X	X	
	8.7.0	X	X	X	X	X	X	X	X
	8.7.1	X	X	X	X	X	X	X	X
	8.7.2	X	X	X	X	X	X	X	X
	8.7.3	X	X	X	X	X	X	X	X
LifeKeeper	v9.0.0 (v9.0.0-6488 or later)	X	X						
	v9.0.1(v9.0.1-6492 or later)	X	X						
	v9.0.2(v9.0.2-6513	X	X	X					

	or later)								
	v9.1.0(v9.1.0-6538 or later)	X	X	X					
	v9.1.1(v9.1.1-6594 or later)	X	X	X	X				
	v9.1.2(v9.1.2-6609 or later)	X	X	X	X				
	v9.2.0(v9.2.0-6629 or later)	X	X	X	X	X			
	v9.2.1(v9.2.1-6653 or later)	X	X	X	X	X			
	v9.2.2(v9.2.2-6679 or later)	X	X	X	X	X			
	v9.3(v9.3.0-6738 or later)	X	X	X	X	X	X		
	v9.3.1(v9.3.1-6750 or later)	X	X	X	X	X	X		
	v9.3.2(v9.3.2-6863 or later)	X	X	X	X	X	X	X	
	v9.4.0 (v9.4.0-6959 or later)	X	X	X	X	X	X	X	X
	v9.4.1 (v9.4.1-6983 or later)	X	X	X	X	X	X	X	X
HDLM ARK	9.0.0-6488	X	X						
	9.0.1-6492	X	X						
	9.0.2-6513	X	X	X					
	9.1.0-6538	X	X	X					
	9.1.1-6594	X	X	X	X				
	9.1.2-6609	X	X	X	X				
	9.2.0-6629	X	X	X	X	X			
	9.2.1-6653	X	X	X	X	X			
	9.2.2-6679	X	X	X	X	X			
	9.3.0-6738	X	X	X	X	X	X		
	9.3.1-6750	X	X	X	X	X	X		
	9.3.2-6863	X	X	X	X	X	X	X	
	9.4.0-6959	X	X	X	X	X	X	X	X
	9.4.1-6983	X	X	X	X	X	X	X	X
X = supported blank = not supported									

Note: 1 If you are running the system with LifeKeeper v9.0.x on RHEL7/7.1/7.2, you need to apply the Bug7205's patch.

Note: 2 The Raw device configuration is not supported on RHEL7/7.1/7.2/7.3/7.4/7.5/7.6/7.7.

Note: 3 Supported with HDLM 8.6.2-02 or later.

Device Mapper Multipath I/O Configurations

Protecting Applications and File Systems That Use Device Mapper Multipath Devices	In order for LifeKeeper to operate with and protect applications or file systems that use Device Mapper Multipath devices, the Device Mapper Multipath (DMMP) Recovery Kit must be installed. Once the DMMP Kit is installed, simply creating an application hierarchy that uses one or more of the multipath device nodes will automatically incorporate the new resource types provided by the DMMP Kit.
Multipath Device Nodes	To use the DMMP Kit, any file systems and raw devices must be mounted or configured on the multipath device nodes rather than on the native /dev/sd* device nodes. The supported multipath device nodes to address the full disk are /dev/dm-#, /dev/mapper/<uuid>, /dev/mapper/<user_friendly_name> and /dev/mpath/<uuid>. To address the partitions of a disk, use the device nodes for each partition created in the /dev/mapper directory.
Use of SCSI-3 Persistent Reservations	The Device Mapper Multipath Recovery Kit uses SCSI-3 persistent reservations with a "Write Exclusive" reservation type. This means that devices reserved by one node in the cluster will remain read-accessible to other nodes in the cluster, but those other nodes will be unable to write to the device. Note that **this does not mean** that you can expect to be able to mount file systems on those other nodes for ongoing read-only access. LifeKeeper uses the sg_persist utility to issue and monitor persistent reservations. If necessary, LifeKeeper will install the sg_persist(8) utility. SCSI-3 Persistent Reservations must be enabled on a per LUN basis when using EMC Symmetrix (including VMAX) arrays with multipathing software and LifeKeeper. This applies to both DMMP and PowerPath.
Hardware Requirements	The Device Mapper Multipath Kit has been tested by SIOS Technology Corp. with the EMC CLARiiON CX300, the HP EVA 8000, HP MSA1500, HP P2000, the IBM SAN Volume Controller (SVC), the IBM DS8100, the IBM DS6800, the IBM ESS, the DataCore SANsymphony, and the HDS 9980V. Check with your storage vendor to determine their support for Device Mapper Multipath. Enabling support for the use of reservations on the CX300 and the VNX Series requires that the hardware handler be notified to honor reservations. Set the following parameter in */etc/multipath.conf* for this array: `hardware_handler "3 emc 0 1"` The HP MSA1500 returns a reservation conflict with the default path checker setting (tur). This will cause the standby node to mark all paths as failed. To avoid this condition, set the following parameter in */etc/multipath.conf* for this array: `path_checker readsector0` The HP 3PAR F400 returns a reservation conflict with the default path checker. To avoid this conflict, set (add) the following parameter in */etc/default/LifeKeeper* for this array:

	`DMMP_REGISTRATION_TYPE=hba` For the HDS 9980V the following settings are required: • Host mode: 00 • System option: 254 (must be enabled; global HDS setting affecting all servers) • Device emulation: OPEN-V Refer to the HDS documentation “Suse Linux Device Mapper Multipath for HDS Storage” or “Red Hat Linux Device Mapper Multipath for HDS Storage” v1.15 or later for details on configuring DMMP for HDS. This documentation also provides a compatible multipath.conf file. For the EVA storage with firmware version 6 or higher, DMMP Recovery Kit v6.1.2-3 or later is required. Earlier versions of the DMMP Recovery Kit are supported with the EVA storage with firmware versions prior to version 6.
Multipath Software Requirements	For SUSE, multipath-tools-0.4.5-0.14 or later is required. • For Red Hat, device-mapper-multipath-0.4.5-12.0.RHEL4 or later is required. • It is advised to run the latest set of multipath tools available from the vendor. The feature content and the stability of this multipath product are improving at a very fast rate.
Linux Distribution Requirements	Some storage vendors such as IBM have not certified DMMP with SLES 11 at this time. • SIOS Technology Corp. is currently investigating reported issues with DMMP, SLES 11, and EMCs CLARiiON and Symmetrix arrays.
Transient path failures	While running IO tests on Device Mapper Multipath devices, it is not uncommon for actions on the SAN, for example, a server rebooting, to cause paths to temporarily be reported as failed. In most cases, this will simply cause one path to fail leaving other paths to send IOs down resulting in no observable failures other than a small performance impact. In some cases, multiple paths can be reported as failed leaving no paths working. This can cause an application, such as a file system or database, to see IO errors. There has been much improvement in Device Mapper Multipath and the vendor support to eliminate these failures. However, at times, these can still be seen. To avoid these situations, consider these actions: 1. Verify that the multipath configuration is set correctly per the instructions of the disk array vendor. 2. Check the setting of the “failback” feature. This feature determines how quickly a path is reactivated after failing and being repaired. A setting of “immediate” indicates to resume use of a path as soon as it comes back online. A setting of an integer indicates the number of seconds after a path comes back online to resume using it. A setting of 10 to 15 generally provides sufficient settle time to avoid thrashing on the SAN. 3. Check the setting of the “no_path_retry” feature. This feature determines what Device Mapper Multipath should do if all paths fail. We recommend a setting of 10 to 15. This allows some ability to “ride out” temporary events where all paths fail while still providing a reasonable recovery time. The LifeKeeper DMMP kit will monitor IOs to the storage and if they are not responded to within four minutes LifeKeeper will switch the resources to the standby server. **NOTE:** LifeKeeper

	does not recommend setting “no_path_retry” to “queue” since this will result in IOs that are not easily killed. The only mechanism found to kill them is on newer versions of DM, the settings of the device can be changed: `/sbin/dmsetup message -u 'DMid' 0 fail_if_no_path` This will temporarily change the setting for no_path_retry to fail causing any outstanding IOs to fail. However, multipathd can reset no_path_retry to the default at times. When the setting is changed to fail_if_no_path to flush failed IOs, it should then be reset to its default prior to accessing the device (manually or via LifeKeeper). If “no_path_retry” is set to “queue” and a failure occurs, LifeKeeper will switch the resources over to the standby server. However, LifeKeeper will not kill these IOs. The recommended method to clear these IOs is through a reboot but can also be done by an administrator using the dmsetup command above. If the IOs are not cleared, then data corruption can occur if/when the resources are taken out of service on the other server thereby releasing the locks and allowing the “old” IOs to be issued.

5.3.2.10. LifeKeeper I/O Fencing Introduction

I/O fencing is the locking away of data from a malfunctioning node preventing uncoordinated access to shared storage. In an environment where multiple servers can access the same data, it is essential that all writes are performed in a controlled manner to avoid data corruption. Problems can arise when the failure detection mechanism breaks down because the symptoms of this breakdown can mimic a failed node. For example, in a two-node cluster, if the connection between the two nodes fails, each node would “think” the other has failed, causing both to attempt to take control of the data resulting in data corruption. I/O fencing removes this data corruption risk by blocking access to data from specific nodes.

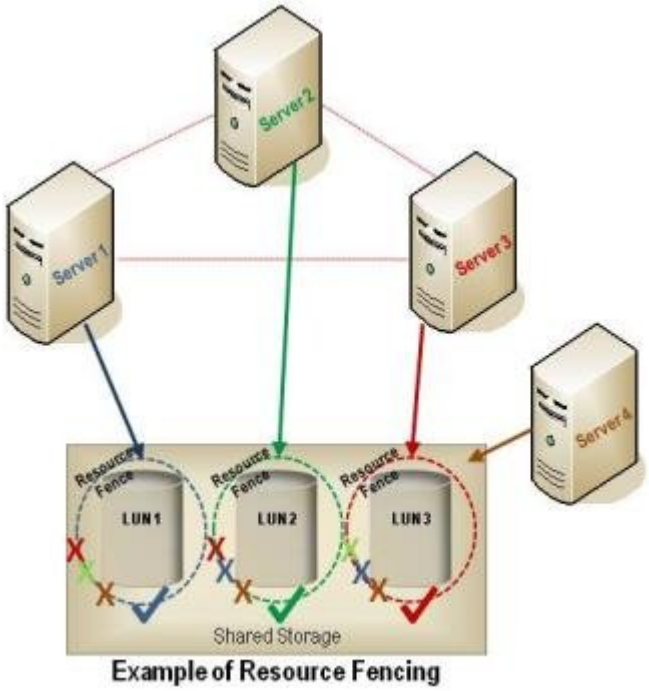
For DataKeeper, please refer to DataKeeper I-O Fencing Introduction.

5.3.2.10.1. SCSI Reservations

Storage Fencing Using SCSI Reservations

While LifeKeeper for Linux supports both resource fencing and node fencing, its primary fencing mechanism is storage fencing through SCSI reservations. This fence, which provides the highest level of data protection for shared storage, allows for maximum flexibility and maximum security providing very granular locking to the LUN level. The underlying shared resource (LUN) is the primary quorum device in this architecture. Quorum can be defined as exclusive access to shared storage, meaning this shared storage can only be accessed by one server at a time. The server who has quorum (exclusive access) owns the role of “primary.” The establishment of quorum (who gets this exclusive access) is determined by the “quorum device.”

As stated above, with reservations enabled, the quorum device is the shared resource. The shared resource establishes quorum by determining who owns the reservation on it. This allows a cluster to continue to operate down to a single server as long as that single server can access the LUN.



Example of Resource Fencing

SCSI reservations protect the shared user data so that only the system designated by LifeKeeper can modify the data. No other system in the cluster or outside the cluster is allowed to modify that data. SCSI reservations also allow the application being protected by LifeKeeper to safely access the shared user data when there are multiple server failures in the cluster. A majority quorum of servers is not required; the only requirement is establishing ownership of the shared data.

Adding quorum/witness capabilities provides for the establishment of quorum membership. Without this membership, split-brain situations could result in multiple servers, even all servers, killing each other. Watchdog added to configurations with reservations enabled provides a mechanism to recover from partially hung servers. In cases where a hung server goes undetected by LifeKeeper, watchdog will

begin recovery. Also, in the case where a server is hung and not able to detect that the reservation has been stolen, watchdog can reboot the server to begin its recovery.

Alternative Methods for I/O Fencing

In addition to resource fencing using SCSI reservations, LifeKeeper for Linux also supports disabling reservations. Regardless of whether reservations are enabled or disabled, there are two issues to be aware of:

- Access to the storage must be controlled by LifeKeeper.
- Great care must be taken to ensure that the storage is not accessed unintentionally such as by mounting file systems manually, fsck manually, etc.

If these two rules are followed and reservations are enabled, LifeKeeper will prevent most errors from occurring. With reservations disabled (alone), there is no protection. Therefore, other options must be explored in order to provide this protection. The following sections discuss these different fencing options and alternatives that help LifeKeeper provide a reliable configuration even without reservations.

5.3.2.10.2. Disabling Reservations

While reservations provide the highest level of data protection for shared storage, in some cases, the use of reservations is not available and must be disabled within LifeKeeper. With reservations disabled, the storage no longer acts as an arbitrator in cases where multiple systems attempt to access the storage, intentionally or unintentionally.

Consideration should be given to the use of other methods to fence the storage through cluster membership which is needed to handle system hangs, system busy situations and any situation where a server can appear to not be alive.

The key to a reliable configuration without reservations is to “know” that when a failover occurs, the “other” server has been powered off or power cycled. There are four fencing options that help accomplish this, allowing LifeKeeper to provide a very reliable configuration, even without SCSI reservations. These include the following:

- STONITH (Shoot the Other Node in the Head) using a highly reliable interconnect, i.e. serial connection between server and STONITH device. STONITH is the technique to physically disable or power-off a server when it is no longer considered part of the cluster. LifeKeeper supports the ability to power off servers during a failover event thereby insuring safe access to the shared data. This option provides reliability similar to reservations but is limited to two nodes physically located together.

- Quorum/Witness – Quorum/witness servers are used to confirm membership in the cluster, especially when the cluster servers are at different locations. While this option can handle split-brain, it, alone, is not recommended due to the fact that it does not handle system hangs.

- Watchdog – Watchdog monitors the health of a server. If a problem is detected, the server with the problem is rebooted or powered down. This option can recover from a server hang; however, it does not handle split-brain; therefore this option alone is also not recommended.

- CONFIRM_SO – This option requires that automatic failover be turned off, so while very reliable (depending upon the knowledge of the administrator), it is not as available.

While none of these alternative fencing methods alone are likely to be adequate, when used in combination, a very reliable configuration can be obtained.

Non-Shared Storage

If planning to use LifeKeeper in a non-shared storage environment, the risk of data corruption that exists with shared storage is not an issue; therefore, reservations are not necessary. However, partial or full resyncs and merging of data may be required. To optimize reliability and availability, the above options should be considered with non-shared storage as well.

Note: For further information comparing the reliability and availability of the different options, see the I/O Fencing Comparison Chart.

It is important to note that no option will provide complete data protection, but the following combination will provide almost the same level of protection as reservations.

Configuring I/O Fencing Without Reservations

To configure a cluster to support node fencing, complete the following steps:

1. Stop LifeKeeper.

2. Disable the use of SCSI reservations within LifeKeeper. This is accomplished by editing the LifeKeeper defaults file, /etc/default/LifeKeeper, on all nodes in the cluster. Add or modify the Reservations variable to be “none”, e.g. `RESERVATIONS=”none”`. (**Note:** This option should only be used when reservations are not available.)

3. Obtain and configure a STONITH device or devices to provide I/O fencing. Note that for this configuration, STONITH devices should be configured to do a system “poweroff” command rather than a “reboot”. Take care to avoid bringing a device hierarchy in service on both nodes simultaneously via a manual operation when LifeKeeper communications have been disrupted for some reason.

4. If desired, obtain and configure a quorum/witness server(s). For complete instructions and information on configuring and using a witness server, see Quorum/Witness Server Support Package topic.

Note: The quorum/witness server should reside at a site apart from the other servers in the cluster to provide the greatest degree of protection in the event of a site failure.

5. If desired, configure watchdog. For more information, see the Watchdog topic.

5.3.2.10.2.1. I/O Fencing Chart

	Split-Brain	Hung Server
Reservations On		
Alone		
Quorum/Witness		
Watchdog		
Watchdog & Quorum/ Witness		
STONITH (serial)		
Reservations Off		
Nothing		
STONITH (serial)		
CONFIRM_SO*		
Quorum/Witness		
Watchdog		
Non-Shared Storage		
Default Features		
Quorum/Witness		
CONFIRM_SO*		
Watchdog		
STONITH (serial)		

* *While CONFIRM_SO is highly reliable (depending upon the knowledge of the administrator), it has lower availability due to the fact that automatic failover is turned off.*

5.3.2.10.3. Quorum/Witness

Quorum/Witness Server Support Package for LifeKeeper

Feature Summary

The Quorum/Witness Server Support Package for LifeKeeper (steeleye-lkQWK, hereinafter “Quorum/Witness Package”) combined with the existing failover process of the LifeKeeper core allows system failover to occur with a greater degree of confidence in situations where total network failure could be common. This effectively means that local site failovers and failovers to nodes across a WAN can be done while greatly reducing the risk of split-brain situations.

In a distributed system that takes network partitioning into account, there is a concept called quorum to obtain consensus across the cluster. A node having quorum is a node that can obtain consensus of all the clusters and is allowed to bring resources in service. On the other hand, a node not having quorum is a node that cannot obtain consensus of all the clusters and it is not allowed to bring resources in service. This will prevent split brain from happening. To check whether a node has quorum is called quorum check. It is expressed as “quorum check succeeded” if it has quorum, and “quorum check failed” if it does not have quorum.

In case of a communication failure, using one node where failure occurred and another multiple nodes (or other devices) will allow a node to get a “second opinion” on the status of the failing node. The node to get a “second opinion” is called a witness node (or a witness device), and getting a “second opinion” is called witness checking. When determining when to fail over, the witness node (the witness device) allows resources to be brought in service on a backup server only in cases where it verifies the primary server has failed and is no longer part of the cluster. This will prevent failovers from happening due to simple communication failures between nodes when those failures don’t affect the overall access to, and performance of, the in-service node. During actual operation, the witness node (the witness device) will be consulted when LifeKeeper is started or the failed communication path is restored. Witness checking can only be performed for nodes having quorum.

Package Installation and Configuration

The Quorum/Witness Server Support Package for LifeKeeper will need to be installed on every node in the cluster that uses quorum/witness functionality, including a witness-only node. The only configuration requirement for the witness node is to create appropriate comm paths. When using a quorum mode with tcp_remote, LifeKeeper does not need to be installed on the host which was set as `QUORUM_MODE in /etc/default/LifeKeeper` configuration file.

The general process for setting up quorum/witness functionality will involve the following steps:

1. Set up the server and make sure that it can communicate with other servers.

2. Install LifeKeeper on the server. During the installation, enable “Use Quorum / Witness functions”

with the setup command and install the quorum/witness package as well.

3. Create appropriate communication paths between the nodes including witness-only nodes.
4. Configure quorum/witness.

When the above steps are completed, the quorum/witness functions will be activated in the cluster and quorum checking and witness checking will be performed before failovers are allowed.

See the Configurable Components section below for additional configuration options.

> **Note:** Any node that has the quorum/witness package installed can participate in quorum/witness functionality. The witness-only nodes will have communication paths with all the other nodes and will not host any protected resources.

Configurable Components

The quorum/witness package contains two configurable modes: quorum and witness. By default, installing the quorum/witness package will enable both quorum and witness modes.

The behavior of these modes can be customized via the `/etc/default/LifeKeeper` configuration file, and the quorum and witness modes can be individually adjusted. The package installs default settings into the configuration file when it is installed, *majority* being the default quorum mode and *remote_verify* being the default witness mode. An example is shown below:

```
QUORUM_MODE=majority
WITNESS_MODE=remote_verify
```

Available Quorum Modes

Four quorum checking modes are available which can be set via the QUORUM_MODE setting in `/etc/default/LifeKeeper`.

QUORUM_MODE	Description
majority (default)	With majority as the quorum mode setting quorum checks occur via SPS for Linux communication paths. A node has quorum when it is able to communicate with the majority of the nodes in the cluster. This quorum mode is available on clusters with three or more nodes. A witness – only node needs to be added when using a two-node configuration. See “majority mode” for details.
tcp_remote	Checks the connection to the TCP/IP service on the specified port for the host independent from the communication path. It is determined that the node has quorum when it is able to communicate with the majority of the nodes in the cluster. A host for connection checking is required separately. See “tcp_remote mode” for details.
storage	With storage as the quorum mode setting quorum checks occur using a “shared storage” device. A node has quorum when it is able to access the shared storage

	device and update its own quorum object. A cluster is considered to have quorum consensus with this mode when each node is able to access the shared storage device This mode can be used for 2, 3 or 4 node clusters. Shared storage devices can be a block storage device shared to all nodes, a file accessed via a NFS share on all nodes or an Amazon S3 storage object. The "shared storage" used for this solution must be obtained separately. See "storage mode" for details. When *storage* is selected for the quorum mode, the witness mode, which is describe later, must also be set to *storage*.
none/off	Quorum checking is disabled. With this configuration, quorum checking is always determined to be successful.

Available Witness Modes

Three witness modes are available which can be set via the WITNESS_MODE setting in `/etc/default/LifeKeeper`.

WITNESS_MODE	Description
remote_verify (default)	Consults all the other nodes in the cluster about their view of the status of a node which appears to be failing. If any node determines that there is no failure, witness checking determines that there is no failure. If all the nodes determine that there is failure, witness checking determines that the node is failing.
storage	A witness mode where shared storage is used as a witness device. The share storage device is used to "share" status information between nodes in the cluster. Each node updates its own information and reads the other nodes information. If a node detects that information for another node is not being updated then that node will be considered failed. See "storage mode" for details. When storage is selected for the witness mode, then storage must be selected for quorum mode. See above.
none/off	In this mode, witness checking is disabled. With this setting, it is always determined that there is no failure.

Note: It would be unnecessary for witness checks to ever be performed by servers acting as dedicated quorum/witness nodes that do not host resources; therefore, this setting should be set to none/off on these servers.

Supported Combinations of a Quorum Mode and Witness Mode

LifeKeeper supports the following combinations.

		QUORUM_MODE			
		majority	***tcp_remote***	***storage***	***none/off***
WITNESS_MODE	***remote_verify***	Supported 3 or more	Supported 3 or more	Not supported	Supported 3 or more

		nodes	nodes		nodes
	storage	Not Supported	Not Supported	Supported Between 2 and 4 nodes	Not supported
	none/off	Supported 3 or more nodes	Supported 2 or more nodes	Not supported	Supported

Available Actions When Quorum is Lost

The quorum/witness package offers three different options for how the system should react if quorum is lost — *"fastboot"*, *"fastkill"* and *"osu"*. These options can be selected via the QUORUM_LOSS_ACTION setting in `/etc/default/LifeKeeper`. All three options take the system's resources out of service; however, they each allow a different behavior.

Mode	Description
fastboot	The system will be **immediately** rebooted when a loss of quorum is detected (from a communication path failure). Although this is an aggressive option, it ensures that the system will be disconnected from any external resources right away. In many cases, such as with storage-level replication, this immediate release of resources is desired. Two important notes on this option are: 1. 2. The system performs an **immediate** hard reboot without first performing any shut-down procedure; no shutdown tasks are performed (disk syncing, etc.). 3. The system will come back up performing normal startup routines, including negotiating storage and resource access, etc.
fastkill (default)	The fastkill option is very similar to the fastboot option, but instead of a hard reboot, the system will **immediately** halt when quorum is lost. As with the fastboot option, no tasks are performed (disk syncing, etc.), and the system will then need to be manually started and will come back up performing normal startup routines, including negotiating storage and resource access, etc.
osu	This is the least aggressive option, leaving the system operational but taking resources out of service on the system where quorum is lost. In some cluster configurations, this is all that is needed, but it may not be strong enough or fast enough in others.

5.3.2.10.3.1. Majority Mode

Quorum checking is performed via SPS for Linux communication paths. A node has quorum when it is able to communicate with the majority of the nodes in the cluster.This quorum mode is available on clusters with three or more nodes. A node dedicated for witness checking needs to be added when using a two-node configuration.

Note: Due to requirements for node majority, it is recommended that clusters always be configured with an odd number of nodes (the count includes the quorum node).

Stopping the cluster with quorum:

1. Stop the target
2. Stop the source
3. Stop the witness

Starting the cluster with quorum:

1. Start the witness
2. Start the target
3. Start the source

Majority Mode Configuration

Set QUORUM_MODE to majority in `/etc/default/LifeKeeper`. No other setting is required for this mode.

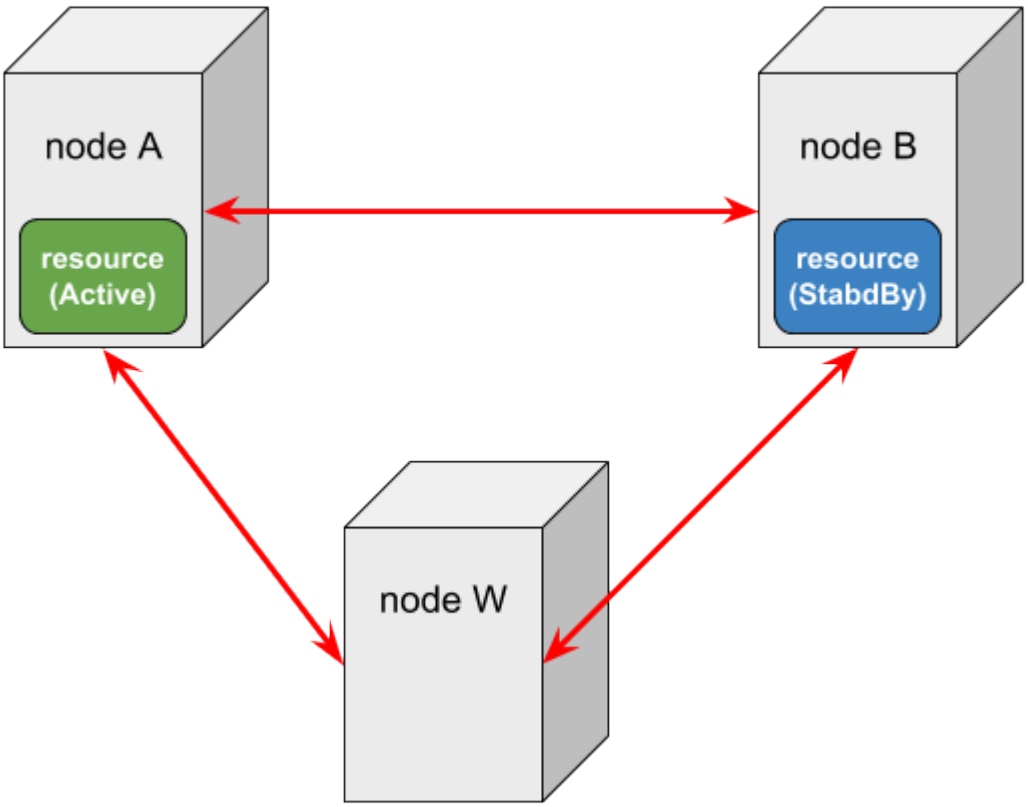
Available Witness mode settings for Majority Mode

The following witness modes are available for majority mode. For details on each mode, please refer to "Available Witness Mode".

- *remote_verify*
- *none/off*

Expected Behaviors for Majority Mode (Assuming Default Modes)

The scenarios listed below shows the SPS for Linux behavior of a three-node cluster with Node A (resources are in-service), Node B (resources are on stand-by), and Node W (a witness-only node without protected resources).



The following three events may change the resource status on a node failure.

- COMM_DOWN event
  An event called when all the communication paths between nodes are disconnected.

- COMM_UP event
  An event called when communication paths are recovered from a COMM_DOWN state.

- LCM_AVAIL event
  An event called after LCM initialization is completed and it is called only once when starting LifeKeeper. Once this state has been reached heartbeat transmission to other nodes in the cluster begins over the established communication paths. It also ready to receive heartbeat requests from other nodes cluster. LCM_AVAIL is always processed before processing a COMM_UP event.

Scenario 1

A communication path fails between Node A and B

In this case, the following will happen:

1. Both Node A and Node B will begin processing COMM_DOWN events, though not necessarily at exactly the same time.

2. Both nodes will perform the quorum check and determine that they still have quorum (since both

Node A and B can see Node W and they have communication with two of the three known nodes, they think that they are in the majority).

3. Each will consult the other nodes with whom they can still communicate about the true status of the server with whom they've lost communications (witness checking). In this scenario, this means that Node A will consult Node W about Node B's status and Node B will also consult Node W about Node A's status.

4. Node A and Node B will both determine that the other is still alive by having consulted Node W and no failover processing will occur. Resources will be left in service on Node A.

Scenario 2

A communication path fails between Node A and Node W

Since all nodes can and will act as witness nodes when the quorum/witness package is installed, this scenario is the same as the previous. In this case, Node A and Node W will determine that the other is still alive by consulting with Node B.

Scenario 3

Node A fails and stops

In this case, Node B will do the following:

1. Begin processing the COMM_DOWN event from Node A.

2. Determine that it can still communicate with Node W and thus has quorum.

3. Verify via Node W that Node A really appears to be lost and, begin the usual failover activity.

4. Node B will continue processing the event and bring the protected resources in service.

With resources being in-service on Node B, Node A is powered on and establishes communications with the other nodes

In this case, Node A will process an LCM_AVAIL event. Node A will determine that it has quorum and not bring resources in service because they are currently in service on Node B. Next, a COMM_UP event will be processed between Node A and Node B and also between Node A and Node W (processed twice at Node A). Each node will determine that it has quorum during the COMM_UP events and will not bring resources in service because they are currently in service on Node B.

With resources being in-service on Node B, Node A is powered on and cannot establish communications to the other nodes

In this case, Node A will process an LCM_AVAIL event and Node B and Node W will do nothing since they can't communicate with Node A. Node A will determine that it does not have quorum since it can

only communicate with one of the three nodes (Node A itself). Because it does not have quorum, Node A will not bring resources in service.

Scenario 4

A failure occurs with the network for Node A (Node A is running without communications to other nodes)

In this case, Node A will do the following:

1. Begin processing a COMM_DOWN event from Node B (processing of a COMM_DOWN event from Node W is started almost simultaneously).
2. Determine that it cannot communicate with Node B or Node W and thus does not have quorum.
3. LifeKeeper takes action based on the QUORUM_LOSS_ACTION (see Quorum/Witness for more details).

Node B will do the following:

1. Begin processing a COMM_DOWN event from Node A.
2. Determine that it can still communicate with Node W and thus has quorum.
3. Verify via Node W that Node A really appears to be lost (witness checking) and, begin the usual failover activity.
4. Node B will now have the protected resources in service.

With resources being in-service at Node B, communication resumes for Node A

In this case, Node B will process a COMM_UP event, determine that it has quorum (all three of the nodes are visible) and that it has the resources in service. Node A will process a COMM_UP event, determine that it also has quorum and that the resources are in service on Node B. Node A will not bring resources in service at this time.

Scenario 5

All three nodes lose communications with each other

In this case, Node A will do the following:

1. Begin processing COMM_DOWN events between node B. (Processing of a COMM_DOWN event from Node W is started almost simultaneously).
2. Determine that it cannot communicate with Node B or Node W and thus does not have quorum.

3. LifeKeeper takes action based on the QUORUM_LOSS_ACTION (see Quorum/Witness for more details).

Node B will do the following:

1. Begin processing a COMM_DOWN event between Node A.

2. Determine that it cannot communicate with Node A or Node W and thus does not have quorum.

3. Since it does not have the resources in service, no QUORUM_LOSS_ACTION will occur.

If all the communication paths are recovered, Node A will bring the resources in service. The following requirements should be met for this behavior.

- As initialization behavior, AUTORES_ISP is set for the resources on Node A.
- The Resource Priority value is the highest on Node A.

5.3.2.10.3.2. tcp_remote Mode

In this setting Quorum is determined by checking the ability to connect to TCP/IP services on remote hosts. Connections are done via TCP/IP to a specific port on a host and are done independent of any of the defined communication paths. Being able to connect to the majority of the specified hosts will determine if the node has quorum. Host and port combinations are defined via the QUORUM_HOSTS setting discussed below. This mode is also available with a two-node cluster, however, three nodes are required when using a witness node for *remote_verify*.

Note: Due to majority-based quorum, it is recommended that the hosts always be specified with an odd number of nodes.

Note: Due to the inherent flexibility and complexity of this mode, it should be used with caution by someone experienced with both LifeKeeper and the particular network/cluster configuration involved.

tcp_remote Mode Configuration

Set QUORUM_MODE> to *tcp_remote* in `/etc/default/LifeKeeper`. The following configuration settings are required when using *tcp_remote*:

- QUORUM_HOSTS – This is a comma delimited list of host:port values used to define the hosts and ports to connect to when checking for quorum.
- QUORUM_TIMEOUT_SECS – This is the time allowed for TCP/IP connections to complete. It defaults to 20 seconds.

See "Quorum Parameter List" for more information.

Available Witness Mode setting with tcp_remote mode

The following witness mode settings are available with *tcp_remote*. Refer to "Available Witness Mode" for more details on each mode.

- *remote_verify*
- *none/off*

5.3.2.10.3.3. Storage Mode

With this mode each node writes information about itself to a shared storage device on a regular basis and periodically reads the information written by the other nodes. A cluster is considered to have quorum consensus when each node is able to access the shared storage device and update its quorum object as well as see that the quorum objects for all other nodes are being updated. The node information located on the shared storage device is called a quorum (QWK) object or QWK object for short. QWK objects are required for every node configured in the cluster.

Quorum checking determines that a node has quorum when it has access to the shared storage device. Witness checking accesses the QWK objects for the other nodes to determine that node's current state. During a check it is verifying that updates to the QWK objects of the other nodes are still occurring on a regular basis. If no updates have occurred on a particular node after a certain period of time, the node will be considered in a failed state. During this time the checking node will update its own QWK object. Witness checking is performed when quorum checking is performed.

When *"storage"* is selected for quorum mode, *"storage"* must be selected for witness mode.

This quorum mode setting can be used for a two-node, three node, or four node cluster. The shared storage used for storing QWK objects for all the nodes must be configured separately. If a node loses access to the shared storage, it affects bringing resources in service. Select a shared storage device which is always accessible from all the nodes.

Note: Using Storage for the Quorum Mode requires a storage device that can be accessed by all nodes in the cluster. The storage solution is to be used for quorum / witness functionality and must not be protected by SPS for Linux. For supported storage solutions see the topic on Available Share Storage.

Note: In order to use this mode, initialization of the QWK object is required after configuring (See "Storage Mode Configuration"). In addition, reinitialization is necessary to add/delete nodes in the cluster or change the configuration after initial configuration.

Note: This mode cannot be used if the names of the nodes in the cluster are similar such that the only difference is in the use of '-' and '.'.
For example a cluster with nodes named lifekeeper-sios and lifekeeper.sios would not be allowed but a cluster with nodes named lifekeeper-sios and lifekeeper.sios2 would be acceptable.

Available Shared Storage

The purpose of the quorum/witness function is to avoid a split brain scenario. Therefore, correctly configuring the storage quorum mode choice is critical to ensure all nodes in the cluster can see all the

QWK objects. This is accomplished by placing all the QWK objects in the same type of shared storage: block devices, regular files, or S3 objects.

The available shared storage choices are shown below. Specify the type of shared storage being used via the QWK_STORAGE_TYPE setting in the `/etc/default/LifeKeeper` configuration file.

<table>
<tr><th>QWK_STORAGE_TYPE</th><th>QWK Object Location</th></tr>
<tr><td rowspan="2">block</td><td>When using physical storage, RDM (physical compatibility), iSCSI (in-VM initiator) for shared storage, allocate one QWK object in one of the following ways:

(a) 1 QWK object = 1 partition

(b) 1 QWK object = 1 LU

In the case of (a), since multiple hosts will write to one LU, align the offset with 4K (sector size of the storage device) within the LU of the partition. Also, do not mix partitions used for other purposes.
In the case of (b), do not create partitions within LU.
No file system needs to be created for either (a) or (b).</td></tr>
<tr><td>When using VMDK for shared storage, allocate one QWK object as follows:

1 QWK object = 1 VMDK

Do not create partitions. Also, no file system needs to be created.
Set the provisioning option for VMDK as follows:

thick (eager zeroed)</td></tr>
<tr><td>file</td><td>When using NFS for shared storage, allocate one QWK object as follows:

1 QWK object = 1 regular file system in the NFS file system

Set the export option for the NFS server as follows:

rw,no_root_squash,sync,no_wdelay

Set the mount option for the NFS server as follows:

soft,timeo=20,retrans=1,intr,noac</td></tr>
</table>

	Configure `/etc/fstab` to mount automatically after rebooting the OS.
aws_s3	When using Amazon Simple Storage Service (S3) for shared storage, allocate one QWK object as follows: **1 QWK object = 1 S3 object** Use S3 in a region different from the region where LifeKeeper is running. Also, due to the Amazon S3 Data Consistency Model, the old data may be returned if the request is made right after updating the QWK objects; therefore, two QWK objects can be specified on one node when using S3(this is only available with S3). All of the nodes configured in the cluster need to satisfy the following requirements: • AWS Command Line Interface (AWS CLI) is installed and available to the root user. See "Installing the AWS Command Line Interface". • Ability to access the endpoint in Amazon S3 (the AWS region and endpoint) with the HTTP and HTTPS protocols. • Ability to access the S3 object as the root user by properly configuring the IAM role for EC2 and the AWS CLI **Note:** If the path name for the AWS CLI executable files are not already specified as a part of the "PATH" parameter in the LifeKeeper defaults file `/etc/default/LifeKeeper`, you must append the path to the AWS CLI exectuables for LifeKeeper to function correctly when using S3 objects.

The size of 1 QWK object is 4096 bytes.

Quorum witness checking performs a read and/or /write to its own QWK object and will only read the QWK objects of other nodes. Set the access rights appropriately (be careful of permission restrictions such as granting Persistent Reservation to the shared storage).

Note: 4K native disk of vSphere cannot be used as QWK object's shared storage.

Storage Mode Configuration

QUORUM_MODE and WITNESS_MODE should be configured as storage in the `/etc/default/LifeKeeper` configuration file. The following configuration parameters are also available when using storage:

- QWK_STORAGE_TYPE – Specifies the type of shared storage being used.

- QWK_STORAGE_HBEATTIME – Specifies the interval in seconds between reading and writing the QWK objects.
- QWK_STORAGE_NUMHBEATS – Specifies the number of consecutive heartbeat checks that when missed indicates the target node has failed. A missed heartbeat occurs when the QWK object has not been updated since the last check.
- QWK_STORAGE_OBJECT_ – Specifies the path to the QWK object for each node in the cluster. Entries for all nodes in the cluster are required.
- HTTP_PROXY, HTTPS_PROXY, NO_PROXY – Set this parameter when using HTTP proxy for accessing the service endpoint. The value set here will be passed to AWS CLI.

See the "Quorum Parameter List" for more information.

How to use Storage Mode

Initialization is required in order to use the storage quorum mode. The initialization steps for all the nodes in the cluster are as follows.

1. Set up all the nodes and make sure that they can communicate with each other.
2. On all the nodes run the SPS for Linux setup and enable "Use Quorum/Witness Functions" to install the Quorum/Witness package.
3. Create communication paths between all the nodes.
4. Configure the quorum setting in the `/etc/default/LifeKeeper` configuration file on all nodes.
5. Run the `qwk_storage_init command` on all nodes. This command will wait until the initialization of the QWK objects on all nodes is complete. Quorum/Witness functions will become available in the storage mode once the init completes on all nodes.

Reinitialization is necessary to add/delete cluster nodes after initial configuration, or when quorum parameters are changed in the `/etc/default/LifeKeeper`configuration file. Please reinitialize according to the following steps.

1. Execute the `qwk_storage_exit` command on all nodes.
2. Delete communication paths between the node that is being deleted and all the other nodes. Create communication paths between the node that is being added and all the other nodes.
3. Modify the quorum parameters in the `/etc/default/LifeKeeper` configuration file on all nodes.
4. Execute the `qwk_storage_init` command on all nodes.

Troubleshooting

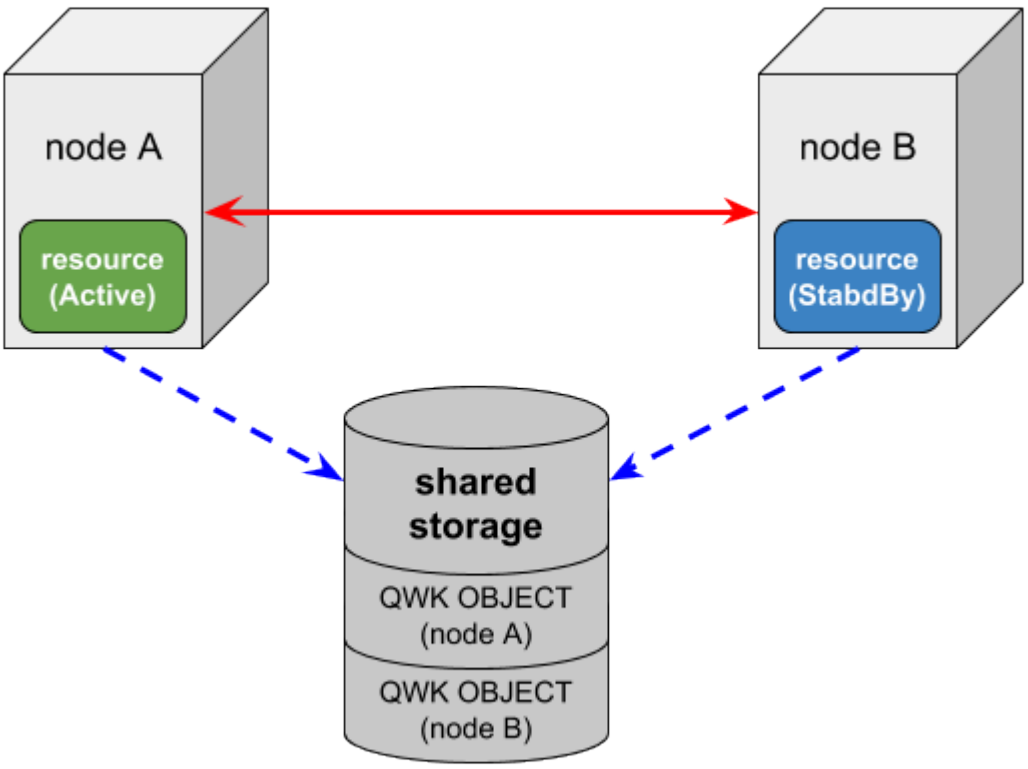
The frequent logging of message ID 135802 in the lifekeeper.log indicates that the periodic reading and writing of the QWK objects is overloaded and causing a delay in processing. If the number of nodes in the cluster is large and S3 is used for the shared storage (especially when used in two regions), the load from reading and writing of the QWK objects can be high.

Follow these steps to avoid the frequent logging of message ID 135802 in the lifekeeper.log.

- Increase the value of QWK_STORAGE_HBEATTIME (the time required for a failure to be detected and for a failover to begin will increase)
- If the shared storage choice is S3, use only in regions with the lowest network latency.
- Increase the throughput (for Amazon EC2 instances, change the instance type, etc.)

Expected Behaviors for Storage Mode (Assuming Default Modes)

Behavior of a two-node cluster; Node A (resources are in-service) and Node B (resources are on stand-by), is shown below.



The following three events may change the resource status on a node failure:

- COMM_DOWN event
  An event called when all the communication paths between the nodes are disconnected.

- COMM_UP event
  An event called when the communication paths are recovered from a COMM_DOWN state.

- LCM_AVAIL event
  An event called after LCM initialization is completed and it is called only once when starting LifeKeeper. Once this state has been reached heartbeat, transmission to other nodes in the cluster begins over the established communication paths. It is also ready to receive heartbeat requests from other nodes in the cluster. LCM_AVAIL will always processed before processing a COMM_UP event.

Scenario 1

The communication paths fail between Node A and Node B (Both Node A and Node B can access the shared storage)

In this case, the following will happen:

1. Both Node A and Node B will begin processing a COMM_DOWN event, though not necessarily at exactly the same time.

2. Both nodes will perform the quorum check and determine that they still have quorum (both A and B can access the shared storage).

3. Each node will check the QWK object for the node with whom it has lost communication to see if it is still being updated on a regular basis. Both nodes will find that the other's QWK object is being updated on a regular as both nodes are still running witness checks.

4. It will be determined, via the witness checking on each node, that the other is still alive so no failover processing will take place. Resources will be left in service at Node A.

Scenario 2

Node A fails and stops

In this case, Server B will do the following:

1. Begin processing a COMM_DOWN event from Node A.

2. Determine that it can still access the shared storage and thus has quorum.

3. Check to see that updates to the QWK object for Node A have stopped (witness checking).

4. Verify via witness checking that Node A really appears to be lost and begins the usual failover activity. Node B will continue processing and bring the protected resources in service.

With resources being in-service on Node B, Node A is powered on and establishes communications with the other nodes and is able to access the QWK shared storage

In this case, Node A will process a LCM_AVAIL event. Node A will determine that it has quorum and not bring resources in service because they are currently in service on Node B. Next, a COMM_UP event will be processed between Node A and Node B.

Each node will determine that it has quorum during the COMM_UP events and Node A will not bring resources in service because they are currently in service on Node B.

With resources being in-service on Node B, Node A is powered on and cannot establish communications to the other nodes but is able to access the QWK shared storage

In this case, Node A will process a LCM_AVAIL event. Node A will determine that it has quorum since it can access the shared storage for the QWK objects. It will then perform witness checks to determine the status for Node B since the communication to Node B is down. Since Node B is running and has been updating its QWK object, Node A detects this and does not bring resources in service. Node B will do nothing since it can't communicate with Node A and already has the resource in-service.

Scenario 3

A failure occurs with the network for Node A (Node A is running without communication paths to the other nodes and does not have access to the QWK objects on shared storage)

In this case, Node A will do the following:

1. Begin processing a COMM_DOWN event from Node B.
2. Determine that it cannot access the shared storage and thus does not have quorum.
3. **Immediately** force-quit ("*fastkill*", default behavior of QUORUM_LOSS_ACTION).

Also, in this case, Node B will do the following:

1. Begin processing a COMM_DOWN event from Node A.
2. Determine that it can still access the shared storage and thus has quorum.
3. Verify that the updating for the QWK objects for Node A has stopped (witness checking).
4. Verify via witness checking that Node A really appears to be lost and, begin the usual failover activity. Node B will now have the protected resources in service.

With resources being in-service on Node B, Node A is powered on and establishes communications with the other nodes and is able to access the QWK shared storage

Same as scenario 2.

With the resources being in-service on Node B, Node A powered-on but is not able to access the QWK shared storage

In this case, Node A will process an LCM_AVAIL event. Node A will determine that it does not have quorum and will not bring resources in service.

If the communication paths to Node B are available, then a COMM_UP event will be processed. However, because Node A does not have quorum, it will not bring resources in service.

5.3.2.10.4. STONITH

STONITH (Shoot The Other Node in the Head) is a fencing technique for remotely powering down a node in a cluster. LifeKeeper can provide STONITH capabilities by using external power switch controls, IPMI-enabled motherboard controls and hypervisor-provided power capabilities to power off the other nodes in a cluster.

Using IPMI with STONITH

IPMI (Intelligent Platform Management Interface) defines a set of common interfaces to a computer system which can be used to monitor system health and manage the system. Used with STONITH, it allows the cluster software to instruct the switch via a serial or network connection to power off or reboot a cluster node that appears to have died thus ensuring that the unhealthy node cannot access or corrupt any shared data.

Package Requirements

- IPMI tools package (e.g. ipmitool-1.8.11-6.el6.x86_64.rpm)

STONITH in VMware vSphere Environments

vCLI (vSphere Command-Line Interface) is a command-line interface supported by VMware for managing your virtual infrastructure including the ESXi hosts and virtual machines. You can choose the vCLI command best suited for your needs and apply it for your LifeKeeper STONITH usage between VMware virtual machines.

Package Requirements

STONITH Server

- ◦ VMware vSphere SDK Package or VMware vSphere CLI. (vSphere CLI is included in the same installation package as the vSphere SDK).

Monitored Virtual Machine

- ◦ VMware Tools

Installation and Configuration

After installing LifeKeeper and configuring communication paths for each node in the cluster, install and configure STONITH.

1. Install the LifeKeeper STONITH script by running the following command:

```
/opt/LifeKeeper/samples/STONITH/stonith-install
```

2. (*For IPMI usage only) Using BIOS or the ipmitool command, set the following BMC (Baseboard Management Controller) variables:

- Use Static IP
- IP address
- Sub netmask
- User name
- Password
- Add Administrator privilege level to the user
- Enable network access to the user

Example using `ipmitool` command

(For detailed information, see the ipmitool man page.)

```
# ipmitool lan set 1 ipsrc static
# ipmitool lan set 1 ipaddr 192.168.0.1
# ipmitool lan set 1 netmask 255.0.0.0
# ipmitool user set name 1 root
# ipmitool user set password 1 secret
# ipmitool user priv 1 4
# ipmitool user enable 1
```

3. Edit the configuration file.

Update the configuration file to enable STONITH and add the power off command line.

Note: Power off is recommended over reboot to avoid fence loops (i.e. two machines have lost communication but can still STONITH each other, taking turns powering each other off and rebooting).

```
/opt/LifeKeeper/config/stonith.conf
```

```
# LifeKeeper STONITH configuration
#
# Each system in the cluster is listed below. To enable STONITH for a
# given system,
# remove the '#' on that line and insert the STONITH command line to power off
# that system.
```

```
# Example1: ipmi command

# node-1 ipmitool -I lanplus -H 10.0.0.1 -U root -P secret power off

# Example2: vCLI-esxcli command

# node-2 esxcli —server=10.0.0.1 —username=root —password=secret vms vm kill —type='hard' —world-id=1234567

# Example3: vCLI-vmware_cmd command

# node-3 vmware-cmd -H 10.0.0.1 -U root -P secret <vm_id> stop hard

minute-maid ipmitool -I lanplus -H 192.168.0.1 -U root -P secret power off
kool-aid ipmitool -I lanplus -H 192.168.0.2 -U root -P secret power off

vm1 esxcli —server=10.0.0.1 —username=root —password=secret vms vm kill —type='hard' —world-id=1234567
vm2 vmware-cmd -H 10.0.0.1 -U root -P secret <vm_id> stop hard
```

<vm_id>

vSphere CLI commands run on top of vSphere SDK for Perl. is used as an identifier of the VM. This variable should point to the VM's configuration file for the VM being configured.

To find the configuration file path:

1. Type the following command:

   ```
   vmware-cmd -H <vmware host> -l
   ```

2. This will return a list of VMware hosts.

   Example output from `vmware-cmd -l` with three vms listed:

   ```
   /vmfs/volumes/4e08c1b9-d741c09c-1d3e-0019b9cb28be/lampserver/
   ```

```
lampserver.vmx

    /vmfs/volumes/4e1e1386-0b862fae-a859-0019b9cb28bc/oracle10/
oracle.vmx

    /vmfs/volumes/4e08c1b9-d741c09c-1d3e-0019b9cb28be/lampserver02/
lampserver02.vmx
```

Find the VM being configured in the resulting list.

3. Paste the path name into the variable. The example above would then become:

```
    vmware-cmd -H 10.0.0.1 -U root -P secret /vmfs/volumes/
4e08c1b9-d741c09c-1d3e-0019b9cb28be/lampserver/lampserver.vmx stop hard
```

> **Note:** For further information on VMware commands, use `vmware-cmd` with no arguments to display a help page about all options.

Expected Behaviors

When LifeKeeper detects a communication failure with a node, that node will be powered off and a failover will occur. Once the issue is repaired, the node will have to be manually powered on.

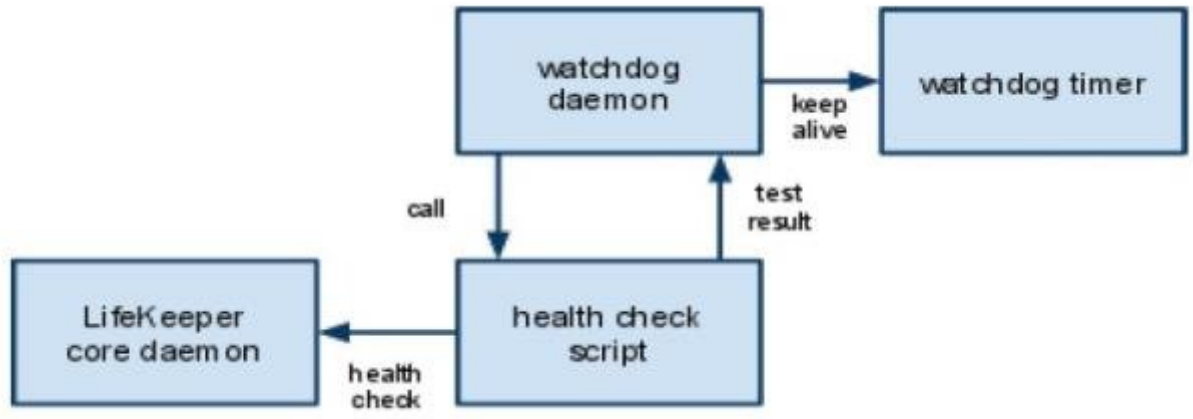
5.3.2.10.5. Watchdog

Watchdog is a method of monitoring a server to ensure that if the server is not working properly, corrective action (reboot) will be taken so that it does not cause problems. Watchdog can be implemented using special watchdog hardware or using a software-only option.

Note: This configuration has only been tested with Red Hat Enterprise Linux Versions 6 and 7. No other operating systems have been tested; therefore, no others are supported at this time.

Components

- Watchdog timer – software driver or an external hardware component
- Watchdog daemon – rpm available through the Linux distribution
- LifeKeeper core daemon – installed with the LifeKeeper installation
- Health check script – Script to check the status of LifeKeeper core



LifeKeeper Interoperability with Watchdog

Read the next section carefully. The daemon is designed to recover from errors and will reset the system if not configured carefully. Planning and care should be given to how this is installed and configured. This section is not intended to explain and configure watchdog, but only to explain and configure how LifeKeeper interoperates in such a configuration.

Configuration

The following steps should be carried out by an administrator with root user privileges. The administrator should already be familiar with some of the risks and issues with watchdog.

The health check script (LifeKeeper monitoring script) is the component that ties the LifeKeeper configuration with the watchdog configuration (`/opt/LifeKeeper/samples/watchdog/LifeKeeper-watchdog`). This script can monitor the basic parts of LifeKeeper core components.

1. If watchdog has been previously configured, enter the following command to stop it. If not, go to Step 2.

   `service watchdog stop` (RHEL6)

   `systemctl stop watchdog` (RHEL7)

2. Edit the watchdog configuration file (*`/etc/watchdog.conf`*) supplied during the installation of watchdog software.

   - ◦ Modify test-binary:

     ```
     test-binary = /opt/LifeKeeper/samples/watchdog/LifeKeeper-watchdog
     ```

   - ◦ Modify test-timeout:

     ```
     test-timeout = 5
     ```

   - ◦ Modify interval:

     ```
     interval = 7
     ```

   The interval value should be less than LifeKeeper communication path timeout (15 seconds), so a good number for the interval is generally half of this value.

3. Make sure LifeKeeper has been started. If not, please refer to the Starting LifeKeeper topic.

4. Start watchdog by entering the following command:

   ```
   service watchdog start (RHEL6)

   systemctl start watchdog (RHEL7)
   ```

5. To start watchdog automatically on future restarts, enter the following command:

   ```
   chkconfig –levels 35 watchdog on (RHEL6)

   systemctl enable watchdog (RHEL7)
   ```

Note: Configuring watchdog may cause some unexpected reboots from time to time. This is the general nature of how watchdog works. If processes are not responding

correctly, the watchdog feature will assume that LifeKeeper (or the operating system) is hung, and it will reboot the system (without warning).

Uninstall

Care should be taken when uninstalling LifeKeeper. The above steps should be done in reverse order as listed below.

WARNING: IF UNINSTALLING LIFEKEEPER BY REMOVING THE RPM PACKAGES THAT MAKE UP LIFEKEEPER, **TURN OFF WATCHDOG FIRST!** In Step 2 above, the watchdog config file was modified to call on the LifeKeeper-watchdog script; therefore, if watchdog is not turned off first, it will call on that script that is no longer there. An error will occur when this script is not found which will trigger a reboot. This will continue until watchdog is turned off.

1. Stop watchdog by entering the following command:

   ```
   service watchdog stop (RHEL6)
   ```

   ```
   systemctl stop watchdog (RHEL7)
   ```

2. Edit the watchdog configuration file (`/etc/watchdog.conf`) supplied during the installation of watchdog software.

   - Modify test-binary and interval by commenting out those entries (add # at the beginning of each line):

     ```
     #test-binary =
     ```

     ```
     #interval =
     ```

   Note: If interval was used previously for other functions, it can be left as-is

3. Uninstall LifeKeeper. See the Removing LifeKeeper topic.

4. Watchdog can now be started again. If only used by LifeKeeper, watchdog can be permanently disabled by entering the following command:

   ```
   chkconfig —levels 35 watchdog off (RHEL6)
   ```

   ```
   systemctl disable watchdog (RHEL7)
   ```

5.3.2.10.6. I/O Fencing Mechanisms

LifeKeeper for Linux provides various fencing mechanisms. Depending on the server and storage configuration, available fencing mechanisms and allowed combinations of these may differ.

Refer to the information linked below for fencing mechanisms available in each server configuration. Details of storage configuration are described on the server configuration pages.

- For physical servers see Available I/O Fencing Mechanisms (Physical Servers)
- For virtual machines in VMware see Available I/O Fencing Mechanisms (Virtual Machines in VMware)

I/O Fencing Mechanism Summary

- SCSI Fencing with SCSI-2 Reservations – By issuing a SCSI-2 reservation to the storage from the active node, the protected logical unit (LU) is locked, preventing simultaneous access from other nodes. When a communication failure is detected, a standby node will disable LU locks from other nodes, and then lock the protected LU, preventing simultaneous access from other nodes.
- SCSI Fencing with SCSI-3 Reservations – By issuing a SCSI-3 reservation to the storage from the active node, the protected logical unit (LU) is locked, preventing simultaneous access from other nodes. When a communication failure is detected, a standby node will disable LU locks from other nodes, and then lock the protected LU, preventing simultaneous access from other nodes.
- IPMI STONITH – When a communication failure is detected, the nodes will issue IPMI “power off” commands to the other nodes, preventing the protected service from starting on multiple nodes. This mechanism is available only for physical servers with IPMI interfaces.
- VMware STONITH – When a communication failure is detected, the nodes will issue “power off” commands to the other nodes via the VMware host or vCenter APIs, preventing the protected service from starting on multiple nodes. This mechanism is available only for virtual machines running in VMware environments.
- Quorum/Witness – When a communication failure is detected, an arbitrator will determine whether failover to a given node would effectively achieve service continuation. LifeKeeper normally prevents simultaneous access to LUs from multiple nodes with the reservation commands described above, but Quorom/Witness is mandatory in environments where reservations are not used. There are three modes of arbitration: dedicated Witness node, independent host, or shared storage (see Quorum/Witness for details).

5.3.2.10.6.1. Available I/O Fencing Mechanisms (Physical Servers)

This page describes the combinations of fencing mechanisms that can be used with various storage configurations in a physical server environment.

Shared Disk Configuration (Single or Multipath Configuration Using SCSI Reservations)

This configuration corresponds to the case where exclusive control is enforced by SCSI reservations in a certified shared storage system, as listed in the Supported Storage List. When using storage that requires a multipath driver for SCSI-3 reservations and multipath control, a multipath kit that supports the multipath driver is required.

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used.

Symbol Definitions

● – Required
○ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations					
SCSI-2 Reservations	●	●	●	?[1]	?[1]	?[1]
SCSI-3 Reservations	?[1]	?[1]	?[1]	●	●	●
IPMI STONITH	?[2]	?[2]	?[2]	?[2]	?[2]	?[2]
Quorum/Witness (tcp_remote)	○	?[3]	?[3]	○	?[3]	?[3]
Quorum/Witness (majority)	?[3]	○	?[3]	?[3]	○	?[3]
Quorum/Witness (storage)	?[3]	?[3]	○	?[3]	?[3]	○

[1]SCSI-2 and SCSI-3 reservations cannot coexist on a single shared disk.

[2] When SCSI Reservations and STONITH are used together, the functions of each may conflict and an unexpected system outage may occur.

[3] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- Fencing mechanisms for virtual and cloud environments cannot be used with this configuration.

Shared Disk Configuration (Single or Multipath Configuration Not Using SCSI Reservations)

This configuration is applicable when SCSI reservations cannot be used with a shared storage system connected with a method including SCSI/FC/iSCSI/SAS (but excluding NAS). When using storage that requires a multipath driver to control multiple paths, a multipath kit that supports the multipath driver is required.

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used.

Symbol Definitions

● – Required
○ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations		
IPMI STONITH	○	○	○
Quorum/Witness (tcp_remote)	●	?[1]	?[1]
Quorum/Witness (majority)	?[1]	●	?[1]
Quorum/Witness (storage)	?[1]	?[1]	●

[1] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- SCSI Fencing cannot be used with storage systems that are not certified to properly support SCSI reservations.
- Fencing mechanisms for virtual and cloud environments cannot be used with this configuration.

Data Replication Configuration

This configuration is applicable when local storage connected to each node is replicated between nodes using DataKeeper.

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used. Fencing is not mandatory in this configuration; however, it is recommended to use fencing to enhance data protection.

Symbol Definitions

● – Required
◯ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations		
IPMI STONITH	◯	◯	◯
Quorum/Witness (tcp_remote)	◯	?[1]	?[1]
Quorum/Witness (majority)	?[1]	◯	?[1]
Quorum/Witness (storage)	?[1]	?[1]	◯

[1] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- SCSI Fencing cannot be used with this configuration.
- Fencing mechanisms for virtual and cloud environments cannot be used with this configuration.

Network Attached Storage (NAS) Configuration

This configuration is applicable when using NAS storage connected using Network File System (NFS).

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used. Fencing is not mandatory in this configuration; however, it is recommended to use fencing to enhance data protection.

Symbol Definitions

● – Required
◯ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations			
IPMI STONITH	◯	◯	◯	
Quorum/Witness (tcp_remote)	◯	?[1]	?[1]	
Quorum/Witness (majority)	?[1]	◯	?[1]	
Quorum/Witness (storage)	?[1]	?[1]	◯	

[1] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- SCSI Fencing cannot be used with this configuration.
- Fencing mechanisms for virtual and cloud environments cannot be used with this configuration.

5.3.2.10.6.2. Available I/O Fencing Mechanisms (Virtual Machines in VMware)

This page describes the combinations of fencing mechanisms that can be used with various storage configurations in a VMware virtual server environment.

Shared Disk Configuration (Single or Multipath Configuration Using SCSI Reservations)

This configuration corresponds to the case where exclusive control is enforced by SCSI reservations in a certified shared storage system, as listed in the Supported Storage List. When using storage that requires a multipath driver for SCSI-3 reservations and multipath control, a multipath kit that supports the multipath driver is required.

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used.

Symbol Definitions

● – Required
○ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations					
SCSI-2 Reservations	●	●	●	?[1]	?[1]	?[1]
SCSI-3 Reservations	?[1]	?[1]	?[1]	●	●	●
VMware STONITH	?[2]	?[2]	?[2]	?[2]	?[2]	?[2]
Quorum/Witness (tcp_remote)	○	?[3]	?[3]	○	?[3]	?[3]
Quorum/Witness (majority)	?[3]	○	?[3]	?[3]	○	?[3]
Quorum/Witness (storage)	?[3]	?[3]	○	?[3]	?[3]	○

[1]SCSI-2 and SCSI-3 reservations cannot coexist on a single shared disk.

[2] When SCSI Reservations and STONITH are used together, the functions of each may conflict and an unexpected system outage may occur.

[3] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- Fencing mechanisms for virtual environments other than VMware cannot be used with this configuration.
- Fencing mechanisms for cloud and physical environments cannot be used with this configuration.

Shared Disk Configuration (Single or Multipath Configuration Not Using SCSI Reservations)

This configuration is applicable when SCSI reservations cannot be used with a shared storage system connected with a method including SCSI/FC/iSCSI/SAS (but excluding NAS). When using storage that requires a multipath driver to control multiple paths, a multipath kit that supports the multipath driver is required.

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used.

Symbol Definitions

● – Required
○ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations		
VMware STONITH	○	○	○
Quorum/Witness (tcp_remote)	●	?[1]	?[1]
Quorum/Witness (majority)	?[1]	●	?[1]
Quorum/Witness (storage)	?[1]	?[1]	●

[1] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- SCSI Fencing cannot be used with storage systems that are not certified to properly support SCSI reservations.
- Fencing mechanisms for virtual environments other than VMware cannot be used with this

configuration.
- Fencing mechanisms for cloud and physical environments cannot be used with this configuration.

Data Replication Configuration

This configuration is applicable when local storage connected to each node is replicated between nodes using DataKeeper.

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used. Fencing is not mandatory in this configuration; however, it is recommended to use fencing to enhance data protection.

Symbol Definitions

● – Required
◯ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations			
VMware STONITH	◯	◯	◯	
Quorum/Witness (tcp_remote)	◯	?[1]	?[1]	
Quorum/Witness (majority)	?[1]	◯	?[1]	
Quorum/Witness (storage)	?[1]	?[1]	◯	

[1] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- SCSI Fencing cannot be used with this configuration.
- Fencing mechanisms for virtual environments other than VMware cannot be used with this configuration.
- Fencing mechanisms for cloud and physical environments cannot be used with this configuration.

Network Attached Storage (NAS) Configuration

This configuration is applicable when using NAS storage connected using Network File System (NFS).

Available Fencing Mechanisms

The following table shows which fencing mechanisms are available and the allowed combinations, if multiple fencing mechanisms are used. Fencing is not mandatory in this configuration; however, it is recommended to use fencing to enhance data protection.

Symbol Definitions

● – Required
◯ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Allowed Combinations			
VMware STONITH	◯	◯	◯	
Quorum/Witness (tcp_remote)	◯	?[1]	?[1]	
Quorum/Witness (majority)	?[1]	◯	?[1]	
Quorum/Witness (storage)	?[1]	?[1]	◯	

[1] Multiple Quorum/Witness modes cannot be used together in a single cluster.

Mechanisms Not Available with this Configuration

- SCSI Fencing cannot be used with this configuration.
- Fencing mechanisms for virtual environments other than VMware cannot be used with this configuration.
- Fencing mechanisms for cloud and physical environments cannot be used with this configuration.

VMDK as Shared Storage Configuration

This configuration is applicable when using a VMware virtual hard disk configured with the VMDK as the Shared Storage method.

Available Functions and Appropriate Combinations

The table below shows the fencing functions available with this configuration and the appropriate combination patterns.

Symbol Definitions

● – Required

◯ – Available as an option
? – Not available

For the functions listed as “Available as an option” in the table, use of the functions in the combination is not mandatory. Please refer to the note below the table for the functions that are “Not available” in the table.

	Combination Pattern		
	A	**B**	**C**
VMware STONITH	?[1]	?[1]	?[1]
Quorum/Witness (tcp_remote)	◯	?[2]	?[2]
Quorum/Witness (majority)	?[2]	◯	?[2]
Quorum/Witness (storage)	?[2]	?[2]	◯

[1]VMDK as Shared Storage and VMware STONITH cannot coexist because service may stop due to a conflict.

[2] Due to the LifeKeeper mechanism, Quorum Witness modes cannot coexist for a single cluster.

Unavailable Function with this Configuration

- The fencing function that requires the use of a shared disk cannot be used with this configuration.

5.3.2.11. Resource Policy Management

Resource Policy Management in SIOS Protection Suite for Linux provides behavior management of resource local recovery and failover. Resource policies are managed with the **lkpolicy** command line tool (CLI).

SIOS Protection Suite

SIOS Protection Suite is designed to monitor individual applications and groups of related applications, periodically performing local recoveries or notifications when protected applications fail. Related applications, by example, are hierarchies where the primary application depends on lower-level storage or network resources. When an application or resource failure occurs, the default behavior is:

1. **Local Recovery:** First, attempt **local** recovery of the resource or application. An attempt will be made to restore the resource or application on the local server without external intervention. If local recovery is successful, then SIOS Protection Suite will not perform any additional action.

2. **Failover:** Second, if a local recovery attempt *fails* to restore the resource or application (or the *recovery kit* monitoring the resource has no support for local recovery), then a ***failover*** will be initiated. The failover action attempts to bring the application (and all dependent resources) into service on another server within the cluster.

Please see SIOS Protection Suite Fault Detection and Recovery Scenarios for more detailed information about our recovery behavior.

Custom and Maintenance-Mode Behavior via Policies

SIOS Protection Suite Version 7.5 and later supports the ability to set additional policies that modify the default recovery behavior. There are four policies that can be set for individual resources (see the section below about precautions regarding individual resource policies) or for an entire server. ***The recommended approach is to alter policies at the server level.***

The available policies are:

Standard Policies

- **Failover** This policy setting can be used to turn on/off resource failover. (**Note:** In order for reservations to be handled correctly, **Failover** cannot be turned off for individual scsi resources.)

- **LocalRecovery** – SIOS Protection Suite, by default, will attempt to recover protected resources by restarting the individual resource or the entire protected application prior to performing a failover. This policy setting can be used to turn on/off local recovery.

- **TemporalRecovery** – Normally, SIOS Protection Suite will perform local recovery of a failed resource. If local recovery fails, SIOS Protection Suite will perform a resource hierarchy failover to

another node. If the local recovery succeeds, failover will not be performed.

There may be cases where the local recovery succeeds, but due to some irregularity in the server, the local recovery is re-attempted within a short time; resulting in multiple, consecutive local recovery attempts. This may degrade availability for the affected application.

To prevent this repetitive local recovery/failure cycle, you may set a temporal recovery policy. The temporal recovery policy allows an administrator to limit the number of local recovery attempts (successful or not) within a defined time period.

> *Example:* If a user sets the policy definition to limit the resource to three local recovery attempts in a 30-minute time period, SIOS Protection Suite will fail over when a third local recovery attempt occurs within the 30-minute period.

Defined temporal recovery policies may be turned *on* or *off*. When a temporal recovery policy is *off*, temporal recovery processing will continue to be done and notifications will appear in the log when the policy *would* have fired; however, no actions will be taken.

> **Note:** It is possible to disable failover and/or local recovery with a temporal recovery policy also in place. This state is illogical as the temporal recovery policy will **never** be acted upon if failover or local recovery are disabled.

Meta Policies

The “meta” policies are the ones that can affect more than one other policy at the same time. These policies are usually used as shortcuts for getting certain system behaviors that would otherwise require setting multiple standard policies.

- **NotificationOnly** – This mode allows administrators to put SIOS Protection Suite in a “monitoring only” state. ***Both*** local recovery ***and*** failover ***of a resource (or all resources in the case of a server-wide policy) are affected***. The user interface will indicate a **Failure** state if a failure is detected; *but no recovery or failover action will be taken*. **Note:** The administrator will need to correct the problem that caused the failure manually and then bring the affected resource(s) back in service to continue normal SIOS Protection Suite operations.

Important Considerations for Resource-Level Policies

Resource level policies are policies that apply to a specific resource only, as opposed to an entire resource hierarchy or server.

Example:

app

- IP

- file system

In the above resource hierarchy, app depends on both IP and file system. A policy can be set to *disable* local recovery or failover of a specific resource. This means that, for example, if the IP resource's local recovery fails and a policy was set to disable failover of the IP resource, then the IP resource will not fail over or cause a failover of the other resources. However, if the file system resource's local recovery fails and the file system resource policy does not have failover disabled, then the entire hierarchy will fail over.

Note: It is important to remember that resource level policies apply *only* to the specific resource for which they are set.

This is a simple example. Complex hierarchies can be configured, so care must be taken when setting resource-level policies.

The lkpolicy Tool

The lkpolicy tool is the command-line tool that allows management (querying, setting, removing) of policies on servers running SIOS Protection Suite for Linux. `lkpolicy` supports setting/modifying policies, removing policies and viewing all available policies and their current settings. In addition, defined policies can be set on or off, preserving resource/server settings while affecting recovery behavior.

The general usage is :

```
lkpolicy [—list-policies | —get-policies | —set-policy | —remove-policy] <name value pair data…>
```

The `<name value pair data…>` differ depending on the operation and the policy being manipulated, particularly when setting policies. *For example:* Most on/off type policies only require `—on` or `—off` switch, but the temporal policy requires additional values to describe the threshold values.

Example lkpolicy Usage

Authenticating With Local and Remote Servers

The lkpolicy tool communicates with SIOS Protection Suite servers via an API that the servers expose. This API requires authentication from clients like the `lkpolicy` tool. The first time the `lkpolicy` tool is asked to access a SIOS Protection Suite server, if the credentials for that server are not known, it will ask the user for credentials for that server. These credentials are in the form of a username and password and:

1. Clients must have SIOS Protection Suite admin rights. This means the username must be in the *lkadmin group* according to the operating system's authentication configuration (via *pam*). It is ***not***

necessary to run as ***root***, but the root user can be used since it is in the appropriate group by default.

2. The credentials will be stored in the *credential store* so they do not have to be entered manually each time the tool is used to access this server.

See Configuring Credentials for SIOS Protection Suite for more information on the credential store and its management with the `credstore` utility.

An example session with `lkpolicy` might look like this:

```
[root@thor49 ~]# lkpolicy -l -d v6test4
Please enter your credentials for the system 'v6test4'.
Username: root
Password:
Confirm password:
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]# lkpolicy -l -d v6test4
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]#
```

Listing Policies

lkpolicy —list-policy-types

Showing Current Policies

lkpolicy —get-policies

lkpolicy —get-policies tag=*

lkpolicy —get-policies —verbose tag=mysql* # all resources starting with mysql

lkpolicy —get-policies tag=mytagonly

Setting Policies

lkpolicy —set-policy Failover —off

lkpolicy —set-policy Failover —on tag=myresource

lkpolicy —set-policy Failover —on tag=*

lkpolicy —set-policy LocalRecovery —off tag=myresource

lkpolicy —set-policy NotificationOnly —on

lkpolicy —set-policy TemporalRecovery —on recoverylimit=5 period=15

lkpolicy —set-policy TemporalRecovery —on —force recoverylimit=5 period=10

Removing Policies

lkpolicy —remove-policy Failover tag=steve

> ***Note:*** *NotificationOnly is a policy alias. Enabling NotificationOnly is the equivalent of disabling the corresponding LocalRecovery and Failover policies.*

5.3.2.12. Configuring Credentials

Credentials for communicating with other systems are managed via a *credential store*. This store can be managed, as needed, by the `/opt/LifeKeeper/bin/credstore` utility. This utility allows server access credentials to be set, changed and removed – on a per server basis.

Adding or Changing Credentials

Adding and changing credentials are handled in the same way. A typical example of adding or changing credentials for a server, `server.mydomain.com`, would look like this:

```
/opt/LifeKeeper/bin/credstore -k server.mydomain.com myuser
```

In this case, *myuser* is the username used to access `server.mydomain.com` and the password will be asked for via a prompt with confirmation (like *passwd*).

> **Note:** The key name used to store LifeKeeper server credentials must match *exactly* the hostname used in commands such as `lkpolicy`. If the hostname used in the command is an FQDN, then the credential key must also be the FQDN. If the hostname is a short name, then the key must also be the short name.

You may wish to set up a ***default*** key in the credential store. The ***default*** credentials will be used for authentication when no specific server key exists. To add or change the ***default*** key, run:

```
/opt/LifeKeeper/bin/credstore -k default myuser
```

Listing Stored Credentials

The currently stored credentials can be listed by the following command:

```
/opt/LifeKeeper/bin/credstore -l
```

This will list the *keys* stored in the credential store and, in this case, the key indicates the server for which the credentials are used. (This command will not actually list the credentials, only the key names, since the credentials themselves may be sensitive.)

Removing Credentials for a Server

Credentials for a given server can be removed with the following command:

```
/opt/LifeKeeper/bin/credstore -d -k myserver.mydomain.com
```

In this case, the credentials for the server `myserver.mydomain.com` will be removed from the store.

Additional Information

More information on the credstore utility can be found by running:

```
/opt/LifeKeeper/bin/credstore —man
```

This will show the entire man/help page for the command.

5.3.2.13. Standby Node Health Check

Overview

The Standby Node Health Check feature allows you to monitor CPU and memory utilization on the standby node and monitor the health of out-of-service resources to detect errors on the standby node. This allows for issues to be resolved in advance, reducing the risk of an unsuccessful failover, if a failure occurs on the active node. This monitoring is performed at the same interval as the normal LifeKeeper resource monitoring (*/etc/default/LifeKeeper* setting LKCHECKINTERVAL).

The Standby Node Health Check performs the following two functions:

Node Monitoring

If all resources on a node are out of service, LifeKeeper considers it a standby node and calls the node monitoring script. The node monitoring script monitors CPU and memory utilization. If it determines that the node cannot be switched to successfully (due to high CPU or memory load), it sends this information to the administrator by email or SNMP event forwarding. See Node Monitoring for details.

Out-of-Service (OSU) Resource Monitoring

For each out-of-service (OSU) resource, *lkcheck* periodically calls the *OSUquickCheck* script. The *OSUquickCheck* script performs a quick health check for the resource. If it determines that the resource cannot start successfully, it changes the state of the resource to OSF and sends this information to the administrator by email or SNMP event forwarding. See OSU Resource Monitoring for details.

Installation and Configuration

There is no special installation required.

Setting up Standby Node Health Check

1. Configure email notification and event forwarding via SNMP2.

2. Configure Standby Node Health Check (Set the SNHC settings in the */etc/default/LifeKeeper* configuration file. See Standby Node Health Check Parameters List for details.)

3. If LifeKeeper is already started, restart the lkcheck process in order to reflect the configuration. Run the following command to restart the lkcheck process:

```
killall lkcheck
```

Once the above steps are completed, the Standby Node Health Check is activated on that node.

5.3.2.13.1. Node Monitoring

If all resources on a node are out of service, LifeKeeper considers it a standby node and calls the node monitoring script. The node monitoring script monitors CPU and memory utilization. If it determines that the node cannot be switched to successfully (due to high CPU or memory load), it sends this information to the administrator by email or SNMP event forwarding. This monitoring is performed at the same interval as the normal LifeKeeper resource monitoring (*/etc/default/LifeKeeper* setting LKCHECKINTERVAL).

Monitored Resources

The following can be monitored with Node Monitoring:

Resource Name	Monitoring Details
CPU Utilization	Check CPU Utilization in */proc/stat* file
Memory Utilization	Check Memory Utilization in */proc/meminfo* file

Node Monitoring Configuration

Set the SNHC_CPUCHECK and SNHC_MEMCHECK settings in the */etc/default/LifeKeeper* configuration file. You will also need to configure the following settings. See Standby Node Health Check Parameters List for details.

- SNHC_CPUCHECK_THRESHOLD
- SNHC_CPUCHECK_TIME
- SNHC_MEMCHECK_THRESHOLD
- SNHC_MEMCHECK_TIME

5.3.2.13.2. OSU Resource Monitoring

For each out-of-service (OSU) resource, *lkcheck* periodically calls the *OSUquickCheck* script for the resource. The *OSUquickCheck* script performs a quick health check for the resource. If it determines that the resource cannot start successfully, it changes the state of the resource to OSF and sends this information to the administrator by email or SNMP event forwarding. This monitoring is performed at the same interval as the normal LifeKeeper resource monitoring (*/etc/default/LifeKeeper* setting LKCHECKINTERVAL).

Monitored Resources

The following can be monitored with OSU Resource Monitoring:

Resource Name	Monitoring Details
IP Resource	Verify the NIC link is up (disable with */etc/default/LifeKeeper* setting IP_NOLINKCHECK=1). Also, verify network reachability (if a ping list is configured).
DMMP Disk Resource	Verify that the paths to the monitored disk are functional.

OSU Resource Monitoring Configuration

Set the SNHC_IPCHECK and SNHC_DISKCHECK settings in the */etc/default/LifeKeeper* configuration file. You may also need to configure the following setting. See Standby Node Health Check Parameters List for details.

- SNHC_IPCHECK_SLEEPTIME

Recovery from Failure

If an error is detected during OSU resource monitoring, the state of the corresponding resource is changed to OSF (out of service with failure). When the status is changed, OSU resource monitoring is no longer performed for the resource. After checking the details of the notified failure and addressing it, you should change the resource state to OSU. The state can be changed from OSF to OSU using the following command:

```
/opt/LifeKeeper/lkadm/bin/retstate <resource tag>
```

5.3.3. LifeKeeper Administration Overview

LifeKeeper does not require administration during operation. LifeKeeper works automatically to monitor protected resources and to perform the specified recovery actions if a fault should occur. You use the LifeKeeper GUI in these cases:

- **Resource and hierarchy definition.** LifeKeeper provides these interface options:
    - º LifeKeeper GUI
    - º LifeKeeper command line interface
- **Resource monitoring**. The LifeKeeper GUI provides access to resource status information and to the LifeKeeper logs.
- **Manual intervention**. You may need to stop servers or specific resources for maintenance or other administrative actions. The LifeKeeper GUI provides menu functions that allow you to bring specific resources in and out of service. Once applications have been placed under LifeKeeper protection, they should be started and stopped only through these LifeKeeper interfaces. Starting and stopping LifeKeeper is done through the command line only.

See GUI Tasks and Maintenance Tasks for detailed instructions on performing LifeKeeper administration, configuration and maintenance operations.

> **Note:** All LifeKeeper executable scripts and programs run via the command line require super user authority.

A super user granted permissions by running the “su” or “sudo” command is able to execute LifeKeeper commands. However, SIOS Technology Corp. has tested executing LifeKeeper commands via the root user only.

5.3.3.1. Error Detection and Notification

The ability to provide detection and alarming for problems within an application is critical to building the best total fault resilient solution. Since every specific application varies on the mechanism and format of failures, no one set of generic mechanisms can be supplied. In general, however, many application configurations can rely on the Core system error detection provided within LifeKeeper. Two common fault situations are used to demonstrate the power of LifeKeeper's core facilities in the topics Resource Error Recovery Scenario and Server Failure Recovery Scenario.

LifeKeeper also provides a complete environment for defining errors, alarms, and events that can trigger recovery procedures. This interfacing usually requires pattern match definitions for the system error log (`/var/log/messages`), or custom-built application specific monitor processes.

5.3.3.2. N-Way Recovery

N-Way recovery allows different resources to fail over to different backup servers in a cluster.

Return to Protected Resources

5.3.3.3. Administrator Tasks

5.3.3.3.1. Editing Server Properties

1. To edit the properties of a server, bring up the Server Properties dialog just as you would for viewing server properties.

2. If you are logged into that server with the appropriate permissions, the following items will be editable.

- º

Shutdown Strategy
- º

Failover Confirmation

3. Once you have made changes, the **Apply** button will be enabled. Clicking this button will apply your changes without closing the window.

4. When you are finished, click **OK** to save any changes and close the window, or **Cancel** to close the window without applying changes.

5.3.3.3.2. Creating a Communication Path

Before configuring a LifeKeeper communication path between servers, verify the hardware and software setup. For more information, see the SPS for Linux Release Notes.

To create a communication path between a pair of servers, you must define the path individually on both servers. LifeKeeper allows you to create both TCP (TCP/IP) and TTY communication paths between a pair of servers. Only one TTY path can be created between a given pair. However, you can create multiple TCP communication paths between a pair of servers by specifying the local and remote addresses that are to be the end-points of the path. A priority value is used to tell LifeKeeper the order in which TCP paths to a given remote server should be used.

> **IMPORTANT:** Using a single communication path can potentially compromise the ability of servers in a cluster to communicate with one another. If a single comm path is used and the comm path fails, LifeKeeper hierarchies may come in service on multiple servers simultaneously. This is known as “false failover”. Additionally, heavy network traffic on a TCP comm path can result in unexpected behavior, including false failovers and LifeKeeper initialization problems.

1. There are four ways to begin.

- º Right-click on a server icon, then click

Create Comm Path when the server context menu appears.
- º On the

global toolbar, click the **Create Comm Path** button.
- º On the

server context toolbar, if displayed, click the **Create Comm Path** button.
- º On the

Edit menu, select **Server**, then **Create Comm Path**.

2. A dialog entitled **Create Comm Path** will appear. For each of the options that follow, click **Help** for an explanation of each choice.

3. Select the **Local Server** from the list box and click **Next**.

4. Select one or more **Remote Servers** in the list box. If a remote server is not listed in the list box (i.e. it is not yet connected to the cluster), you may enter it using **Add**. You must make sure that the network addresses for both the local and remote servers are resolvable (for example, with DNS or added to the */etc/hosts* file). Click **Next**.

5. Select either **TCP** or **TTY** for **Device Type** and click **Next**.

6. Select one or more **Local IP Addresses** if the **Device Type** was set for **TCP**. Select the **Local**

TTY Device if the **Device Type** was set to **TTY**. Click **Next**.

7. Select the **Remote IP Address** if the **Device Type** was set for **TCP**. Select the **Remote TTY Device** if the **Device Type** was set to **TTY**. Click **Next**.

8. Enter or select the **Priority** for this comm path if the **Device Type** was set for **TCP**. Enter or select the **Baud Rate** for this Comm Path if the **Device Type** was set to **TTY**. Click **Next**.

9. Click **Create**. A message should be displayed indicating the network connection is successfully created. Click **Next**.

10. If you selected multiple Local IP Addresses or multiple Remote Servers and the **Device Type** was set for **TCP**, then you will be taken back to Step 6 to continue with the next Comm Path. If you selected multiple Remote Servers and the **Device Type** was set for **TTY**, then you will be taken back to Step 5 to continue with the next Comm Path.

11. Click **Done** when presented with the concluding message.

You can verify the comm path by viewing the Server Properties Dialog or by entering the command **`lcdstatus -q`**. See the LCD man page for information on using **lcdstatus**. You should see an **ALIVE** status.

In addition, check the server icon in the right pane of the GUI. If this is the first comm path that has been created, the server icon shows a yellow heartbeat, indicating that one comm path is **ALIVE**, but there is no redundant comm path. 

The server icon will display a green heartbeat when there are at least two comm paths **ALIVE**. 

IMPORTANT: When using IPv6 addresses to create a comm path, statically assigned addresses should be used instead of auto-configured/stateless addresses as the latter may change over time which will cause the comm path to fail.

If the comm path does not activate after a few minutes, verify that the paired server's computer name is correct. If using TTY comm paths, verify that the cable connection between the two servers is correct and is not loose. Use the `portio(1M)` command if necessary to verify the operation of the TTY connection.

5.3.3.3.3. Deleting a Communication Path

1. There are four ways to begin.

- º Right-click on a server icon, then click

Delete Comm Path when the server context menu appears.
- º On the

global toolbar, click the **Delete Comm Path** button.
- º On the

server context toolbar, if displayed, click the **Delete Comm Path** button.
- º On the

Edit menu, select **Server**, then **Delete Comm Path**.

2. A dialog entitled Delete Comm Path will appear. For each of the options that follow, click **Help** for an explanation of each choice.

3. Select **Local Server** from the list and click **Next**. This dialog will only appear if the delete is selected using the **Delete Comm Path** button on the global toolbar or via the Edit menu selecting **Server**.

4. Select the communications path(s) that you want to delete and click **Next**.

5. Click **Delete Comm Path(s)**. If the output panel is enabled, the dialog closes, and the results of the commands to delete the communications path(s) are shown in the output panel. If not, the dialog remains up to show these results, and you click **Done** to finish when all results have been displayed. A message should be displayed indicating the network connection is successfully removed

5. Click **Done** to close the dialog and return to the GUI status display.

5.3.3.3.4. Server Properties – Failover

In the event that the primary server has attempted and failed local recovery, or failed completely, most server administrators will want LifeKeeper to automatically restore the protected resource(s) to a backup server. This is the default LifeKeeper behavior. However, some administrators may not want the protected resource(s) to automatically go in-service at a recovery site. For example, if LifeKeeper is installed in a WAN environment where the network connection between the servers may not be reliable in a disaster recovery situation.

Automatic failover is enabled by default for all protected resources. To disable automatic failover for protected resources or to prevent automatic failover to a backup server, use the **Failover** section located on the **General** tab of Server Properties to configure as follows:

For each server in the cluster:

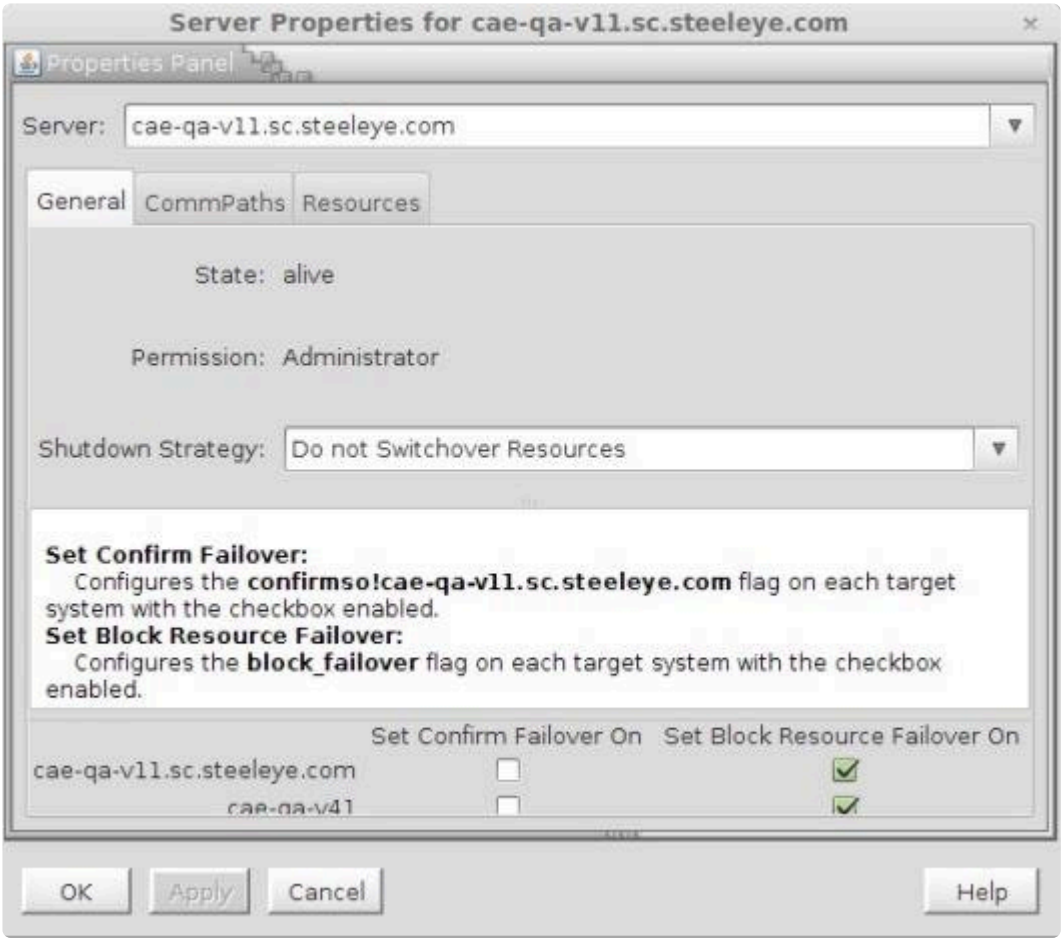
1. Bring up the **Server Properties** dialog just as you would for viewing server properties.

2. Select the **General** tab. In the **Failover** section of the Server Properties dialog, check the server to disable system and resource failover capabilities. By default, all failover capabilities of LifeKeeper are enabled.

   In the **Set Confirm Failover On** column, select the server to be disqualified as a backup server for a complete failure of the local server.

   In the **Set Block Resource Failover On** column, select the server to be disqualified as a backup server for any failed resource hierarchy on this local server. Resource failovers cannot be disabled without first disabling system failover capabilities.

   To commit your selections, press the **Apply** button.



Refer to **[Confirm Failover]** and **[Block Resource Failover]** Settings for configuration details.

5.3.3.3.5. Creating Resource Hierarchies

1. There are four ways to begin creating a resource hierarchy.

- º Right-click on a

server icon to bring up the server context menu, then click on **Create Resource Hierarchy**.
- º On the

global toolbar, click on the **Create Resource Hierarchy** button.
- º On the

server context toolbar, if displayed, click on the **Create Resource Hierarchy** button.
- º On the

Edit menu, select **Server**, then click on **Create Resource Hierarchy**.
2. A dialog entitled Create Resource Wizard will appear with a list of all recognized recovery kits installed within the cluster. Select the **Recovery Kit** that builds resource hierarchies to protect your application and click **Next**.

3. Select the **Switchback Type** and click **Next**.

4. Select the **Server** and click **Next**. **Note:** If you began from the server context menu, the server will be determined automatically from the server icon that you clicked on, and this step will be skipped.

5. Continue through the succeeding dialogs, entering whatever data is needed for the type of resource hierarchy that you are creating.

LifeKeeper Application Resource Hierarchies

If you install LifeKeeper without any recovery kits, the Select Recovery Kit list includes options for File System or Generic Application by default. The Generic Application option may be used for applications that have no associated recovery kits.

If you install the Raw I/O or IP Recovery Kits (both of which are Core Recovery Kits that are packaged separately and included on the LifeKeeper Core media), the Select Recovery Kit list will provide additional options for these Recovery Kits.

See the following topics describing these available options:

- ◦ Creating a File System Resource Hierarchy
- ◦ Creating a Generic Application Resource Hierarchy
- ◦ Creating a Raw Device Resource Hierarchy

See the IP Recovery Kit Technical Documentation for more information.

Recovery Kit Options

Each optional recovery kit that you install adds entries to the Select Recovery Kit list; for example, you may see Oracle, Apache, and NFS Recovery Kits. Refer to the Administration Guide that accompanies each recovery kit for directions on creating the required resource hierarchies.

Note: If you wish to create a File System or other application resource hierarchy that is built on a logical volume, you must first have the Logical Volume Manager (LVM) Recovery Kit installed.

5.3.3.3.5.1. Creating a File System Resource Hierarchy

Use this option to protect a file system only (for example, if you have shared files that need protection).

1. There are four ways to begin creating a file system resource hierarchy.

   • Right-click on a server icon to bring up the server context menu, then click on **Create Resource Hierarchy**.

   • On the global toolbar, click on the **Create Resource Hierarchy** button.

   • On the server context toolbar, if displayed, click on the **Create Resource Hierarchy** button.

   • On the Edit menu, select **Server**, then click on **Create Resource Hierarchy**.

2. A dialog entitled *Create Resource Wizard* will appear with a **Recovery Kit** list. Select *File System Resource* and click **Next**.

3. Select the **Switchback Type** and click **Next**.

4. Select the **Server** and click **Next**. **Note:** *If you began from the server context menu, the server will be determined automatically from the server icon that you clicked on, and this step will be skipped.*

5. The Create *gen/filesys Resource* dialog will now appear. Select the **Mount Point** for the file system resource hierarchy and click **Next**. The selected mount point will be checked to see that it is shared with another server in the cluster by checking each storage kit to see if it recognizes the mounted device as shared. If no storage kit recognizes the mounted device, then an error dialog will be presented:

   <file system> is not a shared file system

   Selecting **OK** will return to the *Create gen/filsys Resource* dialog.

   Notes:

   • In order for a mount point to appear in the choice list, the mount point must be currently mounted. If an entry for the mount point exists in the `/etc/fstab` file, LifeKeeper will remove this entry during the creation and extension of the hierarchy. It is advisable to make a backup of `/etc/fstab` prior to using the NAS Recovery Kit, especially if you have complex mount settings. You can direct that entries are re-populated back into `/etc/fstab` on deletion by setting the `/etc/default/LifeKeeper` tunable `REPLACEFSTAB=true|TRUE`.

   • Many of these resources (SIOS DataKeeper, LVM, Device Mapper Multipath, etc.) require LifeKeeper recovery kits on each server in the cluster in order for the file system resource to

be created. If these kits are not properly installed, then the file system will not appear to be shared in the cluster.

6. LifeKeeper creates a default **Root Tag** for the file system resource hierarchy. (This is the label used for this resource in the status display). You can select this root tag or create your own, then click **Next**.

7. Click **Create Instance**. A window will display a message indicating the status of the instance creation.

8. Click **Next**. A window will display a message that the file system hierarchy has been created successfully.

9. At this point, you can click **Continue** to move on to extending the file system resource hierarchy, or you can click **Cancel** to return to the GUI. If you click **Cancel**, you will receive a warning message that your hierarchy exists on only one server, and it is not protected at this point.

5.3.3.3.5.2. Creating a Generic Application Resource Hierarchy

Use this option to protect a user-defined application that has no associated recovery kit. Templates are provided for the user supplied scripts referenced below in `$LKROOT/lkadm/subsys/gen/app/templates`. Copy these templates to another directory before customizing them for the application that you wish to protect and testing them.

> **Note:** For applications depending upon other resources such as a file system, disk partition, or IP address, create each of these resources separately, and use Create Dependency to create the appropriate dependencies.

1. There are four ways to begin creating a generic application resource hierarchy.

   • Right-click on a server icon to bring up the server context menu, then click on **Create Resource Hierarchy**.

   • On the global toolbar, click on the **Create Resource Hierarchy** button.

   • On the server context toolbar, if displayed, click on the **Create Resource Hierarchy** button.

   • On the Edit menu, select **Server**, then click on **Create Resource Hierarchy**.

2. A dialog entitled *Create Resource Wizard* will appear with a **Recovery Kit** list. Select **Generic Application** and click **Next**.

3. Select the **Switchback Type** and click **Next&**

4. Select the **Server** and click **Next**

   Note: *If you began from the server context menu, the server will be determined automatically from the server icon that you clicked on, and this step will be skipped.*

5. On the next dialog, enter the path to the **Restore Script** for the application and click **Next**. This is the command that starts the application. A template restore script, restore.template, is provided in the templates directory. The restore script must not impact applications that are already started.

6. Enter the path to the **Remove Script** for the application and click **Next**. This is the command that stops the application. A template remove script, remove.template, is provided in the templates directory.

7. Enter the path to the **quickCheck Script** for the application and click **Next**. This is the command that monitors the application. A template quickCheck script, quickCheck.template, is provided in the templates directory.

8. Enter the path to the **Local Recovery Script** for the application and click **Next**. This is the command that attempts to restore a failed application on the local server. A template recover script, recover.template, is provided in the templates directory.

9. Enter any **Application Information** and click **Next**. This is optional information about the application that may be needed by the restore, remove, recover, and quickCheck scripts.

10. Select either **Yes** or **No** for **Bring Resource In Service**, and click **Next**. Selecting No will cause the resource state to be set to OSU following the create; selecting Yes will cause the previously provided restore script to be executed. For applications depending upon other resources such as a file system, disk partition, or IP address, select No if you have not already created the appropriate dependent resources.

11. Enter the **Root Tag**, which is a unique name for the resource instance. (This is the label you will see for this resource in the status display.)

12. Click **Create Instance** to start the creation process. A window will display a message indicating the status of the instance creation.

13. Click **Next**. A window will display a message that the hierarchy has been created successfully.

14. At this point, you can click **Continue** to move on to extending the generic application resource hierarchy, or you can click **Cancel** to return to the GUI. If you click Cancel, you will receive a warning that your hierarchy exists on only one server, and it is not protected at this point.

Note: The scripts which are provided when resource hierarchy is created, such as restore、remove、quickCheck, are located in each directory under LKROOT/subsys/gen/resource/app/.

- restore – actions/!restore/<tag name>
- remove – actions/!remove/<tag name>
- quickCheck – actions/!quickCheck/<tag name>
- recover – recovery/!recover/<tag name>

5.3.3.3.5.3. Creating a Raw Device Resource Hierarchy

Use this option to protect a raw device resource. For example, if you create additional table space on a raw device that needs to be added to an existing database hierarchy, you would use this option to create a raw device resource.

> **Note:** LifeKeeper locks shared disk partition resources at the disk logical unit (or LUN) level to one system in a cluster at a time.

1. There are four ways to begin creating a raw device resource hierarchy.

   • Right-click on a server icon to bring up the server context menu, then click on **Create Resource Hierarchy**.

   • On the global toolbar, click on the **Create Resource Hierarchy** button.

   • On the server context toolbar, if displayed, click on the **Create Resource Hierarchy** button.

   • On the Edit menu, select **Server**, then click on **Create Resource Hierarchy**.

2. A dialog entitled Create Resource Wizard will appear with a **Recovery Kit** list. Select Raw Device and click **Next**.

3. Select the **Switchback Type** and click **Next**.

4. Select the **Server** and click **Next**.

   Note: *If you began from the server context menu, the server will be determined automatically from the server icon that you clicked on, and this step will be skipped.*

5. Select the **Raw Partition** on a shared storage device where this resource will reside, and click **Next**.

6. Enter the **Root Tag**, which is a unique name for the resource instance. (This is the label you will see for this resource in the status display.)

7. Click **Create Instance** to start the creation process. A window titled Creating scsi/raw resource will display text indicating what is happening during creation.

8. Click **Next**. A window will display a message that the hierarchy has been created successfully.

9. At this point, you can click **Continue** to move on the extending the raw resource hierarchy, or you can click **Cancel** to return to the GUI. If you click **Cancel**, you will receive a message warning that

your hierarchy exists on only one server, and it is not protected at this point

5.3.3.3.5.4. Quick Service Protection (QSP) Recovery Kit

The QSP Recovery Kit provides a simplified method to protect the OS service. With the QSP Recovery Kit users can easily create a LifeKeeper resource instance to protect an OS service provided that service can be started and stopped by the OS service command or the systemctl command (start/stop). The service can also be protected via the Generic Application Recovery Kit, but the use of that kit requires code development whereas the QSP Recovery Kit does not. Also, by creating a dependency relationship, protected services can be started and stopped in conjunction with the application that requires the service. The application is protected by another LifeKeeper resource instance not via the QSP resource.

However, the QSP Recovery Kit quickCheck can only perform simple health (using the “status” action of the service command). QSP doesn’t guarantee that the service is provided or the process is functioning. If complicated starting and/or stopping is necessary, or more robust health checking operations are necessary, using a Generic Application is recommended.

Requirements

The service to be protected by the QSP Recovery Kit needs to meet the following requirements.

- It must support start and stop actions via the OS service command or the systemctl command. Also, it must returns 0 when the start and stop action succeeds.
- To perform health checking the service must support the status action via the OS service command or the systemctl command. If it does not support the status action then quickCheck health check operations must be disabled. Also, it must return 0 when the status action succeeds.
- The name of the service to be protected must not exceed 256 characters in length and contain only alphanumeric characters.

Note: The compatible service command may be used to control protected resources even in a systemd environment.

The service to be protected by the QSP resource must be running (started) before attempting a resource create. Please notice that some services which are already supplied with dedicated Recovery Kit are not target of QSP (hereinafter referred as “the Services not targeted by QSP protection”) and cannot be protected by QSP Recovery Kit.

Create QSP Resource Hierarchy

This option is used to protect OS services via the QSP Recovery Kit.

1. There are 4 methods to start the creation of QSP resource instance.

   • Right-click on a server icon to bring up the server context menu, then click on [Create Resource Hierarchy].

   • On the global toolbar, click on the [Create Resource Hierarchy] button.

   • On the server context toolbar, if displayed, click on the [Create Resource Hierarchy] button.

   • On the Edit menu, select [Server], then click on [Create Resource Hierarchy].

2. A dialogue box titled [Create Resource Wizard] is displayed. In the [Recovery Kit] drop down is a list of available resource types to create. Select **Quick Service Protection** and click [Next].
   .
3. Select [Switchback Type] and click [Next].
   .
4. Select [Server] and click [Next].

   Note: If the create was started via the server context menu, this step is skipped because the server is known based on the start context (defaults to name of the server on which the create process started).

5. The next dialog box contains a drop down of the available services that can be protected. Select the [Service Name] to be protected and click [Next].

   Note: The list may not show the service if it is not running. In this case, click **Cancel** to discontinue the process, and start service. Once the service is running restart the create process. The list will not show the Services not targeted by QSP protection.

6. In the next dialog box the quickCheck action is configured.To enable the quickCheck monitoring function, select [enable]. To disable it, select [disable]. Click [Next] to continue. The quickCheck action can be changed at any time.

   Note: If the selected service does not support the “status” action via the OS service command, then set the quickCheck action to “disabled” because the QSP Recovery Kit cannot monitor the service state.

7. Input the [Resource Tag] This is a unique name for the resource instance. (This is the label that uniquely identifies the resource instance and is used whenever displaying LifeKeeper protected resource instances in UI.)

8. Click [Create Instance] to start the creation process. The status of the resource instance creation is displayed in the status window.

9. Click [Next] to display the resource extension dialog. Click [Next] to begin the extension process or click [Cancel] to go back to GUI. When [Cancel] is clicked, an alert is displayed that the hierarchy exist on only one server, and protection by LifeKeeper is not available at this time.

Extending QSP Resource Hierarchy

This function, as explained in the section Extending Resource Hierarchies, starts automatically after finishing the Create QSP Resource Hierarchy (URL) process, or, from right clicking on an existing QSP resource and selecting [Extend Resource Hierarchy]. After finishing the pre-extend process, then, complete the following steps.

1. Select [Resource Tag] provided by LifeKeeper, or, input unique tag for the resource hierarchy on the target server.

2. Click [Extend] to start the extension process. The status of the extension process is displayed in the dialogue box, and when it is finished it will show a message indicating the hierarchy is correctly extended. If the hierarchy is to be extended to another server click [Next Server], otherwise click [Finish\ to complete the extension. If [Next Server] is selected, the extension operation is repeated.

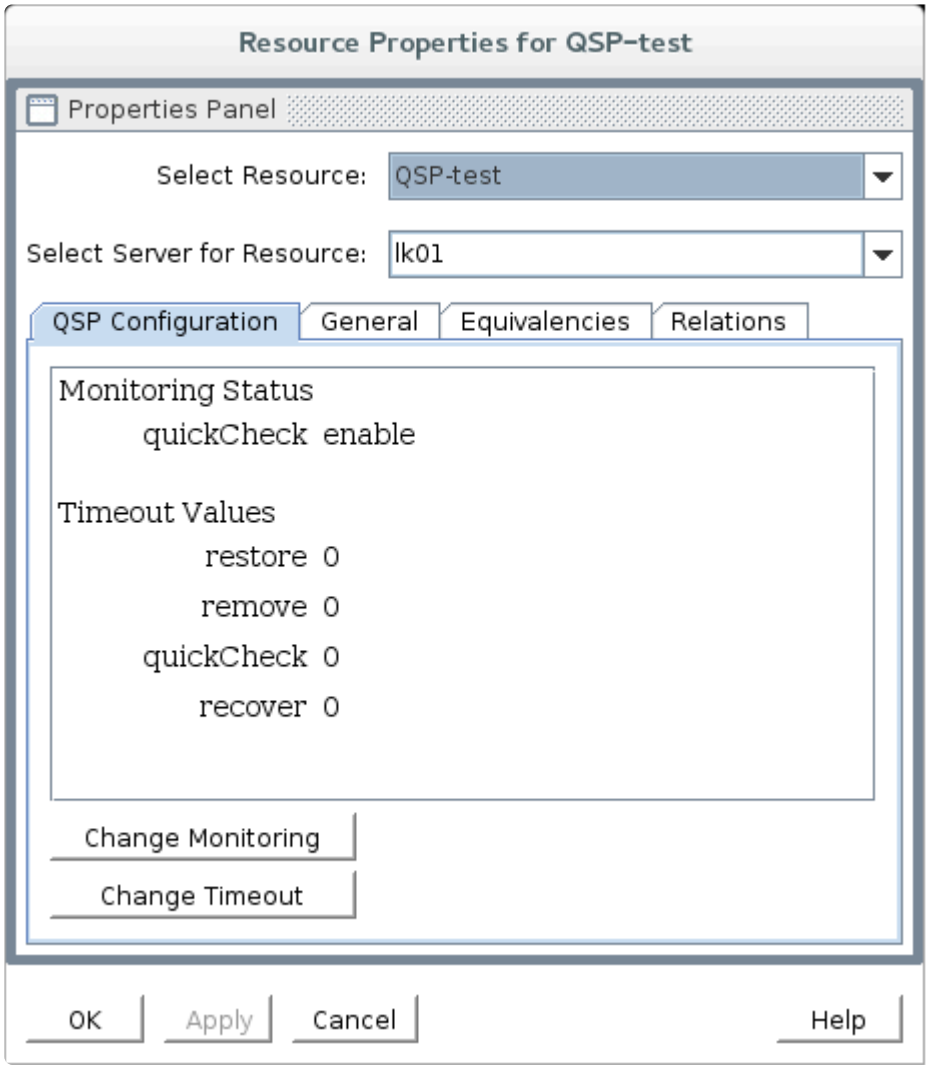
3. When [Finish] is clicked the integrity of the hierarchy is checked. If any problems are detected the extension is reversed. To complete the verification and close the dialog box click [Done].

QSP Resource Configuration

The following parameters are unique to each QSP resource instance are available for modification.

Set Up Items		Default Value	Description
Monitoring	quickCheck	Specified when creating the resource	Set to enable the checking of the status of the service or to disable / skip the monitoring function
Time Out	restore	0	Specify the restore timeout (unit: second). If set to 0 no timeout occurs when restoring the resource instance.
	remove	0	Specify the remove timeout (unit: second). If set to 0 no timeout occurs when removing the resource instance.
	quickCheck	0	Specify the quickCheck timeout (unit: second). If set to 0 no time out occurs when performing health checking of the resource instance.
	recover	0	Specify the recover timeout (unit: second). If set to 0 no timeout occurs during recovery of the resource instance.

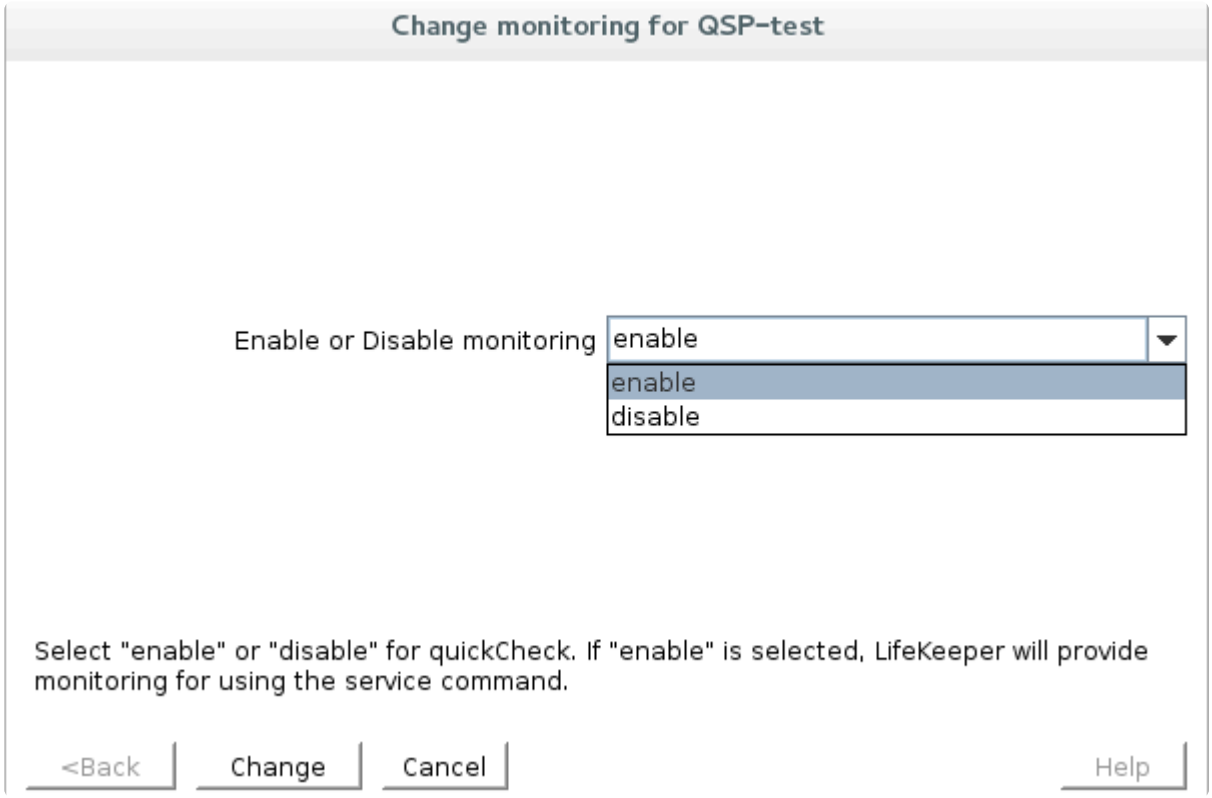
Checking / changing of the set value is possible from the **QSP Configuration** tab by Display Resource Properties and must be performed on each node in the hierarchy. If the quickCheck function is disabled, quickCheck and recover of timeouts are not displayed and thus cannot be changed.



How to Change the Monitoring Function

1. Display the [QSP Configuration] tab of the resource properties, and click [Change quickCheck] .
2. Select [enable] to enable quickCheck, or [disable] to disable it.
3. Clicking [Change] starts the change process, and display change process message.
4. Finish by clicking [Done].

   Note: Modification of these values is a per node operation. If the same change is needed on another node, then the process must be repeated on that node.



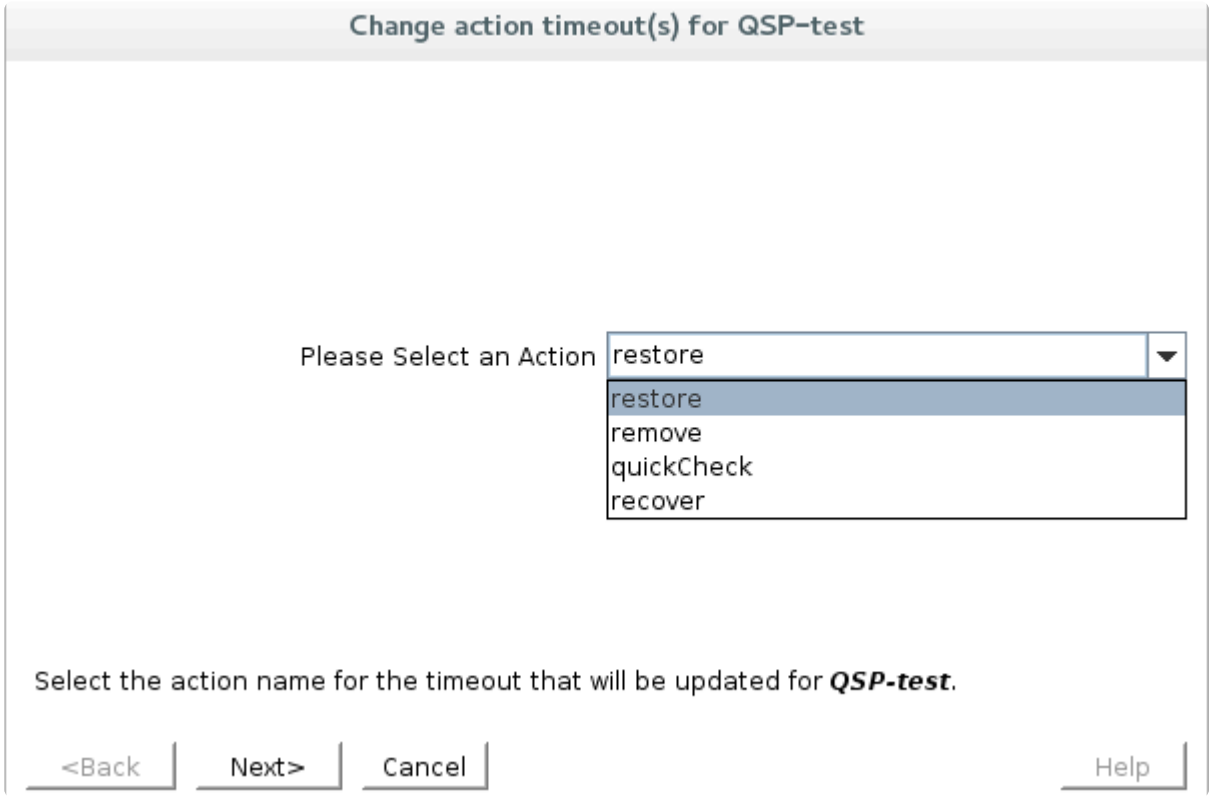
How to Change Timeout Value

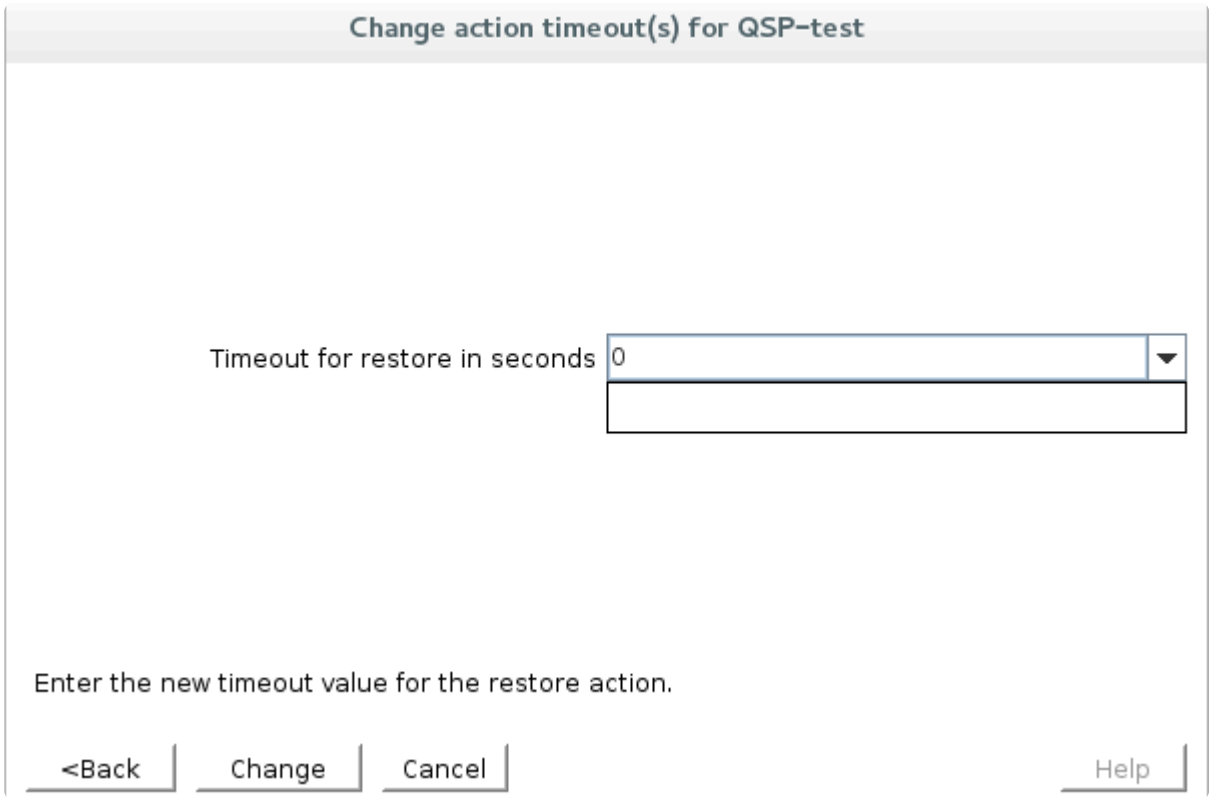
1. Display the [QSP Configuration] tab of resource properties, and click [Change Timeout].
2. Select the timeout action to be changed (restore, remove, quickCheck or recover), and click **Next**.

   Note: [quickCheck] and [recover] timeouts are not displayed in the select list if the monitoring function is disabled.
3. Input the timeout value seconds.

   Note: Input decimal numbers only. Non numerical characters are invalid.
4. Clicking [Change] starts the timeout change process and displays change process messages.
5. Finish by clicking [Done].

   Note: Modification of these values is a per node operation. If the same change is needed on another node, then the process must be repeated on that node.


Change action timeout(s) for QSP-test
Please Select an Action
restore
restore
remove
quickCheck
recover
Select the action name for the timeout that will be updated for QSP-test.
<Back
Next>
Cancel
Help


Change action timeout(s) for QSP-test
Timeout for restore in seconds
0
Enter the new timeout value for the restore action.
<Back
Change
Cancel
Help

5.3.3.3.6. Editing Resource Properties

1. To edit the properties of a resource, bring up the Resource Properties dialog just as you would for viewing resource properties.

2. If you are logged into that server with the appropriate permissions, the following items will be editable.

- º Switchback
- º Resource Configuration (only for resources with specialized configuration settings)
- º

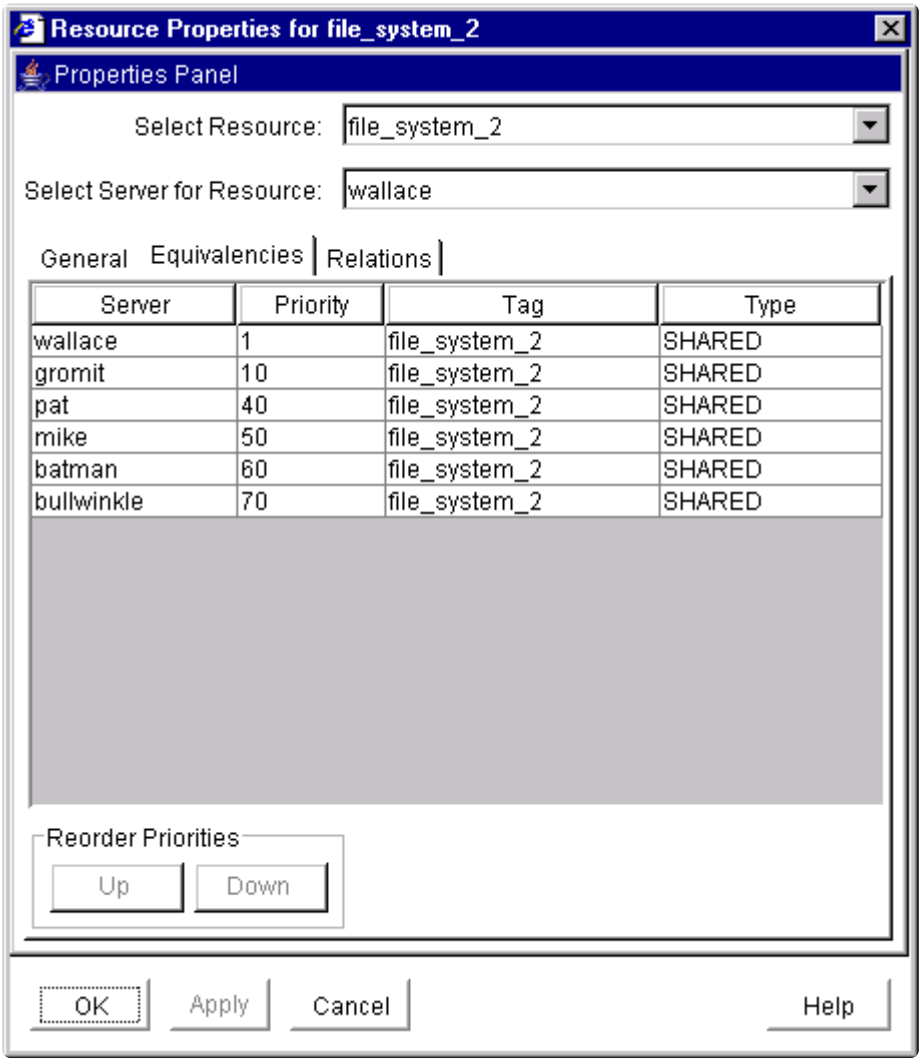
Resource Priorities

3. Once you have made changes, the **Apply** button will be enabled. Clicking this button will apply your changes without closing the window.

4. When you are finished, click **OK** to save any changes and close the window, or **Cancel** to close the window without applying changes.

5.3.3.3.7. Editing Resource Priorities

You can edit or reorder the priorities of servers on which a resource hierarchy has been defined. First, bring up the Resource Properties dialog just as you would for viewing resource properties. The Resource Properties dialog displays the priority for a particular resource on a server in the Equivalencies Tab as shown below.



There are two ways to modify the priorities:

- ◦ Reorder the priorities by moving an equivalency with the **Up/Down** buttons ,or
- ◦ Edit the priority values directly.

Using the Up and Down Buttons

1. Select an equivalency by clicking on a row in the Equivalencies table. The **Up** and/or **Down** buttons will become enabled, depending on which equivalency you have selected. The **Up** button is enabled unless you have selected the highest priority server. The **Down** button is enabled unless you have selected the lowest priority server.

2. Click **Up** or **Down** to move the equivalency in the priority list.

The numerical priorities column will not change, but the equivalency will move up or down in the list.

Editing the Priority Values

1. Select a priority by clicking on a priority value in the Priority column of the Equivalencies table. A box appears around the priority value, and the value is highlighted.

2. Enter the desired priority and press **Enter**.

> **Note:** Valid server priorities are 1 to 999.

After you have edited the priority, the Equivalencies table will be re-sorted.

Applying Your Changes

Once you have the desired priority order in the Equivalencies table, click **Apply** (or **OK**) to commit your changes. The **Apply** button applies any changes that have been made. The **OK** button applies any changes that have been made and then closes the window. The **Cancel** button closes the window without saving any changes made since **Apply** was last clicked.

5.3.3.3.8. Extending Resource Hierarchies

The LifeKeeper **Extend Resource Hierarchy** option copies an existing hierarchy from one server and creates a similar hierarchy on another LifeKeeper server. Once a hierarchy is extended to other servers, cascading failover is available for that resource. The server where the existing hierarchy currently resides is referred to as the template server. The server where the new extended hierarchy will be placed is referred to as the target server.

The target server must be capable of supporting the extended hierarchy and it must be able to communicate with equivalent hierarchies on other remote servers (via active LifeKeeper communications paths). This means that all recovery kits associated with resources in the existing hierarchy must already be installed on the target server, as well as every other server where the hierarchy currently resides.

1. There are five ways to extend a resource hierarchy through the GUI.

- º

Create a new resource hierarchy. When the dialog tells you that the hierarchy has been created, click on the **Continue** button to start extending your new hierarchy via the Pre-Extend Wizard.

- º Right-click on a global or server-specific resource icon to bring up the

resource context menu, then click on Extend Resource Hierarchy to extend the selected resource via the Pre-Extend Wizard.

- º On the

global toolbar, click on the **Extend Resource Hierarchy** button. When the Pre-Extend Wizard dialog appears, select a **Template Server** and a **Tag to Extend**, clicking on **Next** after each choice.

- º On the

resource context toolbar, if displayed, click on the **Extend Resource Hierarchy** button to bring up the Pre-Extend Wizard.

- º On the

Edit menu, select **Resource**, then click on **Extend Resource Hierarchy**. When the Pre-Extend Wizard dialog appears, select a **Template Server** and a **Tag to Extend**, clicking on **Next** after each choice.

2. Either select the default **Target Server** or enter one from the list of choices, then click **Next**.

3. Select the **Switchback Type**, then click **Next**.

4. Either select the default or enter your own **Template Priority**, then click **Next**.

5. Either select or enter your own **Target Priority**, then click **Next**.

6. The dialog will then display the pre-extend checks that occur next. If these tests succeed, LifeKeeper goes on to perform any steps that are needed for the specific type of resource that you are extending.

The **Accept Defaults** button which is available for the **Extend Resource Hierarchy** option is intended

for the user who is familiar with the **LifeKeeper Extend Resource Hierarchy** defaults, and wants to quickly extend a LifeKeeper resource hierarchy without being prompted for input or confirmation. Users who prefer to extend a LifeKeeper resource hierarchy using the interactive, step-by-step interface of the GUI dialogs should use the **Next** button.

Note: ALL roots in a multi-root hierarchy must be extended together, they may not be extended as single root hierarchies.

Note: For command line instructions, see Extending the SAP Resource from the Command Line.

5.3.3.3.8.1. Extending a File System Resource Hierarchy

This operation can be started automatically after you have finished creating a file system resource hierarchy, or from an existing file system resource, as described in the section on extending resource hierarchies. Then complete the steps below, which are specific to file system resources.

1. The *Extend gen/filesys Resource Hierarchy* dialog box appears. Select the **Mount Point** for the file system hierarchy, then click **Next**.

2. Select the **Root Tag** that LifeKeeper offers, or enter your own tag for the resource hierarchy on the target server, then click **Next**.

3. The dialog displays the status of the extend operation, which should finish with a message saying that the hierarchy has been successfully extended. Click **Next Server** if you want to extend the same resource hierarchy to a different server. This will repeat the extend operation. Or click **Finish** to complete this operation.

4. The dialog then displays verification information as the extended hierarchy is validated. When this is finished, the **Done** button will be enabled. Click **Done** to finish.

5.3.3.3.8.2. Extending a Generic Application Resource Hierarchy

This operation can be started automatically after you have finished creating a generic application resource hierarchy, or from an existing generic application resource, as described in the section on extending resource hierarchies. Then complete the steps below, which are specific to generic application resources.

1. Select the **Root Tag** that LifeKeeper offers, or enter your own tag for the resource hierarchy on the target server, then click **Next**.

2. Enter any **Application Information** next (optional), then click **Next**.

3. The dialog displays the status of the extend operation, which should finish with a message saying that the hierarchy has been successfully extended. Click **Next Server** if you want to extend the same resource hierarchy to a different server. This will repeat the extend operation. Or click **Finish** to complete this operation.

4. The dialog then displays verification information as the extended hierarchy is validated. When this is finished, the **Done** button will be enabled. Click **Done** to finish.

5.3.3.3.8.3. Extending a Raw Device Resource Hierarchy

This operation can be started automatically after you have finished creating a raw device resource hierarchy, or from an existing raw device resource, as described in the section on extending resource hierarchies. Then complete the steps below, which are specific to raw device resources.

1. Select the **Root Tag** that LifeKeeper offers, or enter your own tag for the resource hierarchy on the target server, then click **Next**.

2. The dialog displays the status of the extend operation, which should finish with a message saying that the hierarchy has been successfully extended. Click **Next Server** if you want to extend the same resource hierarchy to a different server. This will repeat the extend operation. Or click **Finish** to complete this operation.

3. The dialog then displays verification information as the extended hierarchy is validated. When this is finished, the **Done** button will be enabled. Click **Done** to finish.

5.3.3.3.9. Unextending a Hierarchy

The LifeKeeper **Unextend Resource Hierarchy** option removes a complete hierarchy, including all of its resources, from a single server. This is different than the **Delete Resource Hierarchy** selection which removes a hierarchy from all servers.

When using **Unextend Resource Hierarchy**, the server from which the existing hierarchy is to be removed is referred to as the target server.

The **Unextend Resource Hierarchy** selection can be used from any LifeKeeper server that has active LifeKeeper communications paths to the target server.

1. There are five possible ways to begin.

   - º Right-click on the icon for the resource hierarchy/server combination that you want to unextended. When the

resource context menu appears, click **Unextend Resource Hierarchy**.
   - º Right-click on the icon for the global resource hierarchy that you want to unextended. When the

resource context menu appears, click **Unextend Resource Hierarchy**. When the dialog comes up, select the server in the **Target Server** list from which you want to unextended the resource hierarchy, and click **Next**.
   - º On the

global toolbar, click the **Unextend Resource Hierarchy** button. When the dialog comes up, select the server in the **Target Server** list from which you want to unextended the resource hierarchy, and click **Next**. On the next dialog, select the resource hierarchy that you want to unextended from the **Hierarchy to Unextend** list, and click **Next** again.
   - º On the

resource context toolbar, if displayed, click the **Unextend Resource Hierarchy** button.
   - º On the

Edit menu, point to **Resource** and then click **Unextend Resource Hierarchy**. When the dialog comes up, select the server in the **Target Server** list from which you want to unextended the resource hierarchy, and click **Next**. On the next dialog, select the resource hierarchy that you want to unextended from the **Hierarchy to Unextend** list, and click **Next** again.
2. The dialog will display a message verifying the server and resource hierarchy that you have specified to be unextended. Click **Unextend** to perform the action.

3. If the output panel is enabled, the dialog closes, and the results of the commands to unextended the resource hierarchy are shown in the output panel. If not, the dialog remains up to show these results, and you click **Done** to finish when all results have been displayed.

5.3.3.3.10. Creating a Resource Dependency

While most Recovery Kits create their dependencies during the original resource hierarchy creation task, under certain circumstances, you may want to create new or additional resource dependencies or delete existing ones. An example might be that you wish to change an existing IP dependency to another IP address. Instead of deleting the entire resource hierarchy and creating a new one, you can delete the existing IP dependency and create a new dependency with a different IP address.

1. There are four possible ways to begin.

- º Right-click on the icon for the parent server-specific resource under the server, or the parent global resource, to which you want to add a parent-child dependency. When the

resource context menu appears, click **Create Dependency**.

Note: If you right-clicked on a server-specific resource in the right pane, the value of the **Server** will be that server. If you right-clicked on a global resource in the left pane, the value of the **Server** will be the server where the resource has the highest priority.

- º On the

global toolbar, click the **Create Dependency** button. When the dialog comes up, select the server in the **Server** list from which you want to begin creating the resource dependency, and click **Next**. On the next dialog, select the parent resource from the **Parent Resource Tag** list, and click **Next** again.

- º On the

resource context toolbar, if displayed, click the **Create Dependency** button.

- º On the

Edit menu, point to **Resource** and then click **Create Dependency**. When the dialog comes up, select the server in the **Server** list from which you want to begin creating the resource dependency, and click **Next**. On the next dialog, select the parent resource from the **Parent Resource Tag** list, and click **Next** again.

2. Select a **Child Resource Tag** from the drop down box of existing and valid resources on the server. The dialog will display all the resources available on the server with the following exceptions:

- º The parent resource, its ancestors, and its children.
- º A resource that has not been extended to the same servers as the parent resource.
- º A resource that does not have the same relative priority as the parent resource.
- º Any resource that is not in-service on the same server as the parent, if the parent resource is in-service.

Click **Next** to proceed to the next dialog.

3. The dialog will then confirm that you have selected the appropriate parent and child resource tags

for your dependency creation. Click **Create Dependency** to create the dependency on all servers in the cluster to which the parent has been extended.

4. If the output panel is enabled, the dialog closes, and the results of the commands to create the dependency are shown in the output panel. If not, the dialog remains up to show these results, and you click **Done** to finish when all results have been displayed.

5.3.3.3.11. Deleting a Resource Dependency

1. There are four possible ways to begin.

   - º Right-click on the icon for the parent server-specific resource under the server, or the parent global resource, from which you want to delete a parent-child dependency. When the

resource context menu appears, click **Delete Dependency**.

- º On the

global toolbar, click the **Delete Dependency** button. When the dialog comes up, select the server in the **Server** list from which you want to begin deleting the resource dependency, and click **Next**. On the next dialog, select the parent resource from the **Parent Resource Tag** list, and click **Next** again.

- º On the

resource context toolbar, if displayed, click the **Delete Dependency** button.

- º On the

Edit menu, point to **Resource** and then click **Delete Dependency**. When the dialog comes up, select the server in the **Server** list from which you want to begin deleting the resource dependency, and click **Next**. On the next dialog, select the parent resource from the **Parent Resource Tag** list, and click **Next** again.

2. Select the **Child Resource Tag** from the drop down box. This should be the tag name of the child in the dependency that you want to delete. Click **Next** to proceed to the next dialog box.

3. The dialog then confirms that you have selected the appropriate parent and child resource tags for your dependency deletion. Click **Delete Dependency** to delete the dependency on all servers in the cluster.

4. If the output panel is enabled, the dialog closes, and the results of the commands to delete the dependency are shown in the output panel. If not, the dialog remains up to show these results, and you click Done to finish when all results have been displayed.

5.3.3.3.12. Deleting a Hierarchy from All Servers

1. There are five possible ways to begin.

   - Right-click on the icon for a resource in the hierarchy that you want to delete under the server where you want the deletion to begin. When the resource context menu appears, click **Delete Resource Hierarchy**.
   - Right-click on the icon for a global resource in the hierarchy that you want to delete. When the resource context menu appears, click **Delete Resource Hierarchy**. When the dialog comes up, select the server in the **Target Server** list from which you want to begin deleting the resource hierarchy, and click **Next**.
   - On the global toolbar, click the **Delete Resource Hierarchy** button. When the dialog comes up, select the server in the **Target Server** list from which you want to begin deleting the resource hierarchy, and click **Next**. On the next dialog, select a resource in the hierarchy that you want to delete from the **Hierarchy to Delete** list, and click **Next** again.
   - On the resource context toolbar in the properties panel, if displayed, click the **Delete Resource Hierarchy** button.
   - On the Edit menu, point to **Resource** and then click **Delete Resource Hierarchy**. When the dialog comes up, select the server in the **Target Server** list from which you want to begin deleting the resource hierarchy, and click **Next**. On the next dialog, select a resource in the hierarchy that you want to delete from the **Hierarchy to Delete** list, and click **Next** again.

2. The dialog will display a message verifying the hierarchy you have specified for deletion. Click **Delete** to perform the action.

3. If the output panel is enabled, the dialog closes, and the results of the commands to delete the hierarchy are shown in the output panel. If not, the dialog remains up to show these results, and you click **Done** to finish when all results have been displayed.

5.3.4. User Guide

The User Guide is a complete, searchable resource containing detailed information on the many tasks that can be performed within the LifeKeeper GUI. Click User Guide to access this documentation.

The tasks that can be performed through the GUI can be grouped into three areas:

Common Tasks – These are basic tasks that can be performed by any user such as connecting to a cluster, viewing server or resource properties, viewing log files and changing GUI settings.

Operator Tasks – These are more advanced tasks that require Operator permission, such as bringing resources in and out of service.

Administrator Tasks – These are tasks that require Administrator permission. They include server-level tasks such as editing server properties, creating resources, creating or deleting comm paths and resource-level tasks such as editing, extending, or deleting resources.

The table below lists the default tasks that are available for each user permission. Additional tasks may be available for specific resource types, and these will be described in the associated resource kit documentation.

Task	Permission		
	Guest	Operator	Administrator
View servers and resources	X	X	X
Connect to and disconnect from servers	X	X	X
View server properties and logs	X	X	X
Modify server properties			X
Create resource hierarchies			X
Create and delete comm paths			X
View resource properties	X	X	X
Modify resource properties			X
Take resources into and out of service		X	X
Extend and unextend resource hierarchies			X
Create and delete resource dependencies			X
Delete resource hierarchies			X

5.3.4.1. Using LifeKeeper for Linux

The following topics provide detailed information on the LifeKeeper graphical user interface (GUI) as well as the many tasks that can be performed within the LifeKeeper GUI.

5.3.4.1.1. GUI

The GUI components should have already been installed as part of the LifeKeeper Core installation.

The LifeKeeper GUI uses Java technology to provide a graphical user interface to LifeKeeper and its configuration data. Since the LifeKeeper GUI is a client/server application, a user will run the graphical user interface on a client system in order to monitor or administer a server system where LifeKeeper is running. The client and the server components may or may not be on the same system.

5.3.4.1.1.1. GUI Overview – General

The GUI allows users working on any machine to administer, operate or monitor servers and resources in any cluster as long as they have the required group memberships on the cluster machines. (For details, see Configuring GUI Users. The GUI Server and Client components are described below.

GUI Server

The GUI server communicates with GUI clients using Hypertext Transfer Protocol (HTTP) and Remote Method Invocation (RMI). By default, the GUI server is initialized during LifeKeeper startup, but this can also be configured — see Starting/Stopping the GUI Server.

GUI Client

The GUI client can be run either as an application on any LifeKeeper server or as a web client on any Java-enabled system.

The client includes the following components:

- ◦ The status table on the upper left displays the high level status of connected servers and their resources.
- ◦ The properties panel on the upper right displays detailed information about the most recently selected status table object.
- ◦ The output panel on the bottom displays command output.
- ◦ The message bar at the very bottom of the window displays processing status messages.
- ◦ The context (in the properties panel) and global toolbars provide fast access to frequently used tasks.
- ◦ The context (popup) and global menus provide access to all tasks.

Exiting GUI Clients

Select **Exit** from the File Menu to disconnect from all servers and close the client.

5.3.4.1.1.1.1. LifeKeeper GUI Software Package

The LifeKeeper GUI is included in the **steeleye-lkGUI** software package which is bundled with the LifeKeeper Core Package Cluster. The **steeleye-lkGUI** package:

- ◦ Installs the LifeKeeper GUI Client in Java archive format.
- ◦ Installs the LifeKeeper GUI Server.
- ◦ Installs the LifeKeeper administration web server.

  Note: The LifeKeeper administration web server is configured to use Port 81, which should be different from any public web server.

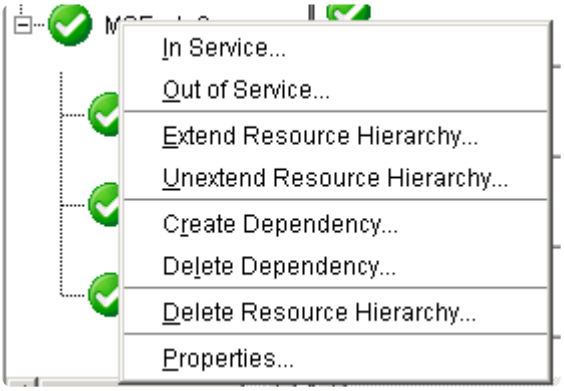
- ◦ Installs a Java policy file in /opt/LifeKeeper/htdoc/ which contains the minimum permissions required to run the LifeKeeper GUI. The LifeKeeper GUI application uses the java.policy file in this location for access control.
- ◦ Prepares LifeKeeper for GUI administration.

Before continuing, you should ensure that the LifeKeeper GUI package has been installed on the LifeKeeper server(s). You can enter the command **`rpm -qi steeleye-lkGUI`** to verify that this package is installed. You should see output including the package name **`steeleye-lkGUI`** if the GUI package is installed.

5.3.4.1.1.2. Menus

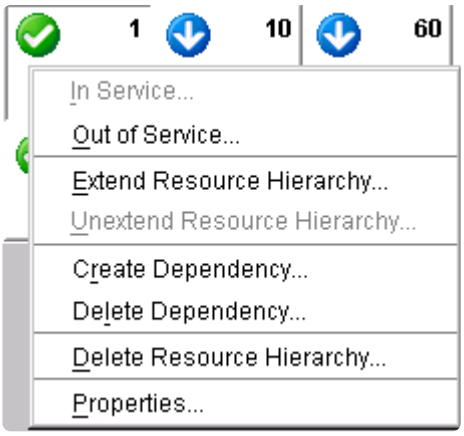
SIOS LifeKeeper for Linux Menus

5.3.4.1.1.2.1. Resource Context Menu



The Resource Context Menu appears when you right-click on a global (cluster-wide) resource, as shown above, or a server-specific resource instance, as shown below, in the status table. The default resource context menu is described here, but this menu might be customized for specific resource types, in which case the menu will be described in the appropriate resource kit documentation.

The actions are invoked for the resource that you select. If you select a resource instance on a specific server, the action is invoked for that server while if you select a global (cluster-wide) resource, you will need to select the server.



In Service. Bring a resource hierarchy into service.

Out of Service. Take a resource hierarchy out of service.

Extend Resource Hierarchy. Copy a resource hierarchy to another server for failover support.

Unextend Resource Hierarchy. Remove an extended resource hierarchy from a single server.

Create Dependency. Create a parent/child relationship between two resources.

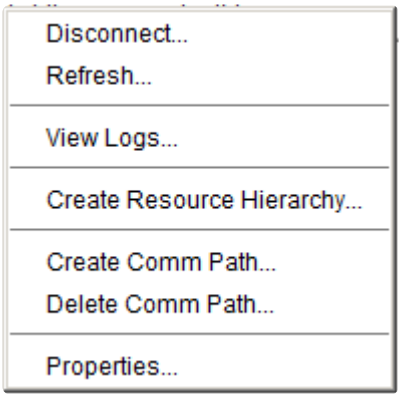
Delete Dependency. Remove a parent/child relationship between two resources.

Delete Resource Hierarchy. Remove a resource hierarchy from all servers in the LifeKeeper cluster.

Properties. Display the Resource Properties Dialog.

5.3.4.1.1.2.2. Server Context Menu

The Server Context Menu appears when you right-click on a server icon in the status table. This menu is the same as the Edit Menu's Server submenu except that the actions are always invoked on the server that you initially selected.



Disconnect. Disconnect from a cluster.

Refresh. Refresh GUI.

View Logs. View LifeKeeper log messages on connected servers.

Create Resource Hierarchy. Create a resource hierarchy.

Create Comm Path. Create a communication path between servers.

Delete Comm Path. Remove communication paths from a server.

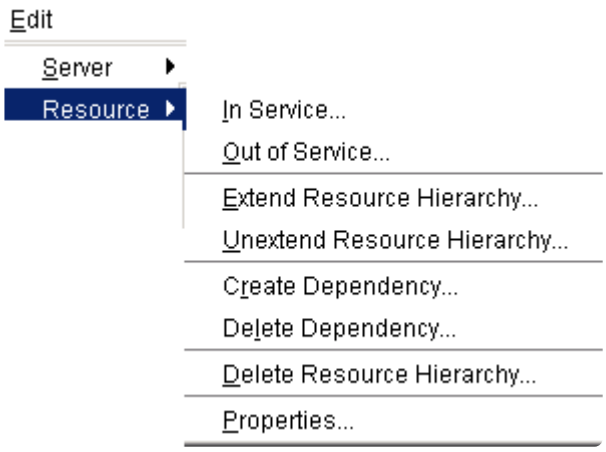
Properties. Display the Server Properties Dialog.

5.3.4.1.1.2.3. File Menu

Connect. Connect to a LifeKeeper cluster. Connection to each server in the LifeKeeper cluster requires login authentication on that server.

Exit. Disconnect from all servers and close the GUI window.

5.3.4.1.1.2.4. Edit Menu – Resource



In Service. Bring a resource hierarchy into service.

Out of Service. Take a resource hierarchy out of service.

Extend Resource Hierarchy. Copy a resource hierarchy to another server for failover support.

Unextend Resource Hierarchy. Remove an extended resource hierarchy from a single server.

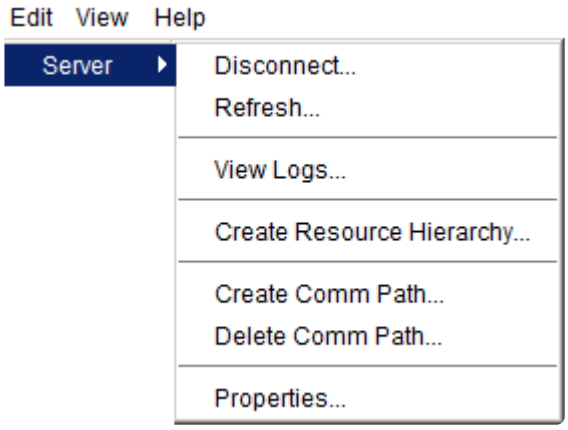
Create Dependency. Create a parent/child relationship between two resources.

Delete Dependency. Remove a parent/child relationship between two resources.

Delete Resource Hierarchy. Remove a resource hierarchy from all servers in the LifeKeeper cluster.

Properties. Display the Resource Properties Dialog.

5.3.4.1.1.2.5. Edit Menu – Server



Disconnect. Disconnect from a cluster.

Refresh. Refresh GUI.

View Logs. View LifeKeeper log messages on connected servers.

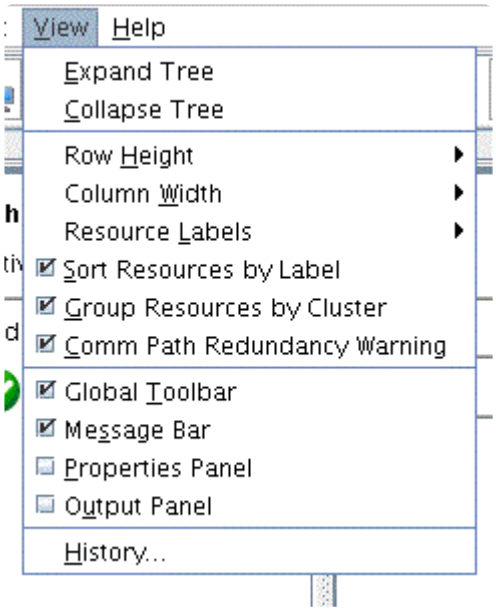
Create Resource Hierarchy. Create a resource hierarchy.

Create Comm Path. Create a communication path between servers.

Delete Comm Path. Remove communication paths from a server.

Properties. Display the Server Properties Dialog.

5.3.4.1.1.2.6. View Menu



Expand Tree. Expand the entire resource hierarchy tree.

Collapse Tree. Collapse the entire resource hierarchy tree.

Row Height. Modify the row viewing size of the resources in the resource hierarchy tree and table. Select small, medium or large row height depending upon the number of resources displayed.

Column Width. Modify the column with viewing size of the resources in the resource hierarchy tree and table. Select fill available space, large, medium or small depending upon the resource displayed.

Resource Labels. This option group allows you to specify whether resources are viewed in the resource hierarchy tree by their tag name or ID.

Sort Resources by Label. will sort resources by resource label only.

Group Resources by Cluster. will sort by server cluster and resource label such that resources belonging in the same cluster of servers will be grouped together.

Comm Path Redundancy Warning. specifies the representation of comm path status in the server status graphic.

- ◦ If selected, the display will show a server warning graphic if the comm paths between a set of servers are not configured with a redundant comm path.
- ◦ If not selected, the display will ignore a lack of redundant comm paths between a pair of servers but will still present server warning graphic if there are comm path failures.

Global Toolbar. Display this component if the checkbox is selected.

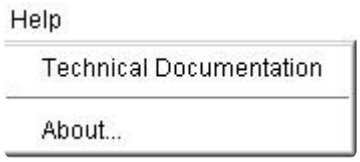
Message Bar. Display this component if the checkbox is selected.

Properties Panel. Display this component if the checkbox is selected.

Output Panel. Display this component if the checkbox is selected.

History. Display the newest messages that have appeared in the Message Bar in the LifeKeeper GUI Message History dialog box (up to 1000 lines).

5.3.4.1.1.2.7. Help Menu



Technical Documentation. Displays the landing page of the SIOS Technology Corp. Technical Documentation.

About.... Displays LifeKeeper GUI version information.

5.3.4.1.1.3. Toolbars

SIOS LifeKeeper for Linux Toolbars

5.3.4.1.1.3.1. GUI Toolbar

This toolbar is a combination of the default server and resource context toolbars which are displayed on the properties panel except that you must select a server and possibly a resource when you invoke actions from this toolbar.

	Connect. Connect to a LifeKeeper cluster.
	Disconnect. Disconnect from a LifeKeeper cluster.
	Refresh. Refresh GUI.
	View Logs. View LifeKeeper log messages on connected servers.
	Create Resource Hierarchy. Create a resource hierarchy.
	Delete Resource Hierarchy. Remove a resource hierarchy from all servers in the LifeKeeper cluster.
	Create Comm Path. Create a communication path between servers.
	Delete Comm Path. Remove communication paths from a server.
	In Service. Bring a resource hierarchy into service.
	Out of Service. Take a resource hierarchy out of service.
	Extend Resource Hierarchy. Copy a resource hierarchy to another server for failover support.
	Unextend Resource Hierarchy. Remove an extended resource hierarchy from a single server.
	Create Dependency. Create a parent/child relationship between two resources.

	Delete Dependency. Remove a parent/child relationship between two resources.
	The Multi-Site feature has been discontinued.

5.3.4.1.1.3.2. Resource Context Toolbar

The resource context toolbar is displayed in the properties panel when you select a server-specific resource instance in the status table.

The actions are invoked for the server and the resource that you select. Actions that are not available for selection for a resource will be grayed out.

	In Service. Bring a resource hierarchy into service.
	Out of Service. Take a resource hierarchy out of service.
	Extend Resource Hierarchy. Copy a resource hierarchy to another server for failover support.
	Unextend Resource Hierarchy. Remove an extended resource hierarchy from a single server.
	Add Dependency. Create a parent/child relationship between two resources.
	Remove Dependency. Remove a parent/child relationship between two resources.
	Delete Resource Hierarchy. Remove a resource hierarchy from all servers.

5.3.4.1.1.3.3. Server Context Toolbar

The server context toolbar is displayed in the properties panel when you select a server in the status table. The actions are invoked for the server that you select.

	Disconnect. Disconnect from a LifeKeeper cluster.
	Refresh. Refresh GUI.
	View Logs. View LifeKeeper log messages on connected servers.
	Create Resource Hierarchy. Create a resource hierarchy.
	Delete Resource Hierarchy. Remove a resource hierarchy from all servers in the LifeKeeper cluster.
	Create Comm Path. Create a communication path between servers.
	Delete Comm Path. Remove communication paths from a server.

5.3.4.1.1.4. Preparing to Run the GUI

5.3.4.1.1.4.1. LifeKeeper GUI – Overview

The LifeKeeper GUI uses Java technology to provide a graphical status interface to LifeKeeper and its configuration data. Since the LifeKeeper GUI is a client/server application, a user will run the graphical user interface on a client system in order to monitor or administer a server system where LifeKeeper is executing. The client and the server may or may not be the same system. The LifeKeeper GUI allows users working on any machine to administer, operate, or monitor servers and resources in any cluster, as long as they have the required group memberships on the cluster machines. For details, see [Configuring GUI Users]. The LifeKeeper GUI Server and Client components are described below.

GUI Server

The LifeKeeper GUI server is initialized on each server in a LifeKeeper cluster at system startup. It communicates with the LifeKeeper core software via the Java Native Interface (JNI), and with the LifeKeeper GUI client using Remote Method Invocation (RMI).

GUI Client

The LifeKeeper GUI client is designed to run either as an application on a Linux system, or as an applet which can be invoked from a web browser on either a Windows or Unix system.

The LifeKeeper GUI client includes the following graphical components:

- ◦ The status table on the upper left displays the high level status of connected servers and their resources.
- ◦ The properties panel on the upper right displays detailed information about the most recently selected status table object.
- ◦ The output panel on the bottom displays command output.
- ◦ The message bar at the very bottom of the window displays processing status messages.
- ◦ The server context and resource context toolbars (in the properties panel) and global toolbar provide fast access to frequently-used tasks.
- ◦ The server context and resource context menus (popup) and global menus (file, edit server, edit resource, view, and help) provide access to all tasks.

Right-clicking on a graphic resource, server, or table cell will display a context menu. Most tasks can also be initiated from these context menus, in which case the resources and servers will be automatically determined.

Starting GUI clients

Starting the LifeKeeper GUI Applet

To run the LifeKeeper GUI applet via the web open your favorite web browser and go to the URL http://<server name>:81 where <server name> is the name of a LifeKeeper server. This will load the LifeKeeper GUI applet from the LifeKeeper GUI server on that machine.

After it has finished loading, you should see the Cluster Connect dialog, which allows you to connect to any GUI server.

> **NOTE:** When you run the applet, if your system does not have the required Java Plug-in, you will be automatically taken to the web site for downloading the plug-in. You must also set your browser security parameters to enable Java.

If you have done this and the client still is not loading, see GUI Troubleshooting.

Starting the Application Client

Users with administrator privileges on a LifeKeeper server can run the application client from that server. To start the LifeKeeper GUI app run `/opt/LifeKeeper/bin/lkGUIapp` from a graphical window.

If you have done this and the client still is not loading, see GUI Troubleshooting.

Exiting GUI Clients

Select Exit from the File menu to disconnect from all servers and close the client.

5.3.4.1.1.4.2. Configuring the LifeKeeper GUI

Installing the LifeKeeper Server for GUI Administration

Perform the following steps for each LifeKeeper server. Each step contains references or links for more detailed instructions.

1. You must install the Java Runtime Environment (JRE) or Java Software Development Kit (JDK) on each server. See the SPS for Linux Release Notes for the required Java version.

2. Start the LifeKeeper GUI Server on each server (see Starting/Stopping the GUI Server). Note: Once the GUI Server has been started following an initial installation, starting and stopping LifeKeeper will start and stop all LifeKeeper daemon processes including the GUI Server.

3. If you plan to allow users other than root to use the GUI, then you need to Configure GUI Users.

Running the GUI

You can run the LifeKeeper GUI:

- ◦ on the LifeKeeper server in the cluster and/or
- ◦ on a remote system outside the cluster

See Running the GUI on the LifeKeeper Server for information on configuring and running the GUI on a server in your LifeKeeper cluster.

See Running the GUI on a Remote System for information on configuring and running the GUI on a remote system outside your LifeKeeper cluster.

GUI Configuration

Item	Description
GUI Client and Server Communication	The LifeKeeper GUI client and server use Java Remote Method Invocation (RMI) to communicate. For RMI to work correctly, the client and server must use resolvable hostnames or IP addresses. If DNS is not implemented (or names are not resolvable using other name lookup mechanisms), edit the /etc/hosts file on each client and server to include the names and addresses of all other LifeKeeper servers.
GUI Server Java Platform	The LifeKeeper GUI server requires that the Java Runtime Environment (JRE) – Java virtual machine, the Java platform core classes and supporting files – be installed.The LifeKeeper GUI supports OpenJDK. In the following environments, the setup script that is executed during installation installs OpenJDK that is included with the OS. If the Linux distributor does not provide OpenJDK, install the OpenJDK

	package included in the LifeKeeper installation image. See the Release Notes for supported OpenJDK versions. • RedHat Enterprise Linux/CentOS/Oracle Linux 6.6 or later • RedHat Enterprise Linux/CentOS/Oracle Linux 7.1 or later • RedHat Enterprise Linux/CentOS/Oracle Linux 8 or later • SUSE Linux Enterprise Server 12 or later (excluding SLES15 and SLES15 SP1) **Note:** When installing LifeKeeper, set the JRE path used by the GUI to LifeKeeper PATH default file */etc/default/LifeKeeper*. Edit this PATH if you want to change the JRE version. If LifeKeeper is running when you edit this file, you should stop and restart the LifeKeeper GUI server to reflect the change.
Uninstall Java Runtime Environment	• Environment where the OpenJDK package included with the LifeKeeper installation image is installed: The OpenJDK package will be uninstalled when uninstalling LifeKeeper. • Environment where OpenJDK provided by Linux distributor is installed: The OpenJDK package is not uninstalled when uninstalling LifeKeeper. If necessary, uninstall it manually.
Java Remote Object Registry Server Port	The LifeKeeper GUI server uses port 82 for the Java remote object registry on each LifeKeeper server. This should allow servers to support RMI calls from clients behind typical firewalls.
LifeKeeper Administration Web Server	The LifeKeeper GUI server requires an administration web server for client browser communication. Currently, the LifeKeeper GUI server is using a private copy of the lighttpd web server for its administration web server. This web server is installed and configured by the steeleye-lighttpd package and uses port 81 to avoid a conflict with other web servers.
GUI Client Network Access	LifeKeeper GUI clients require network access to all hosts in the LifeKeeper cluster. When running the LifeKeeper GUI client in a browser, you will have to lower the security level to allow network access for applets. Be careful not to visit other sites with security set to low values (e.g., change the security settings only for intranet or trusted sites).

GUI Limitations

Item	Description
GUI Interoperability Restriction	The LifeKeeper for Linux client may only be used to administer LifeKeeper on Linux servers. The LifeKeeper for Linux GUI will not interoperate with LifeKeeper for Windows.

5.3.4.1.1.4.3. Starting and Stopping the GUI Server

To Start the LifeKeeper GUI Server

If the LifeKeeper GUI Server is not running, type the following command as root:

```
/opt/LifeKeeper/bin/lkGUIserver start
```

This command starts all LifeKeeper GUI Server daemon processes on the server being administered if they are not currently running. A message similar to the following is displayed.

```
# Installing GUI Log
# LK GUI Server Startup at:
# Mon May 8 14:14:46 EDT 2006
# LifeKeeper GUI Server Startup completed at:
# Mon May 8 14:14:46 EDT 2006
```

Once the LifeKeeper GUI Server is started, all subsequent starts of LifeKeeper will automatically start the LifeKeeper GUI Server processes.

Troubleshooting

The LifeKeeper GUI uses Ports 81 and 82 on each server for its administration web server and Java remote object registry, respectively. If another application is using the same ports, the LifeKeeper GUI will not function properly. These values may be changed by editing the following entries in the LifeKeeper default file ***/etc/default/LifeKeeper***.

```
GUI_WEB_PORT=81 GUI_RMI_PORT=82
```

> **Note:** These port values are initialized in the GUI server at start time. If you alter them, you will need to stop and restart the steeleye-lighttpd process. These values must be the same across all clusters to which you connect.

To Stop the LifeKeeper GUI Server

If the LifeKeeper GUI Server is running, type the following command as *root*:

```
/opt/LifeKeeper/bin/lkGUIserver stop
```

This command halts all LifeKeeper GUI Server daemon processes on the server being administered if they are currently running. The following messages are displayed.

```
# LifeKeeper GUI Server Shutdown at:
# Fri May 19 15:37:27 EDT 2006
# LifeKeeper GUI Server Shutdown Completed at:
# Fri May 19 15:37:28 EDT 2006
```

LifeKeeper GUI Server Processes

To verify that the LifeKeeper GUI Server is running, type the following command:

```
ps -ef | grep runGuiSer
```

You should see output similar to the following:

```
root 2805 1 0 08:24 ? 00:00:00 sh/opt/LifeKeeper/bin/runGuiSer
```

To see a list of the other GUI Server daemon processes currently running, type the following command:

```
ps -ef | grep S_LK
```

You should see output similar to the following:

```
root 30228 30145 0 11:20 ? 00:00:00 java -Xint -Xss3M
-DS_LK=true -Djava.rmi.server.hostname=thor48 …
```

5.3.4.1.1.4.4. Java Security Policy

The LifeKeeper GUI uses policy-based access control. When the GUI client is loaded, it is assigned permissions based on the security policy currently in effect. The policy, which specifies permissions that are available for code from various signers/locations, is initialized from an externally configurable policy file.

There is, by default, a single system-wide policy file and an optional user policy file. The system policy file, which is meant to grant system-wide code permissions, is loaded first, and then the user policy file is added to it. In addition to these policy files, the LifeKeeper GUI policy file may also be loaded if the LifeKeeper GUI is invoked as an application.

Location of Policy Files

The system policy file is by default at:

<JAVA.HOME>/lib/security/java.policy (Linux)

<JAVA.HOME>\lib\security\java.policy (Windows)

Note: JAVA.HOME refers to the value of the system property named “JAVA.HOME”, which specifies the directory into which the JRE or JDK was installed.

The user policy file starts with `.` and is by default at:

<USER.HOME>\.java.policy

Note: USER.HOME refers to the value of the system property named “user.home”, which specifies the user’s home directory. For example, the home directory on a Windows NT workstation for a user named Paul might be “paul.000”.

For Windows systems, the user.home property value defaults to:

C:\WINNT\Profiles\<USER> (on multi-user Windows NT systems)

C:\WINDOWS\Profiles\<USER> (on multi-user Windows 95/98 systems)

C:\WINDOWS (on single-user Windows 95/98 systems)

The LifeKeeper GUI policy file is by default at:

/opt/LifeKeeper/htdoc/java.policy (Linux)

Policy File Creation and Management

By default, the LifeKeeper GUI policy file is used when the LifeKeeper GUI is invoked as an application. If you are running the LifeKeeper GUI as an applet, you will need to create a user policy file in your home directory if one does not already exist. The user policy file should specify the minimum permissions required to run the LifeKeeper GUI, which are provided in the "Sample Policy File" section later in this topic.

A policy file can be created and maintained via a simple text editor, or via the graphical Policy Tool utility included with the Java Runtime Environment (JRE) or Java Development Kit (JDK). Using the Policy Tool saves typing and eliminates the need for you to know the required syntax of policy files. For information about using the Policy Tool, see the Policy Tool documentation at http://docs.oracle.com/javase/8/docs/technotes/tools/.

The **simplest way to create a user policy file** with the minimum permissions required to run the LifeKeeper GUI is to copy the LifeKeeper GUI policy file located in ***/opt/LifeKeeper/htdoc/java.policy*** to your home directory and rename it *.java.policy* (note the leading dot before the filename which is required). On a Windows system, you can copy the LifeKeeper GUI policy file by opening the file *http://<server name*>*:81/java.policy* (where <server name> is the host name of a LifeKeeper server) and saving it as *.java.policy* in your home directory. If you need to determine the correct location for a user policy file, enable the Java Console using the Java Control Panel and start the LifeKeeper GUI as an applet. The home directory path for the user policy file will be displayed in the Java Console.

Granting Permissions in Policy Files

A permission represents access to a system resource. In order for a resource access to be allowed for an applet, the corresponding permission must be explicitly granted to the code attempting the access. A permission typically has a name (referred to as a "target name") and, in some cases, a comma-separated list of one or more actions. For example, the following code creates a FilePermission object representing read access to the file named abc in the */tmp* directory:

```
perm = new java.io.FilePermission("/tmp/abc","read");
```

In this, the target name is "*/tmp/abc*" and the action string is "*read*".

A policy file specifies what permissions are allowed for code from specified code sources. An example policy file entry granting code from the /home/sysadmin directory read access to the file */tmp/abc* is:

```
grant codeBase "file:/home/sysadmin/" { permissionjava.io.FilePermission "/tmp/abc",
"read"; };
```

Sample Policy File

The following sample policy file includes the minimum permissions required to run the LifeKeeper GUI. This policy file is installed in ***/opt/LifeKeeper/htdoc/java.policy*** by the LifeKeeper GUI package.

```
/*
* Permissions needed by the LifeKeeper GUI. You may want to
* restrict this by codebase. However, if you do this, remember
* that the recovery kits can have an arbitrary jar component ** with an
arbitrary codebase, so you'll need to alter the grant
* to cover these as well.
*/
grant {

/*
* Need to be able to do this to all machines in the
* LifeKeeper cluster. You may restrict the network
* specification accordingly.
*/
permission java.net.SocketPermission"*", "accept,connect,resolve";
/*
* We use URLClassLoaders to get remote properties files and
* jar pieces.
*/
permission java.lang.RuntimePermission"createClassLoader";
/*
* The following are needed only for the GUI to run as an
* application (the default RMI security manager is more
* restrictive than the one a browser installs for its
* applets.
*/
permission java.util.PropertyPermission "*","read";
permission java.awt.AWTPermission "*";
permission java.io.FilePermission "<<ALL FILES>>","read,execute";

};
```

5.3.4.1.1.4.5. Java Plug-In

Regardless of the browser you are using (see supported browsers), the first time your browser attempts to load the LifeKeeper GUI, it will either automatically download the Java Plug-In software or redirect you to a web page to download and install it. From that point forward, the browser will automatically invoke the Java Plug-in software every time it comes across web pages that support the technology.

Downloading the Java Plug-in

Java Plug-in software is included as part of the Java Runtime Environment (JRE) for Solaris, Linux and Windows. Downloading the JRE typically takes a total of three to ten minutes, depending on your network and system configuration size. The download web page provides more documentation and installation instructions for the JRE and Java Plug-in software.

Note: You should close and restart your browser after installing the plug-in and whenever plug-in properties are changed.

Note: Only Java Plug-in version 8 update 51 is supported with LifeKeeper.

5.3.4.1.1.4.6. Running the GUI on a Remote System

You may administer LifeKeeper from a Linux, Unix or Windows system outside the LifeKeeper cluster by running the LifeKeeper GUI as a Java applet. Configuring and running the GUI in this environment is described below.

Configuring the GUI on a Remote System

In order to run the LifeKeeper GUI on a remote Linux, Unix or Windows system, your browser must provide full JDK 1.8 (x64) applet support. Refer to the SPS for Linux Release Notes for information on the supported platforms and browsers for the LifeKeeper GUI.

1. If you are running the LifeKeeper GUI as an applet, you need to create a user policy file in your home directory if one does not already exist. The user policy file should specify the minimum permissions required to run the LifeKeeper GUI.

- ◦ The simplest way to create a user policy file with the minimum permissions required to run the LifeKeeper GUI is to copy the LifeKeeper GUI policy file located in */opt/LifeKeeper/ htdoc/java.policy* to your home directory and rename it *.java.policy* (note there is a leading dot in the file name that is required). On a Windows system, you can copy the LifeKeeper GUI policy file by opening the file *http:///:81/java.policy* (where <servername> is the host name of a LifeKeeper server), and saving it as *.java.policy* in your home directory. If you need to determine the correct location for a user policy file, enable the **Java Console** using the **Java Control Panel**, and start the LifeKeeper GUI as an applet. The home directory path for the user policy file will be displayed in the Java Console.

- ◦ If you already have a user policy file, you can add the required entries specified in */opt/ LifeKeeper/ htdoc/java.policy* on a LifeKeeper server into the existing file using a simple text editor. See Java Security Policy for further information.

2. You must set your browser security parameters to **low**. This generally includes enabling of Java and Java applets. Since there are several different browsers and versions, the instructions for setting browser security parameters are covered in Setting Browser Security Parameters for the GUI Applet.

   Note: It is important to use caution in visiting external sites with low security settings.

3. When you run the GUI for the first time, if you are using **Netscape** or **Internet Explorer** and your system does not have the required Java plug-in, you may be automatically taken to the appropriate web site for downloading the plug-in. See the SPS for Linux Release Notes for the required Java Plug-in version and URL to access the download.

Running the GUI on a Remote System

After you have completed the tasks described above, you are ready to run the LifeKeeper GUI as a Java applet on a remote system.

1. Open the URL, http://<server name>:81, for the LifeKeeper GUI webpage (where <server name> is the name of the LifeKeeper server). The web page contains the LifeKeeper splash screen and applet. When the web page is opened, the following actions take place:

- ◦ the splash screen is displayed
- ◦ the applet is loaded
- ◦ the Java Virtual Machine is started
- ◦ some server files are downloaded
- ◦ the applet is initialized

   Depending on your network and system configuration, these actions may take up to 20 seconds. Typically, browsers provide some minimal status as the applet is loading and initializing.

   If everything loads properly, a **Start** button should appear in the applet area. If the splash screen does not display a **Start** button or you suspect that the applet failed to load and initialize, refer to Applet Troubleshooting or see Network-Related Troubleshooting.

2. When prompted, click **Start**. The LifeKeeper GUI appears and the Cluster Connect Dialog is automatically displayed. Once a Server has been entered and connection to the cluster established, the GUI window displays a visual representation and status of the resources protected by the connected servers. The GUI menus and toolbar buttons provide LifeKeeper administration functions.

   Note: Some browsers add **"Warning: Applet Window"** to windows and dialogs created by an applet. This is normal and can be ignored.

Applet Troubleshooting

If you suspect that the applet failed to load and initialize, try the following:

1. Verify that applet failed. Usually a message is printed somewhere in the browser window specifying the state of the applet. In **Netscape** and **Internet Explorer**, an icon may appear instead of the applet in addition to some text status. Clicking this icon may bring up a description of the failure.

2. Verify that you have installed the Java Plug-in. If your problem appears to be Java Plug-in related, refer to the Java Plug-in topic.

3. Verify that you have met the browser configuration requirements, especially the security settings. Refer to Setting Browser Security Parameters for the GUI Applet for more information. If you don't find anything obviously wrong with your configuration, continue with the next steps.

4. Open the **Java Console**.

- ◦ For **Firefox**, **Netscape** and older versions of **Internet Explorer**, run the **Java Plug-In** applet from your machine's **Control Panel** and select the option to show the console, then restart your browser.

- ◦ For recent versions of **Internet Explorer**, select Tools > Java Console. If you do not see the Java Console menu item, select Tools > Manage Add-Ons and enable the console, after which you may need to restart your browser before the console will appear.

- ◦ For **Mozilla**, select Tools > Web Development > Java Console.

5. Reopen the URL, *http://<server name>:81* to start the GUI applet. If you've modified the **Java Plug-In Control Panel**, restart your browser.

6. Check the console for any messages. The messages should help you resolve the problem. If the problem appears to be network related, refer to Network-Related Troubleshooting.

5.3.4.1.1.4.7. Running the GUI on a LifeKeeper Server

The simplest way to run the LifeKeeper GUI is as an application on a LifeKeeper server. By doing so you are, in effect, running the GUI client and server on the same system.

1. After configuring the LifeKeeper server for GUI Administration, you can run the GUI as an application on the server by entering the following command as root:

   ```
   /opt/LifeKeeper/bin/lkGUIapp
   ```

2. The lkGUIapp script sets the appropriate environment variables and starts the application. As the application is loading, an application identity dialog or splash screen for LifeKeeper appears.

3. After the application is loaded, the LifeKeeper GUI appears and the Cluster Connect dialog is automatically displayed. Enter the Server Name you wish to connect to, followed by the login and password.

4. Once a connection to the cluster is established, the GUI window displays a visual representation and status of the resources protected by the connected servers. The GUI menus and toolbar buttons provide administration functions.

5.3.4.1.1.4.8. Browser Security Parameters for GUI Applet

> **WARNING:** Be careful of other sites you visit with security set to low values.

Firefox

1. From the **Edit** menu, select **Preferences**.
2. In the **Preferences** dialog box, select **Content**.
3. Select the **Enable Java** and **Enable Java Script** options.
4. Click **Close**.

Internet Explorer

The most secure method for using Internet Explorer is to add the LifeKeeper server to the **Trusted Sites** zone as follows:

1. From the **Tools** menu, click **Internet Options**.
2. Click the **Security** tab.
3. Select **Trusted Sites** zone and click **Custom Level**.
4. Under **Reset custom settings**, select **Medium/Low**, then click **Reset**.
5. Click **Sites**.
6. Enter the server name and port number for the LifeKeeper server(s) to which you wish to connect (for instance: http://server1:81).

An alternative, but possibly less secure method, is to do the following:

1. From the **Tools** menu, click **Internet Options**.
2. Select either **Internet** or **Local Intranet** (depending upon whether your remote system and the LifeKeeper cluster are on the same intranet).
3. Adjust the **Security Level** bar to **Medium** (for Internet) or **Medium-low** (for Local Intranet). These are the default settings for each zone.
4. Click **OK**.

5.3.4.1.2. Status Table

The status table provides a visual representation of the status of connected servers and their resources. It shows:

- ◦ the state of each server in the top row
- ◦ the global (cross-server) state and the parent-child relationships of each resource in the left-most column
- ◦ the state of each resource on each server in the remaining cells

The states of the servers and resources are shown using graphics, text and color. An empty table cell under a server indicates that a particular resource has not been defined on that server.

If you select a server or a resource instance in the status table, detailed state information and a context-sensitive toolbar for that item are shown in the properties panel. You can also pop up the appropriate server context menu or resource context menu for any item by right-clicking on that cell.

The status table is split into two sections. The relative sizes of the left and right sections can be modified by moving the divider between them. The status table can also be collapsed to show only the highest level items in the hierarchy trees. Collapsing or expanding resource items in the tree causes the hierarchies listed in the table to also expand and collapse.

5.3.4.1.3. Properties Panel

The properties panel displays the properties of the server or resource that is selected in the status table. The properties panel has the same functionality as the server properties dialog or the resource properties dialog, plus a context-sensitive toolbar to provide fast access to commonly used commands. The caption at the top of this panel is ***server_name*** if a server is selected, or ***server_name: resource_name*** if a resource is selected.

The context-sensitive toolbars displayed in the properties panel are the server context toolbar and the resource context toolbar. Server or resource toolbars may also be customized. For more information on customized toolbars, see the corresponding application recovery kit documentation.

The buttons at the bottom of the properties panel function as follows:

- The

Apply button applies any changes that have been made to editable properties on the panel. This button is only enabled if you have changed an editable property.

- The

Reset button queries the server for the current values of all properties, clearing any changes that you may have made. This button is always enabled.

- The

Help button displays context-sensitive help for the properties panel. This button is always enabled.

You increase or decrease the size of the properties panel by sliding the separator at the left of the panel to the left or right. If you want to open or close this panel, use the **Properties Panel checkbox** on the View Menu.

5.3.4.1.4. Output Panel

The output panel collects output from commands issued by the LifeKeeper GUI client. When a command begins to run, a time stamped label is added to the output panel, and all of the output from that command is added under this label. If you are running multiple commands at the same time (typically on different servers), the output from each command is sent to the corresponding section making it easy to see the results of each.

You increase or decrease the size of the output panel by sliding the separator at the top of the panel up or down. If you want to open or close this panel, use the **Output Panel checkbox** on the View Menu. When the output panel is closed, the dialog that initiates each command will stay up, the output will be displayed on that dialog until you dismiss it and you will not be able to review the output from any command after you have closed that dialog. After the output panel is reopened, the LifeKeeper GUI will return to its default behavior.

5.3.4.1.5. Message Bar

The message bar appears beneath the status window. It is used for displaying messages in a single text line. Message such as “Connecting to Server X” or “Failure to connect to Server X” might be displayed.

To hide the message bar, clear the **Message Bar** checkbox in the View Menu.

To display the message bar, select the **Message Bar** checkbox in the View Menu.

To see a history of messages displayed in the message bar, see Viewing Message History.

5.3.4.1.6. Exiting the GUI

Select **Exit** from the File Menu to disconnect from all servers and close the GUI window.

5.3.4.1.7. Common Tasks

The following are basic tasks that can be performed by any user.

5.3.4.1.7.1. Starting LifeKeeper

All SPS software is installed in the directory */opt/LifeKeeper*.

When you have completed all of the verification tasks, you are ready to start LifeKeeper on both servers. This section provides information for starting the LifeKeeper server daemon processes. The LifeKeeper GUI application is launched using a separate command and is described in Configuring the LifeKeeper GUI. LifeKeeper provides a command line interface that starts and stops the LifeKeeper daemon processes. These daemon processes must be running before you start the LifeKeeper GUI.

Starting LifeKeeper Server Processes

If LifeKeeper is not currently running on your system, type the following command as the user root on all servers:

```
/opt/LifeKeeper/bin/lkstart
```

When executing this command, the LifeKeeper service will be started and LifeKeeper will be set to start automatically at system startup.

Following the delay of a few seconds, an informational message is displayed.

> **Note:** If you receive an error message referencing the **LifeKeeper Distribution Enabling Package** when you start LifeKeeper, you should install / re-install the LifeKeeper Installation Image File.

See the LCD help page by entering man LCD at the command line for details on the lkstart command.

To start only the LifeKeeper process without enabling automatic startup, execute the following command:

```
service lifekeeper start (or, systemctl start lifekeeper)
```

Enabling Automatic LifeKeeper Restart

While the above command will start LifeKeeper, it will need to be performed each time the system is re-booted. If you would like LifeKeeper to start automatically when server boots up, type the following command:

```
chkconfig lifekeeper on (or, systemctl enable lifekeeper)
```

See the chkconfig man page for further information.

5.3.4.1.7.2. Stopping LifeKeeper

If you need to stop LifeKeeper, type the following command as root to stop it:

```
/etc/init.d/lifekeeper stop-nofailover
```

or

```
LKSTOP_MODE=stop-nofailover systemctl stop lifekeeper
```

This command will shut down LifeKeeper on the local system if it is currently running. It will first remove all protected resources from service on the local system then shut down the LifeKeeper daemons. Protected resources will not fail over to another system in the cluster. LifeKeeper will automatically restart when the system is restarted.

```
/etc/init.d/lifekeeper stop-daemons
```

or

```
LKSTOP_MODE=stop-daemons systemctl stop lifekeeper
```

This command will skip the section that removes resources from service. The resources will remain running on the local system but will no longer be protected by LifeKeeper. This command should be used with caution, because if resources are not gracefully shut down, then items such as SCSI locks will not be removed. If the system on which this command is executed subsequently fails or is shut down, the system(s) will NOT initiate failover of the appropriate resources. LifeKeeper will automatically restart when the system is restarted.

```
/etc/init.d/lifekeeper stop
```

or

```
systemctl stop lifekeeper
```

This command will remove the resources from service but does not set the !nofailover! flag (see LCDIflag(1M)) on any of the systems that it can communicate with. This means that failover will occur if the shutdown_switchover flag is set. If shutdown_switchover is not set, then this command behaves the same as /etc/init.d/lifekeeper stop-nofailover. LifeKeeper will automatically restart when the system is restarted.

```
/etc/init.d/lifekeeper stop-failover
```

or

```
LKSTOP_MODE=stop-failover systemctl stop lifekeeper
```

This command will remove resources from service and initiate failover. It will behave the same as /etc/init.d/lifekeeper stop when the shutdown_switchover flag is set. LifeKeeper will automatically restart when the system is restarted.

Note: Alternatively, you may stop LifeKeeper using:

- **/opt/LifeKeeper/bin/lkstop**
  This command will shut down LifeKeeper on the local system if it is currently running. It will first remove all protected resources from service on the local system then shut down the LifeKeeper daemons.

- **/opt/LifeKeeper/bin/lkstop -f**
  This command will skip the section that removes resources from service. The resources will remain running on the local system but will no longer be protected by LifeKeeper.

Disabling Automatic LifeKeeper Restart

If you do not want LifeKeeper to automatically restart when the system is restarted, type the following command:

```
chkconfig lifekeeper off
```

or

```
systemctl disable lifekeeper
```

See the chkconfig (or systemctl) man page for further information.

5.3.4.1.7.3. Viewing LifeKeeper Processes

To see a list of all LifeKeeper core daemon processes currently running, type the following command:

```
ps -ef | grep LifeKeeper | grep -w bin | grep -v lklogmsg
```

An example of the output is provided below:

```
root 11663 11662 0 14:03 pts/0 00:00:00 /bin/bash /etc/redhat-lsb/
lsb_start_daemon /opt/LifeKeeper/sbin/runsvdir -P /opt/LifeKeeper/etc/
service log: runit just
started.....

root 11666 11663 0 14:03 pts/0 00:00:00 /bin/bash -c ulimit -S -c 0
>/dev/null 2> &1 ; /opt/LifeKeeper/sbin/runsvdir -P /opt/LifeKeeper/etc/
service log: runit just
started.....

root 11880 11873 0 14:03 ? 00:00:00 /opt/LifeKeeper/bin/lk_logmgr -l/opt/
LifeKeeper/out -d/etc/default/LifeKeeper

root 12240 11877 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lcm

root 12247 11879 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/ttymonlcm

root 12250 11876 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lcd

root 12307 11874 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lkcheck

root 12311 11875 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lkscsid

root 12325 11871 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lkvmhad

root 12335 12330 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/perl /opt/
LifeKeeper/htdoc/cgi-bin/DoRequest.fcgi
```

The run state of LifeKeeper can be determined via the following command:

```
/opt/LifeKeeper/bin/lktest
```

If LifeKeeper is running it will output something similar to the following:

```
F S UID PID PPID C CLS PRI NI SZ STIME TIME CMD

4 S root 12240 11877 0 TS 39 -20 6209 14:04 00:00:00 lcm
```

```
4 S root 12247 11879 0 TS 39 -20 30643 14:04 00:00:00 ttymonlcm

4 S root 12250 11876 0 TS 29 -10 9575 14:04 00:00:00 lcd
```

If LifeKeeper is not running, then nothing is output and the command exists with a 1.

Note: There are additional LifeKeeper processes running that start, stop, and monitor the LifeKeeper core daemon processed along with those required for the Graphical User Interface (GUI). See Viewing LifeKeeper Controlling Processes and Viewing LifeKeeper GUI Server Processes for a list of the processes. Additionally, most LifeKeeper processes have a child lklogmsg to capture and log any unexpected output.

5.3.4.1.7.4. Viewing LifeKeeper GUI Server Processes

To verify that the LifeKeeper GUI Server is running, type the following command:

```
ps -ef | grep runGuiSer
```

You should see output similar to the following:

```
root 2805 1 0 08:24 ? 00:00:00 sh /opt/LifeKeeper/bin/runGuiServer
```

To see a list of the other GUI Server daemon processes currently running, type the following command:

```
ps -efw | grep S_LK
```

You should see output similar to the following:

```
root 819 764 0 Oct16 ? 00:00:00 java -Xint -Xss3M -DS_LK=true
-Djava.rmi.server.hostname=wake -Dcom.steeleye.LifeKeeper.rmiPort=82
-Dcom.steeleye.LifeKeeper.LKROOT=/opt/LifeKeeper
-DGUI_RMI_REGISTRY=internal -DGUI_WEB_PORT=81
com.steeleye.LifeKeeper.beans.S_LK
```

To verify that the LifeKeeper GUI Server Administration Web Server is running type the following command:

```
ps -ef|grep steeleye-light | egrep -v "lklogmsg|runsv"
```

You should see output similar to the following:

```
root 12330 11872 0 14:04 ? 00:00:00 /opt/LifeKeeper/sbin/steeleye-
lighttpd -D -f/opt/LifeKeeper/etc/lighttpd/lighttpd.conf
```

5.3.4.1.7.5. Viewing LifeKeeper Controlling Processes

To verify that the LifeKeeper controlling processes are running, type the following command:

```
ps -ef | grep runsv
```

You should see output similar to the following:

```
root 11663 11662 0 14:03 pts/0 00:00:00 /bin/bash /etc/redhat-lsb/
lsb_start_daemon /opt/LifeKeeper/sbin/runsvdir -P /opt/LifeKeeper/etc/
service log: runit just
started.....

root 11666 11663 0 14:03 pts/0 00:00:00 /bin/bash -c ulimit -S -c 0
>/dev/null 2>&1 ; /opt/LifeKeeper/sbin/runsvdir -P /opt/LifeKeeper/etc/
service log: runit just
started.....

root 11667 11666 0 14:03 pts/0 00:00:00 /opt/LifeKeeper/sbin/runsvdir -P
/opt/LifeKeeper/etc/service log: runit just
started.....

root 11871 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lkvmhad

root 11872 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv steeleye-
lighttpd

root 11873 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lk_logmgr

root 11874 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lkcheck

root 11875 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lkscsid

root 11876 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lcd

root 11877 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lcm

root 11878 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv
lkguiserver

root 11879 11667 0 14:03 ? 00:00:00 /opt/LifeKeeper/sbin/runsv ttymonlcm
```

These processes start, stop, and monitor LifeKeeper core daemon processes and must be running to

start LifeKeeper. These processes are configured by default to start when the system boots and this behavior should not be altered.

5.3.4.1.7.6. Connecting Servers to a Cluster

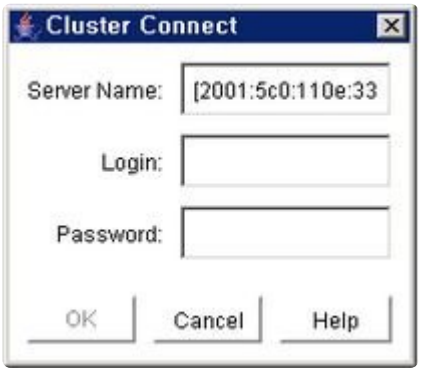
1. There are two possible ways to begin.

- ◦ On the Global Toolbar, click the **Connect** button.

- ◦ On the File Menu, click **Connect**.

2. In the **Server Name** field of the Cluster Connect dialog, enter the name of a server within the cluster to which you want to connect.

> **Note:** If using an **IPv6** address, this address will need to be enclosed in brackets [ ]. This will allow a connection to be established through a machine's IPv6 address. Alternatively, a name can be assigned to the address, and that name can then be used to connect.



3. In the **Login** and **Password** fields, enter the login name and password of a user with LifeKeeper authorization on the specified server.

4. Click **OK**.

If the GUI successfully connects to the specified server, it will continue to connect to (and add to the status display) all known servers in the cluster until no new servers are found.

> **Note:** If the initial login name and password fails to authenticate the client on a server in the cluster, the user is prompted to enter another login name and password for that server. If "**Cancel**" is selected from the Password dialog, connection to that server is aborted and the GUI continues connecting to the rest of the cluster.

5.3.4.1.7.7. Disconnecting from a Cluster

This task disconnects your GUI client from all servers in the cluster, and it does so through the server you select.

1. There are three possible ways to begin.
   - On the Global Toolbar, click the **Disconnect** button.
   - On the Edit Menu, select **Server** and then click **Disconnect**.
   - On the Server Context Toolbar, if displayed, click the **Disconnect** button.
2. In the **Select Server in Cluster** list of the Cluster Disconnect Dialog, select the name of a server in the cluster from which you want to disconnect.
3. Click **OK**. A **Confirmation** dialog listing all the servers in the cluster is displayed.
4. Click **OK** in the **Confirmation** dialog to confirm that you want to disconnect from all servers in the cluster.

After disconnecting from the cluster, all servers in that cluster are removed from the GUI status display.

5.3.4.1.7.8. Viewing Connected Servers

The state of a server can be determined by looking at the graphic representation of the server in the table's header as shown below. See Viewing the Status of a Server for an explanation of the server states indicated visually by the server icon.

5.3.4.1.7.9. Viewing the Status of a Server

The state of a server can be determined by looking at the graphic representation of the server in the table's header as shown below.

Server State	Visual state	What it Means
ALIVE		Client has valid connection to the server. Comm paths originating from this server to an ALIVE remote server are ALIVE. Comm paths which may be marked DEAD and which target a DEAD server are ignored because the DEAD server will be reflected in its own graphic.
ALIVE		Client has valid connection to the server. One or more comm paths from this server to a given remote server are marked as DEAD. No redundant comm path exists from this server to a given remote server.
DEAD		Reported as DEAD by other servers in the cluster.
UNKNOWN		Network connection was lost. Last known LifeKeeper state is ALIVE.

5.3.4.1.7.10. Viewing Server Properties

1. There are two possible ways to begin.

- ◦ Right-click on the icon for the server for which you want to view the properties. When the Server Context Menu appears, click **Properties**. Server properties will also be displayed in the Properties Panel if it is enabled when clicking on the server.

- ◦ On the Edit Menu, point to **Server** and then click **Properties**. When the dialog comes up, select the server for which you want to view the properties from the Server list.

2. If you want to view properties for a different server, select that server from the dialog's **Server** list.

3. When you are finished, click **OK** to close the window.

5.3.4.1.7.11. Viewing Server Log Files

1. There are four ways to begin.

- ◦ Right-click on a server icon to display the Server Context Menu, then click **View Log** to bring up the LifeKeeper Log Viewer Dialog.

- ◦ On the Global Toolbar, click the **View Log** button, then select the server that you want to view from the Server list in the LifeKeeper Log Viewer Dialog.

- ◦ On the Server Context Toolbar, if displayed, click the **View Log** button.

- ◦ On the Edit Menu, point to **Server**, click **View Log**, then select the server that you want to view from the Server list in the **LifeKeeper Log Viewer Dialog**.

2. If you started from the **Global Toolbar** or the **Edit Menu** and you want to view logs for a different server, select that server from the **Server** list in the LifeKeeper Log Viewer Dialog. This feature is not available if you selected **View Logs** from the **Server Context Menu** or **Server Context Toolbar**.

3. When you are finished, click **OK** to close the **Log Viewer** dialog.

5.3.4.1.7.12. Viewing Resource Tags and IDs

A resource's tag and ID can be viewed quickly by positioning the cursor over a resource icon in the status window and clicking the left mouse button once (single-click). The resource tag and ID of the server having the lowest priority number are displayed in the message bar. To display the resource tag and ID for a resource on a specific server, single-click the appropriate resource instance cell in the table.

Messages displayed in the message bar look similar to the following:

```
Resource Tag = ipdnet0-153.98.87.73, Resource ID = IP-153.98.87.73
```

Under certain circumstances, the GUI may not be able to determine the resource ID, in which case only the resource tag is displayed in the message bar.

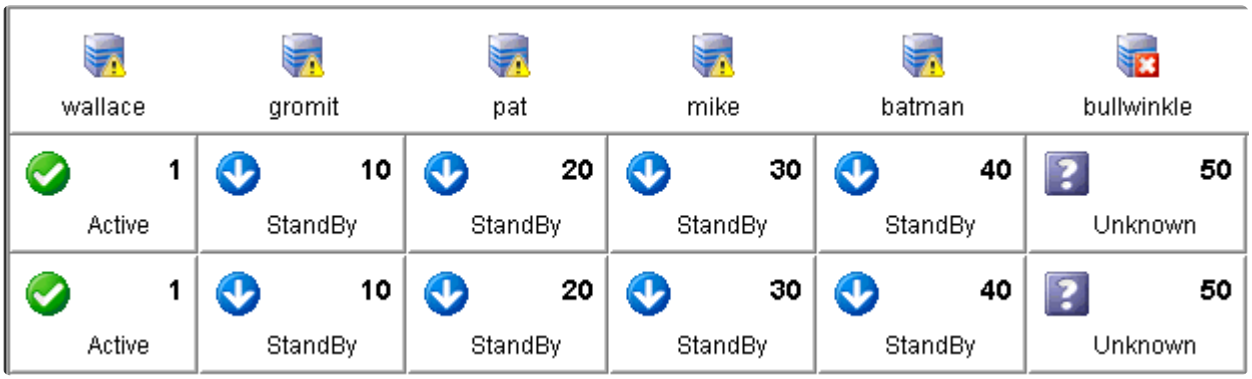
5.3.4.1.7.13. Viewing the Status of Resources

The status or state of a resource is displayed in two formats: **Global Resource Status** (across all servers), and the **Server Resource Status** (on a single server). The global resource status is shown in the **Resource Hierarchy Tree** in the left pane of the status window. The server resource status is found in the table cell where the resource row intersects with the server column.

Server Resource Status

The following figure shows servers with resource statuses of active, standby and unknown.

- ◦ All resources on “wallace” are active
- ◦ All resources on “gromit”, “pat”,“mike” and “batman” are standby
- ◦ All resources on “bullwinkle” are unknown



Server Resource State	Visual State	What it Means
ALIVE		Resource is operational on this server and protected. (ISP)
Degraded		Resource is operational on this server, but not protected by a backup resource. (ISU)
Standby		Server can take over operation of the resource. (OSU)
Failed		Problem with resource detected on this server. For example, an attempt to bring the resource in-service failed. (OSF)
Unknown		Resource is operational on this server, but not protected by a backup resource. (ISU)
	Empty panel	Server does not have the resource defined.

Global Resource Status

Visual State	Description	What it Means / Causes
ALIVE 	Normal	Resource is active (ISP) and all backups are active.
	Warning	Resource is active (ISP). One or more backups are marked as unknown or failed (OSF)
	Failed. Resource is not active on any servers (OSF).	Resource has been taken out-of-service for normal reasons. Resource has stopped running by unconventional means. Recovery has not been completed or has failed.
	Unknown. Could not determine state from available information.	More than one server is claiming to be active. Lost connection to server. All server resource instances are in an unknown state.

5.3.4.1.7.14. Viewing Resource Properties

1. There are three possible ways to begin.

   - º Right-click on the icon for the resource/server combination for which you want to view the properties. When the

Resource Context Menu appears, click **Properties**. Resource properties will also be displayed in the Properties Panel if it is enabled.

   - º Right-click on the icon for the global resource for which you want to view the properties. When the

Resource Context Menu appears, click **Properties**. When the dialog comes up, select the server for which you want to view that resource from the **Server** list.

   - º On the

Edit Menu, point to **Resource** and then click **Properties**. When the dialog comes up, select the resource for which you want to view properties from the **Resource** list, and the server for which you want to view that resource from the **Server** list.

2. If you want to view properties for a different resource, select that resource from the **Resource** list.

3. If you want to view resource properties for a different server, select that server from the **Server** list.

4. When you are finished, click **OK** to close the window.

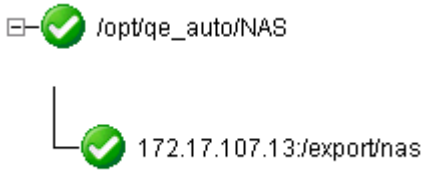
5.3.4.1.7.15. Resource Labels

This option group allows you to specify whether resources are viewed in the resource hierarchy tree by their tag name or ID.

> **Note:** The resource tag/ID shown in the resource hierarchy tree belongs to the server having the lowest priority number. If you wish to see the tag/ID for a resource on a specific server, left-click the resource instance cell in the table and its tag/ID will be displayed in the message bar.

By tag name:

By ID:



5.3.4.1.7.16. Viewing Message History

1. On the View Menu, click **History**. The LifeKeeper GUI Message **History** dialog is displayed.
2. If you want to clear all messages from the history, click **Clear**.
3. Click **OK** to close the dialog.

The **Message History** dialog displays the most recent messages from the message bar. The history list can display a maximum of 1000 lines. When the maximum number of lines is exceeded, the new messages will “push out” the oldest messages.

These messages represent only the actions between the client and the server and are displayed in chronological order, the most recent messages appearing at the top of the list.

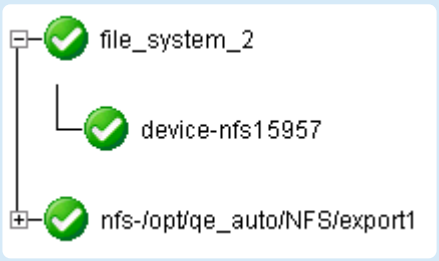
Reading the Message History

```
<– indicates that the message is incoming from a server and typically has a format of:

     <–"server name":"action"

     <–"server name":"app res": "action"

     <–"server name":"res instance":"action"

–> indicates that the message is outgoing from a client and typically has a format of:

     –>"server name":"action"

     –>"server name":"app res": "action"

     –>"server name":"res instance":"action"
```

The **Clear** button clears the history but does not close the dialog.

The **OK** button closes the dialog without clearing the history.

5.3.4.1.7.17. Expanding and Collapsing a Resource Hierarchy Tree

	In this segment of the tree, the resource *file_system_2* is expanded and the resource *nfs-/opt/qe_auto/NFS/export1* is collapsed. ⊟ appears to the left of a resource icon if it is expanded. ⊞ appears if it is collapsed.

To **expand** a resource hierarchy tree,

- ◦ Click the ⊞ or
  ◦
  ◦ Double-click the resource icon to the right of a ⊞ .

To **expand all** resource hierarchy trees,

- ◦ On the **View Menu**, click **Expand Tree** or
  ◦
  ◦ Double-click the **Resource Hierarchy Tree** button in the column header in the left pane of the **Status** window.

Note: The resource tag/ID shown in the resource hierarchy tree belongs to the server having the lowest priority number. If you wish to see the tag/ID for a resource on a specific server, left-click the resource instance cell in the table and its tag/ID will be displayed in the message bar.

To **collapse** a resource hierarchy tree,

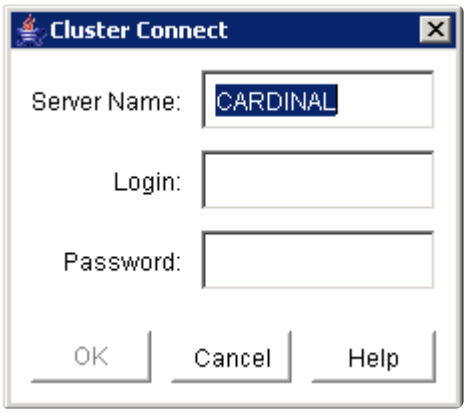
- ◦ click the ⊟ or
  ◦
  ◦ double-click the resource icon to the right of a ⊟ .

To **collapse all** resource hierarchy trees,

- ◦ On the **View Menu**, click **Collapse Tree** or
  ◦
  ◦ Double-click the **Resource Hierarchy Tree** button in the column header in the left pane of the **Status** window

Note: The “9” and “0” keys are defined as hot/accelerator keys to facilitate quickly expanding or collapsing all resource hierarchy trees.

5.3.4.1.7.18. Cluster Connect Dialog

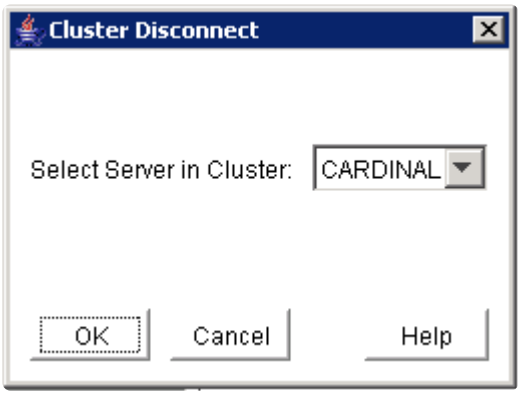


Server Name. The name of the server to which you want to connect.

Login. The login name of a user with LifeKeeper authorization on the server to which you want to connect.

Password. The password that authorizes the specified login on the server to which you want to connect.

5.3.4.1.7.19. Cluster Disconnect Dialog



Select Server in Cluster.

A drop-down list box containing the names of connected servers will appear. From the list, select a server from the cluster from which you want to disconnect. All servers in the cluster to be disconnected are noted in the confirmation dialog.

5.3.4.1.7.20. Resource Properties Dialog

The Resource Properties dialog is available from the Edit menu or from a resource context menu. This dialog displays the properties for a particular resource on a server. When accessed from the Edit menu, you can select the resource and the server. When accessed from a resource context menu, you can select the server.

General Tab

- **Tag**. The name of a resource instance, unique to a system, that identifies the resource to an administrator.

- **ID.** A character string associated with a resource instance, unique among all instances of the resource type, that identifies some internal characteristics of the resource instance to the application software associated with it.

- **Switchback.** (editable if user has Administrator permission) The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is intelligent, the server acts as a possible backup for the given resource. If the setting is automatic, the server actively attempts to re-acquire the resource, providing the following conditions are met:
  - The resource hierarchy must have been in service on the server when it left the cluster.
  - If it is in service at all, then the resource must currently be in service on a server with a lower priority.

> **Note:** Checks for automatic switchback are made only when LifeKeeper starts or when a new server is added to the cluster; they are not performed during normal cluster operation.

- **State.** Current state of the resource instance:
  - *Active* – In-service locally and protected.
  - *Warning* – In-service locally, but local recovery will not be attempted.
  - *Failed* – Out-of-service, failed.
  - *Standby* – Out-of-service, unimpaired.
  - *ILLSTATE* – A resource state has not been initialized properly by the resource initialization process which is run as part of the LifeKeeper startup sequence. Resources in this state are not under LifeKeeper protection.
  - *UNKNOWN* – Resource state could not be determined. The GUI server may not be available.

- **Reason.** If present, describes the reason the resource is in its current state, that is, the reason for the last state change. For example the application on galahad is in the OSU state because the shared primary resource ordbfsaa-on-tristan on tristan is in ISP or ISU state. Shared resources can be active on only one of the grouped systems at a time.

- **Initialization.** The setting that determines resource initialization behavior at boot time, for

example, AUTORES_ISP, INIT_ISP, or INIT_OSU.

Relations Tab

- **Parent.** Identifies the tag names of the resources that are directly dependent on this resource.
- **Child.** Identifies the tag names of all resources on which this resource depends.
- **Root.** Tag name of the resource in this resource hierarchy that has no parent.

Equivalencies Tab

- **Server.** The name of the server on which the resource has a defined equivalency.
- **Priority.** (editable if the user has Administrator permission). The failover priority value of the targeted server, for this resource.
- **Tag.** The tag name of this resource on the equivalent server.
- **Type.** The type of equivalency (SHARED, COMMON, COMPOSITE).
- **Reorder Priorities.** (available if the user has Administrator permission) Up/Down buttons let you to re-order the priority of the selected equivalency.

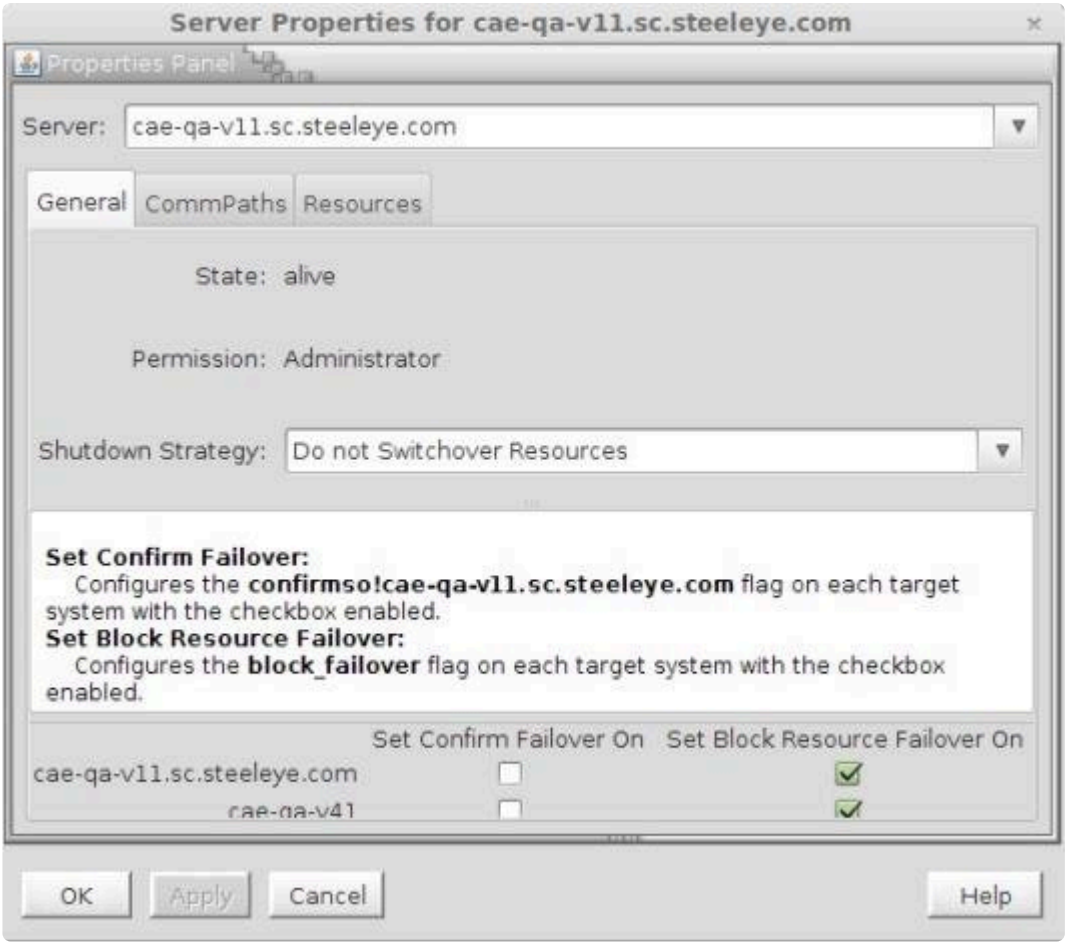
The OK button applies any changes that have been made and then closes the window. The Apply button applies any changes that have been made. The Cancel button, closes the window without saving any changes made since Apply was last clicked.

5.3.4.1.7.21. Server Properties Dialog

The Server Properties dialog is available from a server context menu or from the Edit menu. This dialog displays the properties for a particular server. The properties for the server will also be displayed in the properties panel if it is enabled.

The three tabs of this dialog are described below. The OK button applies any changes that have been made and then closes the window. The Apply button applies any changes that have been made. The Cancel button closes the window without saving any changes made since Apply was last clicked.

General Tab



- **Name.** Name of the selected server.

- **State.** Current state of the server. These are the possible server state values:
  - *ALIVE* – server is available.
  - *DEAD* – server is unavailable.
  - *UNKNOWN* – state could not be determined. The GUI server may not be available.

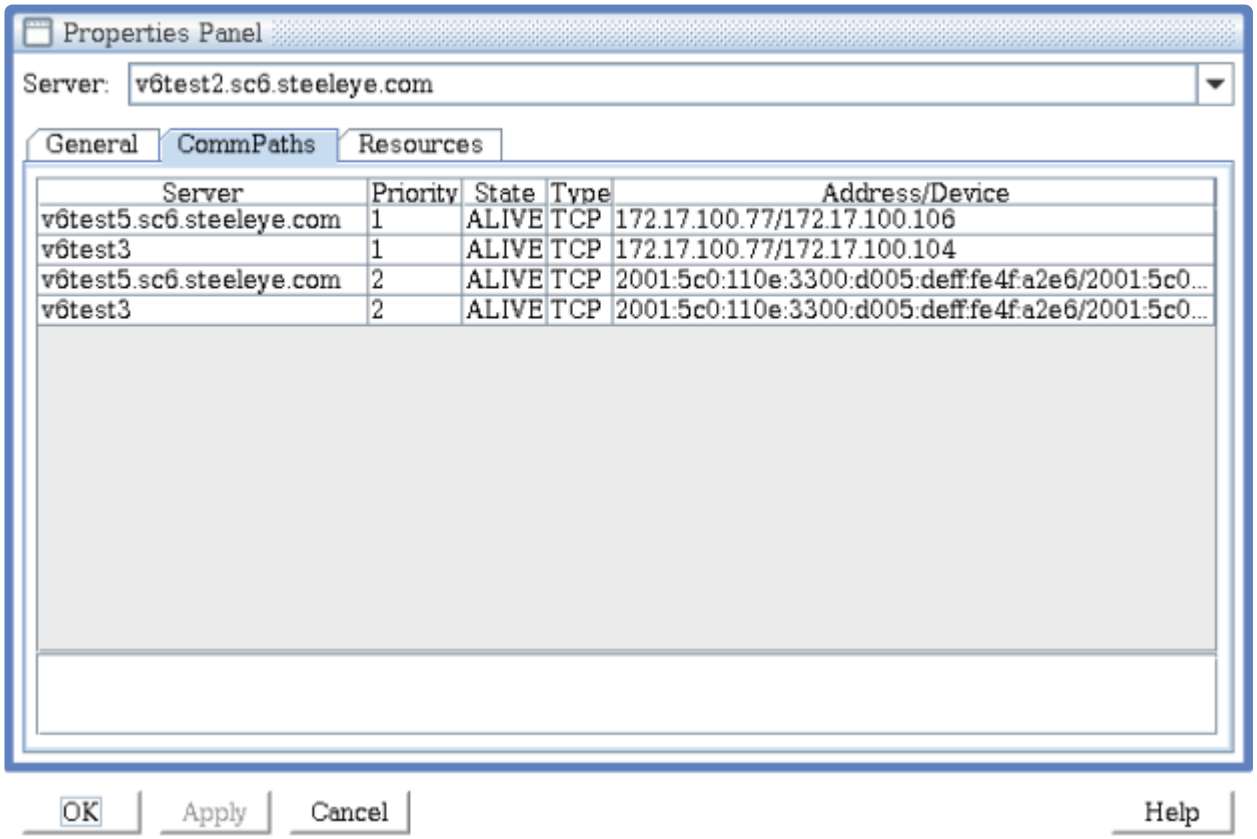
- **Permission.** The permission level of the user currently logged into that server. These are the possible permission values:
  - *Administrator* – the user can perform any LifeKeeper task.

- *Operator* – the user can monitor LifeKeeper resource and server status, and can bring resources in service and take them out of service.
- *Guest* – the user can monitor LifeKeeper resource and server status.

- **Shutdown Strategy.** (editable if the user has Administrator permission) The setting that governs whether or not resources are switched over to a backup server in the cluster when a server is shutdown. The setting "*Switchover Resources*" indicates that resources will be brought in service on a backup server in the cluster. The setting "*Do not Switchover Resources*" indicates that resources will not be brought in service on another server in the cluster.

- **Failover Strategy.** The setting allows you to require the confirmation of failovers from specific systems in the LifeKeeper cluster. It is only available to LifeKeeper administrators. Operators and guests will not be able to see it. By default, all failovers proceed automatically with no user intervention. However, once the confirm failover flag is set, failovers from the designated system will require confirmation by executing the command: `lk_confirmso -y system`. The failover may be blocked by executing the command: `lk_confirmso -n system`. The system will take a pre-programmed default action unless one of these commands is executed within a specified interval. Two flags in the `/etc/default/LifeKeeper` file govern this automatic action.
  - `CONFIRMSODEF` (This specifies the default action. If set to "0", the default action is to proceed with failover. If set to "1", the default action is to block failover.)
  - `CONFIRMSOTO` (This is set to the time in seconds that LifeKeeper should wait before taking the default action.)

CommPaths Tab



- **Server.** The server name of the other server the communication path is connected to in the LifeKeeper cluster.

- **Priority.** The priority determines the order by which communication paths between two servers will be used. Priority 1 is the highest and priority 99 is the lowest.

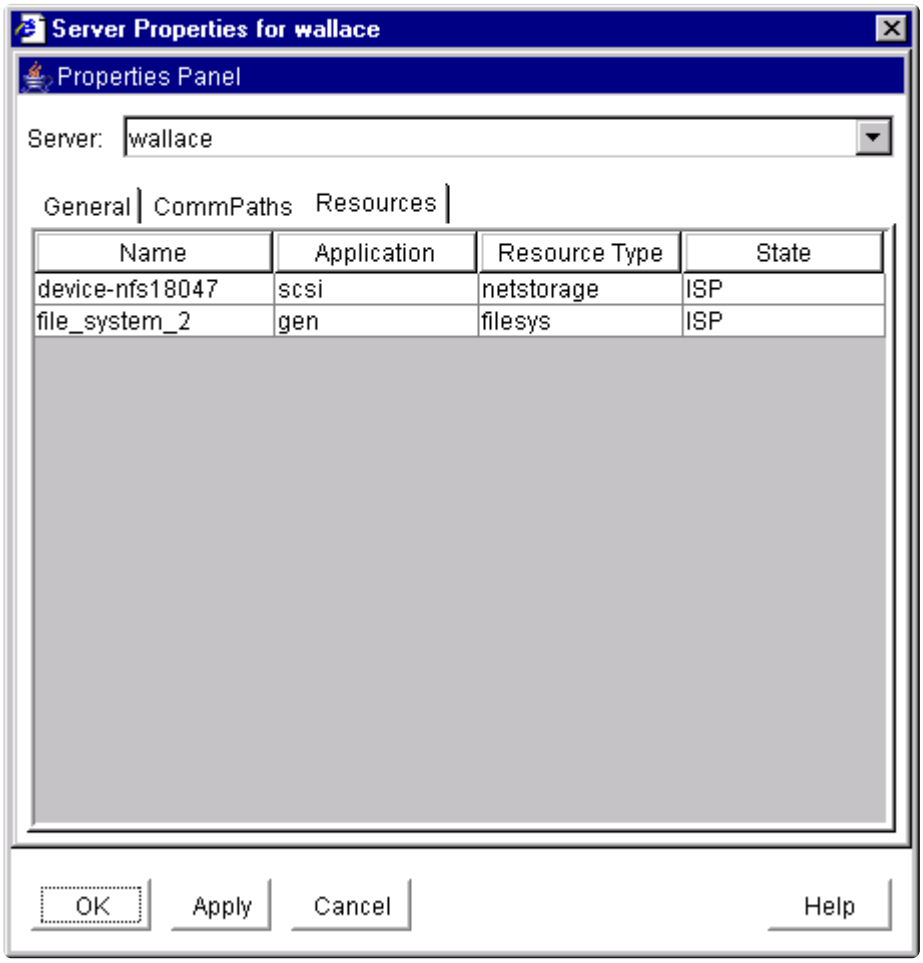
- **State.** State of the communications path in the LifeKeeper Configuration Database (LCD). These are the possible communications path state values:
  - *ALIVE* – functioning normally.
  - *DEAD* – no longer functioning normally.
  - *UNKNOWN* – state could not be determined. The GUI server may not be available.

- **Type.** The type of communications path, TCP (TCP/IP) or TTY, between the server in the list and the server specified in the Server field.

- **Address/Device.** The IP address or device name that this communications path uses.

- **Comm Path Status.** Summary communications path status determined by the GUI based on the state of the communications paths in the LifeKeeper Configuration Database (LCD). These are the possible communications path status values displayed below the detailed text in the lower panel:
  - *NORMAL* – all comm paths functioning normally.
  - *FAILED* – all comm paths to a given server are dead.
  - *UNKNOWN* – comm path status could not be determined. The GUI server may not be available.
  - *WARNING* – one or more comm paths to a given server are dead.
  - *DEGRADED* – one ore more redundant comm paths to a given server are dead.
  - *NONE DEFINED* – no comm paths defined.

Resources Tab



- **Name.** The tag name of a resource instance on the selected server.
- **Application.** The application name of a resource type (gen, scsi, …)
- **Resource Type.** The resource type, a class of hardware, software, or system entities providing a service (for example, app, filesys, nfs, device, disk,…)
- **State.** The current state of a resource instance:
  - *ISP* – In-service locally and protected.
  - *ISU* – In-service locally, but local recovery will not be attempted.
  - *OSF* – Out-of-service, failed.
  - *OSU* – Out-of-service, unimpaired.
  - *ILLSTATE* – Resource state has not been initialized properly by the resource initialization process which is run as part of the LifeKeeper startup sequence. Resources in this state are not under LifeKeeper protection.
  - *UNKNOWN* – Resource state could not be determined. The GUI server may not be available.

5.3.4.1.8. Operator Tasks

The following topics are more advanced tasks that require Operator permission.

5.3.4.1.8.1. Bringing a Resource In Service

Note: LifeKeeper puts resources in-service from the bottom of the hierarchy and works its way to the top level resource. When putting all of the resources in one hierarchy in-service, select the top (parent) resource in the hierarchy.

1. There are five possible ways to begin.

- ◦ Right-click on the icon for the resource/server combination that you want to bring into service. When the Resource Context Menu appears, click **In Service**.

- ◦ Right-click on the icon for the global resource that you want to bring into service. When the **Resource Context Menu** appears, click **In Service**. When the dialog comes up, select the server on which to perform the In Service from the **Server list** and click **Next**.

- ◦ On the Global Toolbar, click the **In Service** button. When the dialog comes up, select the server on which to perform the In Service from the **Server list** and click **Next**. On the next dialog, select one or more resources that you want to bring into service from the **Resource(s)** list and click **Next** again.

- ◦ On the Resource Context Toolbar, if displayed, click the **In Service** button.

- ◦ On the Edit Menu, point to **Resource** and then click **In Service**. When the dialog comes up, select the server on which to perform the **In Service** from the **Server list**, and click **Next**. On the next dialog, select one or more resources that you want to bring into service from the Resource(s) list and click **Next** again.

2. A dialog appears confirming the server and resource(s) that you have selected to bring into service. This dialog will include a warning that if you are bringing a dependent child resource into service, it will also bring the parent resource into service. Click **In Service** to bring the resource(s) into service along with any dependent child resources.

3. If the Output Panel is enabled, the dialog closes and the results of the commands to bring the resource(s) in service are shown in the **output panel**. If not, the dialog remains up to show these results and you click **Done** to finish when all results have been displayed. Any additional dependent (child) resources that were brought into service are noted in the dialog or **output panel**.

4. Errors that occur while bringing a resource in service are logged in the LifeKeeper log of the server on which you want to bring the resource into service.

5.3.4.1.8.2. Taking a Resource Out of Service

> **Note:** LifeKeeper takes resources out of service from the top of the hierarchy and works its way down to the other resources. When taking resources out of service if you want the entire hierarchy (parent with the child resources) to be taken out of service, take the lowest level resource out of service.

1. There are four possible ways to begin.

- ◦ Right-click on the icon for the global resource or resource/server combination that you want to take out of service. When the Resource Context Menu appears, click **Out of Service**.
- ◦ On the Global Toolbar, click the **Out of Service** button. When the Out of Service dialog comes up, select one or more resources that you want to take out of service from the Resource(s) list, and click **Next**.
- ◦ On the Resource Context Toolbar, if displayed, click the **Out of Service** button.
- ◦ On the Edit Menu, point to **Resource** and then click **Out of Service**. When the **Out of Service** dialog comes up, select one or more resources that you want to take out of service from the **Resource(s)** list, and click **Next**.

2. An **Out of Service** dialog appears confirming the selected resource(s) to be taken out of service. This dialog will include a warning that if you are taking a dependent child resource out of service, it will also take the parent resource(s) out of service. Click **Out of Service** to proceed to the next dialog box.

3. If the Output Panel is enabled, the dialog closes, and the results of the commands to take the resource(s) out of service are shown in the output panel. If not, the dialog remains up to show these results, and you click **Done** to finish when all results have been displayed.

4. Errors that occur while taking a resource out of service are logged in the LifeKeeper log of the server on which you want to take the resource out of service.

5.3.4.1.9. Advanced Tasks

LCD

LCM

LifeKeeper API for Monitoring

5.3.4.1.9.1. LCD

LifeKeeper Configuration Database

The LifeKeeper Configuration Database (LCD) maintains the object-oriented resource hierarchy information and stores recovery direction information for all resource types known to LifeKeeper. The data is cached within system shared memory and stored in files so that configuration data is retained over system restarts. The LCD also contains state information and specific details about resource instances required for recovery.

See the following related topics for information on the LCD directory structure, types of data stored, resource types available and use of application scripts.

Related Topics

- LCDI Commands
- LCD Configuration Data
- LCD Directory Structure
- LCD Resource Types
- LifeKeeper Flags
- Resources Subdirectories
- Structure of LCD Directory in /opt/LifeKeeper

5.3.4.1.9.1.1. LCDI Commands

Steps to Create Resources by Defining Your Own Recovery Kit

Note: Use the GUI to create resources if using the existing Recovery Kit.

LifeKeeper provides two mechanisms for defining an application resource hierarchy:

- LifeKeeper GUI
- LifeKeeper Configuration Database Interface (LCDI) commands

The LCDI is a set of interface commands provided by LifeKeeper that you can use to create and customize resource hierarchy configurations to meet your application needs. You use the command interface when an application depends upon multiple resources (such as two or more file systems).

For a description of the commands, see the LCDI manual pages. This topic provides a development scenario that demonstrates the way you can use both the GUI and command functions to create a resource hierarchy.

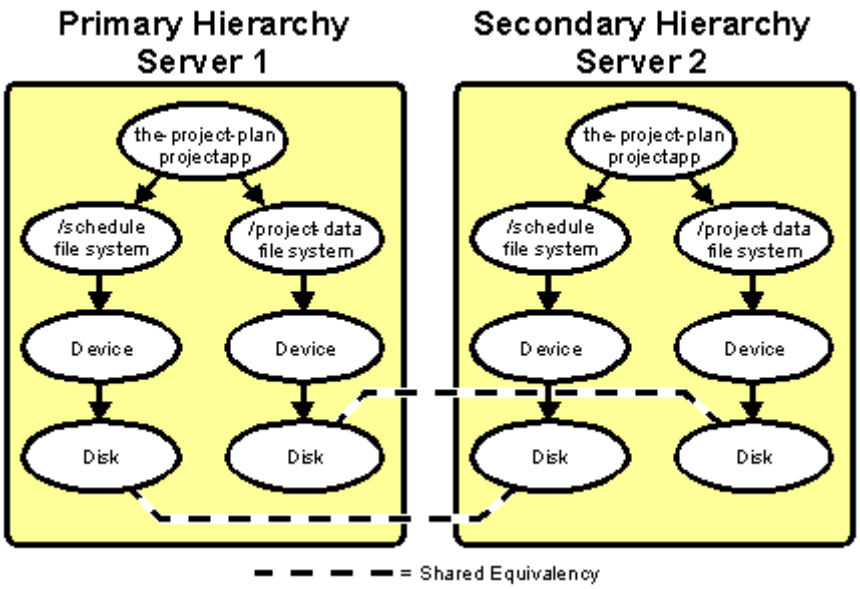
Scenario Situation

The example application, ProjectPlan, has data stored in SCSI file systems shared by Servers 1 and 2. Server 1 will be the primary hierarchy for the application. The application has two file systems:_ /project-data_ and */schedule*. The first step in the hierarchy definition is to determine the dependencies.

The example application has these dependencies:

- **Shared file systems.** The application depends upon its file systems: */project-data* and */schedule*.
- **SCSI disk subsystem.** The file systems in turn depend upon the SCSI disk subsystem, which includes the device, disk and host adapter resources.

As a result, the task is to create a hierarchy that looks like the following diagram.



Hierarchy Definition

These are the tasks required to construct the example application hierarchy:

1. **Create file system resources.** The LifeKeeper GUI provides menus to create file system resources. See Creating File System Resource Hierarchies.

   At the end of this definition task, the LCD has two filesys resources defined as follows:

ID	Tag	Server
/project-data	project-data-on-Server1	Server1
/project-data	project-data-from-Server1	Server2
/schedule	schedule-on-Server1	Server1
/schedule	schedule-from-Server1	Server2

   Note: LifeKeeper does not place any significance on the tag names used; they are simply labels. The tag names shown are the LifeKeeper defaults.

2. **Define resources.** The example requires the following definitions:

Application:	projectapp
Resource Type:	plan
Instance ID:	1yrplan
Tag:	the-project-plan

Note: Although you can create much of the definition using the LifeKeeper GUI, the rest of this example demonstrates the command interface.

3. **Create directories.** On each system, you create the necessary application recovery directories under the directory */opt/LifeKeeper/subsys* with the command:

   ```
   mkdir -p /opt/LifeKeeper/subsys/projectapp/Resources/plan/actions
   ```

4. **Define application.** The following commands create the application named *projectapp*:

   ```
   app_create -d Server1 -a projectapp

   app_create -d Server2 -a projectapp
   ```

5. **Define the resource type.** The following commands create the resource type named *plan*:

   ```
   typ_create -d Server1 -a projectapp -r plan

   typ_create -d Server2 -a projectapp -r plan
   ```

6. **Install recovery scripts.** Copy your restore and remove scripts to the following directory on each server:

   /opt/LifeKeeper/subsys/projectapp/Resources/plan/actions

7. **Define instance.** The following commands define an instance of resource type plan with the id *1yrplan*:

   ```
   ins_create -d Server1 -a projectapp -r plan -I\

   AUTORES_ISP -t the-project-plan -i 1yrplan

   ins_create -d Server2 -a projectapp -r plan -I\

   SEC_ISP -t the-project-plan -i 1yrplan
   ```

   The -I AUTORES_ISP instruction for the instance created on Server1 tells LifeKeeper to automatically bring the resource in service when LifeKeeper is restarted. In this case, the resource's restore script is run and, if successful, the resource is placed in the ISP state. This operation is not performed if the paired resource is already in service.

   The -I SEC_ISP instruction for the instance created on Server2 tells LifeKeeper that this resource instance should not be brought into service when LifeKeeper is restarted. Instead, Server2 will serve as the backup for the resource on Server1, and the local resource will be brought in service upon failure of the primary resource or server.

8. **Define dependencies.** The following commands define the dependencies between the application

and the file systems:

```
dep_create -d Server1 -p the-project-plan -c project-data-on-System1

dep_create -d Server2 -p the-project-plan -c project-data-from-
Server1

dep_create -d Server1 -p the-project-plan -c schedule-on-Server1

dep_create -d Server2 -p the-project-plan -cschedule-from-Server1
```

9. **Execute lcdsync.** Execute the following lcdsync commands to inform LifeKeeper to update its copy of the configuration:

```
lcdsync -d Server1

lcdsync -d Server2
```

10. **Set the resources to “In Service”.** Access LifeKeeper GUI on the primary server, select **[Edit] > [Resource] > [In-Service]** and set the resource “In Service”, or execute the following command on the primary server:

```
perform_action -t the-project-plan -a restore
```

11. **Create equivalency.** Create equivalency of resources registered for each node with the following commands to switch resources:

```
eqv_create -d Server1 -t the-project-plan -p 1 -S Server2 -o the-
project-plan -r 10 -e SHARED

eqv_create -d Server2 -t the-project-plan -p 10 -S Server1 -o the-
project-plan -r 1 -e SHARED
```

5.3.4.1.9.1.2. LCD Configuration Data

LCD stores the following related types of data:

- ◦ Dependency Information
- ◦ Resource Status Information
- ◦ Inter-Server Equivalency Information

Dependency Information

For each defined resource, LifeKeeper maintains a list of dependencies and a list of dependents (resources depending on a resource.) For information, see the `LCDI_relationship (1M)` and `LCDI_instances (1M)` manual pages.

Resource Status Information

LCD maintains status information in memory for each resource instance. The resource states recognized by **LCD** are **ISP**, **ISU**, **OSF**, **OSU** and **ILLSTATE**. Resources may change from one state to another when a system event occurs or when an administrator takes certain actions. When a resource changes states, the status change is reflected in the LCD on the local server as well as in the database of the backup servers for that resource.

Inter-Server Equivalency Information

Relationships may exist between resources on various servers. A shared equivalency is a relationship between two resources on different servers that represents the same physical entity. When two servers have a resource with a shared equivalency relationship, LifeKeeper attempts to ensure in its actions that only one of the two servers has the resource instance in the in-service, protected [ISP] state at any one time. Both servers can have the resource instance in an out-of-service state [**OSU** or **OSF**], but for data integrity reasons, only one server can have the resource in service at any given time.

Disks on a Small Computer System Interface (SCSI) bus are one example of equivalent resources. With the SCSI locking (or reserve) mechanism, only one server can own the lock for a disk device at any point in time. This lock ownership feature guarantees that two or more servers cannot access the same disk resource at the same time.

Furthermore, the dependency relationships within a hierarchy guarantee that all resources that depend upon the disk, such as a file system, are in service on only one server at a time.

5.3.4.1.9.1.3. LCD Directory Structure

Major subdirectories under */opt/LifeKeeper*:

- ◦ **config.** LifeKeeper configuration files, including shared equivalencies.
- ◦ **bin.** LifeKeeper executable programs, such as is_recoverable. See Fault Detection and Recovery Scenarios for descriptions.
- ◦ **subsys.** Resources and types. LifeKeeper provides resource and type definitions for the shared SCSI disk subsystem in scsi and for the generic application menu functions in gen. When you define an application interface, you create directories under subsys.
- ◦ **events.** Alarming events. See LifeKeeper Alarming and Recovery for further information.

The structure of the LCD directory in /opt/LifeKeeper is shown in the topic Structure of LCD Directory in */opt/LifeKeeper*.

5.3.4.1.9.1.4. LCD Resource Types

The LCD is maintained in both shared memory and in the */opt/LifeKeeper* directory. As highlighted on the directory structure diagram, subsys contains two application resource sets you can use to define your application interface:

- ◦ **gen** – generic application and file system information
- ◦ **scsi** – recovery information specific to the SCSI

These subdirectories are discussed in Resources Subdirectories.

5.3.4.1.9.1.5. LifeKeeper Flags

Near the end of the detailed status display, LifeKeeper provides a list of the flags set for the system. A common type is a Lock LCD flag used to ensure that other processes wait until the process lock completes its action. The following is the standard LCD lock format:

```
!action!processID!time!machine:id.
```

These are examples of general LCD lock flags:

- **`!action!02833!701236710!<servername>:filesys`**. The creation of a filesystem hierarchy produces a flag in this format in the status display. The *filesys* designation can be a different resource type for other application resource hierarchies or *app* for generic or user-defined applications.
- Other typical flags include **`!nofailover!machine`** and **`shutdown_switchover`**. The **`!nofailover!machine`** flag is an internal, transient flag created and deleted by LifeKeeper which controls aspects of server failover. The **`shutdown_switchover`** flag indicates that the shutdown strategy for this server has been set to switchover such that a shutdown of the server will cause a switchover to occur. See `LCDI-flag(1M)` for more detailed information on the possible flags.

5.3.4.1.9.1.6. Resources Subdirectories

The ***scsi*** and ***gen*** directories each contain a resources subdirectory. The content of those directories provides a list of the resource types provided by LifeKeeper:

scsi resource types. You find these resource types in the */opt/LifeKeeper/subsys/scsi/resources* directory. Note that there may be additional directories depending upon your configuration.

- ◦ **device** —disk partitions or virtual disk devices
- ◦ **disk** —physical disks or LUNs
- ◦ **hostadp** —host adapters

gen resource types. You find these resource types in the */opt/LifeKeeper/subsys/gen/resources* directory:

- ◦ **filesys** —file systems
- ◦ **app** —generic or user-defined applications that may depend upon additional resources

Each resource type directory contains one or more of the following:

- ◦ **instances.** This file reflects the permanent information saved in the LCD about resource instances. It contains descriptive information for the resource instances associated with this resource type.

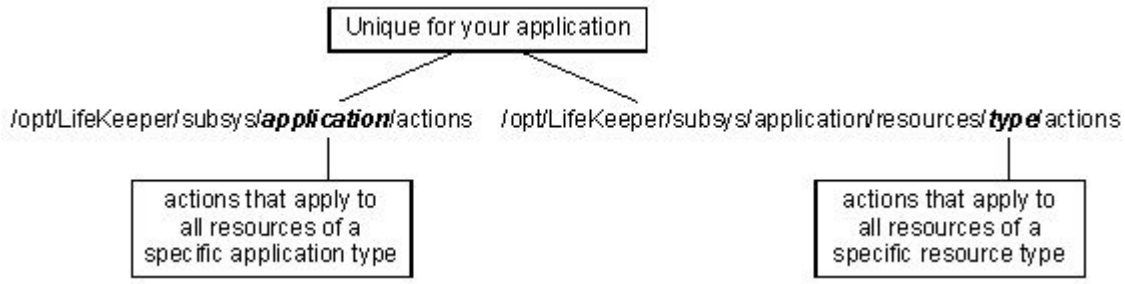
> **WARNING:** Do not modify the instances file (or any LCD file) directly. To create or manipulate resource instances, use only the LifeKeeper GUI functions or the LifeKeeper LCDI_instances commands: ins_create, ins_remove, ins_gettag, ins_setas, ins_setinfo, ins_setinit, ins_setstate and ins_list. Refer to the LCDI_instances (1M) manual pages for explanations of these commands.

- ◦ ***recovery.*** This optional directory contains the programs used to attempt the local recovery of a resource for which a failure has been detected. The recovery directory contains directories that correspond to event classes passed to sendevent. The names of the directories must match the class parameter (-C) passed to the sendevent program. (See LifeKeeper Alarming and Recovery.)

In each subdirectory, the application can place recovery programs that service event types of the corresponding event class. The name of these programs must match the string passed to sendevent with the -E parameter. This optional directory may not exist for many applications.

- ◦ ***actions.*** This directory contains the set of recovery action programs that act only on resource instances of the specific resource type. If, for your application, any actions apply to

all resource types within an application, place them in an ***actions*** subdirectory under the application directory rather than under the ***resource type*** directory.



Recovery direction software is used to modify or recover a resource instance. Two actions, **remove** and **restore**, must exist in the ***actions*** directory for each resource type.

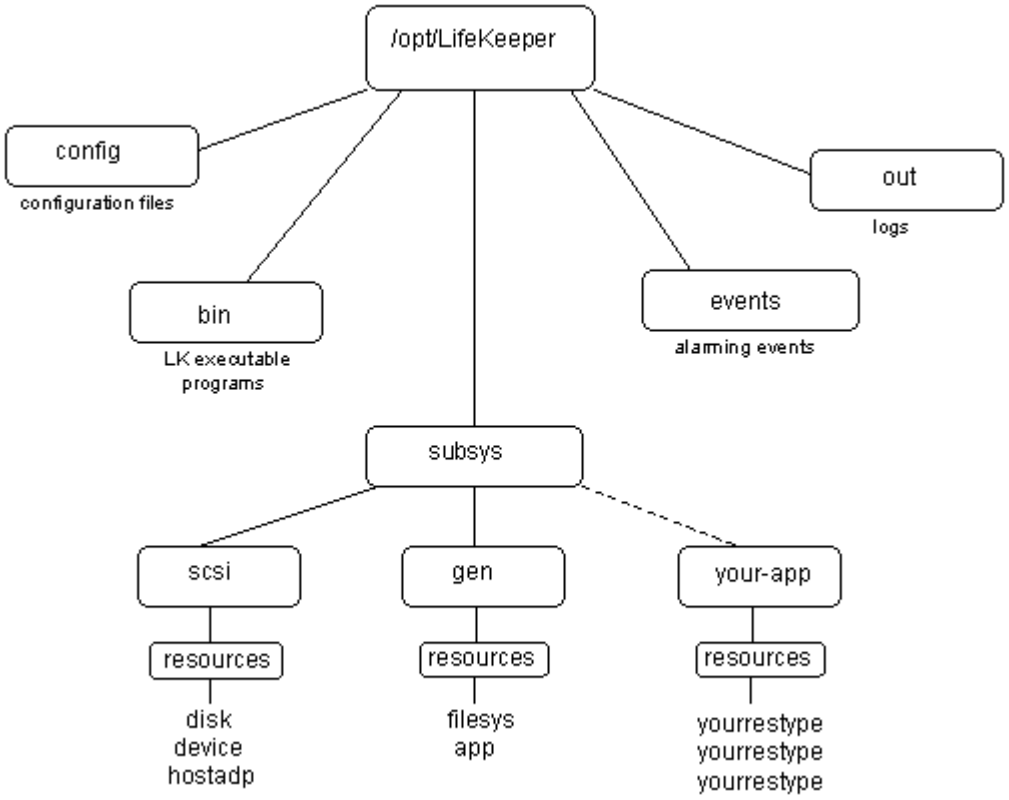
Resource Actions

The ***actions*** directory for a resource type contains the programs (most often shell scripts) that describe specific application functions. Two actions are required for every resource type—restore and remove.

The remove and restore programs should perform symmetrically opposite functions; that is, they undo the effect of one another. These scripts should never be run manually. They should only be run by executing the LifeKeeper Recovery Action and Control Interface (LRACI) `perform_action` shell program described in the `LRACI-perform_action (1M)` manual page.

5.3.4.1.9.1.7. Structure of LCD Directory in /opt/LifeKeeper

The following diagram shows the directory structure of ***/opt/LifeKeeper***.



5.3.4.1.9.2. LCM

The LifeKeeper Communications Manager (LCM) provides reliable communication between processes on one or more LifeKeeper servers. This process can use redundant communication paths between systems so that failure of a single communication path does not cause failure of LifeKeeper or its protected resources. The LCM supports a variety of communication alternatives including RS-232 (TTY) and TCP/IP connections.

The LCM provides the following:

- **LifeKeeper Heartbeat.** Periodic communication with other connected LifeKeeper systems to determine if the other systems are still functioning. LifeKeeper can detect any total system failure that is not detected by another means by recognizing the absence of the heartbeat signal.
- **Administration Services.** The administration functions of LifeKeeper use the LCM facilities to perform remote administration. This facility is used for single-point administration, configuration verification and sanity checking of administrative actions.
- **Configuration and Status Communication.** The LifeKeeper configuration database (LCD) tracks resource status, availability and configuration through the LCM facilities. These facilities allow the LCD to maintain consistent resource information between the primary and secondary systems.
- **Failover Recovery.** If a resource fails on a system, the LCM notifies LifeKeeper to recover the resource on a backup system.

In addition to the LifeKeeper services provided by the LCM, inter-system application communication is possible through a set of shell commands for reliable communication. These commands include `snd_msg`, `rcv_msg`, and `can_talk`. These commands are described in the `LCMI_mailboxes (1M)` manual pages. The LCM runs as a real-time process on the system assuring that critical communications such as system heartbeat will be transmitted.

Related Topics

Communication Status Information

Alarming and Recovery

5.3.4.1.9.2.1. Communication Status Information

The communications status information section of the status display lists the servers known to LifeKeeper and their current state followed by information about each communication path.

The following sample is from the communication status section of a short status display:

```
MACHINE NETWORK ADDRESSES/DEVICE STATE PRIO

tristan TCP 100.10.100.100/100.10.100.200 ALIVE 1

tristan TTY /dev/ttyS0 ALIVE —
```

For more information, see the communication status information section of the topics Detailed Status Display and the Short Status Display.

5.3.4.1.9.2.2. LifeKeeper Alarming and Recovery

LifeKeeper error detection and notification is based on the event alarming mechanism, sendevent. The key concept of the **sendevent** mechanism is that independent applications can register to receive alarms for critical components. Neither the alarm initiation component nor the receiving application(s) need to be modified to know the existence of the other applications. Application-specific errors can trigger LifeKeeper recovery mechanisms via the **sendevent** facility.

This section discusses topics related to alarming including alarm classes, alarm processing and alarm directory layout and then provides a processing scenario that demonstrates the alarming concepts.

Alarm Classes

The */opt/LifeKeeper/events* directory lists a set of alarm classes. These classes correspond to particular sub-components of the system that produces events (for example, *filesys*). For each alarm class, subdirectories contain the set of potential alarms (for example, *badmount* and *diskfull*). You can register an application to receive these alarms by placing shell scripts or programs in the appropriate directories.

LifeKeeper uses a basic alarming notification facility. With this alarming functionality, all applications registered for an event have their handling programs executed asynchronously by sendevent when the appropriate alarm occurs. With LifeKeeper present, the **sendevent** process first determines if the LifeKeeper resource objects can handle the class and event. If LifeKeeper finds a class/event match, it executes the appropriate recover scenario.

Defining additional scripts for the **sendevent** alarming functionality is optional. When you define LifeKeeper resources, LifeKeeper provides the basic alarming functionality described in the processing scenarios later in this chapter.

> **Note:** Local recovery for a resource instance is the attempt by an application under control of LifeKeeper to return interrupted resource services to the end-user on the same system that generated the event. Inter-server recovery allows an application to migrate to a backup system. This type of recovery is tried after local recovery fails or is not possible.

Alarm Processing

Applications or processes that detect an event which may require LifeKeeper attention can report the event by executing the **sendevent** program, passing the following arguments: respective error class, error name and failing instance. Refer to the sendevent(5) manual pages for required specifics and optional parameters and syntax.

Alarm Directory Layout

The */opt/LifeKeeper/events* directory has two types of content:

- ◦ **LifeKeeper supplied classes.** LifeKeeper provides two alarm classes listed under the *events* directory: *lifekeeper* and *filesys*. An example of an alarm event includes *diskfull*. The alarm classes correspond to the strings that are passed with the -C option to the **sendevent** command and the alarm events correspond to the strings that are passed with the -E option. The lifekeeper alarm class is used internally by LifeKeeper for event reporting within the LifeKeeper subsystem.

- ◦ **Application-specific classes.** The other subdirectories in the *events* directory are added when specific applications require alarm class definitions. Applications register to receive these alarms by placing shell scripts or binary programs in the directories. These programs are named after the application package to which they belong.

5.3.4.1.9.3. LifeKeeper API for Monitoring

Introduction

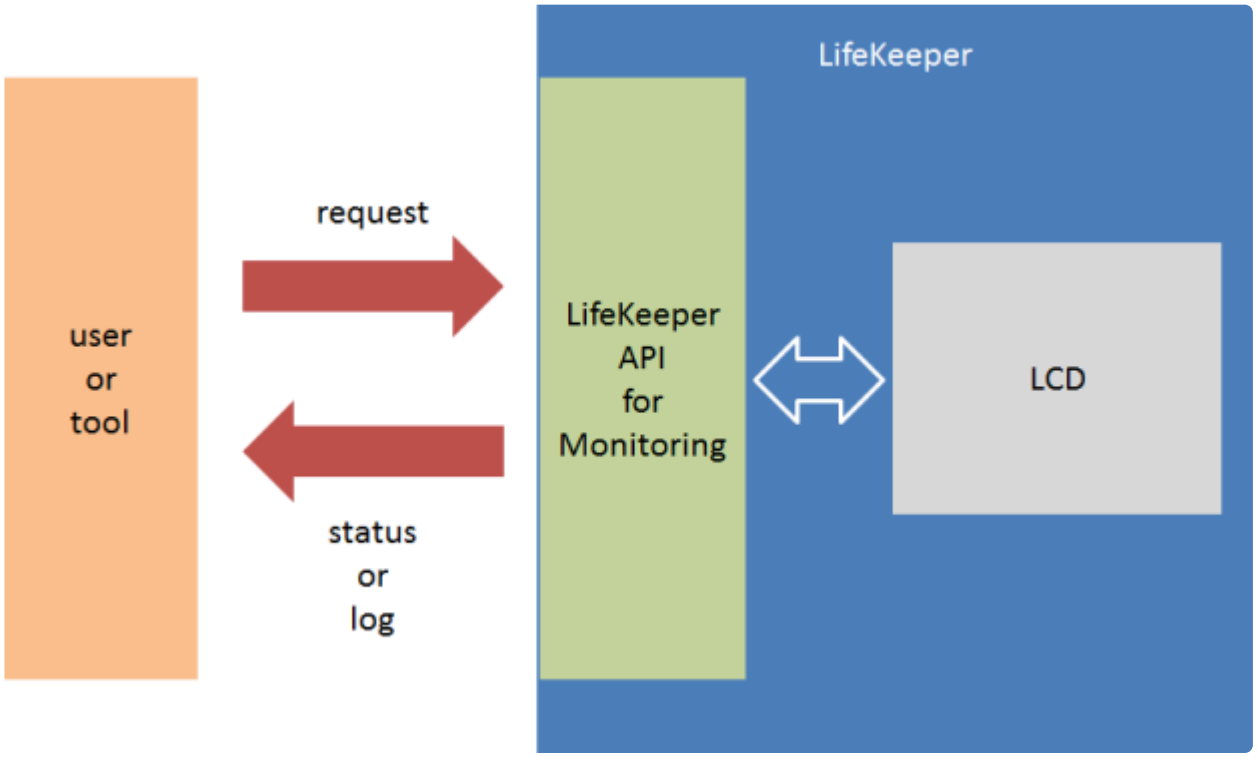
The LifeKeeper API for Monitoring can obtain the operational status of LifeKeeper nodes and their protected resources by making status inquiries to the available nodes in the LifeKeeper cluster.

Summary

This document describes the LifeKeeper API for Monitoring (hereinafter referred to as the API) and is targeted for developers who manage the resource protected by LifeKeeper. By using the API, the information supplied by the lcdstatus command is obtained through CGI script and the lighttpd module. By using this API, users can determine the current status of the LifeKeeper nodes and resources without logging-in to LifeKeeper servers. The API can supply the following information.

- ◦ LifeKeeper node status is the node alive and processing or down
- ◦ Communication path status between nodes in the cluster, are communication path(s) up or down
- ◦ Status of protected resources

To get the detailed status of any abnormal condition requires logging-in to LifeKeeper GUI or checking the LifeKeeper log as necessary.



Information to be supplied with this API

The following information is supplied through this API when the user makes an inquiry to an active LifeKeeper node. The information supplied is about the specific LifeKeeper server to which the inquiry was directed even if the cluster consists of multiple servers.

- ◦ Status
  - º Operating status of each server
    - ▪ Node name
    - ▪ Operational status (ALIVE/DEAD)
  - º Operational status of communication path(s)
    - ▪ Node name
    - ▪ Operational status (ALIVE/DEAD)
    - ▪ Address / device name
  - º Status of protected resources
    - ▪ Node Name
    - ▪ Tag
    - ▪ Status (ISP, OSU, OSF, …)
    - ▪ Dependency setting
    - ▪ Mirror information for Data Replication resources (available only if status is ISP)
      - ▪ Tag
      - ▪ Mirror status (Sync, Paused, …)
      - ▪ Replication status (75%, 100%, …)
- ◦ Log
  - º /var/log/lifekeeper.log *Not supported if log file path is changed

- ▪ Up to 1000 lines (when data output format is HTML)
- ▪ All (when data output format is plain text)

- º /var/log/lifekeeper.err *Not supported if log file path is changed
  - ▪ Up to 1000 lines (when data output format is HTML)
  - ▪ All (when data output format is plain text)

Communication Format

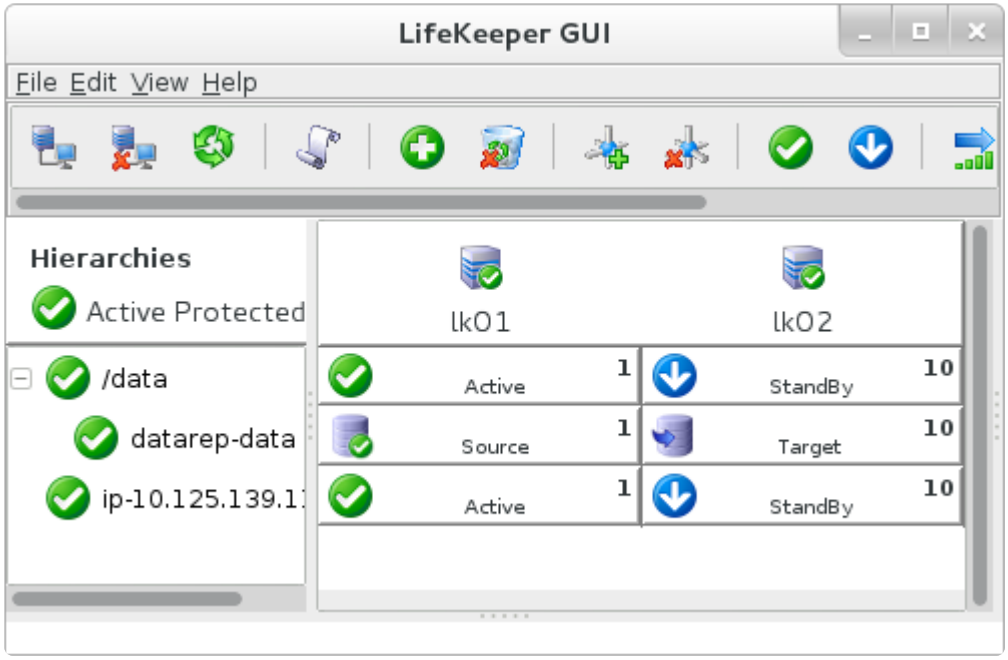
The API uses HTTP to obtain the requested information. To obtain information, the user sends a HTTP GET request to the CGI scripts via lighttpd on the specific server.

Data Format

The following 3 data formats are available.

- ◦ JSON
  - º To be used by an external tool to analyze the status information returned
  - º Status checking is possible
  - º Log output is not available
- ◦ HTML
  - º To be used to visually check via a browser
  - º Status checking is possible
  - º Log information is available up to 1000 lines
- ◦ plain text
  - º Used for regular log checking
  - º For logging purpose only and not for checking the status
  - º All contents of /var/log/lifekeeper.log and /var/log/lifekeeper.err are available

Available JSON format and HTML format from the status in the following figure.




```
{
        "resource": [
                {
                        "replication": {},
                        "child": [
                                {
                                        "tag": "datarep-data"
                                }
                        ],
                        "server": {
                                "status": "ISP",
                                "name": "lk01"
                                },
                        "tag": "/data"
                },
                {
                        "replication": {
                                "percent": "100%",
                                "mirror": "Fully Operational"
                        },
                        "child": [],
                        "server": {
                                "status": "ISP",
                                "name": "lk01"
                        },
                        "tag": "datarep-data"
                },
                {
                        "replication": {},
                        "child": [],
                        "server": {
                                "status": "ISP",
                                "name": "lk01"
                        },
                        "tag": "ip-10.125.139.118"
                }
        ],
        "compath": [
```

```
                {
                        "status": "ALIVE",
                        "server": [
                                {
                                        "name": "lk01",
                                        "term": "192.168.139.18"
                                },
                                {
                                        "name": "lk02",
                                        "term": "192.168.139.19"
                                }
                        ]
                },
                {
                        "status": "ALIVE",
                        "server": [
                                {
                                        "name": "lk01",
                                        "term": "172.20.139.18"
                                },
                                {
                                        "name": "lk02",
                                        "term": "172.20.139.19"
                                }
                        ]
                }
        ],
        "server": [
                {
                        "status": "ALIVE",
                        "name": "lk01"
                },
                {
                        "status": "ALIVE",
                        "name": "lk02"
                }
        ]
}
```


RESOURCEs

tag	lk01
/data	ISP
datarep-data	ISP
ip-10.125.139.118	ISP

DATA REPLICATIONs

tag	nodes	mirror status	replication status
datarep-data	lk01 -> lk02	Fully Operational	100%

COMMUNICATION PATHs

communication path	status
192.168.139.18/192.168.139.19	ALIVE
172.20.139.18/172.20.139.19	ALIVE

How to use

Activate the API

The API is disabled by default. To activate, requires modification of /etc/default/LifeKeeper set the LKAPI_MONITORING configuration parameter to true. Setting of the configuration parameter only activates the API on that node and therefore must be set on each node on which the API will be used. Setting of this configuration parameter does not require a restart of LifeKeeper.

```
LKAPI_MONITORING=true
```

Port Number

The API uses port 779 by default. To change the port number, the user needs to set the following in /etc/default/LifeKeeper.

```
LKAPI_WEB_PORT=<port number>
```

Using Examples

To obtain information a request is made to a server with an active LifeKeeper API configuration. Basic example using curl.

```
curl http://<IPADDR>:779/Monitoring.cgi
```

If no arguments are given, the current status is obtained using the JSON data format. Request for log information using HTML data format.

```
curl http://<IPADDR>:779/Monitoring.cgi?format=html&show=log
```

The list of available arguments can be found in the table below.

Name	Explanation	Value	Comments
show	Specify the target information	status, log, log-err	show=status is the default
format	Specify data format	json, html, plain	format=json is the default. If the format is json an error will be displayed if show=log or show=log-err is set.

Security

All the users requesting information via the API must be authorized to get LifeKeeper status information. For this reason, user security settings can limit the users who can get the status by, configuring SSL, and encrypting the information.

Basic Authentication

To obtain the information via the API, Basic Authentication is required. To setup the authentication requires modification to the lighttpd configuration file (Modify the part in red colored character.) plus a restart of the lighttpd module. See figure 7 for how to configure lightttp.conf.

After modification execute the command ”/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd” and reboot lighttpd to restart lighttpd using the new configuration.

/opt/LifeKeeper/etc/lighttpd/lighttpd.conf

```
server.modules              = (
        :
        %(color-red)"mod_auth",% # uncommenting
```

/opt/LifeKeeper/lib64/steeleye-lighttpd/include_server_bind.pl

```
print qq/    auth.backend = "htpasswd"\n/;
print qq/    auth.backend.htpasswd.userfile = "\/opt\/LifeKeeper\/etc\/lighttp
d\/lighttpd.user.htpasswd"\n/;
print qq/    auth.require = ( "\/" =>\n/;
print qq/        (\n/;
print qq/                    “method”  => “basic”,\n/;
print qq/                    “realm”   => “LifeKeeperAPI”,\n/;
print qq/                    “require” => “valid-user”\n/;
print qq/        )\n/;
print qq/    )\n/;
print qq/ }\n/;
```

Step to create htpasswd file.

```
htpasswd -c /opt/LifeKeeper/etc/lighttpd/lighttpd.user.htpasswd
<USERNAME>
```

SSL/TLS Set Up

SSL/TLS is available for the communication via this API. The lighttpd modifications for SSL/TLS is shown in the example in the following figure. After modification execute the command ”/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd” and reboot lighttpd to restart lighttpd with the new configuration.

/opt/LifeKeeper/etc/lighttpd/include_ssl_port.pl

```
configAPI(“0.0.0.0”, 443);

if(socket($sock, AF_INET6, SOCK_STREAM, 0)) {
        configAPI(“[::]”, 443);

}
```

```
```

sub configAPI { my $addr = shift; my $port = shift;

print qq/\$SERVER[“socket”] == “$addr:$port” {\n/; print qq/ server.document-root = “\/opt\/LifeKeeper\/api”\n/; print qq/ ssl.engine = “enable”\n/; print qq/ ssl.pemfile = “\/opt\/LifeKeeper\/etc\/certs\/LK4LinuxValidNode.pem”\n/; print qq/ ssl.use-sslv2 = “disable”\n/; print qq/ ssl.use-sslv3 = “disable”\n/; print qq/ }\n/;
}

Modification to Support SSL/TLS + Basic Authentication

Using SSL/TLS, modification example to set up Basic authentication is below. After modification, execute the command “/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd“ and restart lighttpd to reflect the modified set up.

/opt/LifeKeeper/etc/lighttpd/lighttpd.conf

```
```

server.modules = (: "mod_auth", # uncommenting

/opt/LifeKeeper/etc/lighttpd/include_ssl_port.pl

```
configAPI(“0.0.0.0”, 443);

if(socket($sock, AF_INET6, SOCK_STREAM, 0)) {
        configAPI(“[::]”, 443);

}
```

sub configAPI { my $addr = shift; my $port = shift;

print qq/\$SERVER[“socket”] == “$addr:$port” {\n/; print qq/ server.document-root = “\/opt\/LifeKeeper\/api”\n/; print qq/ ssl.engine = “enable”\n/; print qq/ ssl.pemfile = “\/opt\/LifeKeeper\/etc\/certs\/LK4LinuxValidNode.pem”\n/; print qq/ ssl.use-sslv2 = “disable”\n/; print qq/ ssl.use-sslv3 = “disable”\n/; print qq/ auth.backend = “htpasswd”\n/; print qq/ auth.backend.htpasswd.userfile = “\/opt\/LifeKeeper\/etc\/lighttpd\/lighttpd.user.htpasswd”\n/; print qq/ auth.require = (“\/” =>\n/; print qq/ (\n/; print qq/ “method” => “basic”,\n/; print qq/ “realm” => “LifeKeeperAPI”,\n/; print qq/ “require” => “valid-user”\n/; print qq/)\n/; print qq/)\n/; print qq/ }\n/;
}

IP Address Access Limitation

The lightppd configuration can also be setup to limit IP addresses that can be used to access data via the API. The lighttpd configuration to limit access is shown in Figure 9. The example will reject the

connections from IP address other than 192.168.10.1. After modification execute the command ”/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd” to restart lighttpd with the new configuration.

/opt/LifeKeeper/etc/lighttpd/conf.d/lkapi_user.conf

```
$HTTP[“remoteip”] != “192.168.10.1” {
url.access-deny = ( “” )
}
```

Error

Errors can occur during the usage of this API when enabled. Should this occur, the summary of the error is output. Error example when JSON format is shown below. HTTP status code returned by lighttpd is not described here.

```
{
“error” : { id : -1, message : “Failed to get LCD status” }
}
```

Similar message is output in the case the output format is HTML.

5.3.4.1.10. Maintenance Tasks

The following are tasks for maintaining LifeKeeper.

5.3.4.1.10.1. Changing LifeKeeper Configuration Values

There are a number of values in LifeKeeper that may need to be changed after LifeKeeper has been configured and set up. Examples of values that may be modified include the uname of LifeKeeper servers, comm path ip addresses, ip resource addresses and tag names. To change these values, carefully follow the instructions below.

1. Stop LifeKeeper on all servers in the cluster using the command:

   ```
   /etc/init.d/lifekeeper stop-nofailover
   ```

   or

   ```
   LKSTOP_MODE=stop-nofailover systemctl stop lifekeeper
   ```

   There is no need to delete comm paths or unextend resource hierarchies from any of the servers.

2. If you are changing the uname of a LifeKeeper server, change the server's hostname using the Linux `hostname(1)` command.

3. Before continuing, ensure that any new host names are resolvable by all of the servers in the cluster. If you are changing comm path addresses, check that the new addresses are configured and working (the **ping** and **telnet** utilities can be used to verify this).

4. If more than one LifeKeeper value is to be changed, old and new values should be specified in a file on each server in the cluster in the following format:

   ```
   old_value1=new_value1
   ....
   old_value9=new_value9
   ```

5. Verify that the changes to be made do not have any unexpected side effects by examining the output of running the **lk_chg_value** command on **all** servers in the cluster. If there is more than one value to change, run the command:

   ```
   $LKROOT/bin/lk_chg_value -Mvf file_name
   ```

   where ***file_name*** is the name of the file created in Step 4.

   If there is only one value to change, run the command:

   ```
   $LKROOT/bin/lk_chg_value -Mvo old_value -n new_value
   ```

The **-M** option specifies that no modifications should be made to any LifeKeeper files.

6. Modify LifeKeeper files by running the `lk_chg_value` command without the **-M** option on all servers in the cluster. If there is more than one value to change, run the command:

   ```
   $LKROOT/bin/lk_chg_value -vf file_name
   ```

   where ***file_name*** is the name of the file created in Step 4.

   If there is only one value to change, run the command:

   ```
   $LKROOT/bin/lk_chg_value -vo old_value -n new_value
   ```

7. Restart LifeKeeper using the command:

   ```
   /etc/init.d/lifekeeper start
   ```

   or

   ```
   systemctl start lifekeeper
   ```

   If the cluster is being viewed using the LifeKeeper GUI, it may be necessary to close and restart the GUI.

Example:

Server1 and *Server2* are the LifeKeeper server unames in a two-node cluster. *Server1* has a comm path with address 172.17.100.48. *Server2* has an ip resource with address 172.17.100.220 which is extended to *Server1*. To change the following values for *Server1*:

Value	Old	New
uname	Server1	Newserver1
comm path address	172.17.100.48	172.17.105.49
IP resource address	172.17.100.220	172.17.100.221

The following steps should be performed to make these changes.

1. Stop LifeKeeper on both *Server1* and *Server2* using the command:

   ```
   /etc/init.d/lifekeeper stop-nofailover
   ```

   or

   ```
   LKSTOP_MODE=stop-nofailover systemctl stop lifekeeper
   ```

2. Change the uname of # Server1 to *Newserver1* using the command:

```
hostname Newserver1
```

3. Create the file, */tmp/subs*, with the content below, on both *Newserver1* and *Server2*:

```
Server1=Newserver1

172.17.100.48=172.17.105.49

172.17.100.220=172.17.100.221
```

4. Verify that the changes specified will not have any unexpected side effects by examining the output of running the following command on both servers:

```
$LKROOT/bin/lk_chg_value -Mvf /tmp/subs
```

5. Modify the LifeKeeper files by running the **lk_chg_value** command without the **-M** option on both servers:

```
$LKROOT/bin/lk_chg_value -vf /tmp/subs
```

6. Restart LifeKeeper on both servers using the command:

```
/etc/init.d/lifekeeper start
```

or

```
systemctl start lifekeeper
```

Note: To see the changes **lk_chg_value** will make without modifying any LifeKeeper files, use the **-M** option. To see the files **lk_chg_value** is examining, use **-v**. To not modify tag names, use the **-T** option. To not modify resource ids, use the **-I** option.

5.3.4.1.10.2. File System Health Monitoring

The File System Health Monitoring feature detects conditions that could cause LifeKeeper protected applications that depend on the file system to fail. Monitoring occurs on active/in-service resources (i.e. file systems) only. The two conditions that are monitored are:

- A full (or almost full) file system, and
- An improperly mounted (or unmounted) file system.

When either of these two conditions is detected, one of several actions might be taken.

- A warning message can be logged and email sent to a system administrator.
- Local recovery of the resource can be attempted.
- The resource can be failed over to a backup server.

Condition Definitions

Full or Almost Full File System

A “disk full” condition can be detected, but cannot be resolved by performing a local recovery or failover – administrator intervention is required. A message will be logged by default. Additional notification functionality is available. For example, an email can be sent to a system administrator, or another application can be invoked to send a warning message by some other means.To enable notification for the full/almost full disk conditions a basic event notification script name notify has been provided in the directory /opt/LifeKeeper/events/filesys/diskfull. Simply add the functionality required to send email or execute another application.

In addition to a “disk full” condition, a “disk almost full” condition can be detected and a warning message logged in the LifeKeeper log.

The “disk full” threshold is:

```
FILESYSFULLERROR=95
```

The “disk almost full” threshold is:

```
FILESYSFULLWARN=90
```

The default values are 90% and 95% as shown, but are configurable via tunables in the */etc/default/LifeKeeper* file. The meanings of these two thresholds are as follows:

`FILESYSFULLWARN` – When a file system reaches this percentage full, a message will be

displayed in the LifeKeeper log.

`FILESYSFULLERROR` – When a file system reaches this percentage full, a message will be displayed in the LifeKeeper log as well as the system log. The file system notify script will also be called.

Unmounted or Improperly Mounted File System

LifeKeeper checks the */etc/mtab* file to determine whether a LifeKeeper protected file system that is in service is actually mounted. In addition, the mount options are checked against the stored mount options in the *filesys* resource information field to ensure that they match the original mount options used at the time the hierarchy was created.

If an unmounted or improperly mounted file system is detected, local recovery is invoked and will attempt to remount the file system with the correct mount options.

If the remount fails, failover will be attempted to resolve the condition. The following is a list of common causes for remount failure which would lead to a failover:

- corrupted file system (fsck failure)
- failure to create mount point directory
- mount point is busy
- mount failure
- LifeKeeper internal error

5.3.4.1.10.3. Maintaining a LifeKeeper Protected System

When performing shutdown and maintenance on a LifeKeeper-protected server, you must put that system's resource hierarchies in service on the backup server before performing maintenance. This process stops all activity for shared disks on the system needing maintenance.

Perform these actions in the order specified, where *Server A* is the primary system in need of maintenance and *Server B* is the backup server:

1. **Bring hierarchies in service on *Server B*.** On the backup, *Server B*, use the LifeKeeper GUI to bring in service any resource hierarchies that are currently in service on *Server A*. This will unmount any file systems currently mounted on *Server A* that reside on the shared disks under LifeKeeper protection. See Bringing a Resource In Service for instructions.

2. **Stop LifeKeeper on *Server A*.** Use the LifeKeeper command `/opt/LifeKeeper/bin/lkstop -f` to stop LifeKeeper. Your resources are now unprotected.

3. **Shut down Linux and power down *Server A*.** Shut down the Linux operating system on *Server A*, then power off the server.

4. **Perform maintenance.** Perform the necessary maintenance on *Server A*.

5. **Power on *Server A* and restart Linux.** Power on *Server A*, then reboot the Linux operating system.

6. **Start LifeKeeper on *Server A*.** Use the LifeKeeper command `/opt/LifeKeeper/bin/lkstart` to start LifeKeeper. Your resources are now protected.

7. **Bring hierarchies back in-service on *Server A*, if desired**. On *Server A*, use the LifeKeeper GUI to bring in service all resource hierarchies that were switched over to *Server B*.

5.3.4.1.10.4. Maintaining a Resource Hierarchy

You can perform maintenance on a resource hierarchy while maintaining LifeKeeper protection of all other hierarchies on the system. This involves taking the hierarchy in need of maintenance out of service and then bringing it back in-service after you complete the maintenance tasks.

To perform maintenance on a resource hierarchy:

1. **Take the hierarchy out of service.** Use the LifeKeeper GUI to take as much of the resource hierarchy out of service as you need to perform the maintenance. See Taking a Resource Out of Service for instructions.

2. **Perform maintenance.** Perform the necessary maintenance on the resource hierarchy.

3. **Restore the hierarchy.** Use the LifeKeeper GUI to bring the resource hierarchy back in service. See Bringing a Resource In Service for instructions.

5.3.4.1.10.5. Recovering After a Failover

After LifeKeeper performs a failover recovery from a primary server (*Server A*) to a backup server (*Server B*), perform the following steps:

1. **Review logs.** When LifeKeeper on *Server B* performs a failover recovery from *Server A*, status messages are displayed during the failover.

   The exact output depends upon the configuration. Some messages on failure to mount or unmount are expected and do not suggest failure of recovery. These messages, as well as any errors that occur while bringing the resource in service on *Server B*, are logged in the LifeKeeper log.

2. **Perform maintenance.** Determine and fix the cause of the failure on *Server A*. *Server A* may need to be powered down to perform maintenance.

3. **Reboot *Server A*, if necessary.** Once maintenance is complete, reboot *Server A* if necessary.

4. **Start LifeKeeper, if necessary.** If LifeKeeper is not running on *Server A*, use the command `/etc/init.d/lifekeeper start (or, systemctl start lifekeeper)` to start LifeKeeper.

5. **Move application back to *Server A*.** At a convenient time, use the LifeKeeper GUI to bring the application back into service on *Server A*. See Bringing a Resource In Service for instructions. Note that this step may be unnecessary if the application on *Server A* was configured for **Automatic Switchback**.

5.3.4.1.10.6. Removing LifeKeeper

You can uninstall LifeKeeper in a Linux environment via the command line by entering the following command.

```
/opt/LifeKeeper/bin/rmlk
```

This command uninstalls all the LifeKeeper packages and removes the directory /opt/LifeKeeper from the system. The command can be run with or without LifeKeeper running on the system. If the command is run with LifeKeeper running on all nodes in the cluster, then hierarchies are unextended and any comm paths are removed. This ensures that all remnants of the node on which the command was run are removed from the remaining nodes which effectively removes the node from the cluster. If LifeKeeper is not running at the time the command is executed, then the other nodes will have remnants remaining which may impact the running system.

This command takes 2 optional arguments:

- -j which removes any Java packages installed by LifeKeeper
- -l which remove all LifeKeeper licenses

Use this command carefully.

Note: The periodic backup of the LifeKeeper configuration via the command `lkbackup` automatically archives the results in /opt/LifeKeeper/config/. Because the `rmlk` command will remove the /opt/LifeKeeper directory you may wish to back-up the archives before running the command.

5.3.4.1.10.7. Running LifeKeeper With a Firewall

LifeKeeper for Linux can work with a firewall in place on the same server if you address the following network access requirements.

Note: If you wish to simply disable your firewall, see Disabling a Firewall below.

LifeKeeper Communication Paths

Communication paths are established between pairs of servers within the LifeKeeper cluster using specific IP addresses. Although TCP Port 7365 is used by default on the remote side of each connection as it is being created, the TCP port on the initiating side of the connection is arbitrary. The recommended approach is to configure the firewall on each LifeKeeper server to allow both incoming and outgoing traffic for each specific pair of local and remote IP addresses in the communication paths known to that system.

LifeKeeper GUI Connections

The LifeKeeper GUI uses a number of specific TCP ports, including Ports 81 and 82 as the default initial connection ports. The GUI also uses Remote Method Invocation (RMI), which uses Ports 1024 and above to send and receive objects. All of these ports must be open in the firewall on each LifeKeeper server to at least those external systems on which the GUI client will be run.

LifeKeeper IP Address Resources

The firewall should be configured to allow access to any IP address resources in your LifeKeeper hierarchies from those client systems that need to access the application associated with the IP address. Remember that the IP address resource can move from one server to another in the LifeKeeper cluster; therefore, the firewalls on all of the LifeKeeper servers must be configured properly.

LifeKeeper also uses a broadcast ping test to periodically check the health of an IP address resource. This test involves sending a broadcast ping packet from the virtual IP address and waiting for the first response from any other system on the local subnet. To prevent this test from failing, the firewall on each LifeKeeper server should be configured to allow the following types of network activity.

- Outgoing Internet Control Message Protocol (ICMP) packets from the virtual IP address (so that the active LifeKeeper server can send broadcast pings)
- Incoming ICMP packets from the virtual IP address (so that other LifeKeeper servers can receive broadcast pings)

- Outgoing ICMP reply packets from any local address (so that other LifeKeeper servers can respond to broadcast pings)
- Incoming ICMP reply packets to the virtual IP address (so that the active LifeKeeper server can receive broadcast ping replies)

LifeKeeper Data Replication

When using LifeKeeper Data Replication, the firewall should be configured to allow access to any of the ports used by nbd for replication. The ports used by nbd can be calculated using the following formula:

```
10001 + <mirror number> + <256 * i>
```

where **i** starts at zero and is incremented until the formula calculates a port number that is not in use. In use constitutes any port found defined in */etc/services*, found in the output of *netstat -an —inet*, or already defined as in use by another LifeKeeper Data Replication resource.

For example: If the mirror number for the LifeKeeper Data Replication resource is 0, then the formula would initially calculate the port to use as 10001, but that number is defined in */etc/services* on some Linux distributions as the SCP Configuration port. In this case, ***i*** is incremented by 1 resulting in Port Number 10257, which is not in */etc/services* on these Linux distributions.

Other Inter-node Communications

Each LifeKeeper server communicates using SSL connection on port 778. You can change this port using the configuration variable API_SSL_PORT in /etc/default/LifeKeeper.

Disabling a Firewall

To disable the firewall, please follow the procedure described in the manual of your distribution.

5.3.4.1.10.8. Running the LifeKeeper GUI Through a Firewall

In some situations, a LifeKeeper cluster is placed behind a corporate firewall and administrators wish to run the LifeKeeper GUI from a remote system outside the firewall.

LifeKeeper uses Remote Method Invocation (RMI) to communicate between the GUI server and client. The RMI client must to be able to make connections in each direction. Because the RMI client uses dynamic ports, you can not use preferential ports for the client.

One solution is to use ssh to tunnel through the firewall as follows:

1. Make sure your IT department has opened the secure shell port on the corporate firewall sufficiently to allow you to get behind the firewall. Often the machine IT allows you to get to is not actually a machine in your cluster but an intermediate one from which you can get into the cluster. This machine must be a Unix or Linux machine.

2. Make sure both the intermediate machine and the LifeKeeper server are running sshd (the secure shell daemon) and that X11 port forwarding is enabled (this is usually the line `X11Forwarding yes' in */etc/ssh/sshd_config*, but if you are unsure, have your IT do this for you.

3. From your Unix client in X, tunnel to the intermediate machine using:

   ```
   ssh -X -C <intermediate machine>
   ```

   The **-C** means `compress the traffic' and is often useful when coming in over slower internet links.

4. From the intermediate machine, tunnel to the LifeKeeper server using:

   ```
   ssh -X <LifeKeeper server>
   ```

   You should not need to compress this time since the intermediate machine should have a reasonably high bandwidth connection to the LifeKeeper server.

5. If all has gone well, when you issue the command:

   ```
   echo $DISPLAY
   ```

   it should be set to something like *`localhost:10.0'*. If it is not set, it is likely that X11 forwarding is disabled in one of the sshd config files.

6. Verify that you can pop up a simple *xterm* from the LifeKeeper server by issuing the command:

   ```
   /usr/X11R6/bin/xterm
   ```

7. If the *xterm* appears, you're ready to run **lkGUIapp** on the LifeKeeper server using the following command:

   ```
   /opt/LifeKeeper/bin/lkGUIapp
   ```

8. Wait (and wait some more). Java uses a lot of graphics operations which take time to propagate over a slow link (even with compression), but the GUI console should eventually appear.

5.3.4.1.10.9. Transferring Resource Hierarchies

When you need to perform routine maintenance or other tasks on a LifeKeeper Server, you can use the LifeKeeper GUI to move in-service resources to another server. To transfer in-service resource hierarchies from *Server A* to *Server B*, use the GUI to bring the hierarchies into service on *Server B*. Repeat until all of *Server A's* resources have been placed in-service on their respective backup servers. See Bringing a Resource In Service for instructions.

When all of *Server A's* resources are active on their backup server(s), you can shut down *Server A* without affecting application processing. For the maintenance period, however, the resources may not have LifeKeeper protection depending on the number of servers in the cluster.

5.3.4.1.11. Technical Notes

We strongly recommend that you read the following technical notes concerning configuration and operational issues related to your LifeKeeper environment.

LifeKeeper Features

Item	Description
Licensing	LifeKeeper requires unique runtime license keys for each server. This applies to both physical and virtual servers. A license key is required for the LifeKeeper core software, as well as for each separately packaged LifeKeeper recovery kit. The installation script installs a License Utilities package that obtains and displays the Host ID of your server during the initial install of LifeKeeper. Once your licenses have been installed the utility will return the Entitlement ID if it is available or the Host ID if it is not. The Host IDs, along with the Activation ID(s) provided with your software, are used to obtain license keys from the **SIOS Technology Corp. website**.
Large Cluster Support	LifeKeeper supports large cluster configurations, up to 32 servers. There are many factors other than LifeKeeper, however, that can affect the number of servers supported in a cluster. This includes items such as the storage interconnect and operating system or storage software limitations. Refer to the vendor-specific hardware and software configuration information to determine the maximum supported cluster size.
Internationalization and Localization	LifeKeeper for Linux v5.2 and later does support wide/multi-byte characters in resource and tag names but does not include native language message support. The LifeKeeper GUI can be localized by creating locale-specific versions of the Java property files, although currently only the English version is fully localized. However, many of the messages displayed by the LifeKeeper GUI come from the LifeKeeper core, so localization of the GUI will provide only a partial solution for users until the core software is fully localized. See also **Language Environment Effects** in Known Issues and Restrictions for additional information.
LifeKeeper MIB File	LifeKeeper can be configured to issue SNMP traps describing the events that are occurring within the LifeKeeper cluster. See the lk_configsnmp(8) man page for more information about configuring this capability. The MIB file describing the LifeKeeper traps can be found at */opt/LifeKeeper/include/LifeKeeper-MIB.txt*.
Watchdog	LifeKeeper supports the watchdog feature. The feature was tested by SIOS Technology Corp. on Red Hat EL 5.5 64-bit, and Red Hat EL 6 + softdog.
STONITH	LifeKeeper supports the STONITH feature. This feature was tested by SIOS Technology Corp. on SLES 11 on IBM x3550 ×86_64 architecture and RHEL5.5 64-bit.
XFS File System	The XFS file system does not use the fsck utility to check and fix a file system but instead relies on mount to replay the log. If there is a concern that there may be a consistency problem, the system administrator should unmount the file system by taking it out of service and run `xfs_check(8)` and `xfs_repair(8)` to resolve any issues.
IPv6	SIOS has migrated to the use of the `ip` command and away from the `ifconfig`

	command (for more information, see IPv6 Known Issue).

Tuning

Item	Description
IPC Semaphores and IPC Shared Memory	LifeKeeper requires Inter-Process Communication (IPC) semaphores and IPC shared memory. The default Red Hat values for the following Linux kernel options are located in */usr/src/linux/include/linux/sem.h* and should be sufficient to support most LifeKeeper configurations. **Option** **Required** **Default Red Hat 6.2** SEMOPM 14 32 SEMUME 20 32 SEMMNU 60 32000 SEMMAP 25 32000 SEMMNI 25 128
System File Table	LifeKeeper requires that system resources be available in order to failover successfully to a backup system. For example, if the system file table is full, LifeKeeper may be unable to start new processes and perform a recovery. In kernels with enterprise patches, including those supported by LifeKeeper, file-max, the maximum number of open files in the system, is configured by default to 1/10 of the system memory size, which should be sufficient to support most LifeKeeper configurations. Configuring the file-max value lower than the default could result in unexpected LifeKeeper failures. The value of file-max may be obtained using the following command: `cat /proc/sys/fs/file-nr` This will return three numbers. The first represents the high watermark of file table entries (i.e. the maximum the system has seen so far); the second represents the current number of file table entries, and the third is the file-max value. To adjust file-max, add (or alter) the *"fs,file-max"* value in */etc/sysctl.conf* (see sysctl.conf(5) for the format) and then run `sysctl -p` to update the system. The value in */etc/sysctl.conf* will persist across reboots.

LifeKeeper Operations

Item	Description
Kernel Debugger (kdb)	Before using the Kernel Debugger (**kdb**) or moving to **init s** on a LifeKeeper protected server, you shou either shut down LifeKeeper on that server or switch any LifeKeeper protected resources over to the ba server. Use of **kdb** with the LifeKeeper SCSI Reservation Daemons (**lkscsid** and **lkccissd**) enabled (th enabled by default) can also lead to unexpected panics.

init s	
System Panic on Locked Shared Devices	LifeKeeper uses a lock to protect shared data from being accessed by other servers on a shared SCSI LifeKeeper cannot access a device as a result of another server taking the lock on a device, then a criti has occurred and quick action should be taken or data can be corrupted. When this condition is detecte LifeKeeper enables a feature that will cause the system to panic. If LifeKeeper is stopped by some means other than `'/etc/init.d/lifekeeper stop-nofail` with shared devices still reserved (this could be caused by executing **init s**), then the LifeKeeper lo mechanism may trigger a kernel panic when the other server recovers the resource(s). All resource be placed out-of-service before stopping LifeKeeper in this manner.
nolock Option	When using storage applications with locking and following recommendations for the NFS mount option requires the additional nolock option be set, e.g. `rw,nolock,bg,hard,nointr,tcp,nfsvers=3,timeo=600,rsize=32768,wsize=32768,act`
Recovering Out-of-Service Hierarchies	As part of the recovery following the failure of a LifeKeeper server, resource hierarchies that were confi the failed server but which were not *in-service* anywhere at the time of the server failure are recovered highest priority alive server at the time of the failure. This is the case no matter where the *out-of-service* hierarchy was last in service, including the failed server, the recovering server, or some other server in cluster.
Coexistence with Linux Firewalls	The firewall is enabled upon installation. After installation is complete, the firewall should be modifi LifeKeeper will function if a host firewall is enabled. However, unless absolutely necessary, it is recommended that the firewall be disabled and that the LifeKeeper protected resources reside beh another shielding firewall. If LifeKeeper must coexist on firewall enabled hosts, then the specific ports that LifeKeeper is using to be opened. Please note that LifeKeeper uses specific ports for communication paths, GUI, IP an Replication. Refer to Running LifeKeeper with a Firewall for details. To disable or modify the firewall please refer to the documentation for your OS distribution.
Coexistence with SELinux	Disable SELinux. To Disable SELinux, please refer to the documentation for your OS distribution. AppArmor (for distributions that use this security model) may be enabled.
Suid Mount Option	The suid mount option is the default when mounting as *root* and is not written to the */etc/mtab* by the mo command. The suid mount option is not needed in LifeKeeper environments.

Server Configuration

Item	Description
BIOS Updates	The latest available BIOS should always be installed on all LifeKeeper servers.

LifeKeeper Version 8.2.0 and Later GUI Requirement

64-bit versions of any PAM related packages will be required for the LifeKeeper GUI Client to successfully authenticate users.

Confirm Failover and Block Resource Failover Settings

Make sure you review and understand the following descriptions, examples and considerations before setting the **Confirm Failover** or **Block Resource Failover** in your LifeKeeper environment. These settings are available from the command line or on the **Properties** panel in the LifeKeeper GUI.

Confirm Failover On:

Definition – Enables manual failover confirmation from *System A* to *System B* (where *System A* is the server whose properties are being displayed in the Properties Panel and *System B* is the system to the left of the checkbox). If this option is set on a system, it will require a manual confirmation by a system administrator before allowing LifeKeeper to perform a failover recovery of a system that it detects as failed.

Use the **`lk_confirmso`** command to confirm the failover. By default, the administrator has 10 minutes to run this command. This time can be changed by modifying the **CONFIRMSOTO** setting in */etc/default/LifeKeeper*. If the administrator does not run the **`lk_confirmso`** command within the time allowed, the failover will either proceed or be blocked. By default, the failover will proceed. This behavior can be changed by modifying the **COMFIRMSODEF** setting in */etc/default/LifeKeeper*.

Example: If you wish to block automatic failovers completely, then you should set the **Confirm Failover On** option in the **Properties** panel and also set **CONFIRMSODEF** to **1** (block failover) and **CONFIRMSOTO** to **0** (do not wait to decide on the failover action).

When to select this setting:

This setting is used in most Disaster Recovery and other WAN configurations where the configuration does not include redundant heartbeat communications paths.

Open the **Properties** page from one server and then select the server that you want the **Confirm Failover flag** to be set on.

Set Block Resource Failover On:

Definition – By default, all resource failures will result in a recover event that will attempt to recover the failed resource on the local system. If local recovery fails or is not enabled, then LifeKeeper transfers the resource hierarchy to the next highest priority system for which the resource is defined. However, if this setting is selected on a designated system(s), all resource transfers due to a resource failure will be blocked from the given system.

When the setting is enabled, the following message is logged:

Local recovery failure, failover blocked, MANUAL INTERVENTION REQUIRED

NFS Client Options

When setting up a LifeKeeper protected NFS server, how the NFS clients connect to that server can make a significant impact on the speed of reconnection on failover.

NFS Client Mounting Considerations

An NFS Server provides a network-based storage system to client computers. To utilize this resource, the client systems must “mount” the file systems that have been NFS exported by the NFS server. There are several options that system administrators must consider on how NFS clients are connected to the LifeKeeper protected NFS resources.

UDP or TCP?

The NFS Protocol can utilize either the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP). NFS has historically used the UDP protocol for client-server communication. One reason for this is that it is easier for NFS to work in a stateless fashion using the UDP protocol. This “statelessness” is valuable in a high availability clustering environment, as it permits easy reconnection of clients if the protected NFS server resource is switched between cluster hosts. In general, when working with a LifeKeeper protected NFS resource, the UDP protocol tends to work better than TCP.

Sync Option in /etc/exports

Specifying “*sync*” as an export option is recommended for LifeKeeper protected NFS resources. The “*sync*” option tells NFS to commit writes to the disk before sending an acknowledgment back to the NFS client. The contrasting “*async*” option is also available, but using this option can lead to data corruption, as the NFS server will acknowledge NFS writes to the client before committing them to disk. NFS clients can also specify “*sync*” as an option when they mount the NFS file system.

Red Hat EL6 (and Fedora 14) Clients with Red Hat EL6 NFS Server

Due to what appears to be a bug in the NFS server for Red Hat EL6, NFS clients running Red Hat EL6 (and Fedora 14) cannot specify both an NFS version (*nfsvers*) and UDP in the mount command. This same behavior has been observed on an Ubuntu10.10 client as well. This behavior is not seen with Red Hat EL5 clients when using a Red Hat EL6 NFS server, and it is also not seen with any clients using a Red Hat EL5 NFS server. The best combination of NFS mount directives to use with Red Hat EL6 (Fedora 14) clients and a Red Hat EL 6 NFS server is:

```
mount <protected-IP>:<export> <mount point>
-o nfsvers=2,sync,hard,intr,timeo=1
```

- This combination produces the fastest re-connection times for the client in case of a switchover or failover of the LifeKeeper protected NFS server.

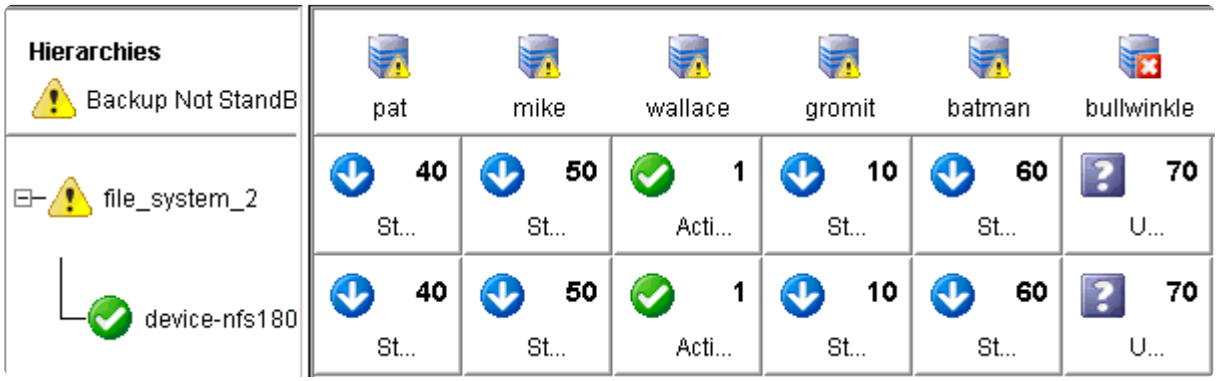
Red Hat EL5 NFS Clients with a Red Hat EL6 NFS Server

The best combination of options when using NFS clients running Red Hat EL5 with a Red Hat EL6 NFS server for fast reconnection times is:

```
mount <protected-IP>:<export> <mount point>
-o nfsvers=3,sync,hard,intr,timeo=1,udp
```

5.3.4.2. Cluster Example

Expanded Multicluster Example



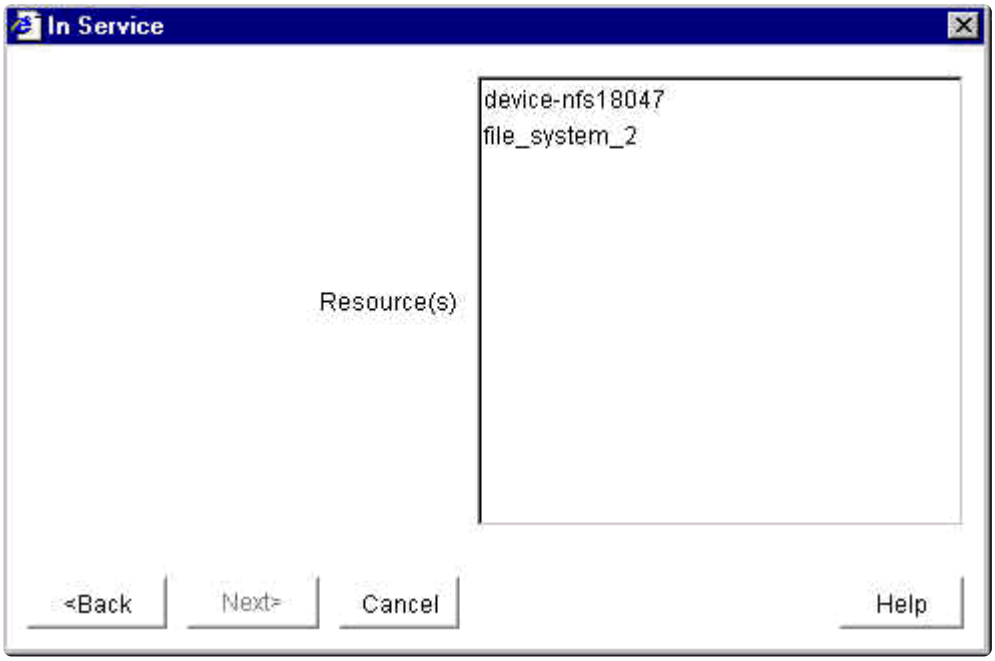
5.3.4.3. Dialogs

In Service Dialog

Select a Server. The first dialog provides a drop-down list box containing the names of servers in your LifeKeeper cluster. Select the **Server** where you want to bring the resource instance into service. Click on the **Next** button to proceed to the next dialog.

> **Note:** If you initiated the In Service task by right-clicking from the right pane on a server-specific resource, this dialog and the next will not appear since you will have already specified the server and the resource that you want to bring into service.

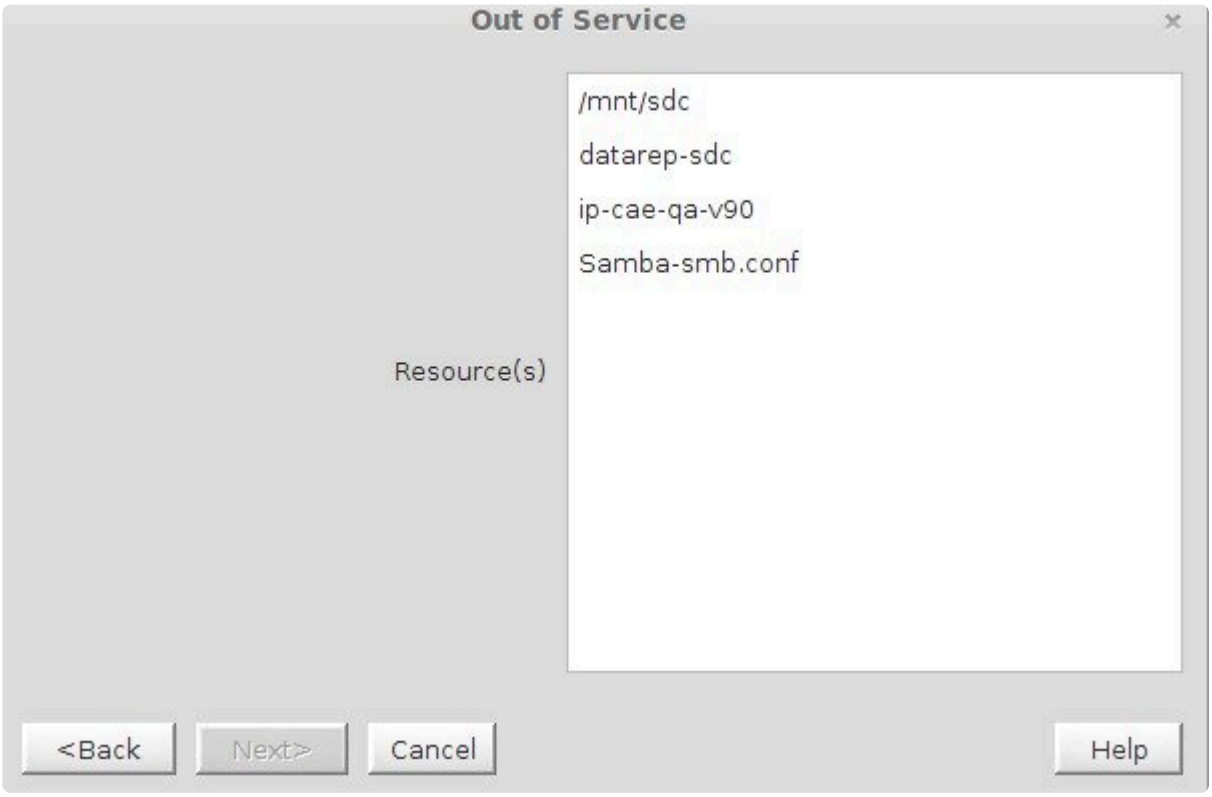
Select a Resource. The second dialog provides a drop-down list box containing the names of all the available resources on the server you selected in the previous dialog. Select the resource that you want to bring into service.



> **Note:** If you initiated the In Service task by right-clicking from the left pane on a global resource, this dialog will not appear since you will have already identified the resource that you want to bring into service.

Out-of-Service Resource Dialog

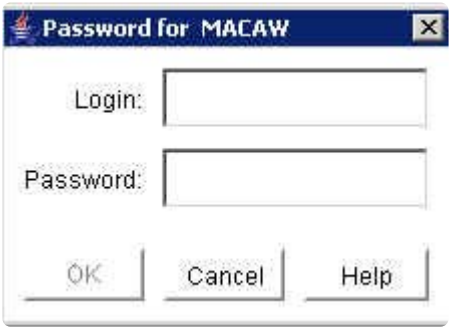
Select a Resource. This dialog provides a list containing the names of all resources that are in service in your LifeKeeper cluster. Select the resource that you want to take out of service.



Note: If you initiated the Out-of-Service task by right-clicking from the left pane on an in-service global resource or from the right pane on an in-service server-specific resource instance, this dialog will not appear since you will have already specified the resource that you want to take out of service.

Password Dialog

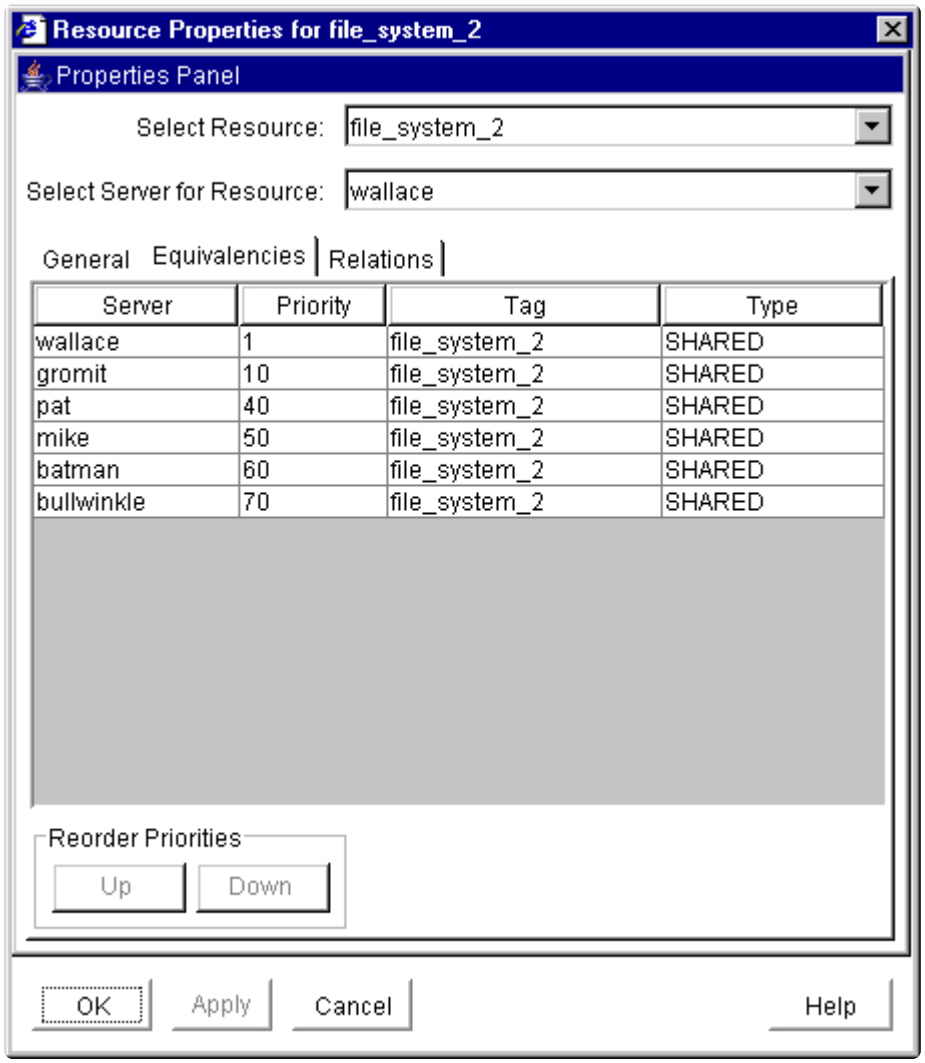
This dialog is displayed if the initial login name or password entered in the Cluster Connect dialog is invalid.



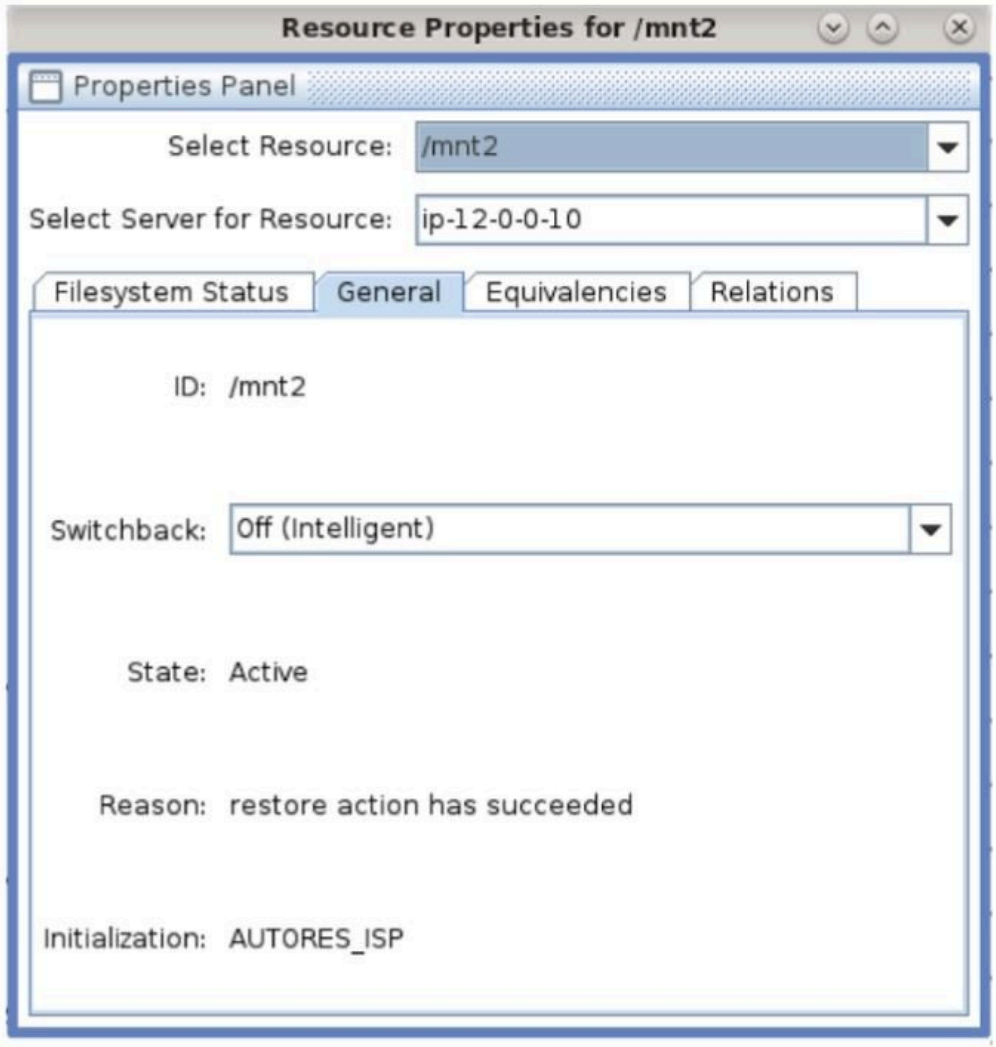
Login. The login name of a user with LifeKeeper authorization on the specified server.

Password. The password that authorizes the specified login on the server.

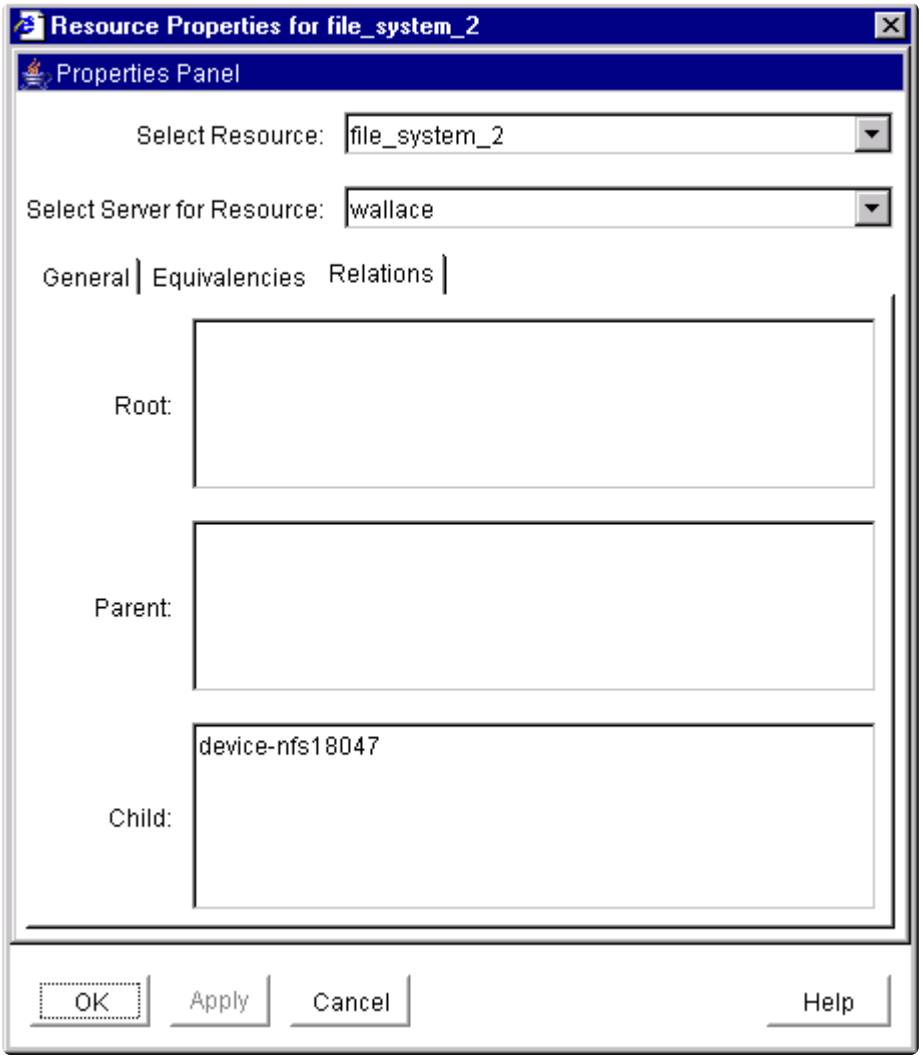
Resource Properties – Equivalencies



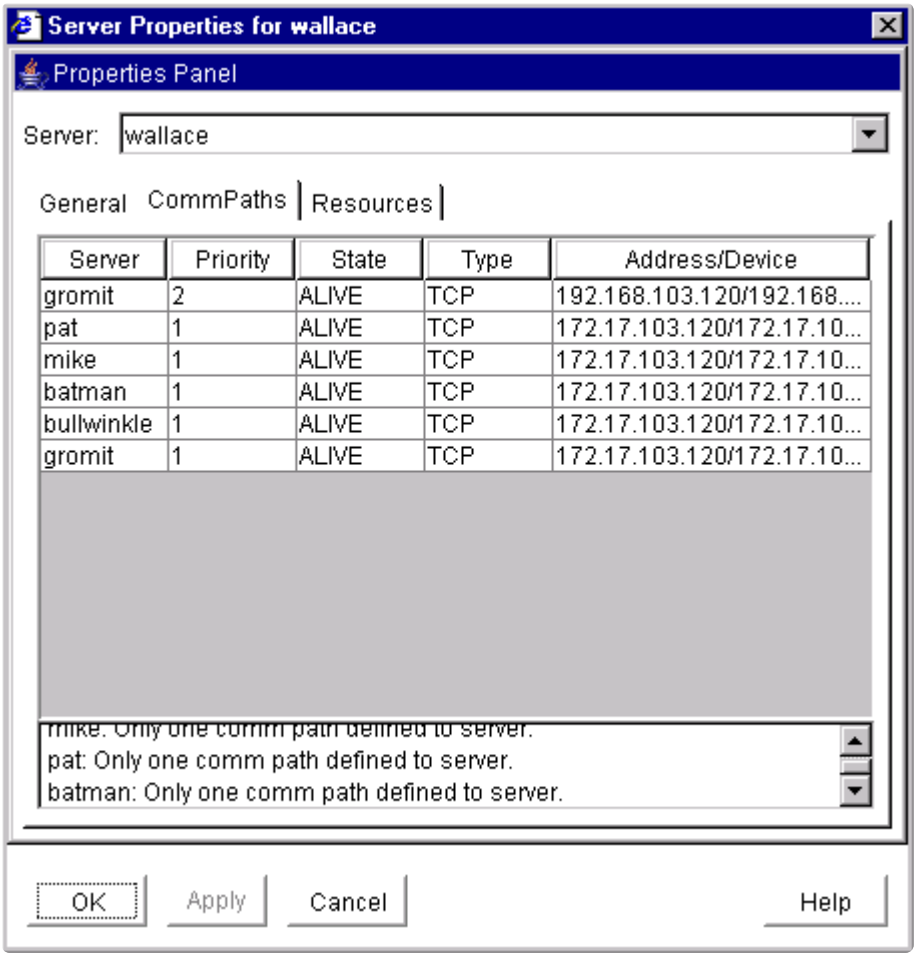
Resource Properties – General



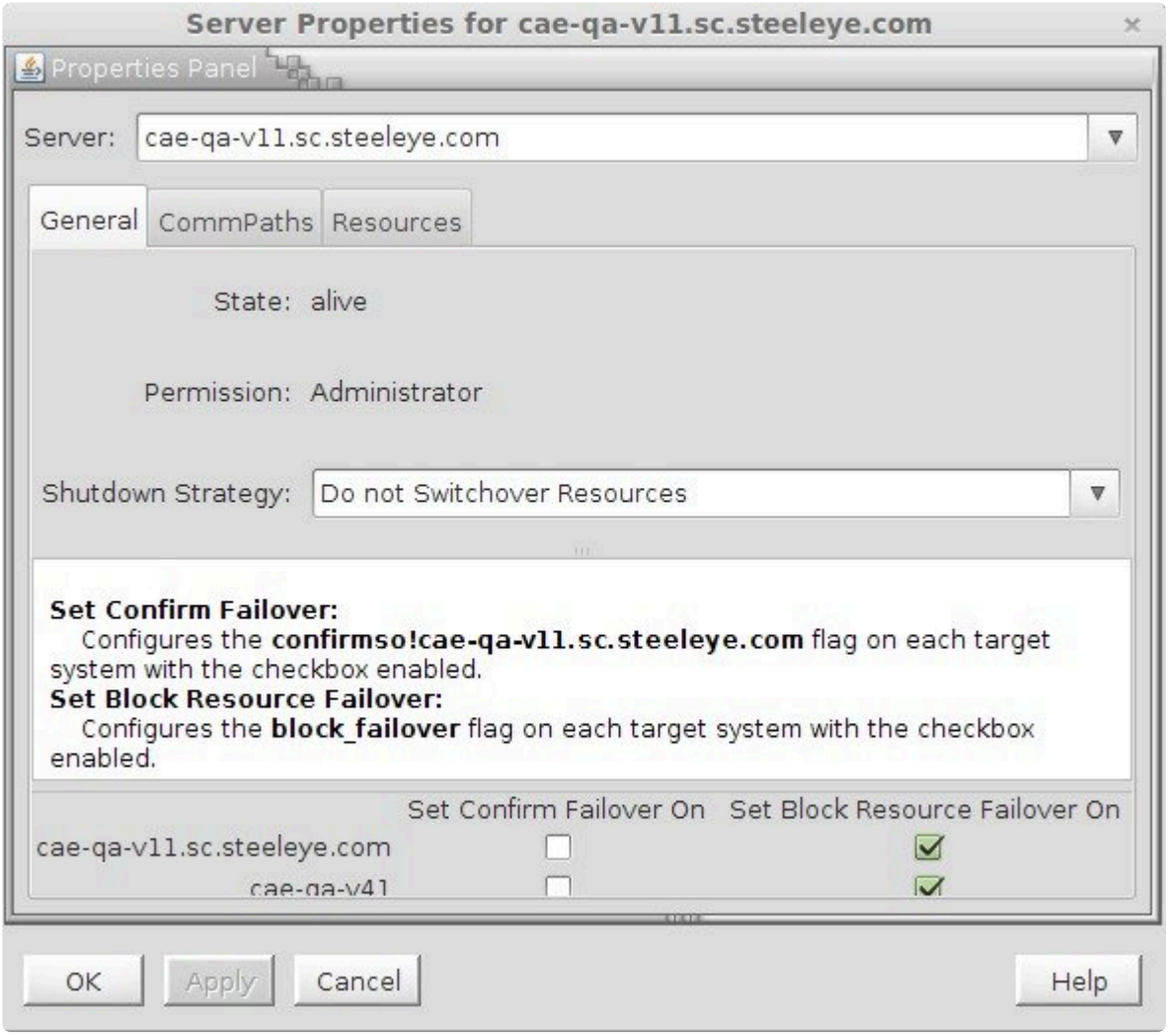
Resource Properties – Relations



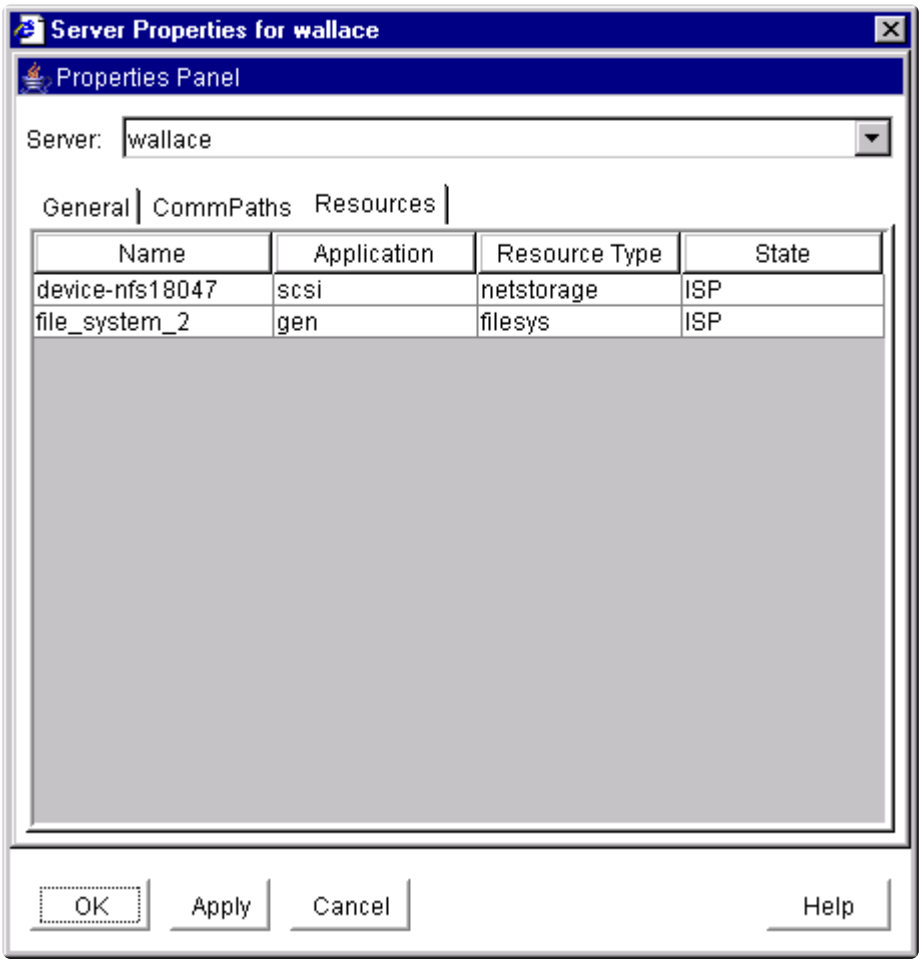
Server Properties – Commpath



Server Properties – General



Server Properties – Resource



5.3.5. Troubleshooting

The Message Catalog provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the following individual Message Catalogs:

- Core Message Catalog
- DB2 Kit Message Catalog
- DMMP Kit Message Catalog
- Recovery Kit for EC2 Message Catalog
- File System Kit Message Catalog
- Gen/App Kit Message Catalog
- IP Kit Message Catalog
- Oracle Listener Kit Message Catalog
- Oracle Kit Message Catalog
- SCSI Kit Message Catalog
- DataKeeper Kit Message Catalog
- Quick Service Protection Kit Message Catalog
- GUI Message Catalog

In addition to utilizing the Message Catalog described above, the following topics detail troubleshooting issues, restrictions, etc., that may be encountered:

- Common Causes of Failover
- Known Issues and Restrictions
- GUI Troubleshooting
- Communication Paths Going Up and Down

5.3.5.1. Common Causes of an SPS Initiated Failover

In the event of a failure, SPS has two methods of recovery: local recovery and inter-server recovery. If local recovery fails, a “failover” is implemented. A failover is defined as automatic switching to a backup server upon the failure or abnormal termination of the previously active application, server, system, hardware component or network. Failover and switchover are essentially the same operation, except that failover is automatic and usually operates without warning, while switchover requires human intervention. This automatic failover can occur for a number of reasons. Below is a list of the most common examples of an SPS initiated failover.

Server Level Causes

Server Failure

SPS has a built-in heartbeat signal that periodically notifies each server in the configuration that its paired server is operating. A failure is detected if a server fails to receive the heartbeat message.

- Primary server loses power or is turned off.
- CPU Usage caused by excessive load — Under very heavy I/O loads, delays and low memory conditions can cause system to become unresponsive such that SPS may detect a server as down and initiate a failover.
- Quorum/Witness – As part of the I/O fencing mechanism of quorum/witness, when a primary server loses quorum, a “*fastboot*”:, “*fastkill*” or “*osu*” is performed (*based on settings*) and a failover is initiated. When determining when to fail over, the witness server allows resources to be brought in service on a backup server only in cases where it verifies the primary server has failed and is no longer part of the cluster. This will prevent failovers from happening due to simple communication failures between nodes when those failures don’t affect the overall access to, and performance of, the in-service node.

Relevant Topics
Supported Storage List
Server Failure Recovery Scenario
Tuning the LifeKeeper Heartbeat
Quorum/Witness

Communication Failures/Network Failures

SPS sends the heartbeat between servers every five seconds. If a communication problem causes the heartbeat to skip two beats but it resumes on the third heartbeat, SPS takes no action. However, if the communication path remains dead for three beats, SPS will label that communication path as dead but

will initiate a failover only if the redundant communication path is also dead.

- Network connection to the primary server is lost.
- Network latency.
- Heavy network traffic on a TCP comm path can result in unexpected behavior, including false failovers and LifeKeeper initialization problems.
- Using STONITH, when SPS detects a communication failure with a node, that node will be powered off and a failover will occur.
- Failed NIC.
- Failed network switch.
- Manually pulling/removing network connectivity.

Relevant Topics
Creating a Communication Path
Tuning the LifeKeeper Heartbeat
Network Configuration
Verifying Network Configuration
LifeKeeper Event Forwarding via SNMP
Network-Related Troubleshooting
Running LifeKeeper With a Firewall
STONITH

Split-Brain

If a single comm path is used and the comm path fails, then SPS hierarchies may try to come into service on multiple systems simultaneously. This is known as a false failover or a “split-brain” scenario. In the “**split-brain**” scenario, each server believes it is in control of the application and thus may try to access and write data to the shared storage device. To resolve the split-brain scenario, SPS may cause servers to be powered off or rebooted or leave hierarchies out-of-service to assure data integrity on all shared data. Additionally, heavy network traffic on a TCP comm path can result in unexpected behavior, including false failovers and the failure of LifeKeeper to initialize properly.

The following are scenarios that can cause split-brain:

- Any of the comm failures listed above
- Improper shutdown of LifeKeeper

- Server resource starvation
- Losing all network paths
- DNS or other network glitch
- System lockup/thaw

Resource Level Causes

SPS is designed to monitor individual applications and groups of related applications, periodically performing local recoveries or notifications when protected applications fail. Related applications, by example, are hierarchies where the primary application depends on lower-level storage or network resources. SPS monitors the status and health of these protected resources. If the resource is determined to be in a failed state, an attempt will be made to restore the resource or application on the current system (in-service node) without external intervention. If this local recovery fails, a resource failover will be initiated.

Application Failure

- An application failure is detected, but the local recovery process fails.
- Remove Failure – During the resource failover process, certain resources need to be removed from service on the primary server and then brought into service on the selected backup server to provide full functionality of the critical applications. **If this remove process fails, a reboot of the primary server will be performed** resulting in a complete server failover.

  Examples of remove failures:

- ◦ Unable to unmount file system
- ◦ Unable to shut down protected application (oracle, mysql, postgres, etc)

Relevant Topics
File System Health Monitoring
Resource Error Recovery Scenario

File System

- Disk Full — SPS's File System Health Monitoring can detect disk full file system conditions which may result in failover of the file system resource.
- Unmounted or Improperly Mounted File System — User manually unmounts or changes options on an in-service and LK protected file system.
- Remount Failure — The following is a list of common causes for remount failure which would lead to a failover:

- ◦ corrupted file system (fsck failure)
- ◦ failure to create mount point directory
- ◦ mount point is busy
- ◦ mount failure
- ◦ SPS internal error

Relevant Topics
File System Health Monitoring

IP Address Failure

When a failure of an IP address is detected by the IP Recovery Kit, the resulting failure triggers the execution of the IP local recovery script. SPS first attempts to bring the IP address back in service on the current network interface. If the local recovery attempt fails, SPS will perform a failover of the IP address and all dependent resources to a backup server. During failover, the remove process will un-configure the IP address on the current server so that it can be configured on the backup server. **Failure of this remove process will cause the system to reboot**.

- IP conflict
- IP collision
- DNS resolution failure
- NIC or Switch Failures

Relevant Topics
Creating Switchable IP Address
IP Local Recovery

Reservation Conflict

- A reservation to a protected device is lost or stolen
- Unable to regain reservation or control of a protected resource device (caused by manual user intervention, HBA or switch failure)

Relevant Topics
SCSI Reservations
Disabling Reservations

SCSI Device

- Protected SCSI device could not be opened. The device may be failing or may have been removed from the system.

5.3.5.2. Known Issues and Restrictions

Included below are the restrictions or known issues open against LifeKeeper for Linux, broken down by functional area.

5.3.5.2.1. Product Incompatibility Issue / Restriction

Description
SIOS AppKeeper Because SIOS LifeKeeper monitors and remediates OS and application services in clustered environments in AWS EC2, the use of SIOS AppKeeper in addition to SIOS LifeKeeper in those environments is not recommended or supported.

5.3.5.2.2. Installation – Known Issues / Restrictions

Description
SAP Recovery Kit is Automatically Installed During Upgrade if SAP DB Recovery Kit has been Previously Installed If the SAP DB/SAP MaxDB Recovery Kit is installed, the SAP Recovery Kit will automatically be selected for installation when upgrading LifeKeeper even if it was not previously installed. This will be fixed in a future version of LifeKeeper. **Workaround:** When upgrading a system with the SAP DB/SAP MaxDB Recovery Kit installed where the user does not intend to install the SAP Recovery Kit, manually deselect the SAP Recovery Kit (under Application Suite → SAP Recovery Kit) for installation.
A functional yum or zypper configuration is required for the successful installation of LfieKeeper. A misconfigured or non-functional yum or zipper configuration can result in the failure of the LifeKeeper installation script. Output such as the following may be seen: `Install LifeKeeper and dependent packages done.` `Setup high availability data replication features.. done.` `Setup NFS high availability features…` `Configure LifeKeeper management group` `Setup failed.` `Fix the problem and try again.` `sed: can’t read /etc/default/LifeKeeper: No such file or directory` `/tmp/mnt/setuplibs/install.sh: line 118: /config/system_information: No such file or directory` `/tmp/mnt/setuplibs/install.sh: line 122: /config/system_information: No such file or directory` `/tmp/mnt/setuplibs/install.sh: line 122: /config/system_information: No such file or directory` `/tmp/mnt/setuplibs/install.sh: line 122: /config/system_information: No such file or directory` Additionally, the following may be seen in the /var/log/LK_install.log file:

```
Abort, retry, ignore? [a/r/i/? shows all options] (a): a
done.
```

In Release 7.4 and forward, relocation of the SIOS product RPM packages is no longer supported.

Linux Dependencies

Installing SIOS Protection Suite for Linux including the optional Recovery Kits requires several packages which have dependencies. If the package manager is properly configured, the required package is automatically installed by the package manager.

If the installation can not be done automatically, the setup script will be interrupted. After manually installing dependent packages (see Linux dependencies for details), re-execute the setup script.

Note: If the installation of these dependent packages is not completed successfully, it could affect the ability to start SIOS Protection Suite for Linux as well as the loading of the SIOS Protection Suite for Linux GUI.

The multipathd daemon will log errors in the error log when the nbd driver is loaded as it tries to scan the new devices

Solution: To avoid these errors in the log, add **devnode "^nbd"** to the blacklist in */etc/multipath.conf*.

`mksh` conflicts with SIOS Protection Suite for Linux setup needing ksh

If the `mksh` package is installed, the SIOS Protection Suite for Linux setup will fail indicating a package conflict. The SIOS Protection Suite for Linux requires the ksh package.

Workaround: On RHEL, CentOS or Oracle Linux, remove the mksh package and install the ksh package. After installing the ksh package, re-run the SIOS Protection Suite for Linux setup.

Example:

1. Remove the mksh package
   ```
   yum remove mksh
   ```

2. Install the ksh package
   ```
   yum install ksh
   ```

3. Re-run setup

Unexpected termination of daemons

Daemons using IPC terminate unexpectedly after update to Red Hat Enterprise Linux 7.2 and Red Hat 7.2 derivative systems. A new systemd feature was introduced in Red Hat Enterprise Linux 7.2 related to the cleanup of all allocated inter-process communication (IPC) resources when the last user session finishes. A session can be an administrative cron job or an interactive session. This behavior can cause daemons running under the same user, and using the same resources, to terminate unexpectedly.

To work around this problem, edit the file /etc/systemd/logind.conf and add the following line:
`RemoveIPC=no`

Then, execute the following command, so that the change is put into effect:
`systemctl restart systemd-logind.service`

After performing these steps, daemons no longer crash in the described situation. Applications (such as MQ, Oracle, SAP, etc) using shared memory and semaphores may be affected by this issue and therefore require this change.

Re-execution of LifeKeeper's "setup" script may be required after updating the kernel

In Red Hat Enterprise Linux 7.x/CentOS 7.x/Oracle Linux 7.x environment, DataKeeper may not function properly when updating the kernel to 7.3 or later.

Workaround:
The problem will be solved by re-running the "setup" script that was executed when installing LifeKeeper on the updated system.

Description:
A loaded kernel module cannot be used after updating the kernel to 7.3 or later due to the compatibility of OS kernel modules.
DataKeeper uses a kernel module called nbd.ko, which accesses disks through the network. A correct nbd.ko module is installed when executing setup script for LifeKeeper installation.
nbd.ko for the new kernel will be installed by executing setup script again after updating the kernel.

Unnecessary warning, displayed from the setup script

Depending on the installation status of the LifeKeeper packages, the following warning is displayed when the setup script is executed.

•
Found changes in following files.
These files are overwritten in install process.

If you want to keep changes, please backup these files.

missing /opt/LifeKeeper/lkadm/subsys/scsi/DEVNAME
missing /opt/LifeKeeper/lkadm/subsys/scsi/DEVNAME/bin
missing /opt/LifeKeeper/subsys/scsi/resources/DEVNAME

This warning is caused by package management issues, but does not affect the setup and operation of LifeKeeper.

This issue will be fixed in a future version.

5.3.5.2.3. LifeKeeper Core – Known Issues / Restrictions

Description
If there is a problem with a network connection, stop the service that automatically configures the network In an environment where IP addresses are protected using LifeKeeper, IP resources may conflict with daemons and services that automatically configure the network, such as avahi-daemon. If there is a problem when restoring communication paths or starting IP resources, stop the services that automatically configure the network.
Do not disconnect the network using the ifconfig down or the ip link down command When a network interface is disconnected using the `ifconfig down` or `ip link down` command, a communication path may not be restored after reconnecting, if a virtual IP resource is configured on the interface.
LifeKeeper does not start with systemd target set to multi-user In order for LifeKeeper to function properly, when running `systemctl set-default` or `systemctl isolate`, you must use the `lifekeeper-graphical.target` (for graphical mode) or `lifekeeper-multi-user.target` (for console mode). Do not use the normal `graphical.target` and `multi-user.target` systemd targets.
DataKeeper Disk UUID Restriction Starting in version 9.5.0, DataKeeper can no longer mirror disks that do not present a UUID to the operating system. The best way to mirror such a disk is to partition it with a GPT (GUID Partition Table). The “parted” tool can be used for this purpose. **Caution:** partitioning a disk will destroy any data that is already stored on the disk. **Workaround:** See DataKeeper for Linux Troubleshooting
On SLES 15, LifeKeeper logging may not appear in the LifeKeeper log file following a log rotation If `logrotate` is run on the command line or if a background log rotation occurs due to the size of the log, LifeKeeper will stop logging. **Workaround:** Run `systemctl reload rsyslog` to resume LifeKeeper logging.
When running `lkbackup`, an error may appear in the LifeKeeper log `lkbackup[30809: ERROR:lkbackup:::010064:Possible Configuration error: More than one LifeKeeper version is installed on this host` This error message can safely be ignored.
File system labels should not be used in large configurations

The use of file system labels can cause performance problems during boot-up with large clusters. The problems are generally the result of the requirement that to use labels all devices connected to a system must be scanned. For systems connected to a SAN, especially those with LifeKeeper where accessing a device is blocked, this scanning can be very slow.

To avoid this performance problem on Red Hat systems, edit /etc/fstab and replace the labels with the path names.

`lkscsid` will halt the system when it should issue a sendevent when a disk fails in certain environments

When `lkscsid` detects a disk failure, it should, by default, issue a sendevent to LifeKeeper to recover from the failure. The `sendevent` will first try to recover the failure locally and if that fails, will try to recover the failure by switching the hierarchy with the disk to another server. On some versions of Linux (RHEL 5 and SLES11), `lkscsid` will not be able to issue the `sendevent` but instead will immediately halt the system. This only affects hierarchies using the SCSI device nodes such as */dev/sda* in a shared storage configuration.

DataKeeper Create Resource fails

When using DataKeeper in certain environments (e.g., virtualized environments with IDE disk emulation, or servers with HP CCISS storage), an error may occur when a mirror is created:

```
ERROR 104052: Cannot get the hardware ID of the device "/dev/hda3"
```

This is because LifeKeeper does not recognize the disk in question and cannot get a unique ID to associate with the device.

Workaround: Use a GUID Partition so that LifeKeeper can recognize the disk in question. Otherwise add a pattern for the disk(s) to the DEVNAME device_pattern file, e.g.:

```
# cat /opt/LifeKeeper/subsys/scsi/resources/DEVNAME/device_pattern
/dev/hda*
```

Specifying hostnames for API access

The key name used to store LifeKeeper server credentials must match the hostname of the other LifeKeeper server ***exactly*** (as displayed by the `hostname` command on that server). If the hostname is an FQDN, then the credential key must also be the FQDN. If the hostname is a short name, then the key must also be the short name.

Workaround: Make sure that the hostname(s) stored by credstore match the hostname exactly.

Restore of an lkbackup after a resource has been created may leave broken equivalencies

The configuration files for created resources are saved during an `lkbackup`. If a resource is created for the first time after an `lkbackup` has been taken, that resource may not be properly accounted for when restoring from this previous backup.

Solution: Restore from `lkbackup` prior to adding a new resource for the first time. If a new resource has been added after an `lkbackup`, it should either be deleted prior to performing the restore, or delete an instance of the resource hierarchy, then re-extend the hierarchy after the restore. **Note:** It is recommended

that an `lkbackup` be run when a resource of a particular type is created for the first time.

Resources removed in the wrong order during failover

In cases where a hierarchy shares a common resource instance with another root hierarchy, resources are sometimes removed in the wrong order during a cascading failover or resource failover.

Solution: Creating a common root will ensure that resource removals in the hierarchy occur from the top down.

1. Create a gen/app that always succeeds on restore and remove.
2. Make all current roots children of this new gen/app.

Note: Using */bin/true* for the restore and remove script would accomplish this.

RHEL 6.0 is NOT Recommended

SIOS strongly discourages the use of RHEL 6.0. If RHEL 6.0 is used, please understand that an OS update may be required to fix certain issues including, but not limited to:

- DMMP fails to recover from cable pull with EMC CLARiiON (*fixed in RHEL 6.1*)
- md recovery process hangs (*fixed in the first update kernel of RHEL 6.1*)

Note: In DataKeeper configurations, if the operating system is updated, a reinstall/upgrade of SIOS Protection Suite for Linux is required.

Delete of nested file system hierarchy generates “Object does not exist” message

Solution: This message can be disregarded as it does not create any issues.

filesyshier returns the wrong tag on a nested mount create

When a database has nested file system resources, the file system kit will create the file system for both the parent and the nested child. However, `filesyshier` returns only the child tag. This causes the application to create a dependency on the child but not the parent.

Solution: When multiple file systems are nested within a single mount point, it may be necessary to manually create the additional dependencies to the parent application tag using dep_create or via the UI Create Dependency.

DataKeeper: Nested file system create will fail with DataKeeper

When creating a DataKeeper mirror for replicating an existing file system, if a file system is nested within this structure, you must unmount it first before creating the File System resource.

Workaround: Manually unmount the nested file systems and remount / create each nested mount.

Changing the mount point of the device protected by Filesystem resource may lead data corruption

The mount point of the device protected by LifeKeeper via the File System resource (filesys) must not be changed. Doing so may lead to the device being mounted on multiple nodes and if a switchover is done and this could lead to data corruption.

XFS file system usage may cause quickCheck to fail.

In the case CHECK_FS_QUOTAS setting is enabled for LifeKeeper installed on Red Hat Enterprise Linux 7 / Oracle Linux 7 / CentOS 7, quickCheck fails if uquota, gquota option is set to the XFS file system resource, which is to be protected.

Solution: Use usrquota, grpquota instead of uquota, gquota for mount options of XFS file system, or, disable CHECK_FS_QUOTAS setting.

Btrfs is not supported

Btrfs (or any other SPS for Linux unsupported filesystem) cannot be used for LifeKeeper files (/opt/LifeKeeper), bitmap files if they are not in /opt/LifeKeeper, lkbackupfiles, or any other LifeKeeper related files. In addition, LifeKeeper does not support protecting Btrfs (or any other SPS for Linux unsupported filesystem) within a resource hierarchy.

Solution: A simple work around for placing /optLifeKeeper on a Btrfs file system is to add a small disk to your instances and format that disk with ext4 or xfs, and mount this filesystem as /opt/LifeKeeper.

1. Create a small disk to be used for /opt/LifeKeeper
   • A minimum of 110MB is required for software installs
   • Note: In Azure, you can create a 1 GB data disk at a minimum.
   • Note: Additional ARKs and the number of mirrors may increase the total required space.

2. Once the disk is added to the node and visible, partition the disk (or use lvm).
   • Example: gdisk /dev/sdb
   • Note: How to add a disk to your system is outside the scope of this KBA (contact your sysadmin for your environment)

3. Format the partition with a supported filesystem (see release notes: http://docs.us.sios.com/spslinux/9.4.1/en/topic/sios-protection-suite-for-linux-release-notes).
   • Example: mkfs.ext4 /dev/sdb1 (where sdb1 was created in step 2)

4. Add the newly created and formatted partition to /etc/fstab and set it to be automatically mounted on system boot.

5. Mount the new partition as /opt/LifeKeeper
   • Example: mount /dev/sdb1 /opt/LifeKeeper
   • verify filesystem is mounted

6. Install SPS-L

7. After the installation, edit /etc/fstab and add the entry, so the disk can be mounted on reboot.
   • Example: /dev/sdb /opt/LifeKeeper ext4

SLES12 SP1 or later on AWS

The following restrictions apply with SLES12 SP1 or later on AWS:

• Cannot set static routing configuration
Automatic IP address configuration via DHCP does not work if a static routing configuration is set in /etc/sysconfig/network/routes. This causes the network not to start correctly.

Solution: Update the routing information in the configuration file by modifying the “ROUTE” parameter in /etc/sysconfig/network/ifroute-ethX

• Hostname is changed even if the “Change Hostname via DHCP” setting is disabled.
The LK service does not work properly if the hostname is rewritten. In SLES12 SP1 or later on AWS, the hostname is changed even after the “Change Hostname via DHCP” setting is disabled.

Solution:

º Update /etc/cloud/cloud.cfg to comment out the “update_hostname” parameter

º Update /etc/cloud/cloud.cfg to set the preserve_hostname parameter to “true”

º Update /etc/sysconfig/network/dhcp to set the DHCLIENT_SET_HOSTNAME parameter to “no”

Shutdown Strategy set to “Switchover Resources” may fail when using Quorum/Witness Kit in Witness mode

Hierarchy switchover during LifeKeeper shutdown may fail to occur when using the Quorum/Witness Kit in Witness mode.

Workaround: Manually switchover resource hierarchies before shutdown.

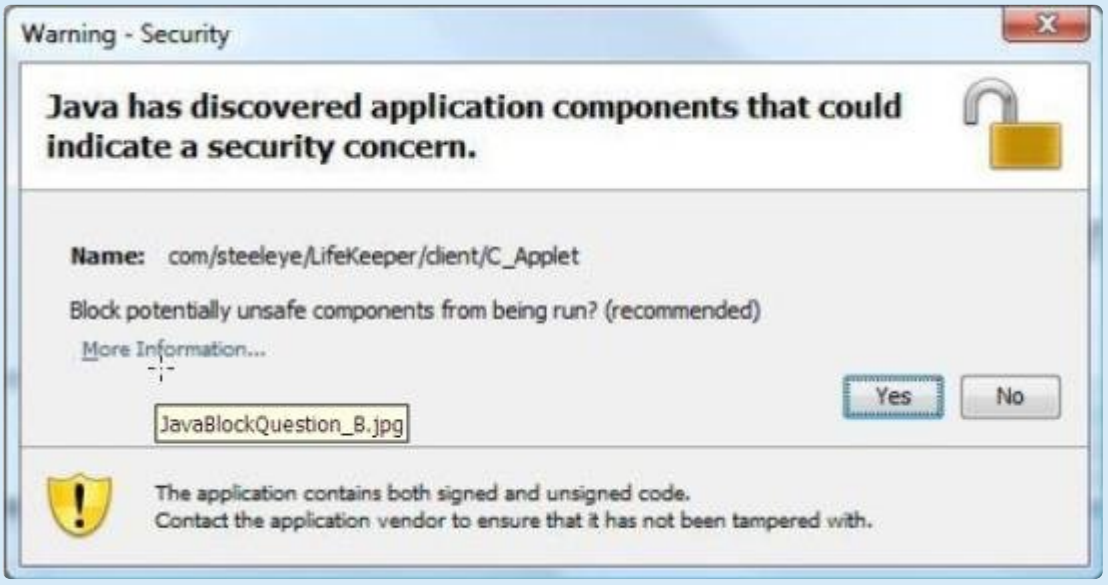
Edit /etc/services If the following entry in /etc/service is deleted, LifeKeeper cannot start up. lcm_server 7365/tcp Don't delete this entry when editing the file.
Any storage unit which returns a string including a space for the SCSI ID cannot be protected by LifeKeeper.
Automatic recovery of network may fail when link status goes down on RHEL 6.x systems when using bonded interfaces. The network doesn't recover automatically when using bonded interfaces when the link status is lost and then restored on RHEL 6.x. Loss of the link status can occur via a bad network cable, bad switch or hub or a cable reconnection and ifdown -> ifup. To recover from this status, restart the network manually by executing the following command by a user with root authorization. # service network restart Please note that this problem is already corrected for RHEL7. Also, this problem doesn't occur with SLES.
Using bind mounts is not supported Bind mounts (mount —bind) cannot be used for the file system protected by LifeKeeper.
On SLES running on AWS or Azure, change the network interface configuration file in order to prevent a cloud network plug-in from removing the virtual IP address. Click here for more details.

5.3.5.2.4. Internet/IP Licensing – Known Issues / Restrictions

Description
***/etc/hosts* settings dependency** **/etc/hosts settings:** When using internet-based licensing (IPv4 address), the configuration of */etc/hosts* can negatively impact license validation. If LifeKeeper startup fails with: • `Error in obtaining LifeKeeper license key:` `Invalid host.` `The hostid of this system does not match the hostid specified in the license file.` and the listed internet hostid is correct, then the configuration of */etc/hosts* may be the cause. To correctly match */etc/hosts* entries, IPv4 entries must be listed before any IPv6 entries. To verify if the */etc/hosts* configuration is the cause, run the following command: • `/opt/LifeKeeper/bin/lmutil lmhostid -internet -n` If the IPv4 address listed does not match the IPv4 address in the installed license file, then */etc/hosts* must be modified to place IPv4 entries before IPv6 entries to return the correct address.

5.3.5.2.5. GUI – Known Issues / Restrictions

Description
GUI login prompt may not re-appear when reconnecting via a web browser after exiting the GUI When you exit or disconnect from the GUI applet and then try to reconnect from the same web browser session, the login prompt may not appear. **Workaround:** Close the web browser, re-open the browser and then connect to the server. When using the Firefox browser, close all Firefox windows and re-open.
GUI does not immediately update IP resource state after network is disconnected and then reconnected When the primary network between servers in a cluster is disconnected and then reconnected, the IP resource state on a remote GUI client may take as long as 1 minute and 25 seconds to be updated due to a problem in the RMI/TCP layer.
Java Mixed Signed/Unsigned Code Warning – When loading the LifeKeeper Java GUI client applet from a remote system, the following security warning may be displayed:    Enter “**Run**” and the following dialog will be displayed:



Block? Enter "**No**" and the LifeKeeper GUI will be allowed to operate.

Recommended Actions: To reduce the number of security warnings, you have two options:

1. Check the "**Always trust content from this publisher**" box and select "**Run**". The next time the LifeKeeper GUI Java client is loaded, the warning message will not be displayed.

   or

2. Add the following entry to your Java "**deployment.properties**" file to eliminate the second dialog about blocking. The security warning will still be displayed when you load the Java client, however, the applet will not be blocked and the Block "**Yes**" or "**No**" dialog will not be displayed. Please note this setting will apply to all of your Java applets.
   - `deployment.security.mixcode=HIDE_RUN`

   To bypass both messages, implement 1 and 2.

steeleye-lighttpd process fails to start if Port 778 and 779 are in use

If a process is using Port 778 and 779 when steeleye-lighttpd starts up, steeleye-lighttpd fails which can cause GUI connect failiures and resource hierarchy extend issues.

Solution: Set the following tunables on all nodes in the cluster and then restart LifeKeeper on all the nodes:

- Add the following lines to */etc/default/LifeKeeper:*

-

```
API_SSL_PORT=port_number
LKAPI_WEB_PORT=port_number
```

where port_number is the new port to use.

5.3.5.2.6. Data Replication – Known Issues / Restrictions

Description
Partitions with an odd number of sectors are not supported when running kernel 4.12 or later The use of a partition with an odd number of sectors is not supported in a DataKeeper mirror in environments running kernel 4.12 or later. This is due to an issue where a resync may fail when attempting to write past the end of the disk.
Important reminder about DataKeeper for Linux asynchronous mode in an LVM over DataKeeper configuration Kernel panics may occur in configurations were LVM resources sit above multiple asynchronous mirrors. In these configurations data consistency may be an issue if a panic occurs. Therefore the required configurations are a single DataKeeper mirror or multiple synchronous DataKeeper mirrors.
In symmetric active SDR configurations with significant I/O traffic on both servers, the filesystem mounted on the mirror stops responding and eventually the whole system hangs Due to the single threaded nature of the Linux buffer cache, the buffer cache flushing daemon can hang trying to flush out a buffer which needs to be committed remotely. While the flushing daemon is hung, all activities in the Linux system with dirty buffers will stop if the number of dirty buffers goes over the system accepted limit (set in */proc/sys/kernel/vm/bdflush*). Usually this is not a serious problem unless something happens to prevent the remote system from clearing remote buffers (e.g. a network failure). LifeKeeper will detect a network failure and stop replication in that event, thus clearing a hang condition. However, if the remote system is also replicating to the local system (i.e. they are both symmetrically replicating to each other), they can deadlock forever if they both get into this flushing daemon hang situation. The deadlock can be released by manually killing the nbd-client daemons on both systems (which will break the mirrors). To avoid this potential deadlock entirely, however, symmetric active replication is not recommended.
Mirror breaks and fills up */var/log/messages* with errors This issue has been seen occasionally (on Red Hat EL 6.x and CentOS 6.x) during stress tests with induced failures, especially in killing the `nbd-server` process that runs on a mirror target system.

Upgrading to the latest kernel for your distribution may help lower the risk of seeing this particular issue, such as kernel-2.6.32-131.17.1 or later. Rebooting the source system will clear up this issue.

With the default kernel that comes with CentOS 6 (2.6.32-71), this issue may occur much more frequently (even when the mirror is just under a heavy load).

Note: Beginning with SPS 8.1, when performing a kernel upgrade on Red Hat Enterprise Linux systems, it is no longer a requirement that the setup script (`./setup`) from the installation image be rerun. Modules should be automatically available to the upgraded kernel without any intervention as long as the kernel was installed from a proper Red Hat package (rpm file).

High CPU usage reported by top for md_raid1 process with large mirror sizes

With the `mdX_raid1` process (*with X representing the mirror number*), high CPU usage as reported by top can be seen on some OS distributions when working with very large mirrors (500GB or more).

Solution: To reduce the CPU usage percent, modify the chunk size to 1024 via the LifeKeeper tunable `LKDR_CHUNK_SIZE` then delete and recreate the mirror in order to use this new setting.

The use of lkbackup with DataKeeper resources requires a full resync

Although `lkbackup` will save the *instance* and *mirror_info* files, it is best practice to perform a full resync of DataKeeper mirrors after a restore from `lkbackup` as the status of source and target cannot be guaranteed while a resource does not exist.

Mirror resyncs may hang in early Red Hat/CentOS 6.x kernels with a “Failed to remove device” message in the LifeKeeper log

Kernel versions prior to version 2.6.32-131.17.1 (RHEL 6.1 kernel version 2.6.32-131.0.15 before update, etc) contain a problem in the md driver used for replication. This problem prevents the release of the nbd device from the mirror resulting in the logging of multiple “Failed to remove device” messages and the aborting of the mirror resync. A system reboot may be required to clear the condition. This problem has been observed during initial resyncs after mirror creation and when the mirror is under stress.

Solution: Kernel 2.6.32-131.17.1 has been verified to contain the fix for this problem. If you are using DataKeeper with Red Hat or CentOS 6 kernels before the 2.6.32-131.17.1 version, we recommend updating to this or the latest available version.

DataKeeper does not support using Network Compression on SLES11 SP4 and SLES12 SP1 or later

DataKeeper does not support using Network Compression on SLES11 SP4 and SLES12 SP1 or later due to disk I/O performance problem.

Certain kernel versions do not support DataKeeper asynchronous mode.

It has been observed that kernel panic will occur with certain kernel versions when using DataKeeper resource asynchronous mode with LifeKeeper for Linux. Since this is a kernel dependent problem, there is no fundamental solution with LifeKeeper. In order to use DataKeeper asynchronous mode configuration, it is necessary to update or downgrade the kernel.

The kernel versions that do not support the DataKeeper asynchronous mode are as follows.
3.10.0-693. series for 3.10.0-693.24.1.el7.x86_64 or later
3.10.0-862.el7.x86_64 ~ 3.10.0-862.26.x.el7.x86_64
3.10.0-957.el7.x86_64 ~ 3.10.0-957.3.x.el7.x86_64

If you use the kernel version listed above and use DataKeeper resources in asynchronous mode, please update (or downgrade) to the following kernel version.
3.10.0-693. series kernel for before 3.10.0-693.24.1.el7.x86_64
3.10.0-862.29.1.el7.x86_64 or later
3.10.0-957.4.1.el7.x86_64 or later

If you cannot update (or downgrade) the kernel, do not use DataKeeper asynchronous mode.

Secure Boot is not supported on RHEL/CentOS/Oracle Linux

If Secure Boot is enabled on RHEL7 or later, CentOS7 or later, or Oracle Linux 7 or later, the nbd module fails to load. For this reason, Secure Boot cannot be enabled in a DataKeeper environment. When using UEK with Oracle Linux, Secure Boot can be enabled.

Solution: Take one of the following actions:

1. Disable Secure Boot – Disable Secure Boot in the UEFI configuration.
2. Disable signature verification – Disable signature verification with the “mokutil `--disable-validation`“ command. See mokutil documentations for details.

   Solution 1 is recommended. Both require a system reboot.

5.3.5.2.7. IPv6 – Known Issues / Restrictions

Description
SIOS has migrated to the use of the `ip` command and away from the `ifconfig` command. Because of this change, customers with external scripts are advised to make a similar change. Instead of issuing the `ifconfig` command and parsing the output looking for a specific interface, scripts should instead use "`ip -o addr show`" and parse the output looking for lines that contain the words "inet" and "secondary". `# ip -o addr show` `1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN` `\ link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00` `1: lo inet 127.0.0.1/8 scope host lo` `1: lo inet6 ::1/128 scope host` `\ valid_lft forever preferred_lft forever` `2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state U P qlen 1000` `\ link/ether d2:05:de:4f:a2:e6 brd ff:ff:ff:ff:ff:ff` `2: eth0 inet 172.17.100.77/22 brd 172.17.103.255 scope global eth0` `2: eth0 inet 172.17.100.79/22 scope global secondary eth0` `2: eth0 inet 172.17.100.80/22 scope global secondary eth0` `2: eth0 inet6 2001:5c0:110e:3364::1:2/64 scope global` `\ valid_lft forever preferred_lft forever` `2: eth0 inet6 2001:5c0:110e:3300:d005:deff:fe4f:a2e6/64 scope global dynam ic` `\ valid_lft 86393sec preferred_lft 14393sec` `2: eth0 inet6 fe80::d005:deff:fe4f:a2e6/64 scope link` `\ valid_lft forever preferred_lft forever` So for the above output from the ip command, the following lines contain virtual IP addresses for the eth0 interface: `2: eth0 inet 172.17.100.79/22 scope global secondary eth0` `2: eth0 inet 172.17.100.80/22 scope global secondary eth0`
'IPV6_AUTOCONF = No' for *<u>/etc/sysconfig/network-scripts/ifcfg-<nicName></u>* is not being honored on reboot or boot On boot, a stateless, auto-configured IPv6 address is assigned to the network interface. If a comm path is created with a stateless IPv6 address of an interface that has IPV6_AUTOCONF=No set, the address will be removed if any system resources manage the interface, e.g. ifdown <nicName>;ifup <nicName>. Comm path using auto-configured IPv6 addresses did not recover and remained dead after rebooting primary server because IPV6_AUTOCONF was set to No.

Solution: Use Static IPv6 addresses only. The use of auto-configured IPv6 addresses could cause a comm loss after a reboot, a NIC change, etc.

While IPv6 auto-configured addresses may be used for comm path creation, the system administrator should be aware of the following conditions:

- IPv6 auto-configured/stateless addresses are dependent on the network interface (NIC) MAC address. If a comm path was created and the associated NIC is later replaced, the auto-configured IPv6 address will be different and LifeKeeper will show the comm path is dead. The comm path will need to be recreated.

- With Red Hat Enterprise Linux, implementing the intended behavior for assuring consistent IPv6 auto-configuration during all phases of host operation requires specific domain knowledge for accurately and precisely setting the individual interface config files AS WELL AS the sysctl.conf, net.ipv6.* directives (i.e. explicitly setting IPV6_AUTOCONF in the ifcfg-<nic> which is referenced by the 'if/ip' utilities AND setting directives in /etc/sysctl.conf which impact NIC control when the system is booting and switching init levels).

IP: Modify Source Address Setting for IPv6 doesn't set source address

When attempting to set the source address for an IPv6 IP resource, it will report success when nothing was changed.

Workaround: Currently no workaround is available. This will be addressed in a future release.

IP: Invalid IPv6 addressing allowed in IP resource creation

Entering IPv6 addresses of the format 2001:5c0:110e:3368:000000:000000001:61:14 is accepted when the octets contain more than four characters.

Workaround: Enter correctly formatted IPv6 addresses.

IPv6 resource reported as ISP when address assigned to bonded NIC but in 'tentative' state

IPv6 protected resources in LifeKeeper will incorrectly be identified as 'In Service Protected' (ISP) on SLES systems where the IPv6 resource is on a bonded interface, a mode other than 'active-backup' (1) and Linux kernel 2.6.21 or lower. The IPv6 bonded link will remain in the 'tentative' state with the address unresolvable.

Workaround: Set the bonded interface mode to 'active-backup' (1) or operate with an updated kernel which will set the link state from 'tentative' to 'valid' for modes other than 'active-backup' (1).

5.3.5.2.8. Apache – Known Issues / Restrictions

Description
Apache Kit does not support IPv6; doesn't indentify IPv6 in *httpd.conf* Any IPv6 addresses assigned to the 'Listen' directive entry in the *httpd.conf* file will cause problems. **Solution:** Until there is support for IPv6 in the Apache Recovery Kit, there can be no IPv6 address in the *httpd.conf* file after the resource has been created.

5.3.5.2.9. Oracle – Known Issues / Restrictions

Description
The Oracle Recovery Kit does not include support for Connection Manager and Oracle Names features The LifeKeeper Oracle Recovery Kit does not include support for the following Oracle Net features of Oracle: Oracle Connection Manager, a routing process that manages a large number of connections that need to access the same service; and Oracle Names, the Oracle-specific name service that maintains a central store of service addresses. The LifeKeeper Oracle Recovery Kit does protect the Oracle Net Listener process that listens for incoming client connection requests and manages traffic to the server. Refer to the LifeKeeper for Linux Oracle Recovery Kit Administration Guide for LifeKeeper configuration specific information regarding the Oracle Listener.
The Oracle Recovery Kit does not support the ASM or grid component features The Oracle Automatic Storage Manager (ASM) feature provided in Oracle is not currently supported with LifeKeeper. In addition, the grid components are not protected by the LifeKeeper Oracle Recovery Kit. Support for raw devices, file systems, and logical volumes are included in the current LifeKeeper for Linux Oracle Recovery Kit. The support for the grid components can be added to LifeKeeper protection using the gen/app recovery kit.
The Oracle Recovery Kit does not support NFS Version 4 The Oracle Recovery Kit supports NFS Version 3 for shared database storage. NFS Version 4 is not supported at this time due to NFSv4 file locking mechanisms.
Oracle listener stays in service on primary server after failover Network failures may result in the listener process remaining active on the primary server after an application failover to the backup server. Though connections to the correct database are unaffected, you may still want to kill that listener process.

5.3.5.2.10. MySQL – Known Issues / Restrictions

Description
The “include” directive is not supported The “include” directive is not supported. All the setup configuration information must be described in a single my.cnf file.
Crash Recovery Restarting MySQL after an abnormal termination initiates a MySQL crash recovery. While in this recovery state MySQL client connections are denied. This will prevent LifeKeeper from checking the state of MySQL causing a possible failover to the standby node.

5.3.5.2.11. NFS Server – Known Issues / Restrictions

Description
NFS v4 and bind mounts cannot be used together on systems where /etc/mtab is a symlink to /proc/self/mounts Bind mounts cannot be used with NFS v4 shares on these systems (RHEL 7.0 and later, CentOS 7.0 and later, OEL 7.0 and later, and SLES 12 SP1 and later) because the bind information that was formerly found in /etc/mtab is not found in /proc/self/mounts.
Top level NFS resource hierarchy uses the switchback type of the hanfs resource The switchback type, which dictates whether the NFS resource hierarchy will automatically switch back to the primary server when it comes back into service after a failure, is defined by the hanfs resource.
Some clients are unable to reacquire nfs file locks When acting as NFS clients, some Linux kernels do not respond correctly to notifications from an NFS server that an NFS lock has been dropped and needs to be reacquired. As a result, when these systems are the clients of an NFS file share that is protected by LifeKeeper, the NFS locks held by these clients are lost during a failover or switchover. When using storage applications with locking and following recommendations for the NFS mount options, SPS requires the additional nolock option be set, e.g. `rw,nolock,bg,hard,nointr,tcp,nfsvers=3,timeo=600,rsize=32768,wsize=32768,actimeo=0.`
NFS v4 changes not compatible with SLES 11 nfs subsystem operation The mounting of a non-NFS v4 remote export on SLES 11 starts rpc.statd. The start up of rpc.statd on the out of service node in a cluster protecting an NFS v4 root export will fail. **Solution:** Do not mix NFS v2/v3 with a cluster protecting an NFS v4 root export.
NFS v4 cannot be configured with IPv6 IPv6 virtual IP gets rolled up into the NFSv4 heirarchy.

Solution: Do not use an IPv6 virtual IP resource when creating an NFSv4 resource.

NFS v4: Unable to re-extend hierarchy after unextend

Extend fails because export point is already exported on the target server. A re-extend to server A of a NFS v4 hierarchy will fail if a hierarchy is created on server A and extended to server B, brought in service on server B and then unextended from server A.

Solution: On server A run the command "`exportfs -ra`" to clean up the extra export information left behind.

File Lock switchover with NFSv3 fails on some operating systems

Failover file locks with NFSv3 during resources switchover/failover does not work on the operating systems listed below. Lock failover with NFSv3 is currently not supported on these OS versions.

- RHEL7, CentOS7, OL7
- RHEL6, CentOS6, OL6
- SLES12
- SLES11

Solution: Use the lock failover features available with NFSv4.

The Oracle Recovery Kit does not support NFSv4

The Oracle Recovery Kit supports NFSv3 for shared database storage. NFSv4 is not supported at this time due to NFSv4 file locking mechanisms.

Stopping and starting NFS subsystem adversely impacts SIOS Protection Suite protected NFS exports.

If the NFS subsystem (`/etc/init.d/nfs` on Red Hat or `/etc/init.d/nfsserver` on SuSE) is stopped while SIOS Protection Suite for Linux is protecting NFS exports, then all SIOS Protection Suite for Linux protected exported directories will be impacted as the NFS stop action performs an unexport of all the directories. The SIOS Protection Suite for Linux NFS quickCheck script will detect the stopped NFS processes and the unexported directories and run a local recovery to restart the processes and re-export the directories. However, it will take a quickCheck run for each protected export for the SIOS Protection Suite NFS ARK to recover everything. For example, if five exports are protected by the SIOS Protection Suite for Linux NFS ARK, it will take five quickCheck runs to recover all the exported directories the kit protects. Based on the default

quickCheck time of two minutes, it could take up to ten minutes to recover all the exported directories.

Workaround: Do not stop the NFS subsystem while the SIOS Protection Suite NFS ARK is actively protecting exported directories on the system. If the NFS subsystem must be stopped, all NFS resources should be switched to the standby node before stopping the NFS subsystem. Use of the `exportfs` command should also be considered. This command line utility provides the ability to export and unexport a single directory thus bypassing the need to stop the entire NFS subsystem.

5.3.5.2.12. SAP Recovery Kit – Known Issues / Restrictions

Description
SAP resources fail to come in-service due to csh bug in RHEL 8 Due to a bug in the tcsh package available for RHEL 8, the SAP administrative user's .cshrc and .login files are not sourced correctly in certain situations. Due to this, important environment variables that the SAP Recovery Kit depends on are not properly exported, which may cause SAP resources to fail to come in-service on RHEL 8. See RedHat Bug 1714267 for more details and a workaround.
SAP Dual Stack Environment Restriction The redesigned ERS resource type introduced in SPS-L 9.4.0 (which operates in a hierarchy separate from the corresponding central services instance) does not support an SAP dual stack (ABAP+Java) environment where there are two pairs of central services and enqueue replication server instances (e.g., ASCS00/ERS10 and SCS01/ERS11) installed under the same SID. Customers with an SAP dual stack (ABAP+Java) environment installed under the same SID should continue to use the pre-9.4.0 ERS resource design (which is located at the top of the SAP hierarchy with a dependency on the corresponding ASCS/SCS resource).
Failed delete or unextend of a SAP hierarchy Deleting or unextending a SAP hierarchy that contains the same IP resource in multiple locations within the hierarchy can sometimes cause a core dump that results in resources not being deleted. To correct the problem, after the failed unextend or delete operation, manually remove any remaining resources using the LifeKeeper GUI. You may also want to remove the core file from the server.
Handle Warnings gives a syntax error at -e line 1 When changing the default behavior of **No** in **Handle Warnings** to **Yes**, an error is received. **Solution:** Leave this option at the default setting of **No**. **Note:** It is highly recommended that this setting be left on the default selection of **No** as Yellow is a transient state that most often does not indicate a failure.
Choosing same setting causes missing button on Update Wizard

If user attempts to update the **Handle Warning** without changing the current setting, the next screen, which indicates that they must go back, is missing the **Done** button.

When changes are made to res_state, monitoring is disabled

If **Protection Level** is set to **BASIC** and SAP is taken down manually (i.e. for maintenance), it will be marked as FAILED and monitoring will stop.

Solution: In order for monitoring to resume, LifeKeeper will need to start up the resource instead of starting it up manually.

ERS in-service fails on remote host if ERS is not parent of Core/CI

Note: This only applies to the pre-9.4.0 ERS resource design (which is located at the top of the SAP hierarchy with a dependency on the corresponding ASCS/SCS resource). For more details see ERS Resource Types in LifeKeeper.

Creating an ERS resource without any additional SAP resource dependents will cause initial in-service to fail on switchover.

Solution: Create ERS as parent of CI/Core instance (SCS or ASCS), then retry in-service.

SAP instance processes in an inconsistent state

Issuing concurrent administrative commands while a migration of an SAP resource is in-progress may leave the SAP instance processes in an inconsistent state, which may require manual intervention to resolve.

5.3.5.2.13. SAP HANA – Known Issues / Restrictions

Description
SAP HANA resource does not function properly when the tag name contains forward slashes If forward slashes (“/”) are used in the tag name for a HANA resource, the resource will not behave as expected. In particular, the health of the resource will not be monitored properly. The resource must be recreated and given a tag name that does not contain forward slashes.

5.3.5.2.14. LVM – Known Issues / Restrictions

Description
Important reminder about DataKeeper for Linux asynchronous mode in an LVM over DataKeepr configuration Kernel panics may occur in configurations were LVM resources sit above multiple asynchronous mirrors. In these configurations data consistency may be an issue if a panic occurs. Therefore the required configurations are a single DataKeeper mirror or multiple synchronous DataKeeper mirrors.
Use of lkID incompatible with LVM overwritten on entire disk When lkID is used to generate unique disk IDs on disks that are configured as LVM physical volumes, there is a conflict in the locations in which the lkID and LVM information is stored on the disk. This causes either the lkID or LVM information to be overwritten depending on the order in which lkID and pvcreate are used. **Workaround:** When it is necessary to use lkID in conjunction with LVM, partition the disk and use the disk partition(s) as the LVM physical volume(s) rather than the entire disk.
LVM actions slow on RHEL 6 When running certain LVM commands on RHEL 6, performance is sometimes slower than in previous releases. This can be seen in slightly longer restore and remove times for hierarchies with LVM resources.
The configuration of Raw and LVM Recovery Kits together is not supported in RHEL 6 environment When creating a Raw resource, the Raw Recovery Kit is looking for a device file based on major # and minor # of Raw device. As the result, `/dev/dm-*` will be the device; however, this type of `/dev/dm-*` cannot be handled by the LVM Recovery Kit and a “`raw device not found`“ error will occur in the GUI.

5.3.5.2.15. Multipath Recovery Kits (DMMP / HDLM / PPATH /NECSPS) Known Issues / Restrictions

Description
Multipath Recovery Kits (DMMP / HDLM / PPATH / NECSPS): Registration conflict error occurs on lkstop when resource OSF The multipath recovery kits (DMMP, HDLM, PPATH, NECSPS) can have a system halt occur on the active (ISP) node when LifeKeeper is stopped on the standby (OSU) node if the Multipath resource state is OSF. **Workarounds:** a) Switch the hierarchy to the standby node before LifeKeeper is stopped **OR** b) Run `ins_setstate` on the standby node and set the Multipath resource state to OSU before LifeKeeper is stopped

5.3.5.2.16. DMMP – Known Issues / Restrictions

Description
DMMP: Write issued on standby server can hang If a write is issued to a DMMP device that is reserved on another server, then the IO can hang indefinitely (or until the device is no longer reserved on the other server). If/when the device is released on the other server and the write is issued, this can cause data corruption. The problem is due to the way the path checking is done along with the IO retries in DMMP. When “no_path_retry” is set to 0 (fail), this hang will not occur. When the path_checker for a device fails when the path is reserved by another server (MSA1000), then this also will not occur. **Workaround:** Set “no_path_retry” to 0 (fail). However, this can cause IO failures due to transient path failures.
DMMP: Multiple initiators are not registered properly for SAS arrays that support ATP_C LifeKeeper does not natively support configurations where there are multiple SAS initiators connected to a SAS array. In these configurations, LifeKeeper will not register each initiator correctly, so only one initiator will be able to issue IOs. Errors will occur if the multipath driver (DMMP for example) tries to issue IOs to an unregistered initiator. **Solution:** Set the following tunable in */etc/default/LifeKeeper* to allow path IDs to be set based on SAS storage information: • `MULTIPATH_SAS=TRUE`
LifeKeeper on RHEL 6.0 cannot support reservations connected to an EMC Clariion
Two or more different storage can not be used concurrently in case of the parameter configuration of DMMP recovery kit is required for some storage model.
DMMP RK doesn’t function correctly if the disk name ends with “p<number>”.

The DMMP RK doesn't function correctly if the disk name ends with "p<number>".

Workaround: Do not create disk names ending in "p<number>".

5.3.5.2.17. DB2 – Known Issues / Restrictions

Description
DB2 Recovery Kit reports unnecessary error If DB2 is installed on a shared disk, the following message may be seen when extending a DB2 resource. • `LifeKeeper was unable to add instance “%s” and/or its variables to the DB2 registry.` This message will not adversely affect the behavior of the DB2 resource extend.

5.3.5.2.18. MD Recovery Kit – Known Issues / Restrictions

Description
MD Kit does not support mirrors created with “homehost” The LifeKeeper MD Recovery Kit will not work properly with a mirror created with the “homehost” feature. Where “homehost” is configured, LifeKeeper will use a unique ID that is improperly formatted such that in-service operations will fail. On SLES 11 systems, the “homehost” will be set by default when a mirror is created. The version of mdadm that supports “homehost” is expected to be available on other distributions and versions as well. When creating a mirror, specify —homehost=”“ on the command line to disable this feature. If a mirror already exists that has been created with the “homehost” setting, the mirror must be recreated to disable the setting. If a LifeKeeper hierarchy has already been built for a mirror created with “homehost”, the hierarchy must be deleted and recreated after the mirror has been built with the “homehost” disabled.
MD Kit does not support MD devices created on LVM devices The LifeKeeper MD Recovery Kit will not work properly with an MD device created on an LVM device. When the MD device is created, it is given a name that LifeKeeper does not recognize.
MD Kit configuration file entries in /etc/mdadm.conf not commented out The LifeKeeper configuration file entries in /etc/mdadm.conf should be commented out after a reboot. These file entries are not commented out.
Local recovery not performed in large configurations In some cases with large configurations (6 or more hierarchies), if a local recovery is triggered (sendevent), not all of the hierarchies are checked resulting in local recovery attempt failures.
Mirrors automatically started during boot On some systems (for example those running RHEL 6), there is an AUTO entry in the configuration file (/etc/mdadm.conf) that will automatically start mirrors during boot (example: AUTO +imsm +1.x –all). **Solution:** Since LifeKeeper requires that mirrors not be automatically started, this entry will need to be edited to make sure that LifeKeeper mirrors will not be automatically started during boot. The previous

example (AUTO +imsm +1.x –all) is telling the system to automatically start mirrors created using imsm metadata and 1.x metadata minus all others. This entry should be changed to “AUTO -all”, telling the system to automatically start everything “minus” all; therefore, nothing will be automatically started.

Important: If system critical resources (such as root) are using MD, make sure that those mirrors are started by other means while the LifeKeeper protected mirrors are not.

5.3.5.2.19. Sybase ASE – Known Issues / Restrictions

Description
User Name/Password Issues: • **If the default user name is password-protected, the create UI does not detect this until after all validation is complete** When creating the Sybase resource, you are prompted to enter the user name. The help to front displays a message that if no user is specified, the default of 'sa' will be used. However, no password validation is done for the default account at this time. When SIOS Protection Suite attempts to create the Sybase resource, the resource creation fails because the password has not been validated or entered. The password validation occurs on the user/password dialog, but only when a valid user is actually entered on the user prompt. Even if using the default user name, it must be specified during the **create** action. • **Password prompt skipped if no user name specified** User/password dialog skips the password prompt if you do not enter a user name. When updating the user/password via the UI option, if you do not enter the Sybase user name, the default of 'sa' will be used and no password validation is done for the account. This causes the monitoring of the database to fail with invalid credential errors. Even if using the default user name, it must be specified during the update action. To fix this failure, perform the following steps: 1. Verify that the required Sybase data files are currently accessible from the intended server. In most instances, this will be the backup server due to the monitoring and local recovery failure on the primary. 2. Start the Sybase database instance from the command line on this server (see the Sybase product documentation for information on starting the database manually). 3. From the command line, change directory (cd) to the LKROOT/bin directory (/opt/LifeKeeper/bin on most installations). 4. Once in the bin directory, execute the following: `./ins_setstate -t <SYBASE_TAG> -S ISP` where `<SYBASE_TAG>` is the tag name of the Sybase resource 5. When the command completes, immediately execute the ***Update User/Password Wizard*** from

the UI and enter a valid user name, even if planning to use the Sybase default of 'sa'. **Note:** The ***Update User/Password Wizard*** can be accessed by right-clicking on the Sybase resource instance and selecting **Change Username/Password**.

6. When the hierarchy has been updated on the local server, verify that the resource can be brought in service on all nodes.

7. **Protecting backup server fails when Sybase local user name >= eight characters**

   The Sybase user name must consist of less than eight characters. If the Sybase local user name is greater than eight characters, the process and user identification checks used for resource creation and monitoring will fail. This will also prevent the protection of a valid Sybase Backup Server instance from being selected for protection. This problem is caused by the operating system translation of user names that are >= eight characters from the name to the UID in various commands (for example, ps). You must use a user name that is less than eight characters long.

Resource Create Issue:

- Default Sybase install prompt is based on ASE 16.0 SP02 (/opt/sybase). During the SIOS Protection Suite resource creation, the default prompt for the location of the Sybase installation shows up relative to Sybase Version 16.0 SP02 (/opt/sybase). You must manually enter or browse to the correct Sybase install location during the resource create prompt.

Extend Issues:

- **The Sybase tag prompt on extend is editable but should not be changed.** The Sybase tag can be changed during extend, but this is not recommended. Using different tags on each server can lead to issues with remote administration via the command line.

Properties Page Issues:

- **Image appears missing for the Properties pane update user/password.** Instead of the proper image, a small square appears on the toolbar. Selecting this square will launch the ***User/ Password Update Wizard***.

Sybase Monitor server is not supported in 15.7 or later with SIOS Protection Suite. If the Sybase Monitor server process is configured in Sybase 15.7 or later, you must use a Generic Application (gen/app) resource to protect this server process.

Remove command not recognized by Sybase on SLES 11

If bringing Sybase in service on the backup server, it must first be taken out of service on the primary server. In order for Sybase to recognize this command, you must add a line in "`locales/locales.dat`" for the SIOS Protection Suite remotely executed Sybase command using "`POSIX`" as "`vendor_locale`".

Example:

```
locale = POSIX, us_english, utf8
```

Cannot create a resource using the Sybase RK OR experiencing issues on a LifeKeeper upgrade when using SLES 15

Symptoms:
When setting up SLES 15 (this includes SLES 15 SP1 or SP2), the Sybase RK may not work on an upgrade or on a fresh install of LifeKeeper.

During a fresh install of LifeKeeper you won't be able to create the Sybase resource, or during a LifeKeeper upgrade there could be issues in controlling the resource from LifeKeeper. The underlying error is "An error occurred when attempting to allocate localization-related structures".

Resolution:
There are 2 possible solutions:

1. Add a line in the [Linux] section of the file */sybase/<SID>/locales/locales.dat*
   `locale = POSIX, us_english, utf8`

   OR

2. Add this line to the Sybase profile for the *syb<SID>* user profile.
   This profile is typically located in */sybase//SYBASE.sh*
   `export LANG=en_US.UTF-8`

   <SID> is the 3 letter system ID.

   These changes force the correct language to be used and is appropriate for US English locale. If another locale is used for Japan or China, please consult the local country and the system locale.

   You can also su to the syb<SID> user and run the command `echo $LANG`. The cshell parameter will point to the correct language for the appropriate locale.

   Which change is appropriate to make? It depends on the comfort level of the Sybase (SAP ASE) database administrator. One is changing the shell script, the other will add the POSIX support to the locales.

Unable to detect that Sybase ARK is running

Symptom: Unable to detect that Sybase ARK is running.

Cause: The Sybase ARK uses the default sql interface tool (isql). On some 64 bit systems, the isql tool is installed as isql64 and not isql.

Solution: The isql64 tool can be copied, in the same path, to isql. Or a link can be created between the isql64 executable and isql.

5.3.5.2.20. WebSphere MQ – Known Issues / Restrictions

Description
Error when lksupport command is executed: The following error can be output when lksupport command is executed in the case MQ queue manager protected by MQ RK is set on the disk shared by NFS. • `cat: <PATH>/mqm/qmgrs/tkqmgr/qm.ini: Operation not permitted` This happens because the root access to NFS area is prohibited. This error output doesn't cause any problem.
Quickcheck fails if queue has long messages if a message is in the test queue of size > 101 characters the put/get fails and the queue fills up
Install fails if the only installed MQ is a relocated install (non-standard and not likely)
Package install fails if the MQ package does not have the default name
Compile samples fails if the software is not installed under /opt/mqm
If two listeners are defined for a single instance and one is set to manual and the other is automatic failures can occur in create and quickCheck

5.3.5.3. GUI Troubleshooting

If you are having problems configuring the LifeKeeper GUI from a remote system, see one of the following topics:

Java Plug-In Troubleshooting

Applet Troubleshooting

Network-Related Troubleshooting

5.3.5.3.1. Network Related Troubleshooting (GUI)

LifeKeeper uses Java RMI (Remote Method Invocation) for communications between GUI clients and servers. Some potential problems may be related to RMI, and others are general network configuration problems.

Long Connection Delays on Windows Platforms

From Sun FAQ:

Most likely, your host's networking setup is incorrect. RMI uses the Java API networking classes, in particular *java.net.InetAddress*, which will cause TCP/IP host name lookups for both host to address mapping and address to hostname. On Windows, the lookup functions are performed by the native Windows socket library, so the delays are not happening in RMI but in the Windows libraries. If your host is set up to use DNS, then this could be a problem with the DNS server not knowing about the hosts involved in communication, and what you are experiencing are DNS lookup timeouts. If this is the case, try specifying all the hostnames/addresses involved in the file *\windows\system32\drivers\etc\hosts*. The format of a typical host file is:

IPAddress Server Name

e.g.:

192.168.100.1 homer.example.com homer

This should reduce the time it takes to make the first lookup.

In addition, incorrect settings of the Subnet Mask and Gateway address may result in connection delays and failures. Verify with your Network Administrator that these settings are correct.

Running from a Modem:

When you connect to a network in which the servers reside via modem (using PPP or SLIP), your computer acquires a temporary IP number for its operation. This temporary number may not be the one your hostname maps to (if it maps to anything at all), so in this case, you must tell the servers to communicate with you by IP alone. To do this, obtain your temporary IP number by opening your modem connection window. This number will be used to set the hostname property for the GUI client.

To set the hostname for a browser with the Plugin, open the **Java Plug-In Control Panel**, and set the hostname for the client by adding the following to "**Java Run Time Parameters**".

```
-Djava.rmi.server.hostname=<MY_HOST>
```

To set the hostname for the HotJava browser, append the following to the hotjava command line:

```
-Djava.rmi.server.hostname=<MY_HOST>
```

For example:

```
-Djava.rmi.server.hostname=192.168.100.2
```

Primary Network Interface Down:

The LifeKeeper GUI uses Remote Method Invocation (RMI) to maintain contact between the GUI client and the GUI server. In nearly every case, contact is established over the primary network interface to the server. This means that if the server's primary Ethernet interface goes down, contact is lost and the GUI client shows that server state as Unknown.

The only solution to this problem is to bring the server's primary Ethernet interface up again. Additionally, due to limitations in RMI, this problem cannot be overcome by using a multi-homed server (server with multiple network interfaces).

No Route To Host Exception:

A socket could not be connected to a remote host because the host could not be contacted. Typically, this means that some link in the network between the local server and the remote host is down, or that the host is behind a firewall.

Unknown Host Exception:

The LifeKeeper GUI Client and Server use Java RMI (Remote Method Invocation) technology to communicate. For RMI to work correctly, the client and server must use resolvable hostname or IP addresses. When unresolvable names, WINS names or unqualified DHCP names are used, this causes Java to throw an UnknownHostException.

This error message may also occur under the following conditions:

- Server name does not exist. Check for misspelled server name.
- Misconfigured DHCP servers may set the fully-qualified domain name of RMI servers to be the domain name of the resolver domain instead of the domain in which the RMI server actually resides. In this case, RMI clients outside the server's DHCP domain will be unable to contact the server because of the its incorrect domain name.
- The server is on a network that is configured to use Windows Internet Naming Service (WINS). Hosts that are registered under WINS may not be reachable by hosts that rely solely upon DNS.
- The RMI client and server reside on opposite sides of a firewall. If your RMI client lies outside a

firewall and the server resides inside of it, the client will not be able to make any remote calls to the server.

When using the LifeKeeper GUI, the hostname supplied by the client must be resolvable from the server and the hostname from the server must be resolvable by the client. The LifeKeeper GUI catches this exception and alerts the user. If the client cannot resolve the server hostname, this exception is caught and Message 115 is displayed. If the server cannot resolve the Client hostname, this exception is caught and Message 116 is displayed. Both these messages include the part of the Java exception which specifies the unqualified hostname that was attempted.

Included below are some procedures that may be used to test or verify that hostname resolution is working correctly.

From Windows:

1. Verify communication with the Linux Server

   From a DOS prompt, ping the target using the hostname:

   ```
   ping <TARGET_NAME>
   ```

   For Example:

   ```
   ping homer
   ```

   A reply listing the target's qualified hostname and IP address should be seen.

2. Verify proper configuration

- ◦ Check configuration of DNS or install a DNS server on your network.
- ◦ Check the settings for ControlPanel->Network->Protocols->TCP/IP. Verify with your Network Administrator that these settings are correct.

  Note: The hostname in the DNS tab should match the name used on the local name server. This should also match the hostname specified in the GUI error message.

  Try editing the hosts file to include entries for the local host and the LifeKeeper servers that it will be connected to.
  On Windows 95/98 systems the hosts file is:

  ```
  %windir%\HOSTS (for example, C:\WINDOWS\HOSTS).
  ```

  Note: On Windows 95/98, if the last entry in the hosts file is not concluded with a carriage-return/line-feed then the hosts file will not be read at all.

  On Windows NT systems the hosts file is:

```
%windir%\System32\DRIVERS\ETC\HOSTS
(for example, C:\WINNT\System32\DRIVERS\ETC\HOSTS).
```

For example, if my system is called *HOSTCLIENT.MYDOMAIN.COM* and uses IP address *192.168.200.10*, add the following entry to the hosts file:

```
192.168.200.10 HOSTCLIENT.EXAMPLE.COM HOSTCLIENT
```

3. Try setting the hostname property to be used by the GUI client. To do this from a browser with the Plugin, open the **Java Plug-In Control Panel**, and set the host name for the client by adding the following to "**Java Run Time Parameters**":

```
Djava.rmi.server.hostname=<MY_HOST>
```

4. Check for Microsoft network-related patches at www.microsoft.com.

From Linux:

1. Verify communication with other servers by pinging the target server from Linux using its hostname or IP address:

```
ping <TARGET_NAME>
```

For example:

```
ping homer
```

A reply listing the target's qualified hostname should be seen.

2. Verify that localhost is resolvable by each server in the cluster using **ping** with its hostname or IP address. If DNS is not implemented, edit the */etc/hosts* file and add an entry for the *localhost* name. This entry can list either the IP address for the local server, or it can list the default entry (127.0.0.1).

3. Check that DNS is specified before NIS. DNS should be put before NIS in the hosts line of */etc/nsswitch.conf*, and */etc/resolv.conf* should point to a properly configured DNS server(s).

4. If DNS is not to be implemented or no other method works, edit the */etc/hosts* file, and add an entry for the hostname.

5. Try setting the hostname property to be used by the GUI client. This will need to be changed for each administrator.

To do this from a browser with the Plugin, open the **Java Plug-In Control Panel** and set the hostname for the client by adding the following to "**Java Run Time Parameters**":

```
-Djava.rmi.server.hostname=<MY_HOST>
```

To do this from the HotJava browser, append the following to the hotjava command line:

```
-Djava.rmi.server.hostname=<MY_HOST>
```

For Example:

```
-Djava.rmi.server.hostname=192.168.200.10
```

```
-Djava.rmi.server.hostname= homer.example.com
```

Unable to Connect to X Window Server:

When running the LifeKeeper GUI application from a telnet session, you need to ensure that the GUI client is allowed to access the X Window Server on the LifeKeeper server. The LifeKeeper server must also be able to resolve the hostname or network address of the GUI client.

When you telnet into the LifeKeeper server to run the LifeKeeper GUI application, the *DISPLAY* environment variable should contain the client's host name and display number. For example, if you telnet into a server named *Server1* from a client named *Client1*, the *DISPLAY* environment variable should be set to *Client1:0*. When you run the LifeKeeper GUI application, it will try to send the output to the *DISPLAY* name for *Client1*. If *Client1* is not allowed access to the *X Window Server*, the LifeKeeper GUI application will fail with an exception.

When starting the LifeKeeper GUI as an application, if an error occurs indicating that you cannot connect to the *X Window Server* or that you cannot open the client *DISPLAY* name, try the following:

1. Set the display variable using the host name or IP address. For example:

   ```
   DISPLAY=Client1.somecompany.com:0
   ```

   ```
   DISPLAY=172.17.5.74:0
   ```

2. Use the `xhost` or `xauth` command to verify that the client may connect to the *X Window Server* on the LifeKeeper server.

3. Add a DNS entry for the client or add an entry for the client to the local hosts file on the LifeKeeper server. Verify communication with the client by pinging the client from the LifeKeeper server using its hostname or IP address.

5.3.5.4. Communication Paths Going Up and Down

If you find the communication paths failing then coming back up repeatedly (the LifeKeeper GUI showing them as Alive, then Dead, then Alive), the heartbeat tunables may not be set to the same values on all servers in the cluster.

This situation is also possible if the tunable name is misspelled in the LifeKeeper defaults file **/etc/default/LifeKeeper** on one of the servers.

Suggested Action

1. Shut down LifeKeeper on all servers in the cluster.

2. On each server in the cluster, check the values and spelling of the **LCMHBEATTIME** and **LCMNUMHBEATS** tunables in **/etc/default/LifeKeeper**. Ensure that for each tunable, the values are the same on ALL servers in the cluster.

3. Restart LifeKeeper on all servers.

5.3.5.5. Incomplete Resource Created

If the resource setup process is interrupted leaving instances only partially created, you must perform manual cleanup before attempting to install the hierarchy again. Use the LifeKeeper GUI to delete any partially-created resources. See Deleting a Hierarchy from All Servers for instructions. If the hierarchy list does not contain these resources, you may need to use the `ins_remove` (see LCDI-instances(1M)) and `dep_remove` (LCDI-relationship(1M)) to clean up the partial hierarchies.

5.3.5.6. Incomplete Resource Priority Modification

A hierarchy in LifeKeeper is defined as all resources associated by parent/child relationships. For resources that have multiple parents, it is not always easy to discern from the GUI all of the root resources for a hierarchy. In order to maintain consistency in a hierarchy, LifeKeeper requires that priority changes be made to all resources in a hierarchy for each server. The GUI enforces this requirement by displaying all root resources for the hierarchy selected after the OK or Apply button is pressed. You have the opportunity at this point to accept all of these roots or cancel the operation. If you accept the list of roots, the new priority values will be applied to all resources in the hierarchy.

You should ensure that no other changes are being made to the hierarchy while the Resource Properties dialog for that hierarchy is displayed. Before you have edited a priority in the Resource Properties dialog, any changes being made to LifeKeeper are dynamically updated in the dialog. Once you have begun making changes, however, the values seen in the dialog are frozen even if underlying changes are being made in LifeKeeper. Only after selecting the Apply or OK button will you be informed that changes were made that will prevent the priority change operation from succeeding as requested.

In order to minimize the likelihood of unrecoverable errors during a priority change operation involving multiple priority changes, the program will execute a multiple priority change operation as a series of individual changes on one server at a time. Additionally, it will assign temporary values to priorities if necessary to prevent temporary priority conflicts during the operation. These temporary values are above the allowed maximum value of 999 and may be temporarily displayed in the GUI during the priority change. Once the operation is completed, these temporary priority values will all be replaced with the requested ones. If an error occurs and priority values cannot be rolled back, it is possible that some of these temporary priority values will remain. If this happens, follow the suggested procedure outlined below to repair the hierarchy.

Restoring Your Hierarchy to a Consistent State

If an error occurs during a priority change operation that prevents the operation from completing, the priorities may be left in an inconsistent state. Errors can occur for a variety of reasons, including system and communications path failure. If an error occurs after the operation has begun, and before it finishes, and the program was not able to roll back to the previous priorities, you will see a message displayed that tells you there was an error during the operation and the previous priorities could not be restored. If this should happen, you should take the following actions to attempt to restore your hierarchy to a consistent state:

1. If possible, determine the source of the problem. Check for system or communications path failure. Verify that other simultaneous operations were not occurring during the same time that the priority administration program was executing.

2. If possible, correct the source of the problem before proceeding. For example, a failed system or communications path must be restored before the hierarchy can be repaired.

3. Re-try the operation from the Resource Properties dialog.

4. If making the change is not possible from the Resource Properties dialog, it may be easier to attempt to repair the hierarchy using the command line `hry_setpri`. This script allows priorities to be changed on one server at a time and does not work through the GUI.

5. After attempting the repair, verify that the LifeKeeper databases are consistent on all servers by executing the `eqv_list` command for all servers where the hierarchy exists and observing the priority values returned for all resources in the hierarchy.

6. As a last resort, if the hierarchy cannot be repaired, you may have to delete and re-create the hierarchy.

5.3.5.7. No Shared Storage Found When Configuring a Hierarchy

When you are configuring resource hierarchies there are a number of situations that might cause LifeKeeper to report a “No shared storage” message:

Possible Cause: Communications paths are not defined between the servers with the shared storage. When a hierarchy is configured on the shared storage device, LifeKeeper verifies that at least one other server in the cluster can also access the storage.
Suggested Action: Use the LifeKeeper GUI or `lcdstatus (1M)` to verify that communication paths are configured and that they are active.

Possible Cause: Communication paths are not operational between the servers with the shared storage.
Suggested Action: Use the LifeKeeper GUI or `lcdstatus (1M)` to verify that communication paths are configured and that they are active.

Possible Cause: Linux is not able to access the shared storage. This could be due to a driver not being loaded, the storage not being powered up when the driver was loaded, or the storage device is not configured properly.
Suggested Action: Verify that the device is properly defined in `/proc/scsi/scsi`

Possible Cause: The storage was not configured in Linux before LifeKeeper started. During the startup of LifeKeeper, all SCSI devices are scanned to determine the mappings for devices. If a device is configured (powered on, connected or driver loaded) after LifeKeeper is started, then LifeKeeper must be stopped and started again to be able to configure and use the device.
Suggested Action: Verify that the device is listed in `$LKROOT/subsys/scsi/resources/hostadp/device_info` where `$LKROOT` is by default `/opt/LifeKeeper`. If the device is not listed in this file, LifeKeeper will not try to use the device.

Possible Cause: The storage is not supported. The Supported Storage List shows specific SCSI devices that are supported and have been tested with LifeKeeper. However, note that this list includes known devices; there may be other devices that SIOS Technology Corp. has not tested which meet LifeKeeper requirements.
Suggested Action: Verify that the device is listed in `$LKROOT/subsys/scsi/resources/hostadp/device_info` where `$LKROOT` is by default `/opt/LifeKeeper`. If the device is listed in this file but the ID following the device name begins with “`NU-`” then LifeKeeper was unable to get a unique ID from the device. Without a unique ID LifeKeeper cannot determine if the device is shared.

Possible Cause: The storage may require a specific LifeKeeper software to be installed before the device can be used by LifeKeeper. Examples are the **steeleye-lkRAW** kit to enable Raw I/O support and the **steeleye-lkDR** software to enable data replication.
Suggested Action: Verify that the necessary LifeKeeper packages are installed on each server. See the SPS for Linux Release Notes for software requirements.

Additional Tip:
The test_`lk(1M)` tool can be used to help debug storage and communication problems.

5.3.5.8. Recovering from a LifeKeeper Server Failure

If any server in your LifeKeeper cluster experiences a failure that causes re-installation of the operating system (and thus LifeKeeper), you will have to re-extend the resource hierarchies from each server in the cluster. If any server in the cluster has a shared equivalency relationship with the re-installed server, however, LifeKeeper will not allow you to extend the existing resource hierarchy to the re-installed server. LifeKeeper will also not allow you to unextend the hierarchy from the re-installed server because the hierarchy does not really exist on the server that was re-installed.

Suggested Action:

1. On each server where the resource hierarchies are configured, use the `eqv_list` command to obtain a list of all the shared equivalencies (see `LCDI-relationship` for details).

   The example below shows the command and resulting output for the IP resource iptag on server1 and server2 where server2 is the server that was re-installed and server1 has the hierarchy configured:

   ```
   eqv_list -f:

   server1:iptag:server2:iptag:SHARED:1:10
   ```

2. On each server where the resource hierarchies are configured, use eqv_remove to manually remove the equivalency relationship for each resource in the hierarchy (see `LCDI-relationship` for details).

   For example, execute the following command on server1 using the example from step 1 above:

   ```
   eqv_remove -t iptag -S server2 -e SHARED
   ```

3. In clusters with more than two servers, steps 1-2 should be repeated on each server in the cluster where equivalency relationships for these resource hierarchies are defined.

4. Finally, extend each resource hierarchy from the server where the resource hierarchy is in-service to the re-installed server using the GUI.

5.3.5.9. Recovering from a Non-Killable Process

If a process is not killable, LifeKeeper may not be able to unmount a shared disk partition. Therefore, the resource cannot be brought into service on the other system. The only way to recover from a non-killable process is to reboot the system.

5.3.5.10. Recovering from a Panic during a Manual Recovery

A PANIC during manual switchover may cause incomplete recovery. If a PANIC or other major system failure occurs during a manual switchover, complete automatic recovery to the back-up system cannot be assured. Check the backup system to make sure all resources required to be in-service are in-service. If they are not in-service, use the LifeKeeper GUI to manually bring the missing resources into service. See Bringing a Resource In-Service for instructions.

5.3.5.11. Recovering Out-of-Service Hierarchies

As a part of the recovery following the failure of a LifeKeeper server, resource hierarchies that are configured on the failed server, but are not in-service anywhere at the time of the server failure, are recovered on the highest priority alive server at the time of the failure. This is the case no matter where the out-of-service hierarchy was last in-service, including the failed server, the recovering server, or some other server in the hierarchy.

5.3.5.12. Resource Tag Name Restrictions

Tag Name Length

All tags within LifeKeeper may not exceed the 256 character limit.

Valid “Special” Characters

- _ . /

The first character in a tag should not contain “.” or “/”.

Invalid Characters

+ ; : ! @ # $ * = “space”

5.3.5.13. Serial (TTY) Console WARNING

If any part of the serial console data path is unreliable or goes out of service, users who have a serial (RS-232 TTY) console can experience severe problems with LifeKeeper service. During operation, LifeKeeper generates console messages. If your configuration has a serial console (instead of the standard VGA console), the entire data path from LifeKeeper to the end-user terminal must be operational in order to ensure the delivery of these console messages.

If there is any break in the data path—such as terminal powered off, modem disconnected, or cable loose—the Linux STREAMS facility queues the console message. If the STREAMS queue becomes full, the Unix kernel suspends LifeKeeper until the STREAMS buffer queue again has room for more messages. This scenario could cause LifeKeeper to HANG.

Note: The use of serial consoles in a LifeKeeper environment is strongly discouraged. SIOS recommends using the VGA console. If you must use a serial console, be sure that your serial console is turned on, the cables and optional modems are connected properly, and that messages are being displayed.

5.3.5.14. Taking the System to init state S WARNING

When LifeKeeper is operational, the system must not be taken directly to init state S. Due to the operation of the Linux init system, such a transition causes all the LifeKeeper processes to be killed immediately and may precipitate a fastfail. Instead, you should either stop LifeKeeper manually (using `/etc/init.d/lifekeeper stop-nofailover`) or take the system first to init state 1 followed by init state S.

5.3.5.15. Thread is Hung Messages on Shared Storage

In situations where the device checking threads are not completing fast enough, this can cause messages to be placed in the LifeKeeper log stating that a thread is hung. This can cause resources to be moved from one server to another and in worse case, cause a server to be killed.

Explanation

The FAILFASTTIMER (in `/etc/default/LifeKeeper`) defines the number of seconds that each device is checked to assure that it is functioning properly, and that all resources that are owned by a particular system are still accessible by that system and owned by it. The FAILFASTTIMER needs to be as small as possible to guarantee this ownership and to provide the highest data reliability. However if a device is busy, it may not be able to respond at peak loads in the specified time. When a device takes longer than the FAILFASTTIMER then LifeKeeper considers that device as possibly hung. If a device has not responded after 3 loops of the FAILFASTTIMER time period then LifeKeeper attempts to perform recovery as if the device has failed. The recovery process is defined by the tunable SCSIERROR. Depending on the setting of SCSIERROR the action can be a sendevent to perform local recovery and then a switchover if that fails or it can cause the system to halt.

Suggested Action:

In cases where a device infrequently has a hung message printed to the error log followed by a message that it is no longer hung and the number in parenthesis is always 1, there should be no reason for alarm. However, if this message is frequently in the log, or the number is 2 or 3, then two actions may be necessary:

- ◦ Attempt to decrease the load on the storage. If the storage is taking longer than 3 times the FAILFASTTIMER (3 times 5 or 15 seconds by default) then one should consider the load that is being placed on the storage and re-balance the load to avoid these long I/O delays. This will not only allow LifeKeeper to check the devices frequently, but it should also help the performance of the application using that device.

- ◦ If the load can not be reduced, then the FAILFASTTIMER can be increased from the default 5 seconds. This value should be as low as possible so slowly increase the value until the messages no longer occur, or occur infrequently.

Note: When the FAILFASTTIMER value is modified LifeKeeper must be stopped and restarted before the new value will take affect.

5.4. DataKeeper

SIOS DataKeeper for Linux provides an integrated data mirroring capability for LifeKeeper environments. This feature enables LifeKeeper resources to operate in shared and non-shared storage environments.

5.4.1. Mirroring with SIOS DataKeeper for Linux

SIOS DataKeeper for Linux offers an alternative for customers who want to build a high availability cluster (using SIOS LifeKeeper) without shared storage or who simply want to replicate business-critical data in real-time between servers.

SIOS DataKeeper uses either synchronous or asynchronous volume-level mirroring to replicate data from the primary server (mirror source) to one or more backup servers (mirror targets).

DataKeeper Features

SIOS DataKeeper includes the following features:

- ◦ Allows data to be reliably, efficiently and consistently mirrored to remote locations over any TCP/IP-based Local Area Network (LAN) or Wide Area Network (WAN).
- ◦ Supports synchronous or asynchronous mirroring.
- ◦ Transparent to the applications involved because replication is done at the block level below the file system.
- ◦ Supports multiple simultaneous mirror targets including cascading failover to those targets when used with LifeKeeper.
- ◦ Built-in network compression allows higher maximum throughput on Wide Area Networks.
- ◦ Supports all major file systems (see the SPS for Linux Release Notes product description for more information regarding journaling file system support).
- ◦ Provides failover protection for mirrored data.
- ◦ Integrates into the LifeKeeper Graphical User Interface.
- ◦ Fully supports other LifeKeeper Application Recovery Kits.
- ◦ Automatically resynchronizes data between the primary server and backup servers upon system recovery.
- ◦ Monitors the health of the underlying system components and performs a local recovery in the event of failure.
- ◦ Supports STONITH devices for I/O fencing. For details, refer to the STONITH topic.

Synchronous vs. Asynchronous Mirroring

Understanding the differences between synchronous and asynchronous mirroring will help you choose the appropriate mirroring method for your application environment.

Synchronous Mirroring

SIOS DataKeeper provides real-time mirroring by employing a synchronous mirroring technique in which data is written simultaneously on the primary and backup servers. For each write operation, DataKeeper forwards the write to the target device(s) and awaits remote confirmation before signaling I/O completion. The advantage of synchronous mirroring is a high level of data protection because it ensures that all copies of the data are always identical. However, the performance may suffer due to the wait for remote confirmation, particularly in a WAN environment.

Asynchronous Mirroring

With asynchronous mirroring, each write is made to the source device and then a copy is queued to be transmitted to the target device(s). This means that at any given time, there may be numerous committed write transactions that are waiting to be sent from the source to the target device. The advantage of asynchronous mirroring is better performance because writes are acknowledged when they reach the primary disk, but it can be less reliable because if the primary system fails, any writes that are in the asynchronous write queue will not be transmitted to the target. To mitigate this issue, SIOS DataKeeper makes an entry to an intent log file for every write made to the primary device. If a large amount of data is written, the I/O performance may decrease temporarily because that data takes priority in the queue for transmission to the other nodes.

The intent log is a bitmap file indicating which data blocks are out of sync between the primary and target mirrors. In the event of a server failure, the intent log can be used to avoid a full resynchronization (or resync) of the data.

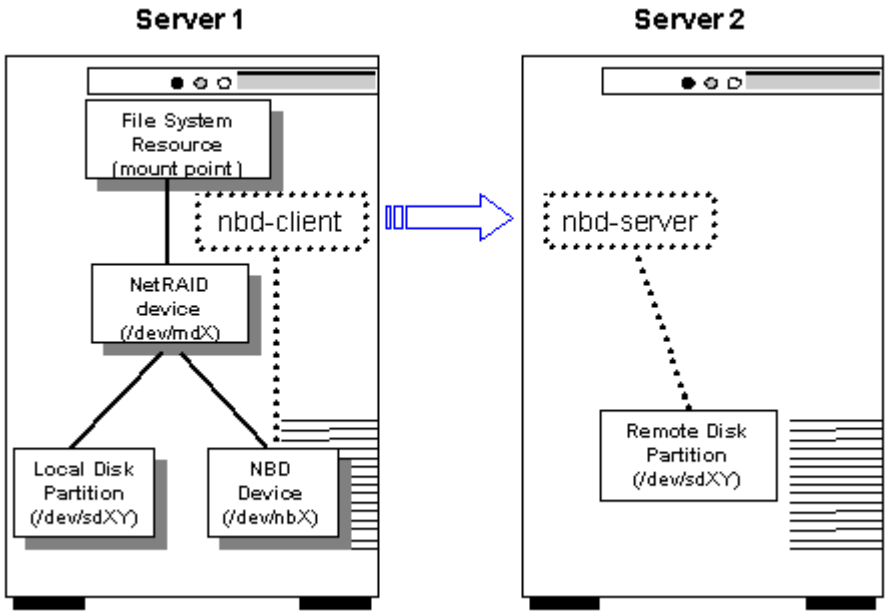
INFORMATION: For async mirrors, DataKeeper allows up to 4096 outstanding target writes to be queued. Set the `LKDR_ASYNC_LIMIT` in /etc/default/LifeKeeper to allow more writes to be queued.

Once the number of outstanding writes to the target reaches the limit, the mirror will revert to synchronous mode until the number drops below the set value.

Assuming a 4K block size, if the value is left at the default (asynchronous limit of 4096 writes), you would have a maximum of 16MB of data in transit.

5.4.2. How SIOS DataKeeper Works

SIOS DataKeeper creates and protects NetRAID devices. A NetRAID device is a RAID1 device that consists of a local disk or partition and a Network Block Device (NBD) as shown in the diagram below.



A LifeKeeper supported file system can be mounted on a NetRAID device like any other storage device. In this case, the file system is called a replicated file system. LifeKeeper protects both the NetRAID device and the replicated file system.

The NetRAID device is created by building the DataKeeper resource hierarchy. Extending the NetRAID device to another server will create the NBD device and make the network connection between the two servers. SIOS DataKeeper starts replicating data as soon as the NBD connection is made.

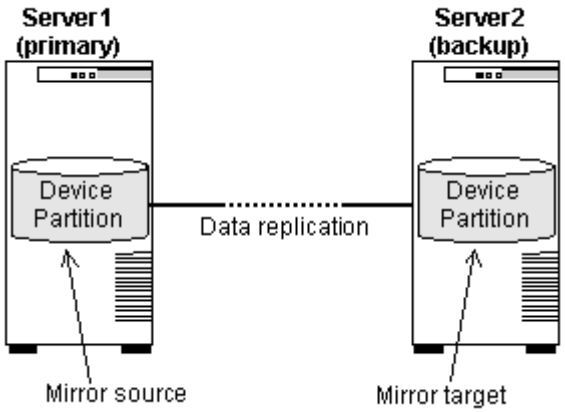
The nbd-client process executes on the primary server and connects to the nbd-server process running on the backup server.

Synchronization (and Resynchronization)

After the DataKeeper resource hierarchy is created and before it is extended, it is in a degraded mode; that is, data will be written to the local disk or partition only. Once the hierarchy is extended to the backup (target) system, SIOS DataKeeper synchronizes the data between the two systems and all subsequent writes are replicated to the target. If at any time the data gets “out-of-sync” (i.e., a system or network failure occurs) SIOS DataKeeper will automatically resynchronize the data on the source and target systems. If the mirror was configured to use an intent log (bitmap file), SIOS DataKeeper uses it to determine what data is out-of-sync so that a full resynchronization is not required. If the mirror was not configured to use a bitmap file, then a full resync is performed after any interruption of data replication.

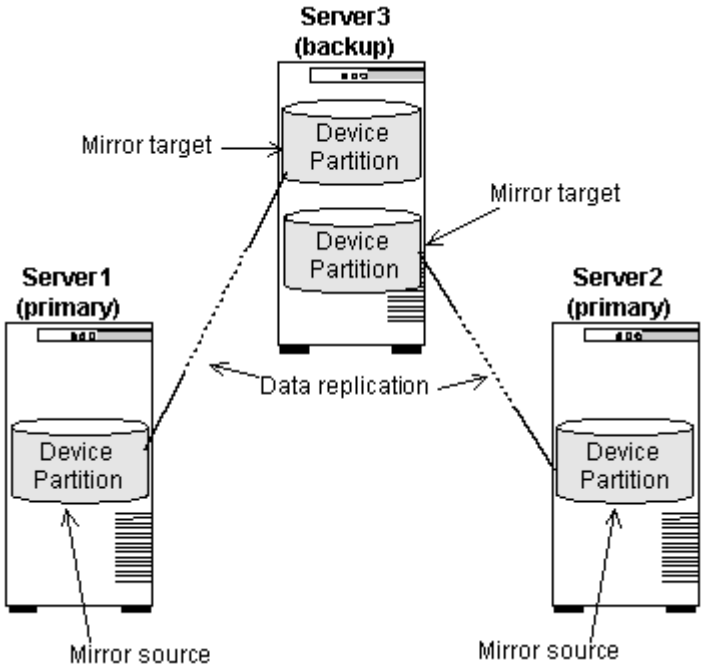
Standard Mirror Configuration

The most common mirror configuration involves two servers with a mirror established between local disks or partitions on each server, as shown below. Server1 is considered the primary server containing the mirror source. Server2 is the backup server containing the mirror target.



N+1 Configuration

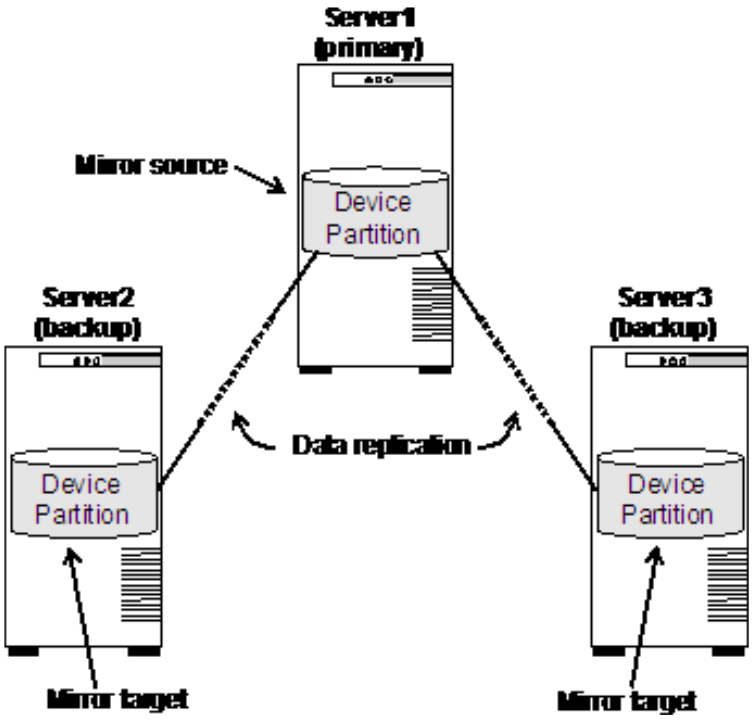
A commonly used variation of the standard mirror configuration above is a cluster in which two or more servers replicate data to a common backup server. In this case, each mirror source must replicate to a separate disk or partition on the backup server, as shown below.



Multiple Target Configuration

When used with an appropriate Linux distribution and kernel version 2.6.7 or higher, SIOS DataKeeper can also replicate data from a single disk or partition on the primary server to multiple backup systems,

as shown below.



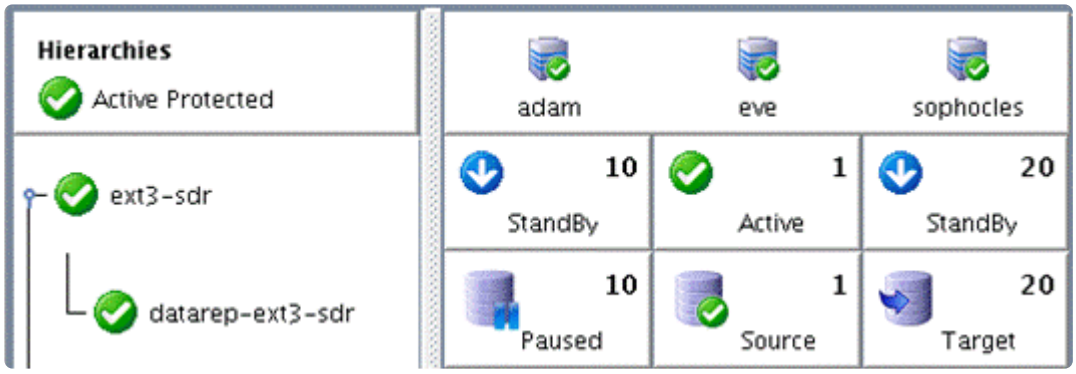
A given source disk or partition can be replicated to a maximum of 7 mirror targets, and each mirror target must be on a separate system (i.e. a source disk or partition cannot be mirrored to more than one disk or partition on the same target system).

This type of configuration allows the use of LifeKeeper's cascading failover feature, providing multiple backup systems for a protected application and its associated data.

To avoid a full resync to all targets when a mirror is started, the bitmap from the previous source must first be merged before the remaining targets in the cluster can be reconnected. Prior to v9.3.2, if the previous source was not available when the mirror was started, a full resync was automatically done to each target. Starting with v9.3.2, when the mirror is started on a system it will wait for the previous source to join the cluster before connecting targets. When the previous source joins the cluster, its bitmap is merged so that all targets can join with a partial resync. When the mirror is stopped and targets are in-sync, no previous source is needed to start the mirror and replicate to targets. If the previous source is not available to rejoin the cluster, targets can manually be resynced with a full resync using the "mirror_action fullresync" command. The variable LKDR_WAIT_FOR_PREVIOUS_SOURCE_TIMEOUT in */etc/default/LifeKeeper* determines the resync behavior (refer to the DataKeeper Parameters List for more information).

SIOS DataKeeper Resource Hierarchy

The following example shows a typical DataKeeper resource hierarchy as it appears in the LifeKeeper GUI:



The resource *datarep-ext3-sdr* is the NetRAID resource, and the parent resource *ext3-sdr* is the file system resource. Note that subsequent references to the DataKeeper resource in this documentation refer to both resources together. Because the file system resource is dependent on the NetRAID resource, performing an action on the NetRAID resource will also affect the file system resource above it.

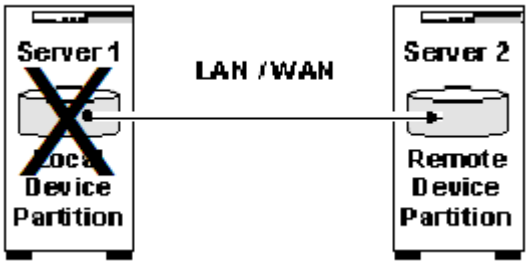
Failover Scenarios

Failover Scenarios – 2 nodes

The following four examples show what happens during a failover using SIOS DataKeeper. In these examples, the LifeKeeper for Linux cluster consists of two servers, Server 1 (primary server) and Server 2 (backup server).

Scenario 1

Server 1 has successfully completed its replication to Server 2 after which Server 1 becomes inoperable.



Result: Failover occurs. Server 2 now takes on the role of primary server and operates in a degraded mode (with no backup) until Server 1 is again operational. SIOS DataKeeper will then initiate a resynchronization from Server 2 to Server 1. This will be a full resynchronization on kernel 2.6.18 and lower. On kernels 2.6.19 and later or with Red Hat Enterprise Linux 5.4 kernels 2.6.18-164 or later (or a supported derivative of Red Hat 5.4 or later), the resynchronization will be partial, meaning only the changed blocks recorded in the bitmap files on the source and target will need to be synchronized.

Note: SIOS DataKeeper sets the following flag on the server that is currently acting as

the mirror source:

```
$LKROOT/subsys/scsi/resources/netraid/$TAG_last_owner
```

When Server 1 fails over to Server 2, this flag is set on Server 2.Thus, when Server 1 comes back up; SIOS DataKeeper removes the last owner flag from Server1. It then begins resynchronizing the data from Server 2 to Server 1.

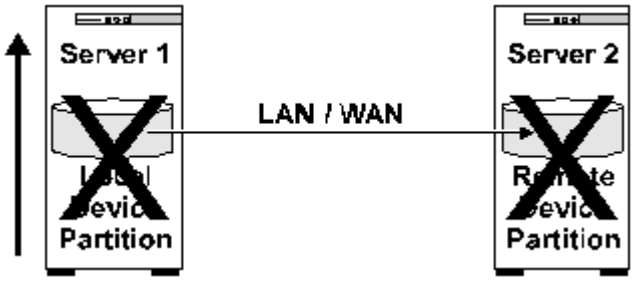
Scenario 2

Considering scenario 1, Server 2 (still the primary server) becomes inoperable during the resynchronization with Server 1 (now the backup server).

Result: Because the resynchronization process did not complete successfully, there is potential for data corruption. As a result, LifeKeeper will not attempt to fail over the DataKeeper resource to Server 1. Only when Server 2 becomes operable will LifeKeeper attempt to bring the DataKeeper resource in-service (ISP) on Server 2.

Scenario 3

Both Server 1 (primary) and Server 2 (target) become inoperable. Server 1 (primary) comes back up first.



Result: Server 1 will not bring the DataKeeper resource in-service. The reason is that if a source server goes down, and then it cannot communicate with the target after it comes back online, it sets the following flag:

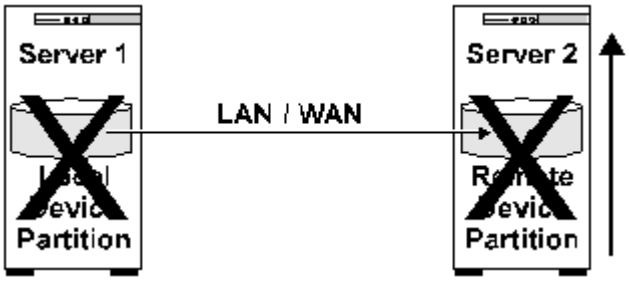
```
$LKROOT/subsys/scsi/resources/netraid/$TAG_data_corrupt
```

This is a safeguard to avoid resynchronizing data in the wrong direction. In this case you will need to force the mirror online on Server1, which will delete the data_corrupt flag and bring the resource into service on Server 1. See Force Mirror Online

Note: The user must be certain that Server 1 was the last primary before removing the $TAG_data_corrupt file. Otherwise data corruption might occur. You can verify this by checking for the presence of the last_owner flag.

Scenario 4

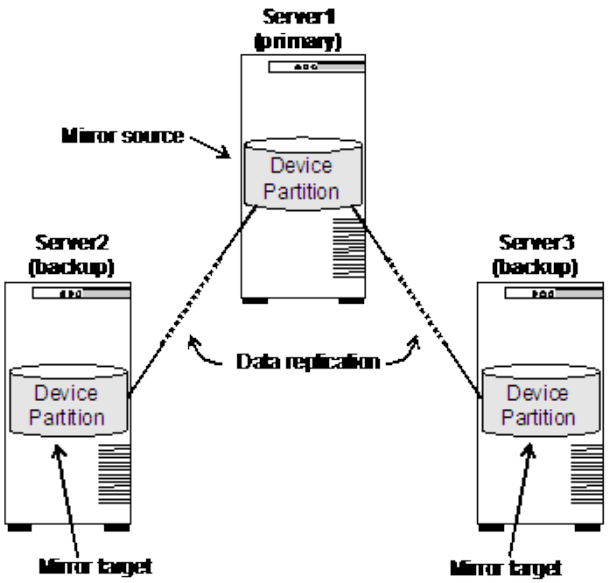
Both Server 1 (primary) and Server 2 (target) become inoperable. Server 2 (target) comes back up first.



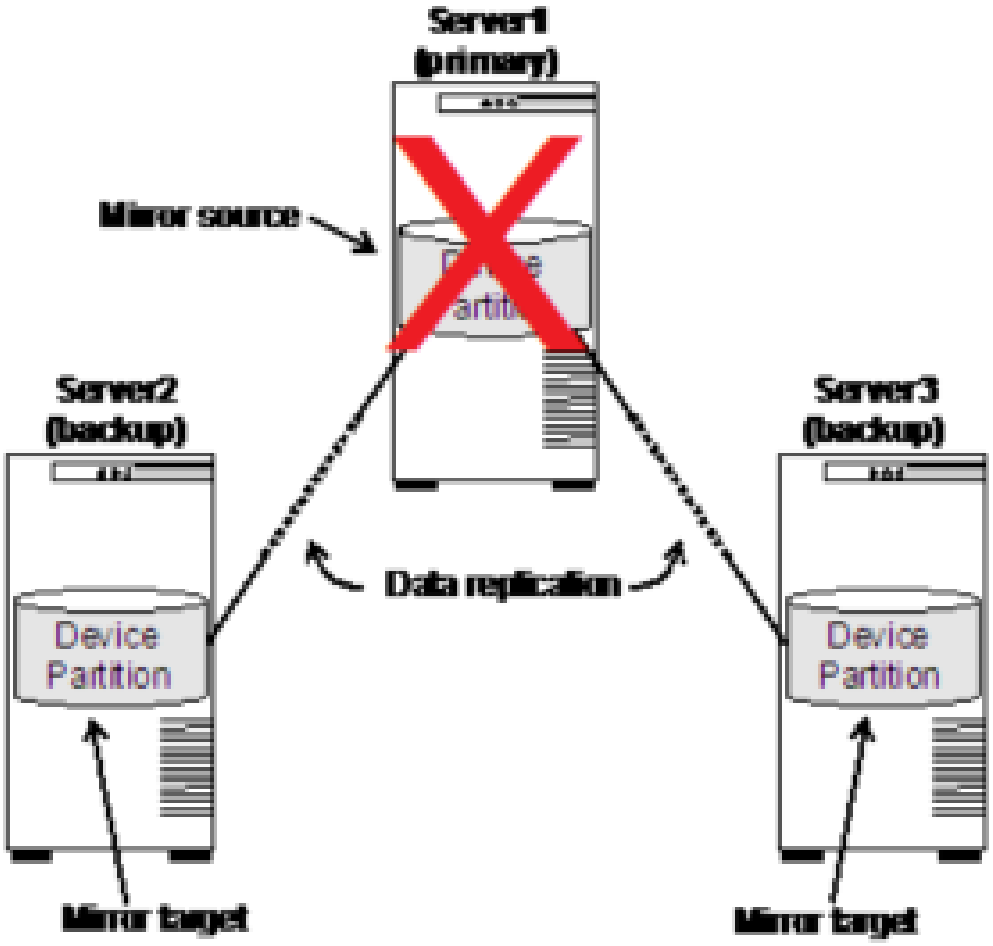
Result: LifeKeeper will not bring the DataKeeper resource ISP on Server 2. When Server 1 comes back up, LifeKeeper will automatically bring the DataKeeper resource ISP on Server 1.

Failover Scenario – 3 nodes

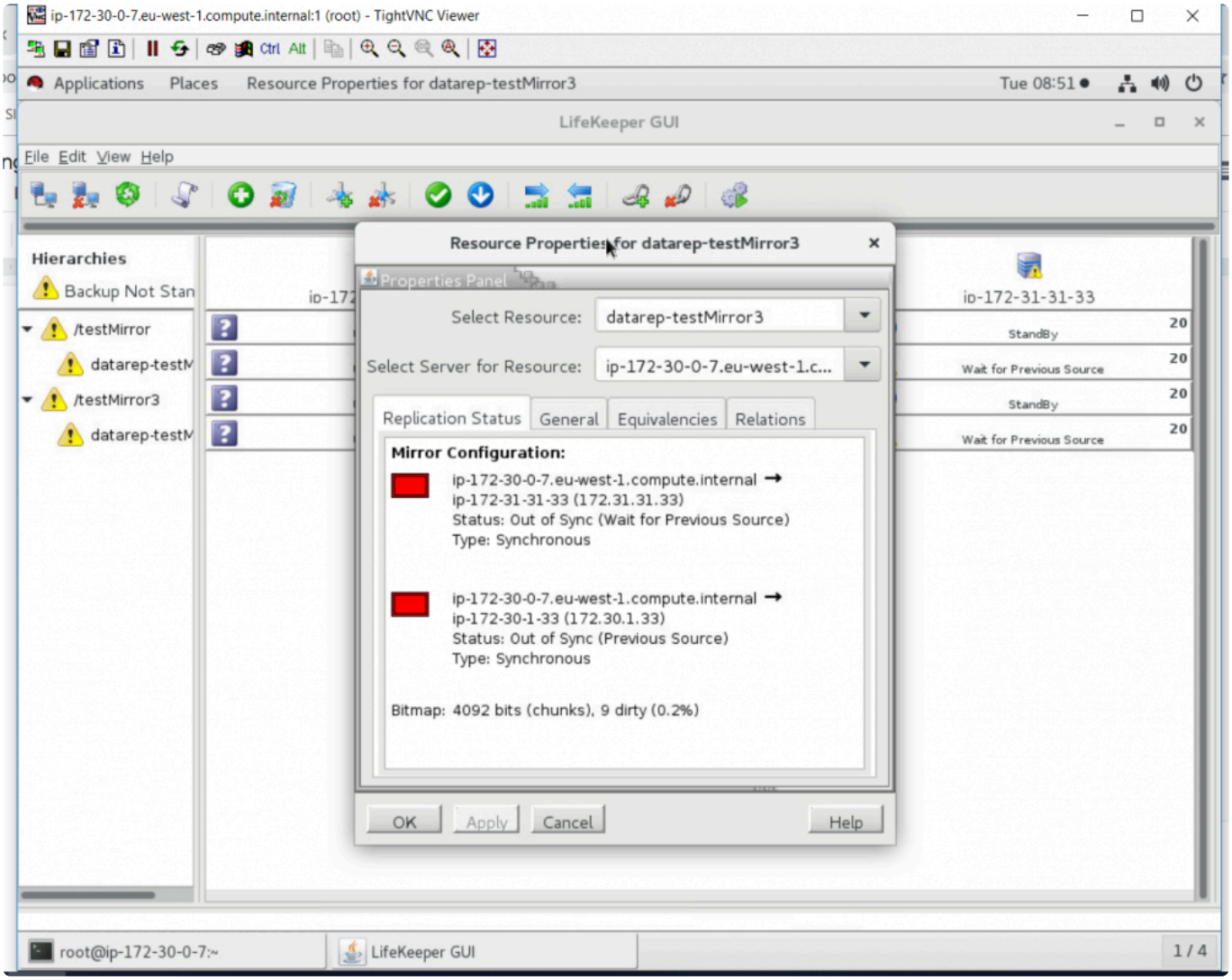
The following example shows what happens during a failover using SIOS DataKeeper. In this example the LifeKeeper for Linux cluster consists of three servers, Server 1 (primary server), Server 2 (backup server) and Server 3 (backup server).

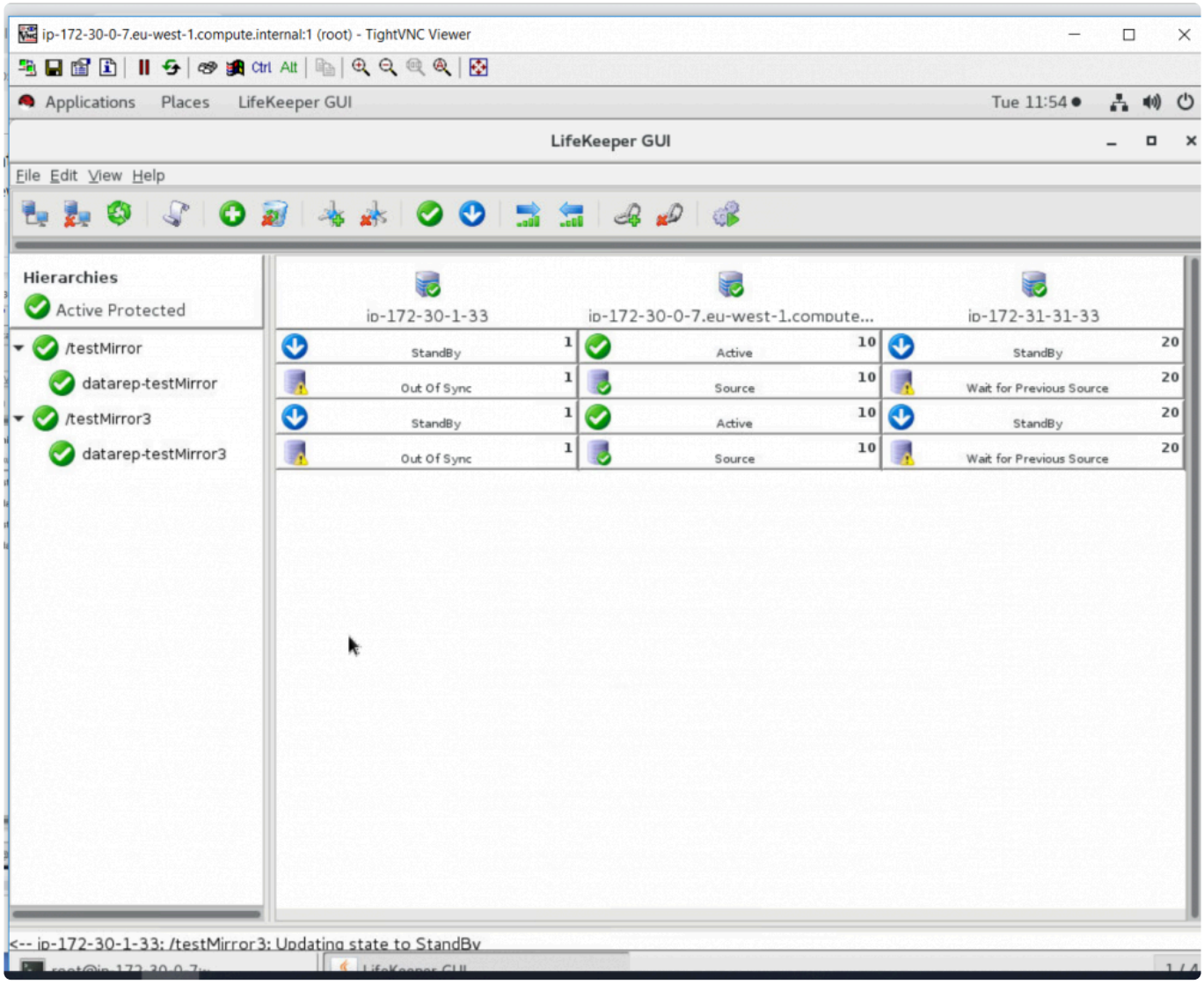


Server 1 (priority 1) has successfully completed its replication to Server 2 (priority 10) and Server 3 (priority 20) after which Server 1 becomes inoperable.

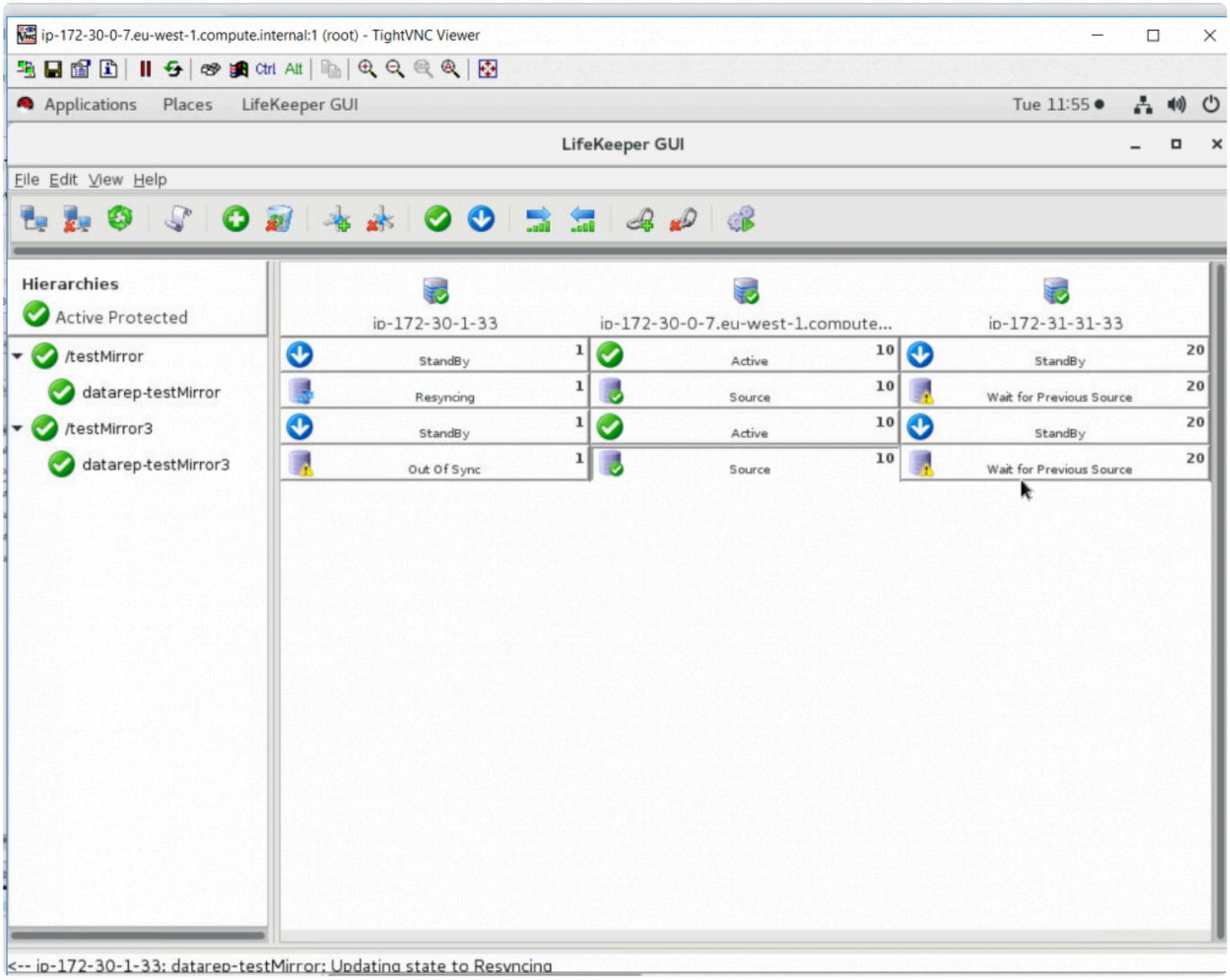


Result: Failover occurs to the next highest priority, Server 2. Server 2 now takes on the role of primary server. Prior to release v9.3.2 Server 3 will be added to the mirror with a full resynchronization. With v9.3.2 Server 2 **waits** for Server 1 (previous server) to return to the cluster before resuming replication to Server 3. This allows the bitmap from Server 1 to be merged with the bitmap from Server 2, allowing for a partial resync to both Server 1 and Server 3. While waiting for Server 1 to reconnect to the cluster, the LifeKeeper GUI will show the status of Server 3 as “Out of Sync (Wait for Previous Source)”. The status of Server 1 will be “Unknown” while the server is not connected. When it initially connects the GUI status will show “Out of Sync”. The properties page for the mirror will identify it as “Out of Sync (Previous Source)”.


ip-172-30-0-7.eu-west-1.compute.internal:1 (root) - TightVNC Viewer
Applications Places Resource Properties for datarep-testMirror3 Tue 08:51
LifeKeeper GUI
File Edit View Help
Hierarchies
Backup Not Stan
/testMirror
datarep-testM
/testMirror3
datarep-testM
Resource Properties for datarep-testMirror3
Properties Panel
Select Resource: datarep-testMirror3
Select Server for Resource: ip-172-30-0-7.eu-west-1.c...
Replication Status General Equivalencies Relations
Mirror Configuration:
ip-172-30-0-7.eu-west-1.compute.internal
ip-172-31-31-33 (172.31.31.33)
Status: Out of Sync (Wait for Previous Source)
Type: Synchronous
ip-172-30-0-7.eu-west-1.compute.internal
ip-172-30-1-33 (172.30.1.33)
Status: Out of Sync (Previous Source)
Type: Synchronous
Bitmap: 4092 bits (chunks), 9 dirty (0.2%)
OK Apply Cancel Help
ip-172-31-31-33
StandBy 20
Wait for Previous Source 20
StandBy 20
Wait for Previous Source 20
root@ip-172-30-0-7:~
LifeKeeper GUI
1 / 4



Once Server 1 reconnects, its bitmap is merged and a resynchronization begins, at which point its status is shown as “Resyncing”.



When resynchronization completes, its status will update to “Target” and Server 3 will begin resynchronization with its status set to “Resyncing”.

Note: SIOS DataKeeper sets the follow flags to track the mirror source:

$LKROOT/subsys/scsi/resources/netraid/$TAG_last_owner

$LKROOT/subsys/scsi/resources/netraid/$TAG_source

The *$TAG_last_owner* flag is on the system that is currently acting as the mirror source while the *$TAG_source* flag contains the name of the system that was source at the last point in time that the local node was part of the mirror.

When Server 1 fails over to Server 2, *$TAG_last_owner* flag is set on Server 2. The *$TAG_source* flag on Server 2 identifies Server 1 as the previous source (that has the bitmap needed to do a partial resync to Server 1 and Server 3). When Server 1 comes back up, SIOS DataKeeper removes the *$TAG_last_owner* flag from Server 1. Server 2 then merges the bitmap from Server 1 and begins resynchronizing the data from Server 2 to Server 1. When resynchronization is complete to Server 1 the *$TAG_source* flag on Server 1 is updated with the name of Server 2. After Server 1 is synchronized, Server 2 will perform the same resynchronization to Server 3. When that resynchronization is complete

to Server 3 the *$TAG_source* flag on Server 3 is updated with the name of Server 2.

5.4.3. SIOS DataKeeper Installation and Configuration

Installing and Configuring SIOS DataKeeper for Linux

Before Configuring Your DataKeeper Resources

The following topics contain information for consideration before beginning to create and administer your DataKeeper resources. They also describe the three types of DataKeeper resources. Please refer to the LifeKeeper Configuration section for instructions on configuring LifeKeeper Core resource hierarchies.

5.4.3.1. Hardware and Software Requirements

Your LifeKeeper configuration should meet the following requirements prior to the installation of SIOS DataKeeper.

Hardware Requirements

- **Servers** – Two or more LifeKeeper for Linux supported servers.
- **IP Network Interface Cards** – Each server requires at least one network interface card. Remember, however, that a LifeKeeper cluster requires two communication paths; two separate LAN-based communication paths using dual independent sub-nets are recommended, and at least one of these should be configured as a private network. However using a combination of TCP and TTY is also supported.

> **Note:** Due to the nature of software mirroring, network traffic between servers can be heavy. Therefore, it is recommended that you implement a separate private network for your SIOS DataKeeper devices which may require additional network interface cards on each server.

- **Disks or Partitions** – Disks or partitions on the primary and backup servers that will act as the source and target disks or partitions. The target disks or partitions must be at least as large as the source disk or partition.

> **Note:** With the release of SIOS Data Replication 7.1.1, it became possible to replicate an entire disk, one that has not been partitioned (i.e. */dev/sdd*). Previous versions of SIOS Data Replication required that a disk be partitioned (even if it was a single large partition; i.e._ /dev/sdd1_) before it could be replicated. SIOS Data Replication 7.1.1 removed that restriction.

> With the release of SPS for Linux v9.5.0, all disks must be uniquely identifiable. DataKeeper had allowed the device name (i.e. /dev/sdd) to be used to identify a device but in some situations the device names can change that can lead to data corruption. The use of a GPT partition table can provide a unique identifier.

Software Requirements

- **Operating System** – SIOS DataKeeper can be used with any major Linux distribution based on the 2.6 Linux kernel. See the SPS for Linux Release Notes for a list of supported distributions. Asynchronous mirroring and intent logs are supported only on distributions that use a 2.6.16 or later Linux kernel. Multiple target support (i.e., support for more than 1 mirror target) requires a

2.6.7 or later Linux kernel.

- **LifeKeeper Installation Script** – In most cases, you will need to install the following package (see the “Product Requirements” section in the SPS for Linux Release Notes for specific SIOS DataKeeper requirements):

  HADR-generic-2.6

  This package must be installed on each server in your LifeKeeper cluster prior to the installation of SIOS DataKeeper. The HADR package is located on the SPS Installation Image File, and the appropriate package is automatically installed by the Installation **setup** script.

- **LifeKeeper Software** – You must install the same version of the LifeKeeper Core on each of your servers. You must also install the same version of each recovery kit that you plan to use on each server. See the SPS for Linux Release Notes for specific SPS requirements.

- **SIOS DataKeeper software** – Each server in your SPS cluster requires SIOS DataKeeper software. Please see the SPS for Linux Installation Guide for specific instructions on the installation and removal of SIOS DataKeeper.

5.4.3.2. General Configuration

- The size of the target disks or partitions (on the backup servers) must be equal to or greater than the size of the source disk or partition (on the primary server).
- Once the DataKeeper resource is created and extended, the synchronization process will delete existing data on the target disks or partitions and replace it with data from the source partition.

5.4.3.3. DataKeeper for Linux Network Configuration

- The network path that is chosen for data replication between each pair of servers must also already be configured as a LifeKeeper communication path between those servers. To change the network path, see Changing the Data Replication Path.

- Avoid a configuration to add virtual IP address with IP Recovery Kit to the network interface that DataKeeper users for data replication. Since the communication line is temporarily disconnected while IP Recovery Kit uses the network interface, data replication may stop at unexpected timing and unnecessary resynchronization may occur.

- This release of SIOS DataKeeper does not support Automatic Switchback for DataKeeper resources. Additionally, the Automatic Switchback restriction is applicable for any other LifeKeeper resource sitting on top of a DataKeeper resource.

- If using Fusion-io, see the Network section of Clustering with Fusion-io for further network configuration information.

5.4.3.4. Changing the Data Replication Path

Starting with LK 7.1, IP addresses for mirror endpoints can be modified using `lk_chg_value`. For example, to change a mirror endpoint from IP address 192.168.0.1 to 192.168.1.1:

`# lkstop` (`lk_chg_value` cannot be run while LifeKeeper is running)

```
# lk_chg_value -o 192.168.0.1 -n 192.168.1.1
```

```
# lkstart
```

Execute these commands on all servers involved in the mirror(s) that are using this IP address.

Note: This command will also modify communication paths that are using the address in question.

5.4.3.5. Network Bandwidth Requirements

Prior to installing SIOS DataKeeper, you should determine the network bandwidth requirements for replicating your current configuration whether you are employing virtual machines or using physical Linux servers. If you are employing virtual machines (VMs), use the method Measuring Rate of Change on a Linux System (Physical or Virtual) to measure the rate of change for the virtual machines that you plan to replicate. This value indicates the amount of network bandwidth that will be required to replicate the virtual machines.

After determining the network bandwidth requirements, ensure that your network is configured to perform optimally. If your network bandwidth requirements are above your current available network capacity, you may need to consider one or more of the following options:

- Enable compression in SIOS DataKeeper (or in the network hardware, if possible)
- Increase your network capacity
- Reduce the amount of data being replicated
- Create a local, non-replicated storage repository for temporary data and swap files
- Manually schedule replication to take place daily at off-peak hours

5.4.3.5.1. Measuring Rate of Change on a Linux System (Physical or Virtual)

DataKeeper for Linux can replicate data across any available network. In Wide Area Network (WAN) configurations, special consideration must be given to the question, “Is there sufficient bandwidth to successfully replicate the partition and keep the mirror in the mirroring state as the source partition is updated throughout the day?”

Keeping the mirror in the mirroring state is critical because a switchover of the partition is not allowed unless the mirror is in the mirroring state.

SIOS DataKeeper handles short bursts of write activity by adding that data to its async queue. However, make sure that over any extended period of time, the disk write activity for all replicated volumes combined remains, on average, below the amount of change that DataKeeper and your network can transmit.

If the network capacity is not sufficient to keep up with the rate of change that occurs on your disks, and the async queue fills up, the mirror will revert to synchronous behavior, which can negatively affect performance of the source server.

Measuring Basic Rate of Change

Use the following command to determine file(s) or partition(s) to be mirrored. For example /dev/sda3, and then measure the amount of data written in a day:

```
MB_START=`awk '/sda3 / { print $10 / 2 / 1024 }' /proc/diskstats`
```

… wait for a day …

```
MB_END=`awk '/sda3 / { print $10 / 2 / 1024 }' /proc/diskstats`
```

The daily rate of change, in MB, is then `MB_END - MB_START`.

SIOS DataKeeper can mirror daily, approximately:

T1 (1.5Mbps) – 14,000 MB/day (14 GB)

T3 (45Mbps) – 410,000 MB/day (410 GB)

Gigabit (1Gbps) – 5,000,000 MB/day (5 TB)

Measuring Detailed Rate of Change

The best way to collect Rate of Change data is to log disk write activity for some period of time (one day,

for instance) to determine what the peak disk write periods are.

To track disk write activity, create a cron job which will log the timestamp of the system followed by a dump of */proc/diskstats*. For example, to collect disk stats every two minutes, add the following link to */etc/crontab:*

```
*/2 * * * * root ( date ; cat /proc/diskstats ) >> /path_to/
filename.txt
```

… wait for a day, week, etc … then disable the cron job and save the resulting data file in a safe location.

Analyze Collected Detailed Rate of Change Data

The `roc-calc-diskstats` utility analyzes data collected in the previous step. This utility takes a `/proc/diskstats` output file that contains output, logged over time, and calculates the rate of change of the disks in the dataset.

Click Here to download `roc-calc-diskstats`

Usage:

```
# ./roc-calc-diskstats <interval> <start_time> <diskstats-data-file> [dev-
list]
```

Usage Example (Summary only):

```
# ./roc-calc-diskstats 2m "Jul 22 16:04:01" /root/diskstats.txt
sdb1,sdb2,sdc1 > results.txt
```

The above example dumps a summary (with per disk peak I/O information) to *results.txt*

Usage Example (Summary + Graph Data):

```
# export OUTPUT_CSV=1

# ./roc-calc-diskstats 2m "Jul 22 16:04:01" /root/diskstats.txt
sdb1,sdb2,sdc1 2> results.csv > results.txt
```

The above example dumps graph data to *results.csv* and the summary (with per disk peak I/O information) to *results.txt*

Example Results (from results.txt)

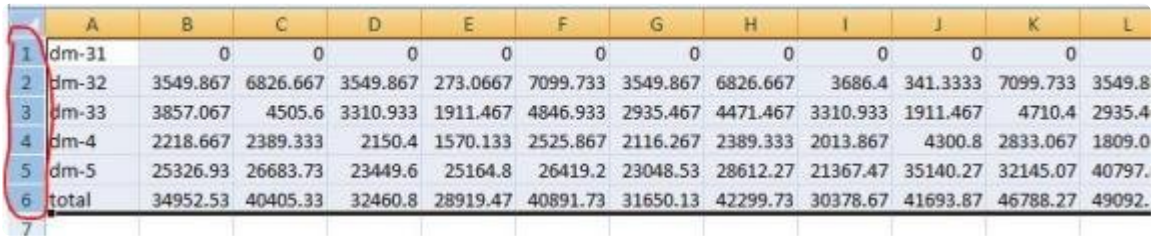
```
Sample start time: Tue Jul 12 23:44:01 2011

Sample end time: Wed Jul 13 23:58:01 2011

Sample interval: 120s #Samples: 727 Sample length: 87240s

(Raw times from file: Tue Jul 12 23:44:01 EST 2011, Wed Jul 13 23:58:01
EST 2011)

Rate of change for devices dm-31, dm-32, dm-33, dm-4, dm-5, total

dm-31 peak:0.0 B/s (0.0 b/s) (@ Tue Jul 12 23:44:01 2011) average:0.0 B/s
(0.0 b/s)

dm-32 peak:398.7 KB/s (3.1 Mb/s) (@ Wed Jul 13 19:28:01 2011) average:19.5
KB/s (156.2 Kb/s)

dm-33 peak:814.9 KB/s (6.4 Mb/s) (@ Wed Jul 13 23:58:01 2011) average:11.6
KB/s (92.9 Kb/s)

dm-4 peak:185.6 KB/s (1.4 Mb/s) (@ Wed Jul 13 15:18:01 2011) average:25.7
KB/s (205.3 Kb/s)

dm-5 peak:2.7 MB/s (21.8 Mb/s) (@ Wed Jul 13 10:18:01 2011) average:293.0
KB/s (2.3 Mb/s)

total peak:2.8 MB/s (22.5 Mb/s) (@ Wed Jul 13 10:18:01 2011) average:349.8
KB/s (2.7 Mb/s)
```

Graph Detailed Rate of Change Data

To help understand your specific bandwidth needs over time, SIOS has created a template spreadsheet called diskstats-template.xlsx. This spreadsheet contains sample data which can be overwritten with the data collected by `roc-calc-diskstats`.

Click Here to download diskstats-template.xslx

1. Open results.csv, and select **all rows**, including the total column.



	A	B	C	D	E	F	G	H	I	J	K	L
1	dm-31	0	0	0	0	0	0	0	0	0	0	
2	dm-32	3549.867	6826.667	3549.867	273.0667	7099.733	3549.867	6826.667	3686.4	341.3333	7099.733	3549.8
3	dm-33	3857.067	4505.6	3310.933	1911.467	4846.933	2935.467	4471.467	3310.933	1911.467	4710.4	2935.4
4	dm-4	2218.667	2389.333	2150.4	1570.133	2525.867	2116.267	2389.333	2013.867	4300.8	2833.067	1809.0
5	dm-5	25326.93	26683.73	23449.6	25164.8	26419.2	23048.53	28612.27	21367.47	35140.27	32145.07	40797.
6	total	34952.53	40405.33	32460.8	28919.47	40891.73	31650.13	42299.73	30378.67	41693.87	46788.27	49092.
7												

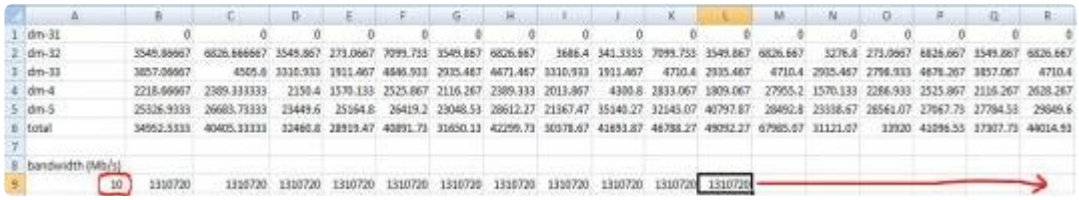
2. Open **diskstats-template.xlsx**, select the **diskstats.csv** worksheet.

3. In cell 1-A, right-click and select **Insert Copied Cells**.

4. Adjust the **bandwidth** value in the cell towards the bottom left of the worksheet to reflect an amount of bandwidth you have allocated for replication.

   Units: Megabits/second (Mb/sec)

> Note: The cells to the right will automatically be converted to bytes/sec to match the raw data collected.

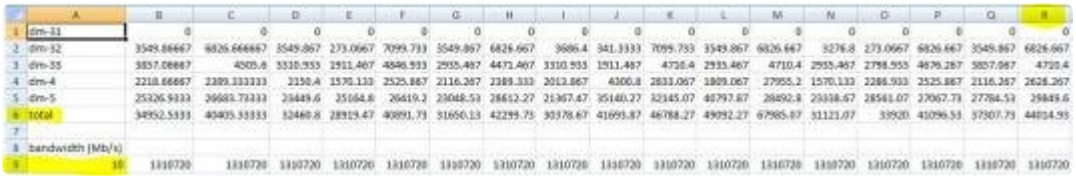


5. Make a note of the following row/column numbers:

   a. Total (row 6 in screenshot below)

   b. Bandwidth (row 9 in screenshot below)

   c. Last datapoint (column R in screenshot below)



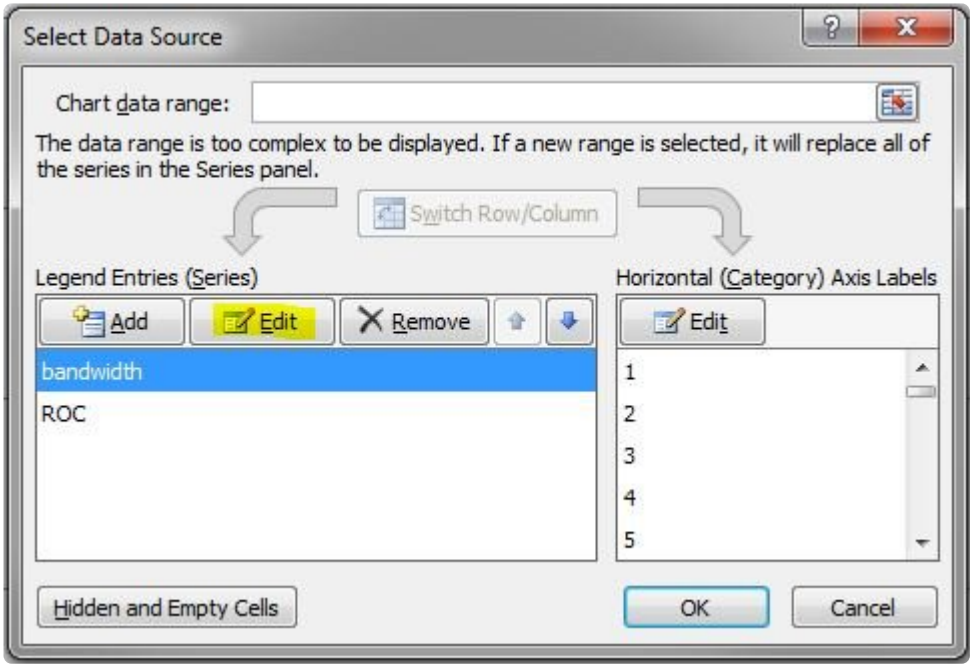
6. Select the **bandwidth vs ROC** worksheet.

7. Right-click on the graph and select **Select Data…**

   a. Adjust **Bandwidth Series**

      i. From the **Series** list on the left, select **bandwidth**

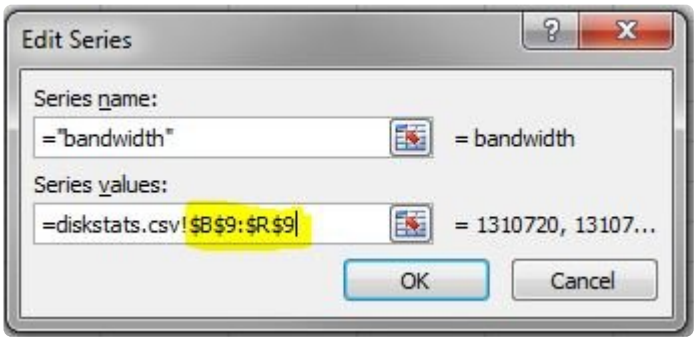
      ii. Click **Edit**



iii. Adjust the **Series Values**: field with the following syntax:

`"=diskstats.csv!$B$<row>:$<final_column>$<row>"`

example: `"=diskstats.csv!$B$9:$R:$9"`

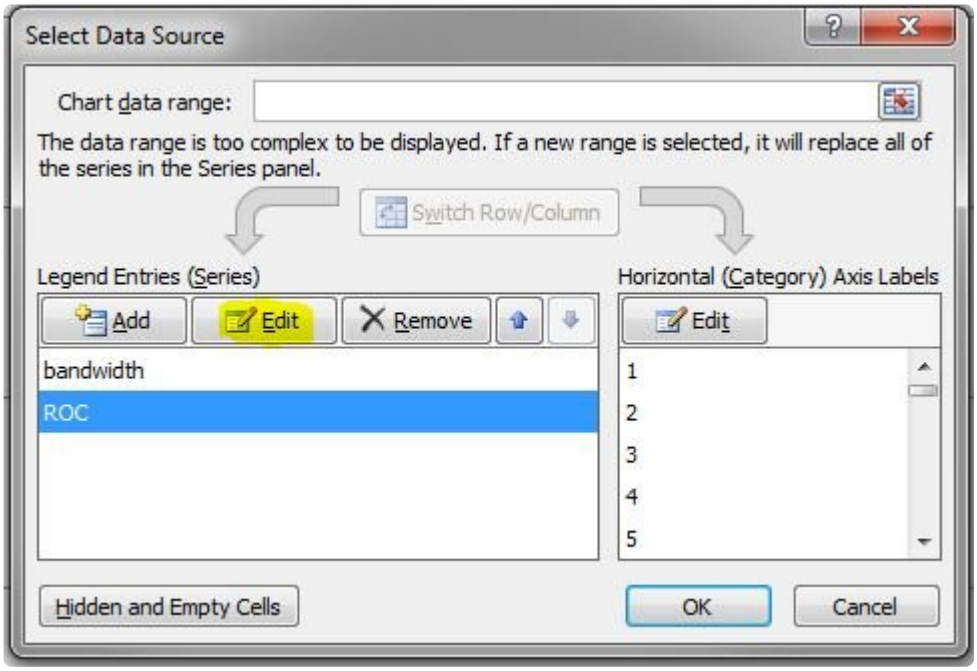


iv. Click **OK**

b. Adjust **ROC Series**

i. From the **Series** list on the left, select **ROC**

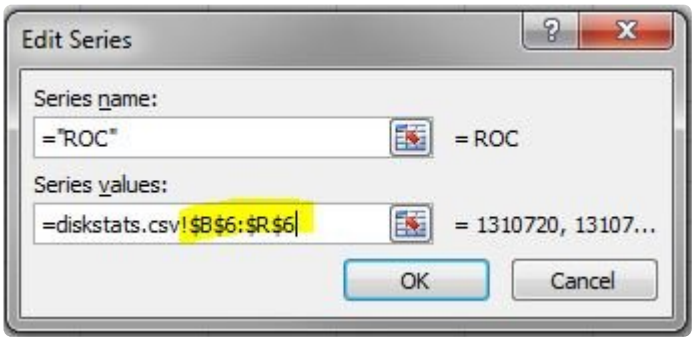
ii. Click **Edit**



iii. Adjust the **Series Values:** field with the following syntax:

`"=diskstats.csv!$B$<row>:$<final_column>$<row>"`

example: `"=diskstats.csv!$B$6:$R:$6"`



iv. Click **OK**

c. Click **OK** to exit the Wizard

8. The Bandwidth vs ROC graph will update. Please analyze your results to determine if you have sufficient bandwidth to support replication of your data.

roc-calc-diskstats

```
#!/usr/bin/perl
# Copyright (c) 2011, SIOS Technology, Corp.
# Author: Paul Clements
use strict;
sub msg {
        printf STDERR _;
}
sub dbg {
        return if (! $ENV{'ROC_DEBUG'});
        msg _;
}
$0 =~ s@^.*/@@; # basename
sub usage {
        msg "Usage: $0 <interval> <start-time> <iostat-data-file> [dev-lis
t]\n";
        msg "\n";
        msg "This utility takes a /proc/diskstats output file that contains\
n";
        msg "output, logged over time, and calculates the rate of change of\
n";
        msg "the disks in the dataset\n";
        msg "OUTPUT_CSV=1 set in env. dumps the full stats to a CSV file on ST
DERR\n";
        msg "\n";
        msg "Example: $0 1hour \"jun 23 12pm\" steeleye-iostat.txt sdg,sdh\n";
        msg "\n";
        msg "interval - interval between samples\n";
        msg "start time - the time when the sampling starts\n";
        msg "iostat-data-file - collect this with a cron job like:\n";
        msg "\t0 * * * * (date ; cat /proc/diskstats) >> /root/diskstats.txt\
n";
        msg "dev-list - list of disks you want ROC for (leave blank for al
l)\n";
        exit 1;
}
usage if (ARGV < 3);
my $interval = TimeHuman($ARGV[0]);
my $starttime = epoch($ARGV[1]);
my $file = $ARGV[2];
my $blksize = 512; # /proc/diskstats is in sectors
my %devs = map { $_ => 1 } split /,/, $ARGV[3];
my %stat;
my $firsttime;
my $lasttime;
# datestamp divides output
my %days = ( 'Sun' => 1, 'Mon' => 1, 'Tue' => 1, 'Wed' => 1,
                'Thu' => 1, 'Fri' => 1, 'Sat' => 1);
my %fields = ( 'major'  => 0,
                'minor' => 1,
                'dev' => 2,
                'reads' => 3,
                'reads_merged' => 4,
                'sectors_read' => 5,
                'ms_time_reading' => 6,
                'writes' => 7,
```

```
                'writes_merged' => 8,
                'sectors_written' => 9,
                'ms_time_writing' => 10,
                'ios_pending' => 11,
                'ms_time_total' => 12,
                'weighted_ms_time_total' => 13 );
my $devfield = $fields{'dev'};
my $calcfield = $ENV{'ROC_CALC_FIELD'} || $fields{'sectors_written'};
dbg "using field $calcfield\n";
open(FD, "$file") or die "Cannot open $file: $!\n";
foreach (<FD>) {
        chomp;
        _ = split;
        if (exists($days{$_[0]})) { # skip datestamp divider
                if ($firsttime eq '') {
                        $firsttime = join ' ', _[0..5];
                }
                $lasttime = join ' ', _[0..5];
                next;
        }
        next if ($_[0] !~ /[0-9]/); # ignore
        if (!%devs || exists $devs{$_[$devfield]}) {
                push {$stat{$_[$devfield]}}, $_[$calcfield];
        }
}
{$stat{'total'}} = totals(\%stat);
printf "Sample start time: %s\n", scalar(localtime($starttime));
printf "Sample end time: %s\n", scalar(localtime($starttime + (({$stat{'tota
l'}} - 1) * $interval)));
printf "Sample interval: %ss #Samples: %s Sample length: %ss\n", $interval,
({$stat{'total'}} - 1), ({$stat{'total'}} - 1) * $interval;
print "(Raw times from file: $firsttime, $lasttime)\n";
print "Rate of change for devices " . (join ', ', sort keys %stat) . "\n";
foreach (sort keys %stat) {
        my vals = {$stat{$_}};
        my ($max, $maxindex, $roc) = roc($_, $blksize, $interval, vals);
        printf "$_ peak:%sB/s (%sb/s) ( %s) average:%sB/s (%sb/s)\n", HumanSiz
e($max), HumanSize($max * 8), scalar localtime($starttime + ($maxindex * $inte
rval)), HumanSize($roc), HumanSize($roc * 8);
}
# functions
sub roc {
        my $dev = shift;
        my $blksize = shift;
        my $interval = shift;
        my ($max, $maxindex, $i, $first, $last, $total);
        my $prev = -1;
        my $first = $_[0];
        if ($ENV{'OUTPUT_CSV'}) { print STDERR "$dev," }
        foreach (_) {
                if ($prev != -1) {
                        if ($_ < $prev) {
                                dbg "wrap detected at $i ($_ < $prev)\n";
                                $prev = 0;
                        }
                        my $this = ($_ - $prev) * $blksize / $interval;
                        if ($this > $max) {
                                $max = $this;
```

```
                                $maxindex = $i;
                        }
                        if ($ENV{'OUTPUT_CSV'}) { print STDERR "$this," }
                }
                $prev = $_; # store current val for next time around
                $last = $_;
                $i++;
        }
        if ($ENV{'OUTPUT_CSV'}) { print STDERR "\n" }
        return ($max, $maxindex, ($last - $first) * $blksize / ($interval *
($i - 1)));
}
sub totals { # params: stat_hash
        my $stat = shift;
        my totalvals;
        foreach (keys %$stat) {
                next if (!defined($stat{$_}));
                my vals = {$stat{$_}};
                my $i;
                foreach (vals) {
                        $totalvals[$i++] += $_;
                }
        }
        return totalvals;
}
# converts to KB, MB, etc. and outputs size in readable form
sub HumanSize { # params: bytes/bits
        my $bytes = shift;
        my suffixes = ( '', 'K', 'M', 'G', 'T', 'P' );
        my $i = 0;
        while ($bytes / 1024.0 >= 1) {
                $bytes /= 1024.0;
                $i++;
        }
        return sprintf("%.1f %s", $bytes, $suffixes[$i]);
}
# convert human-readable time interval to number of seconds
sub TimeHuman { # params: human_time
        my $time = shift;
        my %suffixes = ('s' => 1, 'm' => 60, 'h' => 60 * 60, 'd' => 60 * 60 *
24);
        $time =~ /^([0-9]*)(.*?)$/;
        $time = $1;
        my $suffix = (split //, $2)[0]; # first letter from suffix
        if (exists $suffixes{$suffix}) {
                $time *= $suffixes{$suffix};
        }
        return $time;
}
sub epoch { # params: date
        my $date = shift;
        my $seconds = `date +'%s' --date "$date" 2>&1`;
        if ($? != 0) {
                die "Failed to recognize time stamp: $date\n";
        }
        return $seconds;
}
```

5.4.3.6. WAN Configuration

Using SIOS DataKeeper in a WAN environment requires special configuration due to the nature of WAN networking. The following tips are recommended:

- To prevent false failover, you should enable manual failover confirmation. Because most WANs are somewhat less reliable than LANs and because typical WAN mirror configurations will have only one comm path, this is usually a good idea. With this option enabled, a LifeKeeper failover will proceed only if the user confirms the failover by using the `lk_confirmso` command. Refer to the `lk_confirmso` man page for more details.

- Determine the proper value for `LKDR_ASYNC_LIMIT` , based upon the latency and throughput of the WAN. The `LKDR_ASYNC_LIMIT` parameter (which is set in */etc/default/LifeKeeper*) determines the number of outstanding asynchronous write operations (per mirror) that SIOS DataKeeper will allow. The default value for this parameter is 4096, but a larger number may increase write performance of the mirror. The disadvantage to increasing this value is that more data will be allowed to be out of sync between the primary and secondary at any given time. See the Asynchronous Mirroring Information in **Mirroring with SIOS DataKeeper for Linux** for further information on `LKDR_ASYNC_LIMIT`

- If you are mirroring a large amount of data over a slow WAN link, it may be desirable to avoid the initial full data resynchronization and instead ship or otherwise transport a copy of the source disk or partition to the remote (disaster recovery) site. To avoid the initial resynchronization, follow the steps in Avoiding Full Resynchronizations.

IMPORTANT: This procedure is not necessary if you created your hierarchy using the "New Replicated Filesystem" option in the LifeKeeper GUI. The "New Replicated Filesystem" option has been optimized to avoid the full initial resync.

- If the WAN link experiences periods of downtime in excess of 15 seconds on a regular basis, it may also be wise to tune the LifeKeeper heartbeat parameters. See Tuning the LifeKeeper Heartbeat for details.

5.4.3.7. SIOS DataKeeper for Linux Resource Types

When creating your DataKeeper resource hierarchy, LifeKeeper will prompt you to select a resource type. There are several different DataKeeper resource types. The following information can help you determine which type is best for your environment.

Replicate New File System

Choosing a New Replicated File System will create/extend the NetRAID device, mount the given mount point on the NetRAID device and put both the LifeKeeper supported file system and the NetRAID device under LifeKeeper protection. The local disk or partition will be formatted.

When this resource is extended to the second node, it will not do a full resync but only the file system metadata data. However when it is extended to the 3rd or more node, a full resync will be done to that node. To avoid a full resync, please follow the directions provided in Avoiding Full Resynchronizations.

> **CAUTION: All data will be deleted.**

Replicate Existing File System

Choosing Replicate Existing File System will use a currently mounted disk or partition and create a NetRAID device without deleting the data on the disk or partition. SIOS DataKeeper will unmount the local disk or partition, create the NetRAID device using the local disk or partition and mount the mount point on the NetRAID device. It will then put both the NetRAID device and the LifeKeeper supported file system under LifeKeeper protection.

DataKeeper Resource

Choosing a DataKeeper Resource will create/extend the NetRAID device and put it under LifeKeeper protection without a file system. You might choose this replication type if using a database that can use a raw I/O device.

In order to allow the user continued data access, SIOS DataKeeper will not attempt to unmount and delete a NetRAID device if it is currently mounted. The user must manually unmount it before attempting a manual switchover and mount it on the other server after the manual switchover.

Note: After the DataKeeper resource has been created, should you decide to protect a manually mounted file system with LifeKeeper, you can do so as follows:

1. Format the NetRAID device with a LifeKeeper supported file system.
2. Mount the NetRAID device.

3. Create and extend a file system hierarchy using the NetRAID device as if it were a shared storage disk or partition.

LifeKeeper's file system recovery kit will now be responsible for mounting/unmounting it during failover.

5.4.3.8. I/O Fencing with DataKeeper Configuration

In principle, I/O fencing using storage reservations is not available in DataKeeper configuration and split brain can occur. Therefore, you need to take steps to prevent a split brain from occurring via the following controls.

Exclusive Control using IP Resources

IP resources have an exclusive control functionality using duplication checking to ensure the same IP resource is not activated on multiple nodes. This can be used to avoid a split brain with DataKeeper resources.
Adding an IP resources as a child resource to all DataKeeper resources in the hierarchy can prevent the DataKeeper resource from starting on multiple nodes at the same time.
This method can only be used in environments where all the nodes in the cluster reside in the same subnet. This is required to perform the duplicate IP address checking.

Exclusive Control with Quorum/Witness Functionality

You can use the quorum/witness functionality in LifeKeeper to prevent multiple nodes from becoming active at the same time.
For details, please refer to the Quorum/Witness topic in the Technical Documentation.

5.4.3.9. Resource Configuration Tasks

You can perform all SIOS DataKeeper configuration tasks via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer and monitor SIOS DataKeeper resources.

Overview

The following tasks are available for configuring SIOS DataKeeper:

- **Create a Resource Hierarchy** – Creates a DataKeeper resource hierarchy.
- **Delete a Resource Hierarchy** – Deletes a DataKeeper resource hierarchy.
- **Extend a Resource Hierarchy** – Extends a DataKeeper resource hierarchy from the primary server to a backup server.
- **Unextend a Resource Hierarchy** – Unextends (removes) a DataKeeper resource hierarchy from a single server in the LifeKeeper cluster.
- **Create Dependency** – Creates a child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- **Delete Dependency** – Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- **In Service** – Activates a resource hierarchy.
- **Out of Service** – Deactivatesa resource hierarchy.
- **View/Edit Properties** – View or edit the properties of a resource hierarchy.

5.4.3.9.1. Creating a DataKeeper Resource Hierarchy

Perform the following on your primary server:

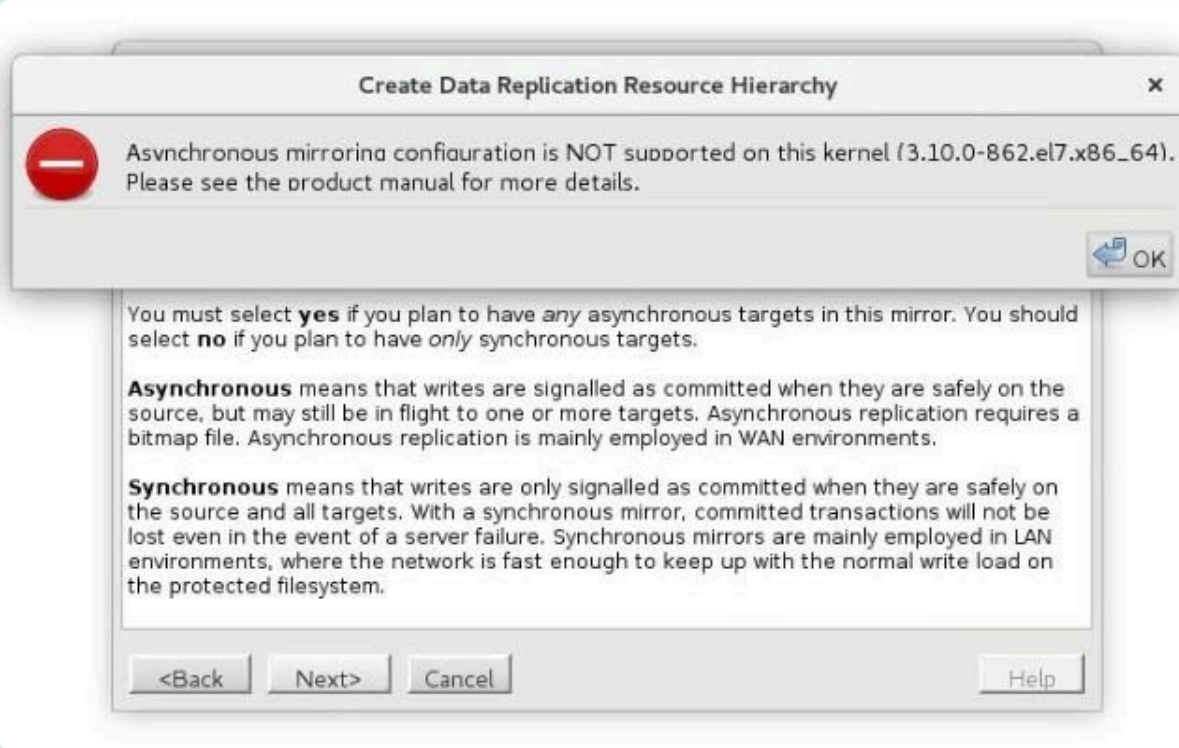
1. Select **Edit > Server > Create Resource Hierarchy**

   The ***Create Resource Wizard*** dialog will appear.

2. Select the **Data Replication** option from the drop down list and click **Next** to continue.

3. You will be prompted for the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter any error requiring you to correct the previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

Field	Tips
Switchback Type	You must select **intelligent switchback**. This means that after a failover to the backup server, an administrator must manually switch the DataKeeper resource back to the primary server. **CAUTION:** This release of SIOS DataKeeper does not support Automatic Switchback for DataKeeper resources. Additionally, the Automatic Switchback restriction is applicable for any other LifeKeeper resource sitting on top of a DataKeeper resource.
Server	Select the name of the server where the NetRAID device will be created (typically this is your primary server). All servers in your cluster are included in the drop down list box.
Hierarchy Type	Choose the data replication type you wish to create by selecting one of the following: • Replicate New File System • Replicate Existing File System • DataKeeper Resource
Bitmap File	Select or edit the name of the bitmap file used for intent logging. If you choose **None**, then an intent log will not be used and every resynchronization will be a full resync instead of a partial resync. **Important:** The bitmap file should not reside on a btrfs filesystem (or any other SPS for Linux unsupported filesystem). Placing data replication bitmap files on a btrfs filesystem (or any other SPS for Linux unsupported filesystem) will result in an “invalid argument” error when LifeKeeper tries to configure the mirror. The default location for the bitmap file is unde /opt/LifeKeeper. This default location should be changed if /opt/LifeKeeper resides on a btrf filesystem. **Note:** btrfs is currently not supported by SIOS Protection Suite for Linux.

Enable Asynchronous Replication?	Select **Yes** to allow this replication resource to support asynchronous replication to target systems. Select **No** if you will use synchronous replication to all targets. You will be asked later to choose the actual type of replication, asynchronous or synchronous, when the replication resource is extended to each target server. (See Mirroring with SIOS DataKeepe for a discussion of both replication types.) If you want the replication to any of these targets to be performed asynchronously, you should choose **Yes** here, even if the replication to other targets will be done synchronously. **Note:** If you select asynchronous mirroring in an environment where asynchronous mirrorin is not supported, the following message is displayed. **Asynchronous mirroring configuration is NOT supported on this kernel (3.10.0-862.el7.x86_64).** 

The next sequence of dialog boxes depends on which **Hierarchy Type** you have chosen. While some of the dialog boxes may be the same for each Hierarchy Type, their sequence and the required information may be slightly different. The next three topics take you through the remainder of the Hierarchy creation process.

- DataKeeper Resource
- Replicate New File System
- Replicate Existing File System

5.4.3.9.1.1. Replicate New File System

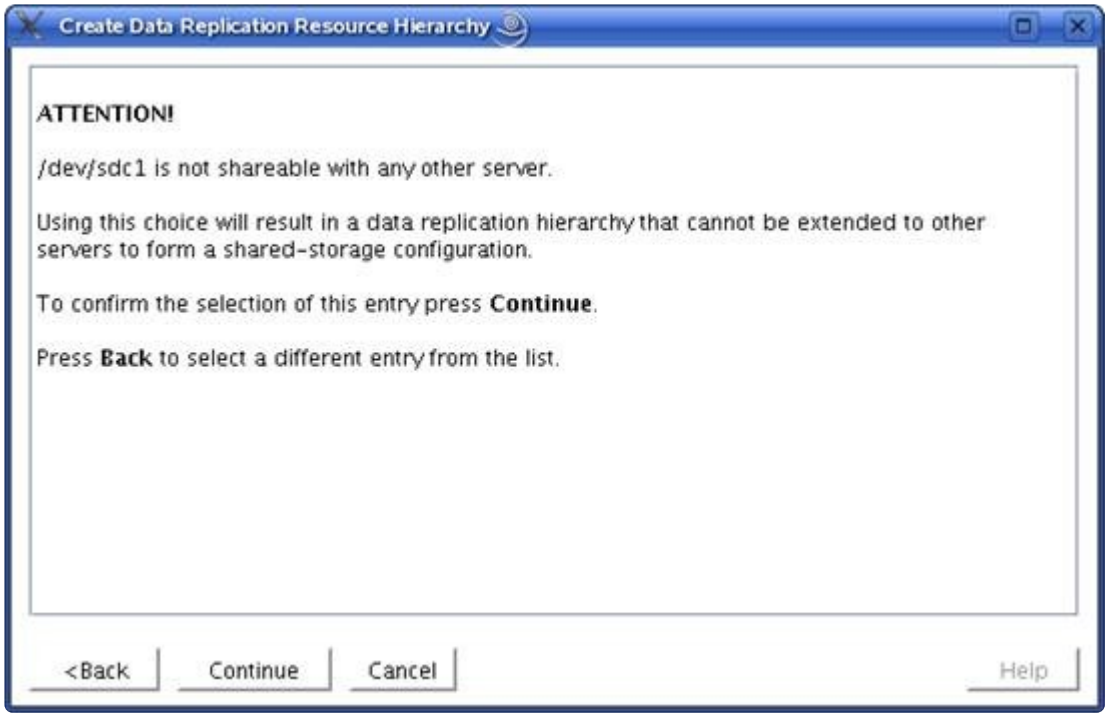
This option will create a NetRAID device, format it with a LifeKeeper supported file system type, mount the file system on the NetRAID device and place both the mounted file system and the NetRAID device under LifeKeeper protection. The NetRAID device and the local disk or partition will be formatted causing existing data to be deleted. You should select this option if you want to create a mirror on a new file system and place it under LifeKeeper protection. You will need one free disk or partition for this resource type.

> ! **CAUTION:** This option will cause your local disk or partition to be formatted and all existing data will be deleted.

1. Enter the following information when prompted:

Field	Tip
Source Disk or Partition	The list of Source Disks or Partitions in the drop-down list contains all the available disks or partitions that are not: • currently mounted • swap disks or partitions • LifeKeeper-protected disks or partitions The drop-down list will also filter out special disks or partitions, for example, root (*/*), boot (*/boot*),_ /proc_, floppy and cdrom. **Note:** The source disk or partition must be uniquely identifiable. Starting in v9.5.0, LifeKeeper will no longer allow the device name to be used to identify a device.

2. The following screen will display if you select a source disk or partition that is not shared.



3. Select **Back** to select different source disk or partition that is shared. Provide the remaining information to finish configuring the resource.

Field	Tips
New Mount Point	Enter the **New Mount Point** of the new file system. This should be the mount point where the replicated disk or partition will be located.
New File System Type	Select the **File System Type**. You may only choose from the LifeKeeper supported file system types.
DataKeeper Resource Tag	Select or enter a unique DataKeeper Resource Tag name for the DataKeeper resource instance.
File System Resource Tag	Select or enter the File System Resource Tag name for the file system resource instance.
Bitmap File	Select the bitmap file entry from the pull down list. Displayed in the list are all available shared file systems that can be used to hold the bitmap file. The bitmap file must be placed on a shared device that can switch between the local nodes in the cluster. **Important:** The bitmap file should not reside on a btrfs filesystem (or any other SPS for Linux unsupported filesystem). Placing data replication bitmap files on a btrfs filesystem (or any other SPS for Linux unsupported filesystem) will result in an “invalid argument” error when LifeKeeper tries to configure the mirror. The default

	location for the bitmap file is under */opt/LifeKeeper*. This default location should be changed if */opt/LifeKeeper* resides on a btrfs filesystem.

4. Click **Next** to continue to the **Confirmation** Screen.

5. A confirmation screen noting the location where the new file system will be created and a warning indicating the pending reformat of the local disk or partition will display. Click **Create** to begin **Resource Creation.**

6. LifeKeeper will verify that you have provided valid data to create your resource on a new file system. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created. Note that the creation of the file system may take several minutes depending upon the disk or partition size.

   Click **Next** to continue.

7. An information box appears announcing the successful creation of your new replicated file system resource hierarchy. You must **Extend** the hierarchy to another server in your cluster to begin data replication and in order to place it under LifeKeeper protection.

   Click **Next** to extend the resource or click **Cancel** if you wish to extend your resource at another time.

   If you click **Continue** LifeKeeper will launch the ***Pre-extend Wizard***.

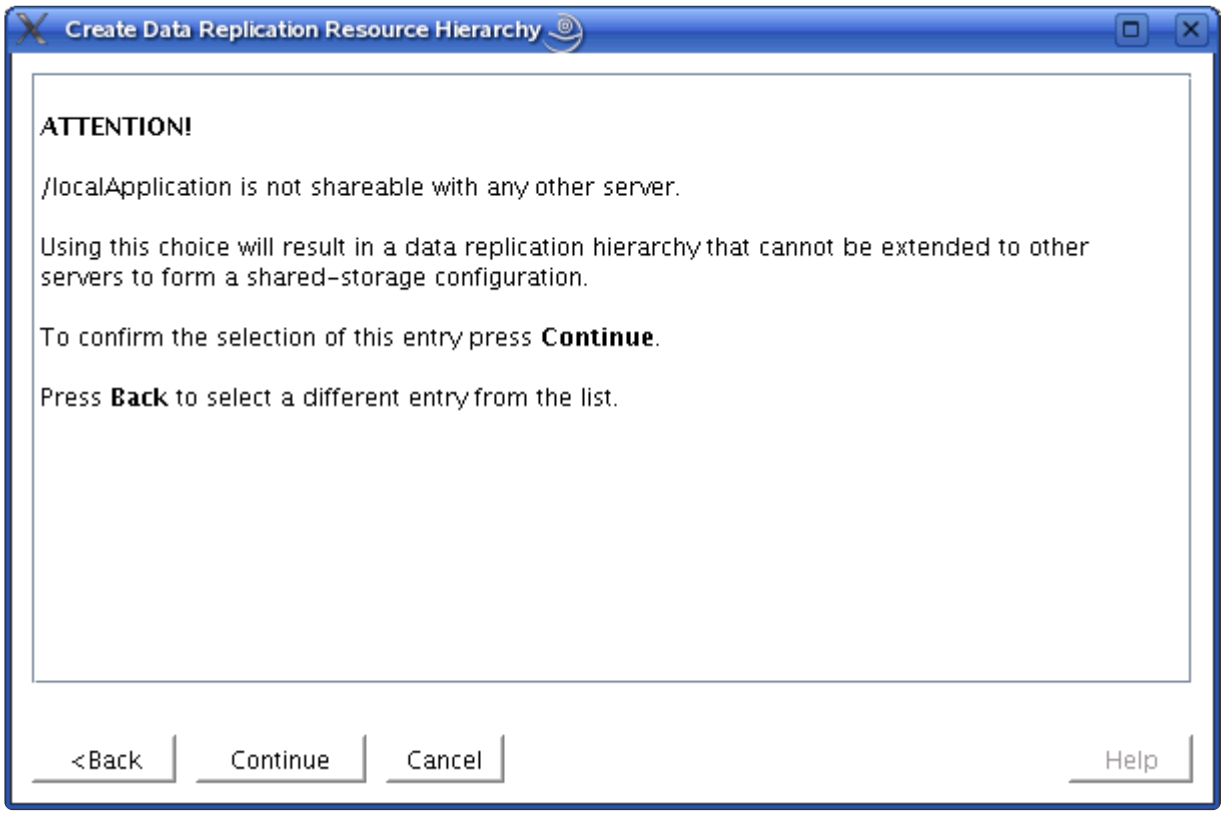
5.4.3.9.1.2. Replicate Existing File System

This option will unmount a currently mounted file system on a local disk or partition, create a NetRAID device, then re-mount the file system on the NetRAID device. Both the NetRAID device and the mounted file system are placed under LifeKeeper protection. You should select this option if you want to create a mirror on an existing file system and place it under LifeKeeper protection.

1. Enter the following information when prompted:

Field	Tip
Existing Mount Point	This should be the mount point for the NetRAID device on the primary server. The local disk or partition should already be mounted at this location. **Note:** The source disk or partition must be uniquely identifiable. Starting in v9.5.0, LifeKeeper will no longer allow the device name to be used to identify a device.

2. The following screen will display if you select a mount point that is not shared.



3. Select **Back** to select a shared mount point. Provide the remaining information to finish configuring the resource.

Field	Tips
DataKeeper	Select or enter a unique **DataKeeper Resource Tag** name for the DataKeeper

Resource Tag	resource instance.
File System Resource Tag	Select or enter the **File System Resource Tag name**.
Bitmap File	Select the bitmap file entry from the pull down list. Displayed in the list are all available shared file systems that can be used to hold the bitmap file. The bitmap file must be placed on a shared device that can switch between the local nodes in the cluster. **Important:** The bitmap file should not reside on a btrfs filesystem (or any other SPS for Linux unsupported filesystem). Placing data replication bitmap files on a btrfs filesystem (or any other SPS for Linux unsupported filesystem) will result in an “invalid argument” error when LifeKeeper tries to configure the mirror. The default location for the bitmap file is under /opt/LifeKeeper. This default location should be changed if /opt/LifeKeeper resides on a btrfs file system. **Important:** Do not select the shared disk area used for replication if displayed in the pull-down selection as the storage area for bitmaps. The shared file system allocated for replication cannot be used as the storage destination of bitmap files. You must use a shared file system location that has been allocated for just bitmap file.

4. Click **Next** to create your DataKeeper resource on the primary server.

5. LifeKeeper will verify that you have provided valid data to create your DataKeeper resource. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created.

   Click **Next**.

6. An information box appears announcing that you have successfully created an existing replicated file system resource hierarchy. You must **Extend** the hierarchy to another server in your cluster to begin replication and to place it under LifeKeeper protection.

   Click **Next** to extend the resource, or click **Cancel** if you wish to extend your resource at another time.

   If you click **Continue** LifeKeeper will launch the *Pre-extend Wizard*. Refer to Step 2 under Extending Your Hierarchy for details on how to extend your resource hierarchy to another server.

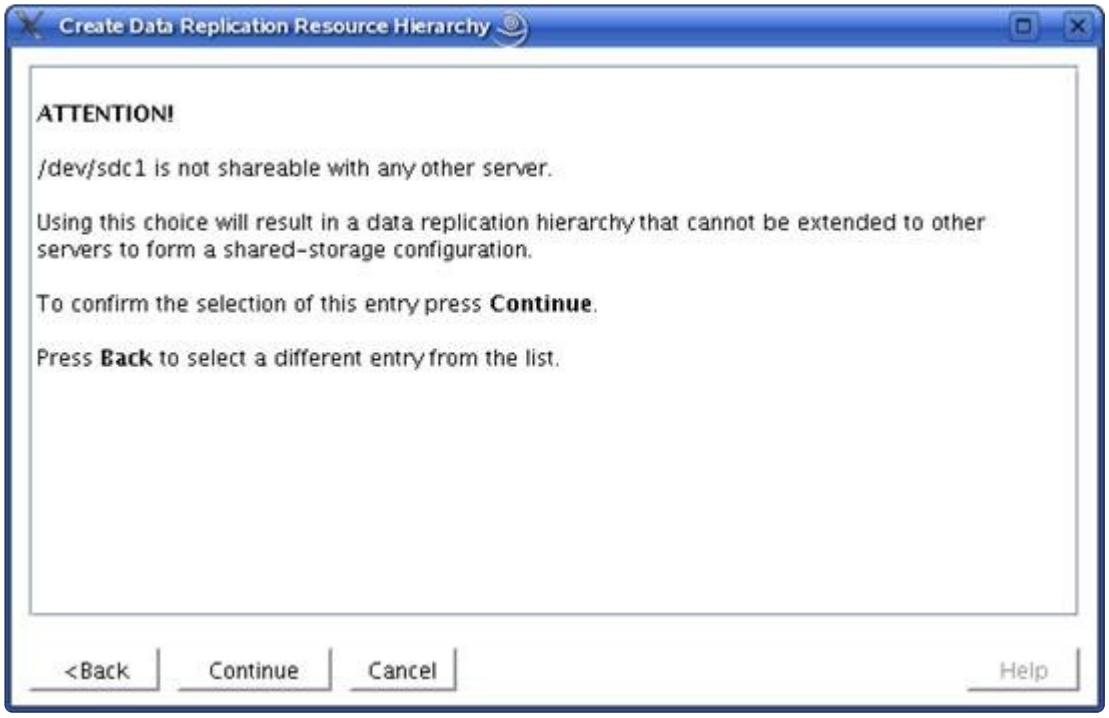
5.4.3.9.1.3. DataKeeper Resource

This option will create only the NetRAID device (not a file system) and place the device under LifeKeeper protection. You should select this option if you only want to create a DataKeeper device on a disk or partition and place the device under LifeKeeper protection. You will need to manually make and mount a file system on this device in order to create a readable mirror. You will need one free disk or partition for this resource type.

1. Enter the following information when prompted:

Field	Tip
Source Disk or Partition	The list of Source Disks or Partitions in the drop-down list contains all the available disks or partitions that are not: º currently mounted º swap disks or partitions º LifeKeeper-protected disks or partitions The drop-down list will also filter out special disks or partitions, for example, root (*/*), boot (*/boot*), */proc*, floppy and cdrom. **Note:** The source disk or partition must be uniquely identifiable. Starting in v9.5.0, LifeKeeper will no longer allow the device name to be used to identify a device.

2. The following screen will display if you select a source disk or partition that is not shared.



3. Select **Back** to select a different source disk or partition that is shared. Provide the remaining information to finish configuring the resource.

Field	Tips
DataKeeper Resource Tag	Select or enter a unique **DataKeeper Resource Tag name** for the DataKeeper resource instance.
Bitmap File	Select the bitmap file entry from the pull down list. Displayed in the list are all available shared file systems that can be used to hold the bitmap file. The bitmap file must be placed on a shared device that can switch between the local nodes in the cluster. **Important:** The bitmap file should not reside on a btrfs filesystem (or any other SPS for Linux unsupported filesystem). Placing data replication bitmap files on a btrfs filesystem (or any other SPS for Linux unsupported filesystem) will result in an “invalid argument” error when LifeKeeper tries to configure the mirror. The default location for the bitmap file is under /opt/LifeKeeper. This default location should be changed if /opt/LifeKeeper resides on a btrfs file system.

4. Click **Next**.

5. An information window appears notifying you that you will have to manually make the file system and mount the NetRAID device (*/dev/mdX*) before being able to use it.

   Click **Create** to create your DataKeeper device on the local disk or partition.

6. An information box appears and LifeKeeper will verify that you have provided valid data to create your DataKeeper resource. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created.

   Click **Next** to continue.

7. An information box appears announcing the successful creation of your DataKeeper resource device. You must **Extend** the hierarchy to another server in your cluster to begin data replication and in order to place it on the backup/target server and under LifeKeeper protection.

   Click **Continue** to extend the resource, or click **Cancel** if you wish to extend your resource at another time.

   If you click **Continue** LifeKeeper will launch the ***Pre-extend Wizard***.

5.4.3.9.2. Extending Your DataKeeper Hierarchy

This operation should be started on the Primary Server to the Secondary Server from the **Edit** menu or initiated automatically upon completing the **Create Resource Hierarchy** option in which case you should refer to Step 2 below.

1. On the **Edit menu**, select **Resource** then **Extend Resource Hierarchy**. The ***Pre-Extend Wizard*** appears. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the LifeKeeper **Extend Resource Hierarchy** defaults and want to bypass the prompts for input/ confirmation, click **Accept Defaults.**

2. The ***Pre-Extend Wizard*** will prompt you to enter the following information.

Note: The first two fields appear only if you initiated the Extend from the **Edit** menu.

Field	Tips
Template Server	Select the **Template Server** *where your DataKeeper resource hierarchy is currently in service*. It is important to remember that the **Template Server** you select now and the **Tag to Extend** that you select in the next dialog box represent an in-service (activated) resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you have selected. The drop down box in this dialog provides the names of all the servers in your cluster.
Tag to Extend	This is the name of the DataKeeper instance you wish to extend from the template server to the target server. The drop down box will list all the resources that you have created on the template server.
Target Server	Enter or select the server you are extending *to*.
Switchback Type	You must select **intelligent switchback.** This means that after a failover to the backup server, an administrator must manually switch the DataKeeper resource back to the primary server. **CAUTION**: This release of SIOS DataKeeper does not support Automatic Switchback for DataKeeper resources. Additionally, the Automatic Switchback restriction is applicable for any other LifeKeeper resource sitting on top of a DataKeeper resource.
Template Priority	Select or enter a **Template Priority**. This is the priority for the DataKeeper hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note**: This selection will appear only for the initial extend of the hierarchy.
Target Priority	Select or enter the **Target Priority**. This is the priority for the new extended DataKeeper hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading

	failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number “1” to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

After receiving the message that the pre-extend checks were successful, click **Next**.

Depending upon the hierarchy being extended, LifeKeeper will display a series of information boxes showing the Resource Tags to be extended, some of which cannot be edited.

3. Click **Next** to launch the **Extend Resource Hierarchy** configuration task.

4. The next section lists the steps required to complete the extension of a DataKeeper resource to another server.

Extending a DataKeeper Resource

1. After you have been notified that your pre-extend script has executed successfully, you will be prompted for the following information:

Field	Tips
Mount Point	Enter the name of the file system mount point on the target server. (This dialog will not appear if there is no LifeKeeper-protected filesystem associated with the DataKeeper Resource.)
Root Tag	Select or enter the **Root Tag**. This is a unique name for the filesystem resource instance on the target server. (This dialog will not appear if there is no LifeKeeper-protected filesystem associated with the DataKeeper Resource.)
Target Disk or Partition	Select the disk or partition where the replicated file system will be located on the target server. The list of disks or partitions in the drop down box contains all the available disks or partitions that are not: • already mounted • swap disks or partitions • LifeKeeper-protected disks or partitions The drop down list will also filter out special disks or partitions, for example, root (*/*), boot (*/boot*), */proc*, floppy and cdrom. **Note:** The size of the target disk or partition must be greater than or equal to that of the source disk or partition.
DataKeeper Resource Tag	Select or enter the **DataKeeper Resource Tag name**.

Bitmap File	Select the name of the bitmap file used for intent logging. If you choose **None**, then an intent log will not be used and every resynchronization will be a full resync instead of a partial resync.
Replication Path	Select the pair of local and remote IP addresses to use for replication between the target server and the other indicated server in the cluster. The valid paths and their associated IP addresses are derived from the set of LifeKeeper communication paths that have been defined for this same pair of servers. Due to the nature of DataKeeper, it is strongly recommended that you use a private (dedicated) network. If the DataKeeper Resource has previously been extended to one or more target servers, the extension to an additional server will loop through each of the pairings of the new target server with existing servers, prompting for a Replication Path for each pair.
Replication Type	Choose “synchronous” or “asynchronous” to indicate the type of replication that should be used between the indicated pair of servers. As for the previous Replication Path field, if the DataKeeper Resource has previously been extended to one or more target servers, the extension to an additional server will loop through each of the pairings of the new target server with existing servers, prompting for a Replication Type for each pair.

2. Click **Next** to continue. An information box will appear verifying that the extension is being performed.

3. Click **Finish** to confirm the successful extension of your DataKeeper resource instance.

4. Click **Done** to exit the **Extend Resources Hierarchy** menu selection.

Note: Be sure to test the functionality of the new instance on all servers by performing a manual switchover. See Testing Your Resource Hierarchy for details.

During resynchronization, the DataKeeper resource and any resource that depends on it will not be able to fail over. This is to avoid data corruption.

5.4.3.9.3. Unextending Your DataKeeper Hierarchy

To remove a resource hierarchy from a single server in the LifeKeeper cluster, do the following:

1. On the **Edit** menu, select **Resource** then **Unextend Resource Hierarchy.**

2. Select the **Target Server** where you want to unextend the DataKeeper resource. It cannot be the server where the DataKeeper resource is currently in service (active).

   Note: If you selected the **Unextend** task by right-clicking from the right pane on an individual resource instance, this dialog box will not appear.

   Click **Next**.

3. Select the **DataKeeper Hierarchy to Unextend** and click **Next**. (This dialog will not appear if you selected the **Unextend** task by right-clicking on a resource instance in either pane).

4. An information box appears confirming the target server and the DataKeeper resource hierarchy you have chosen to unextend. Click **Unextend.**

5. Another information box appears confirming that the DataKeeper resource was unextended successfully. Click **Done** to exit the **Unextend Resource Hierarchy** menu selection.

   Note: At this point, data is not being replicated to the backup server.

5.4.3.9.4. Deleting a DataKeeper Resource Hierarchy

To delete a DataKeeper resource from all servers in your LifeKeeper configuration, complete the following steps.

> **Note:** It is recommended that you take the DataKeeper resource out of service BEFORE deleting it. Otherwise, the **md** and **NetRAID** devices will not be removed, and you will have to unmount the file system manually. See Taking a DataKeeper Resource Out of Service.

1. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.
2. Select the name of the **TargetServer** where you will be deleting your DataKeeper resource hierarchy.
   Note: If you selected the Delete Resource task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.
3. Select the **Hierarchy to Delete.** (This dialog will not appear if you selected the **Delete Resource** task by right-clicking on a resource instance in the left or right pane.) Click **Next.**
4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete.**
5. Another information box appears confirming that the DataKeeper resource was deleted successfully. Click **Done** to exit.

> **Note:** If the NetRAID device was mounted prior to the resource deletion then it will remain mounted. Otherwise, the NetRAID device will also be deleted.

5.4.3.9.5. Taking a DataKeeper Resource Out of Service

Taking a DataKeeper resource out of service removes LifeKeeper protection for the resource. It breaks the mirror, unmounts the file system (if applicable), stops the **md** device and kills the **nbd** server and client.

> **WARNING:** Do not take your DataKeeper resource out of service unless you wish to stop mirroring your data and remove LifeKeeper protection. Use the **Pause** operation to temporarily stop mirroring.

1. In the right pane of the LifeKeeper GUI, right-click on the **DataKeeper resource** that is in service.
2. Click **Out of Service** from the resource popup menu.
3. A dialog box confirms the selected resource to be taken out of service. Any resource dependencies associated with the action are noted in the dialog. Click **Next.**
4. An information box appears showing the results of the resource being taken out of service. Click **Done.**

5.4.3.9.6. Bringing a DataKeeper Resource In Service

Bringing a DataKeeper resource in service is similar to creating the resource: LifeKeeper starts the **nbd** server and client, starts the **md** device which synchronizes the data between the source and target devices, and mounts the file system (if applicable).

1. Right-click on the **DataKeeper resource instance** from the right pane.
2. Click **In Service** from the popup menu. A dialog box appears confirming the server and resource that you have selected to bring into service. Click **In Service** to bring the resource into service.
3. An information box shows the results of the resource being brought into service. Any resource dependencies associated with the action are noted in the confirmation dialog. Click **Done.**

5.4.3.9.7. Testing Your DataKeeper Resource Hierarchy

You can test your DataKeeper resource hierarchy by initiating a manual switchover. This will simulate a failover of the resource instance from the primary server to the backup server.

Performing a Manual Switchover from the LifeKeeper GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, **Resource**, and **InService**. For example, an in-service request executed on a backup server causes the DataKeeper resource hierarchy to be taken out-of-service on the primary server and placed in-service on the backup server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

The state of the DataKeeper resource on the new primary server is set to “Source” in the LifeKeeper GUI. During the switchover, the state of the DataKeeper resource on each target is initially set to “Out of Sync” to show that data is not replicating to that target yet. For a multi-target configuration the previous source will show “Out of Sync’ and other targets will show “Out of Sync (Wait for Previous Source)”. Resynchronization will automatically begin (the state will transition to “Resyncing”) on each target starting with the previous source. Once resynchronization is complete, the state will change to “Target”, which is the normal **Standby** condition. These state transitions will often occur quickly, so they may not be seen in the GUI.

Note: Manual failover is prevented for DataKeeper resources during resynchronization.

If you execute the **Out of Service** request, the resource hierarchy is taken out of service without bringing it in service on the other server. The resource can only be brought in service on the same server if it was taken out of service during resynchronization.

5.4.4. Administering SIOS DataKeeper for Linux

The following topics provide information to help in understanding and managing SIOS DataKeeper for Linux operations and issues after DataKeeper resources are created.

5.4.4.1. Viewing Mirror Status

You can view the **Replication Status** dialog to see the following information about your mirror:

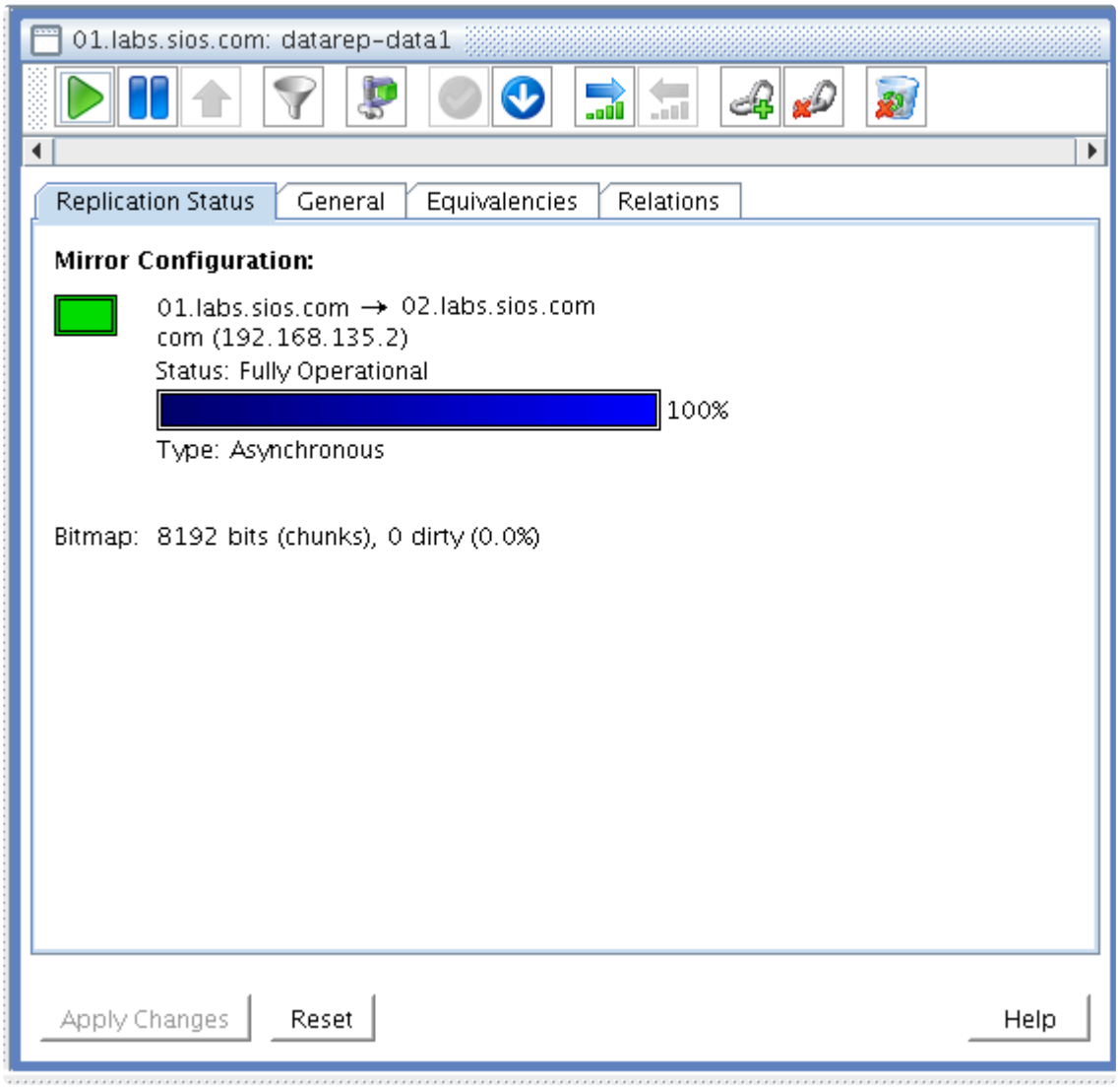
- **Mirror status:** Fully Operational, Paused, Resyncing, or Out Of Sync
- **Synchronization status:** percent complete
- **Replication type:** synchronous or asynchronous
- **Replication direction:** from source server to target server
- **Bitmap:** the state of the bitmap/intent log
- **Network Compression Level:** the compression level (if enabled)

To view the **Replication Status** dialog, do the following:

1. Click the **View** menu, and select **Properties Panel**.
2. Click the **DataKeeper resource** in the **LifeKeeper status** display.

OR

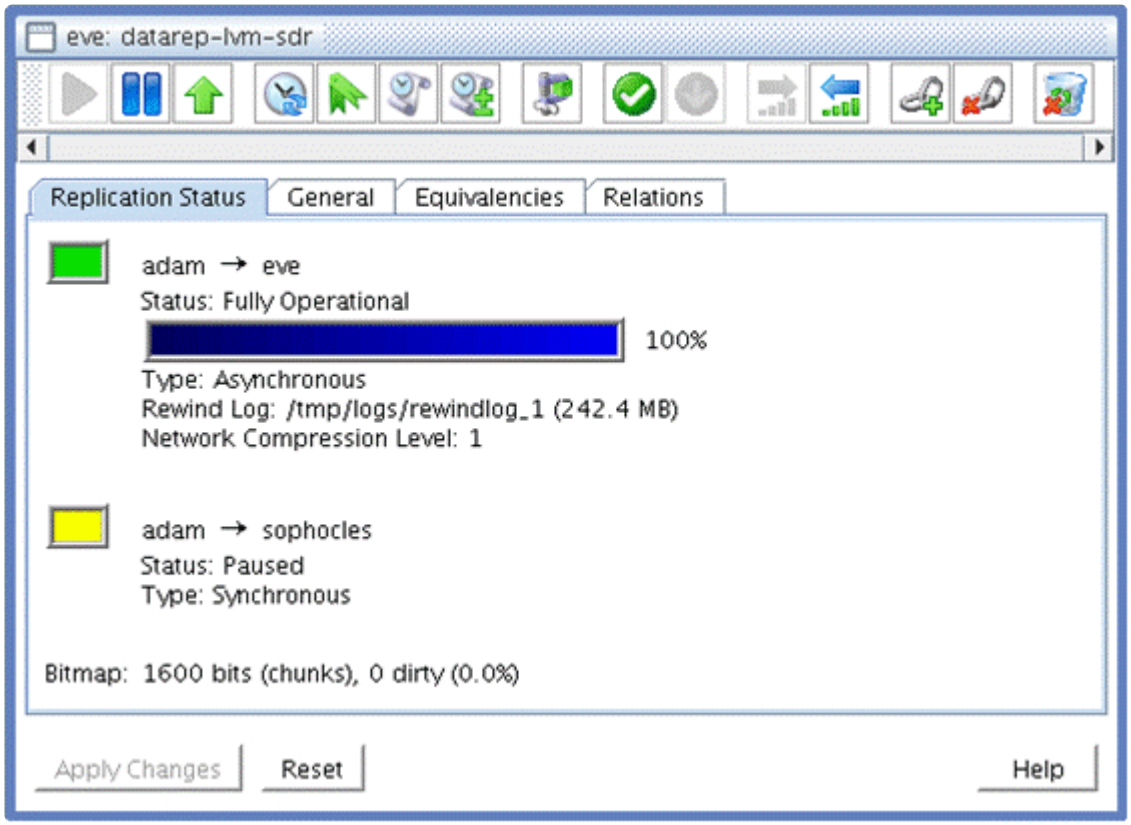
1. Right-click the **DataKeeper resource** in the LifeKeeper status display.
2. From the pop-up menu, select **Properties**.


01.labs.sios.com: datarep-data1
Replication Status
General
Equivalencies
Relations
Mirror Configuration:
01.labs.sios.com → 02.labs.sios.com
com (192.168.135.2)
Status: Fully Operational
100%
Type: Asynchronous
Bitmap: 8192 bits (chunks), 0 dirty (0.0%)
Apply Changes
Reset
Help

5.4.4.2. GUI Mirror Administration

A SIOS DataKeeper mirror can be administered through the LifeKeeper GUI in two ways:

1. By enabling the **Properties Panel** and clicking the toolbar icons (shown in the screenshot).



   Click on each icon below for a description.

   OR

2. By right-clicking the **data replication resource** and selecting an action from the popup menu.

Force Mirror Online

Force Mirror Online should be used only in the event that both servers have become inoperable and the primary server cannot bring the resource in service after rebooting.

Selecting **Force Mirror Online** removes the *data_corrupt* flag and brings the DataKeeper resource in service. For more information, see Primary server cannot bring the resource ISP in the Troubleshooting section.

> **Note:** `mirror_settings` should be run on the target system(s) (or on all systems, if you want the settings to take effect regardless of which system becomes the mirror source). The mirror must be **paused** and **restarted** before any settings changes will take effect.

5.4.4.2.1. Pause and Resume

Pause Mirror

Resume Mirror

You may pause a mirror to temporarily stop all writes from being replicated to the target disk. For example, you might pause the mirror to take a snapshot of the target disk or to increase I/O performance on the source system during peak traffic times.

When the mirror is paused, it will be mounted for read (or read/write with kernel 2.6.19 or higher) access at the normal filesystem mount point on the target system. Any data written to the target while the mirror is paused will be overwritten when the mirror is resumed.

5.4.4.2.2. Set Compression Level

The Network Compression Level may be set to a value from 0 to 9. A value of 0 disables compression entirely. Level 1 is the fastest but least aggressive compression level, while Level 9 is the slowest but best. Network compression is typically effective only on WANs.

5.4.4.3. Command Line Mirror Administration

In addition to performing actions through the LifeKeeper GUI, the mirror can also be administered using the command line. There are several commands (found in the *$LKROOT/bin* directory) that can be used to administer a DataKeeper resource.

Mirror Actions

```
mirror_action <tag> <action> [source] [target(s)]
```

<tag> is the LifeKeeper resource tag of the DataKeeper resource

<action> is one of: `pause, resume, force, fullresync`

[source] (*optional*) is the current source system (if source is not specified, it will use the current system the command was run from)

[target] (*optional*) is the target system (or list of systems) that the action should affect (if target(s) is not specified, it will use all of the applicable target(s))

Note: When using the `force` action, `source` argument is required to specify source node and `target(s)` argument is unnecessary. When using the `pause, resume` or `fullresync` action, if specifying `target(s)` argument, `source` argument is also required.

Examples:

To pause the mirror named *datarep-ext3*:

```
mirror_action datarep-ext3 pause
```

To resume replication from *adam* to both *eve* and *sophocles*:

```
mirror_action datarep-ext3 resume adam eve sophocles
```

To force the mirror online on system *eve*:

```
mirror_action datarep-ext3 force eve
```

To resume replication and force a full resynchronization from *adam* to *sophocles*:

```
mirror_action datarep-ext3 fullresync adam sophocles
```

Mirror Resize

The `mirror_resize` command performs a DataKeeper mirror resize without having to delete and recreate the resource. It should be run on the source system. The underlying devices should be resized before resizing the mirror. The size of the underlying devices will be auto-detected and used as the new mirror size. Optionally the mirror size can be specified.

With LifeKeeper v9.5.0 or later, the mirror can be resized even when the resource is in service. However, when reducing the mirror size, the resource must be out of service.

Note: Some resources do not support reducing the mirror size. Make sure that your file system supports reducing the mirror size.

Note: When the command is interrupted due to an error etc., add the "-f" option and execute again. If not completed successfully, data may become inconsistent.

```
mirror_resize [-f] [-s <size>] <tag>
```

`<tag>` is the tag of a mirror resource

`-f` forces the resize without user prompts (not recommended)

`-s <size>` specifies alternate mirror size (in KB) This parameter is required.

Requirements for Mirror Resize

- Underlying devices should be a logical volume (LV)
- Only a configuration with a single target is supported

NOTE: `mirror_resize` is NOT supported in multi-target configurations.

Recommended Steps for Mirror Resize

1. Perform the device resize on the underlying disk on both the source and the target. (Remember that the target size must be greater than or equal to the source size.)
2. Run `mirror_resize` on the source system. This will update the internal metadata and bitmap for the mirror to reflect the newly expanded disk size.

   Example: `mirror_resize -s <size in KB> <tag>`
3. When the resource is out of service, bring only the mirror (i.e., *datarep*) resource in service. A

resync of the newly expanded device will occur.

4. Perform the file system resize on the mirror device (e.g. resize2fs /dev/mdX where X is md device number for the mirror being resized such as /dev/md0).

   Note: An `fsck` may be required before being able to resize the file system.

   Note: Some file systems may be required to be mounted before being resized. Bring the resource in service if it is not mounted.

5. Bring the file system and application resources in service if they are out of service.

NOTE: `mirror_resize` is NOT supported in multi-target configurations.

Recommended Steps for Mirror Resize with an XFS file system:

1. Take the mirror and all dependent resources out of service.
2. Perform the disk resize on the underlying mirror disks. Perform this on both the source and the target. (Remember that the target size must be greater than or equal to the source size.)
3. Run `mirror_resize` on the source system. This will update the internal metadata and bitmap for the mirror to reflect the newly expanded disk size.

   Example: `mirror_resize -s <size in KB> <tag>`

4. Bring the mirror resource and file system in service. A resync of the newly expanded disk or partition will occur.
5. Perform the file system resize on the file system (e.g. xfs_growfs -D size /path/to/file/system).

Bitmap Administration

```
bitmap -a <num>|-c|-d|-x <size_kb>|-X <bitmap_file>
```

`-a <num>` adds the asynchronous write parameter to the bitmap file. It is needed if a synchronous mirror is upgraded to include an asynchronous target. The default value for `<num>` is 256. To calculate the correct value for this limit, see the Asynchronous Mirroring Information in **Mirroring with SIOS DataKeeper for Linux**.

`-c` cleans the bitmap file (*zeroes all the bits*). This can be used to avoid a full resync in case an exact replica of the source disk exists on the target. **Use this option with extreme caution**.

`-d` dirties the bitmap file (*sets all the bits to ones*). This option can be used to force a full

resync, for example after a split-brain situation has occurred.

`-m` reads the bitmap and produces merge stream.

`-X <bitmap file>` examines the bitmap file and displays useful information about the bitmap and the mirror.

`-x <size_kb>` extends bitmap file to be valid with disk of `size_kb`.
(**Note:** This option is only used internally for mirror resizing.)

In addition, the `mdadm` command may also be used to administer a DataKeeper resource, as the DataKeeper resource is actually an md device. Refer to the `mdadm(8)` man page for details. **Note:** When using `mdadm`, be sure to use the version that is located in *$LKROOT/bin*, as it is more up-to-date than the version included with the operating system.

5.4.4.4. Monitoring Mirror Status via Command Line

Normally, the mirror status can be checked using the **Replication Status** tab in the **Resource Properties** dialog of the LifeKeeper GUI. However, you may also monitor the status of your mirror by executing:

```
$LKROOT/bin/mirror_status <tag>
```

Example:

```
# mirror_status datarep-ext3-sdr

[-]    eve -> adam

   Status: Paused

   Type: Asynchronous

[-]    eve -> sophocles

   Status: Resynchronizing

   [=>             ] 11%

   Resync Speed: 1573K/sec

   Type: Synchronous

Bitmap: 4895 bits (chunks), 4895 dirty (100.0%)
```

The following command may also be helpful:

```
   cat /proc/mdstat
```

A sample *mdstat* file is shown below:

```
eve:~ # cat /proc/mdstat

Personalities : [raid1]

md1 : active raid1 nbd10[1] nbd8[3](F) sdb1[0]

   313236 blocks super non-persistent [3/2] [UU_]
```

```
    bitmap: 3/3 pages [12KB], 64KB chunk, file: /opt/LifeKeeper/
    bitmap_ext3-sdr

unused devices: <none/></tag>
```

5.4.4.5. Server Failure

If both your primary and backup servers become inoperable, your DataKeeper resource will be brought into service/activated only when **both** servers are functional again. This is to avoid data corruption that could result from initiating the resynchronization in the wrong direction. If you are certain that the only operable server was the last server on which the resource was “**In Service Protected**” (**ISP**), then you can force it online by right-clicking the DataKeeper resource and then selecting **Force Mirror Online**.

5.4.4.6. Resynchronization

During the resynchronization of a DataKeeper resource, the state of this resource instance on the target server is “**Resyncing**”. However, the resource instance is “**Source**” (**ISP**) on the primary server. The LifeKeeper GUI reflects this status by representing the DataKeeper resource on the target server with the following icon:

and the DataKeeper resource on the primary server with this icon:

As soon as the resynchronization is complete, the resource state on the target becomes “**Target**” and the icon changes to the following:

The following points should be noted about the resynchronization process:

- A SIOS DataKeeper resource and its parent resources cannot fail over to a target that was in the synchronization process when the primary failed.
- If your DataKeeper resource is taken out of service/deactivated during the synchronization of a target server, that resource can only be brought back into service/activated on the same system or on another target that is already in sync (if multiple targets exist), and the resynchronization will continue.
- If your primary server becomes inoperable during the synchronization process, any target server that is in the synchronization process will not be able to bring your DataKeeper resource into service. Once your primary server becomes functional again, a resynchronization of the mirror will continue.

5.4.4.7. Avoiding Full Resynchronizations

When replicating large amounts of data over a WAN link, it is desirable to avoid full resynchronizations which can consume large amounts of network bandwidth and time. With newer kernels, SIOS DataKeeper can avoid almost all full resyncs by using its bitmap technology. However, the initial full resync, which occurs when the mirror is first set up, cannot be avoided when existing data is being replicated. (For brand new data, SIOS DataKeeper does not perform a full resync, so the steps below are not necessary.)

There are a couple of ways to avoid an initial full resync when replicating existing data. Two recommended methods are described below.

Method 1 – Replicating to a 2nd node

The first method consists of taking a raw disk image and shipping it to the target site. This results in minimal downtime as the mirror can be active on the source system while the data is in transit to the target system.

Procedure

1. Create the mirror (selecting Replicate Existing Filesystem), but do not extend the mirror to the target system.

2. Take the mirror out of service.

3. Take an image of the source disk or partition. For this example, the chosen disk or partition is */dev/sda1*:

   ```
   root@source# dd if=/dev/sda1 of=/tmp/sdr_disk.img bs=65536
   ```

   (The block size argument of 65536 is merely for efficiency).

   This will create a file containing the raw disk image of the disk or partition.

   Note that instead of a file, a hard drive or other storage device could have been used.

4. Optional Step – Take a checksum of the source disk or partition:

   ```
   root@source# md5sum /dev/sda1
   ```

5. Optional Step – Compress the disk image file:

   ```
   root@source# gzip /tmp/sdr_disk.img
   ```

6. Clear the bitmap file (Replace the last argument with the path of the bitmap file which you specified when creating resources.):

```
root@source# /opt/LifeKeeper/bin/bitmap -c /opt/LifeKeeper/
bitmap__dr
```

7. Bring the mirror and dependent filesystem and applications (if any), into service. The bitmap file will track any changes made while the data is transferred to the target system.
8. Transfer the disk image to the target system using your preferred transfer method.
9. Optional Step – Uncompress the disk image file on the target system:

```
root@target# gunzip /tmp/sdr_disk.img.gz
```

10. Optional Step – Verify that the checksum of the image file matches the original checksum taken in Step 4:

```
root@target# md5sum /tmp/sdr_disk.img
```

11. Transfer the image to the target disk, for example, */dev/sda2*:

```
root@target# dd if=/tmp/sdr_disk.img of=/dev/sda2 bs=65536
```

12. Set LKDR_NO_FULL_SYNC=1 in /etc/default/LifeKeeper on both systems:

```
root@source# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper

root@target# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
```

13. Extend the mirror to the target. A partial resync will occur.
14. Edit */etc/default/LifeKeeper* to remove the LKDR_NO_FULL_SYNC entry.

Extending to a 3rd node or any additional nodes without doing a full resync

Procedure for copying data from the Source:

These steps assume the mirror has already been created and extended to the 2nd node, aka target1.

1. Set LKDR_NO_FULL_SYNC=1 in */etc/default/LifeKeeper* on each system:

```
root@source# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper

root@target1# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper

root@target2# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
```

2. Extend the mirror to the new target (target2). A partial resync will occur.

3. Pause the mirror to the new target (target2).
4. Take the mirror out of service.
5. Unmount the file system on the paused mirror on target2 by running '`umount <filesystem>`' on target2.
6. Stop the md device running on target2 by running '`mdadm –stop /dev/md#`' on target2, where # is the value reported in */proc/mdstat*.
7. Make a copy of the source disk or partition on the source node. This could be done using dd or tools from the disk vendor or cloud vendor. The copy **must be** a block-for-block identical copy. It cannot be a file level copy.
8. Optional step – Collect checksum data to verify disk image (md5sum, sha256sum, etc).
9. Optional step – Compress disk image.
10. Bring the mirror and dependent filesystem and applications (if any), into service. The bitmap file will track any changes made while the data is transferred to the target system.
11. Verify that the mirror to target2 is still paused. If it is not then restart at step 4.
12. Verify that the file system and md device are not running on target2. If they are then unmount the file system and stop the md device.
13. Transfer the disk image to the target disk on target2.
14. Verify that the disk image is correct. Perhaps use md5sum or sha256sum to validate the disk contents.
15. Resume the paused mirror to target2. The bitmap on the source was keeping track of any changes made since target2 was paused. When the mirror is resumed these changes will be sent to target2.
16. Edit */etc/default/LifeKeeper* to remove the LKDR_NO_FULL_SYNC entry.

Procedure for copying data from paused target:

This will allow no downtime but while the targets are paused, there is no data redundancy.

These steps assume the mirror has already been created and extended to the 2nd node, aka target1.

1. Set LKDR_NO_FULL_SYNC=1 in /etc/default/LifeKeeper on each system:

```
root@source# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
```

```
root@target1# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper

root@target2# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
```

2. Extend the mirror to the new target (target2). A partial resync will occur.
3. Pause the mirror to the new target (target2).
4. Pause the mirror to target1.
5. Unmount the file system on the paused mirror on target2 by running '`umount <filesystem>`' on target2.
6. Stop the md running on target2 by running '`mdadm –stop /dev/md#`' on target2, where # is the value reported in */proc/mdstat*.
7. Unmount the file system on the paused mirror on target1 by running '`umount <filesystem>`' on target1.
8. Stop the md running on target1 by running '`mdadm –stop /dev/md#`' on target1, where # is the value reported in */proc/mdstat*.
9. Make a copy of the target disk or partition on target1. This could be done using dd or tools from the disk vendor or cloud vendor. The copy **must be** a block-for-block identical copy of the full disk or partition. It cannot be a file level copy.
10. Optional step – Collect checksum data to verify disk image (md5sum, sha256sum, etc).
11. Optional step – Compress disk image.
12. Resume replication to target1. The bitmap file will track any changes made while the data is transferred to target2.
13. Verify that the mirror to target2 is paused. If it is not then restart at step 4.
14. Verify that the file system and md device are not running on target2. If they are, then unmount the file system and stop the md device.
15. Transfer the disk image to the target disk on target2.
16. Optional step – Decompress disk image.
17. Optional step – Verify the disk image is correct (md5sum, sha256sum, etc.).
18. Resume the paused mirror to target2. The bitmap on the source was keeping track of any changes made since target2 was paused. When the mirror is resumed these changes will be sent to target2.

19. Edit */etc/default/LifeKeeper* to remove the LKDR_NO_FULL_SYNC entry.

Method 2

This method can be used if the target system can be easily transported to or will already be at the source site when the systems are configured. This method consists of temporarily modifying network routes to make the eventual WAN mirror into a LAN mirror so that the initial full resync can be performed over a faster local network. In the following example, assume the source site is on subnet 10.10.10.0/24 and the target site is on subnet 10.10.20.0/24. By temporarily setting up static routes on the source and target systems, the “WAN” traffic can be made to go directly from one server to another over a local ethernet connection or loopback cable.

Procedure

1. Install and configure the systems at the source site.
2. Add static routes:

```
root@source# route add -net 10.10.20.0/24 dev eth0
```

```
root@target# route add -net 10.10.10.0/24 dev eth0
```

   The systems should now be able to talk to each other over the LAN.

3. Configure the communication paths in LifeKeeper.
4. Create the mirror and extend to the target. A full resync will occur.
5. Pause the mirror. Changes will be tracked in the bitmap file until the mirror is resumed.
6. Delete the static routes:

```
root@source# route del -net 10.10.20.0/24
```

```
root@target# route del -net 10.10.10.0/24
```

7. Shut down the target system and ship it to its permanent location.
8. Boot the target system and ensure network connectivity with the source.
9. Resume Replication. A partial resync will occur.

5.4.5. Using LVM with DataKeeper

SPS for Linux currently supports both the use of DataKeeper “above” LVM and LVM “above” DataKeeper. In a standard DataKeeper configuration, using DataKeeper above LVM is supported and DO NOT install the SPS LVM Recovery Kit. DataKeeper is the only recovery kit necessary. However, using the LVM above DataKeeper configuration, the LVM Recovery Kit is required.

SIOS recommends using DataKeeper above LVM; however, if the LVM above DataKeeper configuration is being used, a two-phase hierarchy creation process must be used. The DataKeeper devices (i.e. hierarchies) must be configured using the DataKeeper “Data Replication Resource” option prior to the creation of the LVM volume groups and logical volumes on the primary server. Once the desired volume groups and logical volumes have been created, the remainder of the hierarchy is created according to the configuration instructions for the recovery kit associated with the application to be protected. The resulting hierarchy will look something like the one shown in Figure 3 below.

> **Note:** For data consistency reasons, in an LVM over DataKeeper configuration, there must either be only one DataKeeper mirror or multiple ***synchronous*** mirrors.

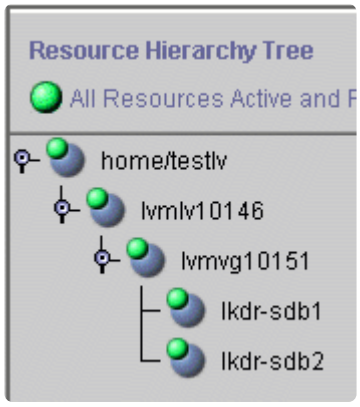
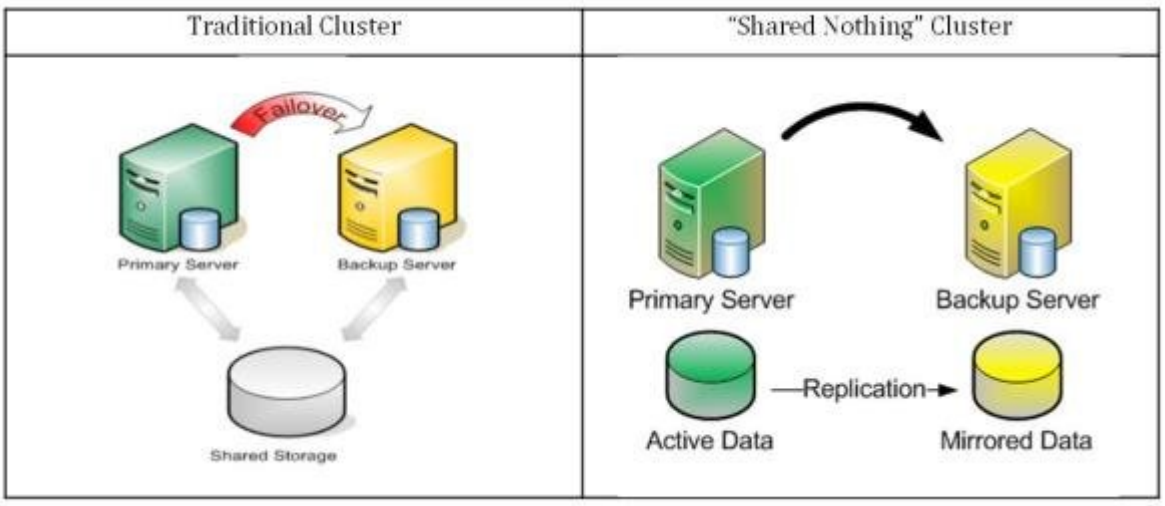


Figure 3: Hierarchy with LVM above DataKeeper

5.4.6. Clustering with Fusion-io

Fusion-io Best Practices for Maximizing DataKeeper Performance

SPS for Linux includes integrated, block level data replication functionality that makes it very easy to set up a cluster when there is no shared storage involved. Using Fusion-io, SPS for Linux allows you to form “shared nothing” clusters for failover protection.



When leveraging data replication as part of a cluster configuration, it is critical that you have enough bandwidth so that data can be replicated across the network just as fast as it is being written to disk. The following best practices will allow you to get the most out of your “shared nothing” SPS cluster configuration when high-speed storage is involved:

Network

- **Use a 10 Gbps NIC:** Flash-based storage devices from Fusion-io (or other similar products from OCZ, LSI, etc.) are capable of writing data at speeds of HUNDREDS (750+) MB/sec or more. A 1 Gbps NIC can only push a theoretical maximum of approximately 125 MB/sec, so anyone taking advantage of an ioDrive’s potential can easily write data much faster than 1 Gbps network connection could replicate it. To ensure that you have sufficient bandwidth between servers to facilitate real-time data replication, a 10 Gbps NIC should always be used to carry replication traffic.

- **Enable Jumbo Frames:** Assuming that your network cards and switches support it, enabling jumbo frames can greatly increase your network’s throughput while at the same time reducing CPU cycles. To enable jumbo frames, perform the following configuration (example on a Red Hat/ CentOS/OEL Linux distribution):

    - º Run the following command:

```
ifconfig <interface_name> mtu 9000
```

- º To ensure change persists across reboots, add “MTU=9000” to the following file:

  /etc/sysconfig/network-scripts/ifcfg-<interface_name>

- º To verify end-to-end jumbo frame operation, run the following command:

  ```
  ping -s 8900 -M do <IP-of-other-server>
  ```

- **Change the NIC’s transmit queue length:**

  - º Run the following command:

    ```
    /sbin/ifconfig <interface_name> txqueuelen 10000
    ```

  - º To preserve the setting across reboots, add to

/etc/rc.local.

- **Change the NIC’s netdev_max_backlog:**

  - º Set the following in

/etc/sysctl.conf:

```
net.core.netdev_max_backlog = 100000
```

TCP/IP Tuning

- **TCP/IP tuning** that has shown to increase replication performance:

  - º Edit

/etc/sysctl.conf and add the following parameters (**Note:** These are examples and may vary according to your environment):

```
net.core.rmem_default = 16777216

net.core.wmem_default = 16777216

net.core.rmem_max = 16777216

net.core.wmem_max = 16777216

net.ipv4.tcp_rmem = 4096 87380 16777216

net.ipv4.tcp_wmem = 4096 65536 16777216

net.ipv4.tcp_timestamps = 0
```

```
net.ipv4.tcp_sack = 0

net.core.optmem_max = 16777216

net.ipv4.tcp_congestion_control=htcp
```

Configuration Recommendations

- Allocate a small (~100 MB) disk partition, located on the Fusion-io drive to place the bitmap file. Create a filesystem on this partition and mount it, for example, at */bitmap*:

```
# mount | grep /bitmap

/dev/fioa1 on /bitmap type ext3 (rw)
```

- Prior to creating your mirror, adjust the following parameters in */etc/default/LifeKeeper*:
    - ◦ `LKDR_CHUNK_SIZE=4096`
        - º Default value is 256
- Create your mirrors and configure the cluster as you normally would.
- The Bitmap file must be set up to be created in the partition, which is created as above.
- Set up for faster resynchronization. Select “Set Resync Speed Limits” from right menu of DataKeeper Resource and set up the following figure to the wizard.
    - º

```
Minimum Resync Speed Limit: 200000
```

- At the same time, set up Resync speed to be allowed during other I/Os operating. This figure must be set up under the half of the maximum wright speed throughput of the drive as the empirical rule not to disturb the normal I/O functions during the Resync operation.
    - º

```
Maximum Resync Speed Limit: 1500000
```

- Set up the maximum bandwidth to use during Resync. This figure must be set up with enough high figure to execute Resync with the available maximum speed.

5.4.7. Using External Snapshot Functions for Disks and Devices Protected by DataKeeper

Full synchronization is required when using a snapshot process to restore data to a disk or device that is actively protected by DataKeeper.

Snapshot capability referred to in this document includes:

1. Snapshots provided by cloud environment services such as AWS
2. Snapshots provided by virtualization software such as vSphere
3. Snapshots provided by shared storage in physical environment

The process of restoring snapshots typically takes place without involvement of the operating system. When the snapshot is restored without involving the OS DataKeeper cannot properly synchronize these changes to the target system. In addition, the changes to the underlying disk or device are not written in the bitmap, so consistency is not maintained with differential synchronization.

The recommended procedure for using snapshots is as follows:

1. Stop the mirror with “pause mirror” command
2. Restore the snapshot
3. Perform a complete re-synchronization with restored data as the source

The source and target data are in an inconsistent state until the mirror is fully synchronized. Therefore, it is essential to execute full synchronization after the snapshot has been restored. Please note a full resync can take a long time depending on the mirror size, available bandwidth, and system resources. During the resync period, the DataKeeper resource and any dependent resources cannot be switched over.

Full synchronization may not be required when restoring the same snapshot to both the source and the target, but this operation is not supported.

5.4.8. DataKeeper for Linux Troubleshooting

This section provides information regarding issues that may be encountered with the use of DataKeeper for Linux. Where appropriate, additional explanation of the cause of an error is provided along with necessary action to resolve the error condition.

Messages specific to DataKeeper for Linux can be found in the DataKeeper Message Catalog. Messages from other SPS components are also possible. In these cases, please refer to the Combined Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

The following table lists possible problems and suggestions.

Symptom	Suggested Action
Warning message that netraid mirror does not have a unique identifier.	The configuration should be modified as soon as possible to use a unique identifier. The recommended steps to repair are: 1. Start with LifeKeeper running on all nodes. 2. Identify unsafe netraid resources and the underlying disks for each resource on each node. It is important to do this on each node as the mapping may be DIFFERENT on each node. # ins_list -r netraid -f: | grep DEVNAME | grep -v mapper | cut -f4,5 -d: datarep-test1:/dev/xvdb datarep-test2:/dev/xvdc datarep-filesys3:/dev/xvdd **NOTE:** This is the list of netraid “tag:ID”. In the following instructions the device name mapping matching above is assumed. 3. Check if device is configured with GPT. For each device run the GPT getId: `#/opt/LifeKeeper/lkadm/subsys/scsi/gpt/bin/getId -i /dev/xvdb` 4757cd62-e065-4013-8514-1031b446aa24 4. If getId returns a unique ID then update the instance:

#ins_setid -t datarep-test1 -i “4757cd62-e065-4013-8514-1031b446aa24”

<If there are any devices that are not GPT then continue with Step 5>

5. Identify resources that depend on the unsafe netraid resources.

```
# ins_list -r netraid -f: | grep DEVNAME | grep -v mapper | cut -f4 -d: | while
read entry; do dep_list -p $entry -f: 2>/dev/null | cut -f1 -d:; done
/test1
/test2
filesys3
```

NOTE: this is the **list of tags** for the file systems associated with the netraid devices.

6. Identify the mount points for the file system. Typically the tag for the file system resource is the same as the mount point. If that is not the case you can match the file system tag with the file system mount point using:

```
# ins_list -t filesys3 -f: | cut -d: -f5
/test3
```

7. Stop all activity leaving only the file system resources in-service on netraid devices.

a. Take all resources out-of-service.
b. Bring in-service only the file systems on unsafe netraid devices.

Follow the steps below only for the devices that are unsafe.

8. Backup all data on affected file systems where the resources are in-service.

9. Take all resources out-of-service on all cluster nodes.

10. Backup the LifeKeeper configuration (lkbackup -c —cluster).

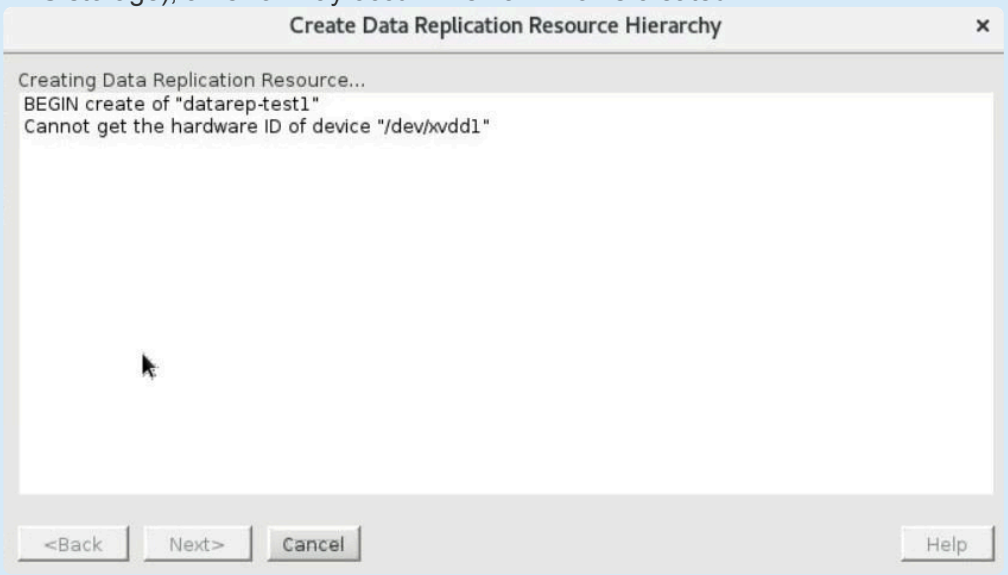
11. Delete the hierarchy with each unsafe netraid resource.

a. Take note of the hierarchy, what application is affected is being deleted.
b. For complex configurations this may require deleting multiple hierarchies where the hierarchy is made up of multiple branches.

12. At this point, the only things left are resources that do not have dependencies with unsafe netraid resources. In most cases that should be the IP resources, EC2 resources, etc.
13. Run lkstop on all nodes.

	14. Verify all affected file systems are unmounted. 15. Reconfigure devices on each node with a GPT partition table (using gdisk, parted, etc) or use LVM. 16. Start LifeKeeper on all nodes. 17. Create new Replicated file systems for each file system, extending each resource to all nodes. • /dev/xvdb1 -> /test1 • /dev/xvdc1 -> /test2 • /dev/xvdd1 -> /test3 18. Restore data from the backup to each mount point. The data will automatically resync to the target(s). 19. Recreate application hierarchies deleted in step 11. Please refer to the SIOS Product Documentation for details on DataKeeper storage configuration options.
After primary server panics, DataKeeper resource goes ISP on the secondary server, but when primary server reboots, the DataKeeper resource becomes OSF on both servers.	Check the “switchback type” selected when creating your DataKeeper resource hierarchy. Automatic switchback is not supported for DataKeeper resources in this release. You can change the Switchback type to “Intelligent” from the resource properties window.
DataKeeper GUI wizard does not list a newly created partition.	The Linux OS may not recognize a newly created partition until the next reboot of the system. View the */proc/partitions* file for an entry of your newly created partition. If your new partition does not appear in the file, you will need to reboot your system.
Errors during failover	Check the status of your device. If resynchronization is in progress you cannot perform a failover.
Error creating a DataKeeper hierarchy on currently mounted NFS file system	You are attempting to create a DataKeeper hierarchy on a file system that is currently exported by NFS. You will need to replicate this file system before you export it.
Extending to a target does not prompt for “Replication Type” to allow setting asynchronous or synchronous.	When the mirror was created, “no” was selected for “Enable Asynchronous Replication.” Delete the mirror and recreate selecting “yes” to “Enable Asynchronous Replication” when prompted.

NetRAID device not deleted after DataKeeper resource deletion.	Deleting a DataKeeper resource will not delete the NetRAID device if the NetRAID device is mounted. You can manually unmount the device and delete it by executing: *mdadm –S <md_device> (cat /proc/mdstat to determine the <md_device>)*.
Primary server cannot bring the resource ISP when it reboots after both servers became inoperable.	If the primary server becomes operable before the secondary server, you can force the DataKeeper resource online by opening the resource properties dialog, clicking the **Replication Status** tab, clicking the **Actions** button, and then selecting **Force Mirror Online**. Click **Continue** to confirm, then **Finish**.
Replication Type is asynchronous instead of synchronous. Replication between two systems was initially configured for asynchronous replication, but synchronous replication is required instead.	Unextend the mirror and extend again, selecting “synchronous” when prompted for the connection.
Replication Type is synchronous instead of asynchronous. Replication between two systems was initially configured for synchronous replication, but asynchronous replication is required instead.	Unextend the mirror and extend again, selecting “asynchronous” when prompted for the connection.
Resources appear green (ISP) on both primary and backup servers.	This is a “split-brain” scenario that can be caused by a temporary communications failure. After communications are resumed, both systems assume they are primary. DataKeeper will not resync the data because it does not know which system was the last primary system. Manual intervention is required. **If not using a bitmap:** You must determine which server was the last backup, then take the resource out of service on that server. DataKeeper will then perform a FULL resync. **If using a bitmap:**

	You must determine which server was the last backup, then take the resource out of service on that server. DataKeeper will then perform a partial resync.
Target(s) are out of sync waiting for the previous source.	Connect the previous source to the cluster. If the previous source can not rejoin the cluster in a timely manner, then targets can be reconnected with a full resync by running the command “$LKROOT/bin/mirror_action fullresync <source> <target>” on the current mirror source.
Core – Language Environment Effects	Some LifeKeeper scripts parse the output of Linux system utilities and rely on certain patterns in order to extract information. When some of these commands run under non-English locales, the expected patterns are altered and LifeKeeper scripts fail to retrieve the needed information. For this reason, the language environment variable LC_MESSAGES has been set to the POSIX “C” locale (LC_MESSAGES=C) in */etc/default/LifeKeeper*. It is not necessary to install Linux with the language set to English (any language variant available with your installation media may be chosen); the setting of LC_MESSAGES in */etc/default/LifeKeeper* will only influence LifeKeeper. If you change the value of LC_MESSAGES in */etc/default/LifeKeeper*, be aware that it may adversely affect the way LifeKeeper operates. The side effects depend on whether or not message catalogs are installed for various languages and utilities and if they produce text output that LifeKeeper does not expect.
GUI – GUI login prompt may not re-appear when reconnecting via a web browser after exiting the GUI	When you exit or disconnect from the GUI applet and then try to reconnect from the same web browser session, the login prompt may not appear. **Workaround:** Close the web browser, re-open the browser and then connect to the server. When using the Firefox browser, close all Firefox windows and re-open.
DataKeeper Create (and Extend) Resource fails	When using DataKeeper in certain environments (e.g., virtualized environments with IDE disk emulation, servers with HP CCISS storage, solid state devices (SSD), or Amazon EBS storage), an error may occur when a mirror is created:  This is because LifeKeeper does not recognize the disk in question and cannot get a unique ID to associate with the device. **Workaround:** Create a GUID partition and assign a unique ID to the partition or use LVM.
The status of the mirror target	The use of an NU device is not recommended for LifeKeeper 9.2.2 or later. A “mirror out of sync” problem occurs in environments where DataKeeper resources are configured

becomes “Out of Sync” after upgrading	with NU devices. When upgrading to LifeKeeper 9.2.2 or later, add the following settings to /etc/default/LifeKeeper if NU devices are used: `LKDR_ALLOW_NU=TRUE` **How to check whether NU devices are used:** Run the lcdstatus command. If a resource instance ID field contains a character string beginning with NU-, then NU devices are used.

5.5. Command Line Interface

SIOS Protection Suite's Command Line Interface can be used as an alternative to the Graphical User Interface. Administrator tasks may be automated by incorporating calls to the CLI in shell scripting.

Document Contents

This guide contains the following topics:

- Commands. Describes CLI commands.
- Shell Script Examples. Provides some examples of CLI used in scripting.

SIOS Protection Suite Documentation

The following is a list of SPS related information available from SIOS Technology Corp.:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- SIOS Protection Suite Installation Guide
- Optional Recovery Kit Documentation

This documentation, along with documentation associated with other SPS Recovery Kits, is available online at:

http://docs.us.sios.com/

Shell Script Examples

Examples Pulled Out of Shell Scripts to Create and Extend Hierarchies.

Also how to create a dependency between two hierarchies.

Needed System Parameters

```
LKROOT=/opt/LifeKeeper

OBJ_DIR=/opt/LifeKeeper/lkadm

LKBIN=/opt/LifeKeeper/bin

ExtendPath=/opt/LifeKeeper/lkadm/bin

PATH=$PATH:$LKBIN
```

Generic ARGS

```
LocalServer=unix121.ha.uk.sbphrd.com

TargetServer=unix122.ha.uk.sbphrd.com

Node2Priority=10
```

The above variables are used for the commands that follow

5.5.1. Commands

Groupings and Basic Descriptions of LifeKeeper bin Commands

The commands will be in: */opt/LifeKeeper/bin*

To place this in your path, execute: `'. /etc/default/LifeKeeper'`

Starting and Stopping LifeKeeper, the GUI, etc.

`lkstart` – Start LifeKeeper core

Options:

None

`lkstop` – Stop LifeKeeper core. The behavior when stopping differs depends on the option.

Options:

None – Stop LifeKeeper core and the protected service. The protected service is switched over to the standby system.

`-r - Leave auto-start on system startup enabled.`

`-n - Stop LifeKeeper core and the protected service. No switchover to the standby system is performed.`

`-f - Stop LifeKeeper core but do not stop the protected resource.`

`lkGUIserver` – Start and stop the LifeKeeper GUI daemon processes

Options:

`start`

`stop`

`restart`

`lkGUIapp` – Starts the LifeKeeper Java application

Options:

None

Monitoring LK and Other Misc. LifeKeeper Commands

`lcdstatus` – Display status of LifeKeeper resources, comm paths, etc.

Options:

```
-d - <node to run command on>

-q - short reports
```

`lcdsync` – Writes LifeKeeper configuration information from memory to disk

Options:

```
-d - <other node to run it on>
```

`lcdrcp` – Transfer files from one LifeKeeper node to another via the comm. path

Options:

```
lcdrcp <file names> {dest:ofile | dest:odir}
```

`lcdremexec` – Execute the given command on the given LifeKeeper node

Options:

```
-d <node to run command on> <command>
```

`lcdrecover` – Checks and sets resource hierarchy instance settings

Options:

see documentation

Bringing a Hierarchy into and out of Service

`perform_action` – Performs a given action on a given resource.

Can be used to switch a given hierarchy to another node.

Options:

```
-a <action name>

-t <tag name>
```

Examples:

`perform_action -a restore -t $LKTag` – bring tier into service

`perform_action -a remove -t $LKTag` – take tier out of service

Checking the LifeKeeper Configuration

lkchkconf : Performs the following checks to verify /etc/default/LifeKeeper settings.

- ◦ Checks that the running system is actually using the current settings found in /etc/default/LifeKeeper.
  If the current setting of LifeKeeper is different from the /etc/default/LifeKeeper setting, an error message is output to inform the user.

- ◦ Checks for any inconsistencies between the resource health check time interval(LKCHECKINTERVAL) and the timeout value of each ARK.
  An error message will be logged if the timeout value of each ARK is longer than the resource health check time interval (LKCHECKINTERVAL).

Options:

None

5.5.1.1. SYS – LifeKeeper Commands Related to the Systems in the LifeKeeper Cluster

`sys_list` – Lists out the systems known to a particular LifeKeeper node

Options:

```
-d <other node to run it on>
```

`sys_create` – Creates knowledge of another system on LifeKeeper node

Options:

```
-s <remote system name>

-d <node to run command on>
```

`sys_remove` – Removes knowledge of another system on a LifeKeeper node

Options:

```
-d <dest>

-s <remote system name to be removed from list>
```

`sys_getstate>` – Lists the state of a given LifeKeeper node on the given LifeKeeper node

Options:

```
-d <node to run command on>

-s <system concerning the state being checked>
```

`sys_setstate` – Sets the state of a given LifeKeeper node on a given LifeKeeper node

Options:

```
-d <node to run command on>

-s <system concerning the state being set>

-S <actual state> {DEAD|ALIVE|UNKNOWN}
```

```
-R <reason for state setting>
```

`sys_getdescr` – Prints some information of why the system went to its current state

Options:

```
-d <node to run command on>

-s <system to get data on>
```

5.5.1.2. NET – Communication Paths Related Commands

`net_create` – Creates a communication path between two SPS nodes

Options:

```
-d <node to run command on>

-s <other system>

-D <device path>

-n <TTY or TCP>

-b <baud rate>

-r <remote IP address>

-l <local IP address>

-p <priority>
```

`net_remove` – Removes a communication path between two SPS nodes

Options:

```
-d <node to run command on>

-s <remote server name to be removed from>

-D <device path>

-r <remote IP address>
```

`net_list` – Lists communication path information on a given SPS node

Option:

```
-d <node to run command on>

-f: <field separator of ':'>

-s <system name>
```

`net_change` – Modify specific information about a given communication path

Options:

```
-d <node to run command on>

-s <server name for data to be modified>

-D <device>
```

`Crelcm` – Create a communication path

```
/opt/LifeKeeper/lkadm/bin/crelcm <node 1> <node 2> <net type> <baud rate> <IP address 1> <IP address 2> <prio>
```

`portio` – Tests the serial connection between two SPS nodes

5.5.1.3. FLAG – Commands Related to Internal LifeKeeper Flags

`flg_create` – Set a given LifeKeeper flag on a given node

Options:

```
-d <node to run command on>

-f <flag name>
```

`flg_remove` – Remove a given LifeKeeper flag on a given node

Options:

```
-d <node to run command on>

-f <flag name>
```

`flg_list` – List all LifeKeeper flags that are set on a given node

Options:

```
-d <node to run command on>
```

5.5.1.4. TYP – LifeKeeper Commands Related to Resource Hierarchy Types

`typ_create` – Creates a given resource type on a given node

Options:

`-d <node to run command on>`

`-a <app type>` (need an app first)

`-r <resource type>`

`typ_remove` – Removes the given resource type from the configuration database set of known resource types on the specified system (or local system if no additional system is specified with the `-d dest` option)

Options:

`-d <node to run command on>`

`-a <application type>`

`-t <resource type>`

`typ_list` – Lists all resource types on a given node

Options:

`-d <node to run command on>`

`-f: <field separator of ':'>`

`-a <app type>`

5.5.1.5. APP – LifeKeeper Commands Related to Resource Applications (Group of Related Types)

`app_create` – Creates a given resource application on a given node

Options:

```
-d <node to run on>

-a <application name>
```

`app_remove` – Removes the given application from the configuration database set of known applications on the specified system (or local system if no additional system is specified with the `-d dest` option)

Options:

```
-d <dest>

-a <application type>
```

`app_list` – Lists all resource applications on a given node

Options:

```
-d <node to run on>
```

5.5.1.6. DEP – LifeKeeper Commands Related to How Resource Applications Relate to Each Other

`dep_create` – Creates a dependency between two resource instances

Options:

`-d <dest>`

`-p <parent tag>`

`-c <child tag>`

`dep_remove` – Removes a dependency between two resource instances

`dep_list` – Lists the dependency relationship between two instances

Options:

see online documentation

`eqv_create` – Creates an equivalency of a given resource between two nodes

Options:

`-d <dest>`

`-t <first tag name`

`-o <second tag name>`

`-S <other system>`

`-e SHARED ?`

`eqv_remove` – Removes an equivalency of a given resource between two nodes

Options:

`-d <dest>`

`-s <system to get info on>`

```
-t <tag name>

-f: <field separator of ':'>
```

`eqv_list` – Lists equivalency relationships between resource instances

Options:

```
d <dest>

-s <system to get info on>

-t <tag name>

-f: <field separator of ':'>
```

`hry_setpri` – Sets the priority of a given node or hierarchy on the node

5.5.1.7. INS – Commands Related to Individual LifeKeeper Hierarchy Instances

`ins_list` – Lists the current information of the given resource hierarchy instance

Options:

```
-d <dest>

-f: <field separator of ':'>

-a / -r / -t / -i specify optional app, type, tag, and id info
```

`ins_setas` – Sets the automatic switchback strategy for a given hierarchy

Options:

```
-d <dest>

-t <tag name>

-s <switchback typ> {INTELLIGENT|AUTOMATIC}
```

`ins_setinit` – Defines how a given resource should initialize when LifeKeeper starts

Options:

```
-d <dest>

-t <tag name>

-I <init state> {AUTORES_ISP|INIT_ISP| INIT_OSU}
```

`ins_setinfo` – Defines an information string for a given resource hierarchy

Options:

```
-d <dest>

-t <tag name>

-v <string of information>
```

`ins_setstate` – Sets the state of a given resource hierarchy on a given node

Options:

```
-d <dest>

-t <tag name>

-S <state to set instance> {ISP|ISU|OSU}

-R <reason for state setting>

-A <recursively set all resources that depend on this one>
```

`ins_gettag` – Lists the tag name of the associated ID

Options:

```
-i <id>
```

5.5.1.7.1. Unextend a Hierarchy

```
/opt/LifeKeeper/lkadm/bin/unextmgr <Node_Name> <Tag_Name>
```

5.5.2. LKCLI (LifeKeeper Command Line Interface)

LKCLI provides functions that can be performed with the LifeKeeper GUI through the command line interface. LifeKeeper provides export/import of communication paths and resource information which the GUI does not provide. Export/import functionality enables duplicating a created system and easy deployment from a testing environment to a production environment.

Supported Environments

LKCLI is supported in the following environments:

- Number of nodes – Only a two-node cluster environment is supported.
- OS – All operating systems supported by LifeKeeper are supported.
- Communication path – Supported only in environments where nodes are connected via TCP/IP.
- Application Recovery Kits – Only environments configured with the Application Recovery Kits in the "Supported ARK List" are supported. If unsupported ARK resources have been created on a node, the environment is not supported. If you want to perform command line operations in an unsupported environment, please consider using the Command Line Interface.

Restrictions

LKCLI can be executed only with root privileges (lkadmin group privileges).

Commands

lkcli license	Install a license key
lkcli start	Starts LifeKeeper
lkcli stop	Stops LifeKeeper
lkcli import	Creates communication paths, resources
lkcli export	Exports communication paths, resources information
lkcli clean	Deletes communication paths, resources
lkcli commpath	Operates on communication paths
lkcli dependency	Operates on dependencies
lkcli resource	Operates resources
lkcli status	Lists resource status
lkcli log	Displays a LifeKeeper log
lkcli server	Configures/operates servers
lkcli mirror	Operates DataKeeper mirroring

Common Options Available for All Commands

Option	Default	Description
[--remote <str>]		Hostname of the machine where you want to run a command. If the option is not specified, the command is executed on the local machine. Before you can execute commands remotely, the machine on which you want to execute the command and the communication path must be bidirectionally connected. **Note:** The `start`,`stop`, `commpath` and `clean` commands are not supported with this option.

Note: The brackets “[ ]” in the options table indicate that you don’t have to use this option.

lkcli license

Registers your LifeKeeper license.

Option	Default	Description
--file <str>		A path to the license file

lkcli start

Starts LifeKeeper

lkcli stop

Stops LifeKeeper

Option	Default	Description
[-f]		Stops only the LifeKeeper daemon and does not stop protected services.
[-r]		Stops LifeKeeper without changing the settings for automatic startup of the LifeKeeper daemon.
[-n]		Performs failover when LifeKeeper stops.This option cannot be used with -f or -r.

lkcli import

Reads the LifeKeeper settings from a file and creates communication paths and resources.

lkcli import commpath

Reads LifeKeeper settings from a file and creates communication paths. In order to connect a communication path bidirectionally, execute this command on both the local machine and the remote machine.

Option	Default	Description

--file <str>		A LifeKeeper configuration file path. Create a communication path from a file (YAML format) where the output was saved with `lkcli export`. The communication path protocol that can be created is TCP/IP (socket).

lkcli import resource

Reads LifeKeeper settings from a file and creates resources.

Option	Default	Description
--file <str>		A LifeKeeper configuration file path. Create a communication path from a file (YAML format) where the output was saved with `lkcli export`. Refer to the ARKs list for files that can be created.

Notes:

- This command fails if the environment including the hostname, IP address or application is not prepared.
- Rollback is not performed even if it fails. Only some resources may be created.
- Execution of the command where resources already exist is not supported.

lkcli export

Exports LifeKeeper settings.
The current LifeKeeper settings are exported. The node where the command is executed and all nodes to which the communication path is connected are targeted.

Save the output in a file in YAML format.

```
# lkcli export > lk_export.yml
```

Notes:

- The communication path protocol that can be export is TCP/IP (socket).
- Refer to the ARKs list for resources that can be exported.
- The resource status (In Service, Out of Service, etc.) is not exported.
- The server property value is not exported.
- Only the configured values on LifeKeeper are exported. The settings of the protected applications are not exported.
- For the exported configuration file, edit only the hostname and IP address manually. Manual changes to other items are not supported.

lkcli clean

Deletes LifeKeeper settings.
Delete the communication path and resource settings of the node where the command was executed. If

you want to delete LifeKeeper configurations on all nodes, execute this command on all nodes. Please note that if the communication path is deleted at this time, commands cannot be executed remotely.

Option	Default	Description
--mode <str>		Specify “all” or “resource”. • **all** – Delete all communication paths and all resources. • **resource** – Delete all resources.

lkcli commpath

Operates communication paths.

lkcli commpath create

Creates a communication path for the node where the command is executed. The created communication path is the path from the local machine to the remote machine. Execute this command on both the local machine and the remote machine to connect a communication path in both directions. The communication path protocol that can be created is TCP/IP (socket).

Option	Default	Description
--laddr <str>		IP address on the local machine to be set for the communication path.
--raddr <str>		IP address on the remote machine to be set for the communication path.
--dest <str>		Hostname of the remote machine to be set for the communication path.
[--priority <str>]	Maximum value of existing paths +1	Priority of the communication path. **Note:** Specify the same value on both the local machine and the remote machine.

lkcli commpath delete

Deletes the communication path of the node where the command was executed.
The communication path to be deleted is the path from the local machine to the remote machine.
Execute this command on both the local machine and the remote machine to delete communication paths in both directions,

Option	Default	Description
--laddr <str>		IP address on the local machine to be set for the communication path.
--raddr <str>		IP address on the remote machine to be set for the communication path.
--dest <str>		Hostname of the remote machine to be set for the communication path.

lkcli dependency

Creates/deletes dependencies for LifeKeeper resources.

lkcli dependency create

Creates a new dependency between two resources.

Option	Default	Description
--parent <str>		The tag name of the parent resource.
--child <str>		The tag name of the child resource.

lkcli dependency delete

Deletes the dependency between the two resources.

Option	Default	Description
--parent <str>		The tag name of the resource that is a parent of the dependency you want to delete.
--child <str>		The tag name of the resource that is a child of the dependency you want to delete.

lkcli resource

Operates on the LifeKeeper resources.

lkcli resource create

Creates resources.

Option	Default	Description
--tag <str>		Tag name of the resource to create. Existing tag names cannot be created. See Resource Tag Name Restrictions for more details.
[--switchback <str>]	INTELLIGENT	Switchback settings for the resource. Specify “INTELLIGENT” or “AUTOMATIC”.
OPTIONS FOR EACH ARK		Options vary by ARK. See Subcommands for Each ARK for options for each ARK.

lkcli resource extend

Extends resources.

> **Note:** Only the target resource is extended even if there are dependencies. An Extend must be done for each resource.

Option	Default	Description
--tag <str>		Tag name of the resource to extend. Tag names that exist on the extension target cannot be extended. See Resource Tag Name Restrictions for more details.
--dest <str>		The hostname of the target server where the resource hierarchy is extended.
[--switchback <str>]	INTELLIGENT	Switchback settings for the resource. Specify “INTELLIGENT” or “AUTOMATIC”.
[--template_priority <num>	1	The priority of the resource hierarchy from which to extend. For the priority, unused values from 1 to 999 are valid, with lower numbers having higher priority.
[--target_priority <num>	10	The priority of the extension target resource hierarchy. For the priority, unused values from 1 to 999 are valid, with lower numbers having higher priority.
OPTIONS FOR EACH ARK		Options vary by ARK See Subcommands for Each ARK for options for each ARK.

lkcli resource config

Changes resource settings.

Option	Default	Description
--tag <str>		The tag name of the resource to change.
OPTIONS FOR EACH ARK		Options vary by ARK See Subcommands for Each ARK for options for each ARK.

lkcli resource unextend

Unextends the resource.

Option	Default	Description
--tag <str>		Tag name of the resource to unextend.
--dest <str>		The hostname of the target server where you want to unextend the resource hierarchy.

lkcli resource delete

Deletes a resource.

Option	Default	Description
--tag <str>		Tag name of the resource to delete.

lkcli resource restore

Brings the resource hierarchy on the active node in service.

Option	Default	Description
--tag <str>		The tag name of the resource to bring in service.

lkcli resource remove

Takes the resource hierarchy on the active node out of service.

Option	Default	Description
--tag <str>		The tag name of the resource to take out of service.

lkcli resource info

Outputs the resource property information. The output differs for each ARK.

Option	Default	Description
--tag <str>		Tag name of the resource for which the property information is output.

lkcli resource eqv

Outputs the equivalency information of the resource.

Option	Default	Description
--tag <str>		Tag name of the resource that for which the equivalency information is output.

lkcli resource reorder-priority

Changes the priority of a resource on an active node.

Option	Default	Description
--tag <str>		The tag name of the resource where the priority is to be changed.
--priority <num>		Priority after the change. For the priority, unused values from 1 to 999 are valid, with lower numbers having higher priority.

lkcli resource switchback

Change the switchback settings of the resource on the active node.

Option	Default	Description
--tag <str>		The tag name of the resource to change.
--switchback <str>	INTELLIGENT	Switchback settings for the resource. Specify “INTELLIGENT” or “AUTOMATIC”.

lkcli status

Displays LifeKeeper status.
See Detailed Status Display for the output status information.

Option	Default	Description
[-q]		Outputs the information of each resource in the local system in a short report.
[-e]		Outputs the information of each resource in the local system in a short report and lists the backup system (with the next highest priority).
[-u]		Suppresses duplicate resource entries in the output of the command with -q or -e options.
[-r <str>]		Specifies the resource root tag. Restrict the report to specific resource root tags.

lkcli log

Displays LifeKeeper logs.

Option	Default	Description
--lines <num>	10	Number of log lines to display.

lkcli server

Performs operations related to the LIfeKeeper server.

lkcli server info

Checks the server shutdown method and failover functionality.

lkcli server shutdown-strategy

Configures the server shutdown method.

Option	Default	Description
--mode <str>		The value of **switchover** or **do_not_switchover**. • **switchover** – LifeKeeper starts the backup server resources with a graceful shutdown. • **do_not_switchover** – LifeKeeper does not start backup server resources upon a graceful shutdown.

lkcli server confirmso

Configures whether to confirm the user for switching to the backup node when a failover occurs due to a node failure on the LifeKeeper cluster.
See Confirm Failover section in Confirm Failover and Block Resource Failover for details.

Option	Default	Description
--server <str>		Name of the failover target server.
--value <str>		**enable** or **disable** • **enable** – Enable user confirmation at failover. • **disable** – Do not confirm the user at failover.

lkcli server block-failover

Configures settings to block failover caused by a resource failure in the specified system.
See the Block Resource Failover section in Confirm Failover and Block Resource Failover for details.

Option	Default	Description
--server <str>		Name of the failover target server.
--value <str>		**enable** or **disable** • **enable** – Block failover to the specified server. • **disable** – Do not block failover to the specified server.

lkcli mirror

Performs mirroring with DataKeeper.
See Mirroring with SIOS DataKeeper for Linux for more information.

> **Note:** `lkcli mirror` command requires DataKeeper to be installed.

Option	Default	Description
--tag <str>		Tag name of the DataKeeper resource.

lkcli mirror status

Displays the status of the mirror.

lkcli mirror resume

Resumes the mirror.

lkcli mirror pause

Pauses the mirror.

lkcli mirror fullresync

Resyncs the entire mirror with a full disk resynchronization.

lkcli mirror force

Forces the mirror to come online, even if LifeKeeper has marked the mirror disk as possibly out of sync.

> **Note:** Forcing a mirror online should be done with great caution, since this can cause data loss.

5.5.2.1. LKCLI Subcommands for Each ARK

ARK List

Apache

See the Configuring LifeKeeper section of the Apache Recovery Kit Administration Guide.

create apache

Option	Default	Description
--root <str>		Full path name (including the file name) of Apache Web Server daemon.
--path <str>		Full path of Apache Web Server root directory. Relative paths and symbolic links cannot be used.

extend apache

No options.

config apache

No options.

DataKeeper

See the Configuring Resources section of SIOS DataKeeper for Linux for more information.

create dk

Option	Default	Description
--mode <str>		Replication type “synchronous” or “asynchronous”.
[--bitmap <str>]	/opt/LifeKeeper/bitmap_<tag name>	Path of bitmap file used for an intent log.
--hierarchy <str>		The type of data replication to create. Options vary depending on the type. < new | existing | dronly >

--hierarchy new

Option	Default	Description
--device <str>		Source disk or partition.
--fstype <str>		File system type. Only file system types supported by LifeKeeper can be specified.
--mount_point <str>		New mount point for new file system.
--fstag <str>		Tag name of the file system resource.

--hierarchy existing

Option	Default	Description
--mount_point <str>		Mount point to mount on the primary server’s NetRAID device.
--fstag <str>		Tag name of the file system resource.

--hierarchy dronly

Option	Default	Description
--device <str>		Source disk or partition.

extend dk

Option	Default	Description
--mode <str>		Replication type. Specify “synchronous” or “asynchronous”.
--laddr		Local IP address.
--raddr		Remote IP address.
[--bitmap <str>]	/opt/LifeKeeper/bitmap_< Tag name >	Path of bitmap file used for an intent log.
[--device <str>]	Same as the extension source.	Source disk or partition.
[--fstag <str>]		Tag name of the file system resource.

config dk

Option	Default	Description

--resync_speed_min <num>		Set the minimum resync speed limit (KB/s).
--resync_speed_max <num>		Set the maximum resynch speed limit (KB/s.)
--compression_level <num>		Set the network compression level (0-9).

EC2

See the Configuration section of the Recovery Kit for EC2 Administration Guide for more details.

create ec2

Option	Default	Description
--type <str>		Specify the type of EC2 resource to be create. Specify “RouteTable” to select a route table scenario, “Elastic IP” to select an Elastic IP scenario.

--type RouteTable

Option	Default	Description
--ip_resource <str>		Specify the tag name of the IP resource created in advance.

--type ElasticIP

Option	Default	Description
--eip <str>		The IP address of the Elastic IP you want to protect.
--dev <str>		Network interface name to which EIP is attached.

extend ec2

No options.

config ec2

No options.

FileSystem

See Creating a File System Resource Hierarchy or Extending a File System Resource Hierarchy for more details.

create fs

Option	Default	Description
--mountpoint <str>		Specify the mount point of the file system.

extend fs

Option	Default	Description
[--mountpoint <str>]	Source system mount point.	Specify the mount point of the file system.

config fs

Option	Default	Description
--mountopt <str>		Specify the mount options for the file system.

Generic Application

See Creating a Generic Application Resource Hierarchy or Extending a Generic Application Resource Hierarchy for more details.

create gen

Option	Default	Description
--restore <str>		Specify the path of the restore script.
--remove <str>		Specify the path of the remove script.
[--quickCheck <str>]		Specify the path of the quickCheck script.
[--recover <str>]		Specify the path of the recover script.
[--appinfo <str>]		Specify optional information about the application.

extend gen

Option	Default	Description
[--appinfo <str>]	Source system appinfo.	Specify optional information about the application.

config gen

Option	Default	Description
[--restore <str>]		Specify the path of the restore script to be updated.
[--remove <str>]		Specify the path of the remove script to be updated.
[--quickCheck <str>]		Specify the path of the quickCheck script to be updated.
[--recover <str>]		Specify the path of the recover script to be updated.
[--all <str>]	No	Specify **Yes** or **No** • **Yes** – Update scripts on all of the cluster nodes. • **No** – Update the script on the node where the command is executed.

IP

See the Configuration section of the IP Recovery Kit Administration Guide for more details.

create ip

Option	Default	Description
--ipaddr <str>		Virtual IP address.
[--netmask <str>]	An appropriate value determined from ipaddr.	Virtual IP netmask.
[--device <str>]	An appropriate value determined from ipaddr and netmask.	Network interface name associated with the virtual IP.

extend ip

Option	Default	Description
[--ipaddr <str>]	Source system ipaddr.	Virtual IP address on the extention destination node.
[--netmask <str>]	An appropriate value determined from ipaddr.	Virtual IP netmask on the extention destination node.
[--device <str>]	An appropriate value determined from ipaddr and netmask.	Network interface name associated with the virtual IP on the extention destination node.

config ip

Option	Default	Description
[--pinglist <str>]		Ping the destination list for options (multiple designations are specified separated by comma).
[--srcaddr <str>]		Specify **0** or **1** Specify whether to use the virtual IP address as the source address for external communication IP traffic to the same subnet. • **0** – Use • **1** – Do not use
[--restoremode <str>]		Specify **Enabled** or **Disabled**. Enable/disable restoration and recovery for IP resources. • **Enabled** – Enable restoration and recovery. • **Disabled** – Disable restoration and recovery.

MySQL

See the Installation section of the MySQL Recovery Kit Administration Guide for more details.

create mysql

Option	Default	Description
--cnf <str>		Absolute path of the MySQL configuration file.
--bin <str>		Absolute path of the directory where the MySQL executable binary is located.
[--instance <str>]	None	MySQL instance number you want to protect. If you are using MySQL on a single instance, do not specify this number.

extend mysql

Option	Default	Description
[--bin <str>]		Absolute path of the directory where the MySQL executable binary is located on the node to which the node is extended.

config mysql

No options.

NFS

See the Configuration section of the NFS Recovery Kit Administration Guide for more details.

create nfs

Option	Default	Description
--export <str>		Export point for the NFS file system.
--ip <str>		Tag name of the IP resource corresponding to the virtual IP address used by the client to access the NFS file system.

extend nfs

No options.

config nfs

No options.

Oracle

See the Configuring LifeKeeper section of the Oracle Recovery Kit for more details.

create oracle

Option	Default	Description

--sid <str>		ORACLE_SID of the database.
[--listener <str>]	None	The tag name of the Oracle Listener resource that is included depending on the Oracle resource.
[--user <str>]	None	Oracle database username.
[--password <str>]	None	Oracle database user password.

extend oracle

No options.

config oracle

Option	Default	Description
--user <str>		Oracle database username.
--password <str>		Oracle database user password.
--role <str>		User role. Specify sysdba or sysoper.

OracleListener

See the Creating a Shared Oracle Listener for Multiple Resources section of the Oracle Recovery Kit Administration Guide for more details.

create listener

Option	Default	Description
--exe <str>		Execution path of the Listener.
--config <str>		Path of the execution setting file of the Listener.
--protection <str>		Protection level of the Listener: • **Full** – Start, stop, monitor and recover • **Intermediate** – Start, monitor and recover • **Minimal** – Only start and monitor
--recovery <str>		Recovery level of the Listener: • **Standard** – Enable the standard LifeKeeper recovery. When all listeners fail locally, perform failover to a valid backup server if necessary. • **Optional** – Enable option LifeKeeper recovery. Even when all listeners fail locally, failover to a valid backup server will not be performed.
[--user <str>]	None	System usename. Specify a system user that has permission to start, stop, monitor and recover the Listener.
[--listener	LISTENER	The name of the Oracle Listener to protect.

<str>]		
[--iptag <str>]	None	The tag name of the IP resource that is protected as a dependency on this resource hierarchy.

extend listener

Option	Default	Description
[--exe <str>]	Source system exe value.	Execution path of the Listener.
[--config <str>]	Source system config value.	Path of the execution setting file of the Listener.

config listener

Option	Default	Description
--type <str>		The item name to change. Options vary depending on the item. • **ProtectionLevel** – Protection level of the Listener. • **RecoveryLevel** – Recovery level of the Listener. • **Listener** – The name of the Oracle Listener to protect.

--type ProtectionLevel

Option	Default	Description
--value <str>		Same as the protection option of create.

--type RecoveryLevel

Option	Default	Description
--value <str>		Same as the recovery option of create.

--type Listener

Option	Default	Description
--value <str>		Name of the Oracle Listener to protect.
[--iptag <str>]	None	The tag name of the IP resource that is protected as a dependency.

PostgreSQL

See the Installation section of the PostgreSQL Recovery Kit for more details.

create pgsql

Option	Default	Description
--datadir <str>		Absolute path of the directory where the database data is located.
--port <num>		Port number used by PostgreSQL.
--socket <str>		The path of the socket used by PostgreSQL.

--dbuser <str>		Username used by PostgreSQL.
--logfile <str>		Absolute path where the logs are output.
[--exepath <str>]	/usr/bin	Absolute path of the directory where the executable is located.
[--clientexe <str>]	<exepath>/psql	Absolute path of the executable “psql”.
[--adminexe <str>]	<exepath>/pg_ctl	Absolute path of the executable “pg_ctl”.

extend pgsql

Option	Default	Description
[--exepath <str>]	Source system exepath.	Absolute path of the directory where the executable file is located on the node to which the resource is extended. If not specified, the setting of the extension origin is inherited.

config pgsql

Option	Default	Description
[--dbuser <str>]	None	Username used by PostgreSQL.

QSP

See the Quick Service Protection (QSP) Recovery Kit for more details.

create qsp

Option	Default	Description
--service <str>		Name of the service to protect.
[--quickCheck <str>]	enable	Enable/disable the monitoring function. Specify “enable” to enable and “disable” to disable the monitoring.

extend qsp

No options.

config qsp

Option	Default	Description
[--service <str>]	None	Enable/disable monitoring. Specify “enable” or “disable”.
[--timeout_restore <num>]	None	Number of seconds for restore timeout. When 0 is specified, timeout does not occur.
[--timeout_remove <num>]	None	Number of seconds for remove timeout. When 0 is specified, timeout does not occur.
[--timeout_quickcheck <num>]	None	Number of seconds for quickCheck timeout. When 0 is specified, timeout does not occur.

[--timeout_recover <num>]	None	Number of seconds for recover timeout.

Route53

See the Configuration section of the Route53 Recovery Kit for more details.

create route53

Option	Default	Description
--domain <str>		Domain name that exists in the Route53 to protect.
--hostname <str>		Name of the host to protect.
--ip_resource <str>		Tag name of the IP resource created in advance.

extend route53

No options.

config route53

No options.

5.5.3. LKCLI Guide

This guide is designed to assist you with configuring your LifeKeeper environment via the command line using LKCLI (LifeKeeper Command Line Interface) .

Please refer to LKCLI (LifeKeeper Command Line Interface) for more details for each LKCLI command.

5.5.3.1. LKCLI – Communication Path Creation and Deletion

What is a communication path?

LifeKeeper configures clusters by connecting multiple nodes to each other. Availability can be maintained by switching to resources on another node in the cluster in the event that a node fails. To achieve this, it is necessary to configure paths for communication between nodes in advance. These paths are called “communication paths” in LifeKeeper.

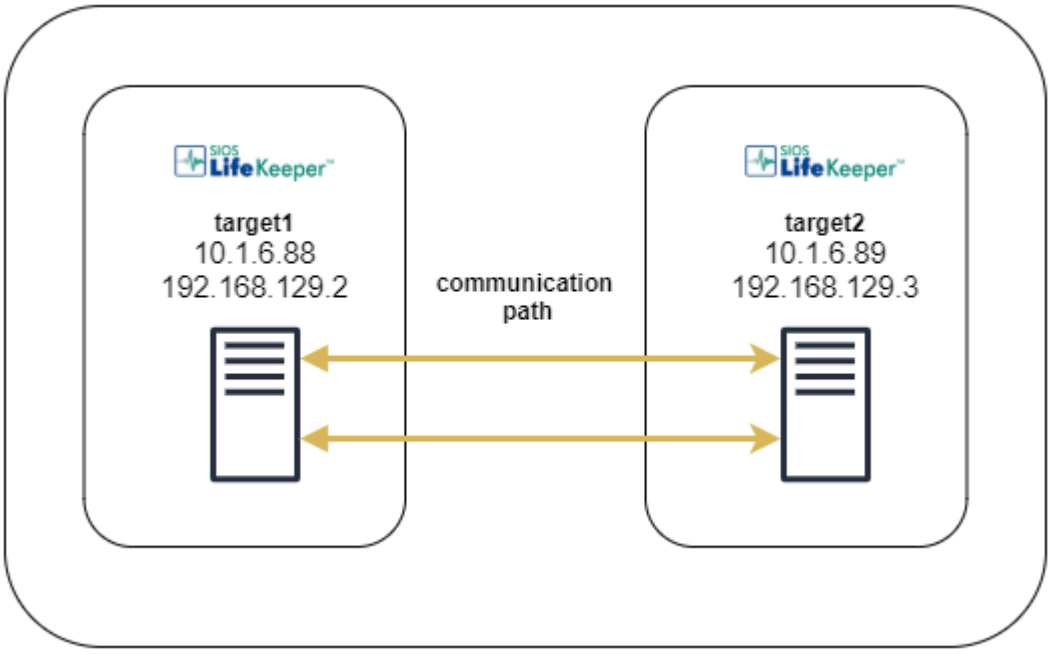
When communication paths are configured, each node sends a signal called a “heartbeat”. The heartbeat indicates that LifeKeeper is working properly. If the heartbeat is not received, the node is considered to have failed. SIOS recommends connecting cluster nodes with two or more communication paths so that in the case of a network failure, the nodes will still be able to receive heartbeats over an alternate path.

Communication paths are used for LifeKeeper internal communication, in addition to heartbeats. A remote LifeKeeper node can be managed over the network as long as communication paths are connected between the nodes.

This guide describes the steps to create and delete a single communication path between two nodes.

Configuration

The commands and other information in this guide are based on the following diagram.



Before you begin, please check that the following conditions are met in your environment:

- TCP Port 7365 is available for the communication path
  - The firewall allows communication or the firewall is disabled
- There are two machines running LifeKeeper

Next, record information about your systems that will be required in order to execute the CLI commands.

1. Available IP address on each system

```
# ip address | grep 'inet'
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
    inet 10.1.6.88/16 brd 10.1.255.255 scope global noprefixroute ens192
    inet6 fe80::a633:2758:4976:b42/64 scope link noprefixroute
    inet 192.168.129.2/24 brd 192.168.129.255 scope global noprefixroute ens22
4
    inet6 fe80::34a4:417f:b58b:dc15/64 scope link noprefixroute
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
```

2. Host name of each system

```
# hostname
target1
```

Creating a LifeKeeper Communication Path

Note: The command for creating a communication path creates the path *in only one direction*. Therefore, in order for the communication path to be able to communicate in both directions, you need to run the command on both systems.

Perform the following steps on target1

1. Make sure that a communication path is not connected.

```
[target1]# lkcli status -q
LOCAL    TAG    ID     STATE      PRIO   PRIMARY
```

2. Create a communication path from target1 to target2.

```
[target1]# lkcli commpath create --laddr 10.1.6.88 --raddr 10.1.6.89 --dest ta
rget2
Performing commpath 'target2:10.1.6.88/10.1.6.89' create...
Commpath 'target2:10.1.6.88/10.1.6.89' created successful.
```

Command Argument

Item	Input Value
--laddr	IP address on the local node to connect from

--raddr	IP address on the remote node to connect to
--dest	Host name of the remote node to connect to

3. Verify that a communication path to target2 has been created. (**Note:** At this point, the STATE is DEAD because only a one-way connection is established.)

```
[target1]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  DEAD         1
```

Perform the following steps on target2

4. Make sure a communication path is not connected.

```
[target2]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY
```

5. Create a communication path from target2 to target1.

```
[target2]# lkcli commpath create --laddr 10.1.6.89 --raddr 10.1.6.88 --dest ta
rget1
Performing commpath 'target1:10.1.6.89/10.1.6.88' create...
Commpath 'target1:10.1.6.89/10.1.6.88' created successful.
```

Command Arguments

Item	Input Value
--laddr	IP address on the local node to connect from
--raddr	IP address on the remote node to connect to
--dest	Host name of the remote node to connect to

6. Verify that the communication path is established. Confirm that the STATE is ALIVE.

*It may take a few seconds for the communication path to become ALIVE.

```
[target2]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

Deleting a LifeKeeper Communication Path

Note: The command for deleting a communication path deletes the path *in only one direction*. Therefore,

in order for the communication path to be deleted in both directions, you need to run the command on both systems.

Perform the following steps on target2

1. Make sure that there is a connected communication path.

```
[target2]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

2. Delete the communication path from target2 to target1.

```
[target2]# lkcli commpath delete --laddr 10.1.6.89 --raddr 10.1.6.88 --dest ta
rget1
```

Command Arguments

Item	Input Value
--laddr	IP address on the local node to connect from
--raddr	IP address on the remote node to connect to
--dest	Host name of the remote node to connect to

3. Confirm that the communication path to target1 has been deleted.

```
[target2]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY
```

Perform the following steps on target1

4. Make sure that a communication path to target2 still exists.

```
[target1]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  DEAD         1
```

5. Delete the communication path to target2.

```
[target1]# lkcli commpath delete --laddr 10.1.6.88 --raddr 10.1.6.89 --dest ta
rget2
```

6. Confirm that the communication path to target2 has been deleted.

```
[target1]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY
```

5.5.3.2. LKCLI – Resource Creation

This topic describes how to create LifeKeeper resources for protected services and applications.

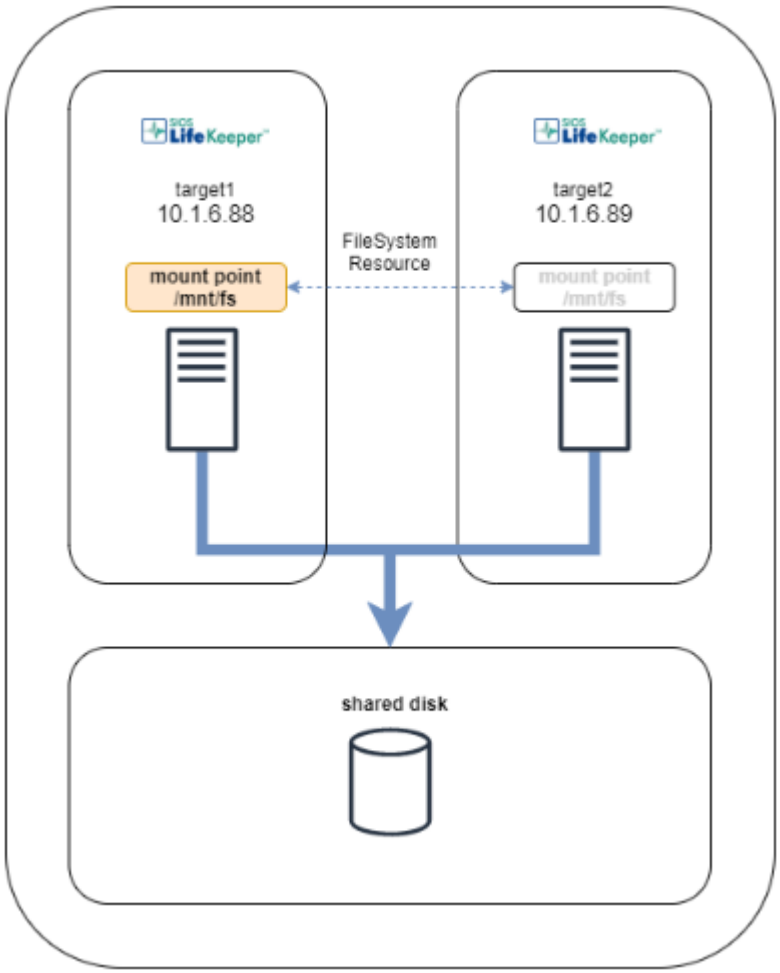
Note: Be sure to create a communication path first (refer to LKCLI Communication Path Creation and Deletion).

Creating File System Resources

The steps for creating file system resources are described below. File system resources provide the capability to switch file systems on shared storage between cluster nodes.

System Configuration

The environment created in this guide is a two-node configuration as shown below.



To create file system resources, the following conditions must be satisfied:

- Shared storage (e.g., iSCSI, Fibre Channel) is physically connected to each node
- A file system has been created using a utility such as mkfs
- The file system can be mounted/unmounted on each node

Perform the following steps on target1

1. Mounting a file system
   Mount a file system for which you want to create a file system resource.
   In the example, */dev/sdb1* is mounted on */mnt/fs*. The mount point is */mnt/fs*.

```
[target1]# df
Filesystem              1K-blocks    Used Available Use% Mounted on
/dev/mapper/centos-root  14034944 6904924   7130020  50% /
devtmpfs                   929204       0    929204   0% /dev
tmpfs                      941312       0    941312   0% /dev/shm
tmpfs                      941312   25948    915364   3% /run
tmpfs                      941312       0    941312   0% /sys/fs/cgroup
/dev/sda1                 1038336  148528    889808  15% /boot
tmpfs                      188264       0    188264   0% /run/user/0
/dev/sdb1                 1044132   32992   1011140   4% /mnt/fs
```

2. Creating a resource
   Run the following command.

```
[target1]# lkcli resource create fs --tag fs-tag --mountpoint /mnt/fs
```

Resource Settings

Item	Input Value
--tag	Tag name
--mountpoint	Mount point

3. Extending a resource
   Run the following command.

```
[target1]# lkcli resource extend fs --tag fs-tag --dest target2
```

Resource Settings

*Item	Input Value
--tag	Tag name of the created resource
--dest	Backup node name

4. Checking the resource
   After creating and extending the resource, run the following command.
   The resource information is displayed.

```
[target1]# lkcli status -q

LOCAL   TAG           ID                                      STATE     PRIO  PRI
MARY
target1 fs-tag        /mnt/fs                                 ISP          1  tar
```

```
get1
target1  device28856  36000c292eb0c693b2efb44ed56556636-1  ISP          1  tar
get1
target1   disk28786   36000c292eb0c693b2efb44ed56556636    ISP          1  tar
get1
```

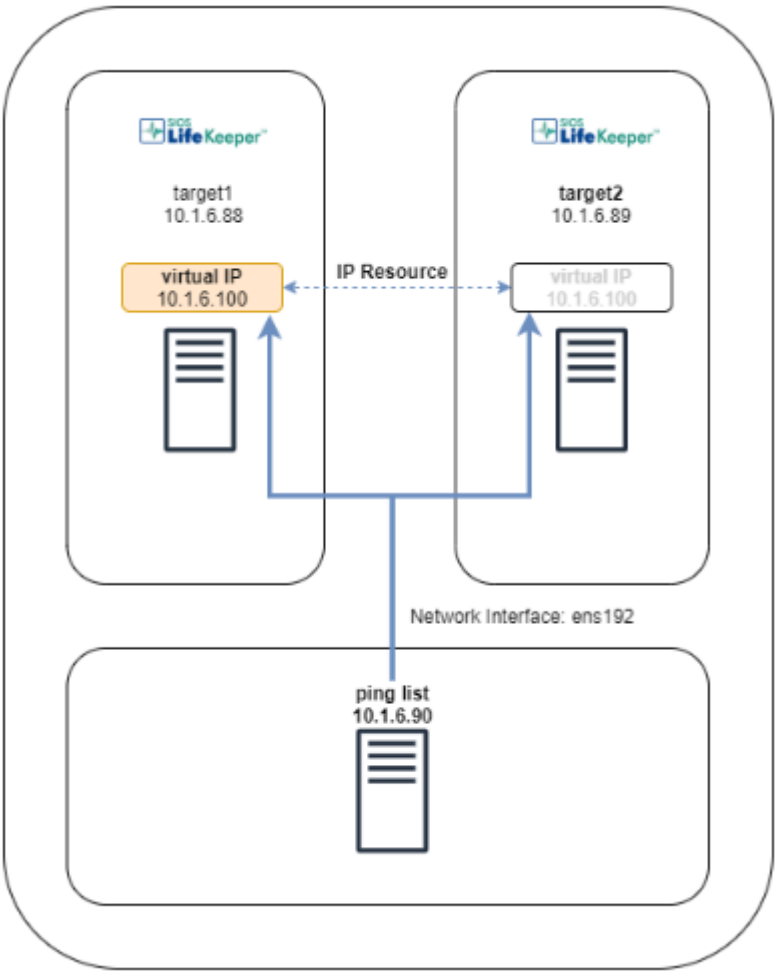
When you create a file system resource, multiple resources are automatically created with dependencies as shown above.

Creating IP Resources

This section describes how to create an IP resource.
The IP resource creates and protects a virtual IP address that can be switched between cluster nodes.

System Configuration

The environment created in this guide is a two-node configuration as shown below.



Prepare a virtual IP address that can be pinged (10.1.6.100 in the above figure).
Also prepare a system that can be pinged (10.1.6.90 in the above).

Check for a ping response using the following command:

```
# ip -4 addr add 10.1.6.100/24 dev ens192

# ping -c3 -I 10.1.6.100 10.1.6.90
```

There should be a ping response.

Once the ping response is verified, remove the IP address from the interface.

```
# ip -4 addr delete 10.1.6.100/24 dev ens192
```

Restrictions:

- Make sure that the virtual IP address you are trying to create is unique.
- Make sure that there is a system (other than the cluster nodes) that can respond to pings on the same network as the virtual IP address.

Note: IP resources use ping to validate the health of the network. Therefore, you need a system outside the cluster that can respond to pings.

Perform the following steps on target1

1. Creating a resource
   Execute the following command:

```
[target1]# lkcli resource create ip --tag ip-tag --ipaddr 10.1.6.100
```

Resource Settings

Item	Input Value
--tag	Tag name
--ipaddr	Virtual IP address

2. Configuring a ping list
   Run the following command to configure a ping list.

```
[target1]# lkcli resource config ip --tag ip-tag --pinglist 10.1.6.90
```

   Then bring the resource in service.

```
[target1]# lkcli resource restore --tag ip-tag
```

3. Extending a resource
   Execute the following command:

```
[target1]# lkcli resource extend ip --tag ip-tag --dest target2
```

Resource Settings

Item	Input Value
--tag	Tag name of the created resource
--dest	Backup node name

4. Setting up a ping list for the extended resource
   Run the following command to set up a ping list for the extended resource as well.

```
[target1]# lkcli resource config ip --tag ip-tag --pinglist 10.1.6.90 --remot
e target2
```

5. Checking the resource
   After creating and extending the resource, run the following command.
   The resource information is displayed.

```
[target1]# lkcli status -q
LOCAL    TAG     ID              STATE     PRIO  PRIMARY
target1  ip-tag  IP-10.1.6.100   ISP          1  target1
```

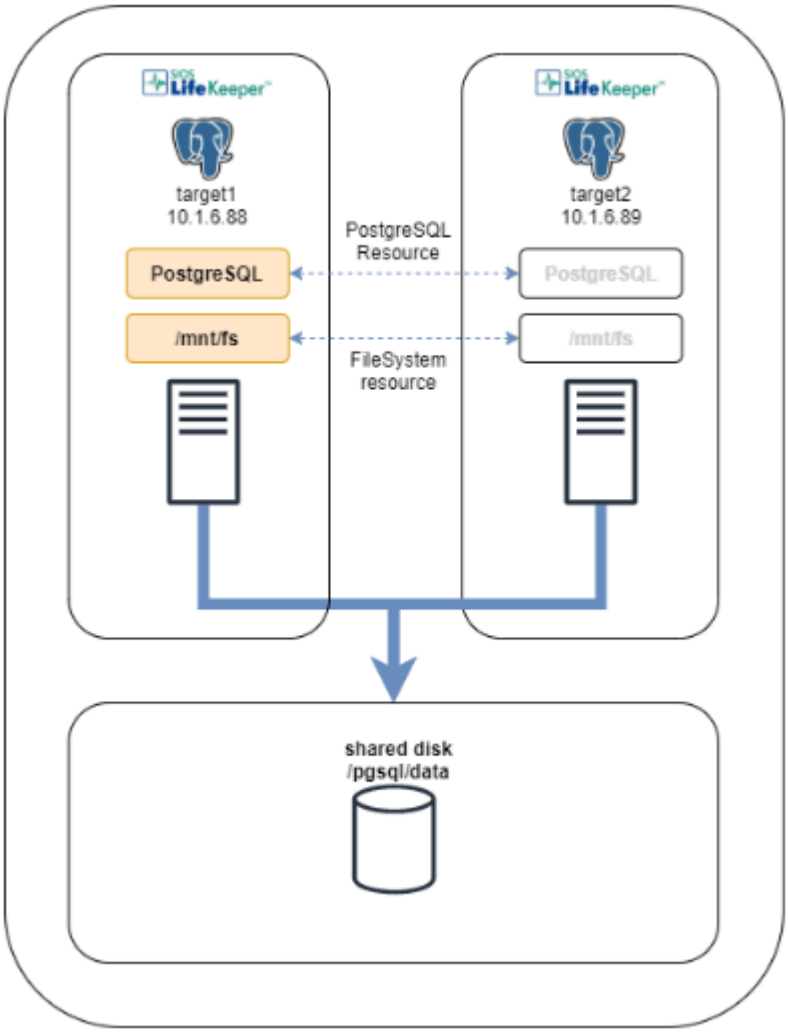
Creating a PostgreSQL Resource

This section describes how to create a PostgreSQL resource. The PostgreSQL resource provides the functionality to protect PostgreSQL database instances on LifeKeeper nodes.

System Configuration

The environment created in this guide is a two-node configuration as shown below.

Install PostgreSQL ARK in advance.



The environment in this guide is PostgreSQL

- PostgreSQL 9.2.24 is used in this guide.
- Create a PostgreSQL data directory in the file system on the shared storage.
- The PostgreSQL database admin user should use the "`postgres`" created when the database is initialized.
- Protect a PostgreSQL resource instance with an active/standby configuration.

*Make sure that your system does not fall under any of file system restrictions. See (LifeKeeper Core – Known Issues / Restrictions) for details.

Perform the following steps on target1 and target2

1. Installing PostgreSQL
   After installing, disable the autostart of the PostgreSQL service as follows:

```
# systemctl disable postgresql.service
```

Perform the following steps on target1

2. Mounting the file system
   Mount the file system in which the data directory will be created, refer to Step 1 in Creating file

system resources.

3. Creating a data directory
   Create a PostgreSQL data directory on the shared disk.

```
[target1]# mkdir -p /mnt/fs/pgsql/data
[target1]# chown -R postgres:postgres /mnt/fs/pgsql
```

   Change the following path described in the `/usr/lib/systemd/system/postgresql.service` file to the data directory on the shared disk.

```
Environment=PGDATA=/mnt/fs/pgsql/data
```

4. Database initialization
   Execute the following command to initialize the database.
   The database is created under the data directory.

```
[target1]# postgresql-setup initdb
```

5. Starting the PostgreSQL service
   Execute the following command to start the PostgreSQL service.

```
[target1]# systemctl start postgresql.service
```

6. Creating a resource
   Execute the following command:

```
[target1]# lkcli resource create pgsql --tag pgsql-tag --datadir /mnt/fs/pgsq
l/data --port 5432 --socket /tmp/.s.PGSQL.5432 --dbuser postgres --logfile /tm
p/pgsql-5432.lk.log
```

Resource Settings

Item	Input Value
--tag	Tag name
--datadir	Absolute path of the directory that contains the PostgreSQL database data
--port	Port number used by PostgreSQL
--socket	Path of the socket used by PostgreSQL
--dbuser	PostgreSQL database administrator user name
--logfile	Absolute path to the pg_ctl log file used to start and stop PostgreSQL

7. Extending the resource
   Execute the following command:

```
[target1]# lkcli resource extend pgsql --tag pgsql-tag --dest target2
```

Resource Settings

Item	Input Value
--tag	Tag name of the created resource
--dest	Backup node name

8. Checking the resource
   After creating and extending the resource, run the following command.
   The resource information is displayed.

```
[target1]# lkcli status -q
LOCAL    TAG            ID                                    STATE     PRIO  P
RIMARY
target1  pgsql-tag      target1.pgsql-5432                    ISP          1  t
arget1
target1   /mnt/fs       /mnt/fs                               ISP          1  t
arget1
target1    device17885  36000c292eb0c693b2efb44ed56556636-1  ISP          1  t
arget1
target1     disk17816   36000c292eb0c693b2efb44ed56556636    ISP          1  t
arget1
```

   When a PostgreSQL resource is created, a file system resource is automatically created as shown above.

5.5.3.3. LKCLI – Checking Cluster Status

Checking the Status of LiKeeper using `lkcli status` .

The `lkcli status -q` command provides current resource information and communication path information.

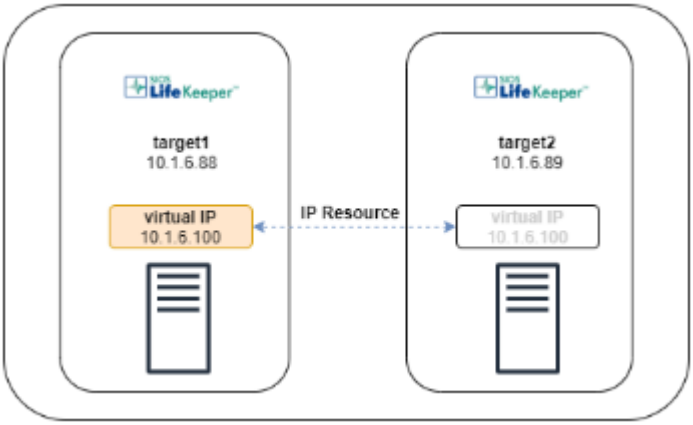
```
# lkcli status -q
LOCAL    TAG             ID              STATE     PRIO  PRIMARY
target1  ip-10.1.6.100   ip-10.1.6.100   ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

Check the resource status on other cluster nodes by using the --remote option. Refer to LKCLI (LifeKeeper Command Line Interface) for more information.

5.5.3.4. LKCLI – Verifying Switchover Behavior

This section explains how to perform a switchover from target1 to target2.

Configuration



Execute the command on target2.

1. Make sure the status of the resource that you will switch over is **ISP on target1**.

```
[target2]# lkcli status -q --remote target1
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. Make sure the status of the resource that you want to switch over is **OSU on target2**. Make a note of the resource tag name.

```
[target2]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target2  ip-10.1.6.100  ip-10.1.6.100  OSU         10  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

3. Switch the resource to target2.

```
[target2]# lkcli resource restore --tag ip-10.1.6.100
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

4. Make sure the resource is ISP on target2.

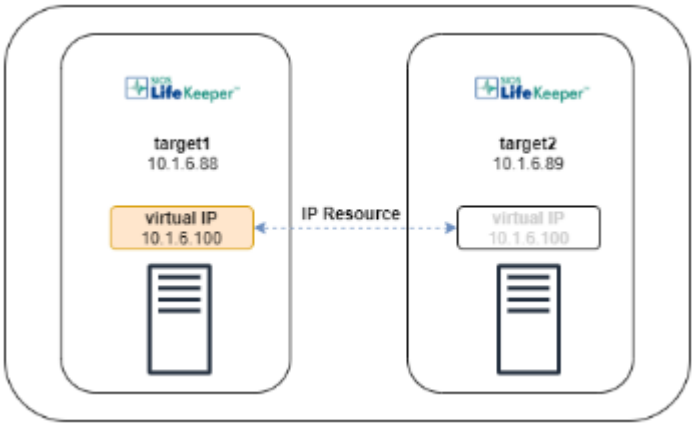
```
[target2]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target2  ip-10.1.6.100  ip-10.1.6.100  ISP         10  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

5.5.3.5. LKCLI – Maintenance Tasks

This section explains how to maintain the LifeKeeper-protected systems and resources.

Configuration

These instructions assume that the configuration is 2 nodes.



Maintaining a LifeKeeper Protected Machine

The maintenance tasks performed on target1 such as shutdown of the LifeKeeper protected machine, will have an impact on LifeKeeper and the resources.

Execute the command on target1.

1. Switch the active resource on target1 to target2

    i. Check the status of all of the resources on target1. Make a note of the resource tag name if there is a resource with an ISP status.

```
[target1]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

    ii. Switch the resources that are ISP on target1 to target2 one by one. You can execute the command from target1 by using “—remote” option.

```
[target1]# lkcli resource restore --tag ip-10.1.6.100 --remote target2
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

    iii. Make sure all the resources are OSU on target1.

```
[target1]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  OSU          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. Stop LifeKeeper on target1. It does not stop the resource by running “-f” option.

```
[target1]# lkcli stop -f
Removed /etc/systemd/system/lifekeeper-graphical.target.requires/lifekeeper.se
rvice.
Removed    /etc/systemd/system/lifekeeper-multi-user.target.requires/lifekeepe
r.service.
```

3. Perform the necessary maintenance on target1.

4. Start LifeKeeper on target1.

```
[target1]# lkcli start
Created symlink /etc/systemd/system/lifekeeper-graphical.target.requires/lifek
eeper.service → /usr/lib/systemd/system/lifekeeper.service.
Created symlink /etc/systemd/system/lifekeeper-multi-user.target.requires/life
keeper.service → /usr/lib/systemd/system/lifekeeper.service.
```

5. Bring the resources in-service on target1, if desired.

```
[target1]# lkcli resource restore --tag ip-10.1.6.100
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

Refer to Maintaining a LifeKeeper Protected System.

Maintaining LifeKeeper Protected Resources

This section explains how to perform maintenance for specific resources only.

Execute the command on target1.

1. Switch the active resources on target1 to target2.

    i. Check the status of all of the resources on target1. If there is a resource that has an ISP status, perform a switchover following the steps below.

```
[target1]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

ii. Check the resource tag names on target2 which are ISP on target1.

```
[target1]# lkcli status -q --remote target2
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target2  ip-10.1.6.100  ip-10.1.6.100  OSU         10  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

iii. Switch the resources that are ISP on target1 to target2 one by one.

```
[target1]# lkcli resource restore --tag ip-10.1.6.100 --remote target2
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

iv. Make sure all the resources have an OSU status on target1.

```
[target1]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  OSU          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. Perform maintenance for the resources that are OSU.

3. Bring the resources in-service on target1, if desired.

```
[target1]# lkcli resource restore --tag ip-10.1.6.100
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

Refer to Maintaining a Resource Hierarchy.

Changing the Resource Settings

This section explains how to change the IP resource settings. For other commands refer to LKCLI Subcommands for Each ARK.

These instructions assume you are using LifeKeeper v9.5.0.

1. Check the resource tag name that you want to change the setting for.

```
# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. Based on the tag name, check the resource type and current value. Refer to LKCLI Subcommands for Each ARK for more information.

```
# lkcli resource info --tag ip-10.1.6.100
---
app: comm
priority: 1
properties:
  device: ens192
  ipaddr: 10.1.6.100
  netmask: 255.255.255.0
  pinglist: ''
  realip: 0
  restoremode: Enabled
  srcaddr: 0
switchback: INTELLIGENT
tag: ip-10.1.6.100
typ: ip
```

3. Change the resource settings (example: `restoremode` will be Disabled at this time).

```
# lkcli resource config ip --tag ip-10.1.6.100 --restoremode Disabled
Performing restoremode change ...

restoremode change successful.
```

4. The resource settings have changed.

```
# lkcli resource info --tag ip-10.1.6.100
---
app: comm
priority: 1
properties:
  device: ens192
  ipaddr: 10.1.6.100
  netmask: 255.255.255.0
  pinglist: ''
  realip: 0
  restoremode: Disabled
  srcaddr: 0
switchback: INTELLIGENT
tag: ip-10.1.6.100
typ: ip
```

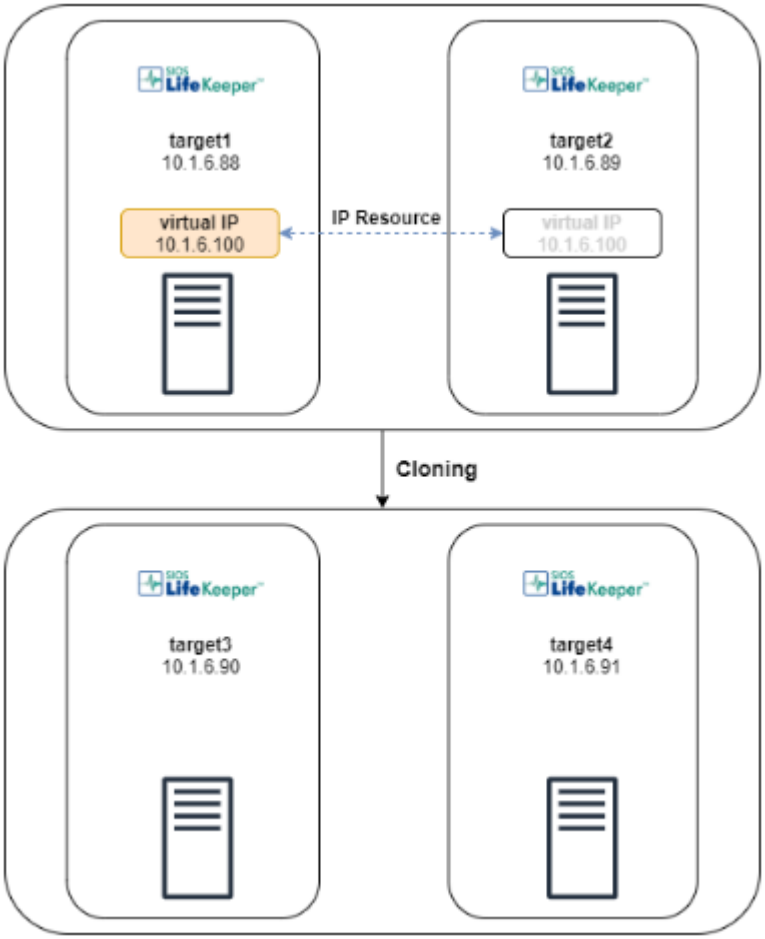
5.5.3.6. LKCLI – Replicate the Existing Cluster Settings

Inherit and Duplicate the Cluster Settings

This section describes the procedure for replicating a cluster with the same settings based on the cluster with the communication path and resource set.

Configuration

The steps describe preparing a 2-node LifeKeeper cluster and another LifeKeeper cluster to be replicated. The other cluster is not configured with communication paths and resources.



Caution

The following are resource and communication path restrictions for the resource and communication path output when using the export command. Before performing the steps, check to see if the environment is supported.

- Make sure the communication path protocol that can be exported is TCP/IP.

```
# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2    TCP   10.1.6.88/10.1.6.89  ALIVE        1
```

- Make sure all resources in the cluster to be exported are supported ARKs with LKCLI. Refer to LKCLI Subcommands for Each ARK for a list of supported ARKs and resource types.

```
# lkcli resource info --tag ip-10.1.6.100
---
app: comm
priority: 1
properties:
  device: ens192
  ipaddr: 10.1.6.100
  netmask: 255.255.255.0
  pinglist: ''
  realip: 0
  restoremode: Disabled
  srcaddr: 0
switchback: INTELLIGENT
tag: ip-10.1.6.100
typ: ip
```

Steps

Execute the command on target1

1. Save the current settings to the file in the cluster where LifeKeeper is configured.

```
[target1]# lkcli export > src_settings.yml
```

2. Copy the exported file to the cluster to be replicated.

Execute the command on target3.

3. For the exported configuration file, edit only the IP address and hostname manually.

```
[target3]# sed -e "s/10\.1\.6\.88/10\.1\.6\.90/" -e "s/target1/target3/" -e
"s/10\.1\.6\.89/10\.1\.6\.91/" -e "s/target2/target4/" src_settings.yml &gt; d
est_settings.yml
```

4. Make sure there are no communication path or resources in the cluster to be replicated (check all nodes in the cluster).

   If any exist, run the following command to delete: `lkcli clean --mode all`. Note that the setting won't be restored after executing the clean command.

```
[target3]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY
```

Execute the command on target3 and target4

5. Import the resources in the replicated system and check that the communication path can be created. Execute this command on all of the nodes in the cluster.

```
# lkcli import commpath --file dest_settings.yml
Performing commpath 'target3:10.1.6.90/10.1.6.91' create...
Commpath 'target3:10.1.6.90/10.1.6.91' created successful.
```

```
# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target4    TCP   10.1.6.90/10.1.6.91  ALIVE        1
```

*The communication path status will be ALIVE when it is created in both directions between 2 nodes.

6. Import the resources in the replicated system and check that the resource can be created. Execute this command only once in the cluster.

```
# lkcli import resource --file settings.yml
BEGIN create of "ip-10.1.6.100"
LifeKeeper application=comm on target1.
LifeKeeper communications resource type= ip on target1.
Creating resource instance with id ip-10.1.6.100 on machine target1
Resource successfully created on target1
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
END successful create of "ip-10.1.6.100".
Removing ping list for subnet 172.31.0.0...
Performing restoremode change ...

restoremode change successful.
Building independent resource list
Checking existence of extend and canextend scripts
Checking extendability for ip-10.1.6.100
Pre Extend checks were successful
Extending resource instances for ip-10.1.6.100
Creating dependencies
Setting switchback type for hierarchy
Creating equivalencies
LifeKeeper Admin Lock (ip-10.1.6.100) Released
Hierarchy successfully extended
Removing ping list for subnet 172.31.0.0...
```

7. The communication path and resource are created.

```
# lkcli status -q
```

```
LOCAL    TAG               ID            STATE      PRIO  PRIMARY
target3  ip-10.1.6.100  ip-10.1.6.100  ISP           1  target3

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target4  TCP     10.1.6.90/10.1.6.91  ALIVE        1
```

6. Application Recovery Kits

SIOS Protection Suite for Linux Application Recovery Kits (ARKs) include tools and utilities that allow SPS to manage and control a specific application. The following optional recovery kits are available with this release of SPS.

Apache Recovery Kit Administration Guide

DB2 Recovery Kit Administration Guide

Recovery Kit for EC2 Administration Guide

LVM Recovery Kit Administration Guide

IP Recovery Kit Administration Guide

MySQL Recovery Kit Administration Guide

MD Recovery Kit Administration Guide

WebSphere MQ Recovery Kit Administration Guide

NAS Recovery Kit Administration Guide

NFS Recovery Kit Administration Guide

Oracle Recovery Kit Administration Guide

PostgreSQL Recovery Kit Administration Guide

Postfix Recovery Kit Administration Guide

Route53 Recovery Kit Administration Guide

Samba Recovery Kit Administration Guide

SAP Recovery Kit Administration Guide

SAP HANA Recovery Kit Administration Guide

SAP MaxDB Recovery Kit Administration Guide

Sybase Recovery Kit Administration Guide

VMDK Shared Storage Recovery Kit Administration Guide

6.1. Apache Recovery Kit Administration Guide

The SIOS Protection Suite (SPS) for Linux Apache Web Server Recovery Kit provides fault resilience for Apache Web Server software in an SPS environment.

This guide explains the following topics:

- SPS Documentation. A list of all the SPS for Linux documentation and where the information is available.
- Requirements. Before you can install and set up the recovery software, your server must meet certain hardware and software requirements. You should refer to the SPS Installation Guide for specific instructions on how to install or remove the SPS Apache Recovery Kit.
- Configuring Your Recovery Kit. To ensure that your SPS configuration provides the protection and flexibility you require, you need to be aware of the configuration rules. To appropriately plan your configuration, you must understand your network configuration, interface selection, user system setup, hierarchy options and the Apache configuration tasks. In addition to planning your configuration, this section also includes configuration examples and the specific tasks required to configure your Recovery Kit.
- Troubleshooting. This section provides a list of informational and error messages with recommended solutions.

6.1.1. SPS Documentation and Apache References

The following is a list of SPS related information available from SIOS Technology Corp.:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation (available from the Help menu within the LifeKeeper GUI)
- SPS for Linux Installation Guide

This documentation, along with documentation associated with other SPS Recovery Kits, is provided online at:

http://docs.us.sios.com

Reference Documents

The following is a list of reference documents associated with the Apache Web Server application and the SPS Apache Recovery Kit:

- Apache Online documentation
- Apache: The Definitive Guide, 2nd Edition, Ben Laurie and Peter Laurie, O'Reilly & Associates, Inc. 1999

6.1.2. Apache Recovery Kit Requirements

Before attempting to install or remove the Apache Recovery Kit, you must understand the hardware and software requirements for the package and the installation and removal procedures.

Kit Hardware and Software Requirements

Before installing and configuring the LifeKeeper Apache Recovery Kit, be sure that your configuration meets the following requirements:

- **Servers**. The Recovery Kit requires two or more supported computers configured in accordance with LifeKeeper requirements described in the SPSfor Linux Technical Documentation and the SPS Release Notes, which are located on our SIOS Technical Documentation site at docs.us.sios.com.

- **LifeKeeper software**. You must install the same version of LifeKeeper software and any patches on each server. Please refer to the SPS Release Notes and SPS for Linux Technical Documentation for specific LifeKeeper requirements.

- **LifeKeeper IP Recovery Kit**. You must have the same version of this Recovery Kit on each server.

- **IP network interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

> **Note:** Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation, and so forth.

- **TCP/IP software**. Each server also requires the TCP/IP software.

- **Apache software**. Each server must have the Apache Web Server software installed and configured prior to configuring LifeKeeper and the LifeKeeper Apache Web Server Recovery Kit, including any DSO (Dynamic Shared Object) modules that will be used. The same versions of all web server software packages should be installed on each server. Consult the SPS Release Notes or your sales representative for the latest release compatibility and ordering information.

Refer to the SPS Installation Guide for specific instructions on how to install or remove the LifeKeeper Apache Recovery Kit.

6.1.3. Configuring Apache Web Server with LifeKeeper

This section contains definitions and examples of typical LifeKeeper Apache Web Server configurations and information you should consider before you start to configure Apache Web Server.

Please refer to the SPS for Linux Technical Documentation for instructions on configuring LifeKeeper Core resource hierarchies.

6.1.3.1. Configuration Definitions and Examples

Apache Web Server supports multiple instances of the httpd daemon running at the same time. Each LifeKeeper Apache Web Server hierarchy corresponds to a separate Apache instance with its own “server root” directory. Each instance may support one or more web sites, depending on whether or not it has been configured to use “virtual hosts.”

Primarily, the server root directory defines an Apache Web Server instance, since this directory will contain the conf/httpd.conf configuration file that specifies how the web instance is configured. The Apache configuration directives within this file will determine where the log files, web documents, other configuration files, etc. are located for the instance, as well as which IP and/or domain name addresses will be used.

It is useful to characterize Apache Web Server configurations with LifeKeeper based on whether or not a LifeKeeper file system (which uses shared storage) will be used. A single shared file system may be used for the server root directory (along with the configuration file conf/httpd.conf) and/or the document root directories (and optionally the httpd executable itself). Whether you choose to use a local or a shared configuration for a particular Apache instance will depend on two main factors: the difficulty of maintaining separate, identical copies of the configuration files and/or web site documents, and the availability and accessibility of storage which can be shared (or mirrored) between two or more servers. Note, however, that you may choose to configure both local and shared Apache instances on the same servers.

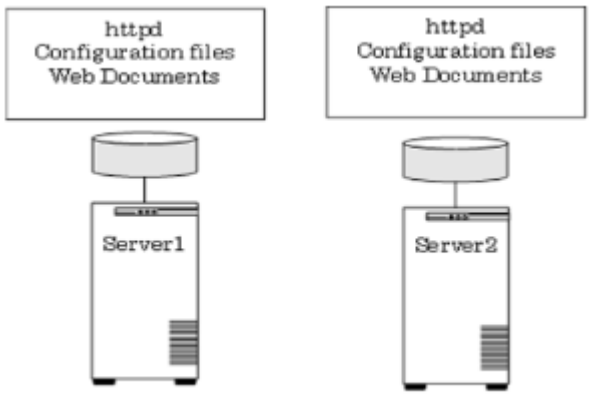
The following sections provide examples of Local and Shared Apache Web Server configurations in a LifeKeeper environment and summarize the main characteristics of each.

Local Configuration

In a typical local configuration, nothing is shared between the servers. Identical copies of the Apache Web Server configuration file, web documents, DSO modules (and their configuration files, if any), and the httpd executable reside in exactly the same locations on each server. It is the responsibility of the Apache administrator to maintain identical copies of the Apache components on the different servers.

Each web site is assigned an IP address – or a domain address that maps to a particular IP address – through the configuration file, and a LifeKeeper IP address is created for each and added to the Apache resource hierarchy. When the Apache hierarchy is switched over from one server to another, this particular httpd instance is stopped and the IP addresses are deactivated on the first server, then the IP addresses are reactivated and the instance started on the other server. Clients will then be automatically connected via TCP/IP to the identical web site on the other server.

Figure 1. Local Configuration



Configuration Notes:

- Figure 1 is an example of a local configuration where nothing is on a shared file system.
- Each server has the same version of the Apache Web Server executable at the same location (typically /usr/bin/httpd).
- Each server has the same server root directory where identical copies of the configuration file for each instance are placed.
- Each server has the same document root directory(s) where identical copies of the web document for each instance are placed.
- If DSO modules are being used, each server has identical copies at the same location.

Creating an Apache Web Server resource hierarchy on Server 1:

Server:	Server 1
Web Server Binary Location:	/usr/sbin/httpd
Web Server Root Directory:	/home/www/examples/instance1/
Root Tag	apache-www.examples.instance1

Extending an Apache Web Server resource hierarchy to Server 2:

Template Server:	Server 1
Tag to Extend	apache-www.examples.instance1
Target Server	Server2
Target Priority:	10

Note that when an Apache resource hierarchy is extended to one or more additional servers, the same Web Server Binary Location and Web Server Root Directory must be used on all servers, regardless of whether this is a local or a shared configuration. See the discussion above and the section on Specific

Configuration Considerations for Apache Web Server for additional information. Also during hierarchy extension, LifeKeeper extends all the dependent resources which are part of the Apache resource hierarchy.

Shared Configuration

In a typical shared configuration, the server root directory and the document root directories are all on the same shared file system. The same configuration file and web documents are shared between the servers, so there is no need to maintain identical copies on each server. If DSO modules are being used, they also can be located on the same shared file system, along with any configuration files or resources they may need.

Note that you may choose to place only the web documents on a shared file system. This will still appear much like a typical local configuration, since the server root directories will be local, but the hierarchy will also include a shared file system.

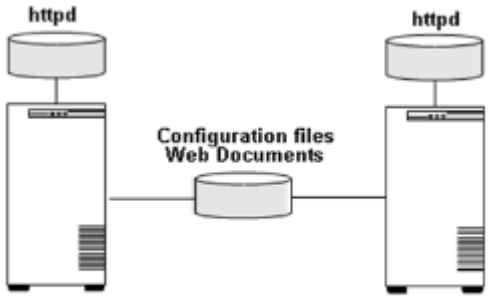
If you wish to use a particular version or a separate copy of the Apache executable for this Apache resource hierarchy, you may place this executable on the shared file system as well and it will be available only to this instance. To do this, simply enter the full path of the httpd executable on the shared file system when prompted for the Web Server Binary Location.

Note that only one shared file system may be used, since this assures that all required components which are on shared storage will be available at the same time. If you choose to use a Web Server Binary Location on a shared file system, you must also choose a Web Server Root Directory on the same shared file system, and all DocumentRoot directories configured for this server root must be on the same shared file system. Likewise, when you choose a Web Server Root Directory on a shared file system, all DocumentRoot directories must be on the same shared file system. If neither the binary nor the server root is placed on a shared file system, but any of the DocumentRoot directories are shared, all DocumentRoot directories must be shared on the same file system.

These rules can be summarized as follows:

- If the Apache executable is shared, then the server root directory must be shared.
- If the server root directory is shared, then all DocumentRoot directories must be shared.
- If any DocumentRoot directory is shared, they must all be shared.
- Only one shared file system is allowed for each Apache resource hierarchy.

Figure 2. Shared Configuration



Configuration Notes:

- Figure 2 is an example of a shared configuration with shared configuration files and web documents.
- You may choose to place only the web documents on a shared file system. This will still appear much like a typical local configuration, except that the hierarchy will also include a shared file system.
- If DSO modules are used, they may reside on the shared file system, along with any configuration files or resources they need.

Creating an Apache Web Server resource hierarchy on Server 1:

Server:	Server1
Web Server Binary Location:	/usr/sbin/httpd …..OR…. /shared/example/instance2/bin/httpd
Web Server Root Directory:	/shared/example/instance2
Root Tag	apache-shared.example.instance2

Extending an Apache Web Server resource hierarchy to Server 2:

Template Server:	Server1
Tag to Expand	apache-shared.example.instance2
Target Server	Server2
Target Priority:	10

6.1.3.1.1. Active/Standby and Active/Active Configurations

Apache Web Server is called an Active/Active application with LifeKeeper. This means that more than one instance of Apache can be running on a server at any time. For example, if two servers are running an instance of Apache and one server fails, the Apache instance on this server can fail over to the other server and it can continue to run its own instance as well. Some applications simply don't support this, so you would have to keep a server available for each instance of the application. These are called Active/Standby applications. Some applications can be configured either way.

There may be circumstances when you might want to operate Apache in an "active/standby" mode, particularly if only one of your servers is used primarily for running Apache. In this particular case, you should disable the automatic startup of the standard Apache default installation so that nothing is running on the backup server(s).

By manually bringing the Apache instances In Service on one or more particular servers, you can distribute the workload as you like. And by adjusting the server priorities for each of your instances, you can configure the Apache instances to fail over to a particular server only as a last resort, or to fail over to different servers to distribute the workload when a failure occurs.

If you disable automatic startup of Apache on all servers in the cluster, it is possible to use the default server root directory "/etc/http" for a single LifeKeeper Apache resource hierarchy by simply configuring this instance to use LifeKeeper IP addresses – and possibly using a shared file system for the document root directories. Note, however, that this would be an Active/Standby configuration (as described above), so you could no longer start up the default instance in the usual way. Of course, the default server root directory cannot be used for more than one hierarchy, since the server root must be unique.

6.1.3.2. Configuration Considerations for Apache Web Server

Before you create Apache resource hierarchies, you will need to make sure you have completed the following configuration tasks for the Apache Web Server application:

1. In the case Apache package attached to a distribution is installed, it is normally set to autostart at system startup, so it conflicts with LifeKeeper's protection. To avoid the conflict, refer to the manual of each distribution and disable the automatic start up.

> **Note: For Apache on SuSE:** The default installation of Apache on SuSE does not place the httpd.conf configuration file in a subdirectory of ServerRoot called conf. If you are using the default installation of Apache on SuSE, you must relocate the configuration file to the directory /etc/httpd/conf.

2. You must create a separate, distinct root directory for each LifeKeeper Apache Web Server hierarchy. This "server root" directory corresponds to the Apache "ServerRoot" configuration and command line parameters. Each LifeKeeper Apache resource hierarchy will correspond to a unique Apache instance with its associated server root directory. Note that the server root directory must be identical on all servers that are configured for a particular Apache hierarchy. You must place all configuration file information for the web site in the standard location relative to the server root (conf/httpd.conf) so that it can be found and accessed by the LifeKeeper software.

3. You must configure all web sites (virtual hosts) to listen on specific LifeKeeper IP addresses using BindAddress or Listen directives. These LifeKeeper-protected IP addresses must already be created and available to be brought in-service where the Apache hierarchy is to be created. They will automatically be added to the Apache resource hierarchy.

   If you will be using a LifeKeeper shared file system, you must make all necessary preparations for the file system creation prior to creating the Apache hierarchy. In particular, the file system must be mounted on the server where the Apache hierarchy is to be created. If the LifeKeeper file system hierarchy has not already been created, it will automatically be created along with the Apache hierarchy, then joined to the Apache resource hierarchy.

   Consult the Apache Web Server documentation for detailed information on configuring virtual hosts. As noted above, you must configure all Apache instances to listen on specific LifeKeeper-protected addresses. For example, the configuration file for an instance that combines IP-based and name-based virtual hosts would include directives like the following:

```
User webuser
Group webgroup
ServerName localhost

Listen 172.17.100.55:8000
```

```
NameVirtualHost 172.17.100.55:8000
Listen 172.17.100.56:80

<Virtualhost site.name_one:8000>
ServerName site.name_one
DocumentRoot /shared/site/name_one
</VirtualHost>

<VirtualHost site.name_two:8000>
ServerName site.name_two
DocumentRoot /shared/site/name_two
</VirtualHost>

<VirtualHost 172.17.100.56:80>
ServerName site.ip
DocumentRoot /shared/site/ip
</VirtualHost>
```

4. If SSL support is enabled for your Apache instance, you must configure the SSL Listen directive, often found in a separate ssl.conf file, to use the appropriate LifeKeeper-protected IP address. Otherwise, the creation of your Apache hierarchy will fail with an error indicating that the IP address 0.0.0.0 is not LifeKeeper protected. Note that SSL support is enabled by default in the Apache configuration files of some Linux distributions.

   For example, change the following entry in the default SSL configuration file at /etc/httpd/conf.d/ssl.conf from

   ```
   Listen 0.0.0.0:443
   ```

   to

   ```
   Listen 172.17.100.55:443
   ```

5. For a Local configuration, you must install and configure Apache in the same location on both the primary and all backup servers and set up identical (or equivalent) configuration files in the same server root directory on all servers. Also, all document root directories must exist on all servers and should contain identical files. (See the section on Local Configuration in Configuration Definitions and Examples.)

6. For a Shared configuration, you will typically configure the server root directory on a LifeKeeper shared file system. Note that only one shared file system may be used, since this assures that all required components which are on shared storage will be available at the same time. Therefore, all document root directories must be subdirectories of the same shared file system, but they need not be subdirectories of the server root directory itself. You may place an Apache executable on the same shared file system as well, but this executable will only be available for use by this

particular Apache resource hierarchy.

> **Note:** You don't necessarily need to place the server root directory on a shared file system in order to make use of shared storage. You may choose a local server root directory for configuration files, etc., and place only the document root directories on a shared file system. However, you must configure identical server root directories and identical (or equivalent) configuration files on all servers (as for a Local configuration as described above), and all document root directories must be on the same shared file system. (See the section on Shared Configuration in Configuration Definitions and Examples.)

7. Some web site implementations make use of DSO (Dynamic Shared Object) modules to extend Apache support for certain features. For example, there are modules available that implement functionality for PHP and Perl. These modules can be loaded and accessed at runtime by the Apache core. If you are using modules, they must be identically configured on every server in the cluster. Consult the documentation for the module package, and the vendor-supplied documentation for configuring Apache to use modules on your Linux platform. Depending on the module and the resources it uses, some objects may be required to reside on shared storage to facilitate proper failover. In some cases, a module may even need to be protected separately using the Generic Application Recovery Kit, or a custom recovery kit.

8. If you are using the SSL (Secure Sockets Layer) module with Apache, it is important that the server not be password protected. When the web server is password protected, the administrator must interactively type in the password at a prompt each time the daemon starts. Since this manual step is not consistent with a High Availability environment where recovery time is critical, LifeKeeper does not support password protected instances. Use the following command to remove the password:

   ```
   openssl rsa -in server.key -out unprotected_server.key
   ```

   Enter the server key password when prompted. To preserve the security of your site, make sure that the file is readable only by root!

   ```
   chmod 400 unprotected_server.key
   ```

   During the hierarchy creation of an Apache instance, the Recovery Kit checks that the resource is not password protected. If it is password protected, hierarchy creation will fail with an error message. However, when the instance is extended to another server, the Recovery Kit does not check for password protection on the backup server. You need to make sure that the hierarchy you are extending is not password protected.

   The server key file(s) (specified by the SSLCertificateKeyFile directive(s) in the Apache configuration file) must have the same name and be at the same location on all servers in the cluster.

Note: The PID file name of the httpd process that LifeKeeper uses has the following format:

```
"/var/run/httpd.<TAG name>.pid"
```

This PID file name is different from the default PID file name used by the OS. If you need to reference this PID file (ex. log rotate), please note the LifeKeeper PID file name and format.

9. In the case where "APACHE_SERVER_FLAGS" is defined in "/etc/sysconfig/apache2" in SuSE environments, add a "-D" in front of the flag name.

   Example: `APACHE_SERVER_FLAGS="-D SSL"`

   If there is no "-D", the resource creation and start up can fail.

6.1.4. LifeKeeper Configuration Tasks for Apache

You can perform the following configuration tasks from the LifeKeeper GUI. The following four tasks are described in this guide, as they are unique to an Apache resource instance and different for each Recovery Kit.

- Create a Resource Hierarchy. Creates an application resource hierarchy in your LifeKeeper cluster.
- Extend a Resource Hierarchy. Extends a resource hierarchy from the primary server to a backup server.
- Unextend a Resource Hierarchy. Unextends (removes) a resource hierarchy from a single server in the LifeKeeper cluster.
- Delete a Resource Hierarchy. Deletes a resource hierarchy from all servers in your LifeKeeper cluster.

The following tasks are described in the GUI Administration section within the SPS Technical Documentation, because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View/Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, we explain how to configure your Recovery Kit by selecting certain tasks from the Edit menu of the LifeKeeper GUI. You can also select each configuration task from the toolbar. You may also right click on a global resource in the Resource Hierarchy Tree (left-hand pane) of the status display window to display the same drop down menu choices as the Edit menu. This, of course, is only an option when a hierarchy already exists.

You can also right click on a resource instance in the Resource Hierarchy Table (right-hand pane) of the status display window to perform all the configuration tasks, except Creating a Resource Hierarchy, depending on the state of the server and the particular resource.

6.1.4.1. Creating an Apache Web Server Resource Hierarchy

IMPORTANT:

Before you create your Web Server resource hierarchy, you must make sure that your Apache configuration file has included an existing LifeKeeper-protected IP resource.

In a shared environment where the web documents and/or configuration files are on a shared disk, you must make sure that the shared file system is mounted. It is also important to remember that you require a working communication path (i.e. heartbeat) before you can extend your resource to a backup server.

To create a resource instance from the primary server, you should complete the following steps:

1. From the LifeKeeper GUI menu, select Edit, then Server. From the menu, select Create Resource Hierarchy.

   The Apache Web Server should not be running when you create the resource. However, if you set up the listen variable in the system configuration file, the default daemon can be allowed to run.

   The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized recovery kits installed within the cluster.

2. Select Apache Web Server and click **Next**.

3. You will be prompted to enter the following information. When the Back button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter an error requiring you to correct previously entered information. You may click Cancel at any time to cancel the entire creation process.

Field	Tips
Switchback Type	Choose either intelligent or automatic This dictates how the Apache instance will be switched back to this server when the server comes back up after a failover. The switchback type can be changed later from the General tab of the Resource Properties dialog box.
Server	Select the Server where you want to place the Apache Web Server (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop down list box.
Web Server Binary Location	Select or enter the full path name (including the file name) of the httpd Apache Web Server daemon. The default is /usr/sbin/httpd.
Web Server Root	You must provide the full path of the Web Server Root directory; a relative path or symbolic link may not be used. The Apache Web Server configuration file is located in conf/ httpd.conf relative to the Server Root.

Directory	**Note:** At this point, LifeKeeper will check that there is a protected IP resource available. It will also validate that you have provided valid data to create your Apache Web Server resource hierarchy. If LifeKeeper detects a problem with either of these validations, an ERROR box will appear on the screen. If the Web Server Root Directory path is valid, but there are errors with the Apache configuration itself, you may pause to correct these errors and continue with the hierarchy creation. You may even pause to create any LifeKeeper IP resources that are required.
Root Tag	Select or enter the tag name given to the Web Server hierarchy. You can select the default, which is apache <root directory>, or enter your own tag name.

4. Click **Create**. The Create Resource Wizard will then create your Apache resource hierarchy. LifeKeeper will validate the data entered. If LifeKeeper detects a problem, an error message will appear in the information box.

5. An information box will appear indicating that you have successfully created a Apache resource hierarchy, and you must Extend that hierarchy to another server in your cluster in order to achieve failover protection. Click **Next**.

> **Note:** You may encounter error messages indicating that the new Apache instance has failed to start correctly. Note that the new Apache hierarchy must be started (In Service) before it can be extended to another system. You may pause at this point and correct the problem based on the error message displayed, then bring the new hierarchy In Service before proceeding with extending the hierarchy.

6. Click **Continue**. LifeKeeper will then launch the Pre-ExtendWizard. Refer to Step 2 under Extending an Apache Resource Hierarchy (below) for details on how to extend your resource hierarchy to another server.

   If you click Cancel, a dialog box will appear warning you that you will need to come back and extend your Apache resource hierarchy to another server at some other time to put it under LifeKeeper protection.

6.1.4.2. Extending an Apache Web Server Resource Hierarchy

This operation can be started from the Edit menu, or initiated automatically upon completing the Create Resource Hierarchy option, in which case you should refer to Step 2 below.

1. On the **Edit** menu, select **Resource**, then **Extend Resource Hierarchy**. The Pre-Extend Wizard appears. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the LifeKeeper Extend Resource Hierarchy defaults and want to bypass the prompts for input/ confirmation, click **Accept Defaults**.

2. The Pre-Extend Wizard will prompt you to enter the following information.

   Note: The first two fields appear only if you initiated the Extend from the Edit menu.

Field	Tips
Template Server	Enter the server where your Apache resource is currently in service. It is important to remember that the Template Server you select now and the Tag to Extend that you select in the next dialog box represent an in service resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you selected. The drop down box in this dialog provides the names of all the servers in your cluster.
Tag to Extend	Select the name of the Web Server instance you wish to extend from the template server to the target server. The wizard will list in the drop down list box all the resources that you have created on the template server, which you selected in the previous dialog box.
Target Server	Select the Target Server where you are extending your Web Server resource hierarchy. The drop down box provides the names of the servers in your cluster that are not already in the selected hierarchy.
Switchback Type	Select either intelligent or automatic. This dictates how the Web Server instance will be switched back to this server when it comes back into service after a failover to the backup server. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths.
Template Priority	Select or enter a priority for the template hierarchy. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note:** This selection will appear only for the initial extend of the hierarchy.
Target Priority	Select or enter the Target Priority of your extended Web Server resource. The priority is a number between 1 and 999 indicating a server's priority in the cascading failover sequence for the resource. The hierarchy priorities are sorted numerically, where a lower number means a higher priority (the number 1 indicates the highest priority). Note that LifeKeeper automatically assigns the number "1" to the server that the hierarchy is created on. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.
	After receiving the message that the pre-extend checks were successful, click Next.

	Depending upon the hierarchy being extended, LifeKeeper will display a series of information boxes showing the Resource Tags to be extended, which cannot be edited. Click Extend
Network Interface	Select or enter the Network Interface. This is the name of the network interface (i.e. Ethernet card) the IP resource will use on the target server.
Backup Interface	Select a Backup Interface if you want to engage the IP Local Recovery feature on the server to which you are extending the IP resource. The default value is none; however, if you have another network interface card configured on this server, it should be listed in the drop down list.
IP Resource Tag	Select or enter the IP Resource Tag. This is the resource tag name to be used by the IP resource being extended to the target server.
Root Tag	Select or enter the Root Tag. This is the tag name given to the Web Server hierarchy. By default, the Root Tag name should be the same on both the template and target server.
Mount Point	This selection appears only when the Web Server Root Directory is on a shared file system. Select or enter the Mount Point of the shared file system where the Web Server Root Directory is located. The Template Server and Target Server should have the same mount point for the shared Web Server Root Directory. The default mount point provided in the dialog box should be selected in most cases.
Root Tag	This selection appears only when the Web Server Root Directory is on a shared file system. Select or enter the Root Tag. This is the tag name of the shared file system.

3. An information box will appear verifying that the extension is being performed. Click Next Server if you want to extend the same Apache resource instance to another server in your cluster. This will repeat the Extend Resource Hierarchy operation.

   If you click **Finish**, LifeKeeper will verify that the extension of the Web Server resource was completed successfully.

4. Click **Done** in the last dialog box to exit from the Extend Resource Hierarchy menu selection.

   Note: Be sure to test the functionality of the new instance on both servers.

6.1.4.3. Unextending an Apache Web Server Resource Hierarchy

1. On the Edit menu, select **Resource**, then **Unextend Resource Hierarchy**

2. Select the **Target Server** where you want to unextend the Web Server resource. It cannot be the server where the Web Server is currently in service. (This dialog box will not appear if you selected the Unextend task by right clicking on a resource instance in the right pane).

   Click **Next**.

3. Select the Web Server hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the Unextend task by right clicking on a resource instance in either pane).

4. An information box appears confirming the target server and the Web Server resource hierarchy you have chosen to unextend. Click **Unextend**.

5. Another information box appears confirming that the Web Server resource was unextended successfully.

6. Click **Done** to exit.

6.1.4.4. Deleting an Apache Web Server Resource Hierarchy

It is important to remember that if you delete a hierarchy before you take it out-of-service, the resource hierarchy will be removed from LifeKeeper protection, but the Apache instance will continue to run on the currently active server unless it is manually stopped or the system is rebooted. Attempting to recreate the same Apache hierarchy with a different IP address(s) or to create a new Apache hierarchy using the previously used IP address(s) (but using a different Server Root), will result in conflicts with the Apache instance that was left running with that same address.

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Delete Resource Hierarchy**.

2. Select the name of the **TargetServer** where you will be deleting your Web Server resource hierarchy. Click **Next** to proceed to the next dialog box.

   Note: If you selected the Delete Resource task by right clicking from the right pane on an individual resource instance, or from the left pane on a global resource where the resource is on only one server, the Target Server dialog will not appear.

3. Select the **Hierarchy to Delete**. (This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in the left or right pane.) Remember that the list box displays every hierarchy on the target server , both in service and out of service. If you want to stop the Apache instance and remove the resource hierarchy from LifeKeeper protection, you must make sure that the hierarchy you choose is out-of-service before deleting it.

   Click **Next**.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Next**.

5. Another information box appears confirming that the Web Server resource was deleted successfully.

6. Click **Done** to exit.

6.1.4.5. Testing an Apache Web Server Resource Hierarchy

You can test your Apache resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Performing a Manual Switchover from the GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting Edit, then Resource, then finally InService from the drop down menu. For example, an in service request executed on a backup server causes the application hierarchy to be placed in service on the backup server and taken out of service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the Out of Service request, the application is taken out of service without bringing it in service on the other server.

Recovery Operations

When the primary server fails, the Apache Recovery Kit software performs the following tasks:

- Brings Apache into service on the backup server by bringing in service the IP address(s) on one/more of that server's physical network interfaces
- Mounts the file system—if one is being used—on the shared disk on that server
- Starts the daemon processes related to Apache

After recovery, Apache Web Server users may reconnect by clicking on the Reload/Refresh button of their browsers.

6.1.5. Apache Web Server Troubleshooting

This section provides a list of messages that you may encounter during the process of creating and extending a LifeKeeper Apache Web Server resource hierarchy, removing and restoring a resource, and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. Messages from other SPS components are also possible. In these cases, please refer to the Message Catalog(located on our Technical Documentation site under “Search for an Error Code”) which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

Messages in this section fall under these topics:

- Hierarchy Creation
- Extend Hierarchy
- Hierarchy Remove, Restore and Recovery

6.1.5.1. Apache Hierarchy Creation Errors

The error messages that might be displayed during the Apache hierarchy creation are listed below, along with a suggested explanation for each. Error messages displayed by the LifeKeeper core and by other recovery kits are not listed in this guide. Note that you may stop to correct any of the problem(s) described here, and then continue with hierarchy creation from the point where you left off – including creating any new LifeKeeper resources you might need for your Apache configuration.

During Validation of Web Server Binary Location

"Error: valid_httpd_path: Must specify absolute path to httpd executable."

Enter the full, absolute path name to a valid Apache httpd executable.

"Error: valid_httpd_path: File does not exist at path specified."

A valid Apache httpd executable does not exist at the location specified.

"Error: valid_httpd_path: Httpd failed to display Server version."

The httpd executable at the location specified does not display the standard Apache "Server version."

"Error: valid_httpd_path: Incorrect version $MAJOR.$MINOR.$POINT of Apache at $HTTPD_PATH.

The Apache httpd executable at the location specified displays the incorrect "Server version."

During Validation of Web Server Root Directory

"Error: valid_http_root: Cannot find Apache configuration file at $CONF_FILE."

Must have valid Apache configuration file at conf/httpd.conf relative to the Server Root directory specified. Note that the default installation of Apache on SuSE does not place the

httpd.conf configuration file in a subdirectory of ServerRoot called conf. If you are using the default installation of Apache on SuSE, you must relocate the configuration file to the directory /etc/httpd/conf.

"Error: valid_http_root: Must specify absolute path to Apache server root directory."

Enter full, absolute path name to a Server Root directory.

"Error: valid_http_root: Apache instance at $HTTP_ROOT is already under LifeKeeper protection."

Each instance must have its own, unique Server Root directory, with configuration file located at conf/httpd.conf. The Server Root directory specified is already being used by another Apache instance.

"Syntax error on line <line number> of <configuration file path>, etc…"

Syntax error(s) were found in the Apache configuration file. These error messages were displayed by the httpd -T command when used to check the syntax of $CONF_FILE. See the error messages displayed for details.

"Error: valid_http_root: Since $HTTPD_PATH is shareable on $HTTPD_PATH_SHARED, $HTTP_ROOT must be also."

If the httpd executable is on shared/shareable storage, the Server Root and all DocumentRoot directories must be also.

"Error: valid_http_root: Since $HTTP_ROOT is shareable on $HTTP_ROOT_SHARED, all document root directories must shareable on this same filesystem."

If the Server Root is on shared/shareable storage, all DocumentRoot directories must be also.

"Error: http_docs_shared: Since one/more Apache document root directories

are shareable on $docs_shared, $curr_root must be also.

If any DocumentRoot directories are on a shared/shareable file system, all DocumentRoot directories must be located on the same file system.

"Error: valid_http_root: Must include BindAddress or Listen directives for each Apache instance. Check the Apache configuration file at $CONF_FILE."

In order to run multiple instances of Apache, each configuration file must contain BindAddress or Listen directives. Please refer to the Configuration Considerations for Apache Web Server section earlier in this guide for further detail.

"Error: valid_http_root: Default IP address * not allowed for LifeKeeper protection. Check the Apache configuration file at $CONF_FILE."

You must specify at least one specific LifeKeeper protected IP address for each Apache instance.

"Error: valid_http_root: A Listen directive is being used which specifies an IP address but no port. Check the Apache configuration file at $CONF_FILE."

The correct syntax for the Listen directive is Listen [IPaddress:] port number. This is not caught as a syntax error by Apache, but is interpreted incorrectly (as though the first number in the IP address was a port number specification).

"Error: valid_http_root: IP address $ip is not LifeKeeper protected."

The Apache configuration file refers to an IP address or domain name not configured under LifeKeeper protection. You must create these LifeKeeper IP address resources in advance.

During Apache Resource Hierarchy Creation

"Error: Could not find IP resource for $IP_ADD on machine $MACH."

You must create this resource before the Apache resource creation will succeed.

"Error: Create Apache file system hierarchy failure for filesystem $FSNAME used by server root $HTTP_ROOT."

"Error: Failure bringing Apache Resource $TAG into service on machine $MACH."

Check the Apache error logs for messages (default location is /var/log/httpd/error_log, but other logs may be listed).

The most likely cause of this problem is an error in the Apache configuration file. You may be able to bring this resource into service manually after correcting the problem.

"LifeKeeper: RESTORE: *ERROR* Apache: The instance is Password Protected."

The LifeKeeper Apache Web Server Recovery Kit cannot support password protected Private Key files for SSL-enabled web servers, since this would require manual interaction each time Apache starts up, and would prevent automatic restart and failover. The section Specific Configuration Considerations for Apache Web Server in this document explains how to remove password protection from the Private Key file (specified by the SSLCertificateKeyFile directive). This message applies only in an environment where the SSL module is used with Apache.

6.1.5.2. Apache Extend Hierarchy Errors

The error messages that might be displayed during Apache hierarchy extension are listed below, along with a suggested explanation for each. Note that these error messages appear when the GUI indicates it is “Executing the pre-extend script….” to validate the hierarchy prior to extending it to the new system.

Each will be preceded by an error message like:

“Error – canextend(template_server, tag, app_type/resource_type, target_server) -”.

Each will be followed by an error message like:

“Error – extmgr(template_server, tag, target_tag, target_server) -”.

During Validation of Web Server Binary Location

See errors listed for validation of Web Server Binary Location under Hierarchy Creation Errors.

During Validation of the Apache Configuration File on the Target System

“Cannot find Apache configuration file at $CONF_FILE on $TARGET_SYS.”

Must have a valid Apache configuration file at conf/httpd.conf relative to Server Root directory specified.

“DocumentRoot directory “$doc” in $CONF_FILE on $TARGET_SYS was not found in the configuration file on $TEMPLATE_SYS.”

or

“DocumentRoot directory “$doc” in $CONF_FILE on $TEMPLATE_SYS was not found in the configuration file on $TARGET_SYS.”

While comparing the configuration files on target and template servers, one or more DocumentRoot directories were found which do not match between the two. Check the details of the error messages displayed to determine the differences between the two. Note that if a DocumentRoot directory path is typed incorrectly, you will generally see both of these error messages, since each configuration file will appear to have an entry not in the other file.

“IP:port combination “$ipp” in $CONF_FILE on $TARGET_SYS was not found in the configuration file on $TEMPLATE_SYS.”

or

“IP:port combination “$ipp” in $CONF_FILE on $TEMPLATE_SYS was not found in the configuration file on $TARGET_SYS.”

While comparing the configuration files on target and template servers, one or more IP/port combinations were configured for use on one server but not on the other. Note that the IP/port combinations used may be specified in terms of IP addresses, ports, and domain names using a variety of Apache configuration directives. It is the actual IP/port combinations used which are compared, not the directives used to specify them. Check the details of the error messages displayed to determine the differences between the two.

“SSLCertificateKeyFile “$file” in $CONF_FILE on $SYS1 was not found in the configuration file on $SYS2.”

The filename specified for the SSLCertificateKeyFile in the Apache configuration file on the target system does not match the one specified on the template system. These configurations must be identical. This message applies only in an environment where the SSL module is used with Apache.

“Apache SSLCertificateKeyFile exists on $SYS1 but not on $SYS2.”

The SSLCertificateKeyFile specified in the Apache configuration files exists on one system, but not on the other. The file must be present on both nodes. This message applies only in an environment where the SSL module is used with Apache.

“WARNING: PHP configuration file $PHP_CONFIG appears to be different on $SYS1 and $SYS2.”

The configuration file for the PHP module on the target system is not identical to the one on the template system. Inspect the configuration on both servers to ensure that they are the same. This message applies only in an environment where the PHP module is used with Apache.

During Apache Resource Hierarchy Creation on Target Server

See errors listed for Apache resource hierarchy creation under Hierarchy Creation Errors.

6.1.5.3. Apache Hierarchy Restore, Remove, and Recover Messages and Errors

The following information and error messages are printed to the LifeKeeper error log.

They may be viewed by typing “lk_log log”.

Bringing an Apache Resource In Service (Restore)

“LifeKeeper: RESTORE: APACHE: RESTORING $TAG TO SERVICE START AT: <date>”

Informational message. Records when the restore begins. Logged at the start of every restore.

“LifeKeeper: RESTORE APACHE RESOURCE $TAG END err=$err AT: <date>”

Informational message. Records when the restore completes. Logged at the end of every restore. If any errors occur during the restore, additional messages will be logged between these two messages and the value displayed for err=$err will be non-zero.

“Apache: No instance information found for Tag=$TAG.”

Error: Indicates no instance is defined with the tag value passed to the “restore” script. Unlikely to occur with the GUI, since only tags known to LifeKeeper are available as choices for the In Service and Out of Service actions.

“LifeKeeper: RESTORE: Apache: Tag=$TAG already running.”

Informational message. Indicates that the instance appeared to already be up and running.

“LifeKeeper: RESTORE: Apache: Existing processes terminated for ID=$ID.”

Informational message. Existing httpd processes were found running for this instance ID, but the PidFile is either missing or invalid. Therefore, the running processes were terminated.

“LifeKeeper: RESTORE: Apache: Invalid PidFile=$PIDFILE has been deleted.”

Informational message. An existing PidFile was found for this instance, but its contents were invalid. Therefore, the PidFile was deleted.

“LifeKeeper: RESTORE: Apache: Tag=$TAG is being restarted.”

Informational message. Indicates that the instance appeared to partially running, but needed to be restarted. If a PidFile still exists (which contains the process ID of the parent httpd process), the instance is restarted with a HUP signal. If the PidFile is missing, the instance is completely stopped and restarted.

*“LifeKeeper: RESTORE: *ERROR* Apache: Error in web server configuration file $CONF_FILE for instance $ID.”*

*“LifeKeeper: RESTORE: *ERROR* Apache: Execute the following command to check the syntax of this file:”*

*“LifeKeeper: RESTORE: *ERROR* Apache: $HTTPD_PATH -t -d $SERVER_ROOT -f $CONF_FILE.”*

Prior to instance startup, the syntax of the configuration file is checked using the httpd -t option. The -d option checks the ServerRoot directory. Additional options related to modules may also be displayed if you have configured Apache to use modules. Any syntax errors caught during hierarchy creation are displayed in the LifeKeeper GUI, but syntax errors introduced later will not be displayed in the GUI or the LifeKeeper logs. You must manually run the following command to determine what is wrong with your configuration (add additional options for modules, if applicable):

$HTTPD_PATH -t -d $SERVER_ROOT -f $CONF_FILE

*“LifeKeeper: RESTORE: *ERROR* Apache: Error starting web server instance $INSTANCE.”*

*“LifeKeeper: RESTORE: *ERROR* Apache: Restore of tag $TAG failed.”*

*“LifeKeeper: RESTORE: *ERROR* Apache: Examine the Apache error log at $ERROR_LOG”*

*“LifeKeeper: RESTORE: *ERROR* Apache: to determine the cause of the problem.”*

An error occurred executing the httpd daemon with the parameters specified. Check the httpd executable being used, configuration file, and general configuration for possible problems.

*“LifeKeeper: RESTORE: *ERROR* Apache: Web server instance $ID did not start correctly.”*

*“LifeKeeper: RESTORE: *ERROR* Apache: Restore of tag $TAG failed.”*

*“LifeKeeper: RESTORE: *ERROR* Apache: Examine the Apache error log at $ERROR_LOG”*

*“LifeKeeper: RESTORE: *ERROR* Apache: to determine the cause of the problem.”*

Note that in many cases the httpd daemon will appear to start even if its web sites don’t respond as expected. The restore script checks all IP/port combinations used to make sure all sites configured are fully functional. If they are not, this message is printed and the restore fails.

Although the site is left in the Out of Service state, one/more httpd processes may still be left running. (This is intentional, since one/more web sites may be operational and we don’t want to kill them off). You should resolve the problem as soon as possible and bring the instance In Service. If you don’t, LifeKeeper will eventually attempt to recover the instance and restore it to service automatically. If it can’t, it will fail over the hierarchy to another server.

Taking an Apache Resource Out of Service (Remove)

“LifeKeeper: REMOVE: APACHE: REMOVE $TAG FROM SERVICE START AT: <date>”

Informational message. Records when the remove begins. Logged at the start of every remove.

“LifeKeeper: REMOVE APACHE RESOURCE $TAG END err=$err AT: <date>”

Informational message. Records when the remove completes. Logged at the end of every remove.

If any errors occur during the remove, additional messages will be logged between these two messages and the value displayed for err=$err will be non-zero.

“LifeKeeper: REMOVE: *WARNING* APACHE: Error attempting to kill parent process for INSTANCE=$INSTANCE.”

There was an error attempting to kill the parent httpd process (whose process ID is stored in the Pidfile).

*“LifeKeeper: REMOVE: *ERROR* APACHE: Error attempting to kill all processes for INSTANCE=$INSTANCE.”*

Although the parent httpd process appeared to be killed successfully, one/more processes for this instance are still running. Normally the remove will be able to terminate any/all processes for this

instance. When it cannot, this message is printed and the remove fails.

Bringing an Apache Resource Back In Service (Recover)

The LifeKeeper core periodically checks the health of every Apache instance In Service on the local server by running an Apache “quickCheck” script, which checks the web sites using the same scripts used to check the state of the instance during restore and remove. If the instance is not fully functional, a “recover” script is invoked to attempt to restart the instance. This simply logs the first message shown below, invokes “restore,” prints the final error or success message shown below—depending on error or success of the “restore” script—and returns the same result as “restore.” If restore/recover fails, this instance is failed over to another server.

“LifeKeeper: RECOVER: APACHE: Invoking restore for Apache instance “$ID” at: <date>”

“LifeKeeper: RECOVER: APACHE: Restore for Apache instance $ID returned error $RET at: <date>”

“LifeKeeper: RECOVER: APACHE: Restore for Apache instance $ID successful at: <date>”

6.2. DB2 Recovery Kit Administration Guide

The SPS for Linux DB2 Recovery Kit provides fault resilient protection for DB2 database instances. LifeKeeper, together with the DB2 Universal Database product family afford increased availability to DB2 operating environments by effectively recovering database server failures without significant down-time or human intervention.

Document Contents

This guide contains the following topics:

- Documentation and References. A list of LifeKeeper for Linux documentation and where to find them.
- Requirements. A description of the hardware and software necessary to properly setup, install, and operate the DB2 Recovery Kit. Refer to SIOS Protection Suite Installation Guide for specific instructions on how to install or remove LifeKeeper for Linux software.DB2 Recovery Kit .
- Overview. A description of the DB2 Recovery Kit's features and functionality.
- Configuring the LifeKeeper for Linux DB2 Recovery Kit. A description of the procedures required to properly configure the DB2 Recovery Kit.
- LifeKeeper Configuration Tasks. A description of the tasks for creating and managing your DB2 resource hierarchies using the LifeKeeper GUI.
- Troubleshooting. A list of LifeKeeper for Linux error messages including a description for each.
- Appendix. Steps for setting up DB2 to use raw I/O.

6.2.1. DB2 Documentation and References

The following LifeKeeper product documentation is available from SIOS Technology Corp.:

- ◦ SPS for Linux Release Notes
- ◦ SPS for Linux Technical Documentation
- ◦ SIOS Protection Suite Installation Guide
- ◦ Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is available on the SIOS Technology Corp. website at:

http://docs.us.sios.com

6.2.2. DB2 Recovery Kit Hardware and Software Requirements

Your LifeKeeper configuration must meet the following requirements **prior** to the installation of the LifeKeeper for Linux DB2 Recovery Kit. Please see SIOS Protection Suite Installation Guide for specific instructions regarding the configuration of your LifeKeeper hardware and software.

Hardware Requirements

- **Servers** – LifeKeeper for Linux supported servers configured in accordance with the requirements described in SIOS Protection Suite Installation Guide and SPS for Linux Release Notes.
- **IP Network Interface Cards** – Each server requires at least one Ethernet TCP/IP-supported network interface card. Remember, however, that a LifeKeeper cluster requires two communications paths; two separate LAN-based communication paths using dual independent sub-nets are recommended for heartbeats, and at least one of these should be configured as a private network. Using a combination of TCP and TTY heartbeats is also supported.

Software Requirements

- **TCP/IP software** – Each server in your LifeKeeper configuration requires TCP/IP software.
- **IBM software** – Please refer to SPS for Linux Release Notes for specific DB2 version requirements on certain Linux distributions and hardware architectures.
- **LifeKeeper Software** – It is imperative that you install the same version of the LifeKeeper software and apply the same versions of the LifeKeeper software patches to each server in your cluster.
- **LifeKeeper for Linux DB2 Recovery Kit** – The DB2 Recovery Kit is provided on a CD. It is packaged, installed and removed via the Red Hat Package Manager, rpm. The following rpm file is supplied on the LifeKeeper for Linux DB2 Recovery Kit CD:

  steeleye-lkDB2

  Please see SIOS Protection Suite Installation Guide for specific instructions on the installation and removal of the LifeKeeper for Linux software.

- **LifeKeeper for Linux NFS Recovery Kit-required for use of DB2 EEE and multiple partition ESE deployments.** This recovery kit is provided on CD in the **steeleye-lkNFS** package.

Important: See Issues Regarding DB2 EEE or multiple partition ESE and NFS for important configuration information

6.2.3. DB2 Recovery Kit Overview

LifeKeeper for Linux DB2 Recovery Kit

In versions 8 and greater, DB2 UDB Enterprise Edition (EE) and Enterprise-Extended Edition (EEE) have been combined into a single product named DB2 UDB Enterprise Server Edition (ESE). Previous versions included two separate enterprise level database servers, the Enterprise Edition (EE) as a standard relational database management system and the Enterprise-Extended Edition (EEE) as an extension of the EE database server to support multi-partition databases.

The LifeKeeper for Linux DB2 Recovery Kit provides protection for the database manager in the EE, WE, and WSE environments, and for the database partition servers in an EEE environment. In a combined ESE environment, the recovery kit provides protection for both the database manager and the database partition servers.

Users may elect to define the DB2 Administration Server for each machine within the LifeKeeper cluster. When the DB2 Administration server is defined, LifeKeeper will attempt to start the DB2 Administration Server as a function of the DB2 hierarchy create and the DB2 hierarchy restore operations.

6.2.4. Configuring the LifeKeeper for Linux DB2 Recovery Kit

This section describes the LifeKeeper for Linux DB2 Recovery Kit configuration details. It also contains information you should consider before you start to configure and administer the DB2 Recovery Kit. Please refer to SPS for Linux Technical Documentation for instructions on configuring LifeKeeper Core resource hierarchies.

6.2.4.1. Using DB2 with Raw I/O

If you plan to use DB2 with Raw I/O devices, you must install the LifeKeeper Raw I/O Recovery Kit from the LifeKeeper Core CD. You must also properly set up the Raw I/O devices prior to use. See the Appendix for instructions.

6.2.4.2. Running DB2

Reducing the DB2 Process Startup Times

In some instances the startup times of the DB2 processes can be excessive when using DB2 8.x under LifeKeeper protection. Making the following change to the kernel network parameters can improve this situation. Add the following line to the_ /etc/sysctl.conf_ file on each LifeKeeper clustered system that will be running DB2 8.x:

```
net.ipv4.tcp_syn_retries=1
```

Then running **sysctl –p** will cause this change to take effect.

Preventing Frequent DB2 Instance Crashes (Panic)

If a LifeKeeper protected DB2 instance is encountering frequent crashes in a systemd environment (RHEL7, CentOS7, OEL7) then altering the automatic IPC cleanup configuration parameter may correct this issue. On each node in the LifeKeeper cluster, set the following configuration parameter in the /etc/systemd/logind.conf file.

```
RemoveIPC=no
```

Then, execute **systemctl restart systemd-logind** to make this change effective. Click here for more details.

6.2.4.3. Configuration Considerations for DB2 Single Partition

The following should be considered before operating the LifeKeeper for Linux DB2 Recovery Kit in the single partition or workgroup environment:

1. LifeKeeper requires the location of the DB2 instance home directory as well as associated databases, tablespaces, and resources be stored on shared drives. The shared drives are automatically protected at the time the hierarchy is created. During creation of the DB2 resource hierarchy, the DB2 database manager is created as the parent resource while the shared file systems containing instance home directories and actual databases are created as dependent resources. Consequently, if **after** the creation of your DB2 hierarchy you decide to create a database on a shared file system that is not protected by LifeKeeper, you will need to create a resource hierarchy for that file system and make it a dependency of your DB2 resource hierarchy.

2. When the database manager becomes inoperable on the primary system, the service fails over to a previously defined backup system. The database service on the backup system becomes available immediately after the dependent resources fail over and the database manager is brought into service. Previously connected DB2 clients are disconnected and must reconnect to the functioning server. Any uncommitted SQL statements are rolled back and should be re-entered.

6.2.4.4. Configuration Considerations for DB2 Multiple Partition

DB2 Multiple Partition RESTRICTIONS: All DB2 multiple database partition servers will be protected on a particular machine when the LifeKeeper DB2 resource hierarchy is created on that machine. The nodes to protect are determined by examining the following file:

<instance home>/sqllib/db2nodes.cfg

Future plans for this recovery kit include added functionality to allow for N-way failover.

6.2.4.4.1. Issues Regarding DB2 EEE or multiple partition ESE and NFS

If the NFS export point for the DB2 instance home directory becomes unavailable while the DB2 instances are running, the system will hang while waiting for the export point to become available again. Many system operations will not work correctly, including a system reboot. You should be aware that the NFS server for the DB2 multiple partitions cluster should be protected by LifeKeeper and should not be manually taken out of service unless all the partitions in the DB2 cluster are also taken out of service before shutting down the NFS resource. Additionally, the DB2 partitions cannot be brought into service unless the NFS resource is in service.

To avoid accidentally causing your cluster to hang by inadvertently stopping the NFS server, we make the following recommendations:

NFS Recommendations

Use additional servers: It is highly recommended that you have a separate cluster for the NFS export point from which the DB2 instance home is mounted. The NFS export point on this cluster should be protected with the LifeKeeper NFS Server Recovery Kit.

If you do not have at least two additional servers available, you can reduce the chances of experiencing the problem described above by adding one additional server to the DB2 cluster. This additional server would export the NFS hierarchy. One of the other nodes in the cluster would serve as a backup. In this configuration the symptoms could occur if the NFS hierarchy were to failover to the backup node. The NFS export point on this cluster should be protected with the LifeKeeper NFS Server Recovery Kit.

If you cannot use additional servers: This is the least desirable option. However, if you decide to run your NFS server in the same cluster as your DB2 multiple partitions, the NFS export point should be protected with the LifeKeeper NFS Server Recovery Kit. You should note that LifeKeeper currently is not aware of the relationship between the DB2 partitions and the NFS server managing the DB2 partitions. Therefore, you must follow these manual procedures before stopping or starting LifeKeeper on any node in the cluster.

1. If you wish to stop LifeKeeper on a single server, you must make sure that the NFS server is active on another server in the cluster. Failure to do this may cause the LifeKeeper shutdown to hang trying to take the DB2 partitions out of service. Generally, you should make sure that all DB2 partitions are either switched to another server or manually taken out of service before you stop LifeKeeper to ensure you don’t have problems trying to restart LifeKeeper.

2. To shut down the entire cluster, you should manually take all DB2 partition resources out of service. Next, take all the DB2 NFS server resources out of service, and finally shut down LifeKeeper.

3. If you remembered to take the DB2 resource out of service before shutting down LifeKeeper, you should be able to restart LifeKeeper normally. Then bring the NFS server resources into service,

followed by any DB2 partitions you wish to restart.

4. If you forgot to take the DB2 partition out of service before shutting down LifeKeeper, you must make sure that the NFS server resources for that partition are active elsewhere in the cluster before you restart LifeKeeper.

6.2.4.4.2. DB2 Configuration Requirements

To ensure proper operation of the DB2 Recovery Kit in a multiple partition environment, LifeKeeper requires the following:

1. If you cannot use an additional cluster for your NFS hierarchy, be aware that the LifeKeeper for Linux DB2 Recovery Kit restricts the occurrence of active inodes on an underlying NFS- protected file system. Therefore, to prevent this condition, we recommend that users protect the top-level directory and export the instance home directory using the fully qualified directory name. The top-level directory is protected in order to prohibit users from changing directories directly into it (i.e. *cd<top level dir>*).

2. Verify the installation of IBM's latest Fix Pack (for EEE deployments) as described in the Software Requirements section of this document.

3. Ensure that the hostname value in your *db2nodes.cfg* file is the same as the value returned from issuing the **hostname** command.

   Example:

   db2nodes.cfg file:

   0 server1.sc.steeleye.com 0

   Additionally, the hostname value in your server's */etc/hosts* file must be the same as the hostname value in your *db2nodes.cfg* file.

   You must also verify that your server's */etc/hosts* file contains both the local hostname and the fully qualified hostname for each server entry included in the file.

   Example:

   /etc/hosts file

   ```
   127.0.0.1 localhost localhost.localdomain

   9.21.55.53 server1.sc.steeleye.com server1
   ```

4. During execution of the db2setup script, **do not** opt to create the DB2 Warehouse Control Database (DWCNTRL) or the DB2 Sample Database at this time. The databases need to be created on a shared file system to ensure successful creation of the DB2 resource hierarchy. Electing to create either of these databases during execution of the db2setup script will cause the database to be created in the home directory and not on a shared file system. Users wishing to create these databases should do so external to the db2setup script in order to specify a shared file system.

In versions later than 8.1, the DB2 Tools Catalog should not be created during the setup script. This database must be placed on a shared file system and should be created after setup has completed and prior to hierarchy creation, if necessary.

5. Active/Active or multiple partition server environments, each server in the configuration must be capable of running all database instances in a failover scenario. Please see the *IBM Getting Started Guide* for help determining the maximum number of DB2 instances or partition servers feasible for a given set of system resources.

6. Select or create a shared file system, then export this file system. (*i.e /export/db2home*). The file system will be used as the DB2 instance home.

7. Protect your exported file system by creating a LifeKeeper NFS resource hierarchy. The file system should be included as a dependent resource in your NFS hierarchy.

8. NFS mount the shared file system on each server in the cluster including the server where it is being exported. See the *DB2 Quickstart Guide* for mount options. When creating the DB2 instance, the home directory of the instance must be located on the NFS mounted file system. Make certain that the file system is mounted using the LifeKeeper protected switchable IP address used when creating the NFS hierarchy. Additionally, the mount point of the home directory must be specified in the */etc/fstab file* on all servers in your LifeKeeper cluster. Each server in your configuration must have the file system mounted on identical mount points (i.e. */db2/home*).

   Note: We recommend that you create and test your NFS hierarchy prior to creating your DB2 resource hierarchy. Please see the NFS Recovery Kit Administration Guide for complete instructions on creating and testing a NFS hierarchy.

9. For all servers in your configuration, set the following DB2 environment variable to equal the total number of partitions in the instance. To set this variable, log on to the server as the instance owner and issue a **db2set** command. Adjusting this variable will accommodate all conceivable failover scenarios.

   db2setDB2_NUM_FAILOVER_NODES=<partitions in the instance>

10. Update your existing DB2 instances and your DB2 Administration servers using the following DB2 utilities:

    db2iupdt and dasiupdt

11. A LifeKeeper DB2 hierarchy must be created on each server in the cluster that has a database partition server managing data for the instance. The databases and tablespaces must be on a shared file system. A separate LUN is required for each database partition server and for the NFS exported home directory. Dependent resources include the file systems where actual databases and tablespaces are located.

12. If you create a database on a non-protected LifeKeeper file system after the creation of your DB2

hierarchy, you will need to create a resource hierarchy for that file system and make it a dependency of your DB2 resource hierarchy. The hierarchy will protect all of the partition servers that the *db2node.cfg* file indicates should run on the server.

13. To ensure proper execution of a failover, it is imperative that the file system of each database partition server is uniquely numbered.

    Example:

    The mount point for your database partition server *node0* should be:

    /<FSROOT>/<db2instancename>/NODE0000

    The mount point for your database partition server *node1* should be:

    /<FSROOT>/<db2instancename>/NODE0001

    Note: In this example there are two partition servers, and the file system for each is mounted on a separate LUN.

14. All database partition servers for a given machine must be running in order to assure the successful creation of your DB2 hierarchy.

15. When the database partition server becomes inoperable on the primary system, the service fails over to a previously defined backup system. The database service on the backup system becomes available immediately after the dependent resources fail over and the database partition server(s) is brought into service. Previously connected DB2 clients are disconnected and must reconnect to the functioning server. Any uncommitted SQL statements are rolled back and should be re-entered.

6.2.4.5. Configuration Considerations for All DB2 Configurations

1. DB2 instance names should contain alphanumeric characters only.

2. DB2 clients should be configured to connect to the database via a LifeKeeper protected IP address. Users can define:

   DB2SYSTEM=<Floating IP>" in $instancehome/sqllib/profile.env

   and catalog the floating IP address on the clients.

3. The */etc/services* file for each server in your configuration protecting a DB2 resource hierarchy must have identical service entries for the protected instance. Additionally, the User ID, Group ID and instance home directory for the protected DB2 instance must be the same on all servers where the resource will be protected.

   DB2 adds the following entries as default in */etc/services*:

   DB2_db2inst1 60000/tcp

   DB2_db2inst1_1 60001/tcp

   DB2_db2inst1_2 60002/tcp

   DB2_db2inst1_END 60003/tcp

   db2c_db2inst1 50001/tcp

4. A recovery is what takes place after DB2 is terminated abruptly, as with a system crash. Following are tips that will significantly reduce the amount of time it takes for DB2 to recover from a failure.

   - º Limit the log records that DB2 will process. You can accomplish this by properly configuring the

SOFTMAX and **LOGFILSIZ** configuration parameters. You should use log files with a size of 4MB (1000 4KB pages) and keep the amount of active log space at 25% of the size of one log file (1MB):

db2 UPDATE DB CFG FOR <dbname> USING SOFTMAX 25

db2 UPDATE DB CFG FOR <dbname> USING LOGFILSIZ 1000

º Ensure that there is a sufficient number of page cleaners to accommodate your work load:

db2 UPDATE DB CFG FOR <dbname> USING NUM_IOCLEANERS <num>

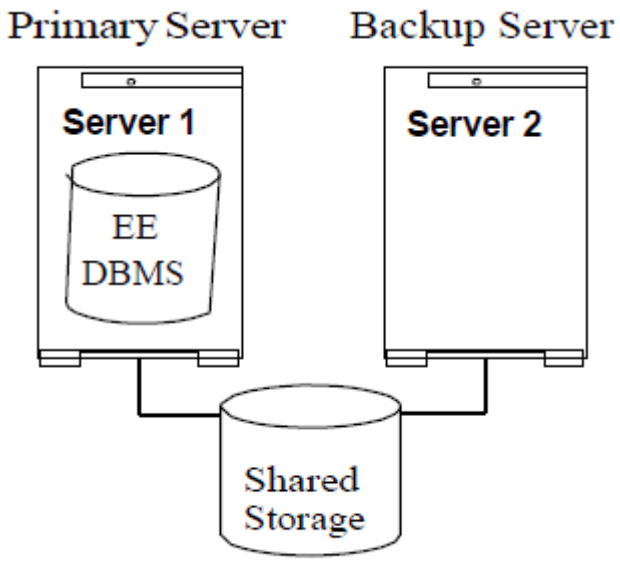
5. DB2 Fault Monitor should be disabled in all servers.

6. DB2 should be installed in all servers.

6.2.4.6. DB2 Configuration Examples

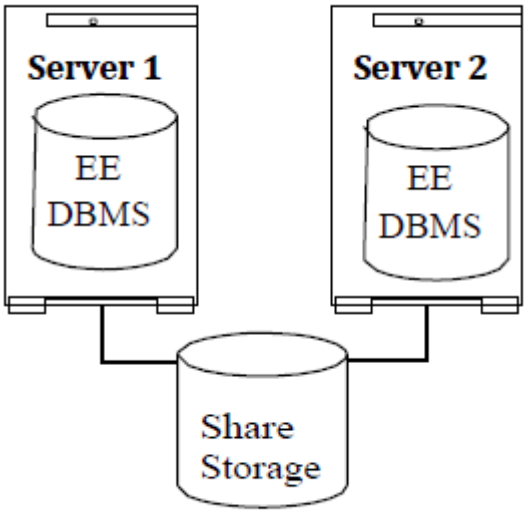
A few examples of what happens during a failover using LifeKeeper for Linux DB2 Recovery Kit are provided below. In the following pictures, EE and EEE are used to denote database configurations; ESE may be substituted wherever appropriate.

Configuration 1: DB2 Single Partition Active/Standby Configuration



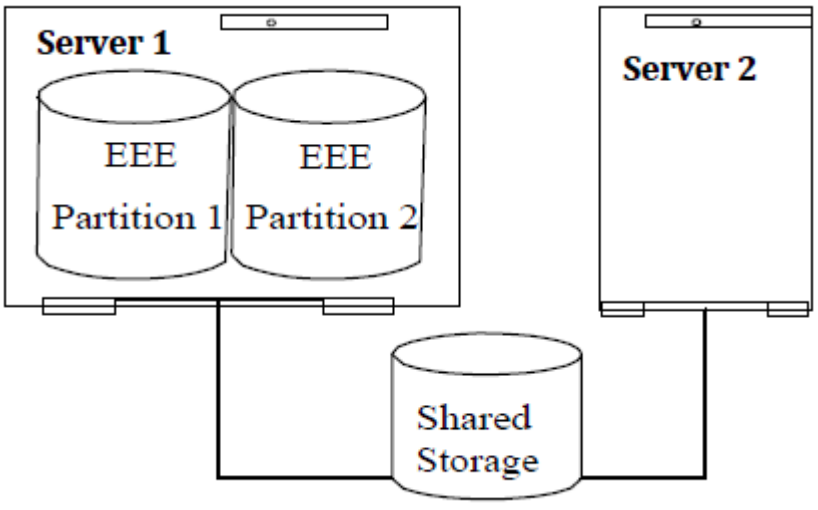
The DB2 instance is protected on Server 1. Server 2 will assume the DB2 resources when a failure occurs.

Configuration 2: DB2 Single Partition Active/Active Configuration



One DB2 instance is protected on Server 1 and another DB2 instance is protected on Server 2. Each server will assume the other's resources when a failure occurs.

Configuration 3: DB2 Multiple Partition Active/Standby (1 Cluster)

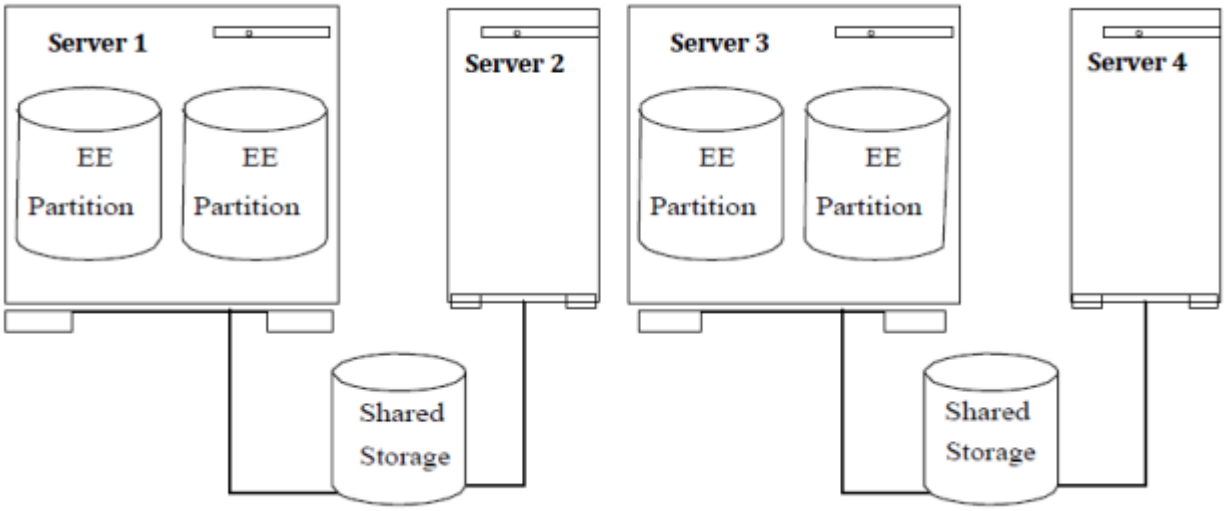


One DB2 instance with two database partition servers is protected on Server 1 with one LifeKeeper DB2 resource hierarchy. Server 2 will assume ownership of the DB2 resource hierarchy when a failure occurs.

Note: For all cluster of cluster configurations listed in the following section, users should be aware that the cluster of cluster configuration is protecting only one DB2 instance with multiple partitions on multiple

physical nodes.

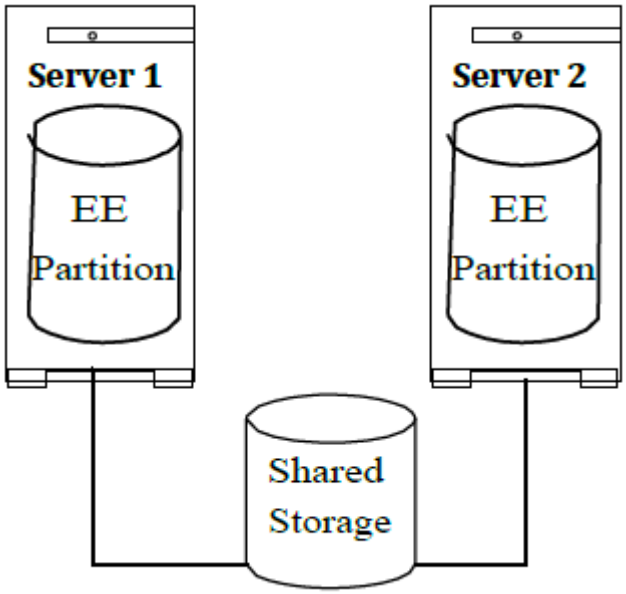
Configuration 4: DB2 Multiple Partition Active/Standby (Cluster of Clusters)



One DB2 instance with two database partition servers is protected on Server 1 and two database partition servers protected on Server 3. There is one LifeKeeper DB2 resource hierarchy on Server 1, extended to Server 2, and another DB2 resource hierarchy on Server 3 extended to Server 4. When a failure occurs on Server 1, Server 2 will assume its resource. When a failure occurs on Server 3, Server 4 will assume its resource.

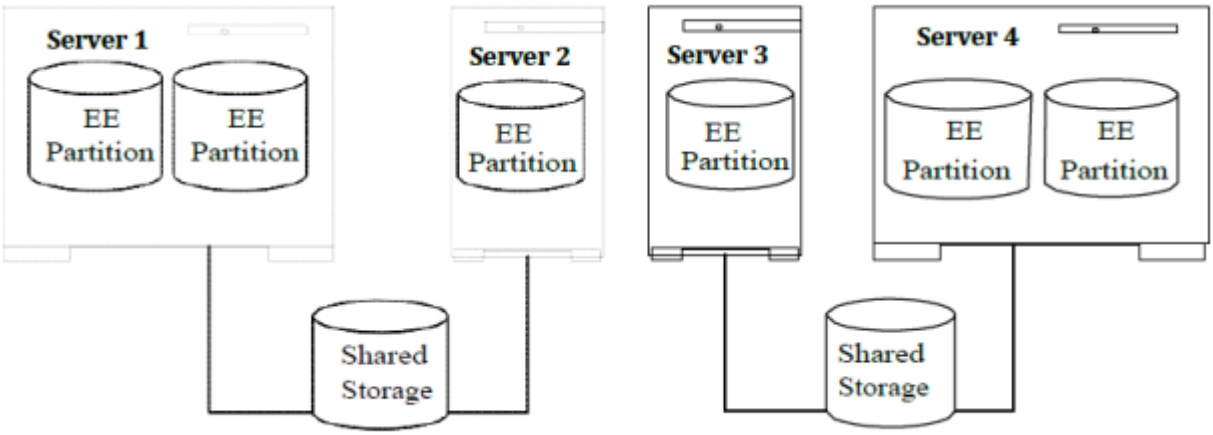
If the server that is exporting the DB2 instance home directory and its backup server become inoperable at once, the DB2 database is inaccessible. In addition, if the NFS hierarchy for the exported DB2 instance directory (primary and all backups) become inoperable at the same time, the DB2 database will be inaccessible until the NFS hierarchy can be restored.

Configuration 5: DB2 Multiple Partitions Active/Active (1 Cluster)



One DB2 instance with one database partition server is protected on Server 1 and one database partition server protected on Server 2. There is one LifeKeeper DB2 resource hierarchy on Server 1 and another DB2 resource hierarchy on Server 2. When a failure occurs each server will assume the other's resources.

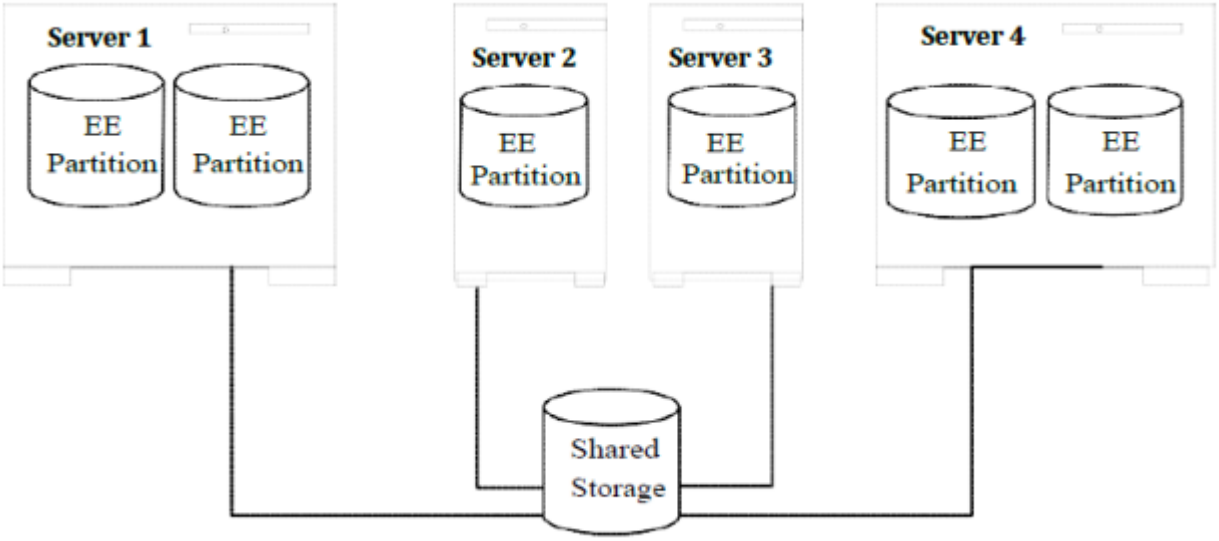
Configuration 6: DB2 Multiple Partitions Active/Active (Cluster of Clusters)



One DB2 instance with two database partition servers is protected on Server 1, one database partition server protected on Server 2, one database partition server protected on Server 3 and two database partition servers protected on Server 4. There is one LifeKeeper DB2 resource hierarchy on each server in the cluster. Upon failure, Server 1 and Server 2 assume each other's resources and Server 3 and Server 4 assume each other's resources.

If the server that is exporting the DB2 instance home directory and its backup server become inoperable at once, the DB2 database is inaccessible. In addition, if the NFS hierarchy for the exported DB2 instance directory (primary and all backups), become inoperable at the same time, the DB2 database will be inaccessible until the NFS hierarchy can be restored.

Configuration 7: DB2 Multiple Partition (4 Node Fibre Channel Cluster)



One DB2 instance with two database partition servers is protected on Server 1, one database partition server protected on Server 2, one database partition server protected on Server 3 and two database partition servers protected on Server 4. There is one LifeKeeper DB2 resource hierarchy on each server in the cluster. Each server in the cluster provides backup protection for the other in the event of failure.

If the server that is exporting the DB2 instance home directory and its backup server become inoperable at once, the DB2 database is inaccessible. In addition, if the NFS hierarchy for the exported DB2 instance directory (primary and all backups), become inoperable at the same time, the DB2 database will be inaccessible until the NFS hierarchy can be restored.

6.2.5. LifeKeeper for Linux DB2 Recovery Kit Configuration Tasks

You can perform all LifeKeeper for Linux DB2 Recovery Kit administrative tasks via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer, and monitor DB2 resources.

The following tasks are available for configuring the LifeKeeper for Linux DB2 Recovery Kit:

- Create a Resource Hierarchy – Creates a DB2 resource hierarchy.
- Delete a Resource Hierarchy – Deletes a DB2 resource hierarchy.
- Extend a Resource Hierarchy – Extends a DB2 resource hierarchy from the primary server to the backup server.
- Unextend a Resource Hierarchy – Unextends (removes) a DB2 resource hierarchy from a single server in the LifeKeeper cluster.
- Create Dependency – Creates a child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete Dependency – Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service – Activates a resource hierarchy.
- Out of Service – Deactivates a resource hierarchy.
- View / Edit Properties – View or edit the properties of a resource hierarchy.

6.2.5.1. Creating a DB2 Resource Hierarchy

Perform the following on your primary server:

1. Select **Edit > Server > Create Resource Hierarchy**.

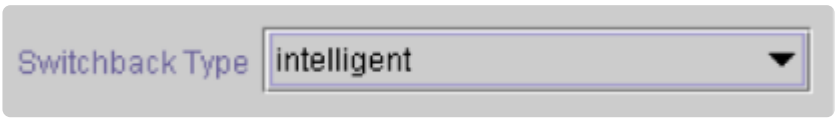
2. The **"Select Recovery Kit"** dialog appears. Select the **DB2 Database** option from the drop down list.

   Click **Next** to continue.

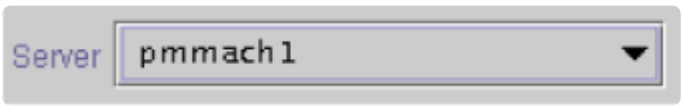
   CAUTION: If you click the **Cancel** button at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

3. The **"Switchback Type"** dialog appears. The switchback type determines how the DB2 resource will be switched back to the primary server when it becomes in-service (active) on the backup server following a failover. Switchback types are either *intelligent* or *automatic*. Intelligent switchback requires administrative intervention to switch the resource back to the primary server while automatic switchback occurs as soon as the primary server is back on line and reestablishes LifeKeeper communication paths.

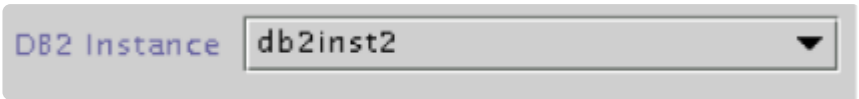
   Click **Next** to continue.

4. The **"Server"** dialog appears. Select the name of the server where the DB2 resource will be created (typically this is your primary server). All servers in your cluster are included in the drop down list box.

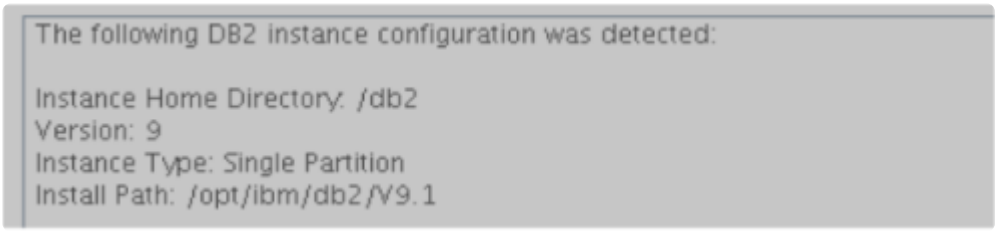
   Click **Next** to continue.

5. The **"DB2 Instance"** dialog appears. Select or enter the name of the **DB2** instance that is being protected.

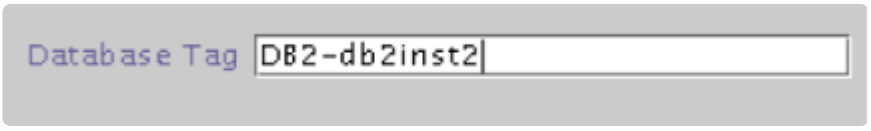
Click **Next** to continue.

6. An information box appears displaying information regarding the instance detected.



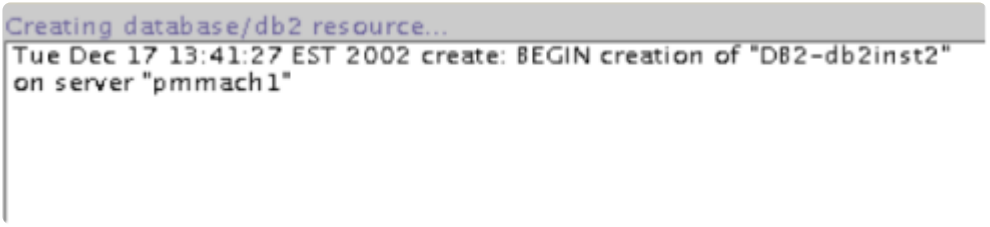
Click **Continue**.

7. The **"Database Tag"** dialog appears. This dialog is populated automatically with a unique tag name for the new DB2 database resource instance.



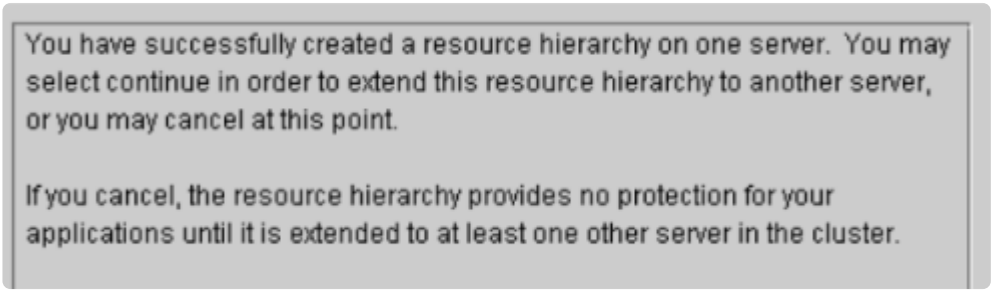
Click **Create** to continue.

8. An information box appears indicating the start of the hierarchy creation.



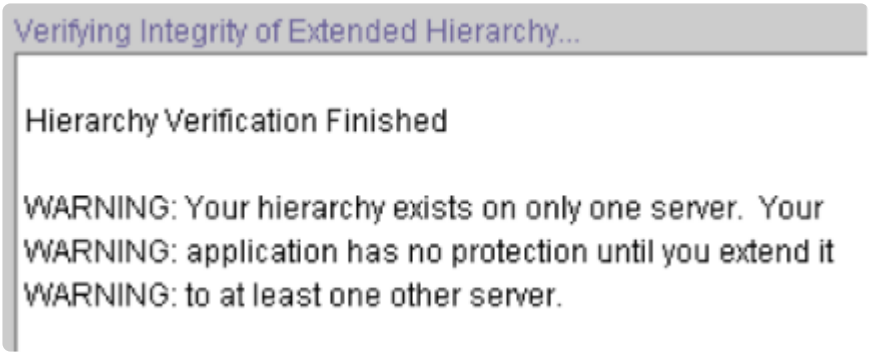
Click **Next** to continue.

9. An information box appears announcing the successful creation of your DB2 resource hierarchy. You must **Extend** the hierarchy to another server in your cluster in order to place it under LifeKeeper protection.



Click **Continue** to extend the resource.

Click **Cancel** if you wish to extend your resource at another time.



10. Click **Done** to exit the Create Resource Hierarchy menu selection.

6.2.5.2. Deleting a DB2 Resource Hierarchy

To delete a DB2 resource from all servers in your LifeKeeper configuration, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Delete Resource Hierarchy.**

2. Select the name of the **Target Server** where you will be deleting your DB2 resource hierarchy.

   Note: If you selected the Delete Resource task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next** to continue.

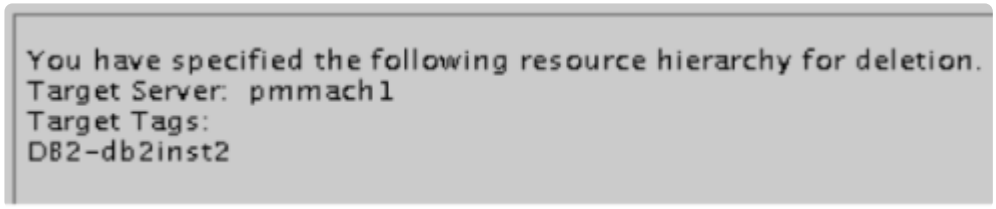
3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete, and highlight it.

   Note: If you selected the Delete Resource task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

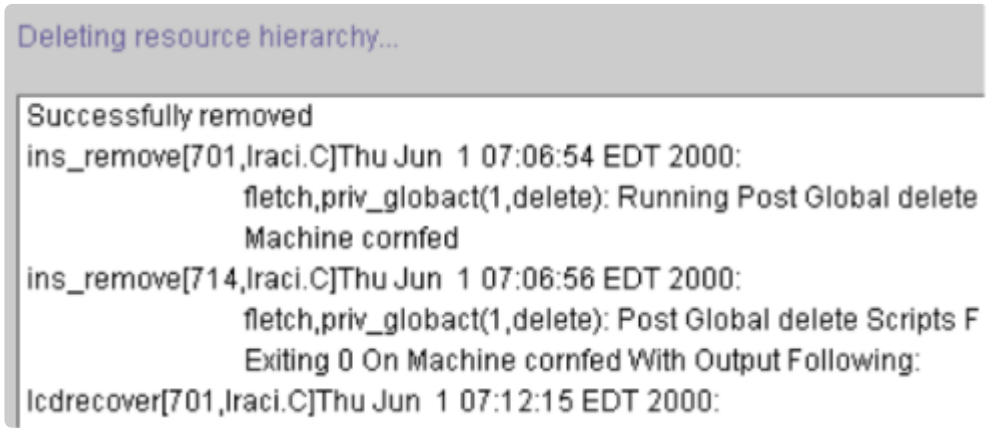
   Click **Next** to continue.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete.



Click **Delete** to continue.

5. An information box appears confirming that the DB2 resource instance was deleted successfully.



6. Click **Done** to exit the Delete Resource Hierarchy menu selection.

6.2.5.3. Extending Your DB2 Resource Hierarchy

After you have created a hierarchy, you should extend that hierarchy to another server in the cluster. There are three possible ways to extend your resource instance:

1. When you successfully create your DB2 resource hierarchy you will have an opportunity to select **Continue** which will allow you to proceed with extending your resource hierarchy to your backup server.

2. Right-click on an unextended hierarchy in either the left or right pane on the LifeKeeper GUI.

3. Select the **"Extend Resource Hierarchy"** task from the LifeKeeper GUI by selecting **Edit, Resource, Extend Resource Hierarchy** from the drop down menu. This sequence of selections will launch the Extend Resource Hierarchy wizard. The **Accept Defaults** button that is available for the **Extend Resource Hierarchy** option is intended for the user who is familiar with the LifeKeeper Extend Resource Hierarchy defaults and wants to quickly extend a LifeKeeper resource hierarchy without being prompted for input or confirmation. Users who prefer to extend a LifeKeeper resource hierarchy using the interactive, step-by- step interface of the GUI dialogs should use the **Next** button.

   a. The first dialog box to appear will ask you to select the **Template Server** where your DB2 resource hierarchy is currently in service. Remember that the **Template Server** you select now and the **Tag to Extend** that you select in the next dialog box represent an in- service (activated) resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you have selected. The drop down box in this dialog provides the names of all the servers in your cluster.

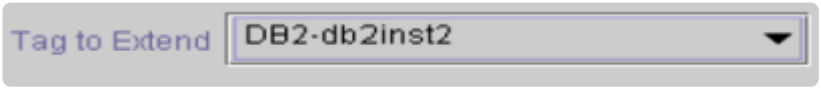
   Note: If you are entering the Extend Resource Hierarchy task by continuing from the creation of a DB2 resource hierarchy, this dialog box will not appear because the wizard has already identified the template server in the create stage. This is also the case when you right-click on either the DB2 resource icon in the left pane or right-click on the DB2 resource box in the right pane of the GUI window and choose Extend Resource Hierarchy.

   CAUTION: If you click the **Cancel** button at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extend hierarchy process. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

   Click **Next** to continue.

b. Select the **Tag to Extend**. This is the name of the DB2 instance you wish to extend from the template server to the target server. The wizard will list in the drop down box all of the resources that you have created on the template server.

Note: Once again, if you are entering the Extend Resource Hierarchy task immediately following the creation of a DB2 hierarchy, this dialog box will not appear because the wizard has already identified the tag name of your resource in the create stage. This is also the case when you right-click on either the DB2 resource icon in the left pane or on the DB2 resource box in the right pane of the GUI window and choose *Extend Resource Hierarchy*.



Click **Next** to continue.

c. Select the **Target Server** where you will extend your DB2 resource hierarchy.

Click **Next** to continue.

d. The **Switchback Type** dialog appears. The switchback type determines how the DB2 resource will be switched back to the primary server when it becomes in service (active) on the backup server following a failover. Switchback types are either intelligent or automatic. Intelligent switchback requires administrative intervention to switch the resource back to the primary server while automatic switchback occurs as soon as the primary server is back on line and reestablishes LifeKeeper communication paths.

Click **Next** to continue.

e. Select or enter a **Template Priority.** This is the priority for the DB2 hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended.

Note: This selection will appear only for the initial extend of the hierarchy.

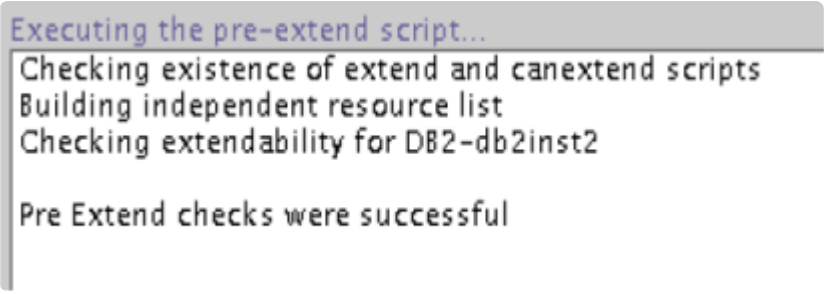
Click **Next** to continue.

f. Select or enter the **Target Priority**. This is the priority for the new extended DB2 hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from

1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

Click **Next** to continue.

g. An information box appears explaining that LifeKeeper has successfully checked your environment and that all requirements for extending this resource have been met. If there are requirements that have not been met, LifeKeeper will disable the **Next** button, and enable the **Back** button.



Click on the **Back** button to make changes to your resource extension.

Click **Cancel** to extend your resource another time.

Click **Next** to launch the Extend Resource Hierarchy configuration task.

Click **Finish** to confirm the successful extension of your DB2 resource instance.

4. Click **Done** to exit the Extend Resources Hierarchy menu selection.

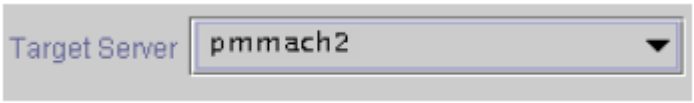
Note: Be sure to test the functionality of the new instance on *both* servers.

6.2.5.4. Unextending Your DB2 Resource Hierarchy

1. From the LifeKeeper GUI menu, select **Edit**, **Resource**, and **Unextend Resource Hierarchy.**

2. Select the **Target Server** where you want to unextend the DB2 resource. It cannot be the server where the resource is currently in service (active).

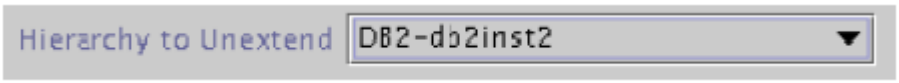
   Note: If you selected the Unextend task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next** to continue.

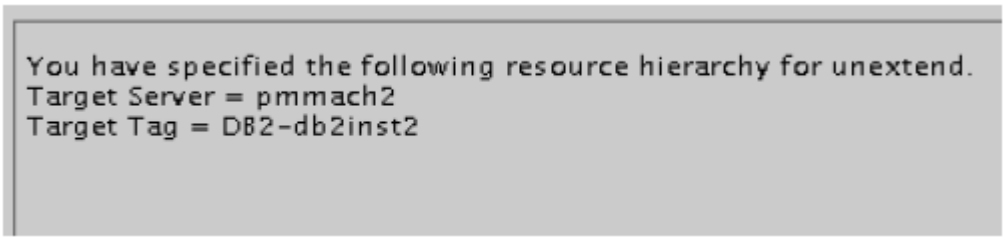
3. Select the DB2 **Hierarchy to Unextend.**

   Note: If you selected the Unextend task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next** to continue.

4. An information box appears confirming the target server and the DB2 resource hierarchy you have chosen to unextend.

   Click **Unextend**.

5. Another information box appears confirming that the DB2 resource was unextended successfully.

6. Click **Done** to exit the Unextend Resource Hierarchy menu selection.

6.2.5.5. Testing Your DB2 Resource Hierarchy

Test your DB2 resource hierarchy by initiating a manual switchover that will simulate a failover of the resource instance from the primary server to the backup server.

Performing a Manual Switchover from the LifeKeeper GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, **Resource**, and **In Service**. For example, an in-service request executed on a backup server causes the DB2 resource hierarchy to be placed in-service on the backup server and taken out-of-service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out of Service** request, the resource hierarchy is taken out-of-service without bringing it in-service on the other server. The resource can only be brought in-service on the same server, if it was taken out-of-service during resynchronization.

> **Important**: After bringing your resource hierarchy in service on the backup server, you should attempt to connect to the databases, especially when using raw devices as tablespace containers. This is necessary to ensure that all disk partitions are visible on the backup servers and the raw bindings are being established correctly.

If the raw bindings have not been established on the backup servers, it is most likely caused by the fact that new partitions were created on the primary server and added to the configuration, but the partition tables have not yet been updated on the backup servers.

The solution is to reboot the backup servers so that the partition tables are updated correctly.

6.2.6. DB2 Troubleshooting

Symptom	Possible Cause
One or more of your DB2 EEE partition servers fail to start	The ***db2nodes.cfg*** file's port number may have erroneously outgrown the range set in the ***/etc/services file***. View the number of ports set in the *db2nodes.cfg* file and ensure that the ports range value in the */etc/services* file is large enough to accommodate.
LifeKeeper *"In-Service"* or *"Out-of-Service"* operation hangs	The DB2 environment variable: **DB2_NUM_FAILOVER_NODES** may not have been properly set. Ensure that for all servers in your configuration, this environment variable is set to equal the total number of partitions in the instance. EXAMPLE: **db2set DB2_NUM_FAILOVER_NODES** =<partitions in instance>
LifeKeeper *"In-Service"* operation hangs	The **dasupdt** command may not have been executed on the DB2 Administration server. Ensure that the **dasupdt** command was successfully executed on the DB2 Administration server.
LifeKeeper First Switch over operation fails	The DB2 Fenced User may not have been created on the backup server. Verify the DB2 Fenced User for the specified instances exists with the same user and group id for the primary. Ensure that the protected instance is also a member of the Administration Server group.
You need to add a new node to your existing DB2 resource hierarchy	Please see the **nodes** utility man page for complete instructions on adding a new node to your currently existing LifeKeeper DB2 resource hierarchy.
Administration Server fails to start	Verify another Administration Server is not already running on specified port.
Creating a DB2 Resource Hierarchy takes long time	Creating a resource may take long time to protect DB2 instance that has large DB. Activate before creating a resource.

Error Messages

Refer to the DB2 Recovery Kit Message Catalog for a list of all messages that may be encountered while utilizing the DB2 kit.

6.2.7. Setting Up DB2 to use Raw I/O

There are several requirements for configuring RAW I/O devices for DB2 so that the DB2 instance can be protected by LifeKeeper.

Requirements

- The Linux OS must support Raw I/O devices. For most distributions this support was included in the 2.4 kernel, but there are some distributions that support Raw I/O on a 2.2. kernel.
- All Raw I/O devices must be bound to a shared disk partition. A number of shared SCSI disk partitions is required. The exact number is determined by the number of tablespaces that will be located on Raw I/O devices. (Please see to DB2 documentation for guidelines for writing tablespaces on raw devices).
- DB2 Version 7.1 Fix Pack 3 or later OR DB2 Version 8 or higher is required.

Naming Conventions

The naming of the raw device and controller varies by Linux distribution.

- On Red Hat the device name is */dev/raw/raw<number>* and the controller is */dev/rawctl*.
- On SuSE the name of the device is */dev/raw<number>* and the controller varies between */dev/raw*, */dev/rawctl*, and */dev/raw/rawctl*.

Raw I/O Setup Steps

The following steps 1-4 were taken from Section 7.3.1.1 (“Using Raw I/O on Linux”) of the *IBM Db2 Universal Database Release Notes Version 7.2/Version 7.1 Fix Pack 3*. In this example, the raw partition to be used is */dev/sda5*. It should not contain any valuable data.

Note that step 4 or 5 will vary depending upon whether you are using Multiple Logical Nodes.

1. Calculate the number of 4 096-byte pages in this partition, rounding down if necessary.

   Example:

   ```
   # fdisk /dev/sda

   Command (m for help):p

   Disk /dev/sda:255 heads, 63 sectors, 1106 cylinders

   Units = cylinders of 16065 * 512 bytes
   ```

Device Boot	Start	End	Blocks	System	ID
dev/sda1	1	23	4200997	83	Linux
/dev/sda2	524	1106	4682947+	5	Extended
/dev/sda5	524	1106	4682947	83	Linux

```
Command (m for help):q

#
```

The number of pages in */dev/sda5* is:

```
num_pages = floor( ((1106-524+1)*16065*512)/4096 )

num_pages = 11170736
```

2. Bind an unused raw device node to this partition. Since this needs to be done every time the machine is rebooted, and requires root access, you may want to add the raw bindings to a system initialization file (i.e. `rc.local` or `boot.local`.) **These bindings must be removed once the hierarchy is under LifeKeeper protection.** LifeKeeper will re-establish the raw bindings for Raw I/O devices that are under LifeKeeper protection.

   Use **raw -qa** to see which raw device nodes are already in use:

   ```
   raw /dev/raw/raw1 /dev/sda5

   /dev/raw/raw1:bound to major 8, minor 5
   ```

3. Set global read permissions on the raw device controller and the disk partition. Set global read and write permissions on the raw device:

   ```
   # chmod a+r /dev/rawctl

   # chmod a+r /dev/sdb1

   # chmod a+rw /dev/raw/raw1
   ```

4. **Important:** This step only applies if you are using DB2 EE OR your DB2 EEE configuration will never run Multiple Logical Nodes (MLNs) even after failover. If the configuration may run MLNs at some point, proceed to step 5.

   Create the tablespace in DB2, specifying the raw device, not the disk partition.

   For example:

   ```
   CREATE TABLESPACE dms1
   ```

```
MANAGED BY DATABASE

USING (DEVICE '/dev/raw/raw1' 11170736)
```

Tablespaces on raw devices are also supported for all other page sizes supported by DB2.

5. **Note:** This step must be followed if the configuration is running MLNs or will run MLNs at some point after failover.

   Create the table space in DB2, specifying the raw device, not the disk partition, and specify a different raw I/O device node for each DB2 instance partition.

   For example:

```
CREATE TABLESPACE dms1

MANAGED BY DATABASE

USING (DEVICE '/dev/raw/raw1' 11170736) on NODE (NODENUM)

USING (DEVICE '/dev/raw/<different raw device node>' ####### ) on NODE
(NODENUM)
```

   Note: ON NODE must be used because each DB2 node (database partition server) must use a different raw I/O device. This must be specified even if the node is running on a different machine so that the failover will work correctly.

6.3. Recovery Kit for EC2 Administration Guide

The Recovery Kit for EC2 provides a mechanism to recover an Elastic IP from a failed primary server to a backup server. It also provides a mechanism to enable the IP Recovery Kit to work in multiple availability zones.

Please see the Principles of Operation for a comparison and additional information about the definition, scenarios, and operation of the Recovery Kit for EC2.

SIOS Protection Suite Documentation

The following is a list of SIOS Protection Suite for Linux related information available from SIOS Technology Corp.

- SPS for Linux Technical Documentation
- SPS for Linux Release Notes
- SIOS Technology Corp. Documentation

Please refer to Amazon Elastic Compute Cloud (EC2) Documentation for more information.

> **Note:** “Amazon Web Services, the “Powered by Amazon Web Services” logo, “AWS”, “Amazon EC2”, “EC2”, “Amazon Elastic Compute Cloud”, “Amazon Virtual Private Cloud”, and “Amazon VPC” are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries.

6.3.1. EC2 Principles of Operation

Recovery Kit for EC2 provides two functions.

1. The Route Table scenario (Backend Cluster) manages Route Table for LifeKeeper-protected IP resources to be reached from clients within the Amazon VPC™.
2. The Elastic IP scenario (Frontend Cluster) manages Elastic IP available from the Internet.

Route Table scenario (Backend Cluster):

To clarify the administration and operation of Route Table, consider the scenario shown in Figure 1.

This example configuration contains one Amazon VPC™, two Availability Zones (AZ).

There are two Subnets in each AZ.

- A first Subnet (hereinafter referred to as “Public Subnet”) connects to the Internet via Internet Gateway by Route Table – see Route Table of 10.0.1.0/24 and 10.0.3.0/24.

- A second Subnet (hereinafter referred to as “Private Subnet”) connects to the Internet via NAT Instance by Route Table – see Route Table of 10.0.2.0/24 and Route Table of 10.0.4.0/24.

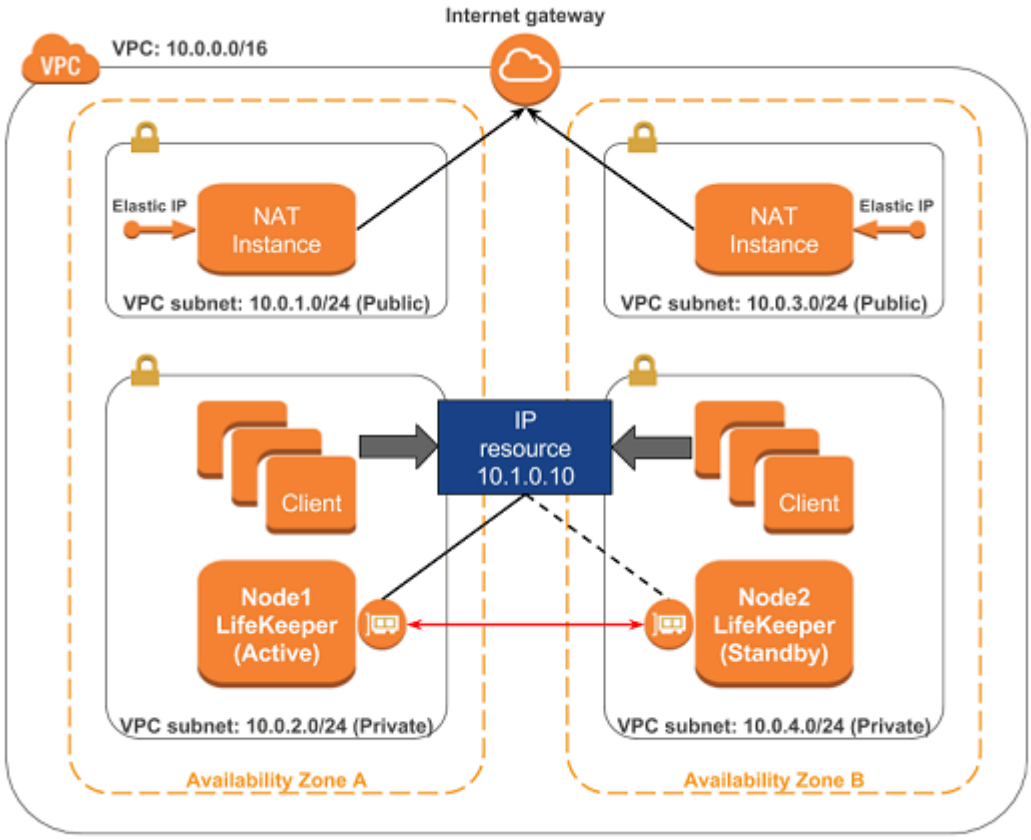
In each Public Subnet, there is an EC2 instance to which you assigned an Elastic IP for NAT (hereinafter referred to as “NAT Instance”).

In each Private Subnet, there is an EC2 instance for LifeKeeper Active/Standby (hereinafter referred to as “Node1” and “Node2”), and there are clients that will use the applications protected by Node1/Node2.

Each Node1/Node2 has two Elastic Network Interfaces (ENIs).

Configure the Network ACLs and Security Groups to be able to communicate between each Instance and each Node.

Figure 1. Route Table scenario



Route Table of 10.0.1.0/24 and 10.0.3.0/24

Destination	Target	Note
10.0.0.0/16	Local	Default
0.0.0.0/0	Internet Gateway	In order to connect to the Internet, requires the allocation of an Elastic IP.

Route Table of 10.0.2.0/24

Destination	Target	Note
10.0.0.0/16	Local	Default
10.1.0.10/32 (IP resource)	Elastic Network Interface (ENI) on LifeKeeper Active Node	This Target is updated by Recovery Kit for EC2 during a switchover.
0.0.0.0/0	NAT instance (10.0.1.0)	Connect to the Internet via NAT

Route Table of 10.0.4.0/24

Destination	Target	Note
10.0.0.0/16	Local	Default
10.1.0.10/32 (IP resource)	Elastic Network Interface (ENI) on LifeKeeper Active Node	This Target is updated by Recovery Kit for EC2 during a switchover.
0.0.0.0/0	NAT instance (10.0.3.0)	Connect to the Internet via NAT

When a resource switchover is performed, LifeKeeper will take the IP resource out of service on Node 1. The Target entry of 10.1.0.10/32 in each Private Subnet will be updated to reflect the ENI of Node2. The IP resource will be brought in-service on Node2. Therefore IP address traffic to 10.1.0.10 is effectively redirected to Node2 by the new Route Table configuration changes in the Private Subnet.

If you need to access the IP address 10.1.0.10 from another subnet containing the public subnet, please add the destination route 10.1.0.10/32 to the route table entry for each subnet. LifeKeeper controls all entries for which the destination is set as “10.1.0.10/32” in the route table within the VPC.

Elastic IP scenario (Frontend cluster):

To clarify the administration and operation of Elastic IP, consider the scenario shown in Figure 2.

This example configuration contains one Amazon VPC™, two Availability Zones (AZ).

There is one Subnet in each AZ.

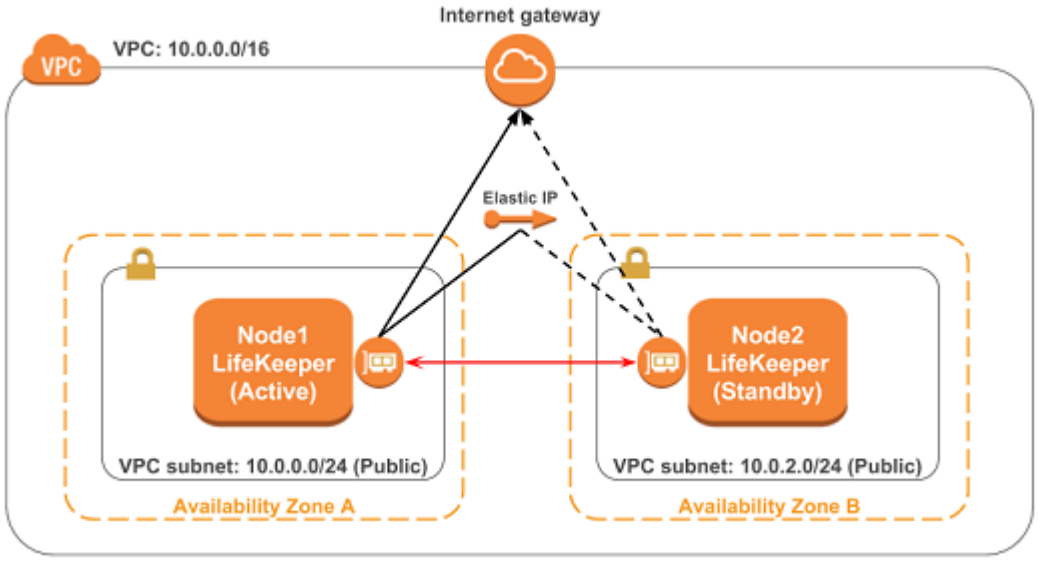
Each Subnet connects to the Internet via Internet Gateway by Route Table.

In Subnet, there is an EC2 instance for LifeKeeper Active/Standby (hereinafter referred to as “Node1” and “Node2”).

Each Node1/Node2 has two Elastic Network Interfaces (ENIs).

Configure the Network ACLs and Security Groups to be able to communicate between each Node.

Figure 2. Elastic IP scenario



The system administrator allocates an Elastic IP address of frontend cluster to the ENI.

Assuming that Node1 is the primary server for the resource, the administrator creates the EC2 resource hierarchy on Node1 using the wizard described in the section entitled Creating a Resource Hierarchy.

When resource switchover is performed, Recovery Kit for EC2 disassociates the Elastic IP from the ENI on Node 1. After that Recovery Kit for EC2 determines if the elastic IP is associated with the ENI on Node 2, if not, associates the Elastic IP to the ENI. Therefore client on the Internet can reach Node 2 via the Elastic IP after switchover.

Note: Standby nodes need to have an access to the end point in order to control the EC2 instance: that is, it is necessary to connect to the outside VPC. Please refer to "Requirements" for details. A public IP address is not necessary to access the endpoint when using PrivateLink. For details, please refer to "VPC Endpoints."

6.3.2. Recovery Kit for EC2 Requirements

Before attempting to install or remove the Recovery Kit for EC2 you must understand Amazon Web Service software requirements, as well as the installation and removal procedures for the Recovery Kit for EC2 package.

Amazon Web Service and Software Requirements

Before installing and configuring the Recovery Kit for EC2, be sure that your configuration meets the following requirements:

Amazon Virtual Private Cloud (VPC):

- The recovery kit requires a VPC be configured within AWS
- Two or more Subnets created on different Availability Zones (AZ)
- Each Subnet contains associated Route Tables
- If you are configuring a Public (Frontend) Cluster, then one or more Elastic IPs must be allocated

Amazon Elastic Compute Cloud (EC2):

- The recovery kit requires two or more EC2 instances.
- The instances are associated on each Subnet.
- The instances are attached to an Elastic Network Interface (ENI).
- AWS Command Line Interface (AWS CLI) needs to be installed in each of EC2 the instances. For the details, please refer to AWS Command Line Interface Installation.
- All the EC2 instances must be able to access Amazon EC2 services endpoints (AWS Regions and Endpoints) using the protocols HTTP and HTTPS. Please configure EC2 and the OS properly.
- In order to obtain metadata of Amazon EC2 instances, it is necessary to have an access to IP address 169.254.169.254 using the HTTP protocol.
- Since the AWS CLI is used, outbound connections on TCP port 443 must be enabled.
- Since the Auto Recovery function may conflict with the recovery function of LifeKeeper, it is not recommended to use these functions together.

Note: If the path name of AWS CLI executable files is not specified on the “PATH” parameter in the LifeKeeper defaults file `/etc/default/LifeKeeper`, you must append the path name of AWS CLI executable files to the “PATH” parameter.

AWS Identity and Access Management (IAM):

In order for LifeKeeper to operate AWS, an IAM user or IAM role with the following access privilege is required. Please configure an EC2 IAM role or configure AWS CLI appropriately so that it can be accessed from root user of the EC2 instance.

-
    - ec2:DisassociateAddress
    - ec2:DescribeAddresses
    - ec2:AssociateAddress
    - ec2:DescribeRouteTables
    - ec2:ReplaceRoute

LifeKeeper software:

You must install the same version of LifeKeeper software and any patches on each server. Please refer to the SPS for Linux Technical Documentation and the SPS for Linux Release Notes for specific LifeKeeper requirements.

LifeKeeper Recovery Kit for EC2:

You must install the same version of Recovery Kit for EC2 software and any patches on each server.

LifeKeeper IP Recovery Kit:

If you are using the Recovery Kit for EC2 to provide protection for the Route Table (Backend Cluster), you must install the same version of LifeKeeper for Linux IP Recovery Kit software and any patches on each server.

Note: Please refer to the SPS for Linux Release Notes or your sales representative for the latest release compatibility and ordering information. You should refer to the SIOS Protection Suite Installation Guide for specific instructions on how to install or remove the LifeKeeper Recovery Kit for EC2.

6.3.3. Recovery Kit for EC2 Configuration

To ensure that your LifeKeeper configuration provides the protection and flexibility you require you'll need to be aware of the configuration requirements. To appropriately plan your configuration you must understand Amazon, Amazon Virtual Private Cloud, (VPC), Amazon Elastic Compute Cloud (EC2), and the user system setup hierarchy options. In addition to planning your configuration, this section also includes the specific tasks required to configure your recovery kit.

Specific Configuration Considerations for Amazon EC2

In order to properly configure your Recovery Kit for EC2 you should review the following topics to ensure that you have the information necessary to complete the configuration tasks:

- User System Setup

See the following topics for further configuration considerations:

- EC2 Resource Monitoring and Configuration Considerations
- EC2 Local Recovery and Configuration Considerations

Specific Configuration Considerations for Amazon EC2

The following configuration tasks for EC2 resources are described in this section. They are unique to an EC2 resource instance and different for each recovery kit.

- Creating a Resource Hierarchy. Creates an application resource hierarchy in your LifeKeeper cluster.
- Deleting a Resource Hierarchy. Deletes a resource hierarchy from all servers in your LifeKeeper cluster.
- Extending Your Hierarchy. Extends a resource hierarchy from the primary server to a backup server.
- Unextending Your Hierarchy. Unextends (removes) a resource hierarchy from a single server in your LifeKeeper cluster.
- Viewing and Editing EC2 Configuration Properties. Displays configuration details for an EC2 resource and allows some of them to be modified.
- Adjusting Recovery Kit for EC2 Tunable Values. Tunes characteristics of the overall behavior of the Recovery Kit for EC2.

The following tasks are described in the Administration section within the SPS for Linux Technical

Documentation. They are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View Properties / Edit Properties View or edit the properties of a resource hierarchy on a specific server.

The rest of this section explains how to configure your recovery kit by selecting certain tasks from the Edit menu of the LifeKeeper GUI. You may also select each configuration task from the toolbar.

- Right-click on a global resource in the Resource Hierarchy Tree (left-hand pane) of the status display window to display the same drop down menu choices as the Edit menu. This is only an option when a hierarchy already exists.
- Right-click on a resource instance in the Resource Hierarchy Table (right-hand pane) of the status display window to perform all the configuration tasks, except Creating a Resource Hierarchy, depending on the state of the server and the particular resource.

6.3.3.1. Adjusting Recovery Kit for EC2 Tunable Values

For the parameters that can be configured in the Recovery Kit for EC2, refer to the EC2 Parameters List.

6.3.3.2. Creating an EC2 Resource Hierarchy

To create a resource instance from the primary server, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop down menu, select **Create Resource Hierarchy**.

2. A dialog box will appear with a drop down list showing all of the recognized recovery kits installed within the cluster. Select **"Amazon EC2**" from the drop down list and click **Next**.

3. You will be prompted to enter the following information. (When the Back button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful in the event that you need to correct previously entered information.)

Note: If you click the Cancel button at any time when creating your hierarchy, LifeKeeper will cancel the entire creation process.

Field	Tips
Switchback Type	This dictates how the EC2 resource will be switched back to this server when the server comes back up after a failover. You can choose either intelligent or automatic. • Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. • Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths. **Note:** The switchback type can be changed later from the General tab of the Resource Properties dialog box.
Server	Select the Server for the EC2 resource (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop down list.
EC2 Resource type	The EC2 Recovery Kit provides protection for two AWS recovery scenarios. The Route Table and Elastic IP scenario. The Route Table scenario is used in conjunction with a local virtual IP address and is typically used for Backend Clusters. The Elastic IP scenario is used for protection of an Elastic IP and is typically used for Frontend Clusters. Select the EC2 type to be used.
IP resource	This field will only appear and be set in the Route Table scenario. Select the IP resource. This is the virtual IP resource that is protected by LifeKeeper and configured in the Route Table address in the VPC. **Note:** The list will only show IP resources that are ISP and IPv4 based.

Network Interface	This field will only appear and be set in the Elastic IP scenario. Select the Network Interface to associate with Elastic IPs.
Elastic IP	This field will only appear and be set in the Elastic IP scenario. Select the Elastic IP to be related to the network interface.
EC2 Resource Tag	Select or enter a unique EC2 Resource Tag name for the EC2 resource instance you are creating.This field is populated automatically with a default tag name, ec2-<resource>, where <resource> is the resource name. This tag can be changed.

1. Click **Create**. The Create Resource Wizard will then create your EC2 resource.

2. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your EC2 resource hierarchy. If LifeKeeper detects a problem an ERROR will appear in the information box. If the validation is successful your resource will be created. Click **Next.**

Another information box will appear confirming that you have successfully created an EC2 resource hierarchy. You must extend that hierarchy to another server in your cluster in order to place it under LifeKeeper protection.

When you click **Continue**, LifeKeeper will launch the Pre-Extend configuration task. Refer to Extending Your Hierarchy for details on how to extend your resource hierarchy to another server.

If you click **Cancel** now, another dialog box will appear alerting you that you will need to manually extend your EC2 resource hierarchy to another server at some other time to put it under LifeKeeper protection.

6.3.3.3. Deleting an EC2 Resource Hierarchy

To delete a resource hierarchy from all of the servers in your LifeKeeper environment, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the dropdown menu, select **Delete Resource Hierarchy**.

2. Select the name of the Target Server that you are deleting from your EC2 resource hierarchy and click **Next**.

   Note: This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in either pane.

3. Select the Hierarchy to Delete. Identify the resource hierarchy you wish to delete, highlight it then click **Next**.

   Note: This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in the left or right pane.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete** to proceed.

5. An information box appears confirming that the EC2 resource was deleted successfully.

6. Click **Done** to exit.

6.3.3.4. Extending Your EC2 Hierarchy

After you have created a hierarchy, you must extend that hierarchy to another server in the cluster. There are three possible scenarios to extend your resource instance from the template server to a target server.

- Continue from creating the resource into extending that resource to another server.
- Enter the Extend Resource Hierarchy task from the edit menu as shown below.
- Right click on an unextended hierarchy in either the left or right hand pane.

Each scenario takes you through the same dialog boxes (with a few exceptions, detailed below).

1. If you are entering the Extend wizard from the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Extend Resource Hierarchy**. This will launch the Extend Resource Hierarchy wizard. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the LifeKeeper Extend Resource Hierarchy defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults**.

2. The Pre-Extend Wizard will prompt you to enter the following information.

Note: The first two fields appear only if you initiated the Extend from the Edit menu. It should be noted that if you click Cancel at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extension process to that particular server. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

Field	Tips
Switchback Type	Select the Switchback Type. This dictates how the EC2 instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either intelligent or automatic. • Intelligent switchback requires administrative intervention to switchback the instance to the primary/original server. • Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box.
Template Priority	Select or enter a Template Priority. This is the priority for the EC2 hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended.

	Note: This selection will appear only for the initial extend of the hierarchy.
Target Priority	Select or enter the Target Priority. This is the priority for the new extended EC2 hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). **Note:** LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities do not need to be consecutive and no two servers can have the same priority for a given resource.

3. An information box will appear explaining that LifeKeeper has successfully checked your environment and that all the requirements for extending this EC2 resource have been met. If there were some requirements that have not been met, LifeKeeper will not allow you to select the **Next** button, and the **Back** button will be enabled. If you click **Back**, you can make changes to your resource extension according to any error messages that may appear in the information box. If you click **Cancel** now, you will need to manually extend your EC2 resource hierarchy to another server to put it under LifeKeeper protection. When you click **Next**, LifeKeeper will launch you into the Extend Resource Hierarchy configuration task.

4. The Extend Resource Hierarchy configuration task will prompt you to enter the following information:

Field	Tip
EC2 Resource Tag	Select or enter the EC2 Resource Tag. This is the resource tag name to be used by the EC2 resource being extended to the target server. **Note:** The field in not editable.

5. An information box will appear verifying that the extension is being performed. Click **Next Server** if you want to extend the same EC2 resource instance to another server in your cluster. This will repeat the Extend Resource Hierarchy operation. If you click **Finish**, LifeKeeper will verify that the extension of the EC2 resource was completed successfully.

6. Click **Done** to exit from the Extend Resources Hierarchy menu selection.

Note: Be sure to test the functionality of the new instance on all servers.

6.3.3.5. EC2 Local Recovery and Configuration

Local Recovery scenario (Backend Cluster):

When a failure of the protected Route Table is detected by Recovery Kit for EC2, the resulting failure triggers the execution of the EC2 local recovery script. The local recovery gathers specified IP resource entries in all Route Tables and changes the entries' Target to the ENI on the active server. If the local recovery attempt fails, LifeKeeper will perform a failover of the EC2 resource and all dependent resources to a standby server. See the Principles of Operation section for the configuration of this scenario.

> **Note:** Since the recovery kit will protect the configuration of the route table once the corresponding EC2 resource gets created, the route table should not be modified manually.

The following example shows a typical scenario of the local recovery: When the recovery kit detects a wrong target setting of IP routing in the route table, the local recovery replaces the target to the ENI on the active server. During this process nothing will be changed regarding the entry of -10.1.0.20/32- on the Route Table B.

IP resource	10.1.0.10
ENI on Active Node	eni-01234567

Route Table A – Before

Destination	Target
10.1.0.10/32	eni-89abcdef
10.0.0.0/16	local

Route Table A – After

Destination	Target
10.1.0.10/32	eni-01234567
10.0.0.0/16	local

Route Table B – Before

Destination	Target
10.1.0.10/32	eni-89abcdef
10.1.0.20/32	eni-89abcdef
10.0.0.0/16	local

Route Table B – After

Destination	Target
10.1.0.10/32	eni-01234567
10.1.0.20/32	eni-89abcdef
10.0.0.0/16	local

Elastic IP scenario (Frontend Cluster):

When a failure of the protected Elastic IP is detected by Recovery Kit for EC2, the resulting failure triggers the execution of the EC2 local recovery script. The local recovery allocates the Elastic IP to the ENI on the active node. If the local recovery attempt fails, LifeKeeper will perform a failover of the EC2 resource and all dependent resources to a standby server. See the Principles of Operation section for the configuration of this scenario.

6.3.3.6. EC2 Resource Monitoring and Configuration

Route Table scenario (Backend Cluster):

The recovery kit uses AWS CLI to perform the monitoring of the Route Table settings to enable an access from clients within the VPC to the protected IP resources. The recovery kit ensures that the target of the IP resources for all the Route Tables in the VPC is correctly set to the ENI on the active server. Otherwise, the recovery kit performs the EC2 local recovery process.

Elastic IP scenario (Frontend Cluster):

The recovery kit uses AWS CLI to monitor the association of the Elastic IP with the ENI on the active server. The recovery kit ensures that the Elastic IP is correctly associated with the ENI attached on the active server. Otherwise, the recovery kit performs the EC2 local recovery process.

Note: In both scenarios, when a timeout occurs at AWS CLI, no failover will be performed and the resource will remain in ISP state. Only a timeout related message will be logged in the LifeKeeper log. The recovery kit will execute the monitoring once again after a check interval. See the EC2 Parameters List for more information about how to configure the value for timeout.

6.3.3.7. Unextending Your EC2 Hierarchy

To unextend a hierarchy complete the following steps:

1. From the **LifeKeeper GUI menu**, select **Edit**, then **Resource**. From the dropdown menu, select **Unextend Resource Hierarchy.**

2. Select the Target Server that you are unextending from the EC2 resource. It cannot be the server that the EC2 resource is currently in service on. Click **Next**.

   Note: If you selected the **Unextend** task by right-clicking from the right pane on an individual resource instance, the dialog box will not appear.

3. Select the EC2 Hierarchy to unextend. Click **Next.**

   Note: If you selected the **Unextend** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, the dialog will not appear.

4. An information box will appear confirming the target server and the EC2 resource hierarchy you have chosen to unextend. Click **Unextend.**

5. An information box will appear confirming the EC2 resource was unextended successfully.

6. Click **Done** to exit.

6.3.3.8. EC2 User System Setup

Route Table scenario (Backend Cluster):

The Route Table protection option in the Recovery Kit for EC2 provides the ability to automatically update the routing in the VPC. During a failover the recovery kit will update the route table to reflect the new Elastic Network Interface (ENI) location of the virtual IP address on the target server. In order for LifeKeeper to protect, monitor and update the Route Table in the VPC, the following configuration steps must be performed:

- The virtual IP address to be protected by the LifeKeeper for Linux IP Recovery Kit must be out of range of the allocated CIDR in the VPC.
- The virtual IP address must be protected by LifeKeeper prior to creating the Recovery Kit for EC2 resource.
- The Source/Dest Checking of the ENI must be disabled. This is required in order for the instance to accept network packets for the virtual IP address.
- Broadcast PING checking of the LifeKeeper IP resources must be disabled. LifeKeeper monitors IP resources by executing the Broadcast PING test of the IP address on the local subnet. In multiple availability zone environments this feature would not be useable because of the different subnets that exist between multiple availability zones. To disable this feature you must set the NOBCASTPING entry in the */etc/default/LifeKeeper* configuration file as follows:

  NOBCASTPING=1

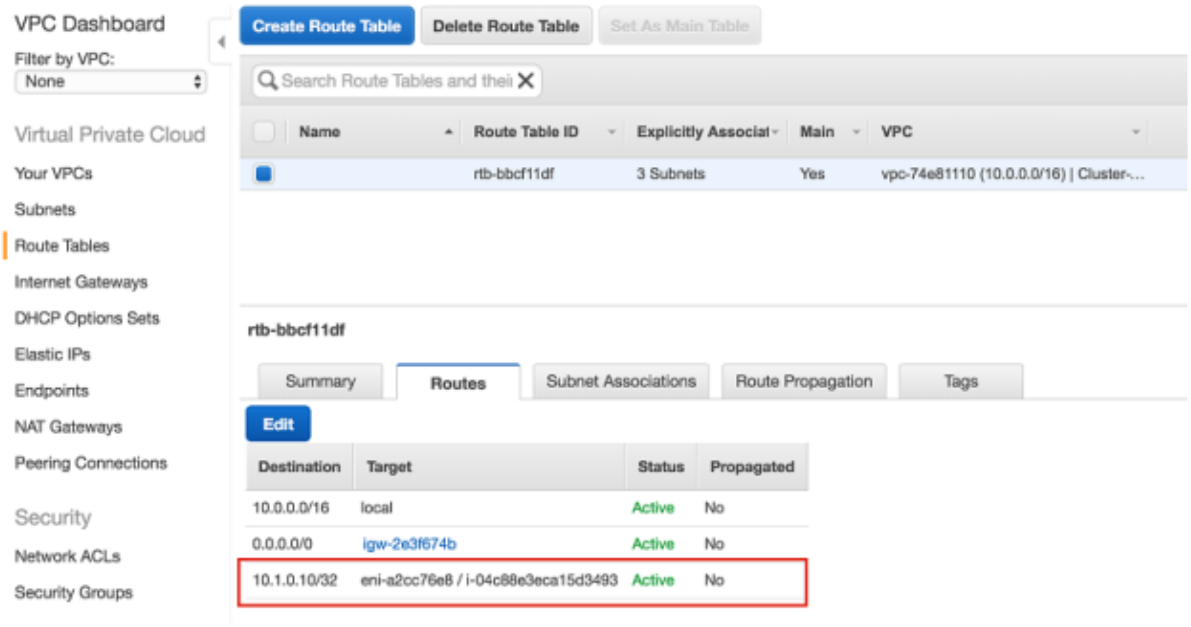
- The Route Table should have a route entry for the virtual IP address and the ENI of the active server.

  Note: Since the EC2 recovery kit will protect the configuration of the Route Table once the corresponding EC2 resource has been created, the Route Table should not be modified manually after hierarchy creation.

  Example:

  Destination: VIP 10.1.0.10/32

  Target: eni-a2cc76e8



Elastic IP scenario (Frontend Cluster):

The Elastic IP (EIP) protection option in the Recovery Kit for EC2 provides the ability to automatically re-associate an EIP with a specific ENI (the ENI used by EC2 resource on the active or backup server).

In order for LifeKeeper to protect, monitor and update the association of an EIP with the ENI on the active or backup server, the following configuration steps must be performed:

- One ENI can be associated with only one Elastic IP. No other EIPs (any EIPs other than the one used by EC2 resource) should be associated with the specific ENIs. Otherwise the recovery kit will disassociate any other EIPs that are already associated with the specific ENIs.

Notes:

- Since an Elastic Block Store (EBS) of AWS can only be attached to one EC2 instance, DataKeeper for Linux is recommended when creating an HA cluster configuration using EBS.
- We recommend increasing RESRVRECTIMEOUT in /etc/default/LifeKeeper to 300 from 150 as the default. RESRVRECTIMEOUT is the number of seconds that a LifeKeeper process will sleep when waiting to reserve a resource for “recovery”, while another process already has the resource reserved.

6.3.4. EC2 Troubleshooting

The Message Catalog provides a list of all messages that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. This full listing may be searched for any error code received, or you may go directly to the Recovery Kit for EC2 Message Catalog which contains a list of all messages that may be encountered while utilizing the Recovery Kit for EC2.

6.4. LVM Recovery Kit Administration Guide

The SIOS Protection Suite (SPS) for Linux Logical Volume Manager (LVM) Recovery Kit provides logical volume support for other SPS Recovery Kits. Thus, SPS-protected applications can take advantage of the benefits offered by the Logical Volume Manager, including simplified storage management and the ability to dynamically re-size volumes as needs change.

The LVM Recovery Kit is different from most other SPS Recovery Kits in that it is never used alone but always as a dependency of another SPS resource. As such, many of the operations typically associated with an SPS Recovery Kit – for example, creating a hierarchy – are not directly applicable to the LVM Recovery Kit.

Document Contents

This guide explains the following topics:

- Documentation and References. Provides a list of related SPS for Linux documents and where to find them, along with references to a number of helpful documents about the LVM product.
- Requirements. Describes the hardware and software necessary to properly set up, install and operate the LVM Recovery Kit. Refer to the SPS for Linux Installation Guide for specific instructions on how to install or remove the SPS for Linux software.
- Overview. Provides a general description of the LVM Recovery Kit and corresponding resource types.
- SPS LVM Hierarchy Creation and Administration. Includes a detailed description of LVM Recovery Kit administration tasks through SPS.
- Troubleshooting. Provides a list of informational and error messages with recommended solutions.

6.4.1. LVM Documentation and References

The following LifeKeeper product documentation is available from SIOS Technology Corp.:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- SIOS Protection Suite Installation Guide
- Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is available on the SIOS Technology Corp. website at:

http://docs.us.sios.com

For information on LVM, refer to:

RedHat: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Logical_Volume_Manager_Administration/index.html

SUSE: https://www.suse.com/documentation/sles11/stor_admin/data/lvm.html

LVM HowTo: http://www.tldp.org/HOWTO/LVM-HOWTO/index.html (This document is out-of-date)

6.4.2. LVM Recovery Kit Requirements

Your SPS configuration must meet the following requirements prior to the installation of the SPS for Linux LVM Recovery Kit. Please see the SIOS Protection Suite Installation Guide for specific instructions regarding the configuration of your SPS for Linux hardware and software.

6.4.2.1. LVM Hardware and Software Requirements

Hardware Requirements

- **Servers**. This recovery kit requires two or more computers configured in accordance with the requirements described in the SPS for Linux Release Notes and the SPS for Linux Installation Guide, which are shipped with the product media.

- **Data Storage**. The LVM Recovery Kit can be used in conjunction both with shared storage and with replicated storage provided by the DataKeeper for Linux product. It cannot be used with network attached storage (NAS). Otherwise, the kit has no specific requirements on storage configurations beyond the requirements of the recovery kit protecting the application sitting on top of the logical volume(s).

Software Requirements

- **Operating System**. LVM is included in all major Linux distributions. See the SPS for Linux Release Notes for a list of supported distributions and LVM versions.

- **Logical Volume Manager**. The recovery kit installation requires that the lvm orlvm2 rpm package be installed. This release of the SPS Logical Volume Manager Recovery Kit supports both LVM Version 1 and LVM Version 2 (LVM2). The specific versions of LVM supported are those delivered by the Linux distributions.

- **SPS Software**. You must install the same version of SPS core software and any recovery kits including the LVM Recovery Kit and any patches on each server. Please refer to the SPS for Linux Release Notes for specific SPS requirements.

- **SPS for Linux Logical Volume Manager Recovery Kit**. The Logical Volume Manager Recovery Kit is provided on the SPS Installation Image File (sps.img). It is packaged, installed and removed via the Red Hat Package Manager, rpm: `steeleye-lkLVM`.

During package installation, checks are made to ensure that supported versions of both the SPS Core package and the LVM package are present on the system where the LVM Recovery Kit is being installed. The SPS for Linux Release Notes contains information on the required versions of these packages.

Refer to the SPS for Linux Installation Guide for instructions on how to install or remove the SPS Core software and the LVM Recovery Kit.

The LVM Recovery Kit must be installed on each server in the cluster on which LVM is being used to manage disk resources that are to be protected by SPS.

The LVM Recovery Kit must be installed prior to the hierarchy creation and extension of applications that

sit on top of an LVM volume.

6.4.3. LVM Recovery Kit Overview

LVM Operation

LVM is currently the standard volume management product included with all of the major Linux distributions. LVM allows multiple physical disks and/or disk partitions to be grouped together into entities known as volume groups. Volume groups may then be divided or partitioned into logical volumes. Logical volumes are accessed as regular block devices and as such may be used by file systems or any application that can operate directly with a block device.

Logical volume managers are principally used to simplify storage management. Logical volumes can be resized dynamically as storage requirements change, and volume groups and logical volumes can be sensibly named with identifiers chosen by the administrator rather than physical disk or partition names such as `sda` or `sdc1`.

The following diagram shows the relationship of the LVM entities. File systems or applications use logical volumes. Logical volumes are created by partitioning volume groups. Volume groups consist of the aggregation of one or more physical disk partitions or disks.

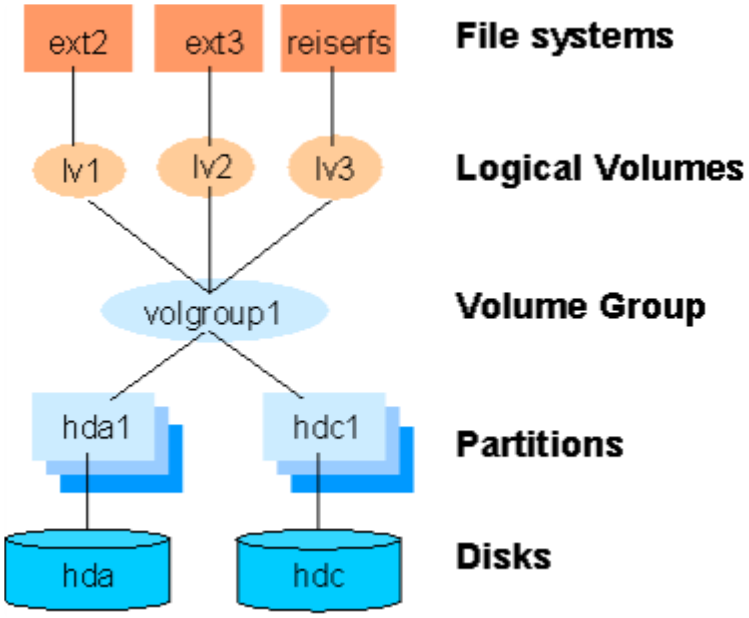


Figure 1: Logical Volume Manager Entity Relationships

SPS for Linux LVM Recovery Kit

The SPS LVM Recovery Kit provides the support needed to allow other SPS recovery kits to operate properly on top of Linux logical volumes. To accomplish this support, the LVM Recovery Kit installs two new resource types: `lvmlv` and `lvmvg` which correspond to logical volumes and volume groups respectively. The `lvmlv` and `lvmvg` resources exist solely for internal use so that other SPS resources can operate.

As shown in Figure 1, each volume group has one or more logical volumes that depend on it. Conversely, each logical volume must have a volume group on which it depends. A typical SPS hierarchy containing these two LVM resources looks much like the relationships shown in Figure 1. Refer to Figure 2 in the SPS LVM Hierarchy Creation and Administration section for an example of an actual SPS hierarchy.

The LVM Recovery Kit uses the commands provided by the lvm package to manage the volume group and logical volume resources in an SPS hierarchy. Volume groups and logical volumes are configured (or activated) when a hierarchy is being brought in service during a failover or switchover operation and are unconfigured when a hierarchy is being taken out of service.

6.4.3.1. LVM Recovery Kit Notes and Restrictions

The following notes and restrictions apply to this version of the LVM Recovery Kit.

Support for Raw I/O and Entire Disks

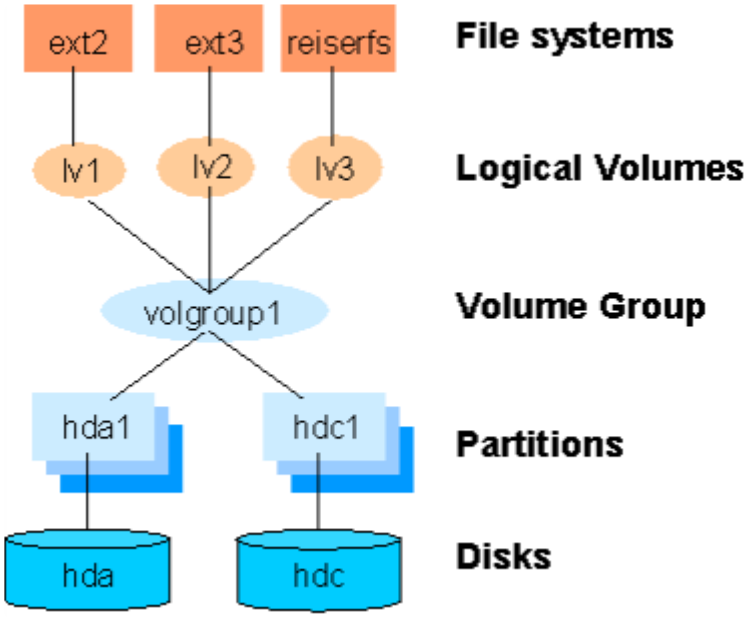
While Figure 1 shows logical volumes residing below various file systems and volume groups on top of disk partitions, it is important to note that the LVM Recovery Kit can support raw access to logical volumes when used in conjunction with the SPS Raw I/O Recovery Kit and can manage volume groups that are composed of one or more entire disks (e.g. `/dev/sdc`) rather than disk partitions (e.g. `/dev/sdc1`)

Also see the section Using LVM with DataKeeper for a further option in the use of LVM.

Volume Group Activation

In the current LVM implementations, when a volume group is activated, all logical volumes associated with that volume group are also activated automatically. For SPS, this means that there will be times when a logical volume is active despite the fact that its associated resource instance is still marked as being Out-of-Service (OSU). In a typical failover or switchover operation, SPS will attempt to bring the logical volumes in service immediately after the volume groups anyway, and the resulting calls to the restore script will return immediately with a success indication. This unneeded attempt to bring the logical volumes in service has no usability impact.

LVM Figure 1


ext2
ext3
reiserfs
lv1
lv2
lv3
volgroup1
hda1
hdc1
hda
hdc
File systems
Logical Volumes
Volume Group
Partitions
Disks

6.4.4. SPS LVM Hierarchy Creation and Administration

SPS LVM hierarchies are created automatically during the hierarchy creation process for resources that sit on top of logical volumes. The creation and extension of hierarchies containing the LVM resource types will always be driven by the create and extend processes of a higher-level resource type, likewise the delete and unextend.

The figure below is a LifeKeeper GUI screen shot showing a complete hierarchy containing LVM resources. Note that the resources in the hierarchy are displayed by their SPS IDs for clarity rather than the default display by tags.

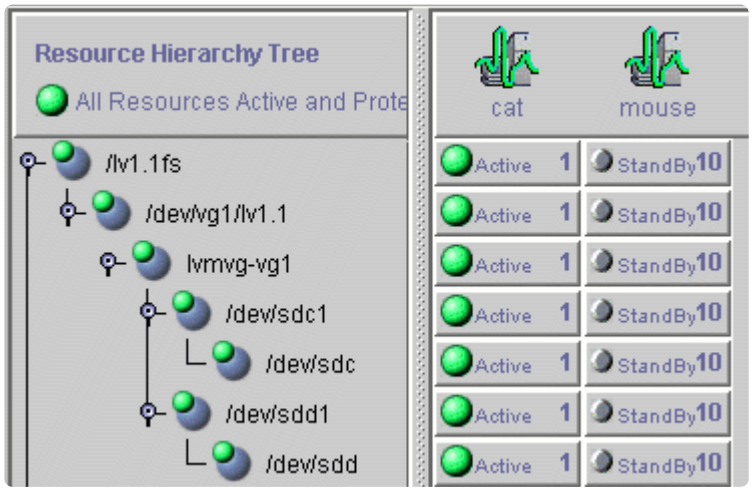


Figure 2: SPS Hierarchy Containing LVM Resources

The hierarchy pictured in Figure 2 is a file system hierarchy created by selecting the File System Recovery Kit under the **Edit > Server > Create Resource Hierarchy** menu selection. It consists of a file system resource, `/lv1.1fs`, mounted on an LVM logical volume, */dev/vg1/lv1.1*. That logical volume is a part of the `vg1` volume group represented with the SPS ID `lvmvg-vg1`. The volume group `vg1` is composed of two physical disk partitions,*/dev/sdc1* and */dev/sdd1*. The hierarchy also includes the underlying disk devices, */dev/sdc* and */dev/sdd*, below each of the disk partitions.

6.4.4.1. LVM Hierarchy Creation Procedures

To create a hierarchy in which a file system or higher-level application uses an LVM logical volume, the following high-level procedure should be followed.

1. Determine the desired configuration of your LVM volume groups and logical volumes. In doing this, keep in mind the following points:

- ◦ All of the disk resources associated with a given volume group must move together from one server to another in the SPS cluster.

- ◦ All of the logical volumes associated with a given volume group (and any file systems or applications which use them) must move together from one server to another in the SPS cluster.

- ◦ lvm2-lvmetad is disabled during the LVM RK installation. If you install lvm2 after installing the LVM RK, you need to disable it manually. Refer to the operating system documentation for more information.

2. On the system which is to be the primary server for your application, create and activate the desired volume groups and logical volumes using the tools provided by the LVM package and described in the *LVM HowTo* document referenced in the Documentation and References topic.

   If you are using shared storage, you must ensure that all physical volumes assigned to a volume group are properly shared between the machines in the SPS cluster on which you intend to run the protected application. If you intend to use LVM with DataKeeper, see the Using LVM with DataKeeper topic.

3. Create file systems on each of the logical volumes. If instead you intend to use raw I/O, bind a raw device to each of the logical volume devices.

4. Configure the protected application on the file systems following the configuration instructions in the administration guide for the SPS recovery kit associated with the application.

   Create and extend the application hierarchy following the instructions in the appropriate application recovery kit administration guide.

> **IMPORTANT:** Perform manual in-service operations to temporarily move the application hierarchy to each of the cluster nodes to which the hierarchy has been extended. This step must be done once prior to any node failover operations in order for the LVM subsystem on each cluster node to know about the configuration of the new volume groups and logical volumes. After you have performed these manual switchovers, move the application hierarchy back to the desired primary cluster node.

6.4.4.2. Using the LVM Recovery Kit with DataKeeper

SPS for Linux currently supports both the use of DataKeeper “above” LVM and LVM “above” DataKeeper. In a standard DataKeeper configuration, using DataKeeper above LVM is supported and DO NOT install the SPS LVM Recovery Kit. DataKeeper is the only recovery kit necessary. However, using the LVM above DataKeeper configuration, the LVM Recovery Kit is required.

SIOS recommends using DataKeeper above LVM; however, if the LVM above DataKeeper configuration is being used, a two-phase hierarchy creation process must be used. The DataKeeper devices (i.e. hierarchies) must be configured using the DataKeeper “Data Replication Resource” option prior to the creation of the LVM volume groups and logical volumes on the primary server. Once the desired volume groups and logical volumes have been created, the remainder of the hierarchy is created according to the configuration instructions for the recovery kit associated with the application to be protected. The resulting hierarchy will look something like the one shown in Figure 3 below.

Note: For data consistency reasons, in an LVM over DataKeeper configuration, there must either be only one DataKeeper mirror or multiple ***synchronous*** mirrors.

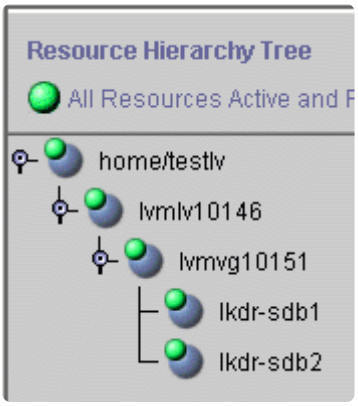


Figure 3: Hierarchy with LVM above DataKeeper

6.4.4.3. Volume Group Reconfiguration

One of the primary benefits of using a logical volume manager is the ability to dynamically resize logical volumes as storage requirements change. Because this may involve adding or deleting physical partitions or disks from an LVM volume group definition, the LVM Recovery Kit includes a mechanism for modifying an existing resource hierarchy to reflect such a change.

All volume group, logical volume and file system reconfiguration should be performed outside of SPS prior to modifying the SPS hierarchy to reflect the changes. Refer to the *LVM HowTo* document referenced in the Documentation and References section for information about how this is done. If any of the steps require you to **unmount** or **unconfigure** a resource that is being protected by SPS, be sure to use the LifeKeeper GUI to do so, using the **Out of Service** operation.

> **IMPORTANT:** The new device MUST be seen by both systems (shared) before SPS will allow the reconfiguration to take place.

To update an SPS hierarchy following these changes, first access the **Resource Properties** dialog for the modified volume group, either by right-clicking on the active volume group resource and selecting **Properties** or by using the **Edit > Resource > Properties** menu selection and selecting the appropriate volume group resource in the **Select Resource** field. The resulting **Resource Properties** dialog should look like the one pictured in Figure 4 below including the **Resource Configuration** button near the bottom.

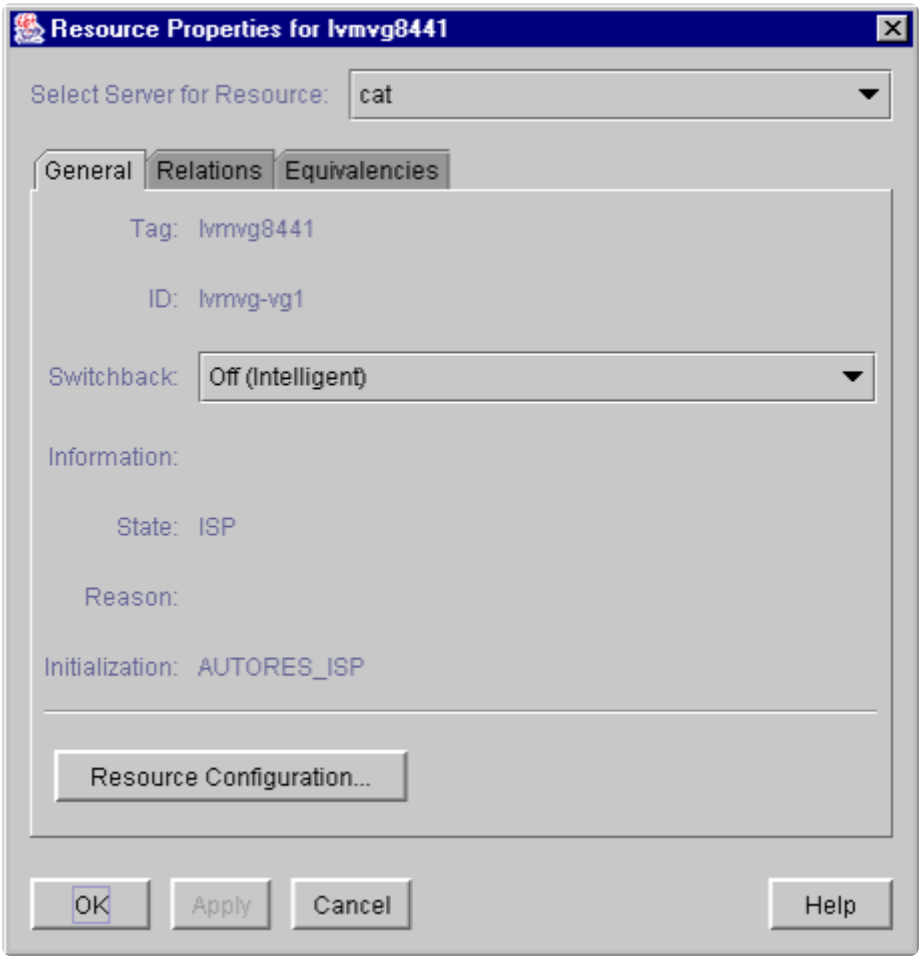


Figure 4: LVM Volume Group Resource Properties Dialog

Clicking the **Resource Configuration** button initiates the mechanism for reconfiguring your hierarchy to reflect any modifications to the volume group resource. After a brief pause, an information box will display the volume group modifications that SPS has detected. Figure 5 below shows an example in which a single disk partition has been added to a volume group.

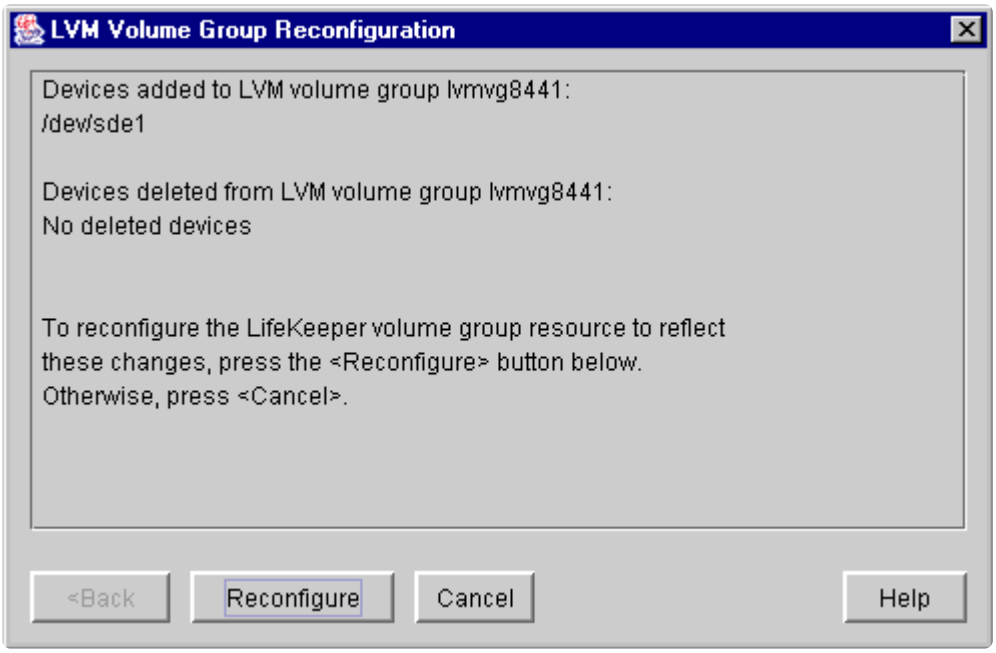


Figure 5: LVM Volume Group Reconfiguration for Added Device

As stated in the information box, to reconfigure the SPS volume group to reflect the changes that have been detected, simply click the **Reconfigure** button. If you do not wish to proceed with the SPS hierarchy modification, click **Cancel.**

After clicking the **Reconfigure** button, an information box will appear showing the progress of the reconfiguration procedure as shown in **Figure 6** below. When the process has been completed successfully, the **Done** button will become enabled. Clicking **Done** will close the information box and return you to the display of the **Resource Properties** dialog.

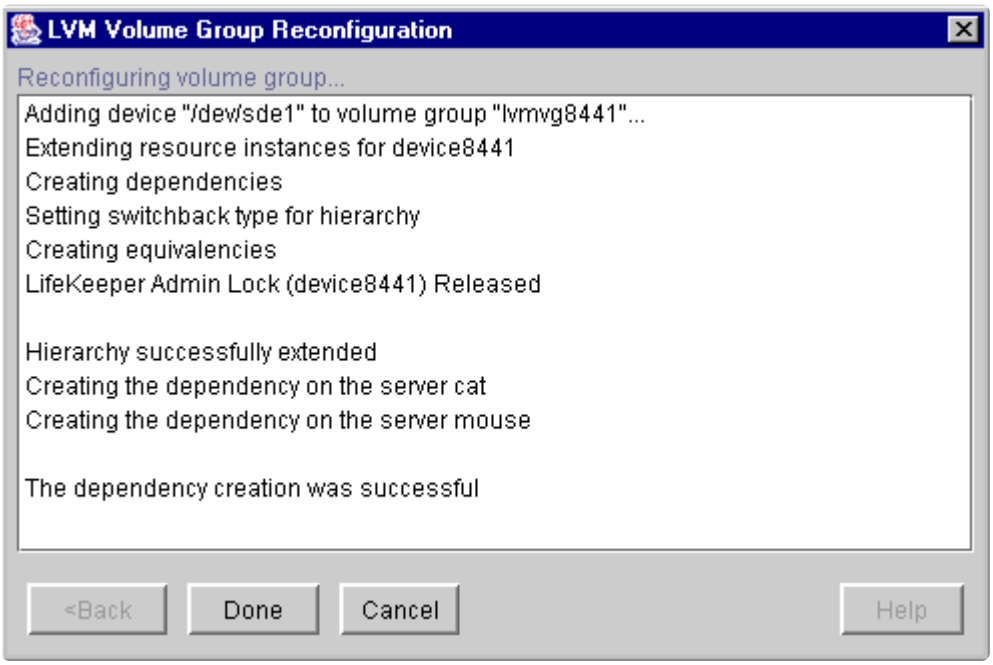


Figure 6: LVM Volume Group Reconfiguration for Added Device

The following two figures show examples of the information boxes that would be displayed during the reconfiguration process when a device partition has been removed from a volume group.

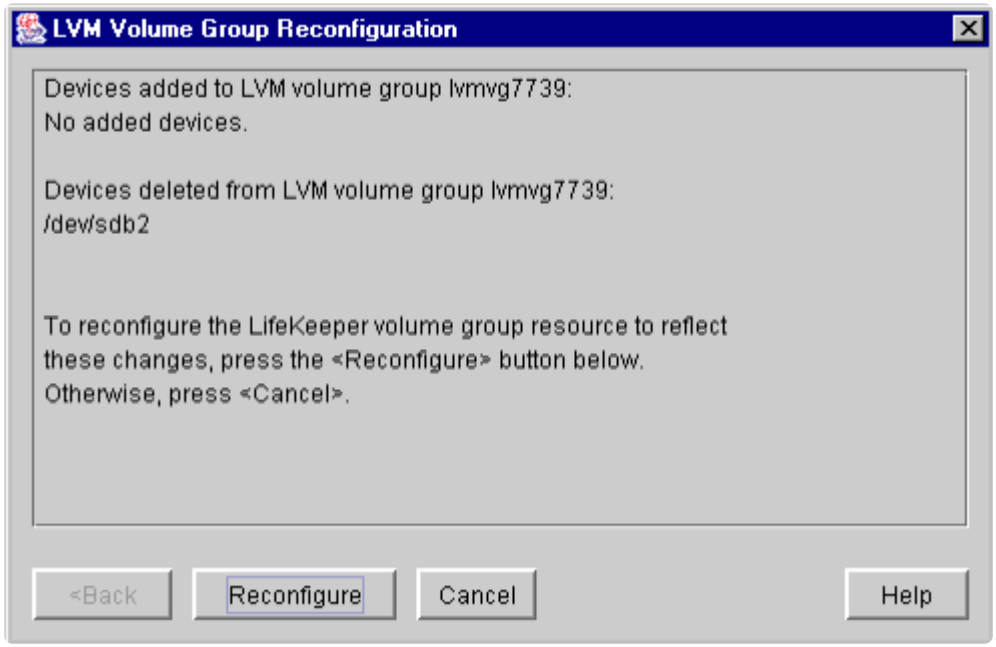


Figure 7: LVM Volume Group Reconfiguration for Deleted Device

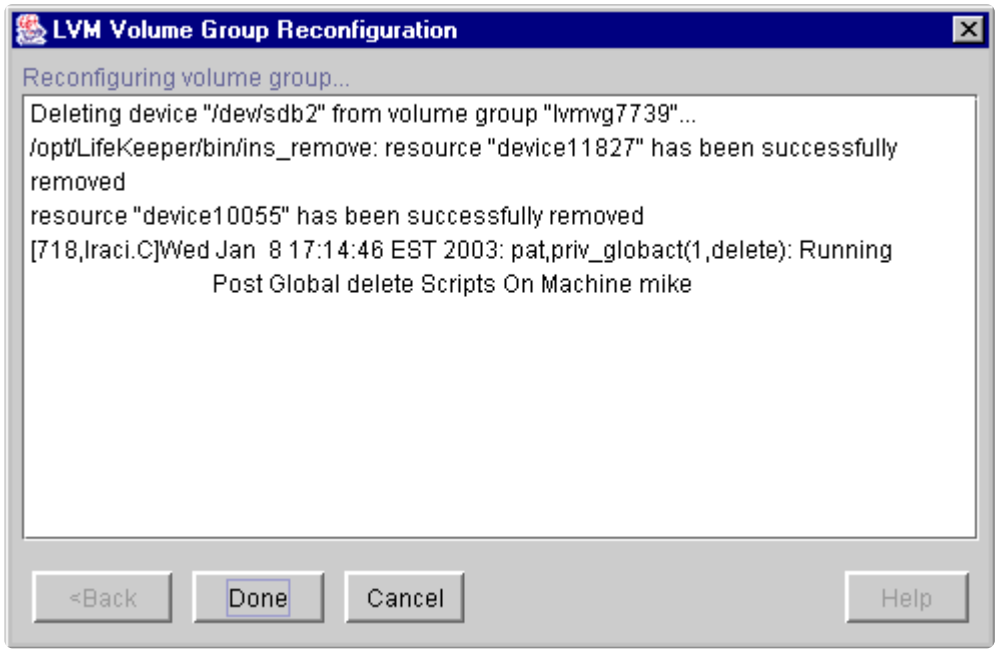


Figure 8: LVM Volume Group Reconfiguration for Deleted Device

6.4.5. LVM Troubleshooting

Error Messages

This section provides a list of messages that you may encounter with the use of the SPS LVM Recovery Kit. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Because the LVM Recovery Kit relies on other SPS components to drive the creation and extension of hierarchies, messages from these other components are also possible. In these cases, please refer to the Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

LVM Recovery Kit Error Messages

Error Number	Error Message
110000	<LVM resource type> resource type is not installed on <SPS server name>. Action: Install the LVM Recovery Kit on the identified system.
110001	This script must be executed on <SPS server name>.
110002	Failed to create <device name> hierarchy.
110003	Failed to create dependency <resource tag>-<resource tag> on machine <SPS server name>.
110004	LifeKeeper internal ID <resource ID> already in use.
110005	<LVM resource type> constructor requires a valid argument.
110006	Usage: adddelpv <VG tag> [addlist|dellist]
110007	WARNING: Failure in updating list of LifeKeeper-controlled volume groups `(/etc/lkvgs)`
110008	WARNING: The device hierarchy for <device name>, with tag <device resource tag>, cannot be extended automatically. Action: If the device hierarchy is not already extended, extend it using the LifeKeeper GUI. Then create a dependency from the volume group resource to the device hierarchy.
110009	Failed to create a dependency between volume group resource <volume group tag> and device resource <device tag>. Action: Create the dependency using the LifeKeeper GUI.
110010	Failed to make the LVM logical volume <Logical Volume Path> active with error code <error code>.
110011	Failed to vgscan the LVM volume group <Volume Group Name> with error code <error code>.
110012	Failed to vgimport the LVM volume group <Volume Group Name> with error code <error code>.

110013	Failed to make the LVM volume group <Volume Group Name> active with error code <error code>.

6.5. IP Recovery Kit Administration Guide

The SIOS Protection Suite for Linux Internet Protocol (IP) Recovery Kit provides a mechanism to recover an IP address from a failed primary server to a backup server in a LifeKeeper environment. The IP Recovery Kit can define an IP address that can be used to connect to a LifeKeeper-protected application. As with other LifeKeeper resources, IP resource switchovers can be initiated automatically as a result of a failure or manually by an administrative action.

The IP Recovery Kit supports the implementation of the TCP/IP protocol suite using secondary addresses on existing network interfaces, allowing it to provide switchover and failover of IP addresses without requiring extra standby network interface cards or *dummy* IP addresses. Starting with Release 7.4, the IP Recovery Kit supports both IPv4 and IPv6 addresses.

The following SPS product documentation is available from the SIOS Technology Corp. website:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- Optional Recovery Kit Documentation

6.5.1. IP Recovery Kit Principles of Operation

LifeKeeper brings an IP resource into service by creating an IP alias address on one of the physical network interfaces on the primary server. Users connect to the node using this alias address.

The IP Recovery Kit software performs checks to help ensure that the selected address, network mask and interface can function properly. The software verifies the following elements:

- **Unused resource**. The new IP address is not already assigned to any other IP resource in the LifeKeeper cluster.

- **Unique address**. The address cannot be currently active on the network. In addition to checking during creation, the software also performs the uniqueness check immediately before bringing the resource into service. If the software detects a duplicate address on the net, it does not bring the resource into service.

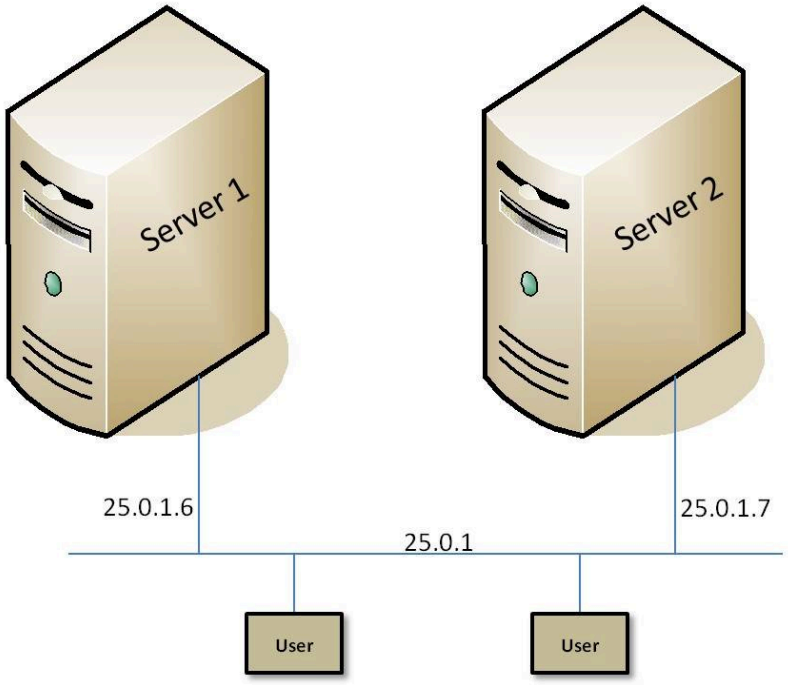
When the primary server fails, the IP Recovery Kit brings the IP resource into service on a backup server by configuring the IP alias on one of that server's physical network interfaces.

Since session context is lost following recovery, after the recovery, IP users must reconnect using exactly the same procedures they used to connect originally.

In a manual switchover, the IP Recovery Kit removes the alias address from service on the active server before adding it to the backup server.

To clarify the administration and operation of the IP Recovery Kit, consider the scenario shown in Figure 1. This example configuration contains two servers, Server 1 and Server 2. Each server has a single LAN interface, eth0, connected to subnet 25.0.1. The user systems are also on this subnet. The LAN interfaces on Server 1 and Server 2 have addresses 25.0.1.6 and 25.0.1.7, respectively.

Figure 1. Administration and Operation Scenario



The system administrator decides to use 25.0.1.10 as the alias address for an IP resource, to be called *ipname*. The administrator creates entries in the */etc/hosts* files (and in the DNS, if used), similar to the following:

25.0.1.6	server1
25.0.1.7	server2
25.0.1.10	ipname

Assuming that Server 1 is the primary server for the resource, the administrator creates the IP resource hierarchy for *ipname* on Server 1 using the wizard described in the section entitled Creating an IP Resource Hierarchy. The software finds the address associated with *ipname* (25.0.1.10) from */etc/hosts*, verifies that it is available and brings it into service by configuring a secondary address on eth0 on Server 1. eth0 on Server 1 now responds to both *server1* and *ipname*.

With LifeKeeper 7.3 or earlier, the new alias address can be verified using the `ifconfig` or `ip addr show` command. Starting with LifeKeeper 7.4, the `ip addr show` command should be used (for more information, see the IPv6 Known Issue).

Users can then connect to Server 1 by entering, for example, `telnet ipname`. If Server 1 crashes, LifeKeeper automatically switches over the ipname address to eth0 on Server 2. The user sessions on Server 1 terminate. When users re-run `telnet ipname`, they are connected to Server 2.

Regardless of where ipname is actively in service, addresses server1 and server2 are active and usable, though not protected by LifeKeeper recovery. The addresses could be used for any cases that require connection to a specific server by name rather than to a switched application. Examples might include remote system management and the LifeKeeper communications path. (In this case, for example, 25.0.1.6 and 25.0.1.7 would be used for the LifeKeeper communications path.)

IP Resource Monitoring

LifeKeeper monitors the health of the IP resources under its control on a periodic basis, using the following techniques, in this order.

1. Check the link status for the network interface on which the IP resource is configured to determine whether the interface is properly connected to the physical network.

2. Verify that the IP resource is still configured as an alias on the appropriate network interface.

3. Perform a broadcast ping test or ping a pre-configured list of addresses, using the protected IP address as the source address of the pings, to determine whether the IP resource can successfully send and receive data on the network.

   The broadcast ping test is the default test mechanism. It operates by sending a broadcast ping packet to the broadcast address of the subnet associated with the IP resource, using the protected IP address as the source address. If a response is received from any address other than addresses on the local system, the test is considered successful.

   For environments in which there are no systems on the network that can respond to the broadcast ping test (which is the default configuration of many systems), LifeKeeper also offers the ability to configure a list of addresses to be pinged as an alternative to the broadcast ping test. If such a list has been specified, the broadcast ping test is skipped, and all of the addresses in the list are pinged in parallel. The test is considered successful if a ping response is received from any one of the addresses in the Ping List. This technique is also useful to reduce broadcast storms on larger networks.

If any of these tests fail during the periodic health check of an IP resource, LifeKeeper is notified of the failure. LifeKeeper will first attempt a local recovery operation to try to restore the IP resource to a working state on the local node. See the section IP Local Recovery and Configuration Considerations for more information about the local recovery procedure. If local recovery is unsuccessful in restoring the IP resource to a working state, LifeKeeper will then attempt to migrate the application hierarchy containing the IP resource to another LifeKeeper system in the cluster.

LifeKeeper also uses these same health checks to verify the proper operation of an IP resource immediately after it is brought in-service. A failure of any of the checks will cause the in-service operation to fail.

The IP health check mechanisms can be tuned and adjusted in many ways. See the sections Viewing/ Editing IP Configuration Properties and Adjusting IP Recovery Kit Tunable Values for details.

6.5.2. IP Recovery Kit Requirements

Before attempting to install or remove the IP Recovery Kit, you must understand the hardware and software requirements for the package and the installation and removal procedures.

Kit Hardware and Software Requirements

Before installing and configuring the LifeKeeper IP Recovery Kit, be sure that your configuration meets the following requirements:

- **Servers**. The recovery kit requires two or more supported computers configured in accordance with LifeKeeper requirements described in the SIOS Protection Suite for Linux Technical Documentation and the SIOS Protection Suite for Linux Release Notes.

- **LifeKeeper software**. You must install the same version of LifeKeeper software and any patches on each server. Please refer to the SPS for Linux Technical Documentation and the SIOS Protection Suite for Linux Release Notes for specific LifeKeeper requirements.

- **LifeKeeper IP Recovery Kit**. You must have the same version of this recovery kit on each server.

- **IP network interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications. This interface should be configured. If there are no *ifcfg** files, IP switchover may fail when the interface is down.

  Note: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons; for example, heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation and local recovery support. Also, set an actual IP address for NIC used for the IP resource setting. By using the actual IP address, confirm the communication on IP network.

  For the configuration of channel bonding and network teaming that have been tested, please click here.

- **TCP/IP software**. Each server also requires the TCP/IP software.

Consult the SPS for Linux Release Notes or your sales representative for the latest release compatibility and ordering information.

You should refer to the SIOS Protection Suite Installation Guide for specific instructions on how to install or remove the LifeKeeper IP Recovery Kit.

6.5.3. IP Recovery Kit Configuration

To ensure that your LifeKeeper configuration provides the protection and flexibility you require, you need to be aware of the configuration rules. To appropriately plan your configuration, you must understand your network configuration, interface selection, user system setup, hierarchy options and the IP configuration tasks. In addition to planning your configuration, this section also includes configuration examples and the specific tasks required to configure your recovery kit.

Configuring TCP/IP with LifeKeeper

This section contains information you should consider before you start to configure TCP/IP and examples of typical LifeKeeper IP configurations.

Please refer to the SPS for Linux Technical Documentation for instructions on configuring your LifeKeeper Core resource hierarchies.

Specific Configuration Considerations for TCP/IP

In order to properly configure your IP Recovery Kit, you should review the following topics to ensure that you have the information necessary to complete the configuration tasks:

- Interface Selection
- User System Setup
- General IP Planning Considerations

See the following topics for further configuration considerations and examples:

- IP Resource Monitoring and Configuration Considerations
- IP Local Recovery and Configuration Considerations
- Configuration Examples
- Guidelines for Creating an IP Dependency

LifeKeeper Configuration Tasks

The following configuration tasks for virtual IP address resources are described in this section, as they are unique to an IP resource instance and different for each recovery kit.

- Creating an IP Resource Hierarchy. Creates an application resource hierarchy in your LifeKeeper cluster.

- Deleting a Resource Hierarchy. Deletes a resource hierarchy from all servers in your LifeKeeper cluster.
- Extending Your Hierarchy. Extends a resource hierarchy from the primary server to a backup server.
- Unextending Your Hierarchy. Unextends (removes) a resource hierarchy from a single server in your LifeKeeper cluster.
- Testing Your Resource Hierarchy. Tests a virtual IP resource hierarchy for proper configuration and operation.
- Viewing/Editing IP Configuration Properties. Displays configuration details for an IP resource and allows some of them to be modified.
- Adjusting IP Recovery Kit Tunable Values. Tunes characteristics of the overall behavior of the IP Recovery Kit.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View / Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, we explain how to configure your recovery kit by selecting certain tasks from the **Edit** menu of the LifeKeeper GUI. You can also select each configuration task from the toolbar. You may also right-click on a global resource in the Resource Hierarchy Tree (left-hand pane) of the status display window to display the same drop down menu choices as the **Edit** menu. This, of course, is only an option when a hierarchy already exists.

You can also right-click on a resource instance in the Resource Hierarchy Table (right-hand pane) of the status display window to perform all the configuration tasks, except Creating a Resource Hierarchy, depending on the state of the server and the particular resource.

6.5.3.1. Adjusting IP Recovery Kit Tunable Values

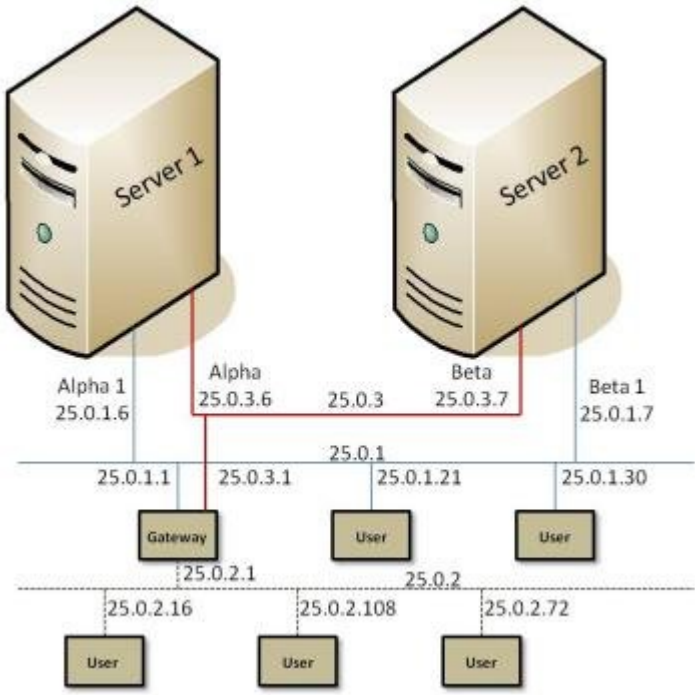
For details about the tunable values that are available for modifying the behavior of the IP Recovery Kit, please click here. These values are tuned by editing the */etc/default/LifeKeeper* configuration file. Because none of the components of the IP Recovery Kit are memory resident, changes to these particular values become effective immediately after they are changed in */etc/default/LifeKeeper*, without requiring a LifeKeeper restart.

6.5.3.2. IP Recovery Kit Configuration Examples

This topic identifies example network configurations and then describes two sample IP configuration exercises. The first example illustrates a typical case of a database application dependent upon a single IP resource and configured on a pre-existing subnet. The second example illustrates an active/active scenario where multiple IP resources are configured.

Network Configuration

The first two configuration examples assume the network configuration diagrammed in the following figure.



The network configuration has these components:

- **Servers**. The configuration has two servers, Server 1 and Server 2, each with the appropriate LifeKeeper and application software installed.
- **Interfaces**. Each server has two Ethernet interfaces, eth0 and eth1, configured as follows:

Interface	Server 1	Server 2
eth0	Server1 25.0.3.6	Server2 25.0.3.7
eth1	Server11 25.9.1.8	Server21 25.0.1.7

- **Network**. The network consists of three subnetworks:
  - º Low traffic backbone (25.0.3) primarily for servers
  - º High traffic backbone (25.0.1) with both servers and clients
  - º High traffic client network (25.0.2.)

    A gateway provides interconnection routing between all LANs. A Domain Name Server (not shown) is used for address resolution.

- **Heartbeat**. TCP heartbeat communication paths would be configured using either or both of the server subnetworks.

Typical Configuration Example

Server 1 and Server 2 have access to an application called mydatabase that resides on a shared disk. To ensure that the application mydatabase and the IP resources used to access it are switched together, the system administrator creates a mydatabase application resource and adds the IP resource to the application hierarchy as a dependency.

These are the configuration issues:

- **Application hierarchy**. The application hierarchy must exist before the administrator names it as a parent of the IP resource. For the purposes of this example, Server 1 is the primary server. The application resource tags are mydatabase-on-server1 and mydatabase-on-server2.

- **IP resource name**. The administrator adds the name and address of the IP resource to the /etc/hosts file on both Server 1 and Server 2 and to the DNS database. In this example, the IP resource name is databaseip and its network address is 25.0.1.2. If no name-to-IP address association is necessary, then this is not required.

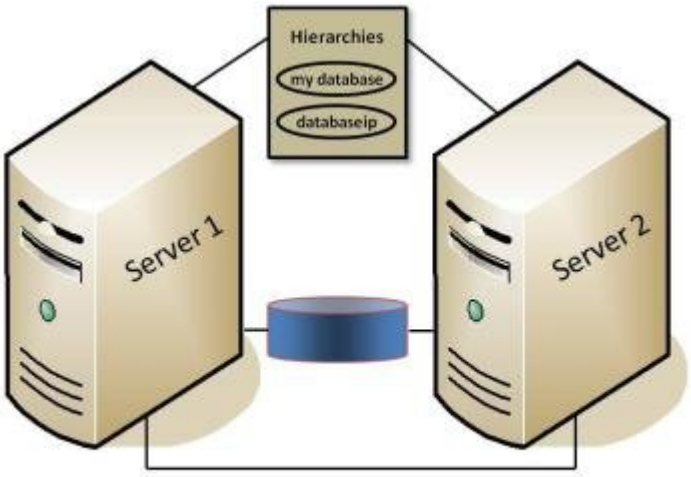
- **Routers, gateways, and users**. Because databaseip is an address on an existing subnet, no additional configuration is necessary. The IP resource is on the 25.0.1 subnet. All users connect to databaseip via the route they currently use to get to the 25.0.1 subnet. For example, users on 25.0.2 go through the gateway and users on 25.0.1 connect directly.

- **IP instance definition**. When the administrator enters databaseip as the IP resource on the Resource Hierarchy Create screen, the software performs several tests. It verifies that Server 1 can determine the address that goes with databaseip (it is in the hosts file and/or can be retrieved from the DNS). It also verifies that the address retrieved, address 25.0.1.2, is not already in use. Since the IP resource is on the 25.0.1 subnet, the IP Recovery software will ensure that it is configured on the eth1 interface. If the IP resource is acceptable, the software fills in the remainder of the wizard dialog boxes with default values, as shown in the table below Figure 3. If

you selected all the default values, an independent IP resource hierarchy called ip-databaseip would be created.

Note: The tables associated with each configuration illustration provide examples of the appropriate information that would be entered in the Create Resource Hierarchy wizard for the primary server (Server 1) and Extend Resource Hierarchy wizard for the backup server (Server 2). For additional details on what information should be entered into the wizards, refer to the LifeKeeper Configuration Tasks section later in this section. These tables can be a helpful reference when configuring your recovery kit.

Figure 3. Typical Configuration Example of IP Resource Creation



Configuration Notes:

1. The application resource is mydatabase-on-server1.
2. The IP resource is databaseip with a tag name of ip-databaseip.
3. If mydatabase-on-server1 fails, LifeKeeper switches it to Server 2; (ip-databaseip is only switched if a dependency exists).
4. If Server 1 fails, both resources are brought in-service on Server 2.
5. During a switchover, databaseip users would be disconnected. When they log back in, they can access any applications on Server 2.
6. During a manual switchover, users connected to Server 1 via connections other than databaseip remain connected to Server 1.

Creating an IP resource hierarchy on Server 1:

Server:	Server1
IP Resource:	databaseip

Netmask:	255.255.252.0
Network Interface:	eth1
IP Resource Tag:	ip-databaseip

Note: See the topic Guidelines for Creating an IP Dependency before extending an IP resource to a backup server.

Extending an IP resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend:	databaseip
Target Server:	Server2
Target Priority:	10
**IP Resource:	25.0.1.2
Netmask:	255.255.252.0
Network Interface:	eth1
IP Resource Tag	ip-databaseip

Note: The actual IP address associated with the DNS name is displayed in the ***Extend Wizard*** as the IP resource.

Test Your IP Resource

To verify the successful creation of the IP resource, the administrator should perform the following tasks:

1. From the LifeKeeper GUI, observe whether ip-databaseip is in-service (ISP) on Server 1.
2. From a remote server, connect to address databaseip using ping or telnet.
3. Test manual switchover by selecting the in_service option on Server 2 and selecting ip-databaseip. Verify that the IP address migrates to Server 2.

Active/Active Configuration Example

The second example, using the same network configuration, describes two IP resources, one active on each server.

Resource Addresses

For this example, the IP resources are server1ip (address 25.0.6.20) and server2ip (address 25.0.6.21). Entries for these resources must be in the /etc/hosts files on each server and in the DNS database.

Router Configuration

Because the selected addresses are on a new (logical) subnet, they can be configured for either eth0 or eth1. However, both must go on the same interface.

For this example, choosing eth0 means that all users would have to go through the gateway. Choosing eth1 would allow the users on the 25.0.1 subnet to access the resources directly (assuming that the new subnet had been added to their internal routing tables). Users on subnet 25.0.2 would still require the gateway. For the purposes of this example, the selected interface is eth1.

Regardless of which physical network is chosen to support the new subnet, the network administrator would have to add routing information to the gateway system before creating the IP resources.

First IP Resource Definition

The administrator creates the first IP resource on Server 1. eth0 is the first available interface on each server and would appear as the default. To define eth1 as the interface, the administrator selects it from the list of available interfaces.

Creating an IP resource hierarchy on Server 1:

Server:	Server1
IP Resource:	server1ip
Netmask:	255.255.252.0
Network Interface:	eth1
IP Resource Tag:	ip-server1ip

Note: See the topic Guidelines for Creating an IP Dependency before extending an IP resource to a backup server.

Extending an IP resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend:	server1ip
Target Server:	Server2
Target Priority:	10
**IP Resouce:	25.0.6.20
Netmask:	255.255.252.0

Network Interface:	eth1
IP Resource Tag:	ip-server1ip

Note ; The actual IP address associated with the DNS name is displayed in the ***Extend Wizard*** as the IP resource.

Second IP resource definition

The administrator creates the second IP resource on Server 2. eth0 is the first available interface on each server and would appear as the default. To define eth1 as the interface, the administrator selects it from the list of available interfaces.

Creating an IP resource hierarchy on Server 2:

Server:	Server2
IP Resource:	server2ip
Netmask:	255.255.252.0
Network Interface:	eth1
P Resource Tag:	ip-server2ip

Note: See the topic Guidelines for Creating an IP Dependency before extending an IP resource to a backup server.

Extending an IP resource hierarchy to Server 1:

Template Server:	Server2
Tag to Extend:	server2ip
Target Server:	Server1
Target Priority:	10
**IP Resouce:	25.0.6.21
Netmask:	255.255.252.0

Network Interface:	eth1
IP Resource Tag:	ip-server2ip

Note: The actual IP address associated with the DNS name is displayed in the ***Extend Wizard*** as the IP resource.

Note: Since subnet 25.0.6 is not active on Server 2, both eth0 and eth1 are available choices for the Primary network interface. On Server 1 (the backup server), the only choice is eth1 because the first IP resource, 25.0.6.20, is in service there. When the administrator saves the definition, LifeKeeper brings address 25.0.6.21 in-service on eth1 on Server 2.

Testing IP Resources

The administrator should verify that the new resources are functioning on both servers by performing the following tests:

1. With each resource on its primary server, verify that each is accessible by using either ping or telnet. The administrator may also want to test connectivity from all user sites.
2. Test switchover by manually bringing ip-server1ip into service on Server 2. Verify both resources are functional on Server 2.
3. Bring both resources into service on Server 1. Verify both resources are functional on Server 1.
4. Bring ip-server2ip back into service on its primary server, Server 2.

6.5.3.3. Creating an IP Resource Hierarchy

To create a resource instance from the primary server, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop down menu, select **Create Resource Hierarchy**.

2. A dialog box will appear with a dropdown list box menu listing all recognized recovery kits installed within the cluster. Select **IP** from the dropdown list and click **Next**.

3. You will be prompted to enter the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information.

   If you click the **Cancel** button at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

Field	Tips
Switchback Type	This dictates how the IP instance will be switched back to this server when the server comes back up after a failover. You can choose either *intelligent* or *automatic*. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths. The switchback type can be changed later from the General tab of the Resource Properties dialog box.
Server	Select the **Server** where you want to place the IP Address (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop down list.
IP Resource	Select or enter the actual **IP Resource**. This is the IP address or symbolic name that LifeKeeper will use for this resource. This is used by client applications to log in to the parent application over a specific network interface. If you use a symbolic name, it must exist in the local */etc/hosts* file or be accessible via a Domain Name Service (DNS). Alias names and domain names are acceptable as long as they meet the criteria listed above. No defaults are provided for this information field. **Note:** If you choose to use a symbolic name, be advised that when you extend this resource, the actual IP address will appear in one of the dialog boxes as the IP resource designation.
Netmask	Select or enter the network mask, **Netmask**, which your IP resource will use on the target server. Any standard netmask for the class of the specific IP resource address is valid. **Note**: The netmask you choose, combined with the IP address, determines the subnet that will be used by the IP resource and should be consistent with the network configuration.
Network Interface	Select or enter the **Network Interface** where your IP resource will be placed under LifeKeeper protection. This is the physical Ethernet card that the IP address is interfacing with. Valid choices will depend on the existing network configuration and values chosen for the IP resource address and netmask. The default value is the interface within the set of valid choices which most closely matches the address and

	netmask values you have selected.
IP Resource Tag	Select or enter a unique **IP Resource Tag** name for the IP resource instance you are creating.This field is populated automatically with a default tag name, ip-<resource>, where <resource> is the resource name or IP address. You can change this tag if you want to.

4. Click **Create**. The ***Create Resource Wizard*** will then create your IP resource.

5. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your IP resource hierarchy. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created. Click **Next**.

6. Another information box will appear explaining that you have successfully created an IP resource hierarchy, and you must Extend that hierarchy to another server in your cluster in order to place it under LifeKeeper protection.

   When you click **Continue**, LifeKeeper will launch the **Pre-Extend configuration task**. Refer to the Extending Your Hierarchy topic for details on how to extend your resource hierarchy to another server.

   If you click **Cancel** now, another dialog box will appear alerting you that you'll need to come back and extend your IP resource hierarchy to another server at some other time to put it under LifeKeeper protection.

6.5.3.4. Deleting an IP Resource Hierarchy

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Delete Resource Hierarchy**.

2. Select the name of the Target Server where you will be deleting your IP resource hierarchy and then click **Next**. (This dialog will not appear if you selected the Delete Resource task by right-clicking on a resource instance in either pane.)

3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete, and highlight it and then click **Next**. (This dialog will not appear if you selected the Delete Resource task by right-clicking on a resource instance in the left or right pane.)

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete** to proceed with resource deletion.

5. Another information box appears confirming that the IP resource was deleted successfully.

6. Click **Done** to exit out of the Delete Resource Hierarchy menu selection.

6.5.3.5. Extending Your IP Hierarchy

> **Note**: See the section on Guidelines for Creating an IP Dependency before extending an IP resource to a backup server.

After you have created a hierarchy, you will want to extend that hierarchy to another server in the cluster. There are three possible scenarios to extend your resource instance from the template server to a target server. The first scenario is when you "Continue" from creating the resource into extending that resource to another server. The second scenario is when you enter the Extend Resource Hierarchy task from the edit menu as shown below. The third scenario is when you right click on an unextended hierarchy in either the left or right hand pane. Each scenario takes you through the same dialog boxes (with a few exceptions, which are clearly detailed below).

1. If you are entering the ***Extend wizard*** from the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Extend Resource Hierarchy**. This will launch the ***Extend Resource Hierarchy wizard***. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the LifeKeeper Extend Resource Hierarchy defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults.**

2. The ***Pre-Extend Wizard*** will prompt you to enter the following information. **Note**: The first two fields appear only if you initiated the **Extend** from the **Edit** menu. It should be noted that if you click **Cancel** at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extension process to that particular server. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

Field	Tips
Template Server	Enter the server where your IP resource is currently in service.
Tag to Extend	Select the IP resource you wish to extend. This is the name of the IP instance you wish to extend from the template server to the target server. The wizard will list in the drop down list box all the resources that you have created on the template server that you selected in the previous dialog box.
Target Server	Select the **Target Server** where you are extending your IP resource hierarchy. The drop down box provides the names of the servers in your cluster that are not already in the selected hierarchy.
Switchback Type	Select the **Switchback Type**. This dictates how the IP instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either *intelligent* or *automatic*. Intelligent switchback requires administrative intervention to switchback the instance to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box.
Template Priority	Select or enter a **Template Priority**. This is the priority for the IP hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid,

	where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note**: This selection will appear only for the initial extend of the hierarchy.
Target Priority	Select or enter the **Target Priority**. This is the priority for the new extended IP hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

3. An information box will appear explaining that LifeKeeper has successfully checked your environment and that all the requirements for extending this IP resource have been met. If there were some requirements that had not been met, LifeKeeper would not allow you to select the Next button, and the Back button would be enabled. If you click Back, you can make changes to your resource extension according to any error messages that may appear in the information box. If you click Cancel now, you will need to come back and extend your IP resource hierarchy to another server at some other time to put it under LifeKeeper protection. When you click Next, LifeKeeper will launch you into the Extend Resource Hierarchy configuration task.

4. The Extend Resource Hierarchy configuration task will prompt you to enter the following information.

Field	Tips
IP Resource	This is the same **IP Resource** or address used in the Create Resource Wizard. This dialog box is for information purposes only. You cannot change the **IP Resource** that appears in the box.
Netmask	This is the same **Netmask** that was selected when the IP resource was created for the template server and will now be used by the IP resource for the target server. This dialog box is for information purposes only. You cannot change the **Netmask** that appears in the box.
Network Interface	Select or enter the **Network Interface**. This is the name of the network interface (i.e. Ethernet card) the IP resource will use on the target server.
IP Resource Tag	Select or enter the **IP Resource Tag**. This is the resource tag name to be used by the IP resource being extended to the target server.

5. An information box will appear verifying that the extension is being performed.

   Click **Next Server** if you want to extend the same IP resource instance to another server in your cluster. This will repeat the Extend Resource Hierarchy operation.

   If you click **Finish**, LifeKeeper will verify that the extension of the IP resource was completed successfully.

6. Click **Done** to exit from the **Extend Resources Hierarchy** menu selection.

Note: Be sure to test the functionality of the new instance on all the servers.

6.5.3.6. General IP Planning Considerations

After you have selected the addresses, netmasks and associated host/domain names you intend to use for IP resource hierarchies, add the appropriate entries to each server's */etc/hosts* file, and to the Domain Name Server (DNS), if used.

> **Note**: Even if you are using a DNS, it is strongly recommended that you place entries for the IP resources in the local */etc/hosts* files on all LifeKeeper servers. This will reduce recovery times. However, if the resource name that you enter when creating the IP instance is the IP address itself, then the host file entry is unnecessary.

Do not configure the protected IP addresses into your system as you would if you were creating a permanent logical interface to be activated at system boot time. The LifeKeeper software will manage them instead of the system software.

If any of the resource addresses are on new (logical) subnets, update routers to handle routing to these subnets.

6.5.3.7. Guidelines for Creating an IP Dependancy

How and/or when you are going to create a parent/child dependency between a LifeKeeper-protected application and a LifeKeeper-protected IP address is typically dependent on the LifeKeeper-protected application. For example, in a LifeKeeper-protected Apache environment, the parent/child dependency is created during the creation of the Apache resource hierarchy (assuming you have already created a protected IP address). In other applications that do not create this dependency automatically, it is recommended you use the following steps:

1. Create the application/parent resource hierarchy. **Note:** Do not extend the resource hierarchy to a backup server at this time. You will receive a warning message when you elect not to extend your hierarchy, but in this particular situation, it is the proper action to take.

2. Create the IP resource hierarchy. **Note:** Do not extend the IP resource hierarchy to a backup server at this time. You will receive a warning message when you elect not to extend your hierarchy, but in this particular situation, it is the proper action to take.

3. Create the parent/child dependency between the parent application resource hierarchy and the IP resource hierarchy using the Create Resource Dependency configuration task (see the LifeKeeper for Linux topic, Creating a Resource Dependency.

4. Finally, extend the application resource hierarchy to the backup server. Since the dependency has already been created, the dependent IP resource instance will also be extended to the backup server as part of the parent application resource hierarchy.

The steps outlined above save you from performing one extra extension (i.e. the extension of the IP resource to the backup server).

6.5.3.8. IP Interface Selection

When creating an IP resource, select the IP resource address, the netmask to use with the address and the network interface. Not all combinations are allowed. The address/netmask pair provided and all the address/netmask pairs currently in-service determine choices. Also, see the section on IP Local Recovery for additional configuration considerations if planning on using this feature of the recovery kit.

The selected address/netmask determines the subnet for the resource. If another address on the same subnet (either a physical or logical interface address) is currently in service on any interface, then the IP resource must be configured on that interface. The software performs tests to determine the allowed choices based upon the current network configuration. Select from any of the choices provided.

Because the IP Recovery Kit software does not distinguish between physical media types, the physical network for the resource must be determined and the address selected appropriately. For example, assume that you have a server connected to an Ethernet backbone on subnet xx.yy.12 and Ethernet LANs on subnets xx.yy.20 and xx.yy.30. If you want to create a resource on the first Ethernet subnet, select an address on that subnet, such as xx.yy.20.120.

In general, even though the IP Recovery Kit software allows you to select almost any value for the netmask, you should avoid selecting multiple netmasks for the same physical interface because multiple masks can cause packet misrouting.

One further consideration is the need to be consistent in your selection of interfaces on all LifeKeeper servers. If you configure several IP resources on a single interface on Server A, they should also be configured on a single interface on Server B.

When creating an IP resource hierarchy, you may utilize any interface which is initially UP and has a corresponding and correct network interface configuration file on both the primary and backup hosts, i.e. if using *eth1*, *eth1* must be UP and *eth1* must have a corresponding and correct *ifcfg-eth1* file (test with `ifup/ifdown ifcfg-eth1`) even if the configuration is minimal without any address assignments or is DOWN on boot.

6.5.3.9. IP Local Recovery and Configuration

The standard Linux NIC bonding mechanism is the recommended means of providing network interface redundancy in a high availability configuration. The LifeKeeper IP Recovery Kit fully supports the creation of virtual IP addresses on bonded interfaces.

Local Recovery Scenario

When a failure of an IP address is detected by the IP Recovery Kit, the resulting failure triggers the execution of the IP local recovery script. LifeKeeper will first attempt to bring the IP address back in-service on the current network interface. If the local recovery attempt fails, LifeKeeper will perform a failover of the IP resource and all dependent resources to a backup server.

6.5.3.10. IP Resource Monitoring and Configuration

By default, the LifeKeeper IP Recovery Kit monitors IP resources by executing a broadcast ping on the IP addresses logical subnet, then listening for replies. For this test to work properly, at least one additional non-LifeKeeper system capable of responding to broadcast pings must exist on the physical network, with an IP address on the same logical subnet as the IP resource. A router on the same logical subnet is usually sufficient to meet this need. Note that the default configuration of many devices is to not respond to broadcast pings, so it may be neccessary to change the configuration of at least one device.

If this requirement cannot be met, you can choose to either disable the broadcast ping test completely, or you can configure a static list of IP addresses that should be pinged as an alternative to the broadcast ping test mechanism. See the Adjusting IP Recovery Kit Tunable Values and Viewing/Editing IP Configuration Properties topics for more information about how to configure these options.

6.5.3.11. Testing Your IP Resource Hierarchy

You can test your IP resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Performing a Manual Switchover from the GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, then **Resource**, then finally **In Service** from the dropdown menu. For example, an in-service request executed on a backup server causes the application hierarchy to be placed in service on the backup server and taken out of service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out of Service** request, the application is taken out of service without bringing it in service on the other server.

In a manual switchover, the IP Recovery Kit removes the address from service on the active server before adding it to the backup server.

After switchover, the IP resource has a different hardware (MAC) address because it is associated with a different LAN interface. Before user systems can reconnect, the user systems' TCP/IP software must determine this new address mapping. The IP Recovery Kit automatically informs all connected servers that they must update their ARP (Address Resolution Protocol) tables to reflect the new mapping.

User systems running full TCP/IP implementations are updated immediately. User systems with less sophisticated implementations may have delayed update or may require routers as addressing intermediaries.

6.5.3.12. Unextending Your IP Hierarchy

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the dropdown menu, select **Unextend Resource Hierarchy**.

2. Select the Target Server where you want to unextend the IP resource. It cannot be the server where the IP address is currently in service.

   Note: If you selected the Unextend task by right-clicking from the right pane on an individual resource instance, this dialog box will not appear.

   Click **Next** to proceed to the next dialog box.

3. Select the **IP Hierarchy to Unextend**.

   Note: If you selected the Unextend task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next** to proceed to the next dialog box.

4. An information box appears confirming the target server and the IP resource hierarchy you have chosen to unextend.

   Click **Unextend**.

5. Another information box appears confirming that the IP resource was unextended successfully.

6. Click **Done** to exit out of the **Unextend Resource Hierarchy menu**.

6.5.3.13. IP User System Setup

When the IP Recover Kit software switches an IP resource from one server to another, the MAC address associated with the switched IP address changes because the interface changes. Each router and user system on the LAN must reflect this change in its ARP table before it can contact the IP address at its new location. In certain operating systems, when a new IP address is added to a network interface, an ARP packet is automatically sent out by the operating system to update all clients' ARP tables on the subnet. This feature does not exist in Linux. LifeKeeper therefore must send out an ARP packet after adding a switchable IP address to an interface to force this client ARP cache update.

TCP/IP implementations differ in their ability to implement the required ARP updates in response to this ARP packet. The following list describes some important cases:

- **Full Linux TCP/IP implementation**. Fully functional TCP implementations in Linux and most other operating systems support ARP cache updates when the systems receive an ARP request packet. LifeKeeper uses this feature, as described above, to force ARP cache updates on such systems.

- **ARP cache**. User systems that do not support the ARP refinements but do support an ARP cache usually have a timer associated with the cache to maintain some level of currency. For some implementations, decreasing the timer value can minimize the time required for that particular user system to reflect the changed address mapping. If the number of users on the LAN is small, this option may be acceptable. For other systems, decreasing the timer value may not be necessary. For example, the TCP implementation shipped with Windows NT uses a ten second timer value, so no change in timer value would be needed.

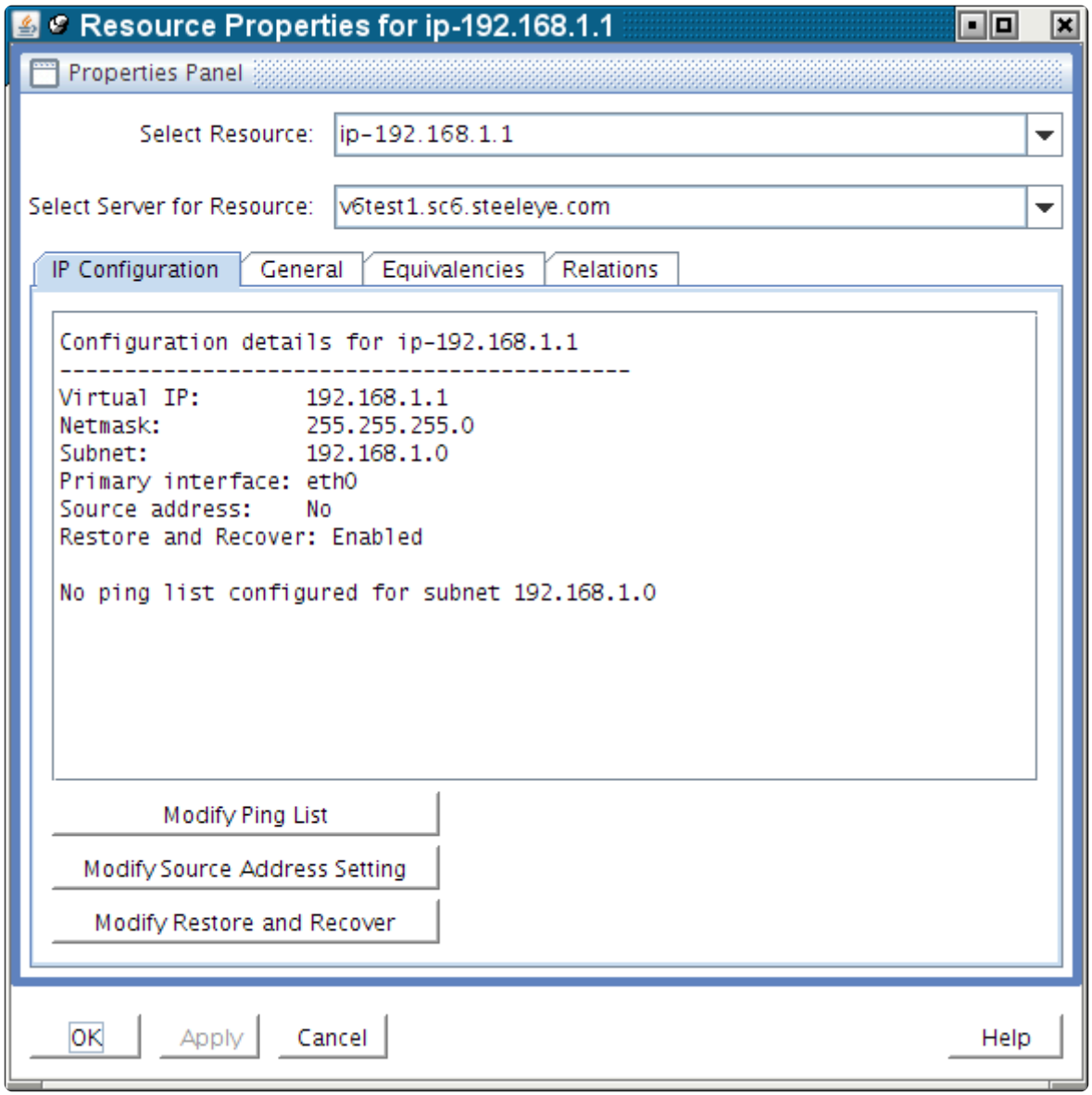
- **Static address mapping**. For systems without a dynamic ARP cache or those where cache timing is not tunable, routers can be used to handle mapping changes. Such user systems would access the IP resource subnet by way of a router (gateway). In this configuration, cache update is needed only for the routers directly connected to the resource subnet and no changes are needed on the user systems themselves.

6.5.3.14. Viewing and Editing IP Configuration Properties

The **IP Configuration Properties** page allows you to view the configuration details for a specific IP resource, as well as to modify a number of selected configuration items.

To access the **IP Configuration Properties** page, from the LifeKeeper GUI menu select **Edit**, then **Resource**. From the dropdown menu, select **Properties**. Then select the resource for which you want to view properties from the **Resource list** and the server for which you want to view that resource from the **Server list**. You can also access the properties page using the context-sensitive menu that appears when you right-click on a specific IP resource instance.

Below is an example of the properties page that will appear for an IP resource.



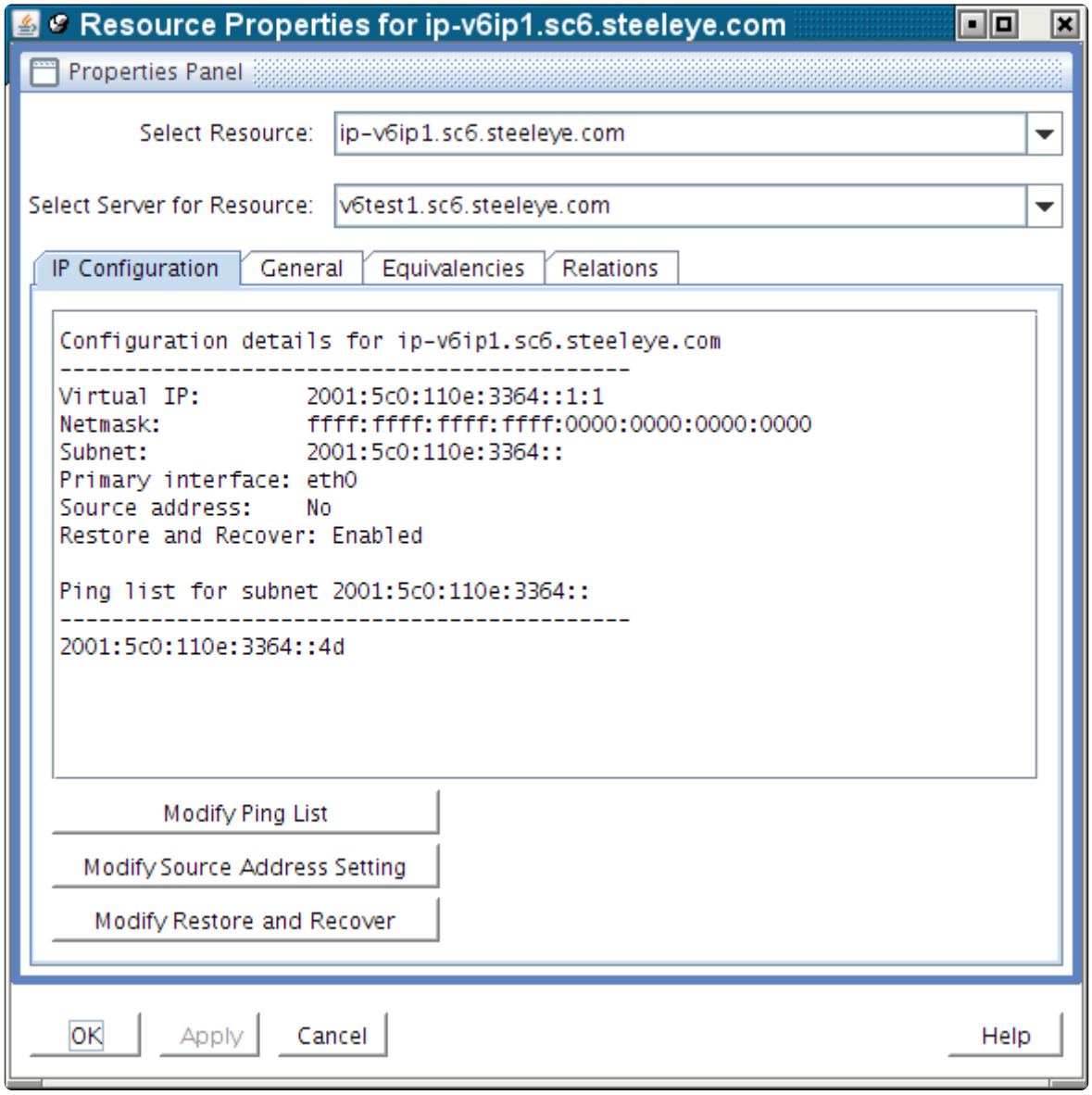
The resulting properties page contains four tabs. The first of those tabs, labeled **IP Configuration**, contains configuration information that is specific to IP resources. The remaining three tabs are available

for all LifeKeeper resource types.

The IP Configuration tab displays the following information elements about the selected IP resource.

Virtual IP	The virtual IP address associated with this IP resource.
Netmask	The netmask for the virtual IP address. This value determines how much of the address makes up the subnet portion.
Subnet	The logical subnet address for the virtual IP address, including the number of bits included in the subnet portion of the address.
Primary interface	The network interface on which the virtual IP address should be configured when it is active.
Source address setting	Specifies whether the virtual IP address should be configured as the source address for outbound IP traffic onto its associated subnet.
Ping List	The optional list of IP addresses to be pinged during IP health checks for this IP resource (and others on the same subnet), as an alternative to the normal broadcast ping mechanism.

In the example above, there is no Ping List configured for this IP resource. When a Ping List is configured, the resulting properties page looks like the following example.

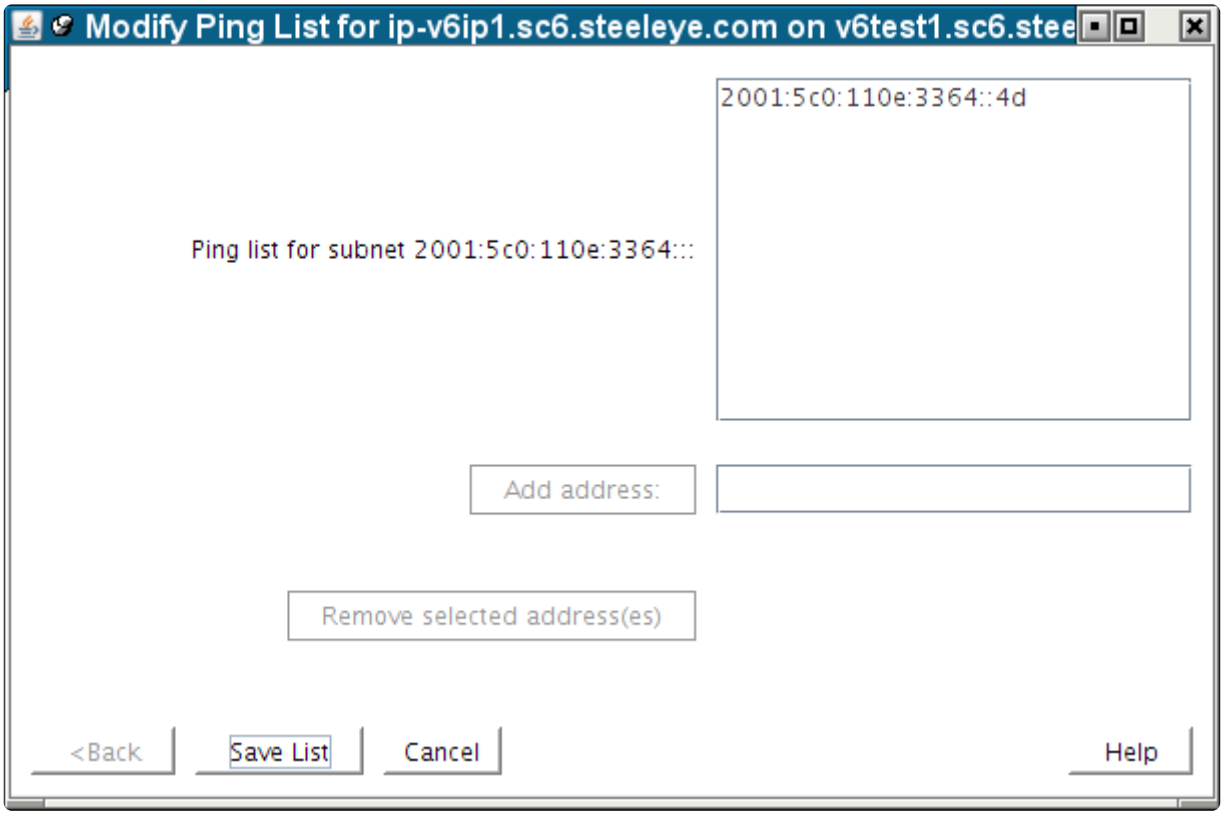


The **Modify Ping** List and **Modify Source Address Setting** buttons can be used to perform modifications to those configuration items, as described in the sections below.

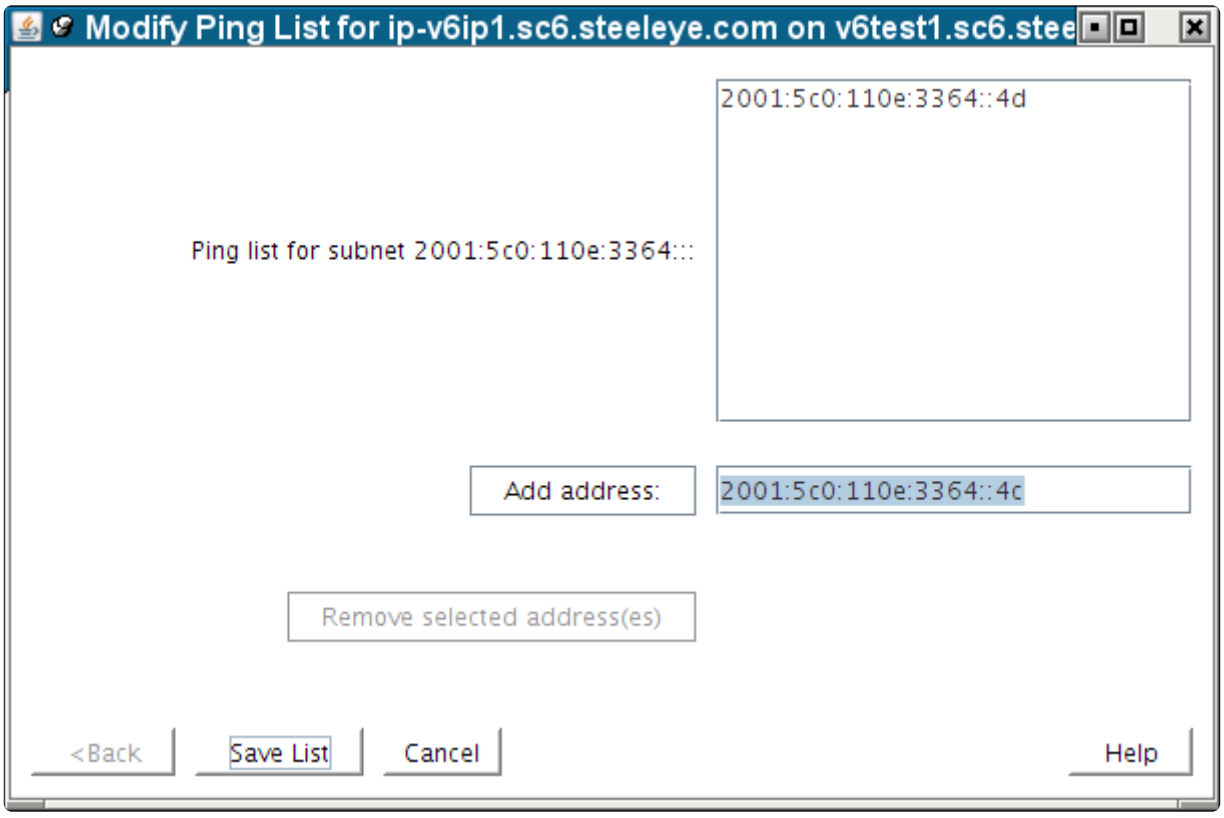
Modifying the Ping List

For a description of the use and function of the Ping List for an IP resource, see the topic IP Resource Monitoring.

To create a Ping List for an IP resource, or to modify an existing list, click the **Modify Ping List** button on the **IP Configuration properties page**. This brings up a dialog window similar to the following example.

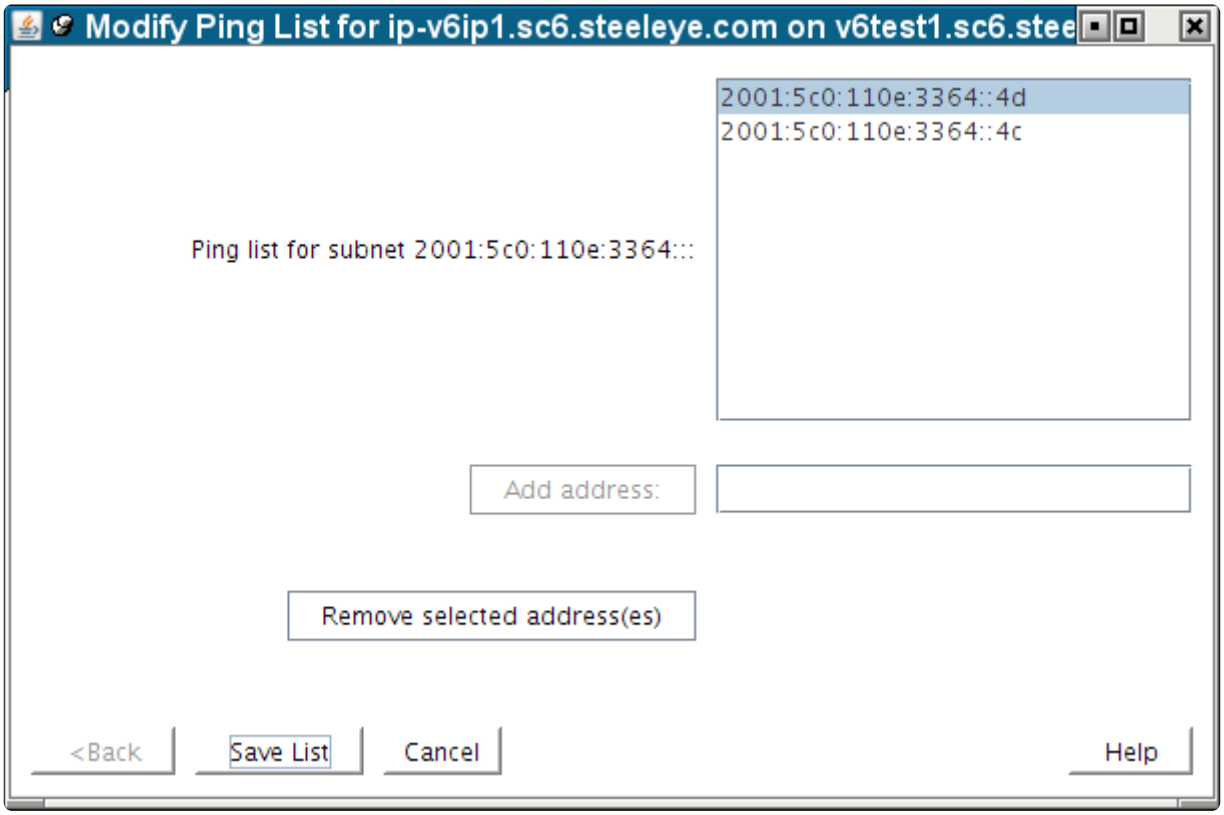


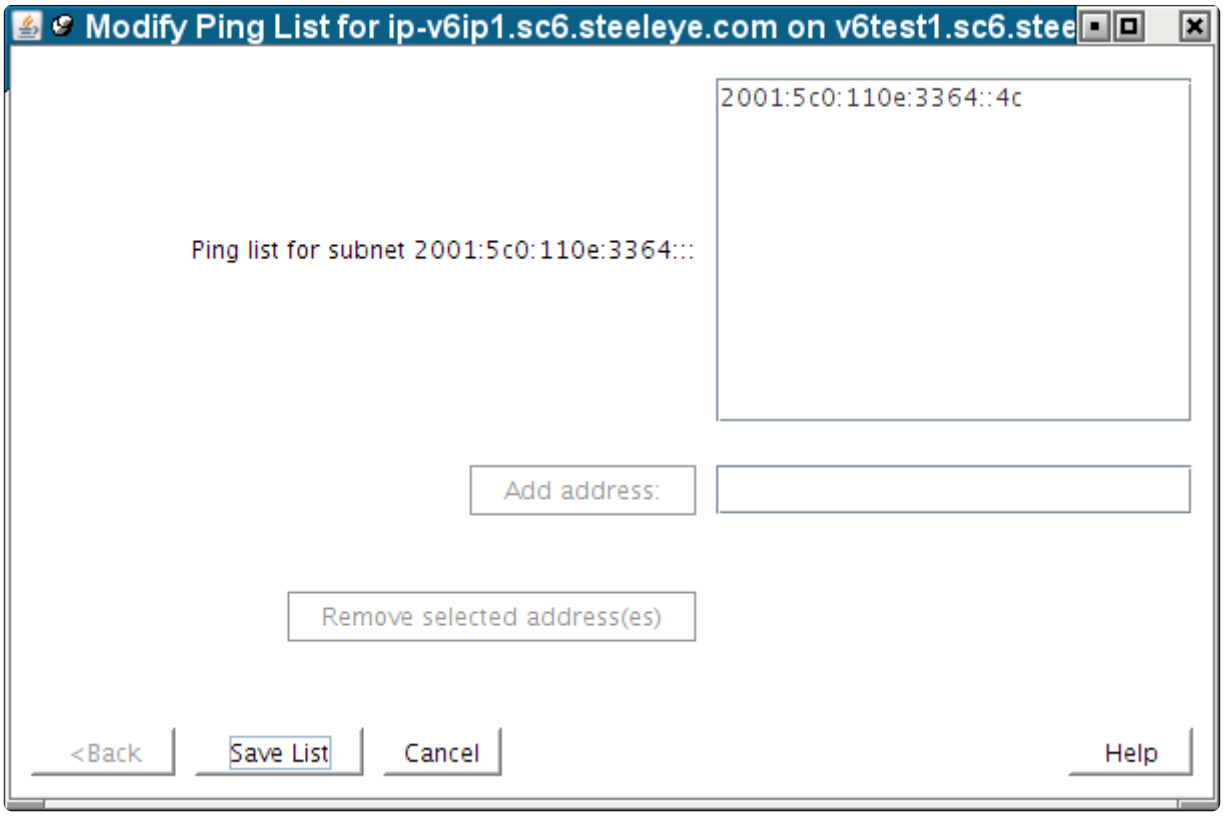
To add an address to the Ping List, type the address in the field next to the **Add address**: button, and push the button, as shown in the following two images. Note that the **Add address**: button is grayed out until you begin typing an address in the field.



To remove one or more addresses from the Ping List, click to select the addresses to be removed and click the **Remove selected address(es)** button. The **Remove selected address(es)** button is also

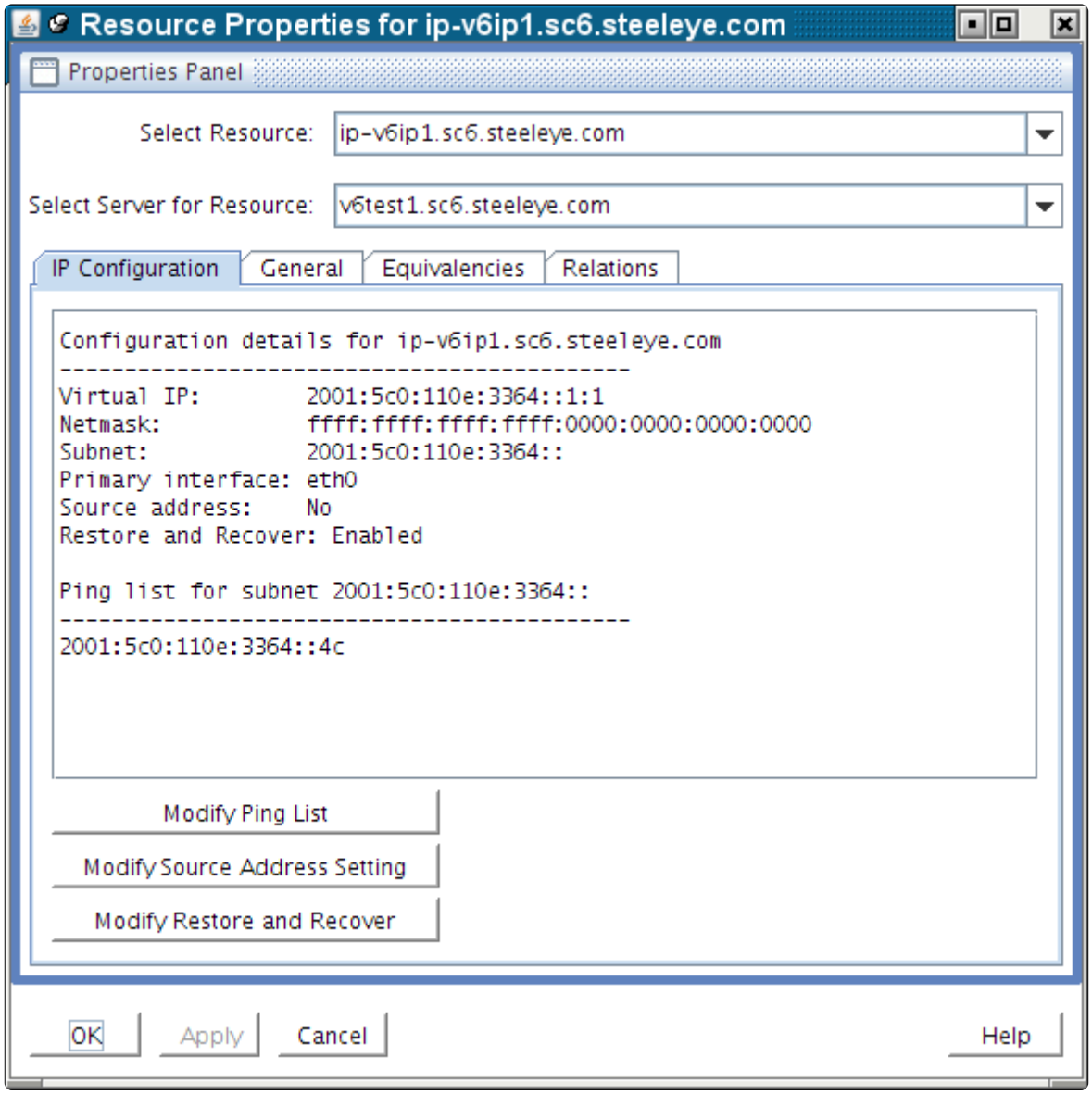
grayed out until at least one address in the list has been selected.





To save the modified list, click **Save List**. This produces the following confirmation window.

Click **Done** to close the window, bringing you back to the **IP Configuration properties page**, where you can see the modified Ping List.



Important Notes About Using a Ping List

A *Ping List* for an IP resource is unique to the LifeKeeper system on which it is configured. If the IP resource is extended to another LifeKeeper system after the Ping List has been created, the Ping List will be copied to the other system as a part of the extension. However, if the IP resource has already been extended, the Ping List must be configured individually for each system on which the IP resource is defined. Ping List modifications can be made to an IP resource regardless of its state, so there is no need to perform switchovers of the IP resource in order to modify the Ping List on each system.

If there are multiple IP resources defined on the same logical subnet, all of those IP resources share a common Ping List. This is reflected in the IP Configuration properties page and the dialogs associated with modifying the Ping List, where the list is identified as being for the subnet associated with the IP resource.

Once a Ping List has been defined for an IP resource, all health checks for that resource will use the Ping List mechanism rather than the default broadcast ping mechanism. To revert back to the broadcast ping mechanism, you must delete the Ping List by removing all of the address entries in the list.

LifeKeeper performs no validation of the IP addresses entered into a Ping List, other than ensuring the validity of the formatting of the addresses. It is important that you ensure that the addresses you are entering actually exist on your network, can be pinged from the LifeKeeper systems, and are expected to be active at all times. You should not choose addresses that exist on the LifeKeeper systems themselves, because local pings to such addresses may be successful regardless of the actual status of the network interface on which the monitored IP resource is defined.

As mentioned above, the definition of a Ping List for an IP resource on a given system causes LifeKeeper to automatically use the Ping List mechanism rather than a broadcast ping for that resource and all other IP resources on the same subnet. It is not necessary to disable the broadcast ping mechanism using the NOBCASTPING setting described in the Adjusting IP Recovery Kit Tunable Values topic. However, if you have a configuration in which there are no systems available on the network to respond to a broadcast ping, you may have to use the NOBCASTPING=1 setting initially in order get the IP resource created, before you can then define a Ping List using the procedure described above. Once the Ping List has been created, you can revert back to the default NOBCASTPING=0 setting.

The contents of Ping List remains even after IP resource is deleted. Please note that the old Ping List setting will remain when IP resource with the old subnet address is created after deleting IP resource.

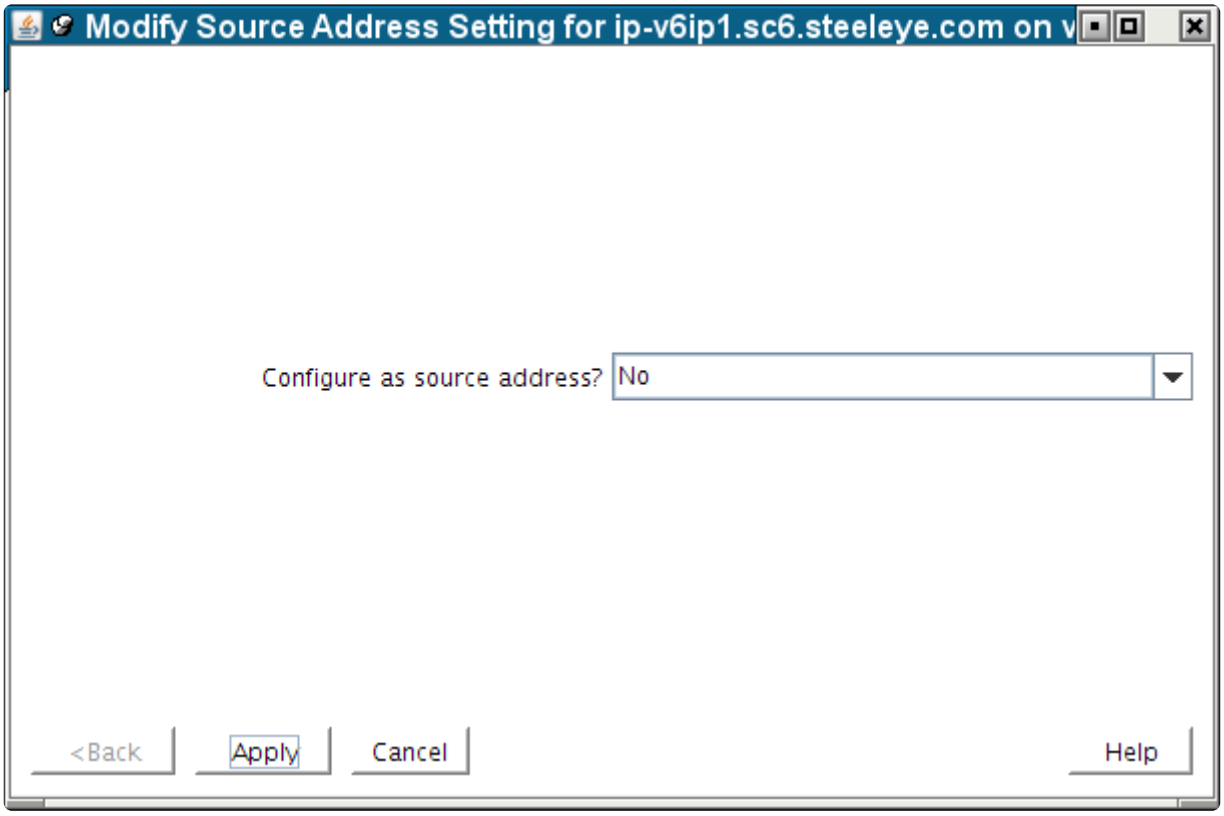
Modifying the Source Address Setting

The Source Address Setting for an IP resource determines whether the virtual IP address should be used as the source address for outgoing traffic onto the subnet associated with that IP resource, when the IP resource is in-service. This value defaults to No, which means that if the virtual IP address is on the same subnet as the primary IP address for the network interface, outgoing traffic onto the subnet will normally appear to be coming from that primary IP address. This is usually appropriate for most configurations, because the virtual IP address is generally used as an incoming connection point for clients, meaning that all connections in which the virtual IP address is used are initiated as incoming traffic.

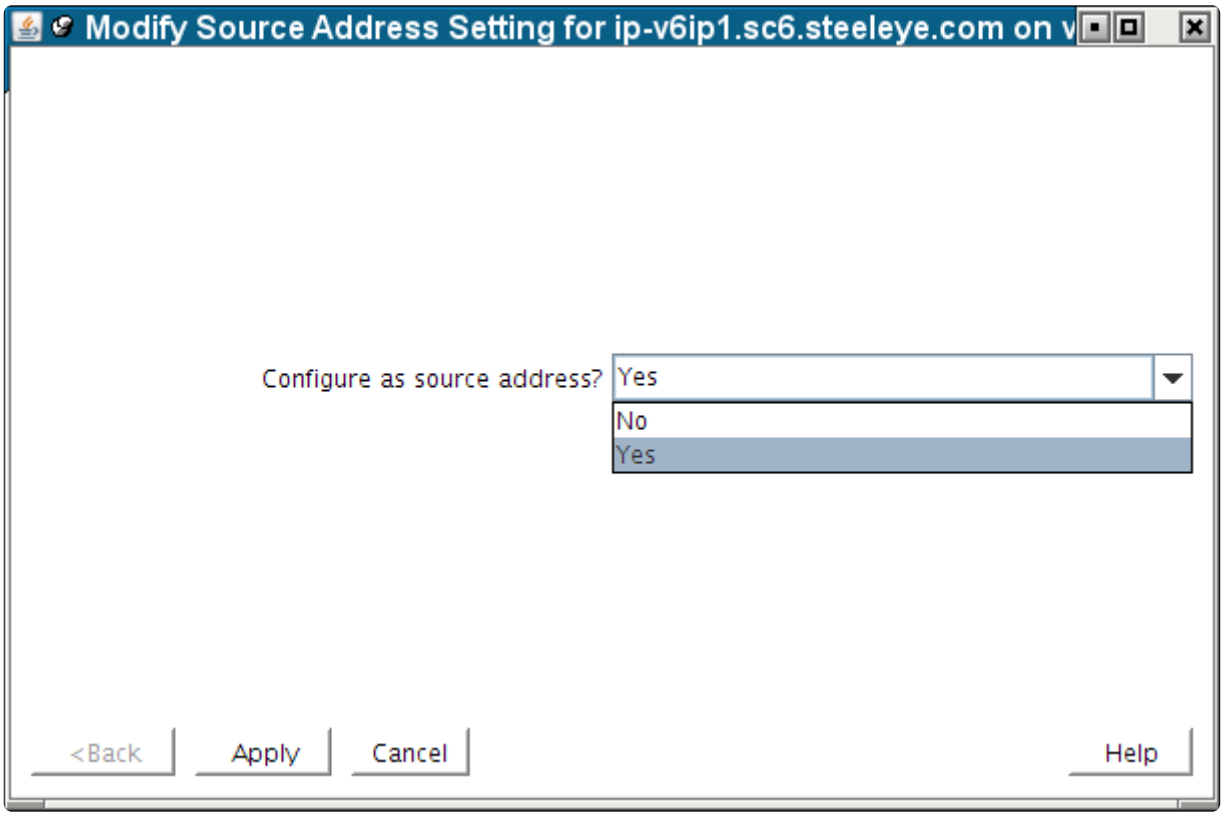
However, there may be situations or configurations in which it is important for connections initiated from the LifeKeeper system to appear to be coming from the virtual IP address. By changing the Source Address Setting for the IP resource to Yes, when the IP resource is brought in-service, the TCP/IP routes on the system are modified such that this will be the case.

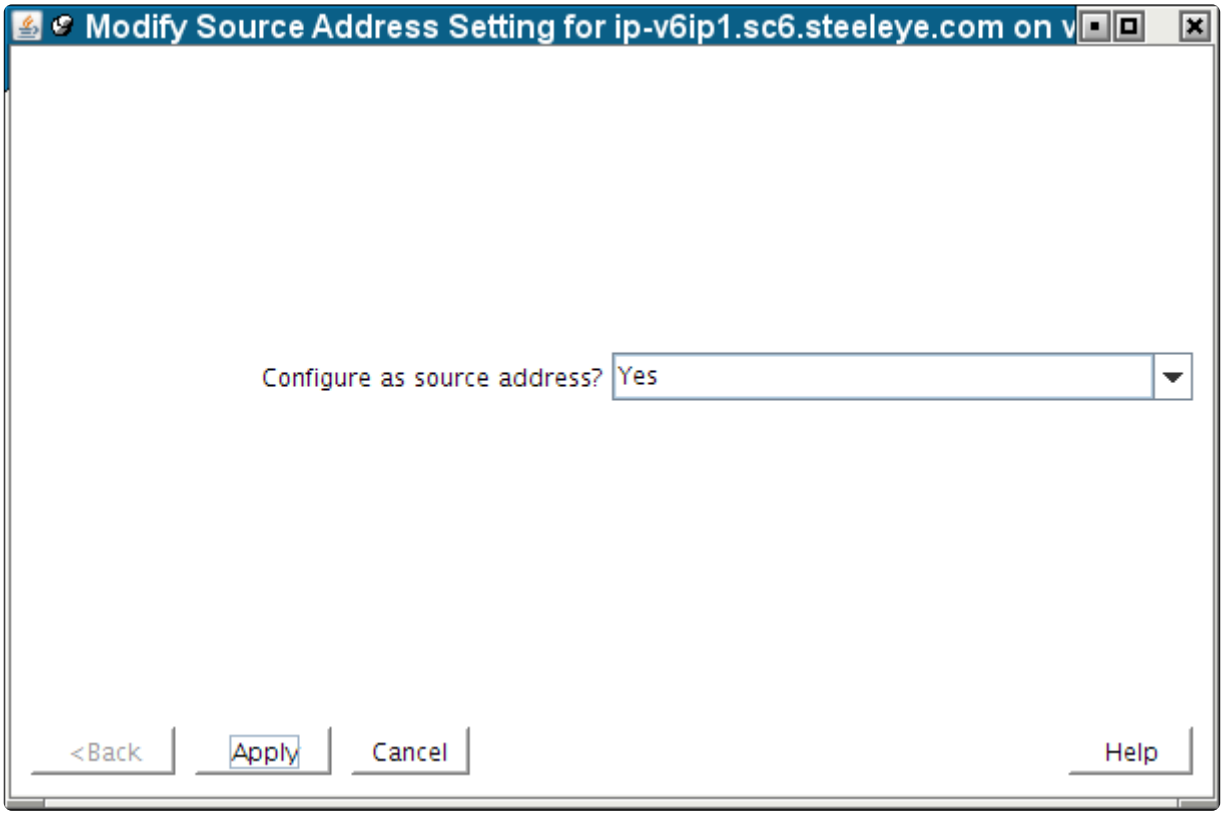
Note that if the virtual IP address is on its own distinct logical subnet from the permanent IP addresses on the system, all outgoing traffic onto that subnet will always come from the virtual IP address without any modifications to the Source Address Setting.

To modify the Source Address Setting for an IP resource, click the Modify Source Address Setting button on the IP Configuration properties page. This brings up a dialog window similar to the following example.

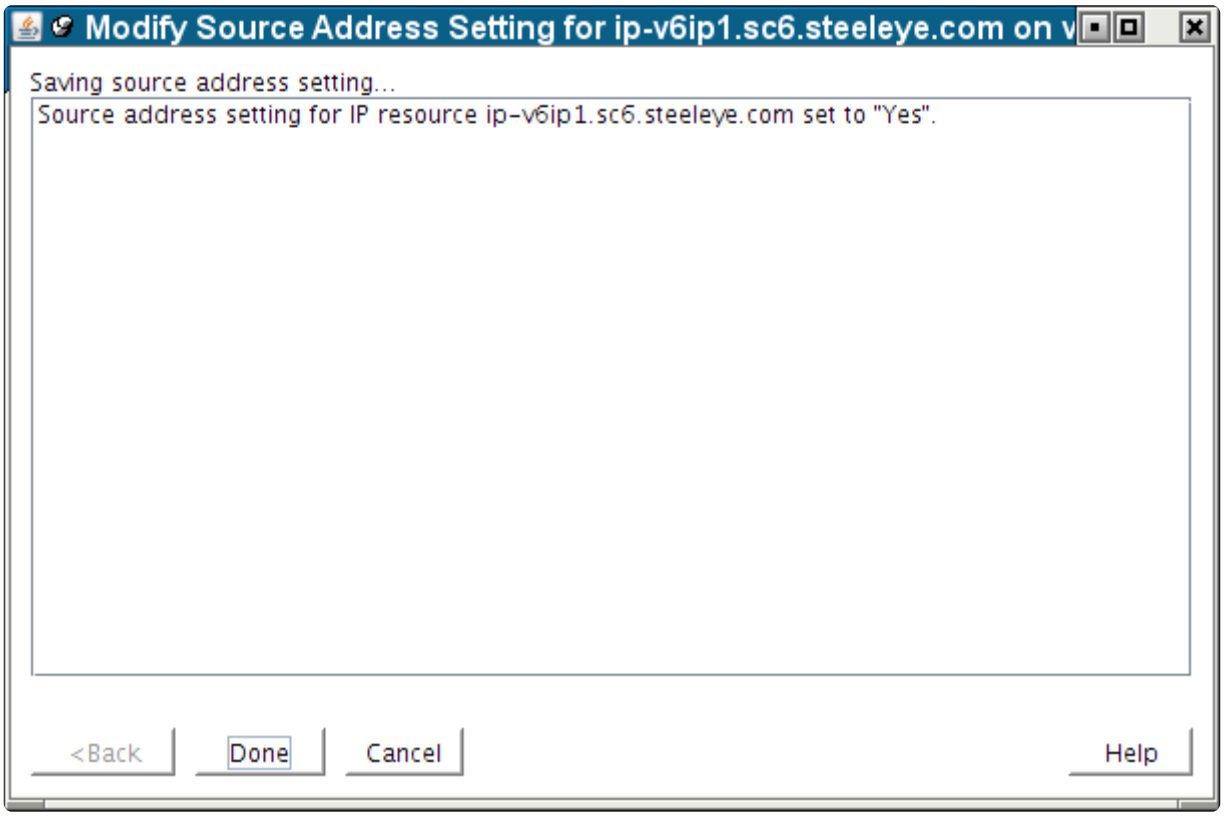


To change the setting, use the dropdown list to select the new value, either **Yes** or **No**.

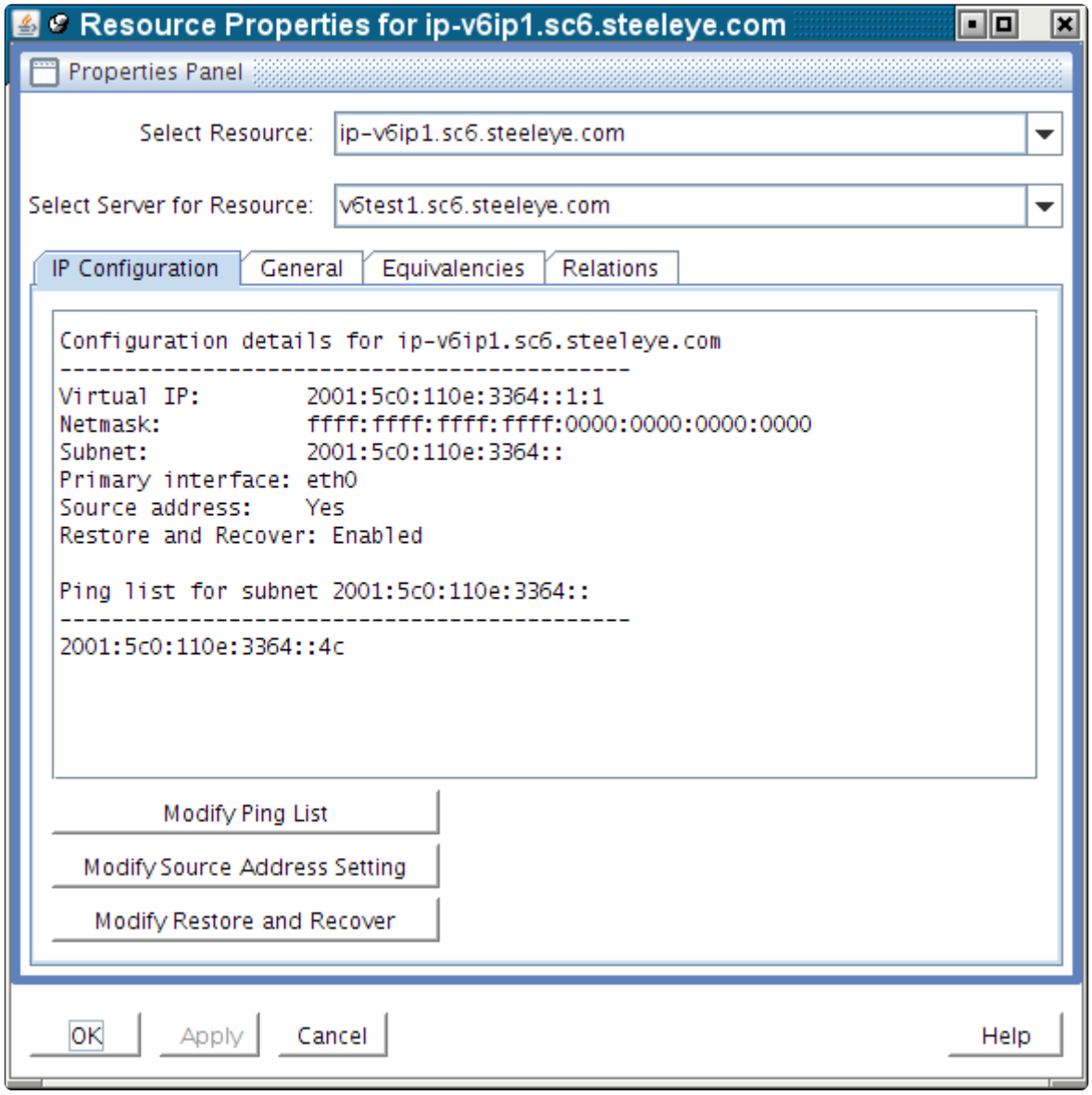




Click **Apply** to save the new setting. This produces the following confirmation window.



Clicking **Done** will close the window and take you back to the **IP Configuration properties page**, where you can see the modified setting.



Important Notes About the Source Address Setting

The **Source Address Setting** for an IP resource is unique to the LifeKeeper system on which it is configured. If the IP resource is extended to another LifeKeeper system after the **Source Address Setting** has been modified, the setting will be copied to the other system as a part of the extension. However, if the IP resource has already been extended, the **Source Address Setting** modification must be made individually for each system on which the IP resource is defined.

It only makes sense for at most one IP resource on a given subnet to have its Source Address Setting set to Yes, because only a single IP address can actually be the source address for outgoing traffic onto the subnet. If there are multiple IP resources on the same subnet with a setting of Yes, the most recent resource to be brought in-service will override any others and become the source address for outgoing traffic onto the subnet.

The **Source Address Setting** only affects the local TCP/IP configuration when the IP resource is brought into service. So if the resource is already active when the setting is changed, the resource must be taken out-of-service and then back in-service before the change is reflected in the TCP/IP configuration.

The **Source Address Setting** only affects IPv4 addresses. This setting has no effect on an IPv6 address.

Modifying Restore and Recover

This feature allows a user to choose to **Enable** or **Disable** the default restore and recovery behavior for an existing IP address resource. If configured with the **Enable** option, the IP address will be brought in-service as normal and the regular monitoring and recovery process will occur. The **Enable** option is the current default behavior for an IP address restore.

If the **Restore and Recover** option is set to **Disable**, the resource will come in-service, but the IP address will not be brought active on the network or network adapter. This setting allows hierarchies in a WAN environment that depend on an IP to be brought in-service on the Disaster Recovery (DR) system where it may be difficult to configure the IP on the DR system due to the WAN configuration.

This setting can be selected after the resource is created and extended.

Important consideration for Active IP addresses (ISP): Setting the action to **Disable** on an ISP and active IP address does not take the active IP out of service.

6.6. MySQL Recovery Kit Administration Guide

The SIOS Protection Suite for Linux MySQL Recovery Kit provides an easy way to add LifeKeeper fault-resilient protection for MySQL resources and databases. This enables a failure on the primary database server to be recovered on a designated backup server without significant lost time or human intervention.

SIOS Protection Suite Documentation

The following SPS product documentation is available from SIOS Technology Corp.:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation (also available from the **Help** menu within the LifeKeeper GUI)

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is available on the SIOS Technical Documentation website.

6.6.1. MySQL Recovery Kit Hardware and Software Requirements

Before you can install and set up the recovery software, your server must meet certain hardware and software requirements. You should refer to the SPS for Linux Installation Guide for specific instructions on how to install or remove the LifeKeeper MySQL Recovery Kit.

Be sure that your configuration meets the following requirements:

- **Servers**. The Recovery Kit requires two or more LifeKeeper supported computers configured in accordance with the requirements described in SPS for Linux Technical Documentation and the SPS for Linux Release Notes.
- **LifeKeeper software**. You must install the same version of LifeKeeper software and any patches on each server. Please refer to the SPS for Linux Technical Documentation and the SPS for Linux Release Notes for specific LifeKeeper requirements.
- **LifeKeeper IP Recovery Kit**. This kit is required if remote clients will be accessing the MySQL database. You must have the same version of this Recovery Kit on each server.
- **IP network interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

> **Note**: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation and so forth.

- **TCP/IP software**. Each server also requires the TCP/IP software.
- **MySQL software**. Each server must have the MySQL software installed and configured prior to configuring LifeKeeper and the LifeKeeper MySQL Recovery Kit. The same version should be installed on each server. Consult the SPS for Linux Release Notes or your sales representative for the latest release compatibility and ordering information.

6.6.2. MySQL Recovery Kit Configuration

This section contains definitions and examples of typical LifeKeeper MySQL configurations and information you should consider before you start to configure MySQL.

Please refer to the Resource Hierarchies section of the SPS for Linux Technical Documentation for instructions on configuring your LifeKeeper Core resource hierarchies.

6.6.2.1. Configuration Considerations for MySQL

Below are some specific considerations you need to think about concerning your LifeKeeper MySQL environment.

To operate MySQL database services on the primary and backup servers, file systems and disk partitions must be accessible from each server. Before you can begin configuring the MySQL Recovery Kit, be sure you have completed the following preliminary steps and have tested/run the databases on each server. In the instructions below, the user “mysql” refers to the operating system user that will start the MySQL server.

1. Install the MySQL server and client components on all servers. Be sure that all of the servers are running the same version of the MySQL client and server components. The MySQL executables can be located on a local or shared drive.

   Note: If you use Red Hat Software Collections and need to export the X_SCLS environment variable in order to run a specific version of MySQL with LifeKeeper, then set the X_SCLS environment variable via `/etc/default/LifeKeeper` by adding the line X_SCLS=VERSION to the file (i.e. X_SCLS=mysql55). This is typically only the case if you want to enable MySQL 5.5 which is included in RHEL 5.10 (MySQL 5.0 is enabled as the default).

2. If mysqld is running on any of the servers on the socket and/or port where you wish to run the LifeKeeper protected MySQL database server, stop each MySQL server using the `mysqladmin` command.

3. Move the contents of the MySQL data directory to a shared location. By default, the MySQL data directory is installed on a local drive. This location depends on the distribution mechanism. The binary RPM installs the data directory at `/var/lib/mysql`. (Be sure that only the contents are moved and the directory remains intact. This allows the MySQL database server to write logs in this directory, if necessary. Make sure that the “mysql” user described in step 4 has permissions to write the logs to this location.)

4. If the installation process did not create the Linux user “mysql”, create this user. For security reasons, the MySQL server should not be run as “root.” (Refer to the MySQL Administration Guide for a full discussion of the security issues.) Make sure that “mysql” is the only user with read/write permissions in the database directories. The “mysql” user and group should be created on all servers. The user ID and group ID must be the same on all servers.

5. **IMPORTANT**: A server started by /etc/rc.d/init.d/mysql cannot be under LifeKeeper protection. In addition, the server can not use the same port number or socket as a server under LifeKeeper protection.

6. It is recommended that the socket be written to the data directory on the shared disk. If the socket will be written to a local disk, make sure the path exists on all LifeKeeper servers where your

hierarchy will exist. Make sure that the user “mysql” has permissions to write the socket to this location.

7. Start the MySQL server using the mysql daemon startup command appropriate for your configuration. For configurations defining a single instance in the *my.cnf file*, use the command:

   ```
   <start command> –user=mysql –socket=<socket> –port=<port number>

   –datadir=<path to the data directory> –log &
   ```

   The <start command> for mysql versions 3.x is `safe_mysqld`, and the command for version 4.x is `mysqld_safe`.

   For configurations using the mysqld Group feature in the *my.cnf file*, use the command:

   ```
   mysqld_multi start <group number>
   ```

   The <group number> represents the numerical instance defined in the *my.cnf* file for the mysqld Group. For more information on using mysqld groups with LifeKeeper, see: Using mysqld Groups with LifeKeeper.

   sytemctl command must be utilized when MySQL (v5.7.6 or later) is set up to use Systemd in the distribution with Systemd. For the details, refer to “Consideration about the use in Systemd environment”.

8. Create a MySQL database user named “mysql”. Give this user a password and grant the user “shutdown” permissions. This only has to be done on one server. (Refer to the MySQL Administration Guide for details on creating users and granting permissions).

9. Copy the sample my.cnf configuration file to the desired location (*/etc* or */<datadir>*). This file contains options for the database server and for client programs.

   The file can be located in either the *MySQL data* directory or the */etc* directory. The */etc/my.cnf* file contains global options. Place the *my.cnf* file in */etc* if only one database will run on the machine at any given time (i.e. an Active/Standby configuration) or if you are using the mysqld Group feature (see Using mysqld Groups with LifeKeeper). If the file is located in /etc, you must copy it to each LifeKeeper backup server. The *my.cnf* file in the data directory should contain server-specific options. For multiple servers and Active/Active configurations, this file must be stored in the data directory for each resource instance unless you are using the mysqld Group feature (see Using mysqld Groups with LifeKeeper).

   Note: The *my.cnf* file should not exist in both the */etc* and */* locations if both copies will contain server specific options. If a *my.cnf* file containing server specific options is located in */etc* along with a protected *my.cnf* file installed in the */* potential conflicts may result. Refer to the MySQL documentation on configuring global settings and server specific options.

Add or edit the following entries:

a. In the “client” section of the file, specify the user and the password that should be used for connections.

```
[client]
user =clientuser
password =password
.
.
.
```

b. In the “mysqld” section of the file, specify the socket and port that should be used for connections, as well as the pid-file location for the mysqld process. The user variable should specify the operating system user that will start the mysqld process.

```
[mysqld]
socket =/home1/test/mysql/mysql.sock
port =3307
pid-file =/home1/test/mysql/mysqld.pid
user =osuser
```

Note: Make sure this file is properly protected and owned by the user “mysql.”

Note: Once the MySQL hierarchy is created, if you need to change any of the information in the `my.cnf` file, you must stop the mysql server instance by taking the hierarchy out-of-service (i.e. the OSU state) before making changes.

Note: The above example `my.cnf` configuration describes a single database instance `mysqld`. See Using mysqld Groups with LifeKeeper for configuration examples using mysqld groups.

Note: “include” directive is not supported. All the setups must be described in a single `my.cnf` file.

6.6.2.2. Client Configuration Considerations for MySQL

Following are some configuration considerations for MySQL database clients:

- If clients will connect from remote hosts, create an IP address under LifeKeeper to be used for client connections.
- Clients must be configured to connect to the database server through a LifeKeeper-protected IP address.
- If the clients will connect through a domain name instead, create an entry in each client's hosts file for the protected IP address, or configure the name in DNS. Test the protected IP address by pinging it from all clients and all LifeKeeper servers in the cluster.
- Although each user can have a *my.cnf* file in the home directory of their machine, LifeKeeper only uses the *my.cnf* file located in the */etc* directory or the data directory. The *my.cnf* file stores the client connection information (i.e. the port, socket identification, user and password).

6.6.2.3. MySQL Configuration Requirements

Each of the examples involves one or two database instances: databaseA and databaseB. The Database Tag names are arbitrary names that describe these database instances to LifeKeeper. The word on and the system identifier that follows provide clarification but are not required. The default tag name suggested by LifeKeeper is mysql or mysql for configurations using mysqld Groups (see Using mysqld Groups with LifeKeeper). To understand the configuration examples, keep these configuration requirements in mind:

- **LifeKeeper hierarchy**. When performing LifeKeeper administration, the primary hierarchy refers to the hierarchy being built on the server you are administering. For the configuration diagrams, the information entered in the first administration screen is from the perspective of Server 1. When a second screen is shown, it refers to the hierarchy being built while administering the second server. In the configuration examples, the second server is Server 2.

- **Shared disk locked by one server**. When you use LifeKeeper, one server reserves shared storage resources that are under LifeKeeper protection for use. This is done using SCSI reservations. If the shared device is a disk array, an entire LUN is reserved; if a shared device is a disk, then the entire disk is reserved. This prevents inadvertent corruption of the data by other servers in the cluster. When a server fails, the highest priority backup server breaks the old reservation and establishes its own reservation, locking out all other servers.

- **Data Directory on shared disk**. In order for the LifeKeeper MySQL Recovery Kit to function properly, the data directory (datadir) of the database instance must always be on a shared disk. The data directory must be on a file system. The file system must be mountable from both the primary and backup servers. The data directory (datadir) can also exist on replicated or network attached storage.

6.6.2.4. MySQL Configuration Examples

The examples in this section show how MySQL database instances can be configured. Each diagram shows the relationship between the type of configuration and the MySQL parameters. Each configuration also adheres to the configuration rules and requirements described in this documentation that ensure compatibility between the MySQL configuration and the LifeKeeper software.

This section describes the configuration requirements and then provides these configuration examples:

- Active/Standby
- Active/Active

The examples in this section are only a sample of the configurations you can establish, but understanding these configurations and adhering to the configuration rules will help you define and set up workable solutions for your computing environment.

Configuration Requirements

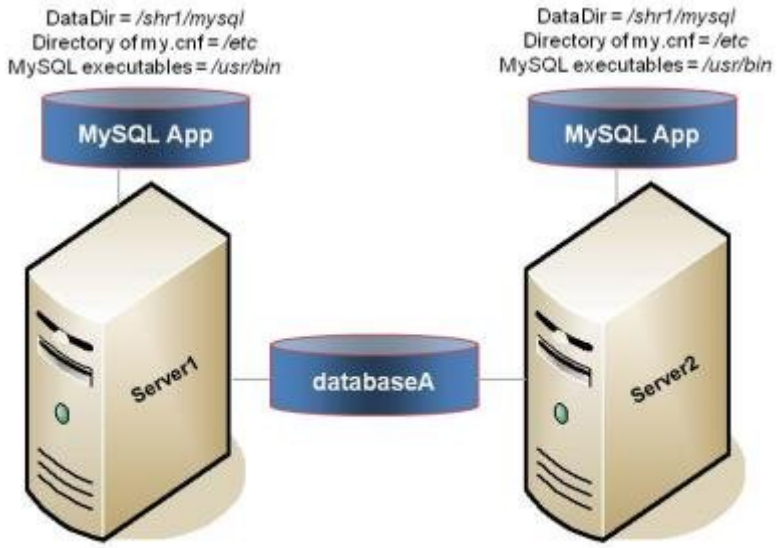
Example 1 – Active/Standby Configuration

Example 2 – Active/Active Configuration

6.6.2.5. Active/Standby MySQL Configuration

This section provides an example of an active/standby configuration. In this configuration, Server 1 is considered active because it has exclusive access to the database. Server 2 does other processing. If Server 1 fails, Server 2 gains access to the database, and LifeKeeper re-establishes the database operations.

Figure 1. Active/Standby Configuration, Example 1



Configuration Notes:

- Both servers use the MySQL data directory (which includes the database (databaseA)) on a shared disk.
- The path to the MySQL data directory is the same on both servers.
- The *my.cnf* configuration file is located on a local disk in */etc*.
- The MySQL executables are located on a local drive on each server in */usr/bin*.
- Server 2 cannot access files and directories on the shared disk while Server 1 is active.

Creating a resource hierarchy on Server 1:

Server:	Server 1
Directory of *my.cnf* File Location:	*/etc*
Directory of MySQL Executables Location:	*/usr/bin*

Database Tag	mysql-on-server1

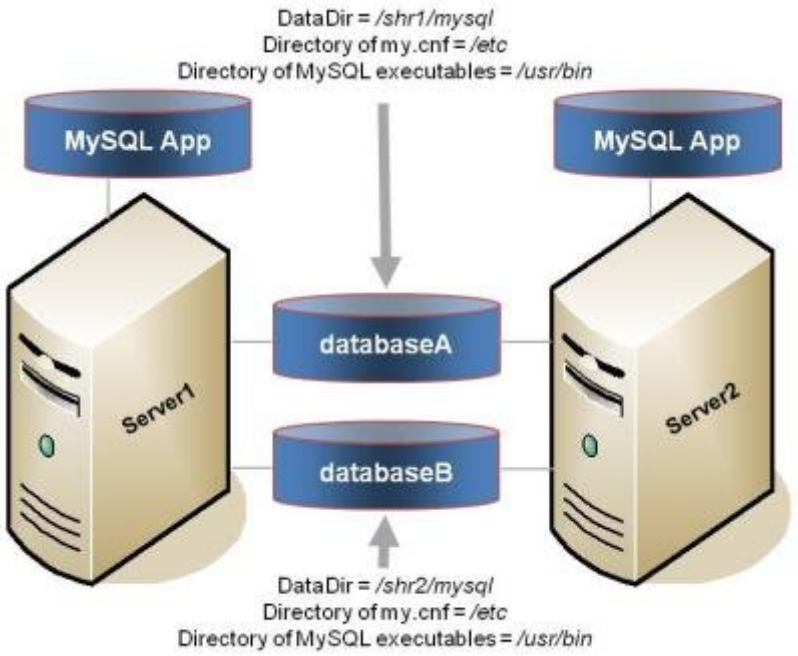
Extending a resource hierarchy to Server 2:

Template Server:	Server 1
Tag to Extend	mysql-on-server1
Target Server	Server2
Target Priority:	10
Directory of *my.cnf* File Location:	*/etc*
Directory of MySQL Executables Location:	*/usr/bin*
Database Tag	mysql-on-server2

6.6.2.6. Active/Active MySQL Configuration

An active/active configuration consists of two or more servers actively running a different database instance with each serving as a backup for each other. The database instances must be on different shared physical disks. For LifeKeeper configurations supporting multiple MySQL database instances (of the same or different versions), SIOS recommends that the mysqld Group feature be used for versions of MySQL that support this feature. For these configurations, the *my.cnf* configuration file will reside in */etc*. For MySQL versions that do not support the mysqld Group feature, the *my.cnf* configuration file must reside in the MySQL data directory shared file system for each database instance (e.g. in Figure 2 below, */shr1/mysql* and */shr2/mysql*).

Figure 2. Active/Active Configuration, Example 2



Configuration Notes:

- Each server uses a different MySQL data directory (which includes the database instances (database A and database B) on different shared disks
- The path to the MySQL data directory is different for each instance defined on the server.
- The *my.cnf* configuration file is located in */etc* and contains mysqld group sections for each database instance. Each section defines a unique MySQL data directory, port and socket for that database instance. The *my.cnf* configuration file must be kept in sync on all nodes in the cluster. For systems running versions of MySQL that do not support mysqld Groups, the *my.cnf* configuration file for each of the database instances is located on the shared drive in the data directory for the database instance. Each configuration file defines a unique MySQL data directory, port and socket definition for that database instance.

- The MySQL executables are located on a local drive on each server in */usr/bin*.
- Initially, Server 1 runs databaseA and Server 2 runs databaseB. In a switchover situation, one server can run both databases.

Creating the first resource hierarchy on Server 1:

Server:	Server 1
Directory of *my.cnf* File Location:	*/etc*
Directory of my MySQL Executables Location:	*/usr/bin*
Database Tag:	mysql-shared.example.instance1

Extending the first resource hierarchy to Server 2:

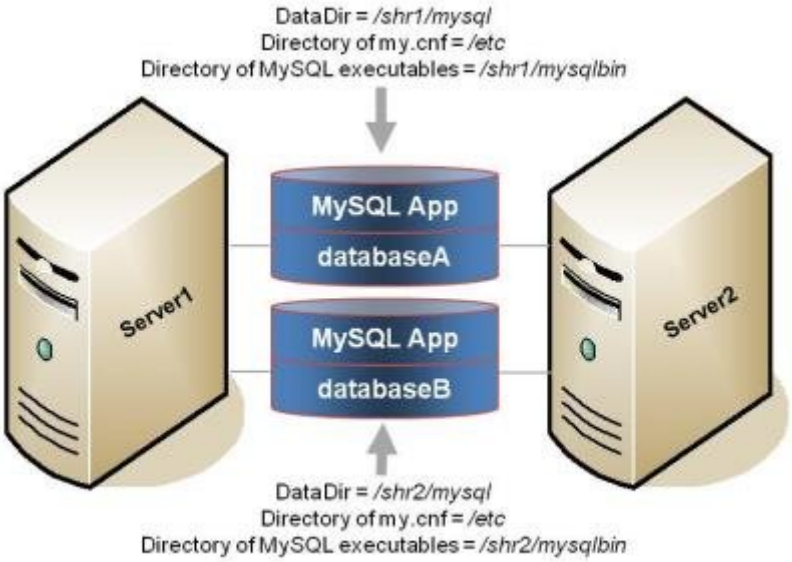
Template Server:	Server 1
Tag to Extend:	mysql-shared.example.instance1
Target Server:	Server2
Target Priority:	10
Directory of *my.cnf* File Location:	*/etc*
Directory of my MySQL Executables Location:	*/usr/bin*
Database Tag:	mysql-shared.example.instance1

Creating the second resource hierarchy on Server 2:

Server:	Server 2
Directory of *my.cnf* File Location:	*/etc*
Directory of my MySQL Executables Location:	*/usr/bin*
Database Tag:	mysql-shared.example.instance2

Extending the second resource hierarchy to Server 1:

Template Server:	Server 2
Tag to Extend:	mysql-shared.example.instance2
Target Server:	Server1
Target Priority:	10
Directory of *my.cnf* File Location:	*/etc*
Directory of my MySQL Executables Location:	*/usr/bin*
Database Tag:	mysql-shared.example.instance2



Configuration Notes:

- Each server uses a different MySQL data directory (which includes the database instances (database A and database B) on different shared disks
- The path to the MySQL data directory is different for each database instance defined on the server.
- The *my.cnf* configuration file is located in */etc* and contains mysqld group sections for each database instance. Each section defines a unique MySQL data directory, port and socket for that database instance. The *my.cnf* configuration file must be kept in sync on all nodes in the cluster. For systems running versions of MySQL that do not support mysqld Groups, the *my.cnf* configuration file for each of the database instances is located on the shared drive in the data directory for the database. Each configuration file defines a unique MySQL data directory, port and socket definition for that database instance.
- There is a copy of the MySQL executables on each of the shared disks that contains the data directories.
- Initially, Server 1 runs databaseA and Server 2 runs databaseB. In a switchover situation, one server can run both database instances.

Creating the first resource hierarchy on Server 1:

Server:	Server1
Directory of *my.cnf* File Location:	*/etc*
Directory of MySQL Executables Location:	*/shr1/mysqlbin*
Database Tag:	mysql-shared.example.instance1

Extending the first resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend:	mysql-shared.example.instance1
Target Server:	Server2
Target Priority:	10
Directory of *my.cnf* File Location:	*/etc*
Directory of MySQL Executables Location:	*/shr1/mysqlbin*
Database Tag:	mysql-shared.example.instance1

Creating the second resource hierarchy on Server 2:

Server:	Server2
Directory of *my.cnf* File Location:	*/etc*
Directory of MySQL Executables Location:	*/shr2/mysqlbin*
Database Tag:	mysql-shared.example.instance2

Extending the second resource hierarchy to Server 1:

Template Server:	Server2
Tag to Extend:	mysql-shared.example.instance2
Target Server:	Server1
Target Priority:	10
Directory of *my.cnf* File Location:	*/etc*
Directory of MySQL Executables Location:	*/shr2/mysqlbin*
Database Tag:	mysql-shared.example.instance2

6.6.2.7. Multiple Database Server Environment

Following are some configuration considerations if you have multiple MySQL database servers and databases:

- If running active/active or muliple MySQL instances (of the same or different versions), please consider using the mysqld Group feature if possible. SIOS recommends using mysqld Groups (mysqld_multi) for multiple MySQL database server configurations.
- If running active/active or multiple instances of MySQL, do not mount a shared file system as */var/lib/mysql*. This causes unexpected shutdown of MySQL servers by the mysql startup command (`safe_mysqld` or `mysqld_safe`).
- The *my.cnf* file must be stored in the data directory for each of the active/active or multiple servers if not using the mysqld group feature. For configurations using mysqld Groups, the *my.cnf* file should be stored in */etc* and not in the data directory. For more information on LifeKeeper and the mysqld Group feature, see Using mysqld Groups with LifeKeeper.
- Additional port numbers for MySQL must be specified in the */etc/services* file.
- Each MySQL database server must be configured to run on a different port and access a different socket file. These configuration options are specified in the *my.cnf* file in the data directory.
- Each server must be configured to access data from a different shared location (i.e. each server must use a different data directory).

6.6.2.8. Using mysqld Groups with LifeKeeper

The MySQL Application Recovery Kit supports my.cnf files using the mysqld group feature managed via mysqld_multi. This MySQL feature allows multiple MySQL instances to be easily configured via a single my.cnf file (typically stored in */etc*.) The kit now detects a my.cnf file using the mysqld group format and prompts the administrator to select the number of the mysqld group to be protected. The choice list provided to the administrator is determined by the group numbers defined in the my.cnf file minus any group numbers already being protected by the kit.

In general, it is easier to set up and control multiple MySQL instances using the mysqld group feature, and SIOS recommends that this approach be used when setting up active/active or multiple instance configurations.

my.cnf File

When using the mysqld group feature, the following are imperative:

a. A single my.cnf file should be used for defining mysqld groups for the database instances.

b. The my.cnf file should NOT be placed on shared storage.

c. An exact copy of the my.cnf file needs to exist on each cluster node (*/etc/my.cnf* is ideal).

d. Any changes made to the my.cnf file must be propagated to every node in the LifeKeeper cluster.

The recovery kit uses mysqld_multi commands when it detects the my.cnf file is using mysqld groups. Based on this, you should be able to use mysqld_multi to test your MySQL instance before placing it under control of LifeKeeper.

The following is a relatively complex my.cnf file using mysqld groups that describes two database instances controlled by mysqld_multi. The `mysqld_multi` command (and the MySQL LifeKeeper recovery kit) gives the administrator a lot of options on how things get set up. In the example below, `[mysqld1]` defines a relatively simple MySQL instance that uses most of the default locations for various MySQL directives. The second example `[mysqld55]` moves things around more. The comments will help describe what each section is doing in terms of LifeKeeper's interaction with MySQL.

```
#The following client section defines which username/password combination wil
l be used for
#LifeKeeper  connections. The username/password combination needs to be define
d in each MySQL
# Database instance that will be described in this my.cnf file.
[client]
user     =  steeleye
password =  password
```

```
# This next section describes the default version of mysqld and mysqldadmin th
at mysqld_multi
# will use when processing mysqld_multi commands. The username/password combo
defines the
# MySQL account that mysqld_multi will use when working with the database inst
ances. This
# username and password combo needs to be defined in each MySQL Database insta
nce that will be
# controlled by mysqld_multi. See how to set up the multi_admin account in th
e MySQL Reference
# Manual, by issuing "mysqld_multi --example".
[mysqld_multi]
mysqld     = /usr/bin/mysqld_safe
mysqladmin = /usr/bin/mysqladmin
user       = multi_admin
<>password   = password
```

```
# The next section defines the first of two MySQL Database instances in this m
y.cnf file. Note
# that each section starts  with a [mysqldNN] where NN is the mysqld group num
ber (or instance).
# Each group name must have a number. There are a  number of directives that t
he LifeKeeper MySQL
# Recovery Kit will be looking for in these sections.
[mysqld1]
datadir  = /s11/mysql-data5077       #Defines where the data files for the ins
tance will live. For
                                     # LifeKeeper, this directory must be on L
ifeKeeper protected
                                     #   (shared or replicated) storage.
mysqld   = /usr/bin/mysqld_safe       # Defines specifically which mysqld comm
and will be used for
                                      #   starting the instance. This one is u
sing the
                                      #   default  mysqld_safe that came with
the distribution.
socket=/s11/mysql-data5077/moe.socket # Defines the location of the socket fo
r this instance.
                                      #   If the socket is not on LifeKeeper p
rotected storage, it
                                      #   needs to be defined in exactly the s
ame place on each
                                      #   node in the cluster and be owned by
the "user" defined
                                      #   below.<
port     =  3307                      # Each instance needs its own, unique TC
P/IP port.
pid-file = /var/run/mysqld/mysqld.pid # The pid-file can be on LifeKeeper prot
ected or
                                     #   non-LifeKeeper protected storage.
log-error= /var/log/mysqld.log        # Location of the MySQL error log for th
is instance. Can be
                                     #   on LifeKeeper protected or non-LifeKe
eper protected
                                     #   storage.
user     = mysql                      # The Linux user name that will run the
MySQL processes.
```

```
#The next section defines the more complicated of the two MySQL instances. Ins
tance "55" is not
#using the default MySQL that came with the Linux distribution as it is using
the 5.5.12 version
#of MySQL that was installed from source. The binaries for this version were i
nstalled onto shared
#storage, and the binary directory is LifeKeeper protected.
[mysqld55]
datadir = /s11/mysql-data5512           # Same as above; this instance uses a
different data
                                        #   directory, and  this directory is
on LifeKeeper
                                        #   protected storage.
mysqld  =/s11/mysql5512/bin/mysqld_safe # For this instance, a different versi
on of mysqld_safe
                                        #   is used; the  one that is include
d with 5.5.12.
socket=/s11/mysql-misc5512/larry.socket # This instance has the socket on Life
Keeper protected
                                        #   storage, but not  in the default l
ocation (datadir).
port     = 3308                         # This instance has a unique TCP/IP po
rt as well.
pid-file = /var/run/mysqld/mysqld55.pid # This instance's pid-file is not on L
ifeKeeper protected
                                        #   storage.
log-error = /var/log/mysqld55.log       # This instance's log-error (error lo
g) is not on
                                        #   LifeKeeper protected  storage.
log-bin   = /s11/mysql-log5512/larry    # The log-bin directive specifies wher
e the binary
                                        #   transaction logs are  located for
this instance.
                                        #   These logs must be on LifeKeeper
protected storage
                                        #   (the recovery kit will enforce thi
s). By default,
                                        #   these logs are in the datadir.
user      = mysql                       # The Linux user name that will run th
e MySQL processes.
```

When describing both sets up of [mysqld<N>] for multi instance and [mysqld] for single instance, the set up for single instance must be described at the last part.

Example:

```
[mysqld1]
(set up for mysqld1)

[mysqld2]
(set up for mysqld2)

[mysqld55]
(set up for mysqld55)
```

```
[mysqld]
(set up for mysqld for single instance )
```

mysqld_multi Commands

For this example, issuing the mysql command:

```
# mysqld_multi start 1
```

would start the mysqld group 1 instance defined in my.cnf as [mysqld1], assuming all of the LifeKeeper protected resources that it depends on were in service on one of the LifeKeeper nodes.

Issuing the mysql command:

```
# mysqld_multi report 1
```

would report on the status of this instance (e.g. running or not running). Once this instance is running, creating a resource for it in LifeKeeper should be easy.

To get more information on setting up a mysqld_multi style my.cnf file, issue the command:

```
# mysqld_multi —example
```

6.6.2.9. Using Network Attached Storage

There are a couple of special considerations to take into account when configuring LifeKeeper to use an NFS file server (Network Attached Storage) as cluster storage.

Use the NAS Recovery Kit

The optional Network Attached Storage (NAS) recovery kit is required when using an NFS server as a shared storage array with LifeKeeper for Linux. Install the NAS recovery kit (and a license) on each cluster node. See the NAS Recovery Kit documentation for more details.

Possible Error Message

When using Network Attached Storage (NAS) with MySQL, you may experience MySQL instances not restarting following a failover due to a system crash. The MySQL error log should indicate the cause of the error.

MySQL 5.0

```
110523 22:10:58 mysqld started
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
110523 22:10:58 InnoDB: Retrying to lock the first data file
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
```

MySQL 5.5

```
110524 10:52:20 InnoDB: The InnoDB memory heap is disabled
110524 10:52:20 InnoDB: Mutexes and rw_locks use GCC atomic builtins
110524 10:52:20 InnoDB: Compressed tables use zlib 1.2.3
110524 10:52:20 InnoDB: Initializing buffer pool, size = 128.0M
110524 10:52:20 InnoDB: Completed initialization of buffer pool
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
110524 10:52:20 InnoDB: Retrying to lock the first data file
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
```

This indicates that the MySQL mysqld process has set an NFS lock on the file “ibdata1” on the NFS file system that is being controlled by LifeKeeper. The lock was not cleared by the system crash, so LifeKeeper is unable to bring the MySQL instance back into service. MySQL thinks that some other process is using the *ibdata1* file.

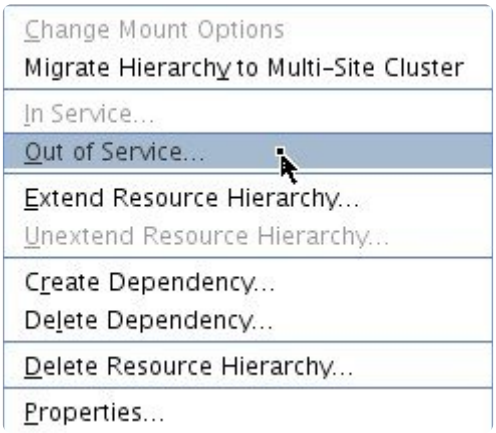
Solution

To fix this, mount the NFS file system that will hold ibdata1 with the “nolock” NFS option before the File System resource is created. By default, NFS allows file locks to be set. If the “nolock” option is used before resource creation, LifeKeeper will pick up this option and use it each time it brings the file system resource in service. Since LifeKeeper will be controlling access (from the cluster nodes) to the file system containing ibdata1, the lock is not typically critical. The NFS mount options used during testing were `“rw,sync,tcp,nfsvers=3,nolock”.`

It is not necessary to use the “nolock” on other file systems used by the MySQL resource hierarchy such as the file system where the MySQL binaries are located.

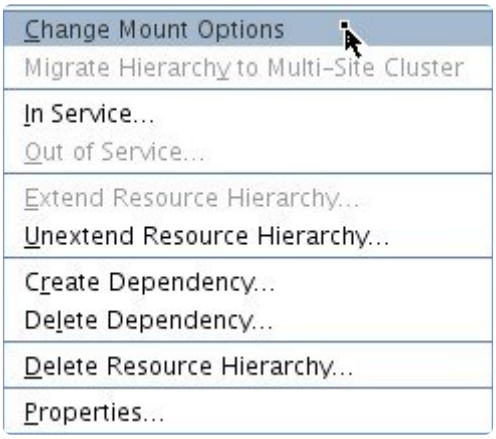
If the NAS File System resource has already been created without the “nolock” option set, use the following procedure to change the mount option:

1. Using the LifeKeeper GUI, take the file system resource that needs to be changed out of service. This can be done from the LifeKeeper GUI putting the pointer on the file system resource and doing a right mouse click, and select **Out of Service** from the drop-down menu. This action may take parent resources out of service as well.

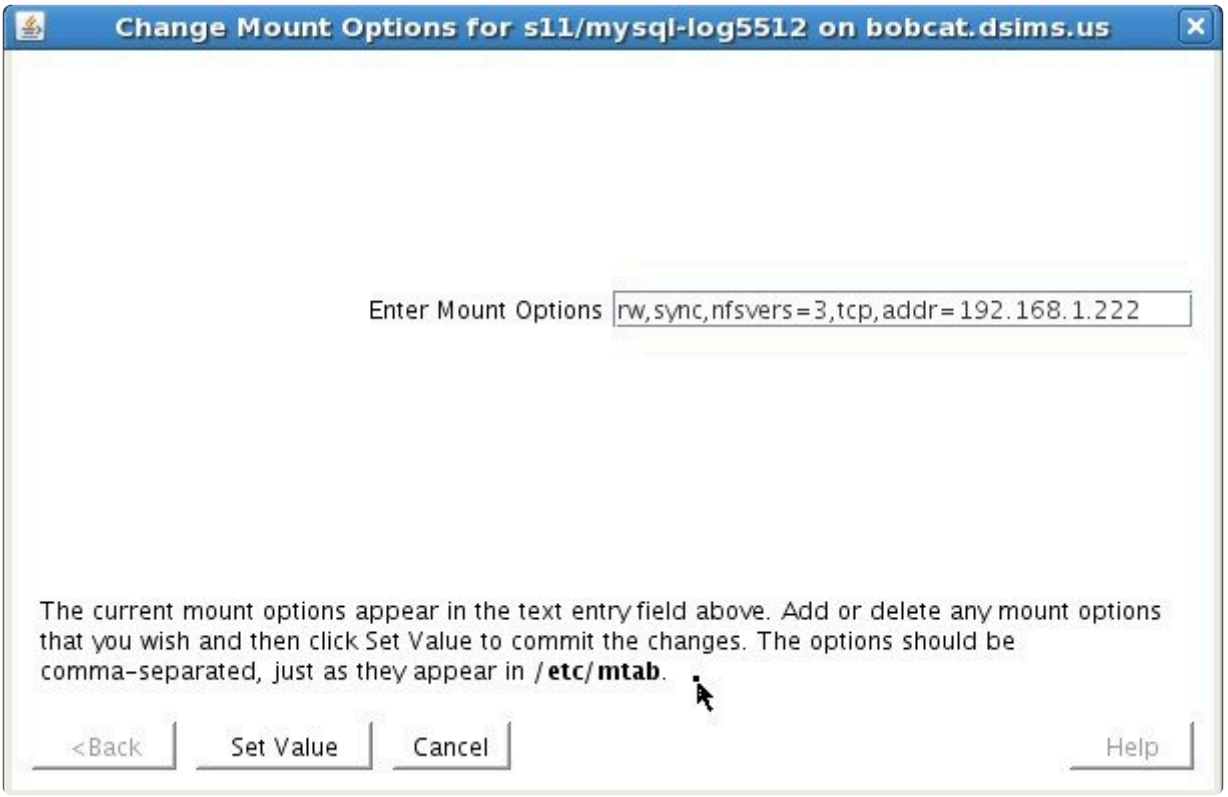


2. Confirm the **Out of Service** action and allow the process to complete.

3. Once the file system resource is out of service, you can put the pointer on the resource and do another right mouse click, and from the drop-down menu select **Change Mount Options**.

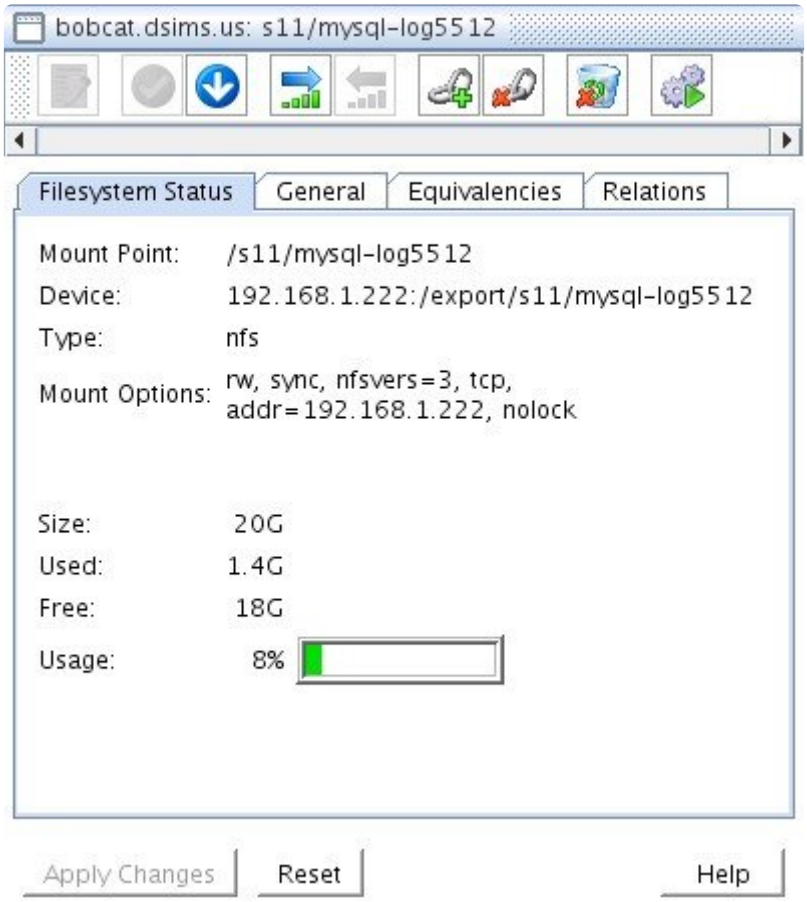


4. In the popup window, add **nolock** to the line of options, and click **Set Value**. You will need to repeat steps 3 and 4 for each node in the cluster.



5. Bring the NAS File System resource back in service by doing a right mouse click, and selecting **In Service**.

6. The File System resource's property panel should now reflect that "nolock" is one of the current mount options.


bobcat.dsims.us: s11/mysql-log5512
Filesystem Status
General
Equivalencies
Relations
Mount Point: /s11/mysql-log5512
Device: 192.168.1.222:/export/s11/mysql-log5512
Type: nfs
Mount Options: rw, sync, nfsvers=3, tcp, addr=192.168.1.222, nolock
Size: 20G
Used: 1.4G
Free: 18G
Usage: 8%
Apply Changes
Reset
Help

6.6.2.10. Considerations on MySQL use in systemd Environments

If MySQL (version 5.7.6 or later) is installed on a OS distribution adopting systemd, the mysqld_safe and mysqld_multi commands are not installed and thus unavailable for LifeKeeper use. In these environments, LifeKeeper will use the systemctl command to start and stop the MySQL service.

Set up MySQL referring to the article Managing MySQL Server with systemd. Specially set up the PID File as this is required for Systemd and LifeKeeper. Systemd MySQL set up and my.cnf set up must be the same on all nodes.

The following set up items must be defined for resource creation.

Set up item	Value to set
Location of my.cnf	The full path of the directory that contains the my.cnf file used when starting the MySQL service with the systemctl command
Location of MySQL executables	The full path of the directory where the mysqladmin command is installed

Notes: The setting of “Location of my.cnf” is used inside LifeKeeper to read the settings in my.cnf. The value set up here is not used by the systemctl startup command. In the case where the my.cnf file is in a different path from the default, set up MySQL for Systemd correctly and make sure the MySQL service starts and stops as expected. Also, the “include” directive is not supported. All the configuration information must be described in a single my.cnf file.

6.6.3. Installing/Configuring MySQL with LifeKeeper

6.6.3.1. LifeKeeper Configuration Tasks for MySQL

You can perform the following configuration tasks from the LifeKeeper GUI. The following four tasks are described in this section, as they are unique to a MySQL resource instance, and different for each Recovery Kit.

- Create a Resource Hierarchy. Creates an application resource hierarchy in your LifeKeeper cluster.
- Delete a Resource Hierarchy. Deletes a resource hierarchy from all servers in your LifeKeeper cluster.
- Extend a Resource Hierarchy. Extends a resource hierarchy from the primary server to a backup server.
- Unextend a Resource Hierarchy. Unextends (removes) a resource hierarchy from a single server in the LifeKeeper cluster.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View / Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, we explain how to configure your Recovery Kit by selecting certain tasks from the **Edit** menu of the LifeKeeper GUI. You can also select each configuration task from the toolbar. You can also right-click a global resource in the **Resource Hierarchy Tree** (left-hand pane) of the status display window to display the same drop-down menu choices as the **Edit** menu.

You can also right-click a resource instance in the **Resource Hierarchy Table** (right-hand pane) of the status display window to perform all the configuration tasks, except **Creating a Resource Hierarchy**, depending on the state of the server and the particular resource.

6.6.3.2. Creating a MySQL Resource Hierarchy

IMPORTANT:

In a LifeKeeper cluster environment where the MySQL data directory (datadir) files are on a shared disk, you must make sure that the shared file system is mounted on the primary/template server. If the file system resource is created first, the shared file system MUST be mounted on the same mount point on each server. It is also important to remember that a working communication path (i.e. heartbeat) is required before you can create your resource. The MySQL data directory can exist on shared, replicated or network attached storage.

To create a resource instance from the primary server, you should complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy**.

   If you wish to change a selection you have already entered or encounter an error message during any step in the creation of your MySQL resource hierarchy, you will generally be able to back up and change your selection or make corrections (assuming the **Back** button is enabled).

Important: The MySQL database server daemon (mysqld) for the MySQL instance you want to protect must be running when you create the resource.

   A dialog box will appear with a drop-down menu listing all recognized Recovery Kits installed within the cluster. Select **MySQL Database** from the drop-down menu.

   Click **Next**.

   If you click the **Cancel** button at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

2. Select the **Switchback Type**. This dictates how the MySQL instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either ***intelligent*** or ***automatic***. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths.

The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box.

Click **Next**.

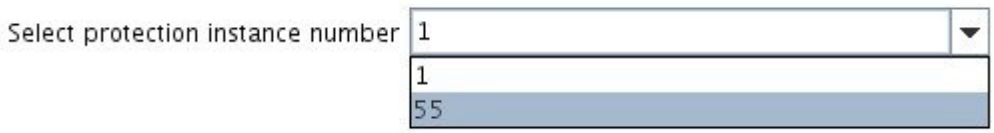
3. Select the **Server** where you want to place the MySQL database instance (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop-down menu.

Click **Next** to proceed to the next dialog box.

4. Select or enter the **Location of my.cnf**. This is the full path name (excluding the file name) where the MySQL configuration file (*my.cnf*) is located.

Click **Next** to proceed to the next dialog box.

5. Select the **Protection Instance Number** if you have a mysqld_multi style my.cnf file. If you are using a more traditional style my.cnf file, you will not see this screen.



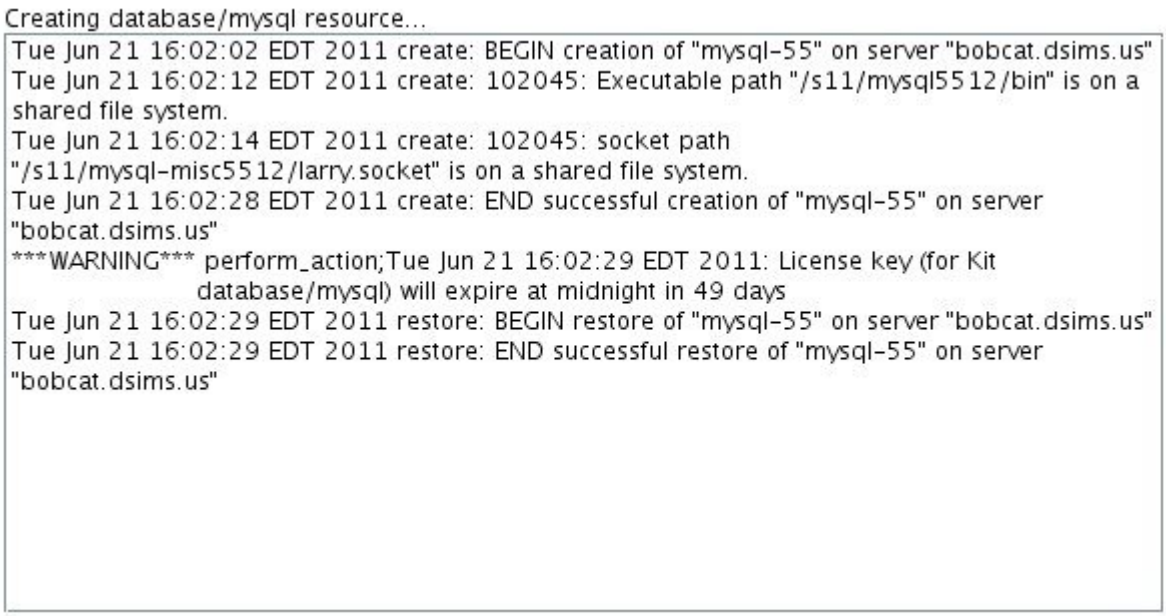
6. Select or enter the* Location of MySQL executables location*. This is the full path name of the binaries used to start and monitor the MySQL database server daemon.

Note: At this point, LifeKeeper will validate that you have provided valid data to create your MySQL resource hierarchy. If LifeKeeper detects a problem with either of this validation, an ERROR will appear on the screen. If the directory paths are valid, but there are errors with the MySQL configuration itself, you may pause to correct these errors and continue with the hierarchy creation.

Click **Next** to proceed to the next dialog box.

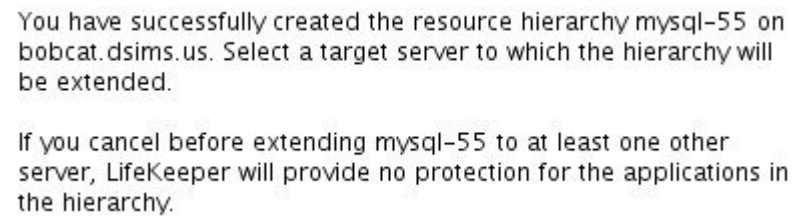
7. Select or enter the **Database Tag**. This is a tag name given to the MySQL hierarchy. You can select the default or enter your own tag name.

When you click **Create**, the **Create Resource Wizard** will create your MySQL resource.



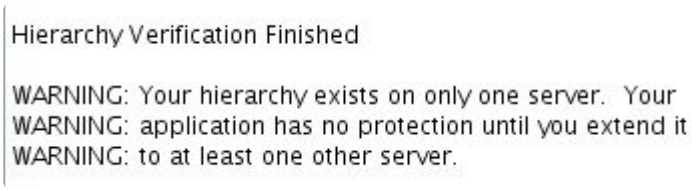
Note: The MySQL resource hierarchy should be created successfully at this point.

8. Another information box will appear explaining that you have successfully created an MySQL resource hierarchy, and you must **Extend** that hierarchy to another server in your cluster in order to place it under LifeKeeper protection.



When you click **Continue**, LifeKeeper will launch the **Pre-Extend Wizard** that is explained in the next section.

If you click **Cancel** now, a dialog box will appear warning you that you will need to come back and extend your MySQL resource hierarchy to another server at some other time to put it under LifeKeeper protection.



9. Click **Done** to exit.

6.6.3.3. Deleting a MySQL Resource Hierarchy

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, and then **Resource**. From the drop-down menu, select **Delete Resource Hierarchy**.

2. Select the name of the **Target Server** where you will be deleting your MySQL resource hierarchy.

> **Note:** If you selected the **Delete Resource** task by right-clicking from the right pane on an individual resource instance, or from the left pane on a global resource where the resource is on only one server, this dialog box will not appear.

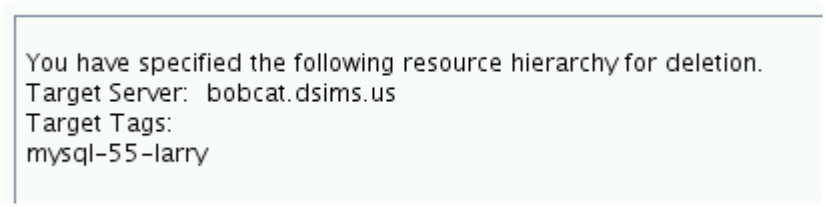
Click **Next**.

3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete, and highlight it.

> **Note:** If you selected the **Delete Resource** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

Click **Next**.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete.



   Click **Delete**.

5. Another information box appears confirming that the MySQL resource was deleted successfully.

6. Click **Done** to exit.

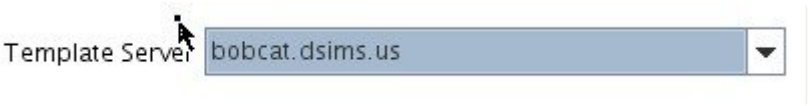
6.6.3.4. Extending Your MySQL Hierarchy

After you have created a hierarchy, you will want to extend that hierarchy to another server in the cluster. There are three possible scenarios to extend your resource instance from the template server to a target server. The first scenario is when you **Continue** from creating the resource into extending that resource to another server. The second scenario is when you enter the **Extend Resource Hierarchy** task from the edit menu as shown below. The third scenario is when you right-click on an unextended hierarchy in either the left or right pane. Each scenario takes you through the same dialog boxes (with a few exceptions, which are clearly detailed below).

1.
2. If you are entering the **Extend** wizard from the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop-down menu, select **Extend Resource Hierarchy**. This will launch the **Extend Resource Hierarchy** wizard.

3. The first dialog box to appear will ask you select the **Template Server** where your MySQL resource hierarchy is currently in service. It is important to remember that the **Template Server** you select now and the **Tag to Extend** that you select in the next dialog box represent an in service resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you selected. The drop-down box in this dialog provides the names of all the servers in your cluster.

   Note: If you are entering the **Extend Resource Hierarchy** task immediately following the creation of a MySQL resource hierarchy, this dialog box will not appear, since the wizard has already identified the template server in the create stage. This is also the case when you right-click either the MySQL resource icon in the left pane or right-click on the MySQL resource box in the right pane the of the GUI window and choose **Extend Resource Hierarchy**.

   It should be noted that if you click **Cancel** at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extension process to that particular server. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

   For example, let us say you have created your resource on Server 1 and extended that resource to Server 2. In the middle of extending the same resource to Server 3, you change your mind and click **Cancel** inside one of the dialog boxes. This will cancel only your action to extend the resource to Server 3, not the extension you created to Server 2. If you want to remove Server 2 from this hierarchy, you must unextend the resource from Server 2.

   Click **Next** to proceed to the next dialog box.

4. Select the **Tag to Extend**. This is the name of the MySQL instance you wish to extend from the

template server to the target server. The wizard will list in the drop-down menu all the resources that you have created on the template server, which you selected in the previous dialog box.

Note: Once again, if you are entering the Extend Resource Hierarchy task immediately following the creation of a MySQL resource hierarchy, this dialog box will not appear, since the wizard has already identified the tag name of your MySQL resource in the create stage. This is also the case when you right-click either the MySQL resource icon in the left hand pane or on the MySQL resource box in the right hand pane of the GUI window and choose **Extend Resource Hierarchy**.

Click **Next**.

5. Select the **Target Server** where you are extending your MySQL resource hierarchy. The drop-down box provides the names of the servers in your cluster that are not already in the selected hierarchy.

Click **Next**.

6. Select the **Switchback Type**. This dictates how the MySQL instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either ***intelligent*** or ***automatic***. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back online and reestablishes LifeKeeper communication paths.

The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box.

Click **Next**.

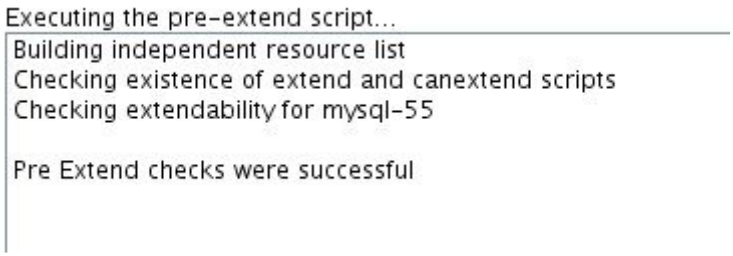
7. Select or enter a **Template Priority**. This is the priority for the MySQL hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note**: This selection will appear only for the initial extend of the hierarchy.

Click **Next**.

8. Select or enter the **Target Priority**. This is the priority for the new extended MySQL hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

Click **Next**.

9. An information box will appear explaining that LifeKeeper has successfully checked your environment and that all the requirements for extending this MySQL resource have been met. If there were some requirements that had not been met, LifeKeeper would not allow you to select the **Next** button, and the **Back** button would be enabled.

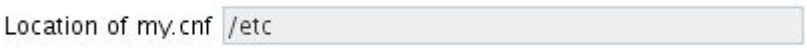


If you click **Back**, you can make changes to your resource extension according to any error messages that may appear in the information box.

If you click **Cancel** now, you will need to come back and extend your MySQL resource hierarchy to another server at some other time to put it under LifeKeeper protection.

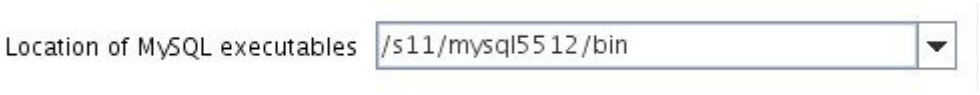
When you click **Next**, LifeKeeper will launch you into the **Extend Resource Hierarchy** configuration task.

10. This dialog box is for information purposes only. You cannot change the **Location of my.cnf** that appears in the box. The MySQL instance acquired the location information from its configuration file.



Click **Next**.

11. Select or enter the **Location of MySQL executables**.This is the full path name of the binaries used to start and monitor the MySQL database server daemon.

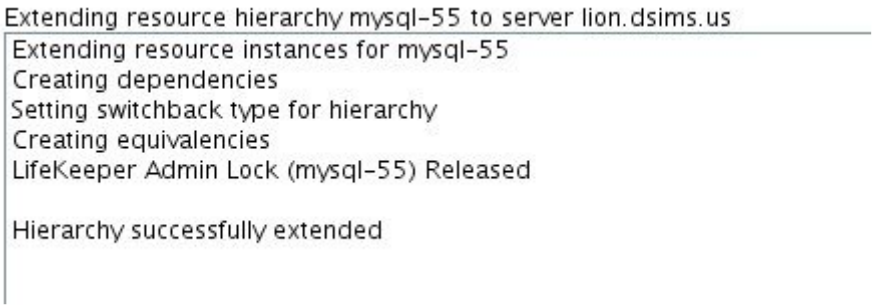


Click **Next**.

12. Select or enter the **Database Tag**. This is a tag name given to the MySQL hierarchy. You can select the default or enter your own tag name.

Click **Extend**.

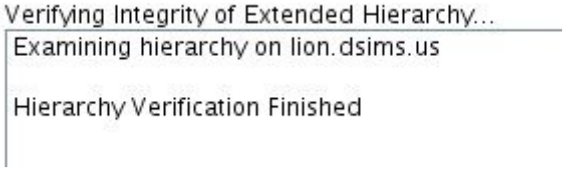
13. An information box will appear verifying that the extension is being performed.



Click **Next Server** if you want to extend the same MySQL resource instance to another server in your cluster. This will repeat the **Extend Resource Hierarchy** operation.

If you click **Finish**, LifeKeeper will verify that the extension of the MySQL resource was completed successfully.

14. If you clicked **Finish**, the following screen appears.



15. Click **Done** in the last dialog box to exit.

Note: Be sure to test the functionality of the new instance on both servers.

6.6.3.5. Unextending Your MySQL Hierarchy

1. From the LifeKeeper GUI menu, select **Edit**, and **Resource**. From the drop-down menu, select **Unextend Resource Hierarchy**.

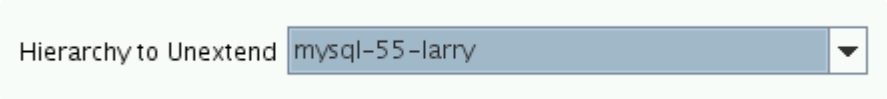
2. Select the **Target Server** where you want to unextend the MySQL resource. It cannot be the server where the MySQL resource is currently in service.

> **Note**: If you selected the **Unextend** task by right-clicking from the right pane on an individual resource instance, this dialog box will not appear.

Click **Next**.

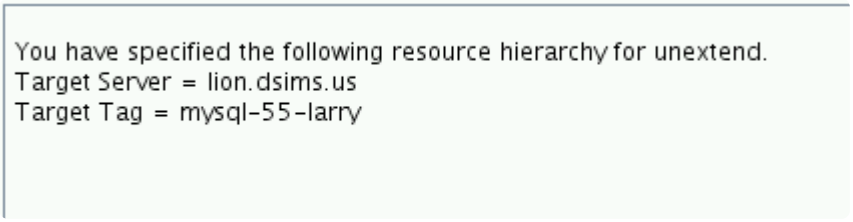
3. Select the MySQL **Hierarchy to Unextend.**

> **Note**: If you selected the **Unextend** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.



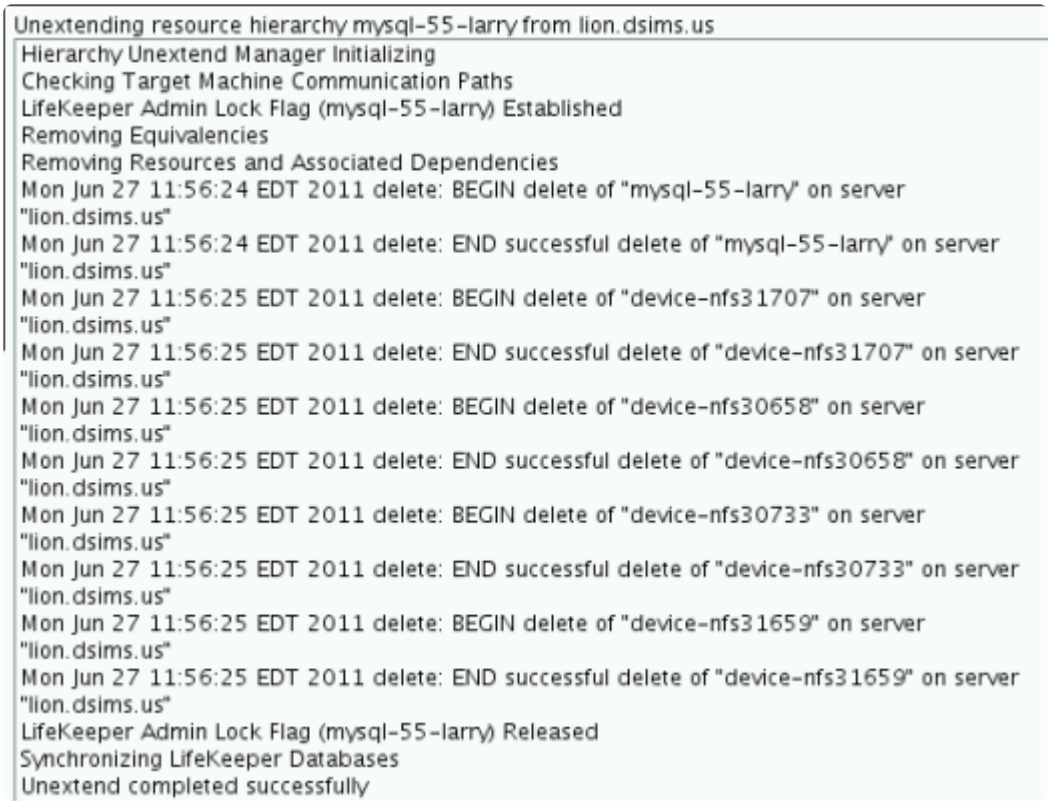
Click **Next**.

4. An information box appears confirming the target server and the MySQL resource hierarchy you have chosen to unextend.



Click **Unextend**.

5. Another information box appears confirming that the MySQL resource was unextended successfully.



6. Click **Done** to exit.

6.6.4. MySQL Administration

Testing Your Resource Hierarchy

You can test your MySQL resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Performing a Manual Switchover from the GUI

6.6.4.1. Performing a Manual Switchover from the GUI

You can test your MySQL resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Performing a Manual Switchover from the GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, **Resource** and **In-Service** from the drop-down menu. For example, an **In-Service** request executed on a backup server causes the application hierarchy to be placed in service on the backup server and taken out of service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out-of-Service** request, the application is taken out of service without bringing it in service on the other server.

LifeKeeper does not regulate or control internal operations such as rollbacks and backing up archives. Tape archiving and restoration are the responsibility of the application administrator.

Recovery Operations

When the primary server fails, the MySQL Recovery Kit software performs the following tasks:

- Mounts the file system(s) – shared or replicated – on the backup server
- Starts the daemon processes related to MySQL

6.6.5. MySQL Troubleshooting

Common Error Messages

This section provides a list of messages that you may encounter while creating and extending an SPS MySQL resource hierarchy or removing and restoring a resource. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition. Messages from other SPS components are also possible.

In these cases, please refer to the Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

MySQL Specific Error Messages

Note: In the Error Message column, a word in quotations and all capital letters refers to the name of a resource on the server (for example, “SERVER” might actually be a server named “Server1”).

Error Number	Error Message
102001	Usage: “SCRIPT NAME” sysname dbvarname cnfpath exepath instance
102002	Usage: “SCRIPT NAME” cnfpath
102003	Usage: “SCRIPT NAME” exepath cnfpath
102004	Unable to obtain a valid value for the “socket” variable in “PATH”/my.cnf **Action**: There must be an entry for the “socket” in the ‘mysqld’ section of the my.cnf configuration file
102005	Unable to obtain a valid value for the “port” in “PATH”/my.cnf **Action**: There must be an entry for the “port” in the ‘mysqld’ section of the my.cnf configuration file
102006	Unable to obtain the data directory location “PATH” **Action**: Please make sure that the database is running using the socket and port specified.
102007	Must specify the absolute path to the my.cnf configuration file
102008	Must specify the absolute path to the MySQL executables
102009	The file my.cnf does not exist in the path specified
102010	The MySQL executables do not exist in the path specified
102011	LifeKeeper was unable to start the MySQL database server
102012	LifeKeeper successfully started the MySQL database server
102013	LifeKeeper was unable to stop the MySQL database server

102014	LifeKeeper successfully stopped the MySQL database server
102015	The port “PORT NUMBER” is in use on the target server “SERVER”
102016	The MySQL database server is not running on server “SERVER”
102017	Unable to open the configuration file “PATH”/my.cnf
102018	Unable to get the Data Directory information for resource “TAG” on server “SERVER”
102019	Unable to get the configuration file location information for resource “TAG” on server “SERVER”
102020	Unable to get the executable location information for resource “TAG” on server “SERVER”
102021	The argument for the configuration file path is empty
102022	The argument for the executable path is empty
102023	The path “PATH” for directive “DIRECTIVE” is not on a shared filesystem
102024	Unable to get the information for resource “TAG” on system “SYSTEM”
102025	The MySQL data directory “DATADIR” is already under LifeKeeper protection
102026	The port variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match
102027	The socket variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match
102028	The MySQL database server is not running on server “SERVER” **Action**: There must be a valid entry for the “user” variable in the ‘client’ section of the my.cnf configuration file
102029	Unable to obtain a valid value for the “password“ variable in “PATH”/my.cnf **Action**: There must be a valid entry for the “password” variable in the ‘client’ section of the my.cnf configuration file
102030	The user variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match
102031	The password variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match
102032	Unable to obtain the pid file location **Action**: There must be an entry for the “pid-file” variable in the ‘mysqld’ section of the my.cnf configuration file
102033	Unable to obtain a valid value for the “user” variable in “PATH”/my.cnf **Action**: The OS user must be specified using the “user” variable in the ‘mysqld’ section of the my.cnf configuration file
102034	WARNING: A my.cnf file exists at “PATH”, which may override the values specified in the file at “PATH”/my.cnf.
102035	The mysql system user “USER” does not exist on target server “SERVER”
102036	The mysql system user “USER” uids are different on target server

	“SERVER1” and template server “SERVER2”
102037	The mysql system user “USER” gids are different on target server “SERVER1” and template server “SERVER2”
102038	LifeKeeper was unable to stop the MySQL database server using a graceful shutdown. Issuing kill for pid(s): “PROCESS ID LIST”.
102039	LifeKeeper will ignore failed connection as possible max connections error, due to existence of process pid “PROCESS ID”.
102040	The mysql action for resource tag :TAG” returned: “COMMAND OUTPUT”.
102041	LifeKeeper was unable to start the MySQL database server using the defaults-file option. Retrying with individual options.
102042	The LifeKeeper “ACTION” action detected the flag “FLAG”, and will exit.
102043	END of “ACTION” action on due to a(n) “SIGNAL” signal.
102044	The file my.cnf does not exist in the stored path “PATH”.
102045	“DIRECTIVE” path “PATH” is on a shared filesystem.
102046	Starting mysqld daemon with databases from “PATH”.

6.7. MD Recovery Kit Administration Guide

The SIOS Protection Suite (SPS) for Linux Software RAID (md) Recovery Kit provides software RAID support for other LifeKeeper recovery kits. Thus, LifeKeeper-protected applications can take advantage of the benefits offered by software RAID, including lower cost data redundancy, data replication over a SAN and simplified storage management.

The Software RAID Recovery Kit is different from most other LifeKeeper recovery kits in that it is never used alone, but always as a dependency of another LifeKeeper resource. As such, many of the operations typically associated with a LifeKeeper recovery kit – for example, creating a hierarchy – are not directly applicable to the Software RAID Recovery Kit.

Document Contents

This guide explains the following topics:

- Documentation and References. Provides a list of related LifeKeeper for Linux documents and where to find them, along with references to a number of helpful documents about the Linux Software RAID product.
- Requirements. Describes the hardware and software necessary to properly set up, install and operate the Software RAID Recovery Kit. Refer to the SPS for Linux Installation Guide for specific instructions on how to install or remove LifeKeeper for Linux software.
- Overview. Provides a general description of the Software RAID Recovery Kit and corresponding resource types.
- LifeKeeper Software RAID Hierarchy Creation and Administration. Includes a detailed description of Software RAID Recovery Kit administration tasks through LifeKeeper.
- Troubleshooting. Provides a list of informational and error messages with recommended solutions.

6.7.1. Software RAID (md) Documentation and References

The following SPS product documentation is available from the SIOS Technology Corp. website:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is provided on the SIOS Technology Corp. website at:

http://docs.us.sios.com/

For information on Linux Software RAID, refer to the manual pages for md(4) and mdadm(8) as well as the HowTo; Jakob Østergaard and Emilio Bueso, Maintainers, available at

www.unthought.net/Software-RAID.HOWTO.

6.7.2. Software RAID (md) Recovery Kit Hardware and Software Requirements

Your LifeKeeper configuration must meet the following requirements prior to the installation of the LifeKeeper for Linux Software RAID (md) Recovery Kit. Please see the SPS for Linux Installation Guide for specific instructions regarding the configuration of your LifeKeeper hardware and software.

Hardware Requirements

Software Requirements

6.7.2.1. Software RAID (md) Hardware Requirements

- **Servers.** This recovery kit requires two or more computers configured in accordance with the requirements described in the SPS for Linux Release Notes and the SPS for Linux Installation Guide, which are located on the SIOS Technical Documentation site at http://docs.us.sios.com/.

- **Data Storage.** The Software RAID Recovery Kit can be used in conjunction with shared storage. It cannot be used with network-attached storage (NAS). Otherwise, the kit has no specific requirements on storage configurations beyond the requirements of the recovery kit protecting the application sitting on top of the RAID device(s).

6.7.2.2. Software RAID (md) Software Requirements

- **Operating System.** The Linux Software RAID product is included in all major Linux distributions. See the SPS for Linux Release Notes for a list of supported distributions and versions.
- **mdadm(8) utility.** The recovery kit installation requires that the mdadm rpm package be installed. The specific versions of mdadm supported are those delivered by the Linux distributions.
- **LifeKeeper Software.** The same version of LifeKeeper core software and any recovery kits must be installed, including the Software RAID Recovery Kit, and any patches on each server. Please refer to the SPS for Linux Release Notes for specific LifeKeeper requirements.
- **LifeKeeper for Linux Software RAID (md) Recovery Kit.** The Software RAID Recovery Kit is provided on the SPS Installation Image File (sps.img). It is packaged, installed and removed via the Red Hat Package Manager, rpm. The following rpm file is supplied on the SPS Installation Image File (sps.img): steeleye-lkMD.

During package installation, checks are made to ensure that supported versions of both the LifeKeeper Core package and the mdadm package are present on the system where the Software RAID Recovery Kit is being installed. The SPS for Linux Release Notes contains information on the required versions of these packages.

Refer to the SPS for Linux Installation Guide for instructions on how to install or remove the LifeKeeper Core software and the Software RAID Recovery Kit.

The Software RAID Recovery Kit must be installed on each server in the cluster on which software RAID using md is being used to manage disk resources that are to be protected by LifeKeeper.

The Software RAID Recovery Kit must be installed prior to the hierarchy creation and extension of applications that sit on top of a RAID device.

6.7.3. Software RAID (md) Recovery Kit Overview

Software RAID (md) Operation

The Multiple Device driver (md) is currently the standard Linux software RAID product included with all of the major Linux distributions. Linux software RAID allows multiple physical disks and/or disk partitions to be grouped together to form virtual devices. Virtual devices are accessed as regular block devices, and as such may be used by file systems or any application that can operate directly with a block device.

Software RAID is principally used to provide data redundancy where hardware RAID (or storage replication) is not practical or feasible. The following diagram shows the relationship of the software RAID entities. File systems or applications use virtual devices. Virtual devices consist of the aggregation of one or more physical disk partitions or disks.

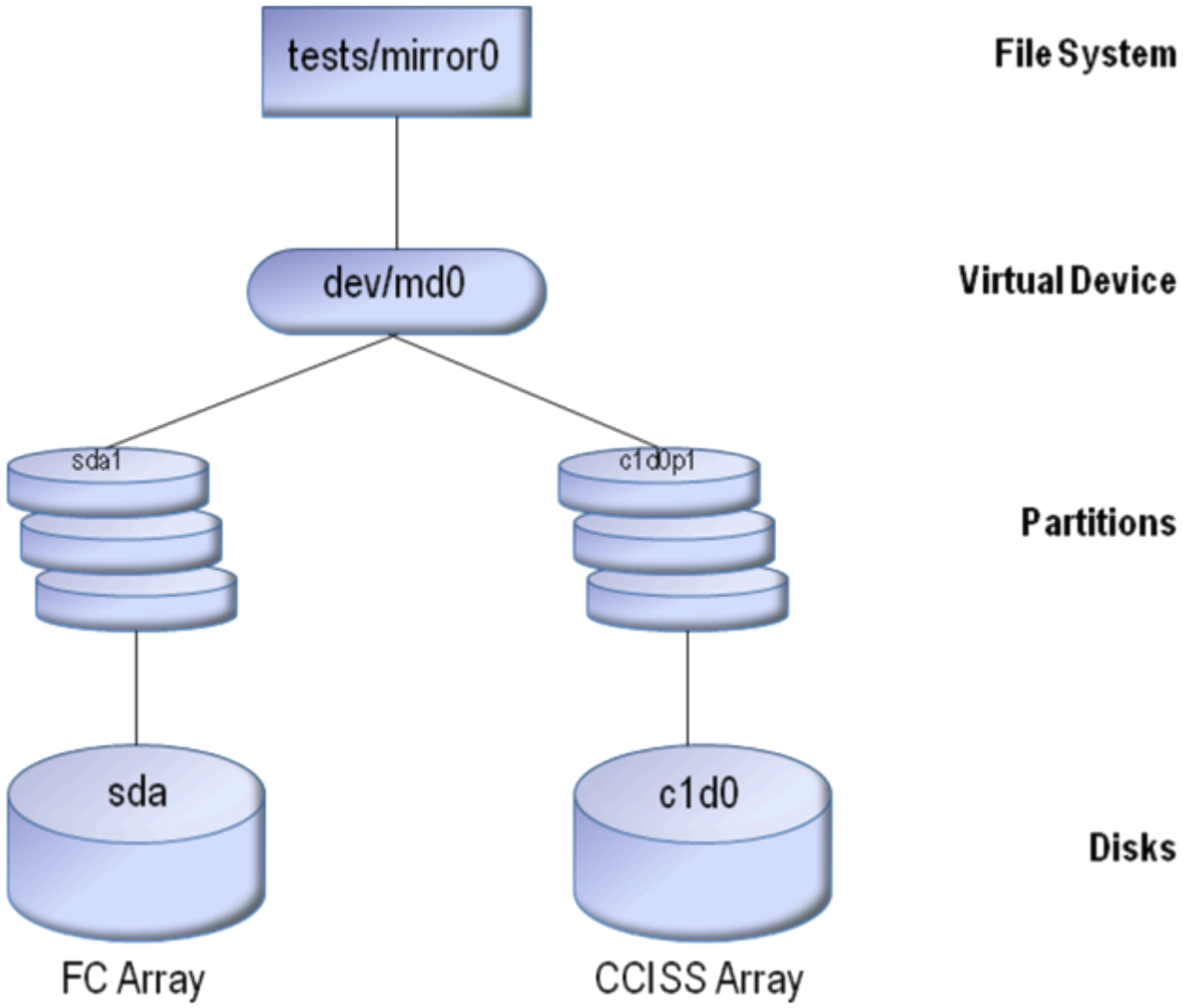


Figure 1: Software RAID Entity Relationships

In Figure 2 below, writes are written to both arrays in this single-path mirror. This is MDs prime function, replacing expensive storage replication.

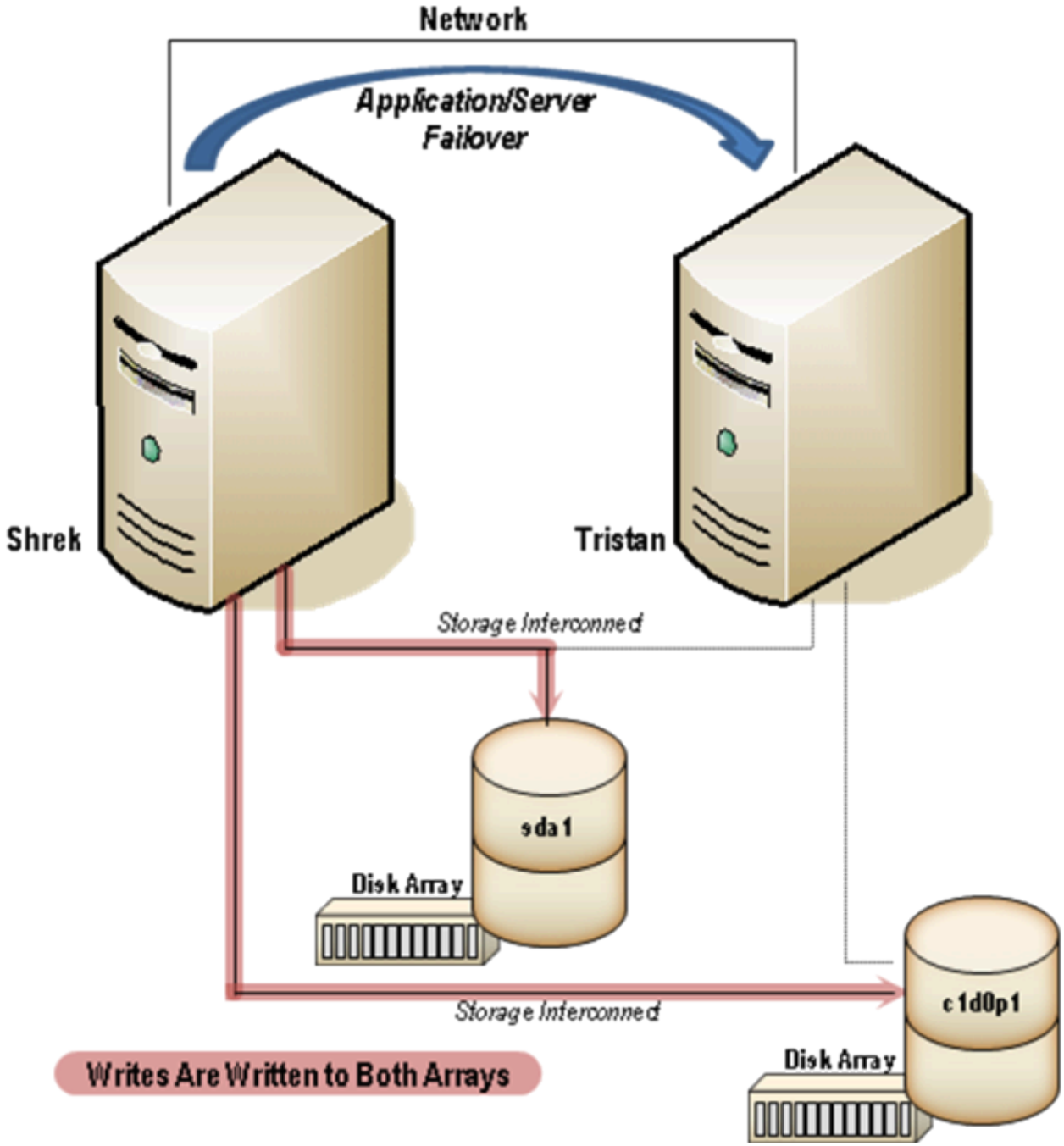


Figure 2: Single-Path MD Configuration

LifeKeeper for Linux Software RAID (md) Recovery Kit

The LifeKeeper Software RAID (md) Recovery Kit provides the support needed to allow other LifeKeeper recovery kits to operate properly with Linux software RAID virtual devices. To accomplish this support, the Software RAID Recovery Kit installs two new resource types: md and mdComponent that correspond to virtual devices and each partition or disk configured in the virtual device. The md and mdComponent resources exist solely for internal use so that other LifeKeeper resources can operate.

The mdComponent resource allows the Software RAID Recovery Kit to present the state of each individual component in the virtual device.

ISP – the component is configured properly in the virtual device and operating normally.

ISU – the component is a spare device. Note that when a device is hot added to a virtual device it will respond as a spare while the device is being restored.

OSU – the component is not configured in the virtual device. This may be a result of the component being removed from the virtual device. If a virtual device has a failed component and is unconfigured (stopped) and reconfigured (assembled), the failed component will no longer appear as a configured device, i.e., it will not show up as failed but as unconfigured.

OSF – the component has failed. **Note**: To receive an email notification when in this state, enable this option using `lk_confignotifyalias(8)`.

As shown in Figure 1, the virtual device md0 is composed of 2 disk partitions, sda1 and c1d0p1. This could reflect a RAID-1 mirror. A typical LifeKeeper hierarchy containing a virtual device looks much like the relationships shown in Figure 1. Refer to Figure 4 in the LifeKeeper Software RAID Hierarchy Creation and Administration section for an example of an actual LifeKeeper hierarchy.

The Software RAID Recovery Kit uses the mdadm(8) command provided by the mdadm package to manage the virtual device resources in a LifeKeeper hierarchy. The virtual device is configured (or assembled) when a hierarchy is being brought in-service during a failover or switchover operation, and is unconfigured (or stopped) when a hierarchy is being taken out of service.

6.7.3.1. Software RAID Notes and Restrictions

The following notes and restrictions apply to this version of the Software RAID Recovery Kit.

Activating Virtual Devices During Boot Up

Virtual devices on shared storage should not be activated during system boot-up.

Persistent Superblock

All virtual devices must be configured with a persistent superblock. The superblock is 4K long and is written in a 64K aligned block that starts at least 64K and less than 128K from the end of the device. This space must be accounted for when planning the size of your virtual device as this space is not usable by an application. Note: MD can now be configured with a bitmap using the “internal” feature. This creates a bitmap in this already required superblock, therefore, no additional space is required or additional LUN or additional file system. The bitmap will not show up in the hierarchy, but will just be “automatically” used. See the manual page for mdadm(8) and md(4) referenced in the Documentation and References section for further details.

HOMEHOST

The HOMEHOST feature in newer versions of mdadm is not supported by LifeKeeper. If a mirror is configured with HOMEHOST set, LifeKeeper will fail during resource creation.

As shown in Figure 3, the following messages will be displayed:

```
    “The MD device “/dev/md5” is configured with the unsupported “homehost” setting.”

    “Recreate the MD device without homehost set.”
```

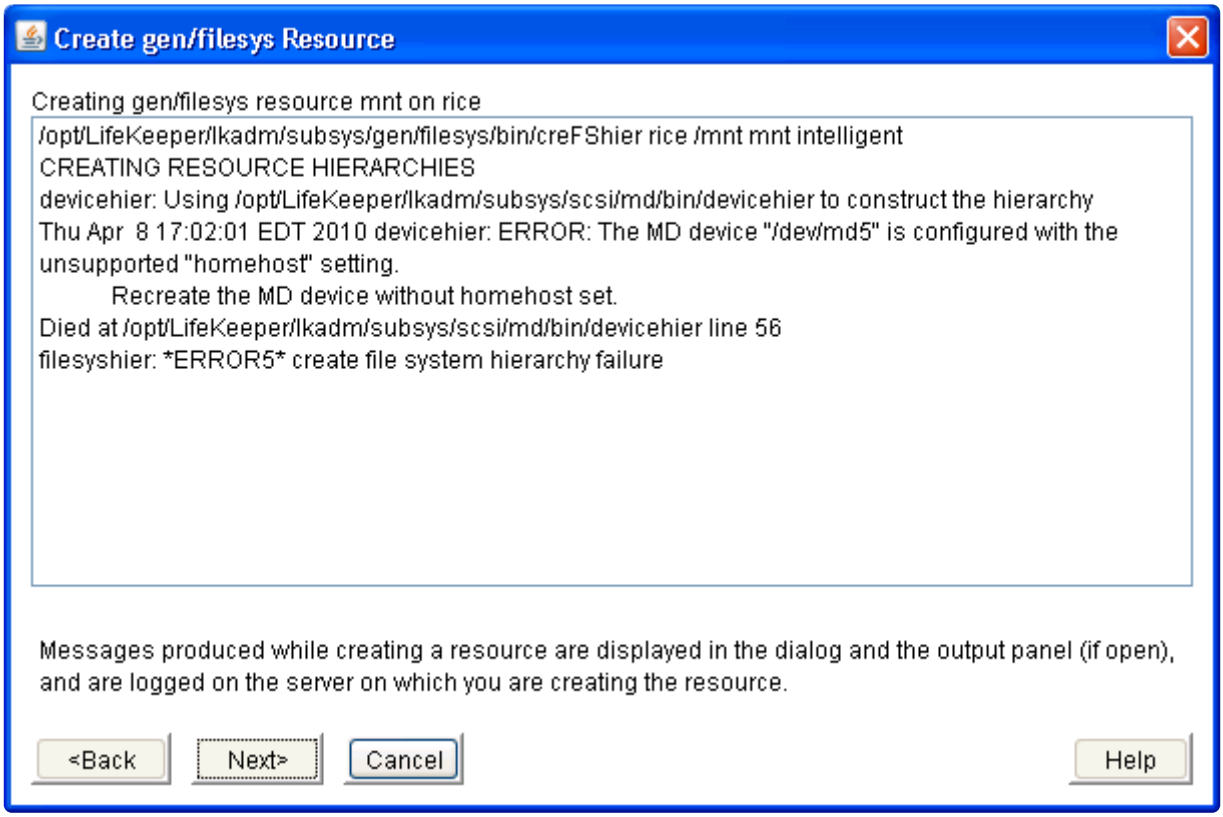


Figure 3: Create File System Hierarchy Failure

Recreating the MD Device Without the Homehost Set

In order to recreate the MD device, the "—homehost='" setting will need to be used:

```
    mdadm –create /dev/md5 –level=1 –raid-devices=2 /dev/sde1 /dev/sdf1
–homehost=''
```

RAID Level Support

The supported RAID levels are linear, RAID 1 (mirroring) and RAID 10 (striped mirror).

Spare Support

Spare components are supported as an element of a specific virtual device. A "spare-group" is not supported.

Support for Raw I/O and Entire Disks

While Figure 1 shows a virtual device residing below a file system, it is important to note that the Software RAID Recovery Kit can support raw access to a virtual device when used in conjunction with the LifeKeeper Raw I/O Recovery Kit, and can manage virtual devices that are composed of one or more entire disks (e.g. /dev/sdc) rather than disk partitions (e.g. /dev/sdc1).

Partitioning Virtual Devices

Linux software RAID does not support direct partitioning of a virtual device. There have been several attempts by individuals to add support for partitioning, but the maintainers of the md driver have not accepted this. In place of direct partitioning, the software RAID HowTo referenced in the Documentation and References section above recommends using LVM. Figure 6 shows a hierarchy using LVM.

MD_ASSEMBLE_OPTIONS

In this version of the Software RAID Recovery Kit, the parameter "—run" has been removed from the mdadm command used to assemble (start) the mirror. This parameter is needed in some error situations where mdadm is not sure about the state of the components. Due to this uncertainty, the data could become corrupted, so by default, this parameter is no longer used. Where before a forced mirror in-service would be attempted, an error will now be displayed similar to the following:

```
    Tue Apr 27 11:46:02 EDT 2010 restore: BEGIN restore of "md23051" on
server "shrek.sc.steeleye.com"

    Tue Apr 27 11:46:06 EDT 2010 restore: start: mdadm: failed to add /dev/
sdc1 to /dev/md1: Invalid argument

    mdadm: /dev/md1 assembled from 0 drives - not enough to start the array
```

Although not recommended, this parameter can be used by adding it to the LifeKeeper defaults file: MD_ASSEMBLE_OPTIONS=—run (this will then be used for every assemble). It is instead recommended that the logs in the cluster be reviewed to determine which component/leg has the best data and then manually assemble the mirror using mdadm.

Note: On some systems (for example those running RHEL 6 or RHEL 7), there is an AUTO entry in the configuration file (/etc/mdadm.conf) that will automatically start mirrors during boot (example: AUTO +imsm +1.x –all). Since LifeKeeper requires that mirrors not be automatically started, this entry will need to be edited to make sure that LifeKeeper mirrors will not be automatically started during boot. The previous example (AUTO +imsm +1.x –all) is telling the system to automatically start mirrors created using imsm metadata and 1.x metadata minus all others. This entry should be changed to "AUTO -all", telling the system to automatically start everything "minus" all; therefore, nothing will be automatically started.

IMPORTANT: If system critical resources (such as root) are using MD, make sure that those mirrors are started by other means while the LifeKeeper protected mirrors are not.

6.7.4. Software RAID Hierarchy Creation and Administration

LifeKeeper software RAID hierarchies are created automatically during the hierarchy creation process for resources that sit on top of virtual devices. The creation and extension of hierarchies containing the software RAID resource types will always be driven by the create and extend processes of a higher-level resource type, likewise the delete and unextend.

Figure 4 is a LifeKeeper GUI screen shot showing a complete hierarchy containing software RAID resources. The resources in the hierarchy are displayed using the default display showing the LifeKeeper tags. Figure 5 displays the same hierarchy with the display showing the LifeKeeper IDs.

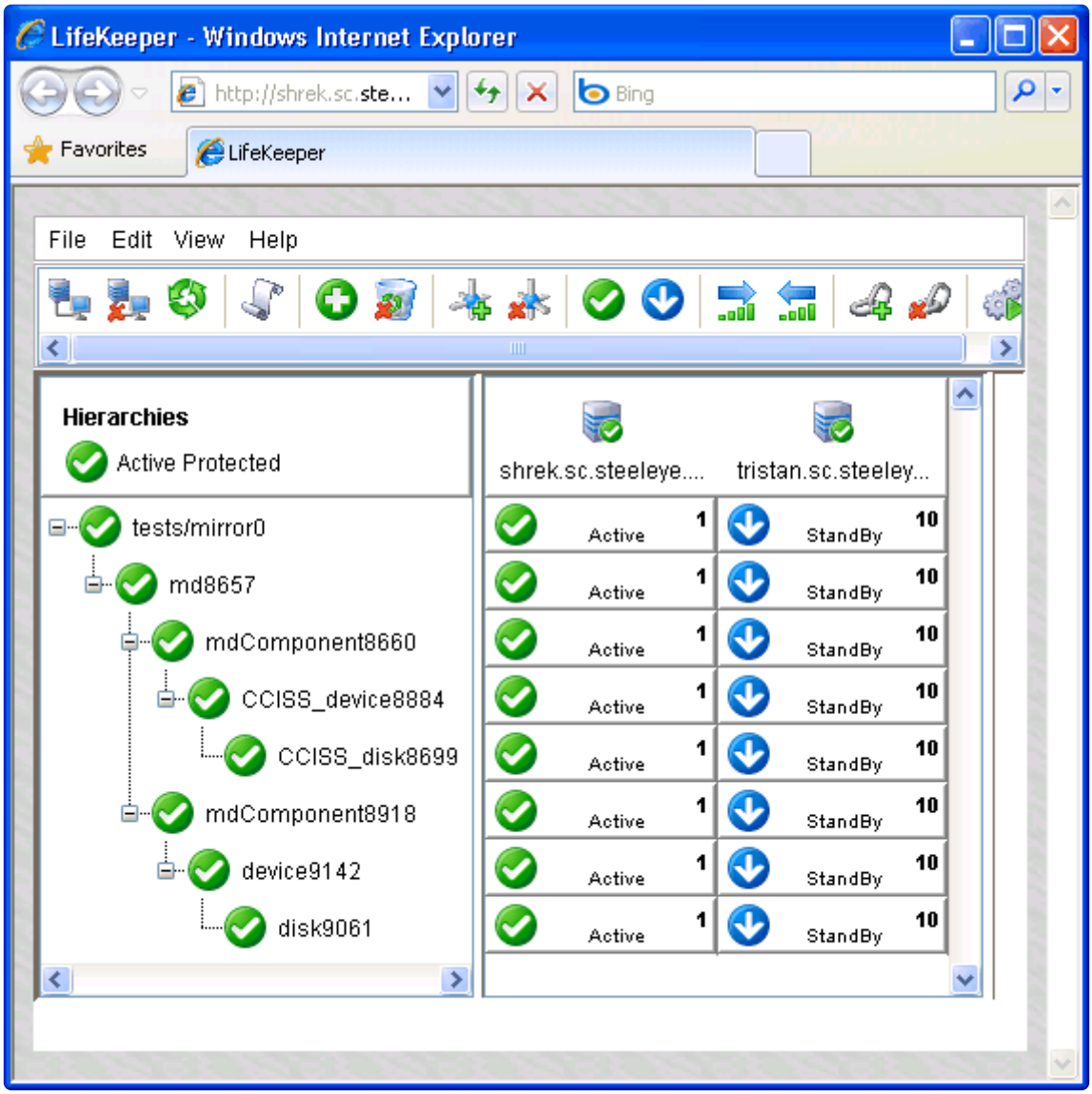


Figure 4: LifeKeeper Hierarchy Containing Software RAID Resources

The hierarchy pictured in Figure 4 is a file system hierarchy, created by selecting the File System recovery kit under the Edit > Server > Create Resource Hierarchy menu selection. It consists of a file system resource, tests/mirror0, mounted on a software RAID virtual device, tag md8657. That virtual

device is a RAID-1 (mirror) with 2 components: mdComponent8660 and mdComponent8918. The components are configured on partitions on different underlying device types, one being from the CCISS recovery kit (CCISS_device8884) and the other using the default SCSI recovery kit (device9142). The hierarchy also includes the underlying disk devices, CCISS_disk8699 and disk9061, below each of the disk partitions. The hierarchy can also include a “terminal resource” to tie the bottom of each hierarchy to a single resource. For more information on the terminal resource, see Terminal Resource in the Best Practices section.

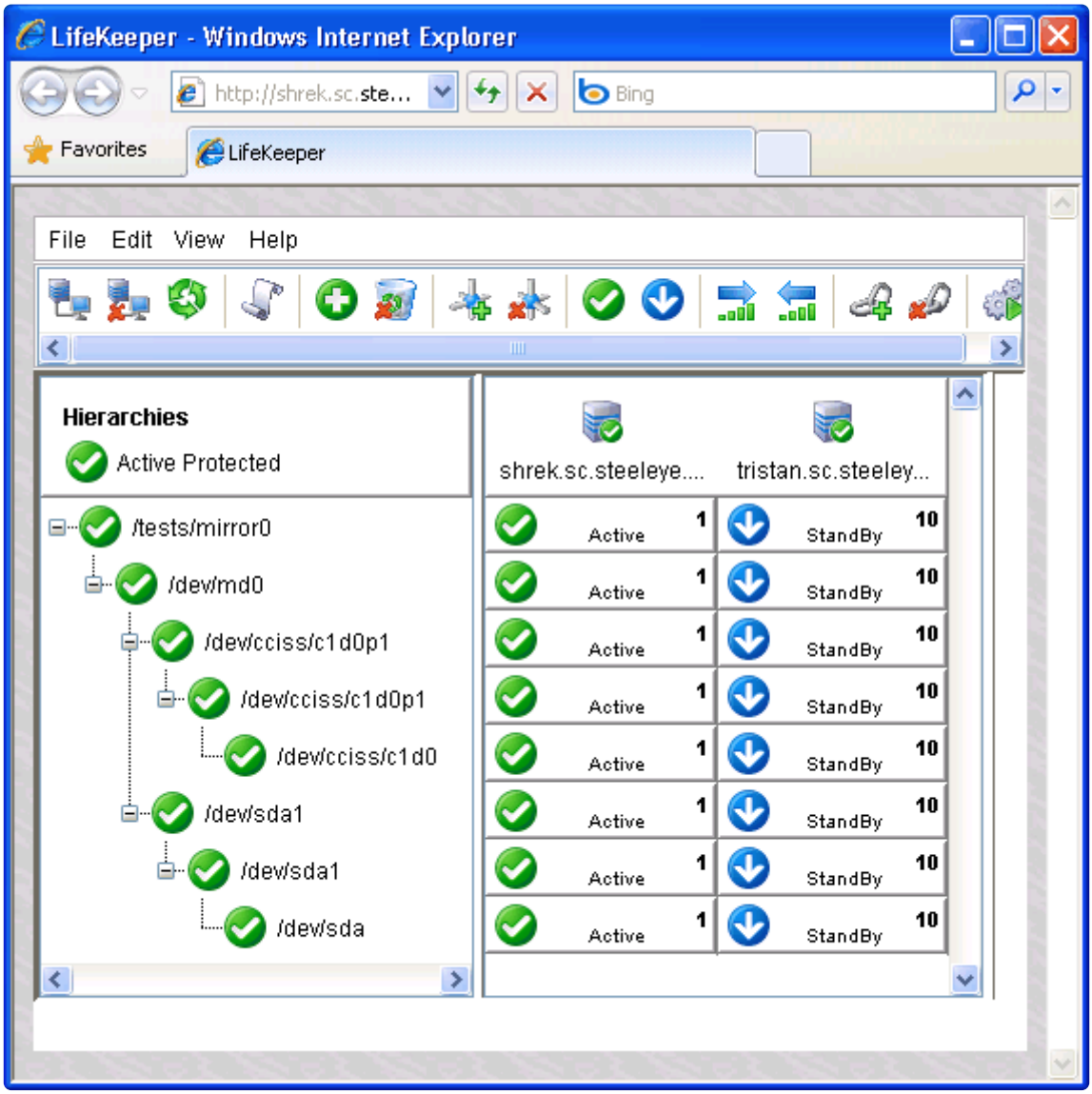


Figure 5: LifeKeeper Hierarchy Containing Software RAID Resources

Notice that the mdComponent resource has the same ID as the underlying device. This is unusual in a LifeKeeper hierarchy but is a result of the mdComponent being a resource to allow the Software RAID Recovery Kit to show the state of each component in a virtual device.

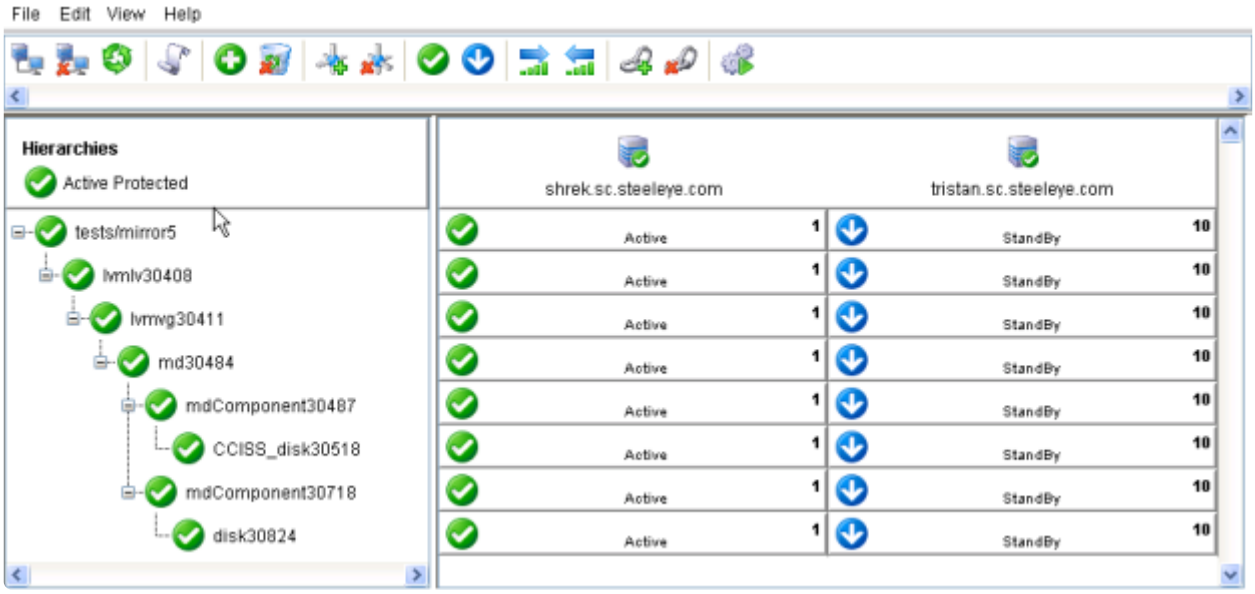


Figure 6: LifeKeeper Hierarchy Containing Software RAID Resources

Figure 6 above shows a hierarchy using LVM with software RAID.

6.7.4.1. Creating a Software RAID Resource

To create a hierarchy in which a file system or higher-level application uses a software RAID virtual device, the following high-level procedure should be followed.

1.
2. Determine the desired configuration of your virtual devices. In doing this, keep in mind all of the disk resources associated with a given virtual device must move together from one server to another in the LifeKeeper cluster.

3. On the system which is to be the primary server for your application, create the desired virtual devices using mdadm(8) provided by the mdadm package and described in the Linux Software RAID HowTo and the mdadm(8) on-line manual page referenced in the Documentation and References section above. When creating the virtual device, a persistent superblock MUST be used. Refer to the section Persistent Superblock above for further details.

4. If using shared storage, ensure that all components of the virtual device are properly shared between the machines in the LifeKeeper cluster on which the protected application will be run.

5. Create file systems on each virtual device. If raw I/O will be used instead, bind a raw device to each of the virtual devices.

6. Configure the protected application on the file systems, following the configuration instructions in the administration guide for the LifeKeeper recovery kit associated with the application.

7. Create and extend the application hierarchy following the instructions in the appropriate application recovery kit administration guide

6.7.4.2. Software RAID Reconfiguration

One of the primary benefits of using software RAID is the ability to dynamically add, remove and resize virtual devices as storage requirements change. Because this may involve adding or deleting physical partitions or disks from a virtual device definition, the Software RAID Recovery Kit includes a mechanism for modifying an existing resource hierarchy to reflect such a change.

All virtual device and file system reconfigurations should be performed outside of LifeKeeper prior to modifying the LifeKeeper hierarchy to reflect the changes. Refer to the *Software RAID HowTo* document referenced in the Documentation and References section for information about how this is done. If any of the steps require a resource that is being protected by LifeKeeper to be unmounted or unconfigured, be sure to use the LifeKeeper GUI to do so, using the **Out-of-Service** operation.

To update a LifeKeeper hierarchy following these changes, first access the **Resource Properties** dialog for the modified **md** resource, either by right-clicking on the **md** resource and selecting **Properties**, or by using the **Edit > Resource > Properties** menu selection and selecting the appropriate **md** resource in the **Select Resource** field. The resulting **Resource Properties** dialog should look like the one pictured in **Figure 7: Software RAID Resource Properties Dialog** below, including the **Status** and **Reconfigure** buttons near the bottom.

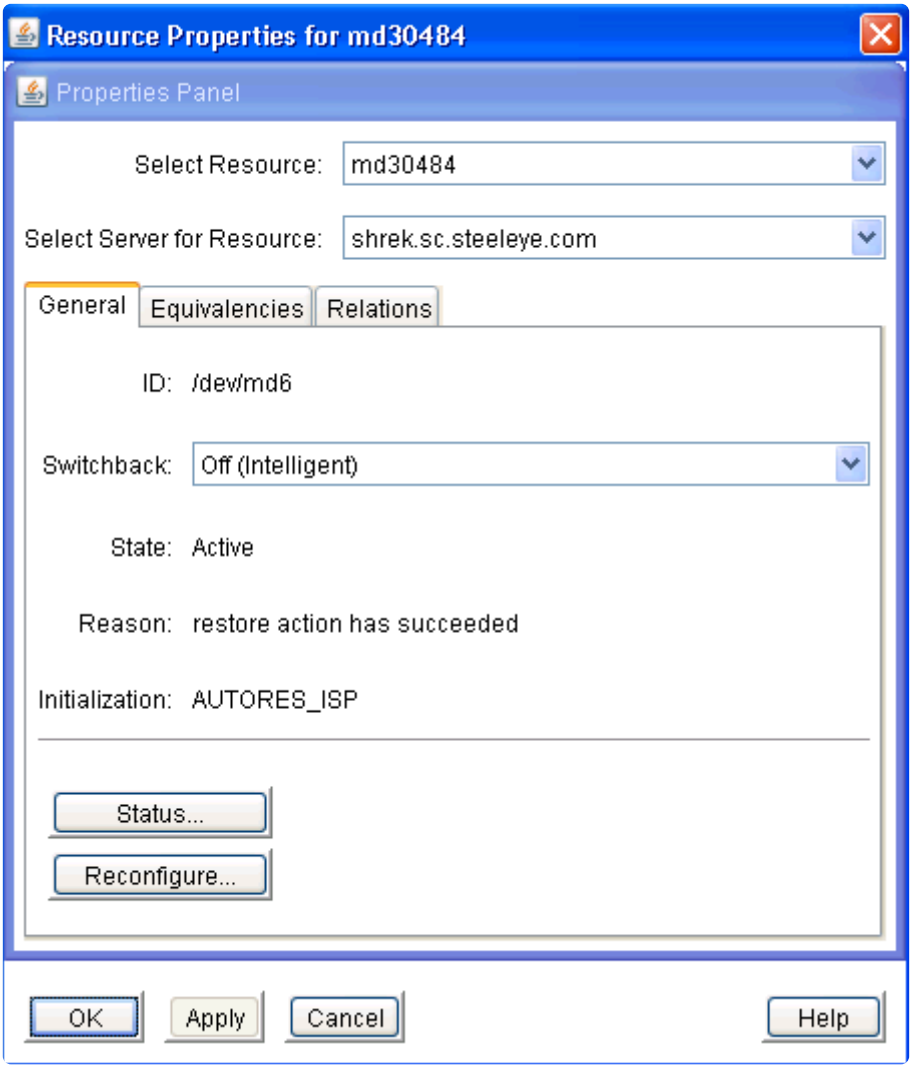


Figure 7: Software RAID Resource Properties Dialog

Clicking the **Status** button will display an information box displaying the current status of the virtual device. **Figure 8: Software RAID Status** below shows an example of the status of a virtual device where all components are functioning properly.

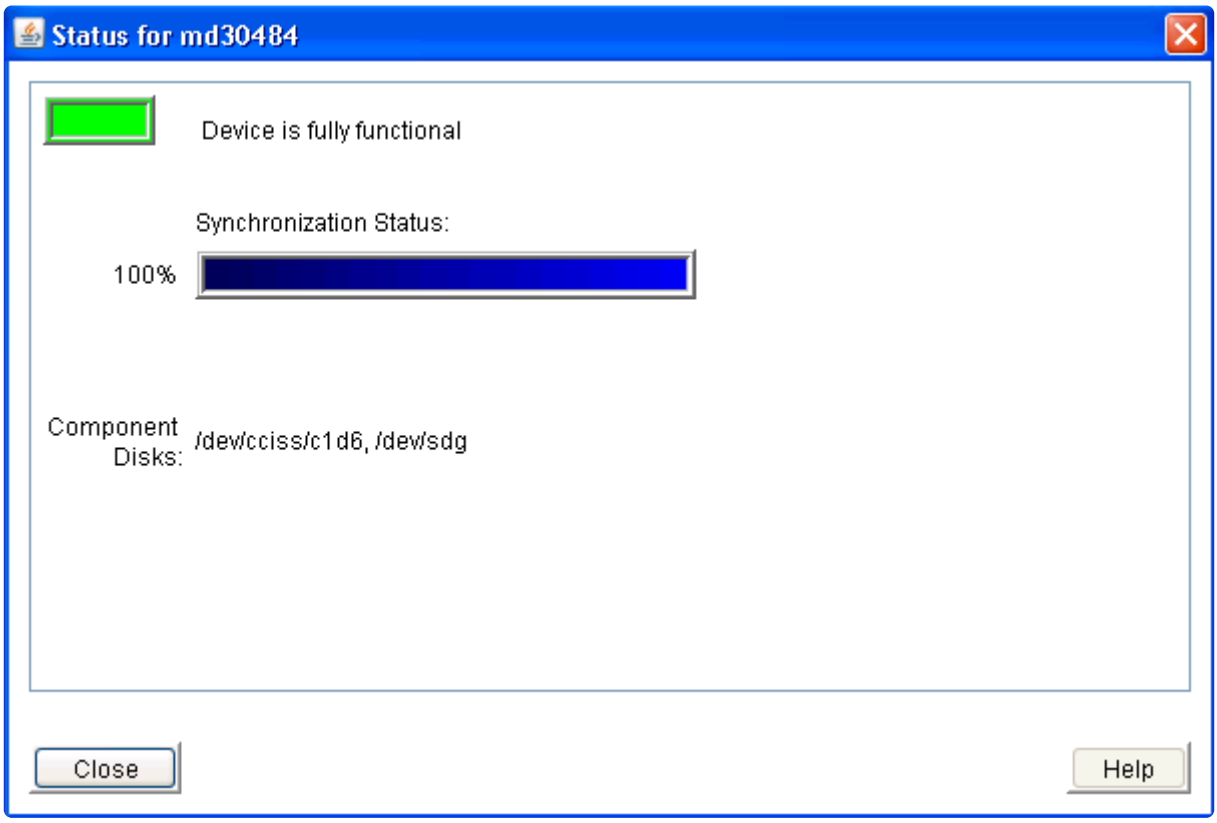


Figure 8: Software RAID Status

Clicking the **Reconfigure** button initiates the mechanism for reconfiguring your hierarchy to reflect any modifications to the virtual device resource. After a brief pause, an information box will display the modifications that LifeKeeper has detected.

The following three figures show examples of the status and configuration information boxes that would be displayed when a device is removed from a virtual device.

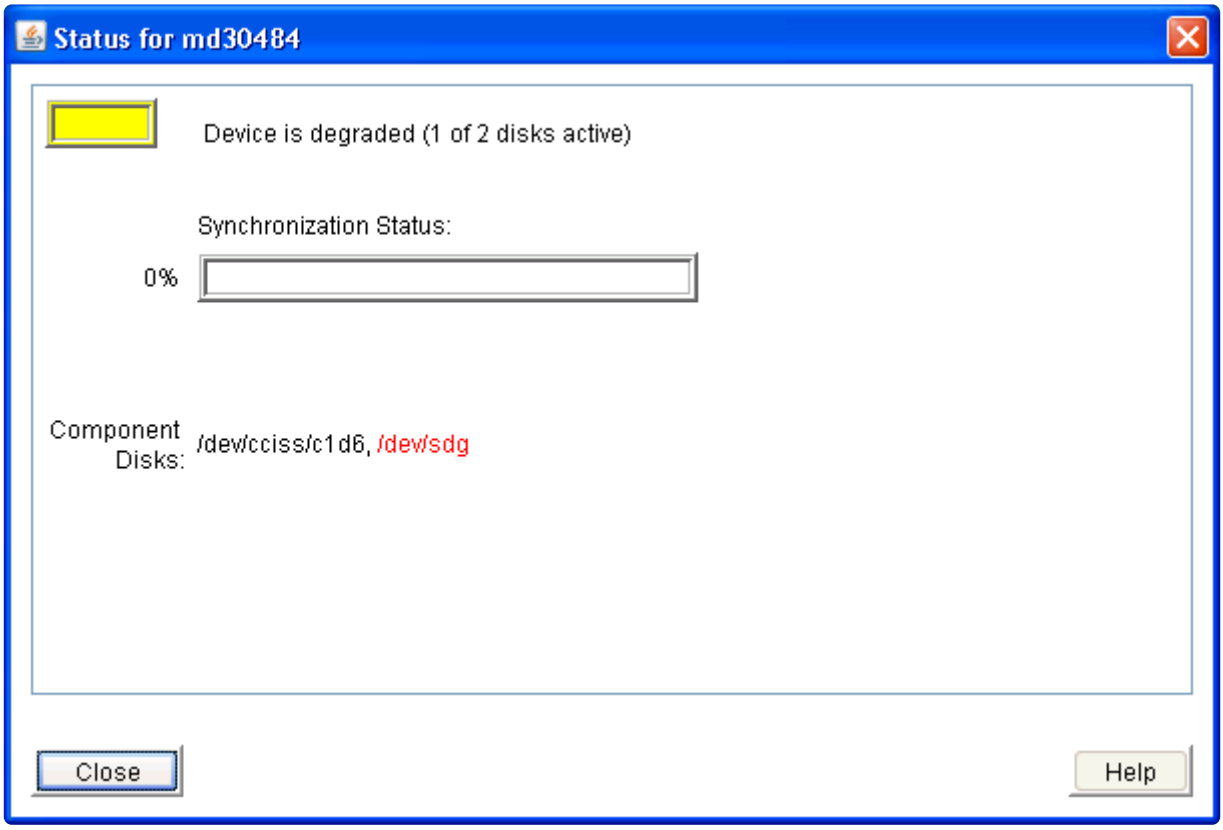


Figure 9: Software RAID Status for a Deleted Device

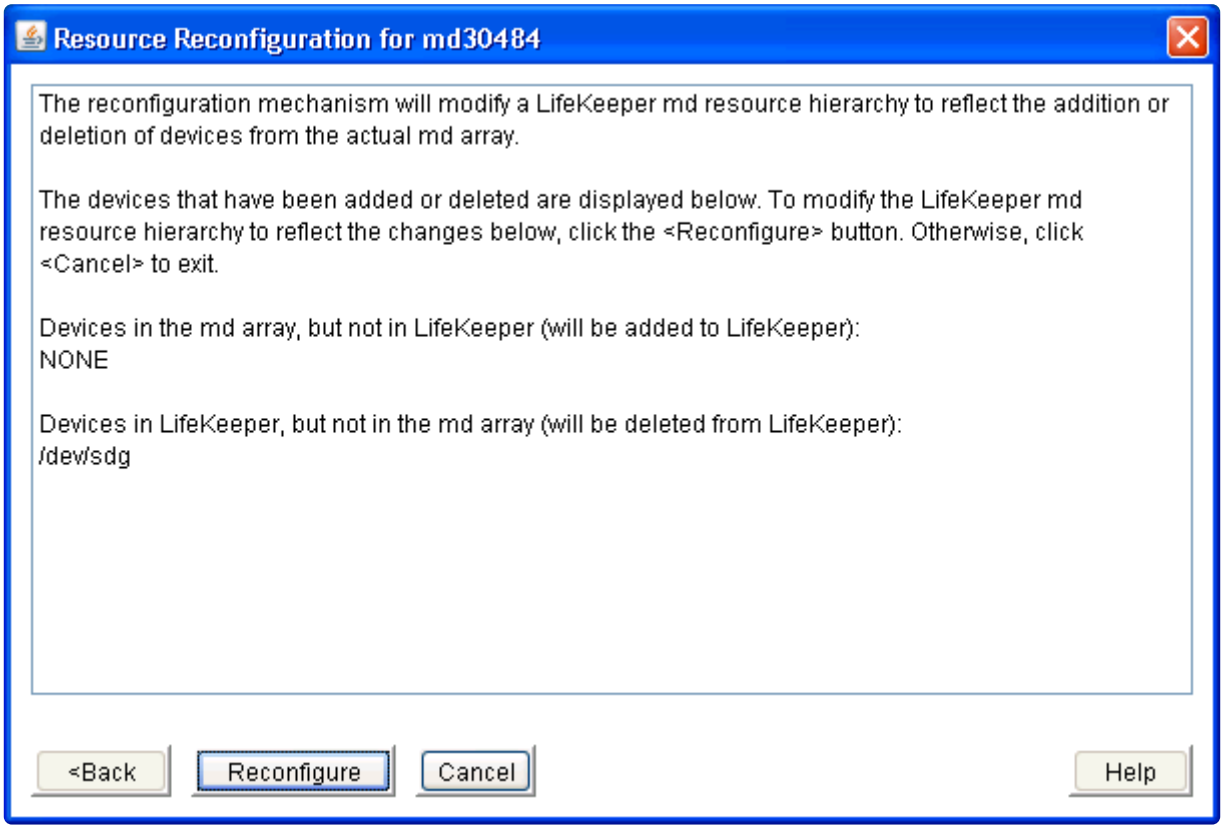


Figure 10: Software RAID Reconfiguration for Deleted Device

As stated in the information box, to reconfigure the LifeKeeper virtual device to reflect the changes that have been detected, simply click the **Reconfigure** button. To cancel the LifeKeeper hierarchy

modification, click **Cancel**.

After clicking the **Reconfigure** button, an information box will appear, showing the progress of the reconfiguration procedure, as shown in **Figure 11: Software RAID Completed Reconfiguration for Deleted Device** below. When the process has been completed successfully, the **Done** button will become enabled. Clicking **Done** will close the information box and display the **Resource Properties** dialog.

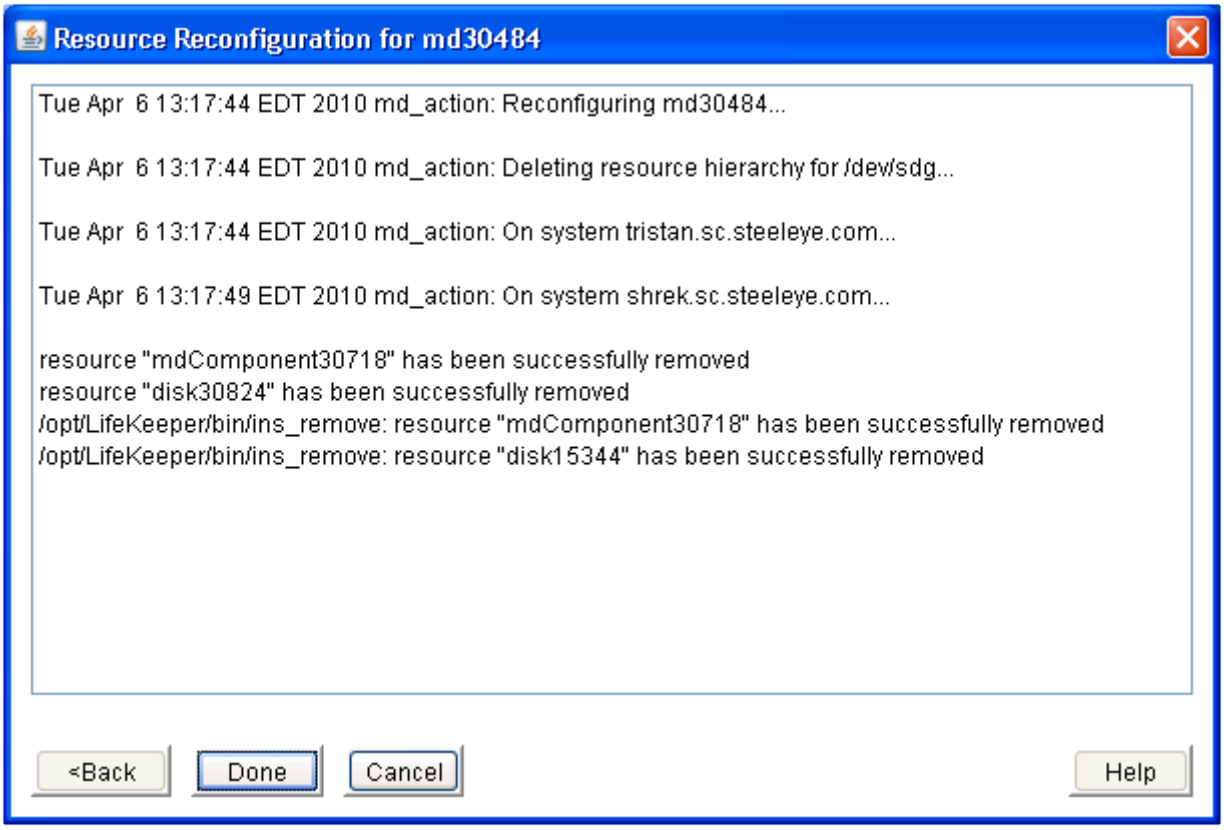


Figure 11: Software RAID Completed Reconfiguration for Deleted Device

The following four figures show examples of the status and configuration information boxes that would be displayed when a device is added to a virtual device.

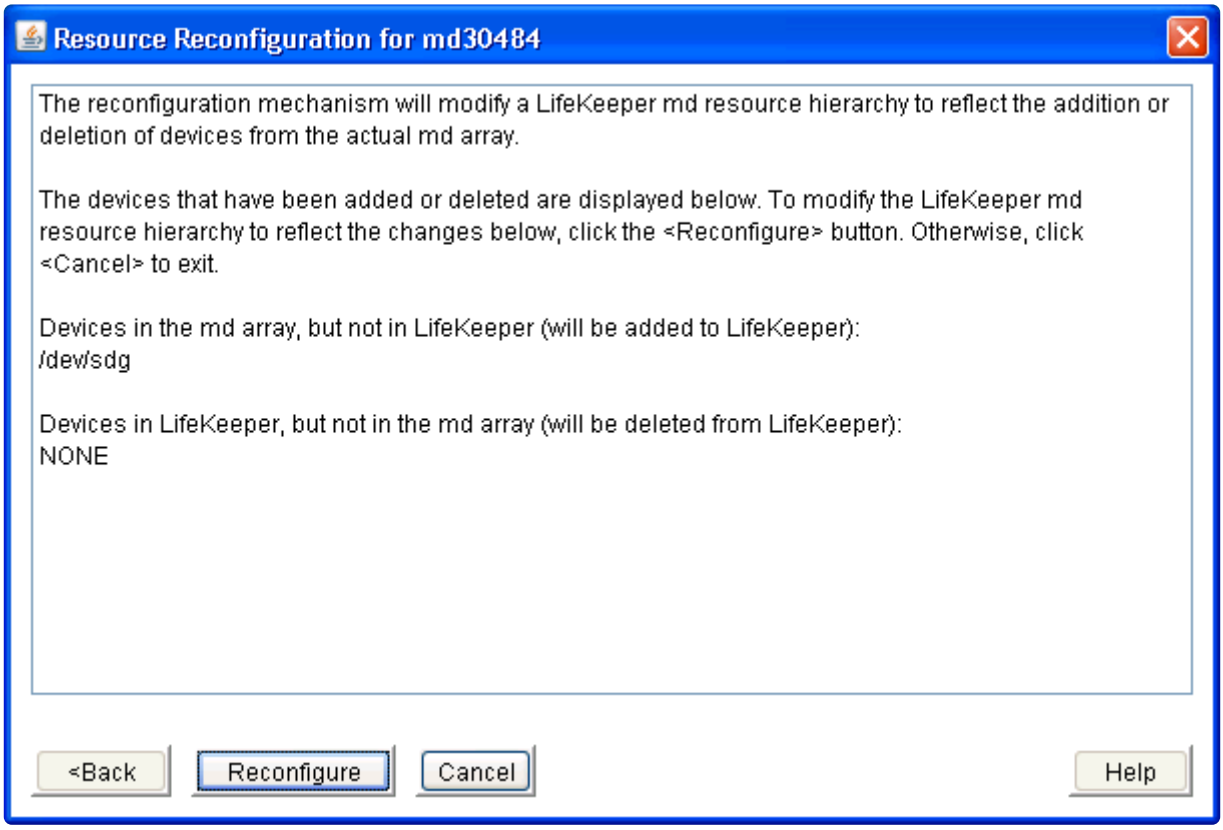


Figure 12: Software RAID Reconfiguration for Added Device

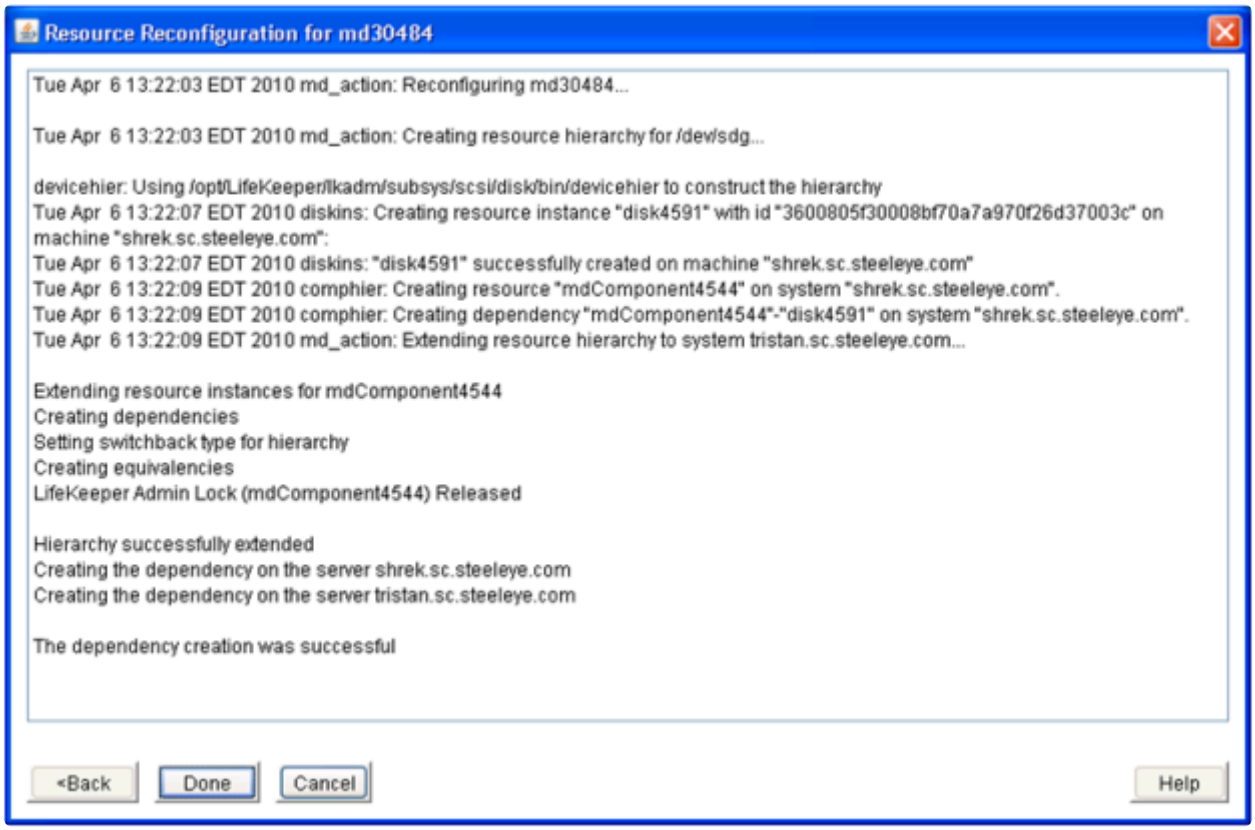


Figure 13: Software RAID Completed Reconfiguration for Added Device

While the component is being configured into the virtual device, the **Status** button will show the synchronization progress.

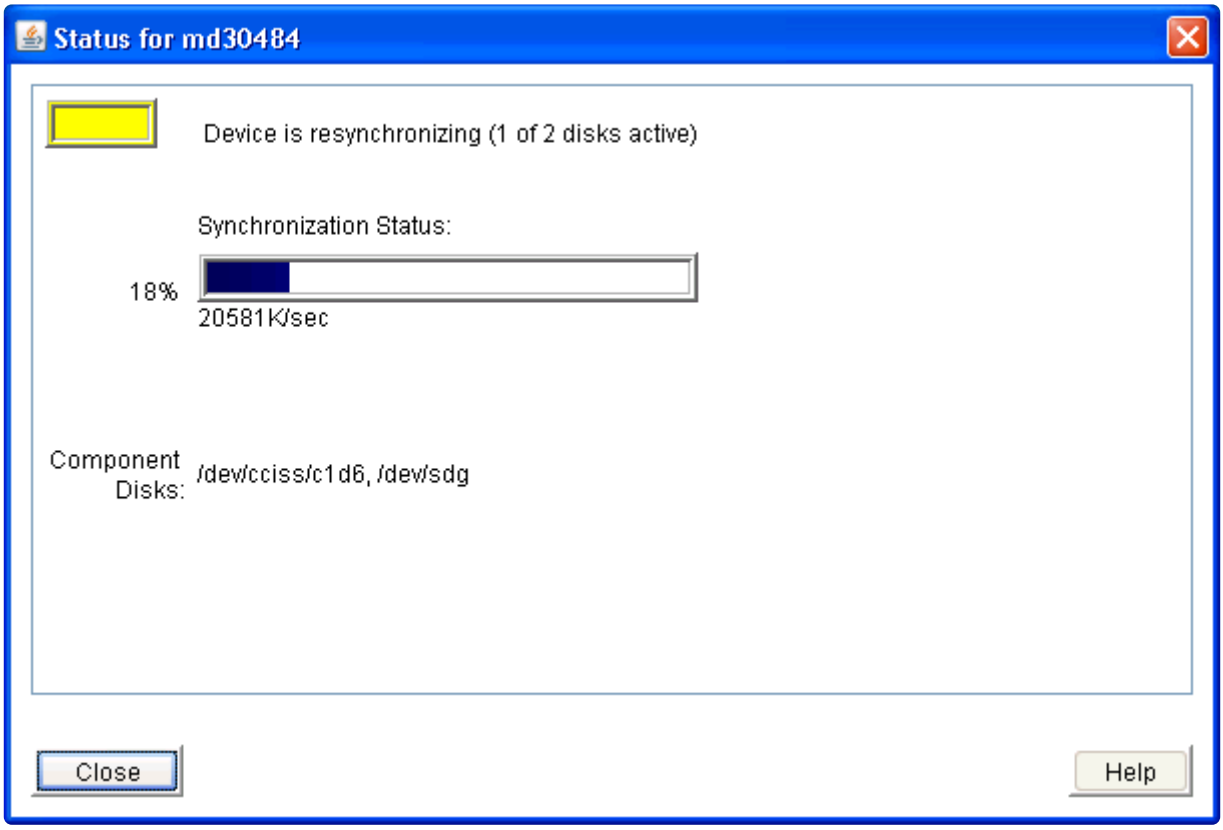


Figure 14: Software RAID Status During Resynchronization

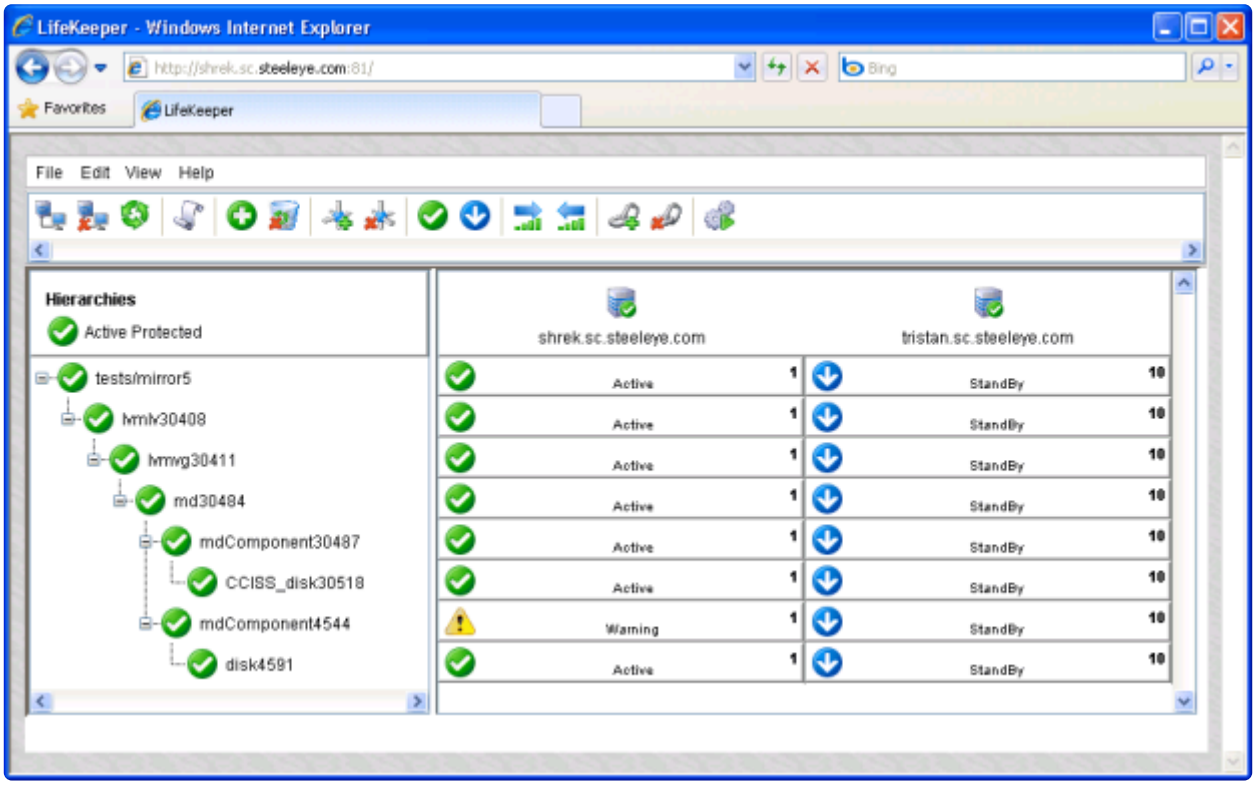


Figure 15: LifeKeeper Hierarchy During Resynchronization

6.7.4.3. Software RAID Repair

If one of the legs of a mirror fails, a repair can be done on that leg.

If a problem occurs, the resource will be marked **OSF**. (**Note**: An email notification will occur if enabled.)

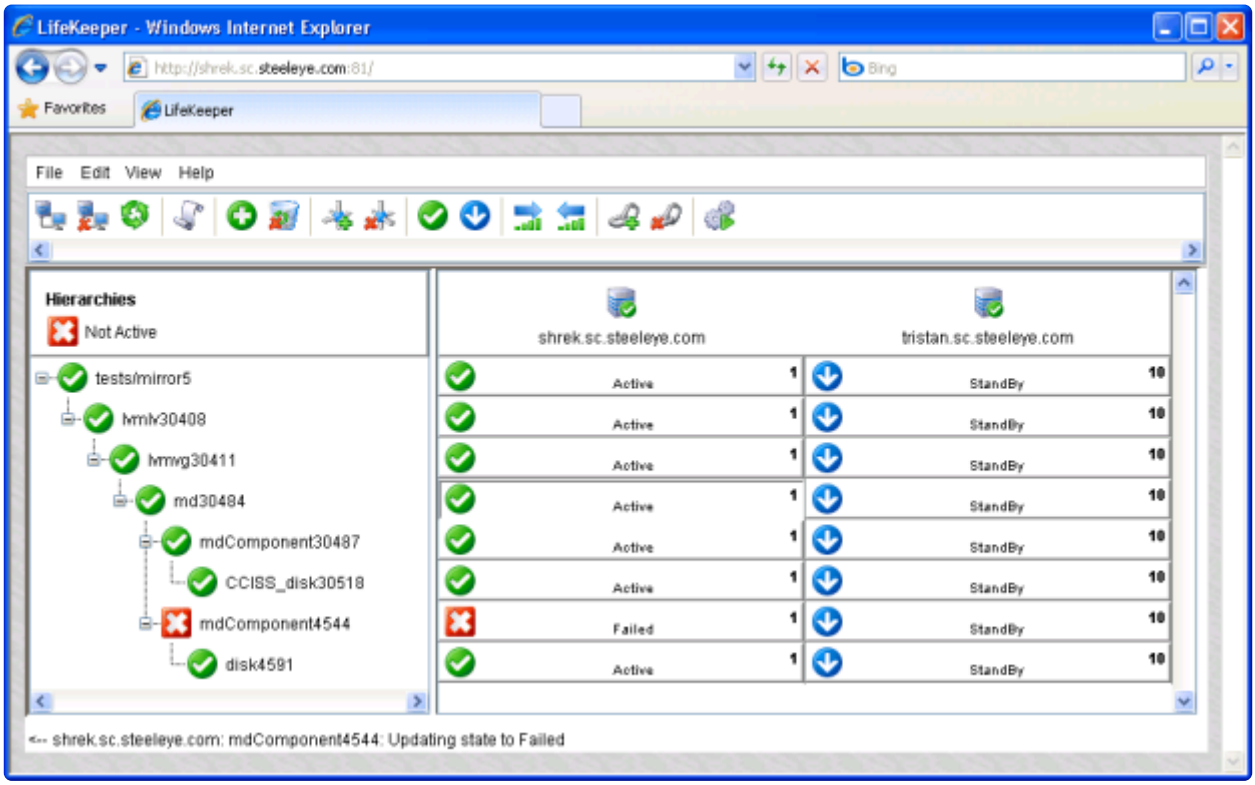


Figure 16: LifeKeeper Hierarchy With Failed Component

The **mdComponent** could be marked **OSF** while the disk is okay, but the component is marked **"faulty"** in the mirror. This can be due to some issue detected by mdadm when the device was brought on-line (check the error log for further information) or could be due to a manual operation where the mdadm utility was used to "break" the mirror.

The **mdComponent** as well as the underlying disk/device could be marked **OSF** if they failed during the in-service operation. For example, the disk was "broken" or physically not connected when the virtual device was started.

The following screen shots depict an array failure from before the array failed and initial handling of that failure to updating the state to "failed" and bringing it back in service. (These screen shots include an example using a "terminal resource" to tie the bottom of each hierarchy to a single resource.)

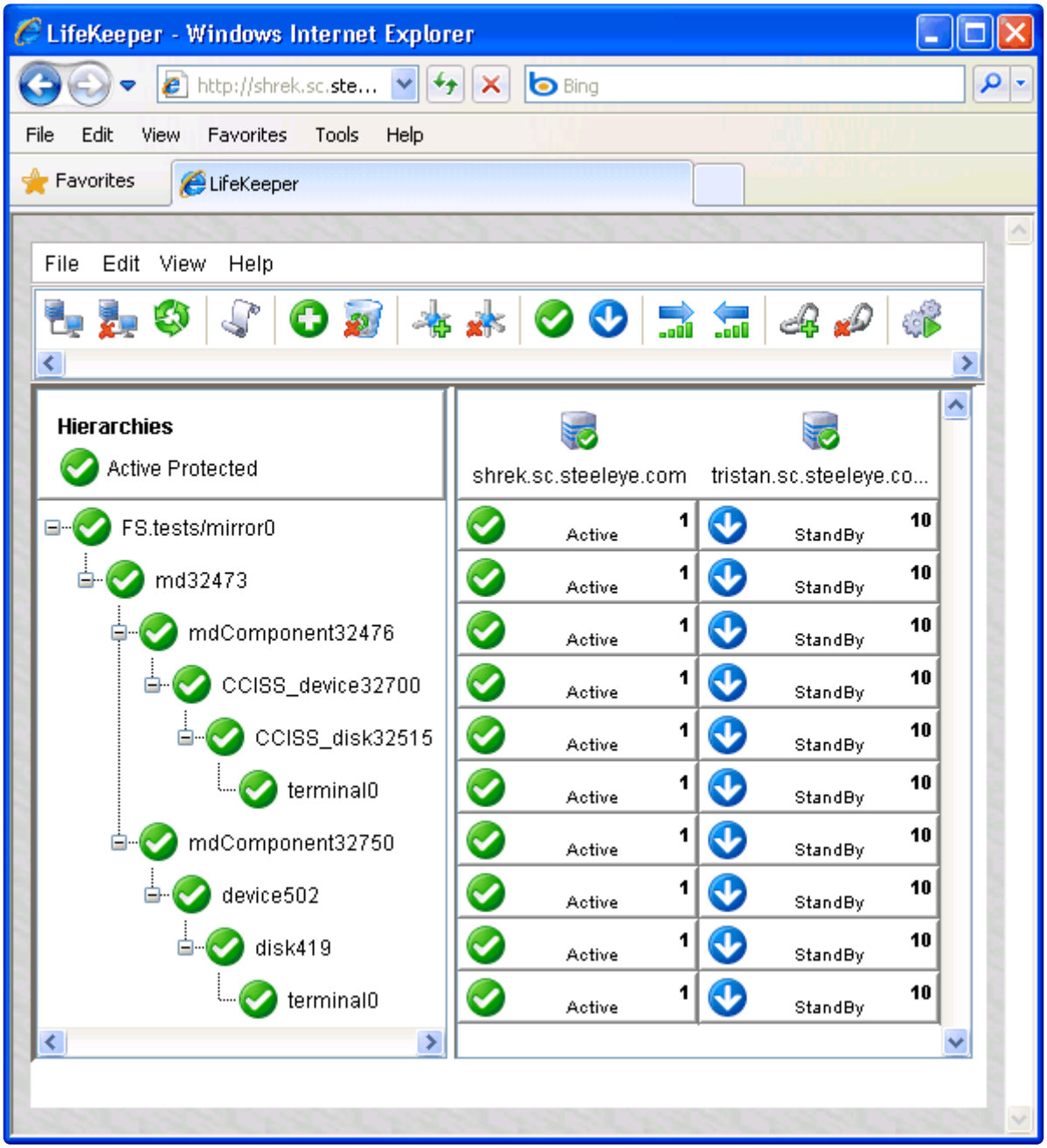


Figure 17 – Before Failure of Array

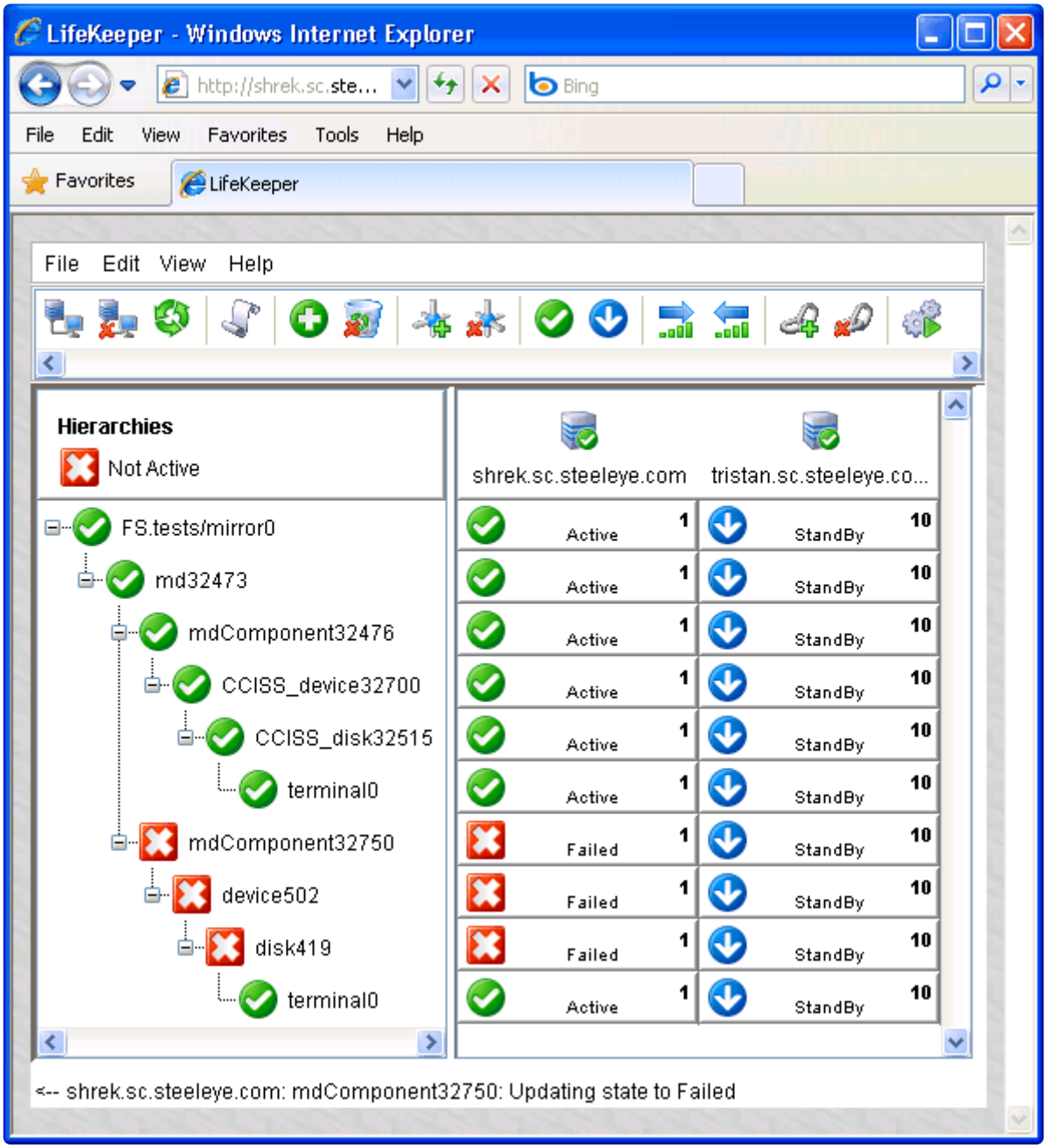


Figure 18 – After Failure of Array

When the failure of the array is initially handled, all resources will be marked **OSF**. During this failure, IOs continue to the good component or leg of the mirror.

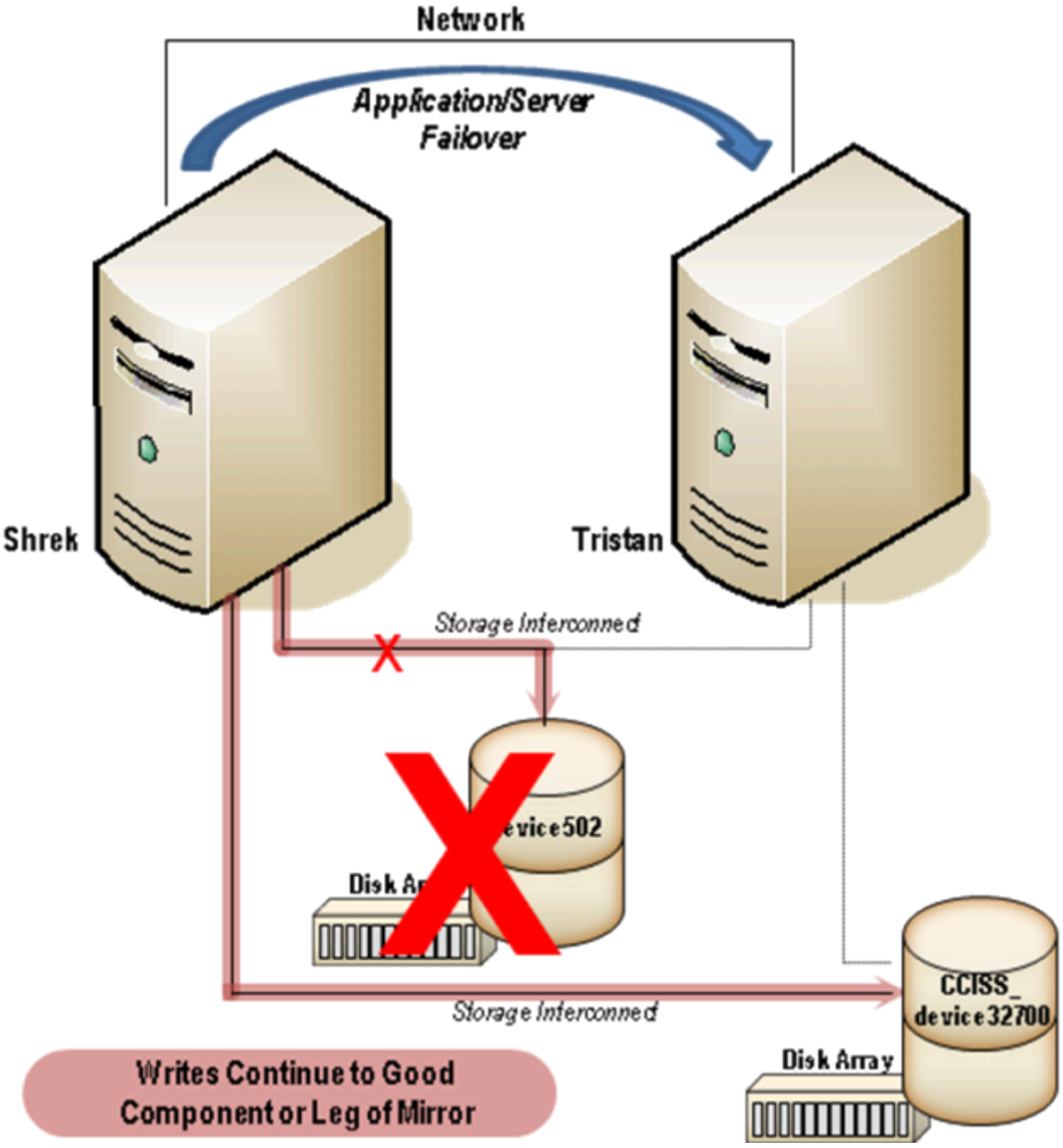


Figure 19 – Failed Disk Array

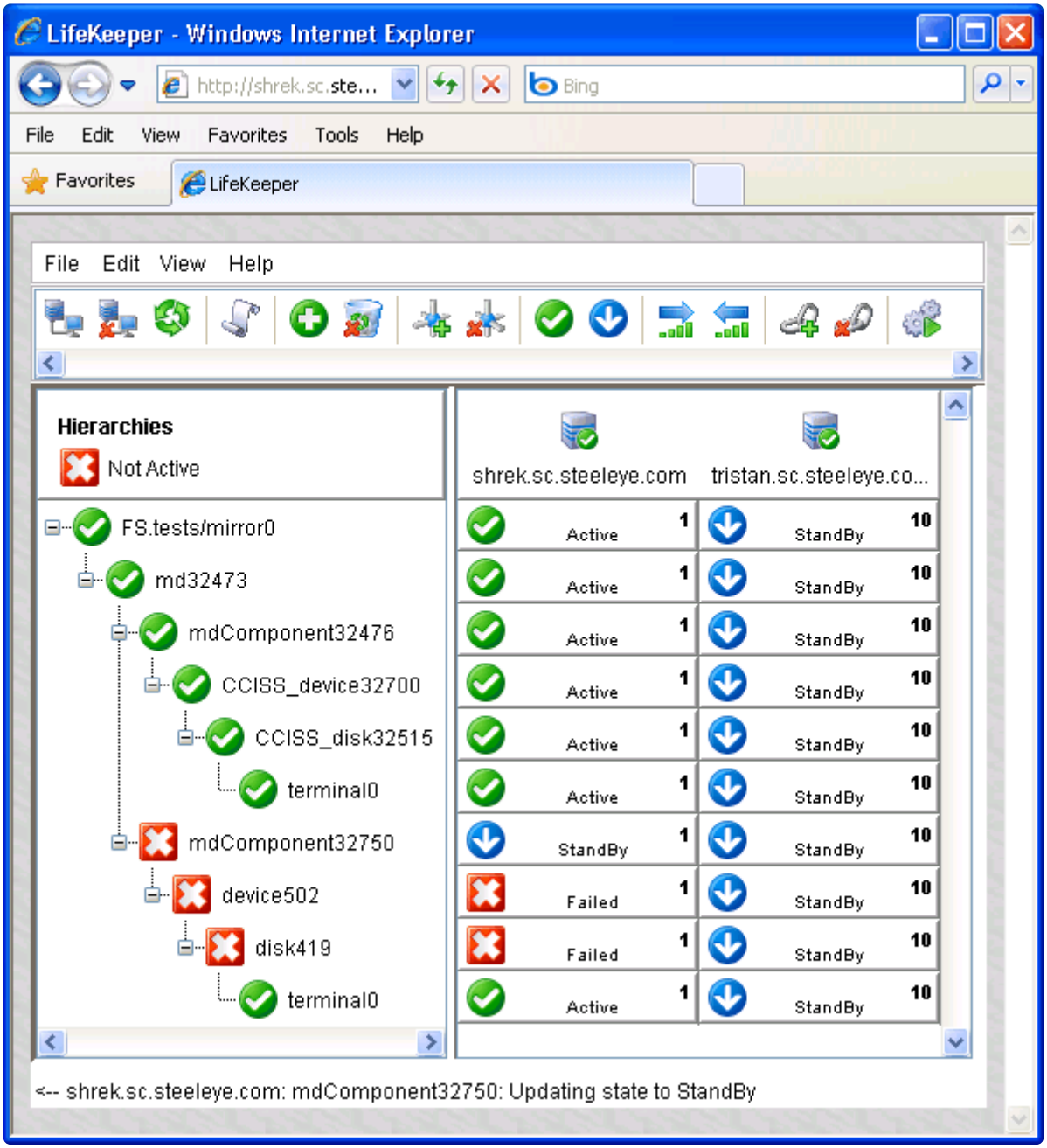


Figure 20 – Updating Failed Component to Standby

If the failed component was successfully removed from the mirror configuration during the error handling, the resource will transition to **OSU**. This is done when the MD quickCheck runs after the failure. If, during the handling, the failed component could not be removed from the mirror configuration, then the resource will remain in the **OSF** state.

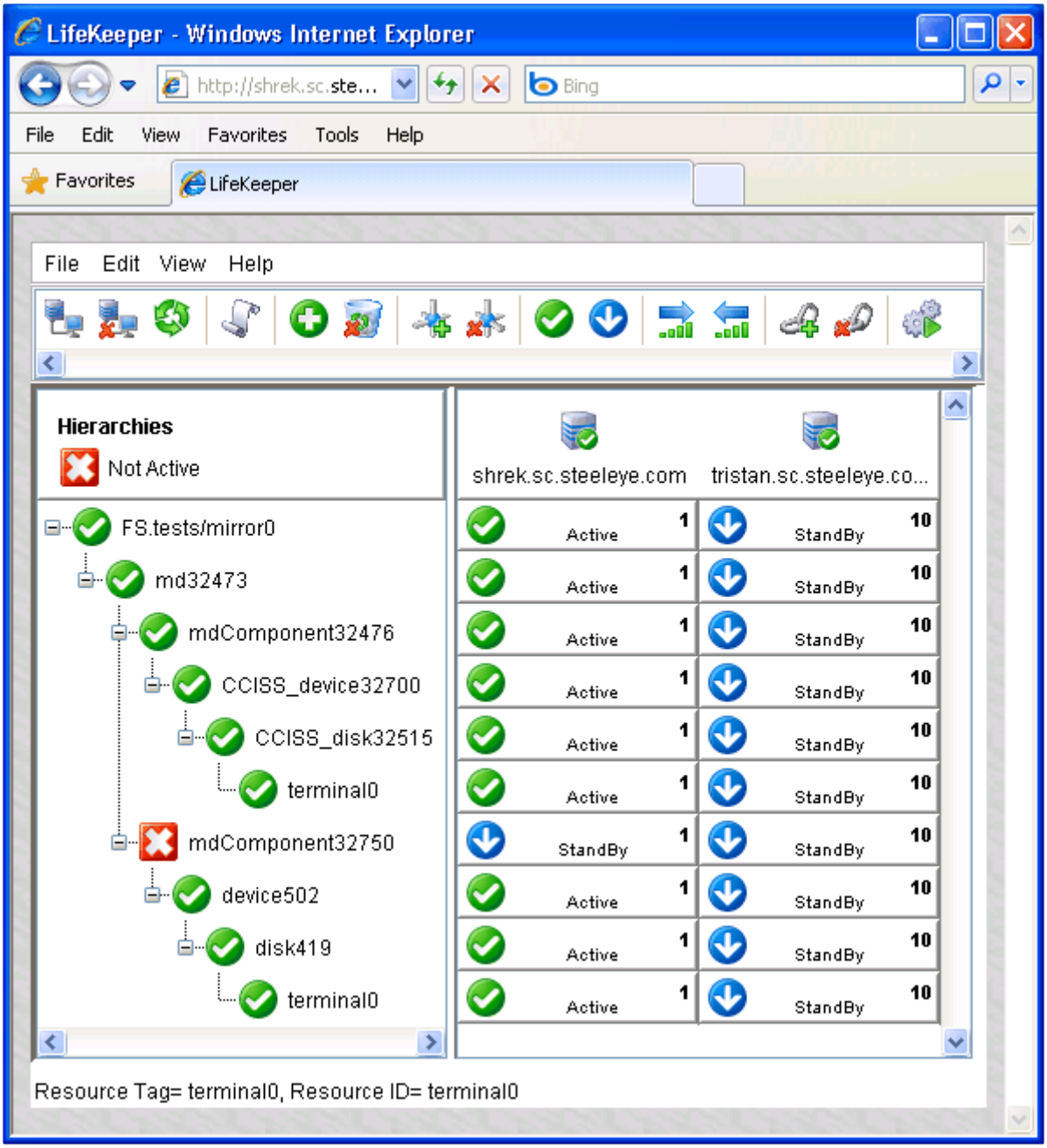


Figure 21 – Restored Storage Resources

If the server has to reboot while in the failed state, perhaps to repair the failure to the storage, then the storage resources under the failed component will be restored (if it was properly repaired), but the failed component will not automatically be re-added into the mirror. An in-service (from the GUI or using `perform_action(1M)`) of the failed component will re-add the failed component. This will trigger a resumption of IO to the leg. The mirror will then do a partial resync if an internal bitmap is configured or a full resync will be done otherwise.

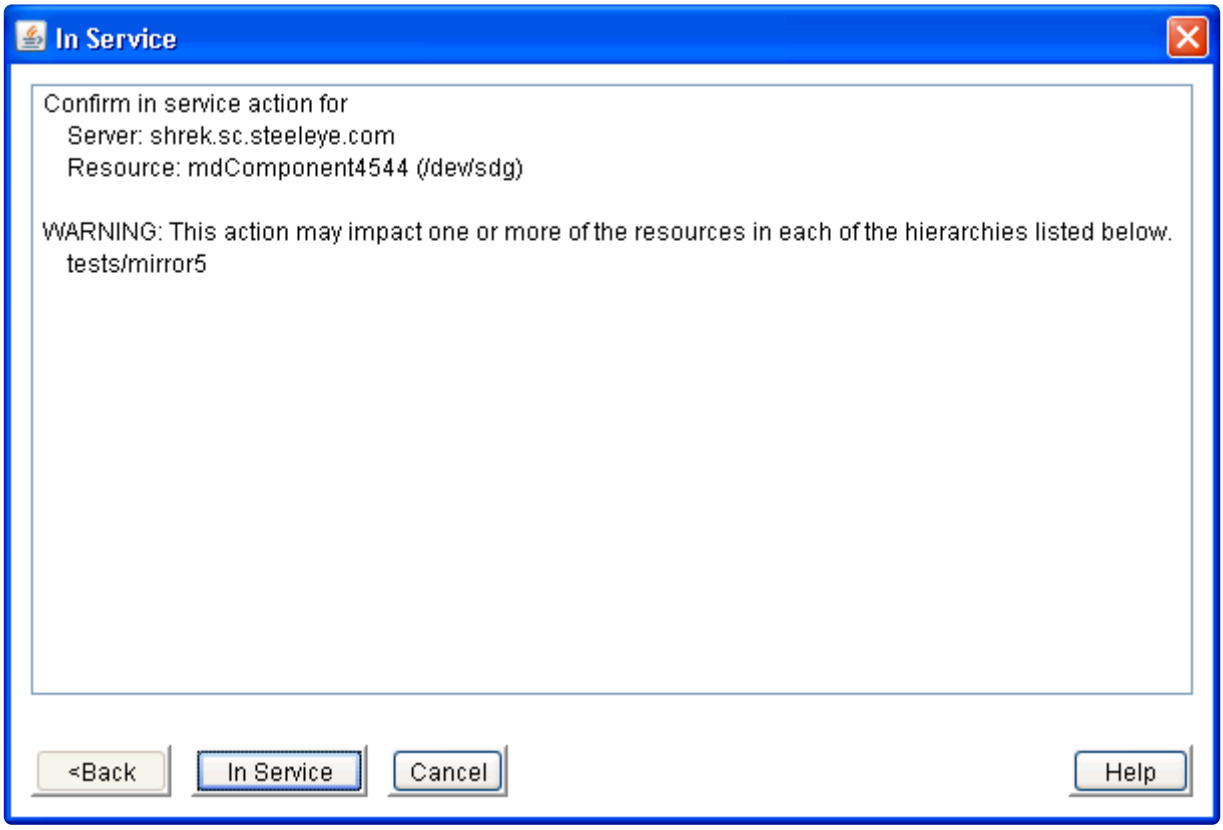


Figure 22: Software RAID In-Service Status

If the failed leg is repaired manually in the virtual device, LifeKeeper will automatically detect the change when quickCheck runs. The state of the resource will change to reflect its new state. However, if the resources below the component are failed, aka the device and/or disk, those states will not be updated. **To update those states, the GUI or perform_action(1M) must be used to bring the resource(s) in-service.**

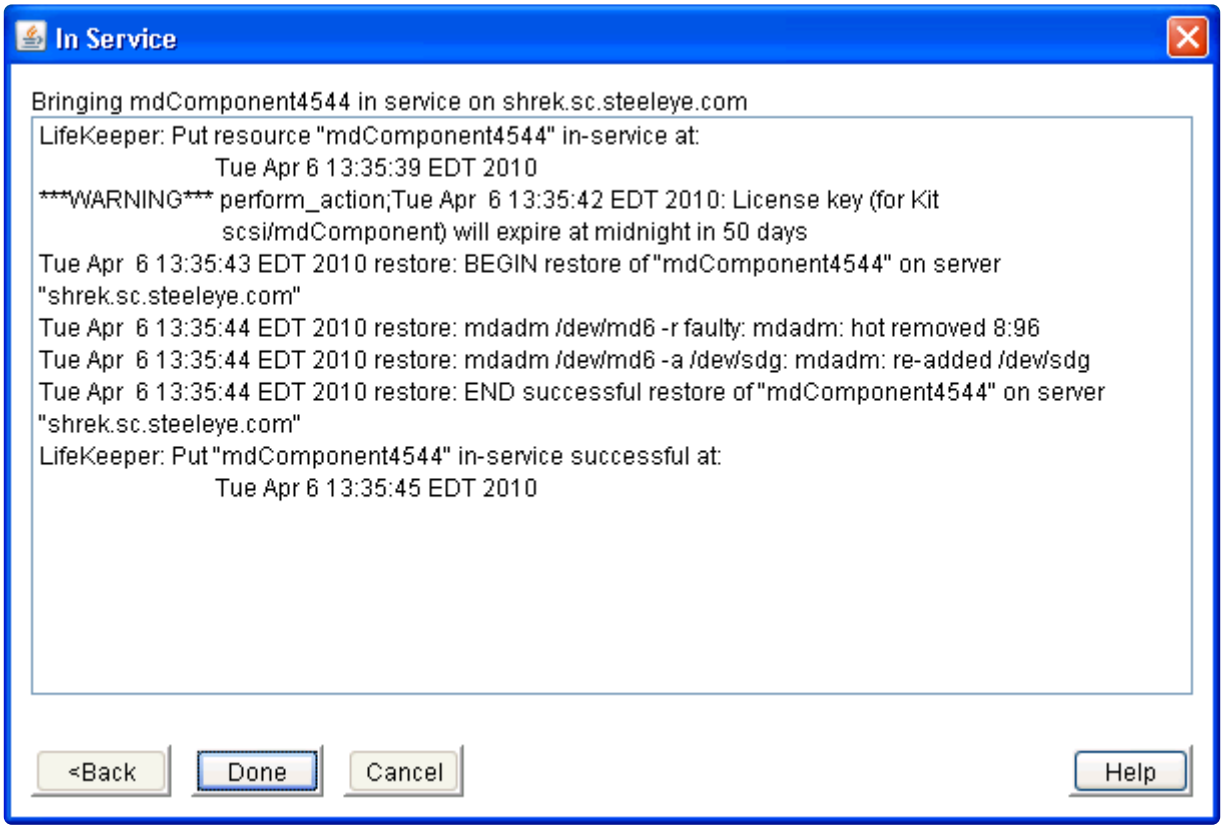


Figure 23: Software RAID Successful In-Service

IMPORTANT: When there is a failure that causes resources to be marked OSF and especially failures that result in resources being moved from one system to another (via a sendevent), it is important that the administrator verify that the failed resource is repaired before trying to bring the resource in-service where it failed.

An example is with the MD kit where there is a complete loss of all paths. When all paths to a mirror fail, the MD kit will recover the failure by moving the mirror to the standby system. The kit will try to clean up or remove all parts of the hierarchy on the failed system before trying to bring the parts in-service on the standby system. However, in many cases, these parts or resources cannot be completely cleaned up due to the failure.

When the administrator repairs the failure, the administrator must also make sure all residual OS items are cleaned up. If there is a mounted file system on the failed mirror, this file system often cannot be unmounted, so even though LifeKeeper moves the file system to the standby system, the failed system will show the file system as mounted (via the mount command). This will cause failures if the administrator then moves the LifeKeeper file system hierarchy back to the repaired system.

The administrator needs to not only repair the failed paths but also needs to make sure all parts of the hierarchy are cleaned up (MD device is still not configured, file system is not mounted, application is completely stopped, etc). A clean reboot may be necessary to make sure all aspects of the hierarchy are cleaned up.

6.7.5. Software RAID Best Practices

Terminal Resource

In order to avoid some failures seen when all components of a mirror fail, it is recommended that a terminal resource (or instance or leaf node) be created. This terminal resource is a “gen app” resource that is used to tie all of the components (legs) of a mirror to a single point. This terminal instance is useful for several reasons.

- It provides a single point to take the full hierarchy out of service rather than having to select each component directly.
- It avoids some confusing transient situations where part of the hierarchy is active on one node and part is active on another node. This is especially seen while a hierarchy is being moved from one server to another. When the move is complete, all resources should end up on the same server, but while LifeKeeper is moving everything, it can look strange.
- It avoids some error situations where LifeKeeper is trying to quickly move resources from one system to another (all path failure), but the process of starting a resource is slow due to cluster failures. This will force LifeKeeper to take all resources out of service at the same time instead of taking one component out of service, bringing that component in service, then taking the next component out of service and then bringing it in service.

The terminal resource is created through the **Create Resource Hierarchy** option. This brings up the ***Create Resource Wizard***, where you will select **Generic Application** from the **Recovery Kit** list.

For further information on creating the terminal resource, refer to the Creating a Generic Application Resource Hierarchy section of the SPS for Linux Technical Documentation at http://docs.us.sios.com/ under **LifeKeeper > Administration > Administrator Tasks > Creating Resource Hierarchies > Creating a Generic Application Resource Hierarchy.**

6.7.5.1. MD Device Number

If/when configuring an MD device on a node in a cluster, use a unique MD number within the cluster, even if the MD device will not be used with or controlled by LifeKeeper.

6.7.5.2. All MD Devices In-Service

When creating a NetRAID resource in a cluster, all MD devices configured in the cluster should be in-service on the node where the NetRAID device is configured. This will enable NetRAID to use an MD number that will not conflict with any existing MD devices. If this is not done, then the MD kit will reorder the numbers used for the MD resources that have a conflict on the next in-service operation.

6.7.6. MD Troubleshooting

Error Messages

This section provides a list of messages that may be encountered with the use of the SPS Software RAID Recovery Kit. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Because the Software RAID Recovery Kit relies on other SPS components to drive the creation and extension of hierarchies, messages from these other components are also possible. In these cases, please refer to the Message Catalog(located on our Technical Documentation site under “Search for an Error Code”) which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

Software RAID Recovery Kit Error Messages

Error Number	Error Message
117000	<resource type> resource type is not installed on <LifeKeeper server name> Action: Install the MD Recovery Kit on the identified system
117001	This script must be executed on <LifeKeeper server name>
117002	Failed to create <device name> hierarchy
117003	Failed to create dependency <resource tag>-<resource tag> on machine <LifeKeeper server name>
117004	LifeKeeper internal ID <resource ID> already in use
117005	<resource type> constructor requires a valid argument

6.8. WebSphere MQ Recovery Kit Administration Guide

The SIOS Protection Suite for Linux WebSphere MQ Recovery Kit provides fault resilient protection for WebSphere MQ queue managers and queue manager storage locations. This kit enables a failure on a primary WebSphere MQ server or queue manager to be recovered on the primary server or a designated backup server without significant lost time or human intervention.

Document Contents

This guide contains the following topics:

- SIOS Protection Suite Documentation. Provides a list of SPS for Linux documentation and where to find it.
- Abbreviations. Contains a list of abbreviations that are used throughout this document along with their meaning.
- Requirements. Describes the hardware and software necessary to properly set up, install and operate the WebSphere MQ Recovery Kit. Refer to the SIOS Protection Suite Installation Guide for specific instructions on how to install or remove SPS for Linux software.
- WebSphere MQ Recovery Kit Overview. Provides a brief description of the WebSphere MQ Recovery Kit's features and functionality as well as lists the versions of the WebSphere MQ software supported by this Recovery Kit.
- WebSphere MQ Configuration Considerations. Provides a general description of configuration issues and shows file system layouts supported by the WebSphere MQ Recovery Kit.
- Configuring WebSphere MQ for Use with LifeKeeper. Provides a step-by-step guide of how to install and configure WebSphere MQ for use with LifeKeeper.
- Configuration Changes Post Resource Creation. Provides information on how WebSphere MQ configuration changes affect LifeKeeper WebSphere MQ resource hierarchies.
- WebSphere MQ Configuration Examples. Provides examples of typical WebSphere MQ configurations and the steps to configure your WebSphere MQ resources.
- LifeKeeper Configuration Tasks Describes the tasks for creating and managing your WebSphere MQ resource hierarchies using the LifeKeeper GUI.
- WebSphere MQ Troubleshooting. Provides a list of informational and error messages with recommended solutions.
- Appendices. Provide sample configuration files for WebSphere MQ and a configuration sheet that

can be used to plan your WebSphere MQ installation.

Reference Documents

MQ documentation is located at the WebSphere MQ Library available at:

http://www.ibm.com/software/integration/wmq/library/

SPS Documentation

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- SIOS Protection Suite Installation Guide
- Optional Recovery Kit Documentation
- SPS for Linux IP Recovery Kit Administration Guide

This documentation, along with documentation associated with other SPS Recovery Kits, is available online at:

http://docs.us.sios.com/

6.8.1. MQ Recovery Kit Abbreviations

The following abbreviations are used throughout this document:

Abbreviation	Meaning
HA	Highly Available, High Availability
QMDIR	WebSphere MQ queue manager directory. This directory holds the queue manager persistent queue data and is typically located in */var/mqm/qmgrs* with the name of the queue manager as subdirectory name. The exact location of this directory is specified in the global *mqs.ini* configuration file. If the DataPath parameter is defined then the DataPath value along with queue manager name specifies the location of the queue manager persistent data, otherwise the default location as noted above is used.
QMLOGDIR	WebSphere MQ queue manager log directory. This directory holds the queue manager log data and is typically located in */var/mqm/log* with the queue manager name as subdirectory. The exact location of this directory is specified in the queue manager configuration file *(QMDIR/qm.ini)*.
MQUSER	The operating system user running all WebSphere MQ commands. This user is the owner of the QMDIR. The user must be a member of the MQGROUP administrative group mqm (see below).
MQGROUP	The operating system user group that the MQUSER must be part of. This group must be named mqm.
UID	Numeric user id of an operating system user.
GID	Numeric group id of an operating system user group.

6.8.2. MQ Recovery Kit Requirements

Your SPS configuration must meet the following requirements prior to the installation of the WebSphere MQ Recovery Kit. Please see the SIOS Protection Suite Installation Guide for specific instructions regarding the installation, removal and configuration of your SPS hardware and software.

6.8.2.1. MQ Hardware and Software Requirements

Hardware Requirements

- **Servers**. The Recovery Kit requires two or more servers configured in accordance with the requirements described in the SIOS Protection Suite Installation Guide. See the Linux Configuration Table for supported Linux distributions.

- **Data Storage**. The WebSphere MQ Recovery Kit can be used in conjunction both with shared storage and with replicated storage provided by the DataKeeper product. It can also be used with network-attached storage (NAS).

Software Requirements

- **SPS Software**. You must install the same version of **SPS** software and any patches on each server.

- **LifeKeeper WebSphere MQ Recovery Kit**. Version 7.5.1 or later of the WebSphere MQ Recovery Kit is required for systems running WebSphere MQ v7.1 or later.

- **LifeKeeper IP Recovery Kit**. You must have the same version of the LifeKeeper IP Recovery Kit on each server.

- **IP Network Interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

  Note: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation and so forth.

- **TCP/IP Software**. Each server also requires the TCP/IP software.

- **WebSphere MQ Software**. IBM WebSphere MQ must be ordered separately from IBM. See the SPS Release Notes for supported WebSphere MQ versions. The WebSphere MQ Software must be installed on each server of the cluster prior to installing the WebSphere MQ Recovery Kit. The following WebSphere MQ packages must be installed to successfully install the WebSphere MQ Recovery Kit:

  MQSeriesServer, MQSeriesSamples, MQSeriesClient, MQSeriesRuntime, MQSeriesSDK

  Beginning with IBM WebSphere MQ Version 7.0.1 Fix Pack 6, a new feature was introduced allowing multiple versions of WebSphere MQ to be installed and run on the same server (e.g. MQ Versions 7.0.1 Fix Pack 6 and 7.1). This feature, known as multi-instance support, is now

supported starting with version 9.0.1 of the WebSphere MQ Recovery Kit. Protecting multiple queue managers within a single IBM WebSphere MQ installation version, protection of queue managers from multiple IBM WebSphere MQ installation versions, as well as the use the DataPath parameter in the `mqs.ini` file introduced as part of the multi-instance feature are all now supported in this version of the recovery kit.

- **Optional C Compiler**. The WebSphere MQ Recovery Kit contains a modified `amqsget0.c` sample program from the WebSphere MQ samples package. This program has been modified to work with a timeout of 0 seconds instead of the default 15 seconds. It is used to perform `PUT/GET` tests for the queue manager. This program is compiled during RPM installation and therefore a C compiler must be installed and must be located in the PATH of the “root” user.

- **Syslog.pm.** If you want to use syslog logging for WebSphere MQ resources, the Syslog.pm PERL module must be installed. This module is part of the standard PERL distribution and is not required to be installed separately.

6.8.2.2. Upgrading an MQ LifeKeeper Cluster

1. Upgrade SPS on all nodes in the cluster including the WebSphere MQ Recovery Kit following the instructions documented in the Upgrading SPS section of the SIOS Protection Suite Installation Guide.

2. Upgrade IBM WebSphere MQ software on each node in the cluster using the following steps:

   a. If one or more LifeKeeper IBM WebSphere MQ resource hierarchies are in service on the node being upgraded, they must be taken out of service before the upgrade of the IBM WebSphere MQ software can be done. This can be done by switching over to the standby node.

   b. Follow the IBM WebSphere upgrade instructions.

3. Once the IBM WebSphere software has been installed on the node, bring the LifeKeeper IBM WebSphere MQ resource hierarchies in service (restore) and verify the operation of each Queue Manager.

4. Once the operation of each Queue Manager is confirmed, upgrade all the other nodes in the cluster.

6.8.3. WebSphere MQ Recovery Kit Overview

WebSphere MQ (formerly known as MQSeries) is an IBM software product that provides reliable and guaranteed one time only delivery of messages. The core element of WebSphere MQ is the queue manager which handles one or more queues that are used to send (put) and receive (get) messages. Once a message is put into a queue, it is guaranteed that this message is persistent and will be delivered only once.

The WebSphere MQ Recovery Kit enables LifeKeeper to protect WebSphere MQ queue managers including the command server, the listener and the persistent queue manager data. Protection of the queue manager listener can be optionally disabled on a per queue manager basis to support configurations that do not handle client connects or to enable the administrator to shut down the listener without causing a LifeKeeper recovery attempt.

The WebSphere MQ Recovery Kit provides a mechanism to recover protected WebSphere MQ queue managers from a failed primary server onto a backup server. LifeKeeper can detect failures either at the server level (via a heartbeat) or resource level (by monitoring the WebSphere MQ daemons) so that control of the protected WebSphere MQ services are transferred to a backup server.

The WebSphere MQ Recovery Kit also supports multiple installations of WebSphere MQ to be installed and run on the same system. With multi-version MQ support a Queue Manager from MQ software version 7.x and a Queue Manager from MQ software version 8.x can both be protected by the Recovery Kit. Prior to the addition of this support in version 9.0.2 of the WebSphere MQ Recovery Kit only 1 version of the MQ software could be installed and running on the system (NOTE: the installation required it be installed in the default location – /opt/mqm).

6.8.3.1. MQ Recovery Kit Resource Hierarchies

A typical WebSphere MQ hierarchy will be comprised of a WebSphere MQ queue manager resource. It also contains one or more file system resources, depending on the file system layout and zero or more IP resources. The exact makeup of the hierarchy depends on what is being protected. If the administrator chooses to include an IP resource in the WebSphere MQ resource hierarchy, that IP must be created prior to creating the WebSphere MQ queue manager resource and that IP resource must be active on the primary server. The file system hierarchies are created automatically during the creation of the WebSphere MQ queue manager resource.

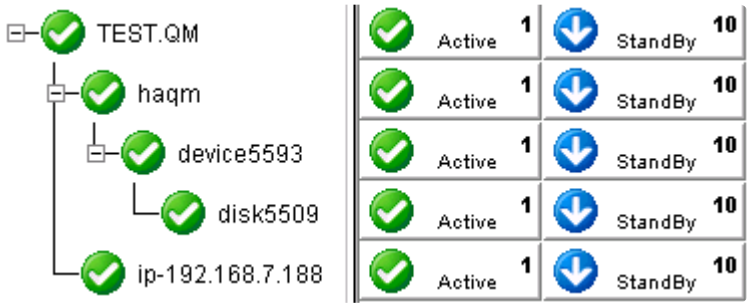


Figure 1 Typical WebSphere MQ hierarchy – symbolic links

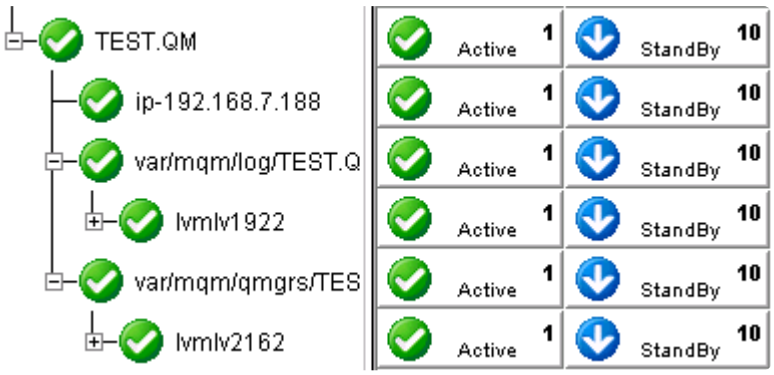


Figure 2 Typical WebSphere MQ hierarchy – LVM configuration

6.8.3.2. MQ Recovery Kit Features

The WebSphere MQ Recovery Kit provides the following features:

- Supports mult-instance Queue Managers (queue managers created with multiple versions of MQ software)
- Supports Active/Active configurations
- Supports LINEAR and CIRCULAR logging (detected automatically)
- Supports end to end application health check via server connect and client connect
- Supports optional PUT/GET tests (with definable test queue via GUI and command line)
- Supports customizable logging levels
- Supports all LifeKeeper supported storage types
- Supports optional listener protection (default: enabled)
- Supports additional syslog message logging (log facility local7)
- Supports multiple levels of Command Server protection (default: full)

6.8.4. WebSphere MQ Configuration Considerations

This section contains information that should be considered before beginning to configure WebSphere MQ. It also contains a step-by-step process for configuring and protecting a WebSphere MQ queue manager with LifeKeeper.

For instructions on installing WebSphere MQ on Linux distributions supported by SPS, please see WebSphere MQ documentation.

6.8.4.1. MQ Configuration Requirements

The section Configuring WebSphere MQ for Use with LifeKeeper contains a process for protecting a queue manager with LifeKeeper. In general, the following requirements must be met to successfully configure a WebSphere MQ queue manager with LifeKeeper:

1. **Configure Kernel Parameters**. Please refer to the WebSphere MQ documentation for information on how Linux kernel parameters such as shared memory and other kernel resources should be configured.

2. **MQUSER** and **MQGROUP**. The MQGROUP and the MQUSER must exist on all servers in the LifeKeeper cluster. Websphere MQ software requires that the MQGROUP mqm exist and that it also have the MQUSER mqm defined that has its primary group membership set to the MQGROUP mqm. If the mqm user and mqm group do not exist at the time the Websphere MQ software is installed they will be automatically created. When installing the WebSphere MQ software most of the files and directories will have their user and group ownership set to the mqm user and mqm group. User and group ownership of the files and directories in the Queue Manager data and log directories will also be set to the mqm user and mqm group. Additionally, when a Queue Manager is started it will run as the mqm user. Therefore, the MQUSER user id (uid) and the MQGROUP group id (gid) must be the same on all servers in the cluster. The MQ Recovery Kit will verify this when attempting to extend the resource. If they do not match the resource extension will fail. Note: If you are using NIS, LDAP or another authentication tool besides the local password and group files you need to set up the MQUSER and MQGROUP prior to the installation of the Websphere MQ and LifeKeeper software. You may also need to create a home directory. If this is an upgrade from a prior release of the WebSphere MQ Recover Kit, then the MQUSER PATH environment variable setting may need to be modified. In prior releases of the Recovery Kit the MQUSER PATH environment variable needed to be modified to include the default install location of the WebSphere MQ software, /opt/mqm. If that change was made in a prior release it must be unset for this version of the Recovery Kit to function correctly.

3. **Alternate MQ user support**. Although the Websphere MQ software will always run as the mqm user an alternate user name can be specified for running all MQ commands provided the alternate user has primary or secondary membership in the mqm group. An alternate user name for starting WebSphere MQ may be required when integrating with other MQ Tools. To change to an alternate user see the MQS_ALT_USER_NAME tunable in the “Changing LifeKeeper WebSphere MQ Recovery Kit Defaults” section of this document.

4. **Manual command server startup**. If you want to have LifeKeeper start the command server, disable the automatic command server startup using the following command on the primary server. Otherwise, the startup of the command server will be performed automatically when the Queue Manager is started:

```
runmqsc QUEUE.MANAGER.NAME

ALTER QMGR SCMDSERV(MANUAL)
```

5. **`QMDIR` and `QMLOGDIR` must be located on shared storage.** The queue manager directory QMDIR and the queue manager log directory *`QMLOGDIR`* must be located on LifeKeeper-supported shared storage to let the WebSphere MQ on the backup server access the data. See Supported File System Layouts for further details.

6. **`QMDIR` and `QMLOGDIR` permissions**. The *`QMDIR`* and *`QMLOGDIR`* directories must be owned by `MQUSER` and the group `MQGROUP`. The ARK dynamically determines the `MQUSER` by looking at the owner of this directory. It also detects symbolic links and follows them to the final targets. Use the system command `chown` to change the owner of these directories if required.

7. **Disable Automatic Startup of Queue Manager**. If you are using an `init` script to start and stop WebSphere MQ, disable it for the queue manager(s) protected by LifeKeeper. To disable the `init` script, use the operating system provided functions like `insserv` on SuSE or `chkconfig` on Red Hat.

8. **Server Connection Channel Authorization**. Beginning with WebSphere MQ version 7.1 changes were made to channel authorization. By default the MQADMIN user (mqm) is unable to authenticate anonymously (no password) thus failing the resource hierarchy create (authorization for queue managers created with a WebSphere MQ release prior to 7.1 should continue to work). Starting with WebSphere MQ 7.1 one method to allow authorization for the MQADMIN user is to disable channel authorization. For WebSphere MQ 8.0 additional changes are required to the authinfo for system.default.authinfo.idpwos (in runmqsc run 'display authinfo(system.default.authinfo.idpwos)' to retrieve the current settings). The chckclnt setting of 'reqdamd' must altered and set to 'optional'. Failure to allow the MQADMIN user anonymous authorization will result in the following error: 'MQCONNX ended with reason code 2035' during resource creation. See the WebSphere MQ documentation for details on how to create channels.

9. **`MQSeriesSamples, MQSeriesSDK and MQSeriesClient` Package**. LifeKeeper uses a client connection to WebSphere MQ to verify that the listener and the channel initiator are fully functional. This is a requirement for remote queue managers and clients to connect to the queue manager. Therefore, the `MQSeriesClient` package must be installed on all LifeKeeper cluster nodes running WebSphere MQ. Also, the `MQSeriesSDK` and `MQSeriesSamples` packages must be installed to perform client connect tests and `PUT/GET` tests.

10. **Optional C Compiler**. For the optional `PUT/GET` tests to take place, a C compiler must be installed on the machine. If not, a warning is issued during the installation.

11. **LifeKeeper Test Queue**. The WebSphere MQ Recovery Kit optionally performs a `PUT/GET` test to verify queue manager operation. A dedicated test queue has to be created because the recovery kit retrieves all messages from this queue and discards them. This queue should have set the default persistency setting to "**yes**" `(DEFPSIST=yes)` When you protect a queue manager in LifeKeeper, a test queue named "LIFEKEEPER.TESTQUEUE" will be automatically created. You can also use the following command to create the test queue manually before protecting the queue manager:

```
su – MQUSER
runmqsc QUEUE.MANAGER.NAME

define qlocal(LIFEKEEPER.TESTQUEUE) DEFPSIST(YES) DESCR('LifeKeeper
test queue')
```

Note: If you want to use a name for the LifeKeeper test queue other than the default "LIFEKEEPER.TESTQUEUE", the name of this test queue must be configured. See Editing Configuration Resource Properties for details.

12. **TCP Port for Listener Object**. Alter the Listener object via runmqsc to reflect the TCP port in use. Use the following command to change the TCP port of the default Listener:

    su – MQUSER
    runmqsc QUEUE.MANAGER.NAME

    alter `LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1414) IPADDR(192.168.1.100)`

    Note: The listener object must be altered even if using the default MQ listener TCP port 1414, but it is not necessary to set a specific IP address `(IPADDR)`. If you skip the `IPADDR` setting, the listener will bind to all interfaces on the server. If you do set `IPADDR`, it is strongly recommended that a virtual IP resource be created in LifeKeeper using the `IPADDR` defined address. This ensures the IP address is available when the MQ listener is started.

13. **TCP Port Number**. Each WebSphere MQ listener must use a different port (default 1414) or bind to a different virtual IP with no listener binding to all interfaces. This includes protected and unprotected queue managers within the cluster.

14. **Queue Manager configured in *mqs.ini***. In Active/Active configurations, each server holds its own copy of the global queue manager configuration file *mqs.ini*. In order to run the protected queue manager on all servers in the cluster, the queue manager must be configured in the *mqs.ini* configuration file of all servers in the cluster. Copy the appropriate QueueManager: stanza from the primary server and add it to the *mqs.ini* configuration files on all backup servers.

6.8.4.1.1. MQ Supported File System Layouts

Depending on your shared storage system and the file system layout, there are three different supported configurations. They differ in the file system layout. The following section describes the supported file system layouts.

- Configuration 1 – /var/mqm on Shared Storage
- Configuration 2 – Direct Mounts
- Configuration 3 – Symbolic Links

6.8.4.1.1.1. Configuration 1 – /var/mqm on Shared Storage

In this configuration, the whole */var/mqm* directory is mounted on LifeKeeper supported shared storage (SCSI, SAN, NAS or replicated).

Note: This only works for Active/Passive configurations.

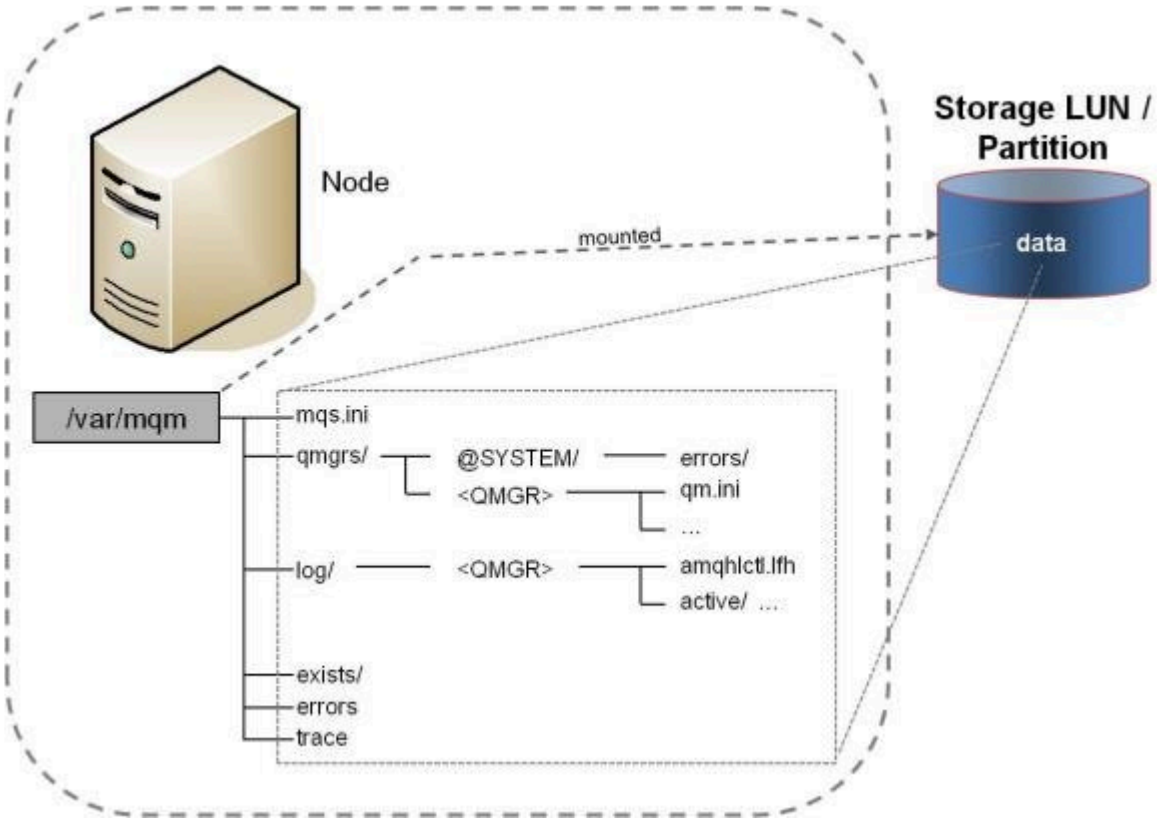


Figure 3 – File System Layout 1 – */var/mqm* **on Shared Storage**

6.8.4.1.1.2. Configuration 2 – Direct Mounts

In this configuration, the *QMDIR* and the *QMLOGDIR* directories are located on shared storage. This requires two dedicated LUNS or partitions or the use of LVM for each queue manager. If LVM is used, two logical volumes from the same LUN can be created and separately mounted on the two directories.

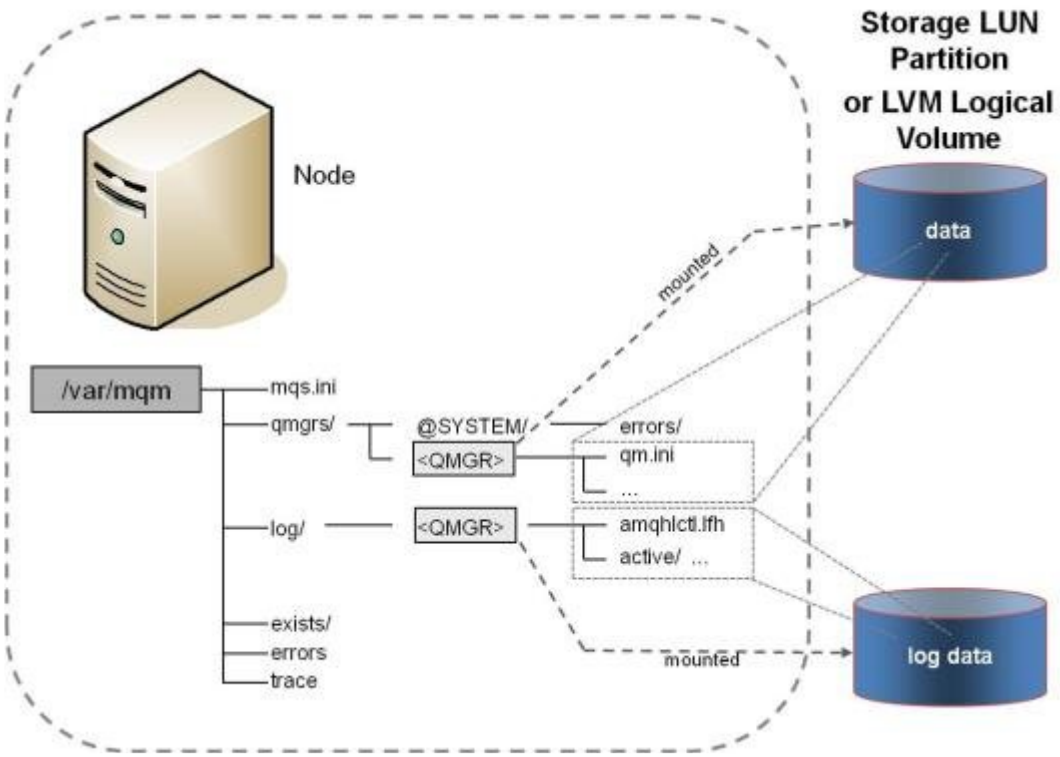


Figure 4 – File System Layout 2 – Direct Mounts

6.8.4.1.1.3. Configuration 3 – Symbolic Links

The recommended configuration for Active/Active configurations without LVM and with a large number of queue managers is the use of symbolic links. In this case, one or more dedicated mount points are created (e.g. `/mq`). A LifeKeeper protected file system is mounted there and subdirectories for each queue manager are created (e.g. `/mq/QUEUE!MANAGER!NAME/log` and `/mq/QUEUE!MANAGER!NAME/qmgrs`). The `QMDIR` and `QMLOGDIR` directories are then linked to this location.

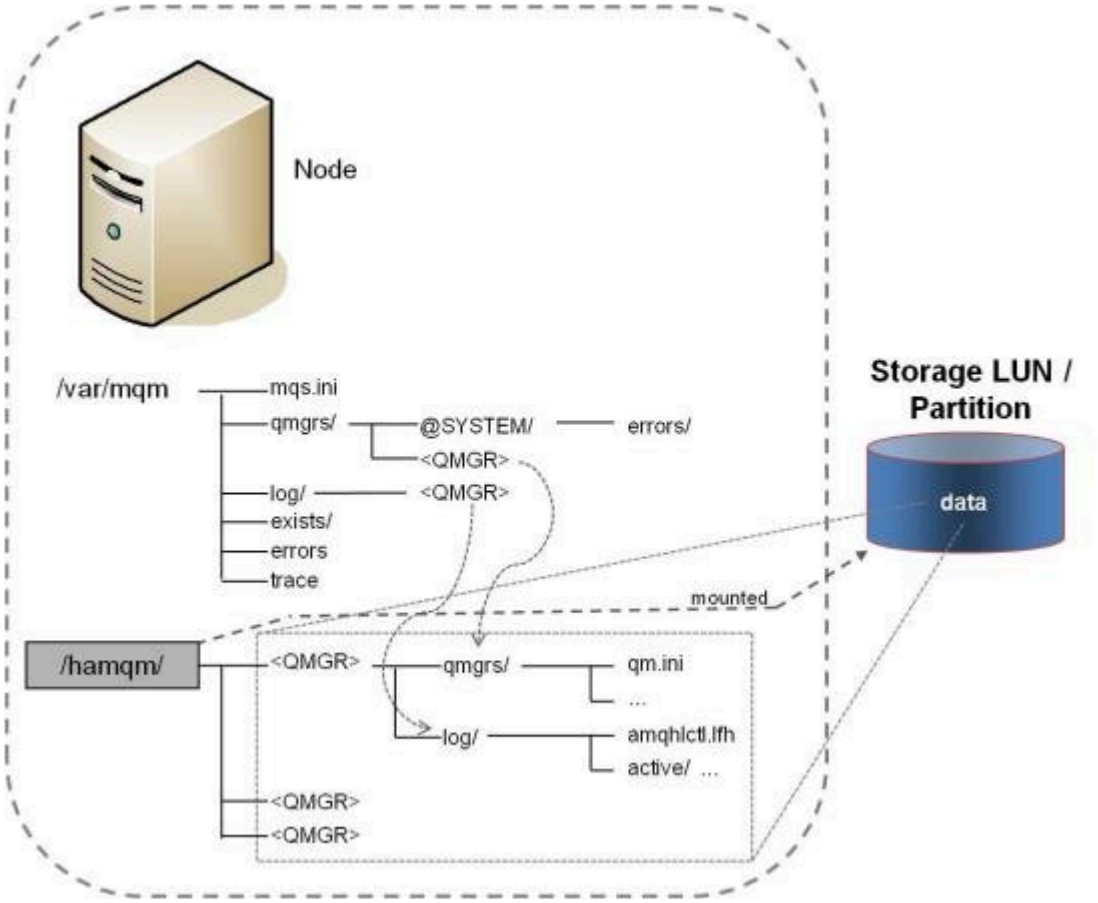


Figure 5 – File System Layout 3 – Symbolic Links

6.8.4.1.2. Configuring WebSphere MQ for use with LifeKeeper

There are a number of WebSphere MQ configuration considerations that need to be made before attempting to create LifeKeeper for Linux WebSphere MQ resource hierarchies. These changes are required to enable the Recovery Kit to perform **PUT/GET** tests and to make the path to WebSphere MQ persistent data highly available. If the WebSphere MQ queue manager handles remote client requests via TCP/IP, a virtual IP resource must be created prior to creating the WebSphere MQ resource hierarchy. Perform the following actions to enable LifeKeeper WebSphere MQ resource creation:

1. Plan your installation (see Appendix C).

   Before installing WebSphere MQ, you must plan your installation. This includes choosing an **MQUSER, MQUSER UID** and **MQGROUP GID**. You must also decide which file system layout you want to use (see Supported File System Layouts). To ease this process, SIOS Technology Corp. provides a form that contains fields for all required information. See Appendix C – WebSphere MQ Configuration Sheet. Fill out this form to be prepared for the installation process.

2. Configure Kernel Parameters on each server.

   WebSphere MQ may require special Linux kernel parameter settings like shared memory. See the WebSphere MQ documentation for your release of WebSphere MQ for the minimum requirements to run WebSphere MQ. To make kernel parameter changes persistent across reboots, you can use the */etc/sysctl.conf* configuration file. It may be necessary to add the command `sysctl -p` to your startup scripts (`boot.local`). On SuSE, you can run `insserv boot.sysctl` to enable the automatic setting of the parameters in the *sysctl.conf* file.

3. Create the `MQUSER` and `MQGROUP` on each server.

   Use the operating system commands `groupadd` and `adduser` to create the MQUSER and MQGROUP with the UID and GID from the “WebSphere MQ Configuration Sheet” you used in Step 1.

   If the `MQUSER` you have chosen is named *mqm* and has `UID 1002` and the `MQGROUP GID` is `1000`, you can run the following command on each server of the cluster (change the `MQUSER`, `UID` and `GID` values to reflect your settings):

   ```
   groupadd -g 1000 mqm
   useradd -m -u 1002 -g mqm mqm
   ```

   Note: These settings must be same on all nodes in the cluster. If you are running `NIS` or `LDAP`, create the user and group only once. You may need to create home directories if you have no central home directory server.

4. Unconfigure the PATH environment variable (upgrade only).

   If this is an upgrade from a prior release of the WebSphere MQ Recover Kit, then the MQUSER PATH environment variable setting may need to be modified. In prior releases of the Recovery Kit the MQUSER PATH environment variable needed to be modified to include the default install location of the WebSphere MQ software, /opt/mqm. If that change was made in a prior release it must be unset for this version of the Recovery Kit to function correctly.

5. Install required packages to install WebSphere MQ on each server.

   MQSeries installation requires the installation of X11 libraries and Java for license activation (`mqlicense_lnx.sh`). Install the required software packages.

6. Install WebSphere MQ software and WebSphere MQ fix packs on each server.

   Follow the steps described in the "WebSphere MQ documentation" for your release of WebSphere MQ.

7. **Server Connection Channel Authorization**. Beginning with WebSphere MQ version 7.1 changes were made to channel authorization. By default the MQADMIN user (mqm) is unable to authenticate anonymously (no password) thus failing the resource hierarchy create (authorization for queue managers created with a WebSphere MQ release prior to 7.1 should continue to work). Starting with WebSphere MQ 7.1 one method to allow authorization for the MQADMIN user is to disable channel authorization. For WebSphere MQ 8.0 additional changes are required to the authinfo for system.default.authinfo.idpwos (in runmqsc run 'display authinfo(system.default.authinfo.idpwos)' to retrieve the current settings). The chckclnt setting of 'reqdamd' must altered and set to 'optional'. Failure to allow the MQADMIN user anonymous authorization will result in the following error: 'MQCONNX ended with reason code 2035' during resource creation. See the WebSphere MQ documentation for details on how to authorize channels and set access permission.

8. If MQ Version 7.1 or later is being used, enable the MQADMIN user for the specified channel within MQ for the Queue Manager being used.

9. Install LifeKeeper and the WebSphere MQ Recovery Kit on each server.

   See the SIOS Protection Suite Installation Guide for details on how to install SPS.

10. Prepare the shared storage and mount the shared storage.

    See section Supported File System Layouts for file system layouts supported. Depending on the file system layout and the storage type, this involves creating volume groups, logical volumes, creating file systems or mounting NFS shares.

    Here is an example of file system layout 2 with NAS storage:

```
node1:/var/mqm/qmgrs # mkdir TEST\!QM

node1:/var/mqm/qmgrs # mkdir ../log/TEST\!QM

node1:/var/mqm/qmgrs # mount 192.168.1.30:/raid5/vmware/shared_NFS/
TEST.QM/qmgrs ./TEST\!QM/

node1:/var/mqm/qmgrs # mount 192.168.1.30:/raid5/vmware/shared_NFS/
TEST.QM/log ../log/TEST\!QM/
```

11. Set the owner and group of *QMDIR* and *QMLOGDIR* to MQUSER and **MQGROUP**.

    The *QMDIR* and *QMLOGDIR* must be owned by MQUSER and MQGROUP. Use the following commands to set the file system rights accordingly:

    ```
    chown MQUSER QMDIR
    chgrp mqm QMDIR
    chown MQUSER QMLOGDIR
    chgrp mqm QMLOGDIR
    ```

    The values of MQUSER, QMDIR and QMLOGDIR depend on your file system layout and the user name of your **MQUSER**. Use the sheet from Step 1 to determine the correct values for the fields.

    Here is an example for MQUSER mqm and queue managerTEST.QM with default QMDIR and QMLOGDIR destinations:

    ```
    node1:/var/mqm/qmgrs # chown mqm TEST\!QM/
    node1:/var/mqm/qmgrs # chgrp mqm TEST\!QM/
    node1:/var/mqm/qmgrs # chown mqm ../log/TEST\!QM/
    node1:/var/mqm/qmgrs # chgrp mqm ../log/TEST\!QM/
    ```

12. Create the queue manager on the primary server.

    Follow the steps described in the WebSphere MQ documentation for how to create a queue manager for the version(s) of the WebSphere MQ software being used..

    Here is an example for MQUSER mqm and queue manager TEST.QM.

    ```
    node1:/var/mqm/qmgrs # su - mqm
    mqm@node1:~> crtmqm TEST.QM
    WebSphere MQ queue manager created.
    Creating or replacing default objects for TEST.QM.
    Default objects statistics : 31 created. 0 replaced. 0 failed.
    Completing setup.
    Setup completed.
    ```

Note: If you want to protect an already existing queue manager, use the following steps to move the queue manager data to the shared storage:

a. Stop the queue manager (`endmqm -i QUEUE.MGR.NAME`).

b. Copy the content of the queue manager directory and the queue manager log directory to the shared storage created in Step 10.

c. Change the global configuration file (*`mqs.ini`*) and queue manager configuration file (*`qm.ini`*) as required to reflect the new location of the `QMDIR` and the `QMLOGDIR`.

d. Start the queue manager to verify its function (`strmqm QUEUE.MGR.NAME`).

e. Stop the queue manager (`endmqm -i QUEUE.MGR.NAME`).

13. **Optional**: Configure a virtual IP resource in LifeKeeper on the primary server.

    Follow the steps and guidelines described in the SPS for Linux IP Recovery Kit Administration Guide and the SIOS Protection Suite Installation Guide.

    Note: If your queue manager is only accessed by server connects, you do not have to configure the LifeKeeper virtual IP.

14. Modify the listener object to reflect your TCP IP address and port:

```
su - MQUSER
runmqsc QUEUE.MANAGER.NAME

alter LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1414)
IPADDR(192.168.1.100)
```

    Note: Use the same IP address used in the Step 13 to set the value for `IPADDR`. Do not set `IPADDR` to have WebSphere MQ bind to all addresses.

15. Start the queue manager on the primary server.

    On the primary server, start the queue manager, the command server if it is configured to be started manually and the listener:

```
su - MQUSER
strmqm QUEUE.MANAGER.NAME
strmqcsv QUEUE.MANAGER.NAME
runmqlsr -m QUEUE.MANAGER.NAME -t TCP &
```

16. Verify that the queue manager has been started successfully:

```
su – MQUSER
echo 'display qlocal()*' | runmqsc QUEUE.MANAGER.NAME
```

17. Add the queue manager stanza to the global queue manager configuration file *mqs.ini* on the backup server.

    Note: This step is required for file system layouts 2 and 3.

18. **Optional**: Create the LifeKeeper test queue on the primary server.

```
runmqsc TEST.QM

5724-B41 © Copyright IBM Corp. 1994, 2002. ALL RIGHTS RESERVED.

Starting MQSC for queue manager TEST.QM.

define qlocal(LIFEKEEPER.TESTQUEUE) defpsist(yes) descr('LifeKeeper
test queue')

     1 : define qlocal(LIFEKEEPER.TESTQUEUE) defpsist(yes)
     descr('LifeKeeper test queue')

AMQ8006: WebSphere MQ queue created.
```

19. If you want to have LifeKeeper start the command server, disable the automatic command server startup using the following command on the primary server. Otherwise, the startup of the command server will be performed automatically when the Queue Manager is started:

```
su – MQUSER
runmqsc TEST.QM
ALTER QMGR SCMDSERV(MANUAL)
```

20. Create queue manager resource hierarchy on the primary server.

    See section LifeKeeper Configuration Tasks for details.

21. Extend queue manager resource hierarchy to the backup system.

    See section LifeKeeper Configuration Tasks for details.

22. Test your configuration.

    To test your HA WebSphere MQ installation, follow the steps described in Testing a WebSphere MQ Resource Hierarchy.

6.8.4.1.3. MQ Configuration Changes After Resource Creation

The SPS WebSphere MQ Recovery Kit uses WebSphere MQ commands to start and stop the queue manager. Some exceptions to this rule follow.

6.8.4.1.3.1. Relocating QMDIR and QMLOGDIR

If the location of the *QMDIR* and *QMLOGDIR* are changed, the LifeKeeper configuration must be modified. You have the following options to do so:

1. Recreate the queue manager resource hierarchies.

   This involves deletion of the queue manager hierarchy and creation of the queue manager hierarchy. See sections Deleting a WebSphere MQ Hierarchy and Creating a WebSphere MQ Resource Hierarchy for details.

2. Create the new file system hierarchies manually and add the new file system hierarchies to the WebSphere MQ hierarchy. Remove the old file system hierarchies from the WebSphere MQ hierarchy and remove the old file system hierarchies. See the SIOS Protection Suite Installation Guide for details on how to create and remove file system hierarchies.

6.8.4.1.3.2. Changing the Listener Port

To change the listener port of a queue manager, follow these steps:

Alter the listener object in `runmqsc` then stop and start the listener:

```
su - MQUSER
runmqsc QUEUE.MANAGER.NAME

alter LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1415)
stop LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
start LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
```

See the section Editing Configuration Resource Properties for details.

6.8.4.1.3.3. Changing the IP for the Queue Manager

To change the LifeKeeper protected IP associated with the WebSphere MQ queue manager, follow these steps:

1. Create a new LifeKeeper virtual IP in the LifeKeeper GUI.
2. Add the new virtual IP to the WebSphere MQ hierarchy.
3. Remove the old virtual IP from the WebSphere MQ hierarchy.
4. Delete the old virtual IP resource.
5. If needed, modify your listener object in runmqsc and restart the listener:

```
su - MQUSER
runmqsc QUEUE.MANAGER.NAME

alter LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1414)
(IPADDR192.168.1.101)
stop LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
start LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
```

As an alternative, you can use the LifeKeeper `lk_chg_value` facility to change the IP. See the `lk_chg_value(8)` man page for details.

6.8.4.1.4. WebSphere MQ Configuration Examples

This section contains definitions and examples of typical WebSphere MQ configurations. Each example includes the configuration file entries that apply to LifeKeeper.

6.8.4.1.4.1. Active/Standby Configuration with /var/mqm on Shared Storage

In the Active/Standby configuration, Node1 is the primary LifeKeeper server. It protects the WebSphere MQ queue managers. All storage resides on a shared array between the cluster servers. While Node2 may be handling other applications/services, it acts only as a backup for the WebSphere MQ resources in LifeKeeper's context. The directory `/var/mqm` is located on shared storage. The primary server can run as many queue managers as it can handle.

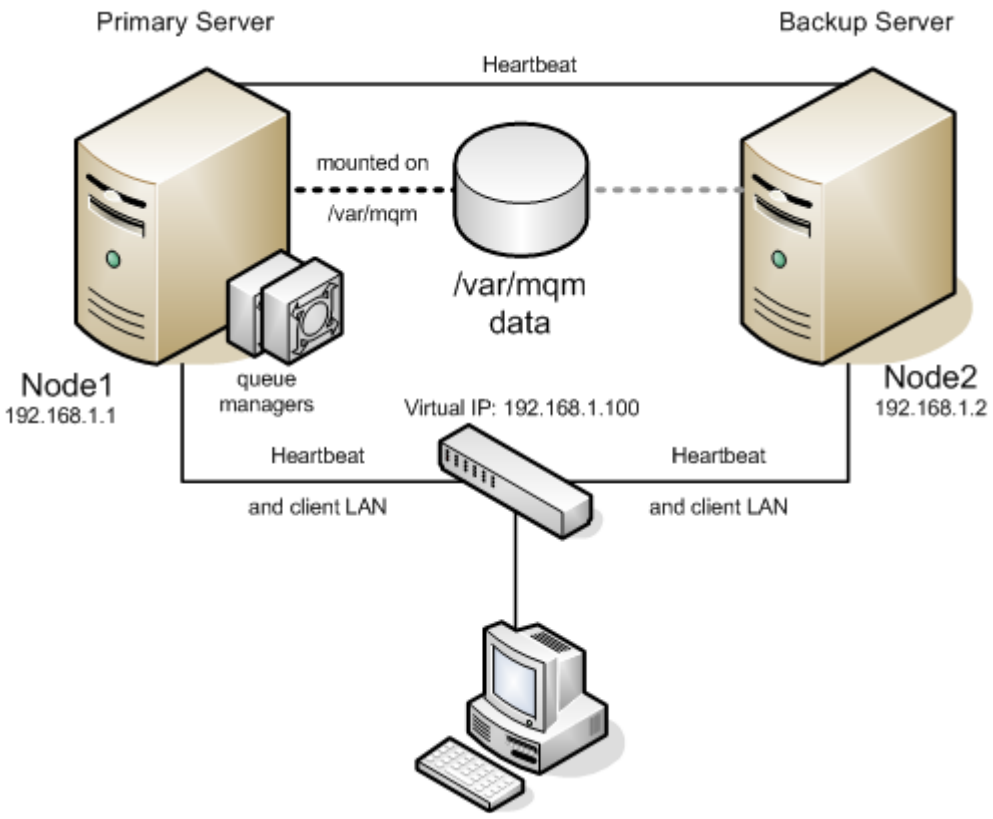


Figure 7 Active/Standby Configuration with Local Storage

Configuration Notes

- The clients connect to the WebSphere MQ servers using the LifeKeeper protected IP 192.168.1.100 designated to float between the servers in the cluster.
- The directory `/var/mqm` is located on shared storage.
- Each queue manager has modified the listener object to contain a unique port number.

6.8.4.1.4.2. Active/Standby Configuration with NAS Storage

In the Active/Standby configuration, Node1 is the primary LifeKeeper server. It protects the WebSphere MQ queue managers. All storage resides on a NAS server with the IP 10.0.0.100. While Node2 may be handling other applications/services, it acts only as a backup for the WebSphere MQ resources in LifeKeeper's context. The directory `/var/mqm` is located from the NAS server's IP 10.0.0.100 and mounted on the active node only. The primary server can run as many queue managers as it can handle.

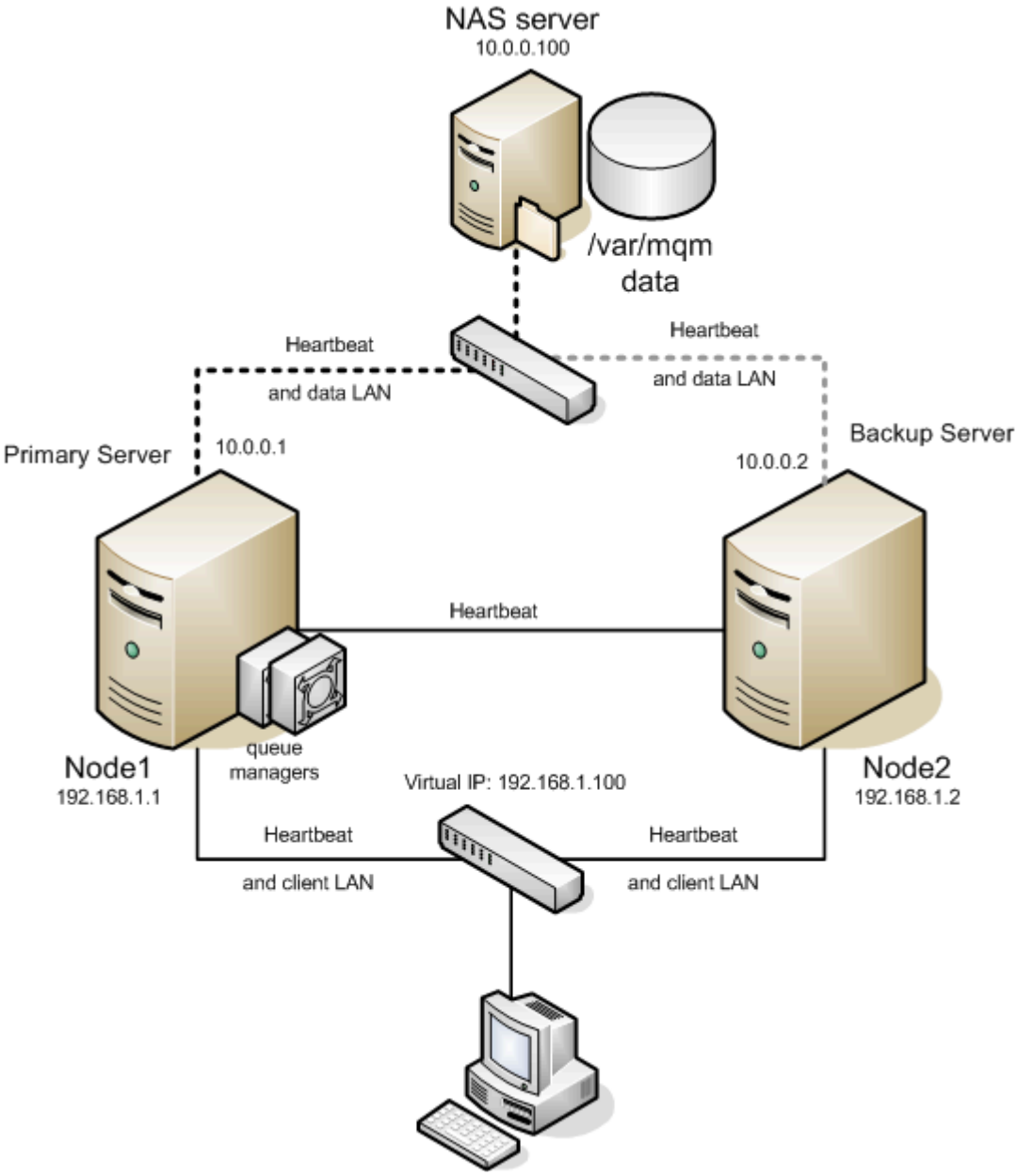


Figure 8 Active/Standby Configuration with NFS Storage

Configuration Notes

- The clients connect to the WebSphere MQ servers using the LifeKeeper protected IP 192.168.1.100 designated to float between the servers in the cluster.

- The directory */var/mqm* is located on the NAS server.
- The active server mounts the directory */var/mqm* from the NAS server with IP 10.0.0.100 using a dedicated network interface.
- There are heartbeats configured on each network interface.
- Each queue manager has modified the listener object to contain a unique port number.

6.8.4.1.4.3. Active/Active Configuration with Shared Storage

In the Active/Active configuration below, both Node1 and Node2 are primary LifeKeeper servers for WebSphere MQ resources. Each server is also the backup server for the other. In this example, Node1 protects the shared storage array for queue manager `QMGR1`. Node2 protects the shared storage array for queue manager `QMGR2` as the primary server. Additionally, each server acts as the backup for the other, which in this example means that Node2 is the backup for the queue manager `QMGR1` on Node1, and Node1 is the backup for the queue manager `QMGR2` on Node2.

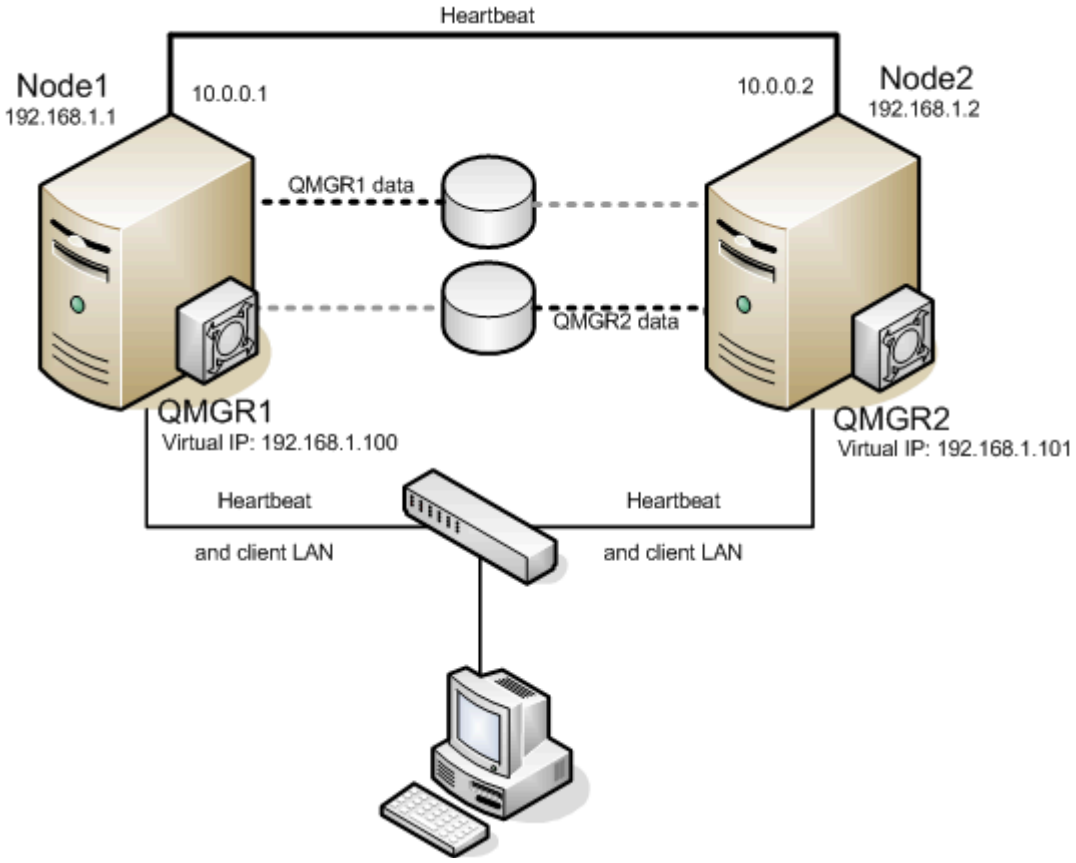


Figure 9 Active/Active Configuration with Shared Storage

Configuration Notes

- The clients connect to the queue manager `QMGR1` using the LifeKeeper floating IP 192.168.1.100.
- The clients connect to the queue manager `QMGR2` using the LifeKeeper floating IP 192.168.1.101.
- There are heartbeats configured on each network interface.
- Each queue manager has modified the listener object to contain a unique port number.
- `QMGR1` data is located on a volume group on the shared storage with two logical volumes

configured. Each logical volume contains a file system that is mounted on `QMDIR` or `QMLOGDIR`

- `QMGR2` data is located on a secondary volume group on the shared storage with two logical volumes configured. Each logical volume contains a file system that is mounted on `QMDIR` or `QMLOGDIR`

6.8.4.1.4.4. Active/Active Configuration with NAS Storage

In the Active/Active configuration below, both Node1 and Node2 are primary LifeKeeper servers for WebSphere MQ resources. Each server is also the backup server for the other. In this example, Node1 protects the NFS mount for queue manager `QMGR1`. Node2 protects the NFS mount for queue manager `QMGR2` as the primary server. Additionally, each server acts as the backup for the other, which in this example means that Node2 is the backup for the queue manager `QMGR1` on Node1, and Node1 is the backup for the queue manager `QMGR2` on Node2.

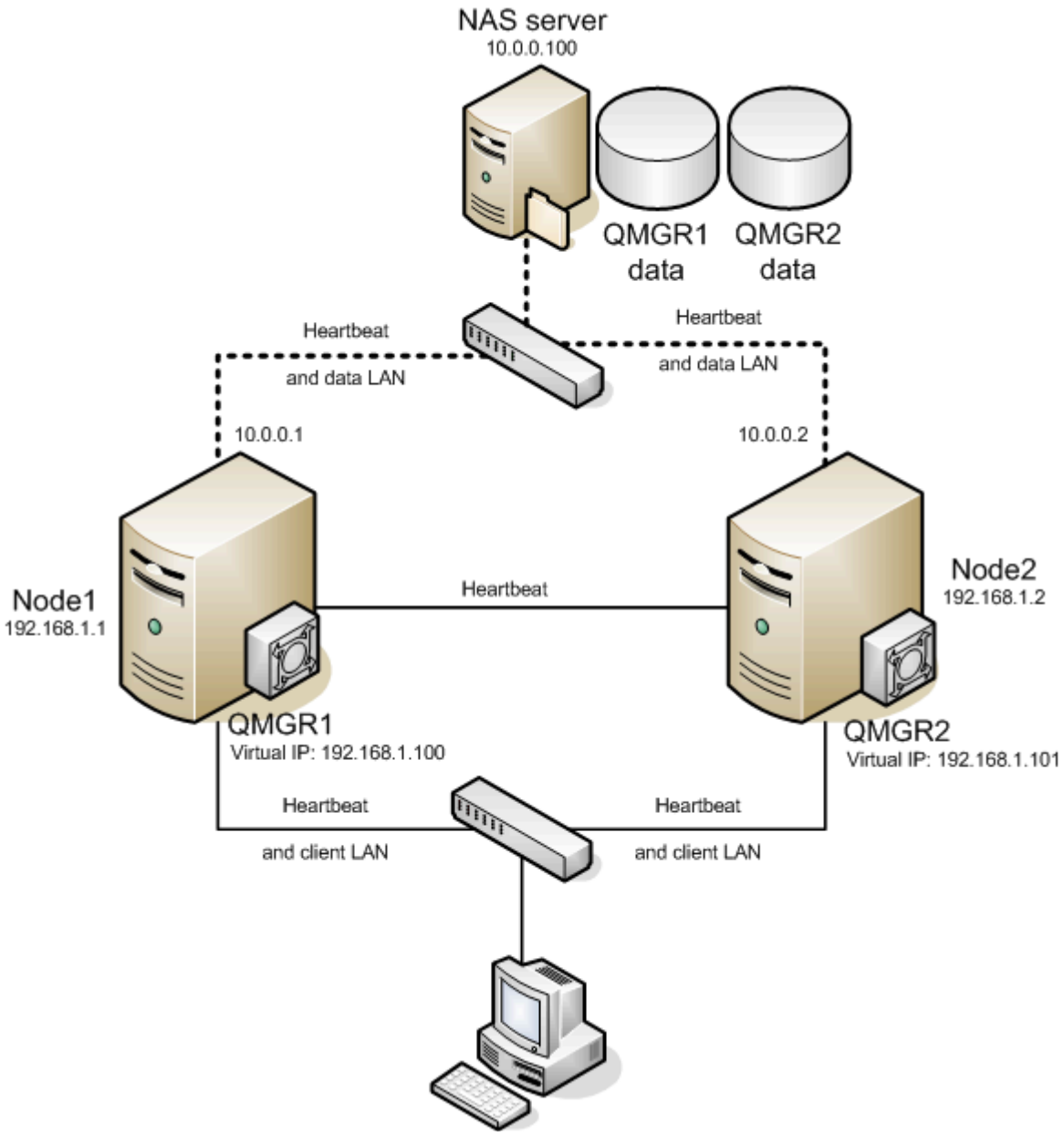


Figure 10 Active/Active Configuration with NFS Storage

Configuration Notes

- The clients connect to the queue manager `QMGR1` using the LifeKeeper floating IP 192.168.1.100.
- The clients connect to the queue manager `QMGR2` using the LifeKeeper floating IP 192.168.1.101.
- Each server has a dedicated network interface to access the NAS server.
- There are heartbeats configured on each network interface.
- Each queue manager has modified the listener object to contain a unique port number.
- `QMGR1` data is located on two NFS exports on the NAS server. The exports are mounted on *`QMDIR`* or *`QMLOGDIR`*. The NAS server IP is 10.0.0.100.
- `QMGR2` data is located on two NFS exports on the NAS server. The exports are mounted on *`QMDIR`* or *`QMLOGDIR`*. The NAS server IP is 10.0.0.100.

6.8.5. LifeKeeper Configuration Tasks for MQ

All SPS for Linux WebSphere MQ Recovery Kit administrative tasks can be performed via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer and monitor WebSphere resources.

Overview

The following tasks are described in this guide, as they are unique to a WebSphere MQ resource instance and different for each Recovery Kit.

- Create a Resource Hierarchy – Creates a WebSphere MQ resource hierarchy.
- Delete a Resource Hierarchy – Deletes a WebSphere MQ resource hierarchy.
- Extend a Resource Hierarchy – Extends a WebSphere MQ resource hierarchy from the primary server to a backup server.
- Unextend a Resource Hierarchy – Unextends (removes) a WebSphere MQ resource hierarchy from a single server in the LifeKeeper cluster.
- Editing Configuration Resource Properties – Reconfigures WebSphere MQ resource parameters including LifeKeeper test queue, listener management and stop timeouts after creation of the WebSphere MQ resource hierarchy.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View / Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You can also perform most of these tasks:

- From the toolbar
- By right-clicking on a global resource in the left pane of the status display
- By right-clicking on a resource instance in the right pane of the status display

Using the right-click method allows you to avoid entering information that is required when using the **Edit** menu.

6.8.5.1. Creating a WebSphere MQ Resource Hierarchy

After completing the necessary setup tasks, use the following steps to define the WebSphere MQ resource hierarchy.

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From here, select **Create Resource Hierarchy**.

   The ***Create Resource Wizard*** dialog box will appear with a drop-down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select **IBM WebSphereMQ** and click **Next**.

3. You will be prompted to enter the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter an error requiring you to correct previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

Field	Tips
Switchback Type	Choose either **Intelligent** or **Automatic**. This dictates how the WebSphere MQ instance will be switched back to this server when the server comes back up after a failover. The switchback type can be changed later from the **General** tab of the **Resource Properties** dialog box. **Note**: The switchback strategy should match that of the IP or File System resource to be used by the WebSphere MQ resource. If they do not match the WebSphere MQ resource, creation will attempt to reset them to match the setting selected for the WebSphere MQ resource.
Server	Select the **Server** on which you want to create the hierarchy.
Queue Manager Name	Select the WebSphere MQ queue manager you want to protect. The queue manager must be created prior to creating the resource hierarchy. Queue managers already under LifeKeeper protection are excluded from this list. The queue managers are taken from the global `mqs.ini` configuration file.
Manage Listener	Select **"YES"** to protect and manage the WebSphere MQ queue manager listener. Select **"NO"** if LifeKeeper should not manage the WebSphere MQ listener. **Note:** You can change this setting later. See Editing Configuration Resource Properties for details.
Server Connection Channel	Select the server connection channel to use for connection tests. By default, the channel `SYSTEM.DEF.SVRCONN` will be used; however, beginning with MQ Version 7.1, changes in MQ's Channel Authentication require that a channel other than the default be used and that the `MQADMIN` user be enabled for the specified channel. **Note:** Make sure the Server Connection Channel has been created PRIOR to creating your resource. For more information, see Configuring WebSphere MQ for Use with LifeKeeper.

	Note: This setting can be changed later. See Editing Configuration Resource Properties for details.
Virtual IP	Select the LifeKeeper virtual IP resource to include in the hierarchy. Select "**None**" if you do not want to include a LifeKeeper virtual IP in the WebSphere MQ hierarchy. **Note:** The virtual IP must be ISP (active) on the primary node to appear in the selection list.
IBM WebSphere MQ Resource Tag	Either select the default root tag offered by LifeKeepe, or enter a unique name for the resource instance on this server. The default is the queue manager name. Letters, numbers and the following special characters may be used: – _ . /

4. Click **Create**. The ***Create Resource Wizard*** will then create your WebSphere MQ resource hierarchy. LifeKeeper will validate the data entered. If LifeKeeper detects a problem, an error message will appear in the information box.

5. An information box will appear indicating that you have successfully created a WebSphere MQ resource hierarchy and that hierarchy must be extended to another server in your cluster in order to achieve failover protection. Click **Next**.

6. Click **Continue**. LifeKeeper will then launch the ***Pre-Extend Wizard***. Refer to Step 2 under Extending a WebSphere MQ Hierarchy for details on how to extend your resource hierarchy to another server.

6.8.5.2. Extending a WebSphere MQ Hierarchy

This operation can be started from the **Edit** menu or initiated automatically upon completing the **Create Resource Hierarchy** option, in which case you should refer to Step 2 below.

1. On the **Edit** menu, select **Resource**, then **Extend Resource Hierarchy**. The ***Pre-Extend Wizard*** appears. If you are unfamiliar with the **Extend** operation, click **Next**. If you are familiar with the **LifeKeeper Extend Resource Hierarchy** defaults and want to bypass the prompts for input/ confirmation, click **Accept Defaults**.

2. The ***Pre-Extend Wizard*** will prompt you to enter the following information.

   Note: The first two fields appear only if you initiated the **Extend** from the **Edit** menu.

Field	Tips
Template Server	Enter the server where your WebSphere MQ resource is currently in service.
Tag to Extend	Select the WebSphere MQ resource you wish to extend.
Target Server	Enter or select the server you are extending to.
Switchback Type	Select either **Intelligent** or **Automatic**. The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box. **Note:** Remember that the switchback strategy must match that of the dependent resources to be used by the WebSphere MQ resource.
Template Priority	Select or enter a priority for the template hierarchy. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (the number 1 indicates the highest priority). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note:** This selection will appear only for the initial extend of the hierarchy.
Target Priority	Either select or enter the priority of the hierarchy for the target server.
Queue Manager Name	This informational field shows the queue manager name you are about to extend. You cannot change this value.
Root Tag	LifeKeeper will provide a default tag name for the new WebSphere MQ resource instance on the target server. The default tag name is the same as the tag name for this resource on the template server. If you enter a new name, be sure it is unique on the target server. Letters, numbers and the following special characters may be used: – _ . /

Note: All configurable queue manager parameters like listener management, the name of the LifeKeeper test queue and the shutdown timeout values are taken from the template server.

3. After receiving the message that the pre-extend checks were successful, click **Next**.

4. Depending upon the hierarchy being extended, LifeKeeper will display a series of information boxes showing the Resource Tags to be extended which cannot be edited. Click **Extend**.

5. After receiving the message "**Hierarchy extend operations completed**", click **Next Server** to extend the hierarchy to another server or click Finish if there are no other extend operations to perform.

6. After receiving the message **"Hierarchy Verification Finished"**, click **Done**.

6.8.5.3. Unextending a WebSphere MQ Hierarchy

To remove a resource hierarchy from a single server in the LifeKeeper cluster, do the following:

1. On the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.
2. Select the **Target Server** where you want to unextend the WebSphere MQ resource. It cannot be the server where the WebSphere MQ resource is currently in service. (This dialog box will not appear if you selected the **Unextend** task by right-clicking on a resource instance in the right pane.) Click **Next**.
3. Select the WebSphere MQ hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the **Unextend** task by right-clicking on a resource instance in either pane.)
4. An information box appears confirming the target server and the WebSphere MQ resource hierarchy you have chosen to unextend. Click **Unextend**.
5. Another information box appears confirming that the WebSphere MQ resource was unextended successfully. Click **Done** to exit the **Unextend Resource Hierarchy** menu selection.

6.8.5.4. Deleting a WebSphere MQ Hierarchy

It is important to understand what happens to dependencies and protected services when a WebSphere hierarchy is deleted.

- **Dependencies**: Before removing a resource hierarchy, you may wish to remove the dependencies. Dependent file systems will be removed. Dependent non-file system resources like IP or Generic Application will not be removed as long as the delete is done via the LifeKeeper GUI or the WebSphere MQ delete script. For LifeKeeper to not delete the dependent file systems of the WebSphere MQ queue manager, manually remove the dependencies prior to deleting the WebSphere MQ hierarchy.

- **Protected Services**: If the WebSphere resource hierarchy is taken out of service before being deleted, the WebSphere daemons for this queue manager will be stopped. If a hierarchy is deleted while it is in service, the WebSphere MQ daemons will continue running and offering services (without LifeKeeper protection) after the hierarchy is deleted.

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.

2. Select the **Target Server** where you will be deleting your WebSphere MQ resource hierarchy and click **Next**. (This dialog will not appear if you selected the Delete Resource task by right-clicking on a resource instance in either pane.)

3. Select the **Hierarchy to Delete**. (This dialog will not appear if you selected the Delete Resource task by right-clicking on a resource instance in the left or right pane.) Click **Next**.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete**.

5. Another information box appears confirming that the WebSphere resource was deleted successfully.

6. Click **Done** to exit.

6.8.5.5. Testing a WebSphere MQ Resource Hierarchy

You can test your WebSphere MQ resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

On the **Edit** menu, select **Resource**, then **In Service**. For example, an In Service request executed on a backup server causes the application hierarchy to be taken out of service on the primary server and placed in service on the backup server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out of Service** request, the application is taken out of service without bringing it in service on the other server.

Testing Shared Storage Configuration

To test WebSphere MQ shared storage operations, perform the following steps:

1. Create a temporary test queue on the primary server with the default persistency of "**yes**"

```
mqm@node1:/opt/mqm/samp/bin> runmqsc TEST.QM
5724-B41 © Copyright IBM Corp. 1994, 2002. ALL RIGHTS RESERVED.
Starting MQSC for queue manager TEST.QM.

define qlocal(TEST) defpsist(yes)
     1 : define qlocal(TEST) defpsist(yes)
AMQ8006: WebSphere MQ queue created.
end

     2 : end
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

2. Put a message into the test queue created on the primary node:

```
mqm@node1:/opt/mqm/samp/bin> echo "HELLO WORLD on NODE1" | ./amqsput TEST
TEST.QM
Sample AMQSPUT0 start
target queue is TEST
Sample AMQSPUT0 end
```

3. Browse the test queue to see if the message has been stored:

```
mqm@node1:/opt/mqm/samp/bin> ./amqsbcg TEST TEST.QM
```

You should see a message with the content “HELLO WORLD on NODE1” and some additional output. Look for the following line and verify that the
persistency is 1:

```
[...]
   Priority : 0 Persistence : 1
[...]
```

4. Switch the resource hierarchy to the standby node.

5. On the standby server where the queue manager is now active, repeat Step 3. The message should be accessible on the standby server. If not, check your storage configuration.

6. On the standby server where the queue manager is now active, get the message from the test queue:

```
mqm@node1:/opt/mqm/samp/bin> ./amqsget TEST TEST.QM
Sample AMQSGET0 start
message <HELLO WORLD on NODE1>
<now wait 15 seconds>
no more messages
Sample AMQSGET0 end
```

7. Delete the test queue created in Step 1.

```
mqm@node1:/opt/mqm/samp/bin> runmqsc TEST.QM
5724-B41 © Copyright IBM Corp. 1994, 2002. ALL RIGHTS RESERVED.
Starting MQSC for queue manager TEST.QM.

delete qlocal(TEST)
   1 : delete qlocal(TEST)
MQ8007: WebSphere MQ queue deleted.
end
   2 : end
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

6.8.5.5.1. Testing MQ Client Connectivity

Testing Client Connectivity

To test client connectivity, perform the following steps:

1. On the primary server, use the amqsbcgc command to connect to the queue manager:

   ```
   export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/192.168.1.90(1414)'
   ```

   Note: Replace the IP 192.168.1.90 with the LifeKeeper protected virtual IP of the queue manager. If your queue manager uses a different port other than 1414, then replace the port number 1414 with the one being used. If the server connection channel being used is not the default `SYSTEM.DEF.SVRCONN` channel, then replace the server connection channel `SYSTEM.DEF.SVRCONN` with the one being used.

   You should see the following output:

   ```
   mqm@node1:/opt/mqm/samp/bin> ./amqsbcgc LIFEKEEPER.TESTQUEUE TEST.QM

   AMQSBCG0 - starts here
   **********************

   MQOPEN - 'LIFEKEEPER.TESTQUEUE'

   No more messages
   MQCLOSE
   ```

If you get a message like the following, then the test queue `LIFEKEEPER.TESTQUEUE` is not configured. Create the test queue as described in section Configuring WebSphere MQ for Use with LifeKeeper and repeat the test.

```
AMQSBCG0 - starts here
**********************

MQOPEN - 'LIFEKEEPER.TESTQUEUE'
MQOPEN failed with CompCode:2, Reason:2085
```

2. Perform a switchover of the resource hierarchy.

3. Repeat Step 1 on the same server as before which is now the standby server after the switchover.

Testing If PUT/GET Tests are Performed

To test if the WebSphere MQ Recovery kit performs all checks including the `PUT/GET` test, perform the

following:

1. Make sure the queue manager is in service (ISP) on any server.

2. Increase the logging level of the queue manager as described in Changing the Log Level to **“FINE”**.

3. Open the log dialog on the machine where the queue manager is active and wait for the next check to happen (max. two minutes).

4. Analyze the log and verify that all checks are performed and none of the tests is skipped. The `PUT/GET` could be skipped for the following reasons:

   a. No LifeKeeper test queue is configured (in this case, configure the test queue as described in Changing the LifeKeeper Test Queue Name).

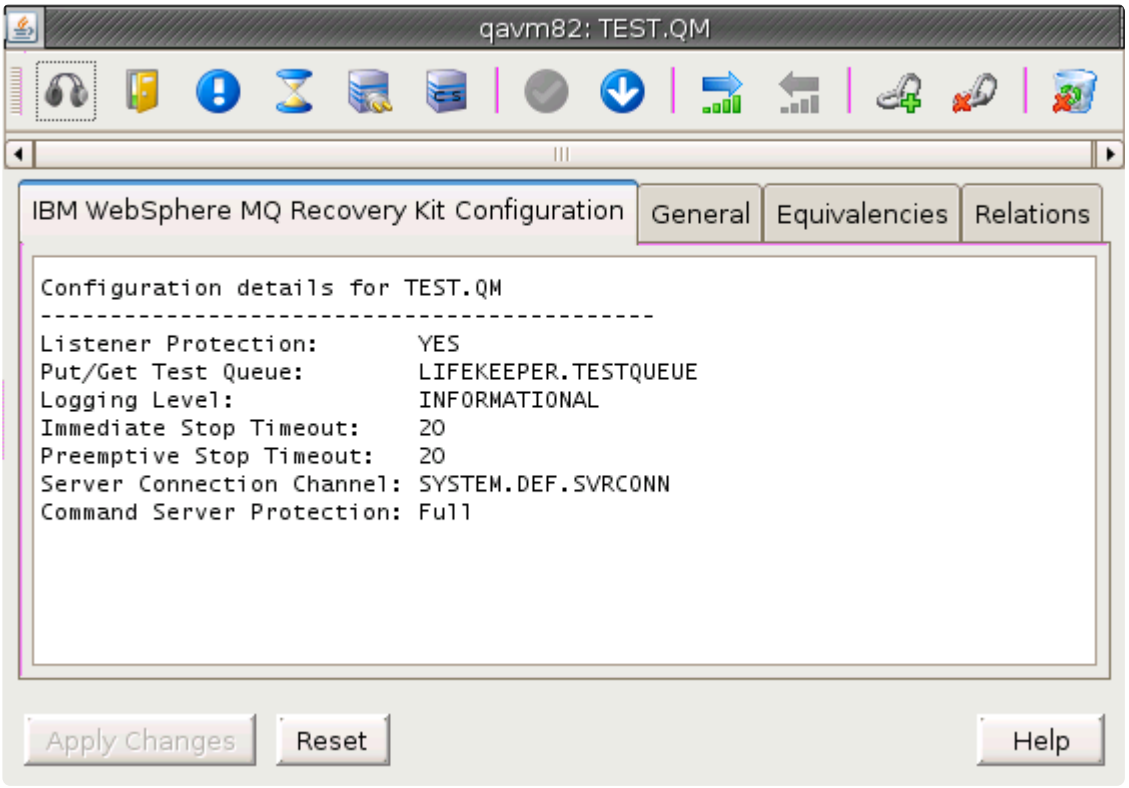
   b. LifeKeeper test queue does not exist (in this case, create the test queue as described in Configuring WebSphere MQ for Use with LifeKeeper).

   c. The modified amqsget© executables are not available (in this case, install a C compiler and rerun the script */opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/compilesamples*).

5. Set the debug level to `“INFORMATIONAL”` again.

6.8.5.6. Viewing MQ Resource Properties

To view the IBM WebSphere MQ resource properties, right-click on the icon for the resource/server combination for which you want to view the properties. When the resource context menu appears, click **Properties**. The following dialog will appear.



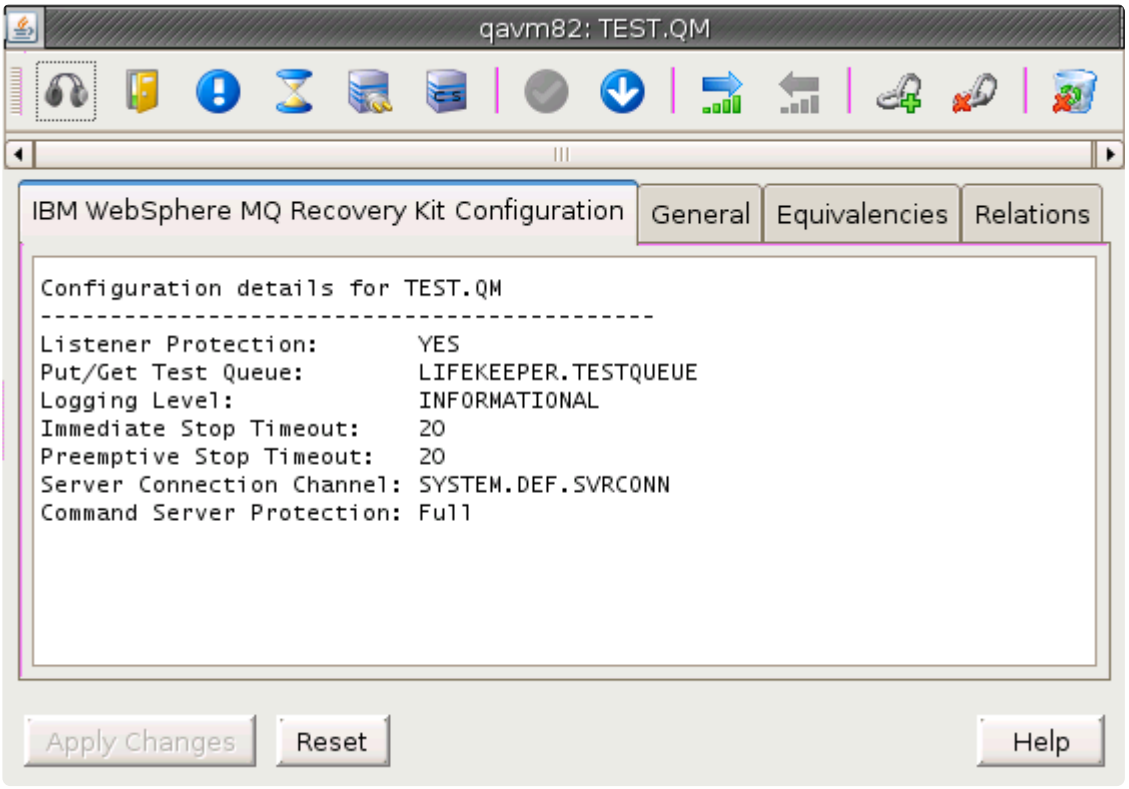
Resource properties will be displayed in the properties panel if it is enabled. You can also right-click on the icon for the global resource for which you want to view the properties. When the **Resource Context Menu** appears, click **Properties**. When the dialog comes up, select the server for which you want to view that resource from the Server list.

6.8.5.7. Editing MQ Configuration Resource Properties

The WebSphere MQ Properties page allows you to view and modify the configuration details for a specific WebSphere MQ resource via the properties panel if it is enabled. Specific WebSphere MQ resource configuration properties can also be modified via the **Resource Context Menu**.

To edit configuration details via the WebSphere MQ Configuration Properties page from the LifeKeeper GUI Properties Panel, you must first ensure the GUI Properties Panel is enabled. To enable the GUI Properties Panel, select **View**, then **Properties Panel** (must have a check mark to indicate it is enabled). Once enabled, left-click on the **WebSphere MQ** resource to display its configuration details in the LifeKeeper GUI Properties Panel.

Below is an example of the properties page that will appear in the LifeKeeper GUI Properties Panel for a WebSphere MQ resource.



The properties page contains four tabs. The first tab, labeled **IBM WebSphere MQ Recovery Kit Configuration**, contains configuration information that is specific to WebSphere MQ resources and allows modification via the resource specific icons. The remaining three tabs are available for all LifeKeeper resource types and their content is described in the topic Resource Properties Dialog in the SPS for Linux Technical Documentation.

The following table displays the WebSphere MQ resource specific icons and the configuration component that can be modified when clicking on the icon.

	**Listener Protection Configuration**	Allows you to specify whether protection of the IBM WebSphere MQ listener is included with the other IBM WebSphere MQ queue manager components being protected.
	**PUT/GET Test Queue Configuration**	Allows you to change the name of the queue that the IBM WebSphere MQ Recovery Kit will use to perform PUT/GET tests for the queue manager being protected.
	**Logging Level Configuration**	Allows you to modify the log level that the IBM WebSphere MQ Recovery Kit will use for the queue manager being protected.
	**Shutdown Timeout Configuration**	Allows you to modify the timeout in seconds for the immediate shutdown and preemptive shutdown timers for the IBM WebSphere MQ queue manager being protected.
	**Server Connection Channel Configuration**	Allows you to modify the server connection channel that is used for client connection and the `PUT/GET` testing for the IBM WebSphere MQ queue manager being protected.
	**Command Server Protection Configuration**	Allows you to specify the protection/recovery level for command server component of the IBM WebSphere MQ queue manager being protected.

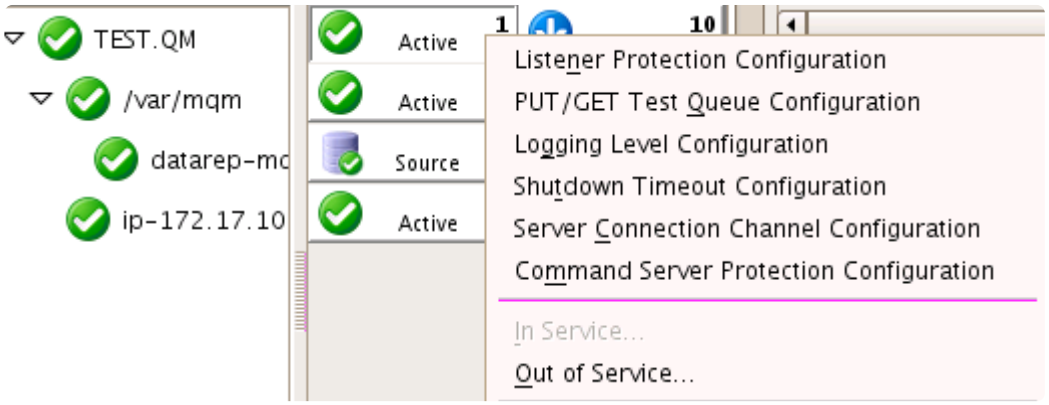
More details on each of these configuration options can be found below.

Listener Management	Specifies whether you want LifeKeeper to protect the listener for the queue manager or not. If listener management is disabled (value of NO), LifeKeeper will not monitor the listener and you can stop the listener without causing LifeKeeper recovery actions. If listener management is enabled (value of YES), LifeKeeper will monitor the listener and restart the listener if the listener is not running. If the recovery fails, a failover of the WebSphere MQ hierarchy to the backup server is initiated.
LifeKeeper Test Queue	LifeKeeper performs `PUT/GET` test to monitor queue manager operations. The WebSphere MQ Recovery Kit uses a **dedicated** test queue to put messages in and retrieve messages again. In case a failure is detected, no recovery or failover is performed. Instead, the Recovery Kit sends an event that you can register to receive. The events are called `putgetfail` and `putgetcfail`. You can add a notification script to the directories */opt/LifeKeeper/events/mqseries/putgetfail* and */opt/LifeKeeper/events/mqseries/putgetcfail* to react to those events. **Note 1**: If the LifeKeeper test queue is not configured in the queue manager, the `PUT/GET` test is skipped. No recovery or failover takes place. **Note 2**: If the listener is protected, a second client connect check will be done. If this check fails, a recovery or failover of the queue manager is attempted.
Logging Level	You can set the logging level of the WebSphere MQ Recovery Kit to four presets: • ERROR In this log level, only errors are logged. No informational messages are logged. • INFORMATIONAL (default) In this log level, LifeKeeper informational messages about start, stop and

	recovery of resources are logged. • DEBUG In this log level, the informational LifeKeeper messages and the command outputs from all WebSphere MQ commands in the restore, remove and recovery scripts are logged. • FINE In this log level, all command outputs from WebSphere MQ commands issued in start, stop, recovery and quickCheck scripts are logged. Additional debug messages are also logged. It is recommended to set this debug level only for debugging purpose. As `quickCheck` actions are also logged, this fills up the log files each time a `quickCheck` for the WebSphere MQ queue manager runs. The default is *INFORMATIONAL*. This is equivalent to normal LifeKeeper logging of other recovery kits. **Note**: Independent of the logging level setting, WebSphere MQ errors during start, stop, recovery or during the check routine are always logged with the complete command output of the last command run.
Stop Timeout Values	The WebSphere MQ Recovery Kit stops the queue manager in 3 steps: 1. immediate stop 2. preemptive stop 3. kill all queue manager processes The timeout values specified determine the time the Recovery Kit waits in Steps 1 and 2 for a successful completion. If this timeout is reached, the next step in the shutdown process is issued. The default for the immediate and preemptive shutdown timeouts is ***20 seconds***.
Server Connection Channel	The WebSphere MQ Recovery Kit allows the specification of the server connection channel. By default, the kit will use the channel `SYSTEM.DEF.SVRCONN`, but an alternate channel can be specified during resource creation or at any time after resource creation.
Command Server	The WebSphere MQ Recovery Kit allows two levels of protection and recovery for the command server component for the protected queue manager. The levels are **Full** and **Minimal**. With **Full** protection, the command server will be started, stopped, monitored and recovered or failed over if recovery is unsuccessful. The recovery steps with **Full** protection are: • Attempt to restart just the command server process. • If that fails, attempt a full restart of the queue manager including the command server process.

	• If both attempts are unsuccessful at restarting the command server, then initiate a failover to the standby node. With **Minimal** protection, the command server will only be started during restore or stopped during remove. No monitoring or recovery of the command server will be performed. **NOTE**: Starting the command server will only be performed by the Recovery Kit during restore if the queue manager `SCMDSERV` parameter is set for manual startup. During a recovery, a failed command server restart will always be attempted regardless of the `SCMDSERV` setting unless the Command Server Protection Level is set to **Minimal**.

As previously noted, these WebSphere MQ resource configuration components can be modified using the resource specific icons in the properties panel or via the **Resource Context Menu**.



The parameters above can be set for each queue manager separately either via the LifeKeeper GUI or via a command line utility.

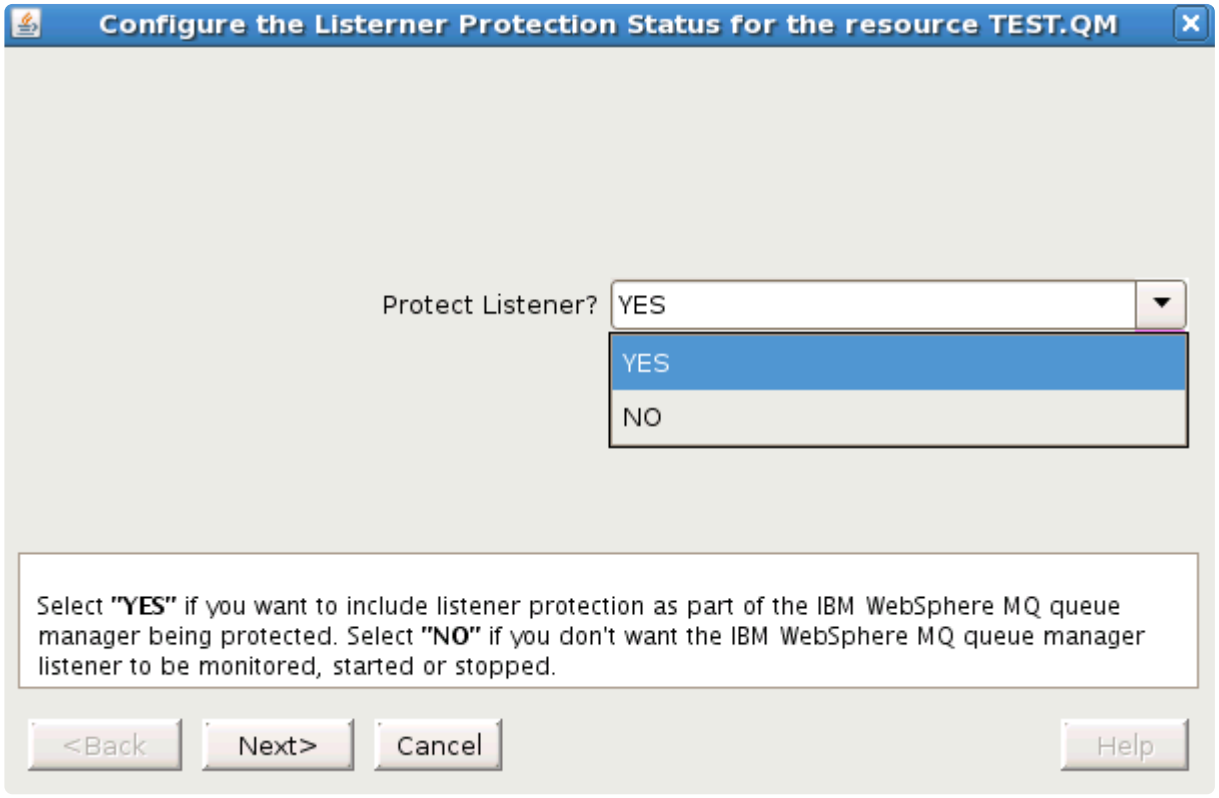
To set the parameters via the command line, use the script:

```
$LKROOT/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam
```

6.8.5.7.1. Enable/Disable Listener Protection

GUI

First navigate to the **WebSphere MQ Resource Properties Panel** or the **Resource Context Menu** described above. The resource must be in service to modify the Listener Protection value. Then click on **Listener Protection Configuration** icon or menu item. The following dialog will appear:



Now select **YES** if you want LifeKeeper to start, stop and monitor the WebSphere MQ listener. Select **NO** if LifeKeeper should not start, stop and monitor the WebSphere MQ listener. Click **Next**. You will be asked if you want to enable or disable listener protection; click **Continue**. If you have chosen to enable listener management, the LifeKeeper GUI checks if the listener is already running. If it is not already running, it will try to start the listener. If the listener start was successful, the LifeKeeper GUI will enable listener management on each server in the cluster. If the listener is not running and could not be started, the LifeKeeper GUI will not enable listener management on the servers in the cluster.

Command Line

To set the LifeKeeper listener management via command line, use the following command:

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p LISTENERPROTECTION -v YES
```

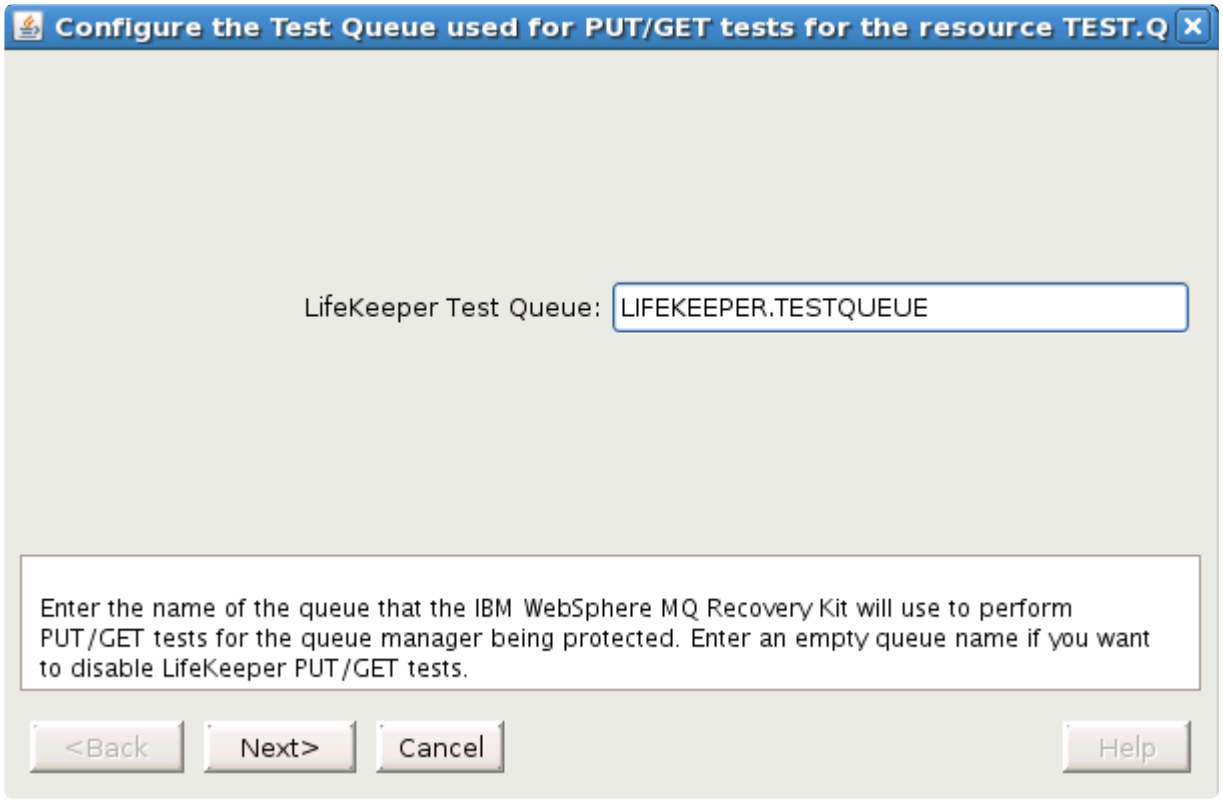
This will set (-s) the LifeKeeper listener management (-p) on each node of the cluster (-c) to **YES** (-v) (enable listener management) for queue manager TEST.QM (-i).

Note: You can either use the queue manager name `(-i)` or the LifeKeeper TAG `(-t)` name.

6.8.5.7.2. Changing the LifeKeeper Test Queue Name

GUI

First navigate to the **WebSphere MQ Resource Properties Panel** or the Resource Context Menu described above. Then click on **`PUT/GET TESTQUEUE` Configuration** icon or menu item. The following dialog will appear:



Now enter the name of the LifeKeeper test queue and click **Next**. You will be asked if you want to set the new LifeKeeper test queue; click **Continue**. Next, the LifeKeeper GUI will set the LifeKeeper test queue on each server in the cluster. If you set the test queue to an empty value, no `PUT/GET` tests are performed.

Command Line

To set the LifeKeeper test queue via command line, use the following command:

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p TESTQUEUE -v "LIFEKEEPER.TESTQUEUE"
```

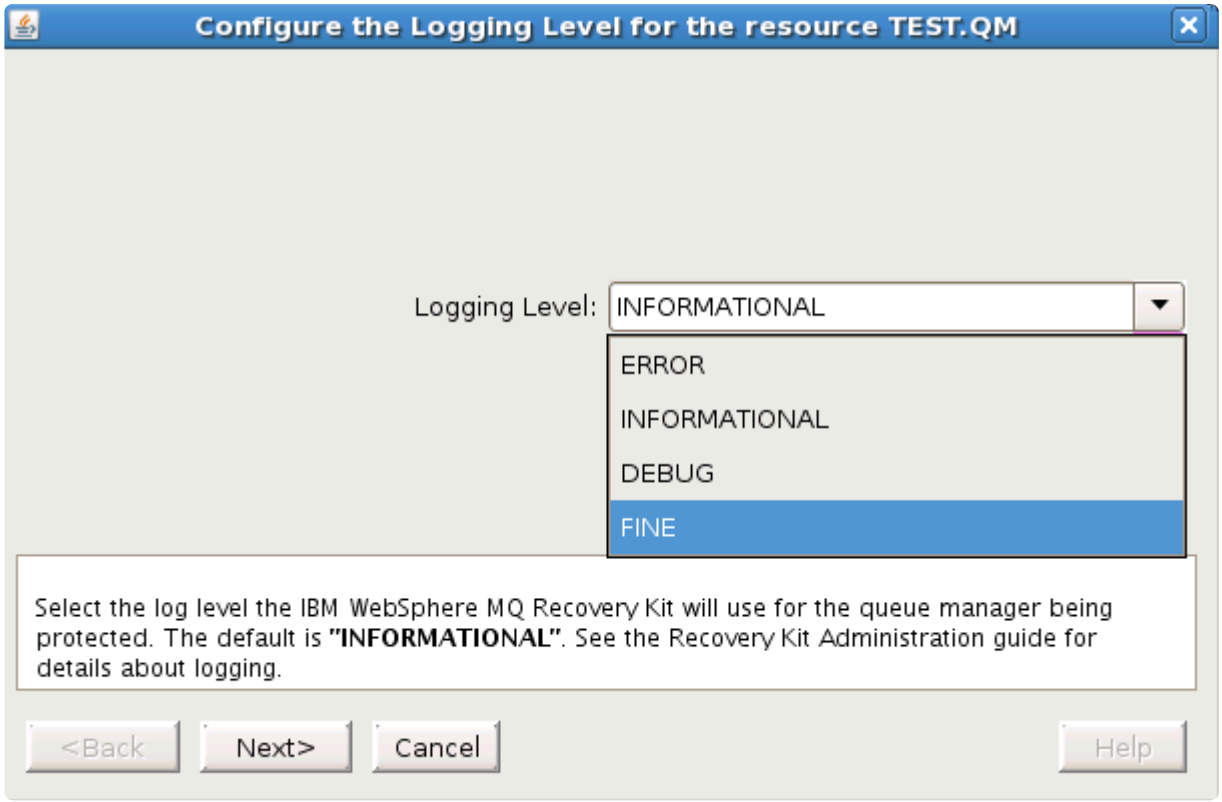
This will set (-s) the LifeKeeper test queue (-p) on each node of the cluster (-c) to `LIFEKEEPER.TESTQUEUE (-v)` for queue manager `TEST.QM (-i)`.

Note: You can either use the queue manager name (-i) or the LifeKeeper TAG (-t) name.

6.8.5.7.3. Changing the Log Level

GUI

First navigate to the **WebSphere MQ Resource Properties Panel** or the **Resource Context Menu** described above. Then click on **Logging Level Configuration** icon or menu item. The following dialog will appear:



Now select the **Logging Level** and click **Next**. You will be asked if you want to set the new LifeKeeper logging level; click **Continue**. Next, the LifeKeeper GUI will set the LifeKeeper logging level for the selected queue manager on each server in the cluster.

Command Line

To set the LifeKeeper logging level via command line, use the following command:

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p DEBUG -v DEBUG
```

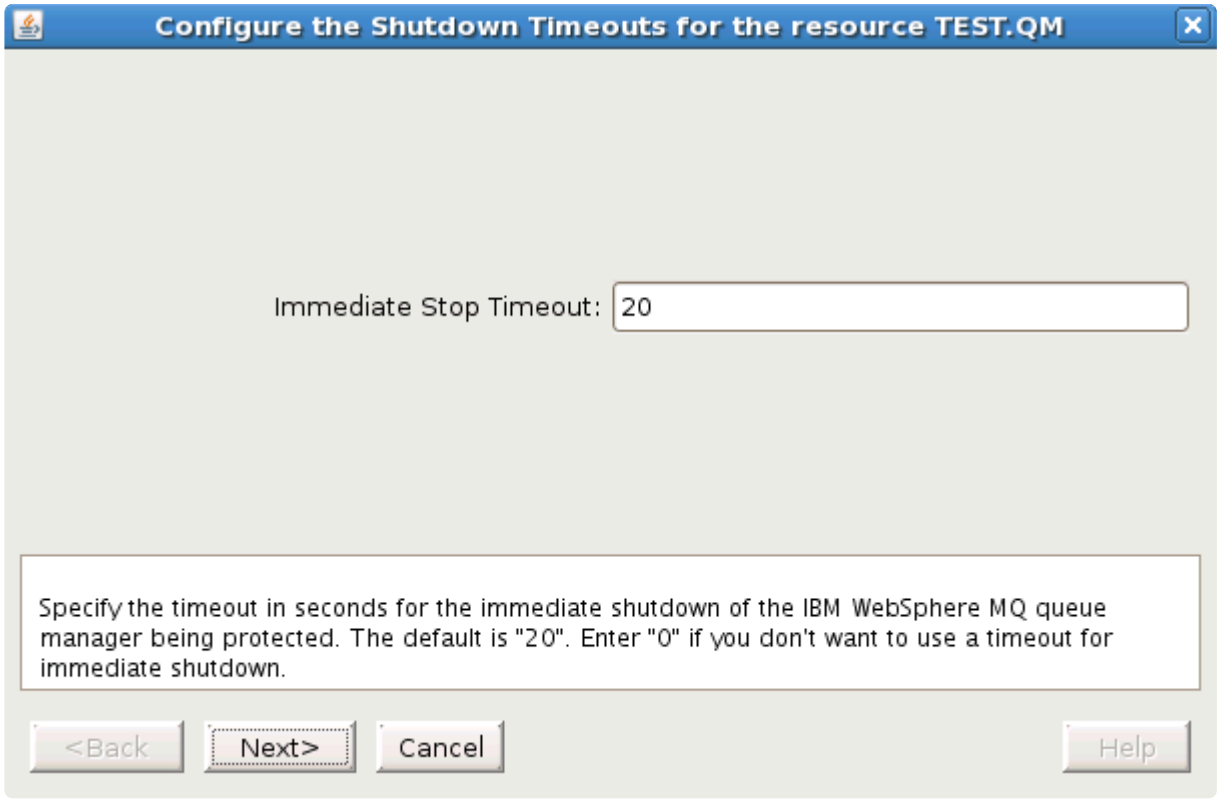
This will set (-s) the LifeKeeper logging level (-p) on each node of the cluster (-c) to `DEBUG (-v)` for queue manager `TEST.QM (-i)`.

Note: You can either use the queue manager name (-i) or the LifeKeeper TAG (-t) name.

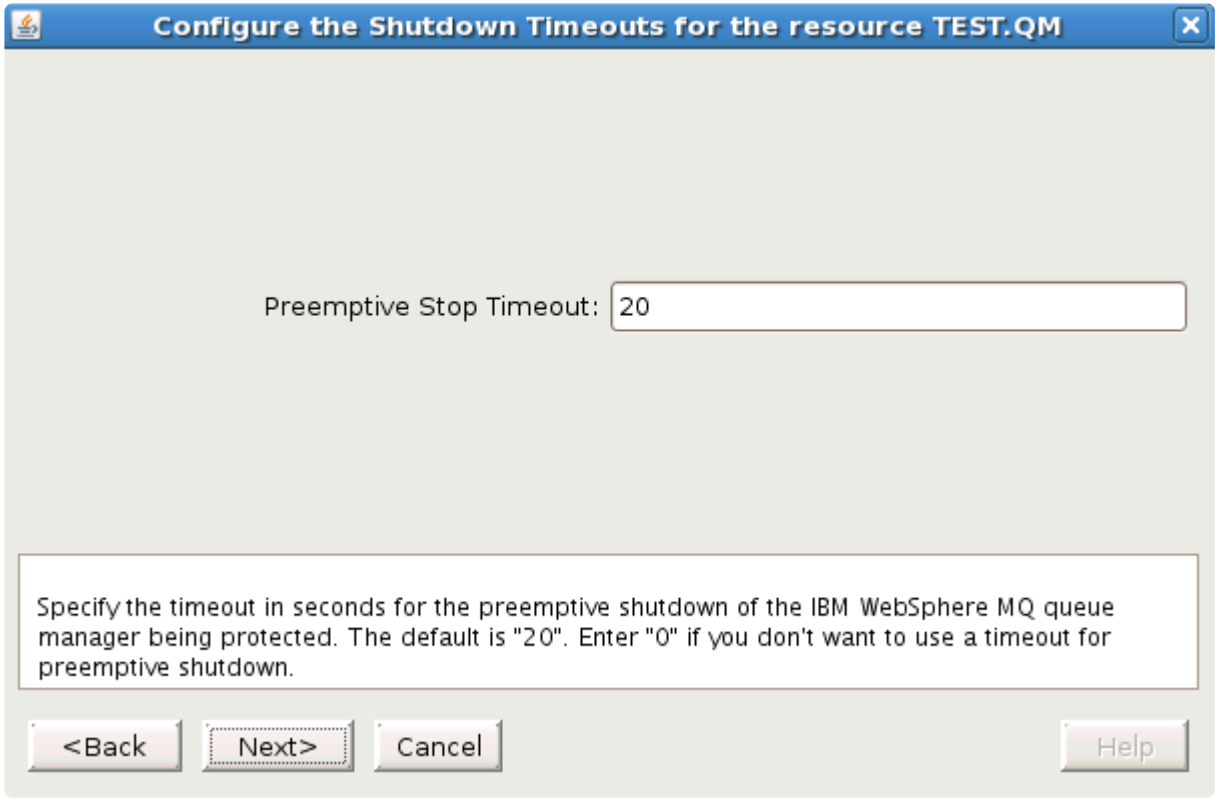
6.8.5.7.4. Changing Shutdown Timeout Values

GUI

First, navigate to the WebSphere MQ resource properties panel or the resource context menu described above. Then click on **Shutdown Timeout Configuration** icon or menu item. The following dialog will appear:



Now enter the immediate shutdown timeout value in seconds and click **Next**. If you want to disable the immediate shutdown timeout, enter **0**. Now the following dialog will appear:



Now enter the preemptive shutdown timeout value in seconds and click **Next**. If you want to disable the preemptive shutdown timeout enter 0. You will be asked if you want to set the new LifeKeeper timeout parameters, click **Continue**. Next, the LifeKeeper GUI will set the LifeKeeper immediate and preemptive timeout values on each server in the cluster.

Command Line

To set the **preemptive shutdown** timeout values via command line, use the following command:

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p PREEMPTIVE_TIMEOUT -v 20
```

This will set (-s) the LifeKeeper preemptive shutdown timeout (-p) on each node of the cluster (-c) to *20 seconds* (-v) for queue manager TEST.QM (-i).

To set the **immediate shutdown** timeout values via command line, use the following command:

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p IMMEDIATE_TIMEOUT -v 20
```

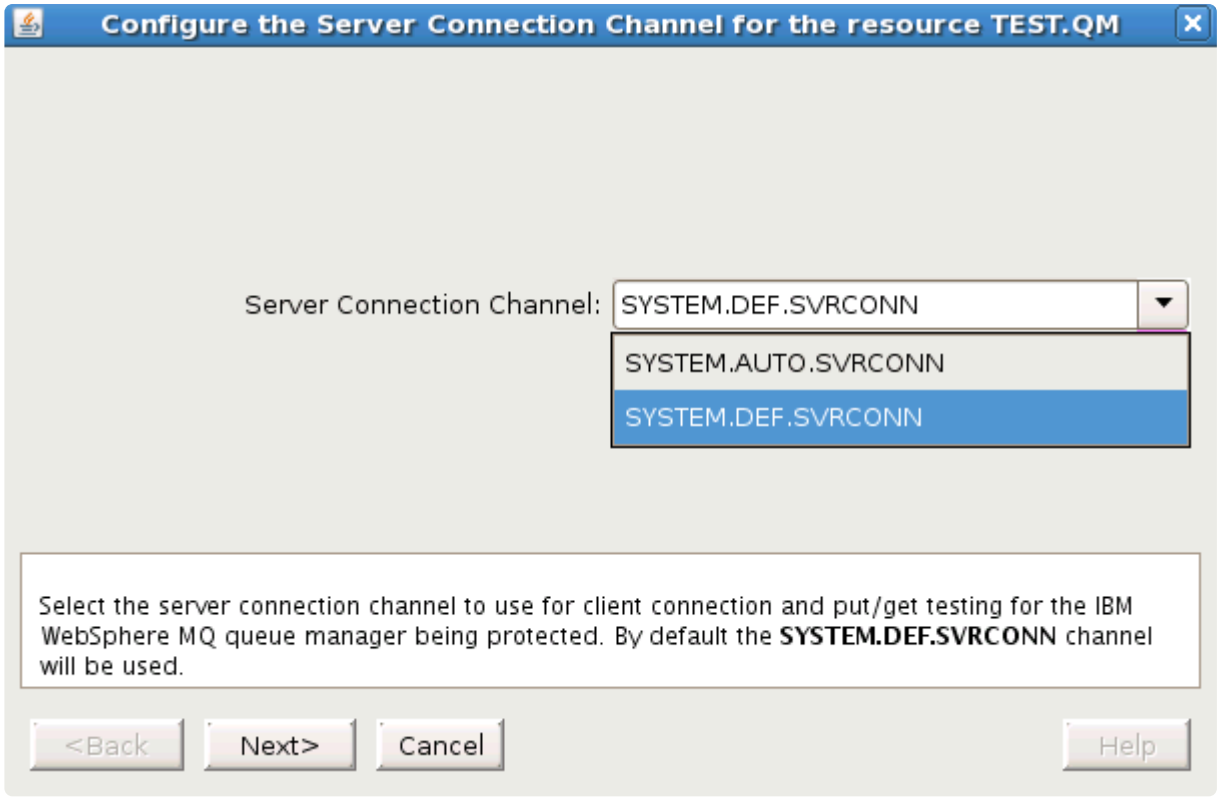
This will set (-s) the LifeKeeper immediate shutdown timeout (-p) on each node of the cluster (-c) to *20 seconds* (-v) for queue manager TEST.QM (-i).

Note: You can either use the queue manager name (-i) or the LifeKeeper TAG (-t) name.

6.8.5.7.5. Changing the Server Connection Channel

GUI

First navigate to the WebSphere MQ resource properties panel or the resource context menu described above. The resource must be in service to modify the **Server Connection Channel** value. Then click on **Server Connection Channel Configuration** icon or menu item. The following dialog will appear:



Now select the **Server Connection Channel** to use and click **Next**. You will be asked if you want to change to the new **Server Connection Channel**, click **Continue**. Next, the LifeKeeper GUI will set the Server Connection Channel for the selected queue manager on each server in the cluster.

Command Line

To set the Server Connection Channel via command line use the following command:

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p CHANNEL -v LK.TEST.SVRCONN
```

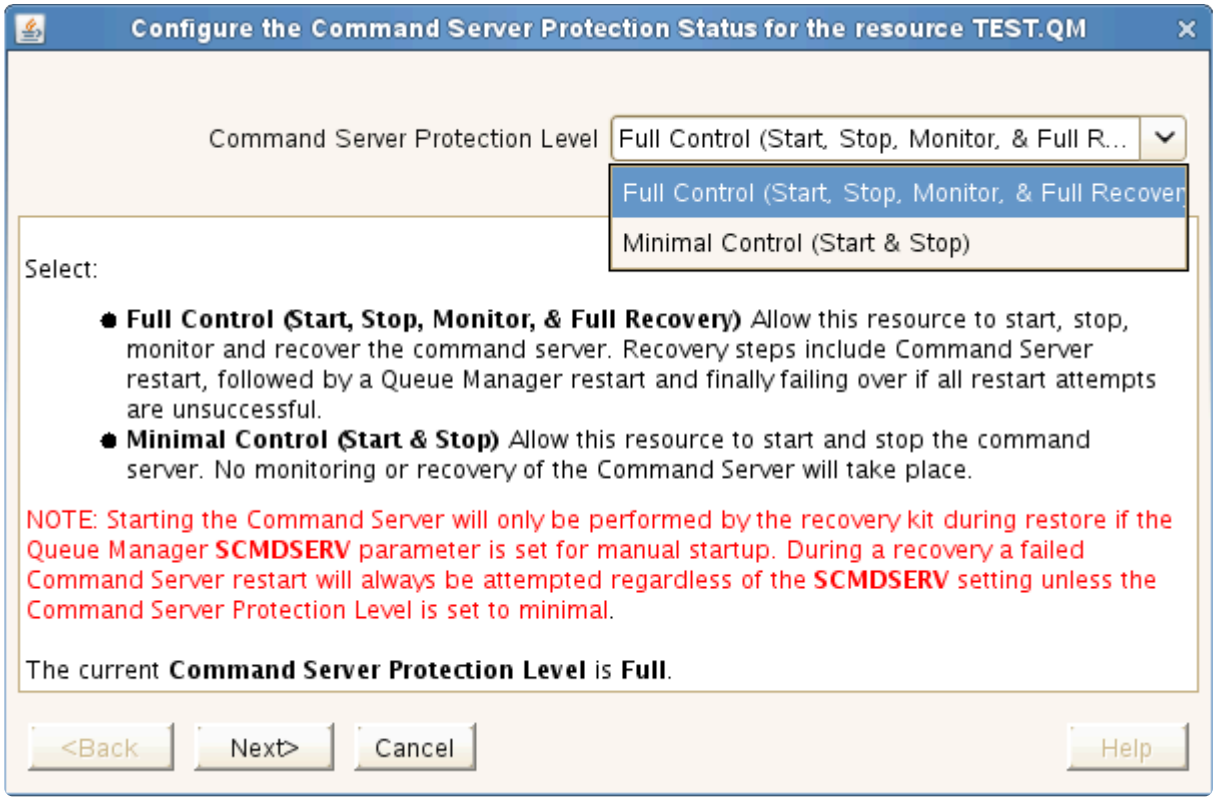
This will set `(-s)` the Server Connection Channel `(-p)` on each node of the cluster `(-c)` to *`LK.TEST.SVRCONN`* for queue manager `TEST.QM (-i)`.

Note: You can either use the queue manager name (-i) or the LifeKeeper TAG (-t) name.

6.8.5.7.6. Changing the Command Server Protection Configuration

GUI

First navigate to the WebSphere MQ **Resource Properties Panel** or the **Resource Context Menu** described above. Then click on **Command Server Protection Configuration** icon or menu item. The following dialog will appear:



Select **Full Control** of the command server component of the WebSphere MQ queue manager to have LifeKeeper start, stop, monitor and attempt to recover and to then fail over if the recovery attempt is unsuccessful.

Select **Minimal Control** of the command server component of the WebSphere MQ queue manager to have LifeKeeper only start and stop but not monitor or attempt any recovery.

See above table for more details. Once the protection control is selected, click **Next**. You will be asked if you want to change the setting of the command server protection from its current setting to the new setting; click **Continue** to make the change on all nodes in the cluster.

Command Line

To set the **LifeKeeper Command Server Protection Configuration** via the command line, use the following command:

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p CMDSERVERPROTECTION -v LEVEL
```

where **`LEVEL`** is **Full** or **Minimal**.

This will set (-s) the LifeKeeper Command Server Protection Configuration (-p) on each node in the cluster (-c) to LEVEL (-v) for queue manager TEST.QM (-i).

Note: You can use either the queue manager name (-i) or the LifeKeeper TAG (-t) name.

6.8.5.7.7. Changing LifeKeeper WebSphere MQ Recovery Kit Defaults

The IBM WebSphere MQ Recovery Kit uses a number of default values which can be tuned and modified if you have problems with the default settings. The default settings should be sufficient for most environments. If you have problems with timeouts you can use the following table to identify tunable parameters. It is recommended that you do not change the parameters until you have problems with your WebSphere MQ resource hierarchies.

Variable Name in /etc/default/LifeKeeper	Default Value	Description
MQS_QUICKCHECK_TIMEOUT_SC	10 (seconds)	Timeout for the client connect check.
MQS_QUICKCHECK_TIMEOUT_CC	10 (seconds)	Timeout for the client connect check.
MQS_QUICKCHECK_TIMEOUT_PUTGET	10 (seconds)	Timeout for the PUT/GET check
MQS_QUICKCHECK_TIMEOUT_PS	5 (seconds)	Timeout for the check whether publish/ subscribe is in use or not
MQS_QUICKCHECK_TIMEOUT_CLUSTER	5 (seconds)	Timeout for the check whether this queue manager is part of an WebSphere MQ cluster
MQS_QUICKCHECK_TIMEOUT	40 (seconds)	Timeout for the quickCheck script (must be at least 10 seconds).
MQS_QMGR_START_TIMEOUT	60 (seconds)	Timeout for the queue manager start command to complete.
MQS_CMDS_START_TIMEOUT	30 (seconds)	Timeout for the command server start command to complete.
MQS_LISTENER_START_TIMEOUT	30 (seconds)	Timeout for the listener start command to complete
MQS_LISTENER_LIST_TIMEOUT	10 (seconds)	Timeout for the listener list command to complete
MQS_CHECK_TIMEOUT_ACTION	ignore	The action in case a server connect check or client connect check times out. The default of “ignore” means that a message about the timeout is logged, but no recovery is initiated. If you set this variable to “sendevent” local recovery is initiated in case a server connect check timed out.
MQS_LISTENER_CHECK_DELAY	2 (seconds)	The time in seconds between the start of the listener and the check for the successful listener start. The default of 2 seconds should be sufficient to detect port in use conditions.
NO_AUTO_STORAGE_DEPS	0	If you set the variable to 1 the recovery

		kit does not check if the queue manager and log directory are located in shared storage. If set to 1 the recovery kit does not create file system hierarchies upon resource configuration too.
MQS_DSPMQVER_TIMEOUT	5 (seconds)	Timeout for the dspmqver command (needed to find out the version of WebSphere MQ), must be at least 2 seconds.
MQS_SKIP_CRT_MISSING_Q	0	Set to 1 to not automatically create a missing test queue.
MQS_ALT_USER_NAME	mqm if not set or the user does not have membership in the “mqm” group	The alternate user name to use for all WebSphere MQ commands. By default the user “mqm” is used. If set the alternate user must have its primary group set to the group “mqm” or must have secondary membership in that group.

To change the parameters add the appropriate variable in the table above to `/etc/default/LifeKeeper`. The line should have the following syntax:

```
[…]
MQS_CHECK_TIMEOUT_ACTION=sendevent
[…]
```

To disable a custom setting and fall back to the default value, just remove the line from `/etc/default/LifeKeeper` or comment out the corresponding line.

6.8.6. WebSphre MQ Troubleshooting

WebSphere MQ Log Locations

If the queue manager name is known and the queue manager is available, WebSphere MQ error logs are located in the directory specified by the `LogPath` parameter defined in the queue manager configuration file *`qm.ini`*. If the queue manager is not available, error logs are located in: */var/mqm/qmgrs/@SYSTEM/errors*. If an error has occurred with a client application, error logs are located on the client's root drive in: */var/mqm/errors*.

If your application gets a return code indicating that a Message Queue Interface (MQI) call has failed, refer to the *WebSphere MQ Application Programming Reference Manual* for a description of that return code.

6.8.6.1. MQ Error Messages

This section provides a list of messages that you may encounter with the use of the SPS MQ Recovery Kit. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Because the MQ Recovery Kit relies on other SPS components to drive the creation and extension of hierarchies, messages from these other components are also possible. In these cases, please refer to the Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

Common Error Messages

Error Number	Error Message	Action
119001	Queue manager with TAG “TAG” failed to start on server “SERVER” with return code “Code”	The **`start`** command was successful, but the check after the start failed. Check the IBM WebSphere MQ alert log on SERVER for possible errors and correct them.
119002	Queue manager with TAG “TAG” start command failed on server “SERVER” with return code “Code”.	The **`start`** command for the queue manager TAG returned with non zero value. Check the IBM WebSphere MQ alert log on SERVER for possible errors and correct them. The return code Code is the return code of the **`strmqm`** command.
119006	Command server start command for queue manager “TAG” failed on server “SERVER” with return code “Code”.	The start command for the command server returned with none zero value. Check the IBM WebSphere MQ alert log on SERVER for possible errors and correct them. The return code Code is the return code of the runmqsc command. For WebSphere MQ v6.0, verify that the command server startup type is “MANUAL”. See section Configuration Requirements for details.
119007	Listener for queue manager “TAG” failed to start on server “SERVER”.	Check the IBM WebSphere MQ alert log on SERVER for possible errors and correct them.

119008	Listener start command for queue manager with TAG “TAG” failed on server “SERVER” with return code “CODE”.	Check the IBM WebSphere MQalert log on SERVER for possible errors and correct them.
119013	Could not create queue manager object for queue manager “QMGR” with TAG “TAG”.	Check the LifeKeeper and WebSphere MQ error logs.
119014	Could not create listener object for queue manager “QMGR” with TAG “TAG”.	Check the LifeKeeper and WebSphere MQ error logs.
119015	No value forthe “PARAMETER” specified.	Run the LifeKeeper MQ Recovery Kit script with the correct arguments.
119016	Instance with ID “ID” does not exist on server “SERVER”.	Check the resource hierarchy.
119017	Instance with TAG “TAG” does not exist on server “SERVER”.	Check the resource hierarchy.
119018	Invalid parameters specified.	Run the script with the correct options.
119019	Too few parameters specified	Run the script with the correct options.
119021	Failed to set “VALUE” for resource instance “TAG” on server “SERVER”.	Check the LifeKeeper log for possible errors setting the value.
119025	Failed to update instance info for queue manager with TAG “TAG” on server “SERVER”.	When the server is up and running again, retry the operation to synchronize the settings.
119026	The following program required does not exist or is not executable: “EXECUTABLE”. Check failed.	The program EXECUTABLE cannot be found. Verify all installation requirements are met and install all required packages. See section Configuration Requirements for details.
119032	Script: usage error (error message)	Start the script Script with the correct arguments
119033	Script: error parsing config file “ConfigFile”.	Make sure ConfigFile exists and is readable.
119034	CHECKTYPE check for queue manager with TAG “TAG” failed on server “SERVER” because the MQUSER could not be determined. This is probably because of a removed configuration file – ignoring.	The CHECKTYPE check for queue manager with tag TAG failed. Make sure the global configuration file (mqs.ini) exists and is readable. If it is removed, recreate the mqs.ini configuration file.
119035	CHECKTYPE check for queue manager with TAG “TAG” failed on server “SERVER” because no TCP PORT directive found in config file “CONFIGFILE” – ignoring.	Make sure the queue manager configuration file (qm.ini) exists and contains a TCP section as required during installation. Add the TCP section to the queue manager configuration file.
119042	“CHECKTYPE” check for queue manager with TAG “TAG” failed on server “SERVER” because no TCP PORT information was found via runmqsc.	Verify that the port information for the listener objects has been defined and is accessible via runmqsc.

119043	TCP Listener configuration could not be read, reason: “REASON”.	Verify that MQ is running and the port information for the listener objects has been defined and is accessible via runmqsc.
119044	No TCP Listener configured, no TCP PORT information was found via runmqsc: “MESSAGE”.	Verify that the port information for the listener objects has been defined and is accessible via runmqsc.

Create

Error Number	Error Message	Action
001022	END failed hierarchy “CREATE” of resource “TAG” on server “SERVER” with return value of “VALUE”.	Check the LifeKeeper log on server “SERVER” for possible errors creating the resource hierarchy. The failure is probably associated with the queue manager not starting.
119020	Create MQSeries queue manager resource with TAG “TAG” for queue manager “QMGR” failed.	Check the LifeKeeper log for possible errors creating the resource. The failure is probably associated with the queue manager not starting.
119022	Failed to create dependency between “PARENT” and “CHILD”.	Check the LifeKeeper log for possible errors creating the dependency.
119023	Creating the filesystem hierarchies for queue manager with TAG “TAG” failed. File systems: “Filesystems”.	Check the LifeKeeper log for possible errors creating the filesystem hierarchies.
119029	No TCP section configured in “CONFIGFILE” on server “SERVER”.	Add the TCP section to the queue manager configuration file on server SERVER. See section Configuration Requirements for details.
119031	Queue manager “DIRTYPE” directory (“DIRECTORY”) not on shared storage.	Move the directory DIRECTORY to shared storage and retry the operation.
119038	Creation of queue manager resource with TAG “TAG” failed on server “SERVER”.	Check the LifeKeeper log on server SERVER for possible errors, correct them and retry the operation.
119039	TCP section in configuration file “FILE” on line “LINE1” is located before LOG section on line “LINE2” on server “SERVER”.	It’s recommended for the TCP section to be located after the LOG: section in the queue manager configuration file. Move the TCP section to the end of the queue manager configuration file and retry the operation.
119040	Creation of MQSeries queue manager resource by create_ins was successful but no resource with TAG “TAG” exists on server “SERVER”. Sanity check failed.	Check the LifeKeeper log for possible errors during resource creation.
119041	Creation of MQSeries queue manager resource was successful but no resource with TAG “TAG” exists on server “SERVER”. Final	Check the LifeKeeper log for possible errors during resource creation.

	sanity check failed.	

Extend

Error Number	Error Message	Action
119024	Instance “TAG” can not be extended from “TEMPLATESYS” to “TARGETSYS”. Reason:REASON	Correct the failure described in REASON and retry the operation.
119027	The user “USER” does not exist on server “SERVER”.	Create the user USER on SERVER with the same UID as on the primary server and retry the operation.
119028	The user “USER” has a different numeric UID on server “SERVER1” (SERVER1UID) then it should be (SERVER2UID).	Change the UID so that USER has the same UID on all servers and reinstall WebSphere MQ on the server where you have changed the UID and retry the operation.
119029	No TCP section configured in “CONFIGFILE” on server “SERVER”.	Add the TCP section to the queue manager configuration file on server SERVER. See section Configuration Requirements for details.
119030	Queue manager “QMGR” not configured in “CONFIGFILE” on server “SERVER”.	The queue manager QMGR you are trying to extend is not configured in the global configuration file on the target server SERVER. Add the queue manager stanza to the config file CONFIGFILE on server SERVER and retry the operation.
119036	Link “LINK” points to “LINKTARGET” but should point to “REALTARGET” on server “SERVER”.	For file system layout 3 symbolic links must point to the same location on the template and target server SERVER. Correct the link LINK on server SERVER to point to REALTARGET and retry the operation.
119037	Link “LINK” that should point to “REALTARGET” does not exist on system “SERVER”.	For file system layout 3 symbolic links must also exist on the target server. Create the required link LINK to REALTARGET on server SERVER and retry the operation.

Remove

Error Number	Error Message	Action

119003	Failed to stop queue manager with TAG “TAG” on server “SERVER”.	The queue manager “TAG” on server “SERVER” could not be stopped through the Recovery Kit. For further information and investigation, change the logging level to DEBUG. Depending on the machine load the shutdown timeout values possibly have to be increased.
119004	Some orphans of queue manager with TAG “TAG” could not be stopped on server “SERVER”. Tried it “tries” times.	Try killing the orphans manually and restart the Queue Manager again. For further information change the logging level to DEBUG.
119010	Listener for queue manager with TAG “TAG” failed to stop on server “SERVER”.	This message will only appear if the monitoring for the listener is enabled. For further information change the logging level to DEBUG.

Resource Monitoring

Error Number	Error Message	Action
119005	Queue manager with TAG “TAG” on server “SERVER” failed.	Check the IBM WebSphere MQ alert log on SERVER for possible errors. This message indicates a queue manager crash
119009	Listener for queue manager with TAG “TAG” failed on server “SERVER”.	This message will only appear if monitoring of the listener is enabled. For further information change the logging level to FINE.
119011	“CHECKTYPE” PUT/GET Test for queue manager with TAG “TAG” failed on server “SERVER” with return code “Code”	This message will only appear if the PUT/GET Test is enabled and the test queue exists. For further information change the logging level to FINE and check theIBM WebSphere queue manager error log (/var/mqm/errors) on SERVER for possible errors and correct them. Verify that the file systems are not full.
119012	Client connect test for queue manager with TAG “TAG” on server “SERVER” failed with return code “Code”.	This message will only appear if Listener management is enabled. This message indicates a problem with the listener or the queue manager. Check the log for possible errors and correct them. The return code Code is the return code of the amqscnxc command.

Warning Messages

Error Number	Error Message	Action
119201	Listener for queue manager with TAG “TAG” is NOT monitored on server “SERVER”.	This is a warning that listener management is not enabled.

119202	Queue manager with TAG “TAG” is not running on server “SERVER” but some orphans are still active. This is attempt number “ATTEMPT” at stopping all orphans processes.	This is a warning that MQ was not stopped properly.
119203	Another instance of recover is running, exiting “EXITCODE”.	Recovery was started, but another recovery process was already running, so this process will not continue.
119204	Queue manager server connect check for queue manager with TAG “TAG%” timed out after “SECONDS” seconds on server “SERVER”.	If you see this message regulary increase the value of MQS_QUICKCHECK_TIMEOUT_SC in /etc/defaul/LifeKeeper. See section Changing the Server Connection Channel for details.
119205	Queue manager client connect check for queue manager with TAG “TAG” timed out after “SECONDS” seconds on server “SERVER”.	If you see this message regulary increase the value of MQS_QUICKCHECK_TIMEOUT_CC in /etc/defaul/LifeKeeper. See section Changing the Server Connection Channel for details.
119206	Server “SERVER” is not available, skipping.	A server was not online while updating a queue manager configuration setting. Wait for the server to be online again and repeat the configuration step.
119207	“CHECKTYPE” PUT/GET test for queue manager with TAG “TAG” failed because test queue “QUEUE” does not exist (reason code “REASONCODE”) – ignoring.	Create the test queue QUEUE configured or reconfigure the test queue to an existing queue. See section Configuration Requirements for details on creating the test queue.
119208	Channel “CHANNEL” does not exist for queue manager with TAG “TAG” (reason code “REASONCODE”) – ignoring.	Create the channel “CHANNEL” which does not appear to exist. By default the channel SYSTEM.DEF.SVRCONN is used. See the WebSphere MQ documentation for details on how to create channels.
119209	PUT/GET test for queue manager with	Configure a LifeKeeper test queue for queue

	TAG “TAG” skipped because no test queue is defined.	manager TAG,
19210	The following program required to perform the PUT/GET test does not exist or is not executable: “EXECUTBALE”. Test skipped.	Install a C compiler on the system and make sure it is in the root users PATH environment variable. Run the script “LKROOT/lkadm/subsys/appsuite/mqseries/bin/compilesamples” to compile the modified sample amqsget and amqsgetc programs.
119211	Queue manager “CHECKTYPE” PUT/GET test for queue manager with TAG “TAG” timed out after “SECONDS” seconds on server “SERVER”.	If you see this message regulary increase the value of MQS_QUICKCHECK_TIMEOUT_PUTGET in /etc/default/LifeKeeper. See section Changing the Server Connection Channel for details.
119212	QuickCheck for queue manager with TAG “TAG” timed out after SECONDS seconds on server “SERVER”.	If you get this message regularly increase the value of MQS_QUICKCHECK_TIMEOUT in /etc/default/LifeKeeper
119213	mqseriesQueueManager::getMQVersion:: ERROR unexpected dspmqver output (OUTPUT) – using installation info instead (Queue QUEUE, Queuemanager QMGR).	Reading the MQ version failed via runmqsc. If you get this message regularly, increase the value of MQS_DSPMQVER_TIMEOUT in /etc/default/LifeKeeper.
119214	mqseriesQueueManager::getMQVersion:: ERROR unexpected output retrieving MQ version information (Queue QUEUE, Queuemanager QMGR). Unexpected results *	Check if the following command yields some output when running as the mqm user: dspmqver –b –p1 –f2. Also, as the mqm user run the command dspmqinst and check what it returns.

6.8.7. Appendix A – Sample mqs.ini Configuration File

```
#********************************************************************#

#*

*#

#*

<START_COPYRIGHT>

*#

#* Licensed Materials - Property of
IBM                                          *#

#*

63H9336

*#

#* © Copyright IBM Corporation 1994,
 2000                                       *#

#*

*#

#*
 <END_COPYRIGHT>
```

#**#
#* Module Name:
mqs.ini *#
#* Type : WebSphere MQ Machine-wide Configuration
File *#
#* Function : Define WebSphere MQ resources for an entire
machine *#
#**#
#* Notes
*#
#* 1) This is the installation time default
configuration *#

*#
#**#
AllQueueManagers:

#**# #* The path to the qmgrs directory, below

which queue
manager data *# #* is
stored
*#
#***# DefaultPrefix=/var/mqm

LogDefaults: LogPrimaryFiles=3 LogSecondaryFiles=2 LogFilePages=1024 LogType=CIRCULAR
LogBufferPages=0 LogDefaultPath=/var/mqm/log
QueueManager: Name=TEST.QM Prefix=/var/mqm Directory=TEST!QM
DefaultQueueManager: Name=TEST.QM
QueueManager: Name=TEST.QM.NEW Prefix=/var/mqm Directory=TEST!QM!NEW
QueueManager: Name=TEST.QM2 Prefix=/var/mqm Directory=TEST!QM2

QueueManager: Name=MULTIINS_1 Prefix=/var/mqm Directory=MULTIINS_1 DataPath=/opt/webmq/
MULTIINS_1/data

InstallationName=Installation1

QueueManager: Name=MULTIINS_2 Prefix=/var/mqm Directory=MULTIINS_2 DataPath=/opt/webmq/
MULTIINS_2/data InstallationName=Installation2

6.8.8. Appendix B – Sample qm.ini Configuration File

```
#********************************************************************#
#* Module Name: qm.ini                                              *#
#* Type       : WebSphere MQ queue manager configuration file       *#
#  Function   : Define the configuration of a single queue manager  *#
#*                                                                  *#
#********************************************************************#
#* Notes      :                                                     *#
#* 1) This file defines the configuration of the queue manager
*#
#********************************************************************#
ExitPath:
   ExitsDefaultPath=/var/mqm/exits/
   ExitsDefaultPath64=/var/mqm/exits64
#*                                                                  *#
#*                                                                  *#
Log:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=1024
   LogType=CIRCULAR
   LogBufferPages=0
   LogPath=/opt/MQ_log/MULTIINS_1
   LogWriteIntegrity=TripleWrite
Service:
   Name=AuthorizationService
   EntryPoints=14
ServiceComponent:
   Service=AuthorizationService
   Name=MQSeries.UNIX.auth.service
   Module=amqzfu
   ComponentDataSize=0
```

6.8.9. Appendix C – WebSphere MQ Configuration Sheet

Cluster name		
Contact information (email or telephone number of person responsible for the cluster)		
LifeKeeper version		
Operating system		
Cluster nodes	name	public IP / netmask

Queue manager name		
Listener management by LifeKeeper	[ ] YES [ ] NO	
WebSphere MQ operating system user	name	numeric (UID/GID)
user (e.g. mqm/1002)		
group (e.g. mqm/200)		
Virtual IP / netmask / network device (eg. 192.168.1.1/24/eth0)		
Filesystem layout	__ Configuration 1 – /var/mqm on Shared Storage	
	__ Configuration 2 – Direct Mounts	
	__ Configuration 3 – Symbolic Links __ Configuration 4 – Multi-Instance Queue Managers	

	__ other
Shared storage type	__ NAS (IP: _____)
	__ SCSI/FC (Type: _____)
	__ SDR
Queue manager `/var/mqm/qmgrs/QM.NAME` physical location (device, mount point or logical volume) (e.g. `LVM /dev/mqm_test_qm/qmgrs`)	
Queue manager `/var/mqm/log/QM.NAME` physical location (device, mount point or logical volume) (e.g. `LVM /dev/mqm_test_qm/log`)	

6.9. NAS Recovery Kit Administration Guide

The LifeKeeper for Linux Network Attached Storage Recovery Kit (hereafter referred to as the NAS Recovery Kit) provides fault resilience for Network File System (NFS) software in a LifeKeeper environment. The NAS Recovery Kit affords LifeKeeper users the opportunity to employ an exported NFS file system as the storage basis for LifeKeeper hierarchies.

Document Contents

This guide contain the following topics:

- Documentation and References. Provides a list of LifeKeeper for Linux documentation and where to find them.
- Requirements. A description of the hardware and software necessary to properly setup, install, and operate the NAS Recovery Kit. Refer to SIOS Protection Suite Installation Guide for specific instructions on how to install or remove LifeKeeper for Linux software.NAS Recovery Kit .
- Overview. A description of the NAS Recovery Kit's features and functionality.
- Configuring the LifeKeeper for Linux NAS Recovery Kit. A description of the procedures required to properly configure the NAS Recovery Kit.
- LifeKeeper Configuration Tasks. A description of the tasks for creating and managing your NAS resource hierarchies using the LifeKeeper GUI.
- Troubleshooting. A list of LifeKeeper for Linux error messages including a description for each.

6.9.1. NAS Documentation and References

The following LifeKeeper product documentation is available from SIOS Technology Corp.:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- SIOS Protection Suite Installation Guide
- Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is available on the SIOS Technology Corp. website at:

http://docs.us.sios.com

6.9.2. NAS Recovery Kit Hardware and Software Requirements

Requirements

Your LifeKeeper configuration must meet the following requirements **prior** to the installation of the LifeKeeper for Linux NAS Recovery Kit. Please see SIOS Protection Suite Installation Guide for specific instructions regarding the configuration of your LifeKeeper hardware and software.

Hardware Requirements

- **Servers** – LifeKeeper for Linux supported servers configured in accordance with the requirements described in SIOS Protection Suite Installation Guide and SPS for Linux Release Notes.

- **IP Network Interface Cards** – Each server requires at least one Ethernet TCP/IP-supported network interface card. Remember, however, that a LifeKeeper cluster requires two communications paths; two separate LAN-based communication paths using dual independent sub-nets are recommended for heartbeats, and at least one of these should be configured as a private network. Using a combination of TCP and TTY heartbeats is also supported.

Software Requirements

- **TCP/IP software** – Each server in your LifeKeeper configuration requires TCP/IP software.

- **LifeKeeper Software** – It is imperative that you install the same version of the LifeKeeper for Linux software and apply the same versions of the LifeKeeper for Linux software patches to each server in your cluster.

- **LifeKeeper for Linux NAS Recovery Kit** – The NAS Recovery Kit is provided on the SPS Installation Image File (sps.img). It is packaged, installed and removed via the Red Hat Package Manager, rpm. The following rpm file is supplied on the SPS Installation Image File (sps.img):

  steeleye-lkNAS

- **Linux software** – Each server in your cluster must have the **util-linux** package installed and configured prior to configuring LifeKeeper and the LifeKeeper NAS Recovery Kit. The NAS Recovery Kit requires version 2.9u or later of the **util-linux** package to assure proper functionality.

  Please see the SIOS Protection Suite Installation Guide for specific instructions on the installation and removal of the LifeKeeper for Linux software.

6.9.3. NAS Recovery Kit Overview

LifeKeeper for Linux NAS Recovery Kit

The primary focus of the LifeKeeper for Linux NAS Recovery Kit is to offer LifeKeeper users an alternative storage method to shared storage and data replication.

The NAS Recovery Kit enables the creation of LifeKeeper resource hierarchies on LifeKeeper protected servers or clients that have imported (mounted) an exported Network File System (NFS) from either a Network Attached Storage device or an NFS server in the cluster. When a failure is detected on the node in the cluster where the exported file system is mounted, the NAS Recovery Kit initiates a fail over to the predetermined backup node.

Therefore, once the exported file system is mounted on a LifeKeeper server or client, it can be fully utilized as an additional storage basis for LifeKeeper hierarchies.

When you elect to use an exported file system as a storage medium, LifeKeeper does not require you to protect the server where the file system is exported. However, to achieve a greater degree of availability, users are encouraged to use the LifeKeeper for Linux NFS Server Recovery Kit to protect the server from failure where the file system is exported.

Resource hierarchies for the NAS Recovery Kit are created using the currently existing File System Recovery Kit available with the LifeKeeper Core product (**steeleye-lk** package).

While the NAS Recovery Kit delivers several advantages, the two most significant advantages are the elimination of the need for costly shared-storage devices and the capability to have multi-node cluster configurations.

NAS Recovery Kit Restrictions

- This version of the NAS Recovery Kit does not include support for a local recovery when access to the NAS device fails. When a failure is detected, the default action is to initiate a transfer of the hierarchy to a backup server. Depending on the makeup of the resource hierarchy, this action can result in hung processes. To avoid hung processes, the default action can be changed to halt the server and force a failover to a backup server. To change the default switchover behavior, alter the setting of LKNASERROR in the LifeKeeper defaults file. See the section **Configuring the NAS Recovery Kit** later in this document for more discussion on LKNASERROR.

- The NAS Recovery Kit does not provide protection for your Network Attached Storage device. The objective of this kit is to expand LifeKeeper storage options into the Network Attached Storage arena.

- The NAS Recovery Kit does not permit the NFS file system to be mounted more than once on different mount points. Attempts to create hierarchies when the file system is found in the */etc/fstab* file multiple times will fail.

- File systems to be protected by the NAS Recovery Kit should be mounted using the IP address rather than the host name (for example, 100.99.100.9/dir instead of server1/dir). This will avoid potential DNS or host file lookup problems. Mounting via host name will result in a “bad mount” being detected, after which LifeKeeper will unmount and re-mount the file system using the IP address. The unmount process could kill processes that are currently using the mount point.

6.9.4. Configuring the LifeKeeper for Linux NAS Recovery Kit

This section describes the LifeKeeper for Linux NAS Recovery Kit configuration details. It also contains information you should consider before you start to configure and administer the NAS Recovery Kit. Please refer to SPS for Linux Technical Documentation for instructions on configuring LifeKeeper Core resource hierarchies.

6.9.4.1. NAS Configuration Considerations

The following should be considered before operating the LifeKeeper for Linux NAS Recovery Kit:

1. Install the NAS Recovery Kit on the server(s) in your cluster configuration where you wish to mount your exported file systems and where you will extend your NAS resource hierarchy. You can export your file system from either a NFS server, which may be protected by LifeKeeper (this is the recommended configuration), or from a Network Attached Storage device.

2. To ensure proper execution of this kit, it is highly recommended that you mount your exported NFS file system using the server's IP address in place of the server name and that you perform your mount operation before you place your file system under LifeKeeper protection. Additionally, if you are mounting a file system that is currently protected by the LifeKeeper for Linux NFS Server Recovery Kit, we strongly suggest that the IP address used to create the NFS Server hierarchy be used to mount the file system on the LifeKeeper NAS server. Use the NFS mount option **"intr"** to ensure that LifeKeeper can interrupt operations being performed on the file system. Failure to use this option can result in a LifeKeeper failure.

3. To eliminate the possibility of split-brain related problems (i.e. more than one node in the cluster has a hierarchy In Service Protected (ISP)), we highly recommend that you establish one of the communication paths between nodes in the cluster on the same network used to access the exported file system. Failure to comply with this recommendation can result in multiple nodes bringing the hierarchy ISP (split-brain) when a communication path failure occurs. To recover from a split-brain scenario, take all but one of the ISP hierarchies out of service. This will ensure that only one node has access to the exported file system.

4. The built-in file system recovery kit used to build NAS hierarchies cannot detect and remove processes not protected by LifeKeeper that are using the mounted file system in a fail over condition. Therefore, it is highly recommended that only LifeKeeper protected processes use the NAS protected file system.

5. The LKNFSTIMEOUT tunable represents the timeout in seconds the NAS Recovery Kit will use when attempting to determine the status of a NFS mounted file system. The default value for this tunable is set to 2 minutes. The LKNFSSYSCALLTO tunable represents the timeout in seconds the NAS Recovery Kit will use for alarms to interrupt system calls when attempting to determine the status of a mount point. Use the formula below to determine the value for this tunable:

   3 times your LKNFSSYSCALLTO value plus 5 should be less than the value of LKNFSTIMEOUT.

6. The LKNASERROR tunable controls the actions the NAS Recovery kit takes when access to the NAS device fails. The tunable has two values, **switch** and **halt**, with **switch** being the default. If the value is set to switch and access fails, the NAS Recovery Kit will initiate a transfer of the resource hierarchy to a backup server when the failure is detected. The attempt to transfer the resource hierarchy to the backup server can hang if any of the resources sitting above the NAS resource attempt to access anything on the NAS file system. To avoid this problem the tunable

value can be set to **halt**, which will immediately halt the system when an access failure is detected. This action will force a failover of all resource hierarchies to the backup server.

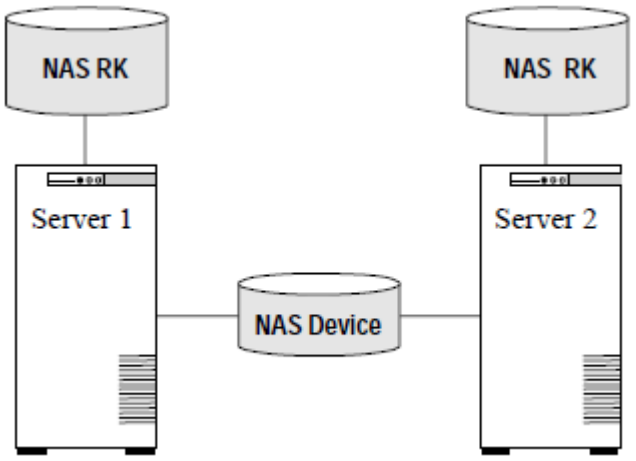
7. STONITH devices or the Quorum/Witness package should be used so that a machine failure (all comm paths are down) does not result in a split brain where all the NAS resources are in service on all nodes in the cluster. This condition can lead to data corruption. More details on the Quorum/Witness package can be found in the SIOS Protection Suite Technical Documentation.

6.9.4.2. NAS Configuration Examples

Configuration Examples

A few examples of what happens during a fail over using LifeKeeper for Linux NAS Recovery Kit are provided below.

Configuration 1: Active/Standby Configuration Example



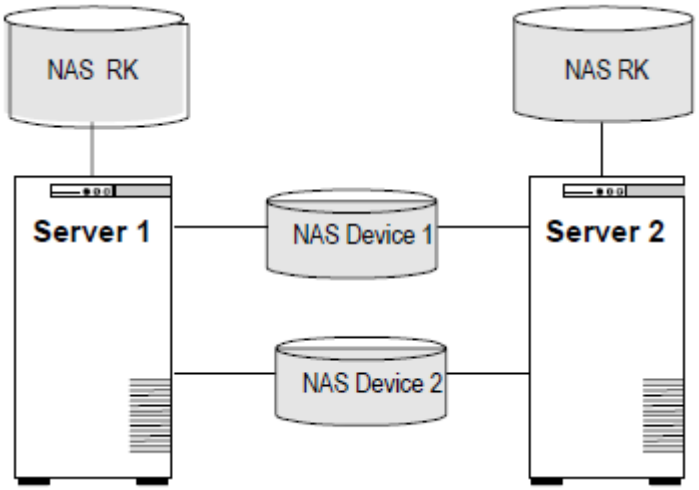
In this configuration, Server 1 is considered active because it is running the NAS Recovery Kit software and has imported (mounted) the file system from the NAS device. Server 2 does other processing. If Server 1 fails, Server 2 gains access to the file system and uses the LifeKeeper secondary hierarchy to make it available to clients.

Configuration Notes:

- The NAS software must be installed on both servers.
- The file system has been imported from a NAS device.
- Server 2 should not access files and directories on the NAS device while Server 1 is active.

Note: In an active/standby configuration, Server 2 might be running the NAS Recovery Kit, but does not have any other NAS resources under LifeKeeper protection.

Configuration 2: Active/Active Configuration Example



An active/active configuration consists of two or more systems actively running the NAS Recovery Kit software and importing file systems from NAS device(s).

Configuration Notes:

- The NAS software must be installed on both servers.
- Initially, Server 1 imports a file system and Server 2 imports a different file system. In a switchover situation, one system can import both file systems.

6.9.5. LifeKeeper Configuration Tasks for NAS

You can perform all LifeKeeper for Linux NAS Recovery Kit administrative tasks via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer, and monitor NAS resources.

The following tasks are available for configuring the LifeKeeper for Linux NAS Recovery Kit:

- Create a Resource Hierarchy – Creates a NAS resource hierarchy.
- Delete a Resource Hierarchy – Deletes a NAS resource hierarchy.
- Extend a Resource Hierarchy – Extends a NAS resource hierarchy from the primary server to the backup server.
- Unextend a Resource Hierarchy – Unextends (removes) a NAS resource hierarchy from a single server in the LifeKeeper cluster.
- Create Dependency – Creates a child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete Dependency – Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service – Activates a resource hierarchy.
- Out of Service – Deactivates a resource hierarchy.
- View / Edit Properties – View or edit the properties of a resource hierarchy.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You can also perform most of these tasks:

1. From the toolbar, right – click on a global resource in the left pane of the status display.
2. Right – click on a resource instance in the right pane of the status display.

Note: Using the right-click method allows you to avoid entering information that is required when using the **Edit** menu.

6.9.5.1. Creating a NAS Resource Hierarchy

Perform the following on your primary server:

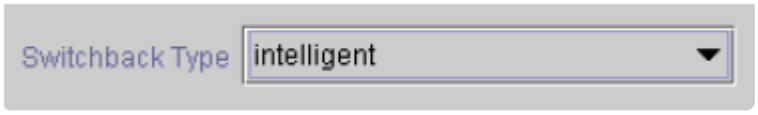
1. Select **Edit** > **Server** > **Create Resource Hierarchy**.

2. The **Select Recovery Kit** dialog appears. Select the **File System** option from the drop down list. Simply put, a NAS Resource Hierarchy is a File System Hierarchy created using a NFS mounted file system.

   Click **Next** to continue.

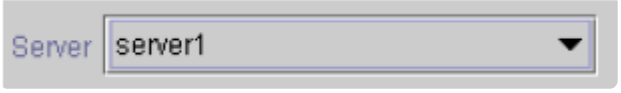
> **CAUTION:** If you click the **Cancel** button at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

3. The **Switchback Type** dialog appears. The switchback type determines how the NAS resource will be switched back to the primary server when it becomes in-service (active) on the backup server following a failover. Switchback types are either *intelligent or automatic*. Intelligent switchback requires administrative intervention to switch the resource back to the primary server while automatic switchback occurs as soon as the primary server is back on line and reestablishes LifeKeeper communication paths.



   Click **Next** to continue.

4. The **Server** dialog appears. Select the name of the server where the NAS resource will be created (typically this is your primary server). All servers in your cluster are included in the drop down list box.



   Click **Next** to continue.

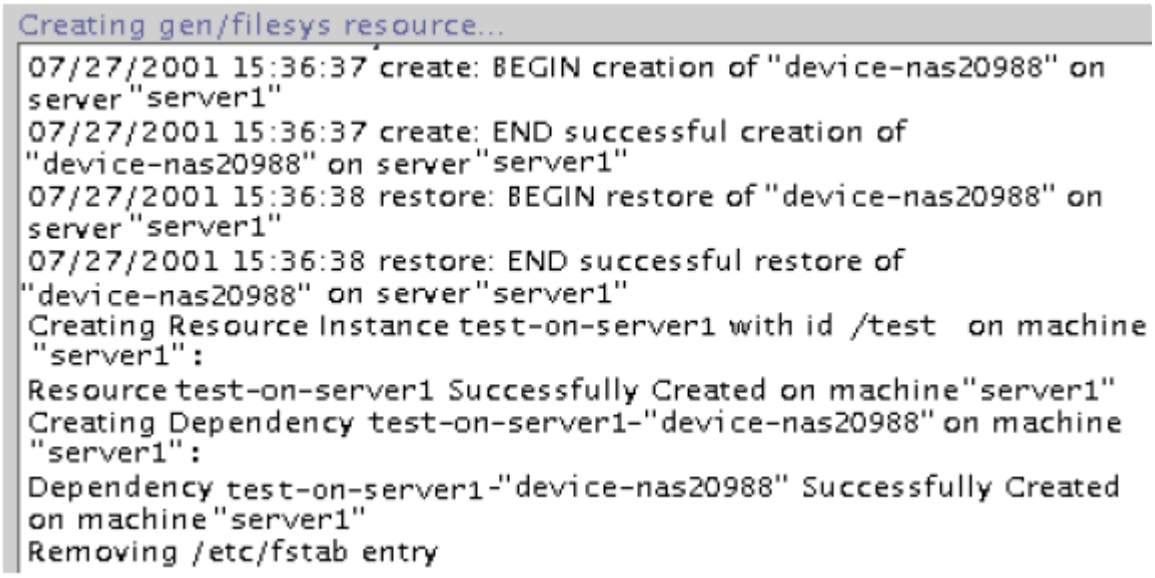
5. Select the **Mount Point** path to be protected by the NAS (File System) Resource Hierarchy. All “local” (i.e. file systems using shared storage) and NFS mounted file systems are listed. Select the NFS mounted file system from the drop down list box.

Click **Next** to continue.

5. The **Root Tag** dialog is automatically populated with a unique name for the resource instance on the target server (i.e. the server selected above). You may accept the default or enter a unique tag consisting of letters, numbers and the following special characters: -,_, ., or /.

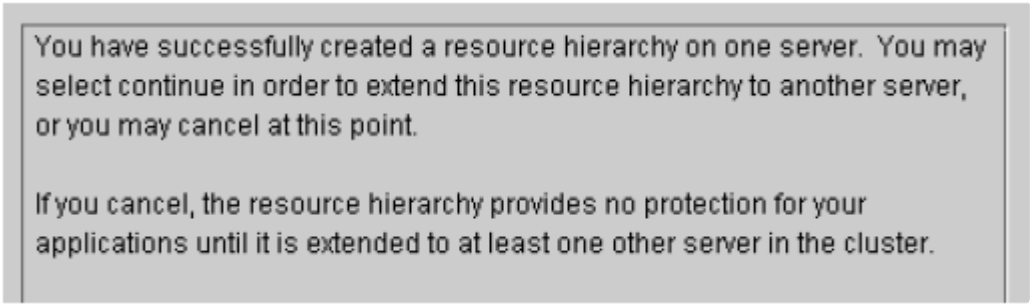
Click **Create Instance**.

7. An information box appears indicating the start of the hierarchy creation.



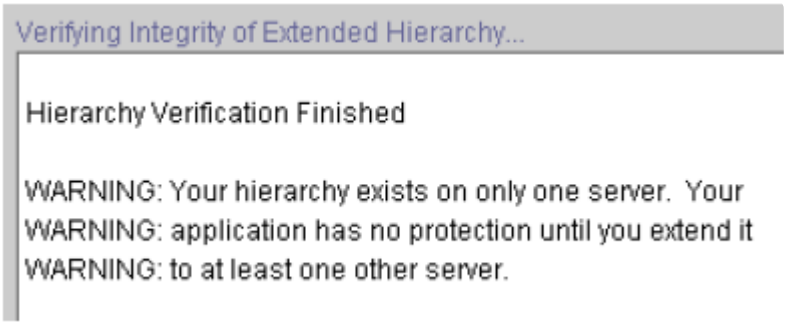
Click **Create** to continue.

8. An information box appears announcing the successful creation of your NAS resource hierarchy. You must **Extend** the hierarchy to another server in your cluster in order to place it under LifeKeeper protection.



Click **Continue** to extend the resource.

Click **Cancel** if you wish to extend your resource at another time.



9. Click **Done** to exit the Create Resource Hierarchy menu selection.

6.9.5.2. Deleting a NAS Resource Hierarchy

To delete a NAS resource from all servers in your LifeKeeper configuration, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Delete Resource Hierarchy**.

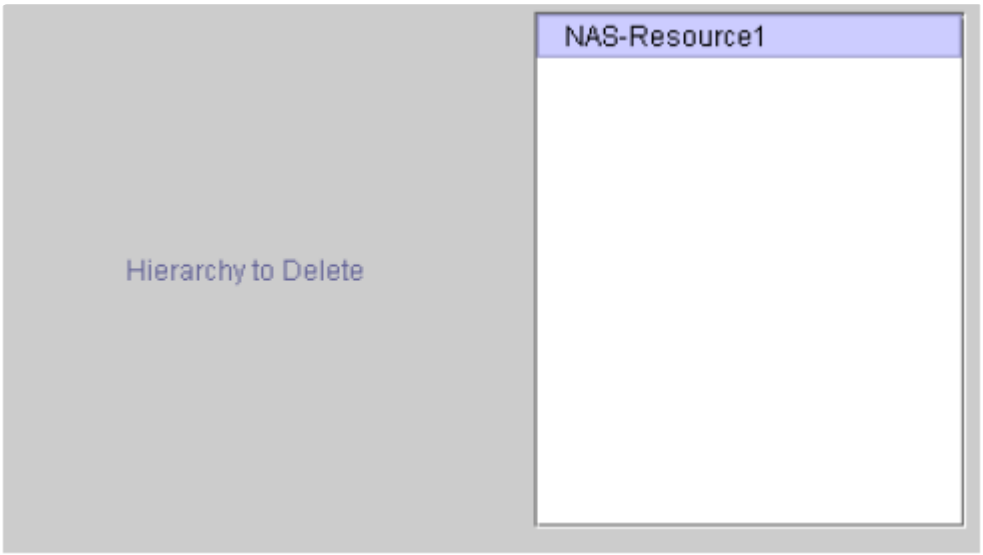
2. Select the name of the **Target Server** where you will be deleting your NAS resource hierarchy.

> **Note**:If you selected the Delete Resource task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

Click **Next** to continue.

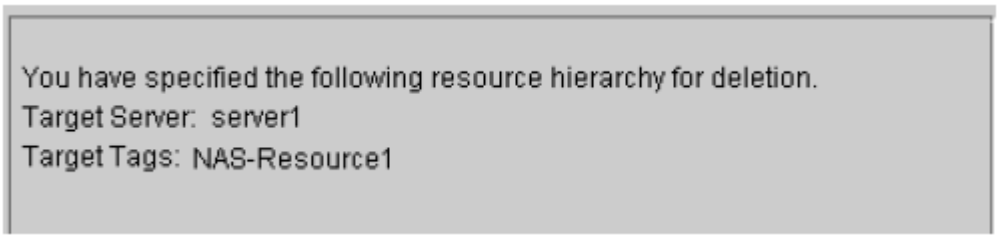
3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete, and highlight it.

> **Note**:If you selected the Delete Resource task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.



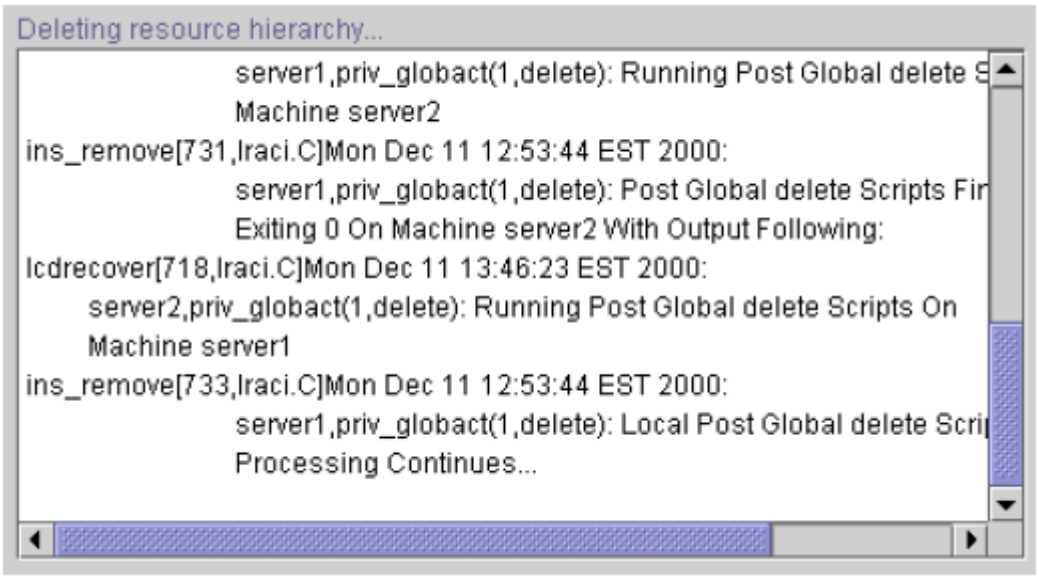
Click **Next** to continue.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete.



Click **Delete** to continue.

5. An information box appears confirming that the NAS resource instance was deleted successfully.



6. Click **Done** to exit the Delete Resource Hierarchy menu selection.

6.9.5.3. Extending Your NAS Hierarchy

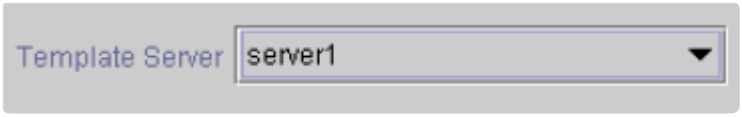
After you have created a hierarchy, you should extend that hierarchy to another server in the cluster. There are three possible ways to extend your resource instance:

1. When you successfully create your NAS resource hierarchy you will have an opportunity to select **Continue** which will allow you to proceed with extending your resource hierarchy to your backup server.

2. Right-click on an unextended hierarchy in either the left or right pane on the LifeKeeper GUI.

3. Select the **Extend Resource Hierarchy** task from the LifeKeeper GUI by selecting **Edit, Resource, Extend Resource Hierarchy** from the drop down menu. This sequence of selections will launch the Extend Resource Hierarchy wizard. The **Accept Defaults** button that is available for the **Extend Resource Hierarchy** option is intended for the user who is familiar with the LifeKeeper Extend Resource Hierarchy defaults and wants to quickly extend a LifeKeeper resource hierarchy without being prompted for input or confirmation. Users who prefer to extend a LifeKeeper resource hierarchy using the interactive, step-by- step interface of the GUI dialogs should use the **Next** button.
    -
    - The first dialog box to appear will ask you to select the **Template Server** where your NAS resource hierarchy is currently in service. Remember that the **Template Server** you select now and the **Tag to Extend** that you select in the next dialog box represent an in- service (activated) resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you have selected. The drop down box in this dialog provides the names of all the servers in your cluster.

Note: If you are entering the Extend Resource Hierarchy task by continuing from the creation of a NAS resource hierarchy, this dialog box will not appear because the wizard has already identified the template server in the create stage. This is also the case when you right-click on either the NAS resource icon in the left pane or right-click on the NAS resource box in the right pane of the GUI window and choose Extend Resource Hierarchy.



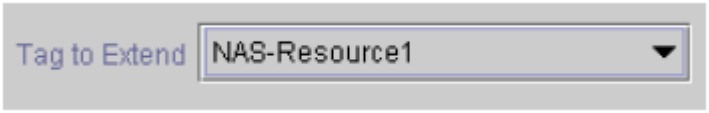
CAUTION: If you click the **Cancel** button at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extend hierarchy process. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

Click **Next** to continue.

Select the **Tag to Extend**. This is the name of the NAS instance you wish to extend from the template server to the target server. The wizard will list in the drop down box all of the resources that you have created on the template server.

Note: Once again, if you are entering the Extend Resource Hierarchy task immediately following the

creation of a NAS hierarchy, this dialog box will not appear because the wizard has already identified the tag name of your resource in the create stage. This is also the case when you right-click on either the NAS resource icon in the left pane or on the NAS resource box in the right pane of the GUI window and choose *Extend Resource Hierarchy*.

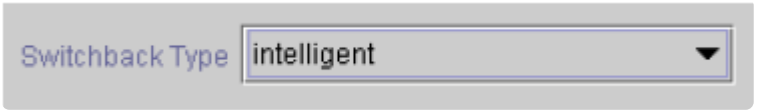


Click **Next** to continue.

Select the **Target Server** where you will extend your NAS resource hierarchy.

Click **Next** to continue.

The **Switchback Type** dialog appears. The switchback type determines how the NAS resource will be switched back to the primary server when it becomes in service (active) on the backup server following a failover. Switchback types are either intelligent or automatic. Intelligent switchback requires administrative intervention to switch the resource back to the primary server while automatic switchback occurs as soon as the primary server is back on line and reestablishes LifeKeeper communication paths.



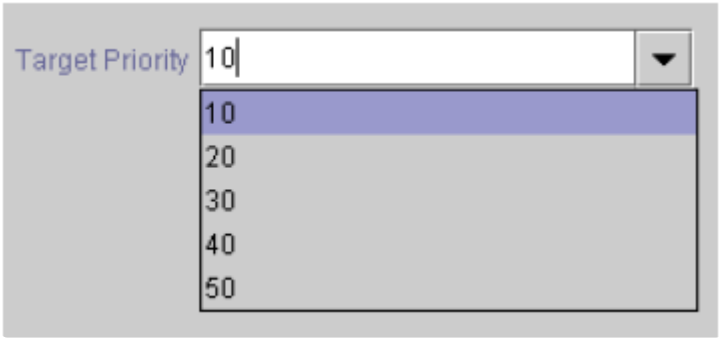
Click **Next** to continue.

Select or enter a **Template Priority**. This is the priority for the NAS hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended.

Note: This selection will appear only for the initial extend of the hierarchy.

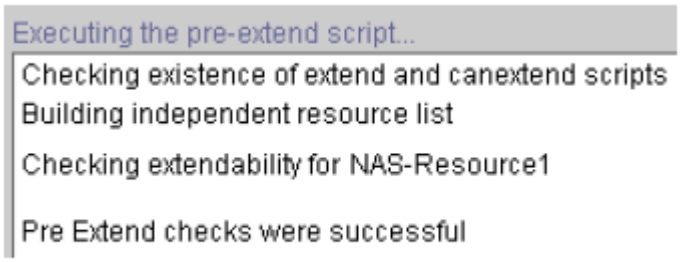
Click **Next** to continue.

Select or enter the **Target Priority**. This is the priority for the new extended NAS hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.



Click **Next** to continue.

An information box appears explaining that LifeKeeper has successfully checked your environment and that all requirements for extending this resource have been met. If there are requirements that have not been met, LifeKeeper will disable the **Next** button, and enable the **Back** button.



Click on the **Back** button to make changes to your resource extension.

Click **Cancel** to extend your resource another time.

Click **Next** to launch the Extend Resource Hierarchy configuration task.

Click **Finish** to confirm the successful extension of your NAS resource instance.

4. Click **Done** to exit the Extend Resources Hierarchy menu selection.

Note: Be sure to test the functionality of the new instance on both servers.

6.9.5.4. Unextending Your NAS Hierarchy

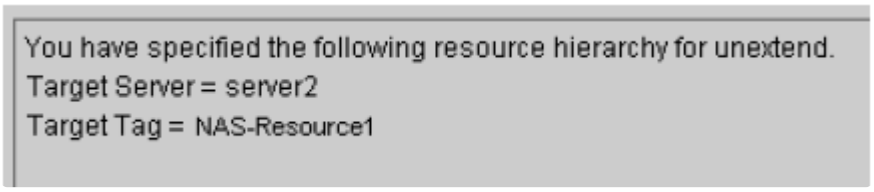
1. From the LifeKeeper GUI menu, select **Edit**, **Resource,** and **Unextend Resource Hierarchy**.
2. Select the **Target Server** where you want to unextend the NAS resource. It cannot be the server where the resource is currently in service (active).

   Note: If you selected the Unextend task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

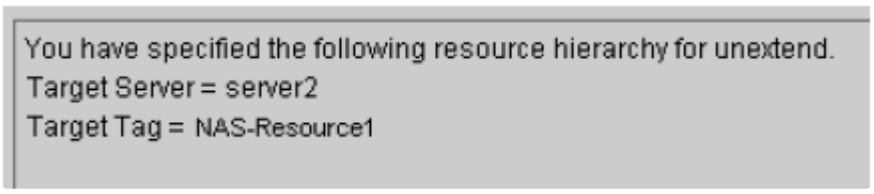
   

   Click **Next** to continue.
3. Select the NAS **Hierarchy to Unextend**.

   Note: If you selected the Unextend task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next** to continue.
4. An information box appears confirming the target server and the NAS resource hierarchy you have chosen to unextend.

   Click **Unextend**.
5. Another information box appears confirming that the NAS resource was unextended successfully.
6. Click **Done** to exit the Unextend Resource Hierarchy menu selection.

6.9.5.5. Testing Your NAS Resource Hierarchy

Testing Your Resource Hierarchy

You can test your NAS resource hierarchy by initiating a manual switchover that will simulate a fail over of the resource instance from the primary server to the backup server.

Performing a Manual Switchover from the LifeKeeper GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, **Resource**, and **In Service**. For example, an in-service request executed on a backup server causes the NAS resource hierarchy to be placed in-service on the backup server and taken out-of-service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out of Service** request, the resource hierarchy is taken out-of-service without bringing it in-service on the other server.

6.9.6. NAS Troubleshooting

Symptom	Possible Cause
LifeKeeper fail over operation fails with umount busy error.	The file system kit used to build NAS hierarchies cannot detect and remove processes not protected by LifeKeeper that are using the mounted file system in a fail over condition. Therefore, it is highly recommended that only LifeKeeper protected processes use the NAS protected file system. In the event of this failure, you must identify the processes using the file system and kill them. The **fuser -m** command can be used to determine the processes currently accessing the file system. Please see the **fuser** man pages for details on its use.
LifeKeeper does local recovery of file system mounted via server name.	If a file system protected by the NAS Recovery Kit was mounted via host name rather than IP address, then after creating the NAS resource, LifeKeeper logs a message similar to the following: `. . . WARNING: Mon Aug 26 11:27:01 2002:` LifeKeeper protected filesystem resource “tmp/mnt-on-tom.brown.com” (/tmp/mnt) is in service but not mounted . . . Attempting Local Recovery of resource LifeKeeper will re-mount the file system using the IP address at this point. However, if it encounters a problem, LifeKeeper will failover the NAS resource to the backup server (if it has been extended), or take the resource out of service (if the resource has not been extended). **Suggested Action:** If the local recovery is successful, no further action is needed. However, if the local recovery fails, you should: 1. Delete the NAS resource in LifeKeeper. 2. Re-mount the file system via IP address rather than host name. 3. Re-create the NAS resource.

6.9.6.1. NAS Error Messages

Error Messages

This section provides a list of messages that you may encounter while creating and extending an SPS NAS resource hierarchy or removing and restoring a resource. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Messages from other SPS components are also possible. In these cases, please refer to the Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

NAS Recovery Kit Error Messages

Error Number	Error Message
107001	Creation of NAS device with tag id <tag id> on server <LifeKeeper server name> failed.
107002	Error getting list of IP addresses for NFS server device <NFS server name> on server <LifeKeeper server name>.
107003	Error attempting to find active address to NFS server <NFS server name> on server <LifeKeeper server name>.
107004	Error in format of device ID <resource device>.
107005	Cannot bring NAS resource <tag id> in service on server <LifeKeeper server name>. **Action**: After correcting the problem, try bringing the resource in service manually.
107006	create: Device not specified.
107007	Null Device returned by getId on <LifeKeeper server name>.
107008	Cannot open /etc/mtab file on <LifeKeeper server name>.
107009	Illogical settings for NAS defaults on <LifeKeeper server name>. Using defaults of 120 for LKNFSTIMEOUT and 5 for LKNFSSYSCALLTO. **Action**: Reset NAS default values so that three times the value for LKNFSSYSCALLTO plus 5 is less than the value of LKNFSTIMEOUT.
107010	Error: detected conflict in expected tag name <tag id> on target machine <LifeKeeper server

	name>. **Action**:Delete the conflicting resource and re-extend the hierarchy.
107011	Error: mkdir of “/tmp/nas_mntpt.2915” on “mouse” failed: “permission denied”.
107012	Error: Exported file system <NFS exported file system name> cannot be accessed on <server name>. Possible causes: - The LifeKeeper node is not in the exported system list on the NFS server, or, - The exported system list has contradictory entries that are not displayed by the **showmount** command. (i.e. if exported system list exports a file system to both the world and to specific systems, **showmount** will report only the specific systems). **Action**: Fix the exported file system access problem and re- extend the hierarchy.
107013	Error: Mount authorization check for “172.25.113.25:/ export” on “fred” appears to be hung. Exiting. **Action**: Fix the access problem and re-extend the hierarchy.

6.9.6.2. LifeKeeper GUI Related Errors

Error Number	Error Message
104901	The mount point %s is mounted **Action**: Please specify a mount point that is not mounted.
104902	The mount point %s is not an absolute path **Action**: Please specify a mount point that begins with a slash.
104903	The mount point %s is not empty. **Action**: Please specify a mount point that does not exist or is empty.

6.10. NFS Server Recovery Kit Administration Guide

The SIOS Protection Suite for Linux NFS Server Recovery Kit provides fault resilience for Network File System (NFS) software in a SIOS Protection Suite environment. This enables a failure on the primary NFS server to be recovered on a designated backup server without significant lost time or human intervention.

SIOS Protection Suite for Linux NFS Server Recovery Kit Overview

SIOS Protection Suite Documentation

The following is a list of SIOS Protection Suite for Linux related information available from SIOS Technology Corp.:

- SPS for Linux Technical Documentation
- SPS for Linux Release Notes
- SIOS Technology Corp. Documentation

Reference Documents

The following is a list of reference documents associated with the SIOS Protection Suite NFS Server Recovery Kit:

- NFS Online documentation
- *Managing NFS and NIS, Hal Stern, O'Reilly & Associates, Inc. 1991*

6.10.1. NFS Server Recovery Kit Overview

The NFS Server Recovery Kit provides a High Availability NFS service in hierarchical cooperation with the Filesystem Recovery Kit (provided as part of the steeleye-lk package) and the IP Recovery Kit (steeleye-lkIP).

The kit ensures that an IP resource and a file system resource containing the shared mount point are always in-service on the same server in the cluster. Clients who mount the file system using the LifeKeeper-protected IP resource can continue processing files on the volume virtually uninterrupted while the actual export service is switched between servers in the cluster (either manually or in response to a failure). Client recovery times will depend on the interaction between the client and the NFS server. For example, with NFSv3, the protocol timeouts for TCP are longer than that of UDP. In order to determine the best transport layer protocol to use with NFS, consider the recommendations of the OS vendor, the advantages and disadvantages of each transport protocol and your specific environment.

> **Note**: TCP is strongly recommended with NFSv4 by most OS vendors and the NFS ARK has been validated with TCP and NFSv4.

Beginning with Version 7.4 of the NFS Server Recovery Kit, an NFS v4 **pseudo file system** export is now supported providing clients with seamless access to all exported objects on the server. Prior to Version 7.4, clients were forced to mount each shared server file system for access. With NFS Version 4, the server still specifies export controls for each server directory or file system to be exported for NFS access, and from these export controls, the server renders a **single directory tree** of all the exported data filling in gaps between the exported directories. This tree is known as a pseudo file system, and it starts at the NFS Version 4 server's pseudo root. This **pseudo file system** model allows an NFS v4 client, depending on its implementation, to perform a single mount of the server's pseudo root in order to access all the server's exported data.

All files on the file system become temporarily unavailable while a switchover or failover is in progress, but they become available again transparently when the resource transfer is complete. For a switchover, this can take between 5 and 30 seconds. For a failover, the recovery time depends on how long it takes to repair the file system. It is strongly recommended that you format the underlying disk volume with a Journaling File System (JFS) which is extremely robust to failure and can be repaired in a few seconds.

You may also choose to use a Linux file system (ext2) as the underlying file system, but in that case, failover times will be much longer.

6.10.2. NFS Server Recovery Kit Requirements

Before installing and configuring the LifeKeeper NFS Server Recovery Kit, be sure that your configuration meets the following requirements:

- **Servers**. The Recovery Kit requires two or more servers configured in accordance with the LifeKeeper requirements described in the SPS for Linux Technical Documentation and the SPS for Linux Release Notes. See the Release Notes for supported Linux distributions.
- **LifeKeeper software**. You must install the same version of LifeKeeper software and any patches on each server. Please refer to the SPS for Linux Technical Documentation and the SPS for Linux Release Notes for specific LifeKeeper requirements.
- **LifeKeeper IP Recovery Kit**. You must have the same version of the LifeKeeper IP Recovery Kit on each server.
- **IP network interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

> **Note:** Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation and so forth.

- **TCP/IP software**. Each server also requires the TCP/IP software.
- **nfs-utils software**. Each server must have a high availability enabled version of the nfs-utils package installed and configured prior to configuring LifeKeeper and the LifeKeeper NFS Server Recovery Kit (on some OS distributions, `nfs-utils` is provided via another package). The same version must be installed on each server.

  The LifeKeeper Installation Support setup script will configure `nfs-utils` for use in an HA environment. Additionally, when using the kit to protect **NFS v2/v3 exports**, the init scripts NFS lock (*/etc/init.d/nfslock*) and NFS server (*/etc/init.d/nfs* or */etc/init.d/nfsserver* depending on the Linux distribution) should be configured to start automatically on system boot. When protecting **NFS v4** root exports, they should not be configured to automatically start at boot.
- Refer to the SPS for Linux Installation Guide for specific instructions on how to install or remove the LifeKeeper NFS Server Recovery Kit.

6.10.3. NFS Server Recovery Kit Configuration Considerations

These following sections contain information to be considered before starting to configure and administer the NFS Server Recovery Kit as well as examples of typical LifeKeeper NFS configurations.

Please refer to SPS for Linux Technical Documentation for instructions on configuring your LifeKeeper Core resource hierarchies.

- Configuring NFS Server with LifeKeeper
- Specific Configuration Considerations
- Configuration Examples

6.10.3.1. NFS Specific Configuration Considerations

The following should be considered before using the LifeKeeper NFS Server Recovery Kit:

- The NFS file system to be placed under LifeKeeper protection must be exported by the primary server (the server where the NFS resource is being created). This implies that NFS is running and the underlying file system is mounted.

> **Note**: If the */home directory* is shared via NFS, then */home* is the underlying file system.

- **When you export a read/write file system, use the *sync* option**. This option requests that all file system writes be committed to disk before the write request completes. **If the *sync* option is not used with an NFS file system under LifeKeeper protection, data may be lost during a failover.**
- The underlying file system must be on a shared device and mounted with write permission.
- If the underlying file system is already protected by LifeKeeper, it must be in service on the primary server and have the highest priority. If the underlying file system is not under LifeKeeper protection, then the Recovery Kit will place it under protection.
- The NFS Server Recovery Kit requires an IP resource that must be created and in service on the primary server. The IP resource must also have its highest priority on the primary server.
- Before creation of the NFS resource, clients must be able to mount the NFS file system using the LifeKeeper-protected IP address.
- When you extend an NFS file system resource, the file system must mount at the same mount point on each server.
- When protecting NFSv4 root exports, */var/lib/nfs* is moved to the NFSv4 root file system which must have write permissions set when mounted. To provide continued access, a symbolic link is created from */var/lib/nfs* to the NFSv4 root. Because of this, Active/Active NFSv4 configurations are not supported nor are configuations with NFSv2/v3 and NFSv4.
- The Oracle Recovery Kit supports NFSv3 for shared database storage. NFSv4 is not supported by the Oracle Recovery Kit at this time due to NFSv4 file locking mechanisms.
- When using a system that adopts Linux kernel 3.12 or later as an NFS client, the file lock at the client side is lost when the communication is disconnected depending on specifications of the kernel. Therefore, takeover of the lock is not guaranteed during switchover or failover.
- In NFS v4, the file lock is lost when the client cannot communicate with the server for more than a

certain period of time. This period of time is set as the variable NFS_V4_LEASE_TIME and the default value is 10 (seconds). When the communication between the client and the server is disconnected for a longer time than the period set in NFS_V4_LEASE_TIME during a switchover or failover, the file lock set by the client is forcibly disabled. In order to change this value, set an appropriate value of the environment variable NFS_V4_LEASE_TIME in /etc/default/LifeKeeper. If you increase the value of NFS_V4_LEASE_TIME, the above mentioned problem will less likely to occur but it will take time to reconnect to the client after the server switching.

NFSv2 is not supported on RHEL 7/CentOS 7/OL 7 or later.

NFS over UDP is not supported on RHEL 8 or later.

6.10.3.2. Configuring NFS Server with LifeKeeper

This section contains information to consider before starting to configure and administer the NFS Server Recovery Kit as well as examples of typical LifeKeeper NFS configurations.

Please refer to SPS for Linux Technical Documentation for instructions on configuring your LifeKeeper Core resource hierarchies.

NFS

The following table describes the NFS files, commands and daemons that are important to the NFS Server Recovery Kit:

NFS Component	Description
exports(5) (/etc/exports)	Access control list for file systems exported to NFS clients. Each line of the file contains an export point, an optional list of clients that can mount the file system and an optional list of mount parameters. **Note**: When you create a LifeKeeper-protected NFS resource, the export information for the file system is removed from the *exports* file and maintained under LifeKeeper. If you delete the NFS resource, the export information is restored to the *exports* file.
/var/lib/nfs	Directory that contains NFS information on current exports, client mounts, locking status and more. In Version 7.4 and later, this directory is moved to the NFS export directory when protecting an NFS v4 psuedo file system. */var/lib/nfs* is replaced with a symbolic link to the new location on both the primary and standby systems.
/var/lib/nfs/etab	File that contains the current table of exported file systems for NFS. This file is maintained by the **`exportfs`** command; the user does not edit the file directly. **Note**: When you bring an NFS resource into service on a backup server, the NFS file system is removed from the *etab* file on the primary server and inserted into the *etab* file on the backup server.
/var/lib/nfs/rpc_pipefs	Used for kernel to userspace communication for NFS. This directory is relocated to */var/lib* during installation of LifeKeeper.
`exportfs(8)` *(/usr/sbin/*	Command used to maintain the table of exported file systems in */var/lib/nfs/etab*.

exportfs)	
rpc.mountd(8) *(/usr/sbin/ rpc.mountd)*	Daemon that authenticates a mount request and returns a filehandle if the client is permitted to mount the file system.
rpc.nfsd(8) *(/usr/sbin/ rpc.nfsd)*	Daemon that handles client file system requests.
rpc.quotad(8) *(/usr/sbin/ rpc.rquotad)*	The rpc server that returns quotas for a user of a local file system which is mounted remotely over NFS.
`rpc.lockd(8)` *(/sbin/rpc.lockd)*	Daemon that handles client file lock requests.
`rpc.statd(8)` *(/usr/sbin/ rpc.statd)*	Daemon that monitors the status of and makes status notifications for NFS clients and servers. This daemon must be running in order for NFS file locking to work properly.
`portmap/ rpcbind`	Daemon process that converts RPC program numbers into port numbers and must be running for NFS. A failure of this process will force a switchover to a standby node. On some systems, this function is provided by portmap while on others this function is provided by `rpcbind`.
`rpc.idmapd`	NFS v4 ID to name mapper daemon process for translating user and group IDs to names and names to user and group IDs. This process must be running for NFS v4 but is not required for NFS v2/v3.

Export Considerations

LifeKeeper protection for a given exported file system depends on the export options being exactly of the form as described in the `exports(5)` man page. In particular, pay attention to the host restriction format. There are only four legal host restrictions: (single host, netgroup, wildcard host *name* and

netmask).

In particular, a wildcard IP address (like 172.13.4.*) is not legal and will lead to potential stale filehandles on switchover or failover. Check very carefully by executing `exportfs -v` and manually comparing the returned export description against the format described in the man page (unfortunately, `exportfs` doesn't check for you and will accept certain illegal export formats).

RPC.MOUNTD Restart

Under certain conditions with multiple NFS resource hierarchies, `rpc.mountd` fails to properly advertise the list of exports available. As such, the NFS Recovery Kit on a restore will stop and restart `rpc.mount` to ensure the proper list of exports is available to all clients. This action of stopping and restarting `rpc.mount` is controlled via the RESTARTMOUNTD entry in */etc/default/LifeKeeper*. By default, this entry is set to true to cause the stop and restart of_ rpc.mount_ on all NFS restores:

RESTARTMOUNT=true

To turn off this action set:

RESTARTMOUNT=false

NFS Resource Hierarchy

When you create a LifeKeeper protected NFS resource, LifeKeeper creates the following hierarchy:

- NFS file system resource (parent or root)
- HA-NFS resource
- File system resource (the underlying file system)

Create the IP address resource before creating the NFS resource.

You have the option of creating the file system resource(s) before creating the NFS resource. If you do this, you can choose the name assigned to the file system resource(s). If not, the NFS Server Recovery Kit automatically creates the file system resource(s) when creating the NFS resource.

Set up for Automatic Startup of the Service

Set up nfs-server.service for automatic startup via the following commands when protecting NFSv2/ NFSv3 exports in RHEL 7, CentOS 7, OL 7 or later and SLES12 or later environments where systemd is used.

```
# systemctl start nfs-server.service
```

```
# systemctl enable nfs-server.service
```

Note: Do not set up nfs-server.service for automatic startup when protecting NFSv4 exports.

Set up for Automatic Startup for rpcbind/portmap

Make sure that automatic startup for rpcbind or portmap is enabled.

- For systemd environment such as RHEL7, CentOS7, OEL7 and SLES12

```
# systemctl is-enabled rpcbind.socket
```

- For environment where systemd is not used

```
# chkconfig —list rpcbind
```

If the automatic startup of rpcbind or portmap is not active, execute following command to activate it:

- For systemd environment such as RHEL7, CentOS7, OEL7 and SLES12

```
# systemctl start rpcbind.socket

# systemctl enable rpcbind.socket
```

- For environment where systemd is not used

```
# service rpcbind start

# chkconfig rpcbind on
```

Handling of NFSv2/v3 and NFSv4

LifeKeeper protects the area as NFSv4 for which option “fsid=0” is specified. The area for which “fsid=0” is not specified is protected as NFSv2/3. Please note that the areas protected as NFSv2/3 and as NFSv4 cannot be mixed.

6.10.3.3. NFS Configuration Examples

The examples in this section show how NFS instances can be configured on shared (or replicated) disks. Each diagram shows the relationship between the type of configuration and the NFS parameters. Each configuration also adheres to the configuration rules and requirements described in this section that ensure compatibility between the NFS configuration and the LifeKeeper software.

This section first describes the configuration requirements and then provides the configuration examples **Active/Standby NFS v2/v3, Active/Active NFS v2/v3** and **Active/Standby NFS v4**.

> **Note**: Active/Active for NFS v4 is not supported nor is any combination of NFS v2/v3 and NFS v4.

The examples in this section are only a sample of the configurations than can be established, but understanding these configurations and adhering to the configuration rules will help define and set up workable solutions for your computing environment.

Configuration Requirements

NFS Tag names are arbitrary names that describe protected file systems to LifeKeeper. The default tag name suggested by LifeKeeper is *"nfs-<export point>."*

To understand the configuration examples, keep in mind that the underlying file system must always be on shared or replicated disks. The file system(s) must be mountable from each of the paired systems.

Examples

- Active/Standby Configuration for NFS v2/v3
- Active/Active Configuration for NFS v2/v3

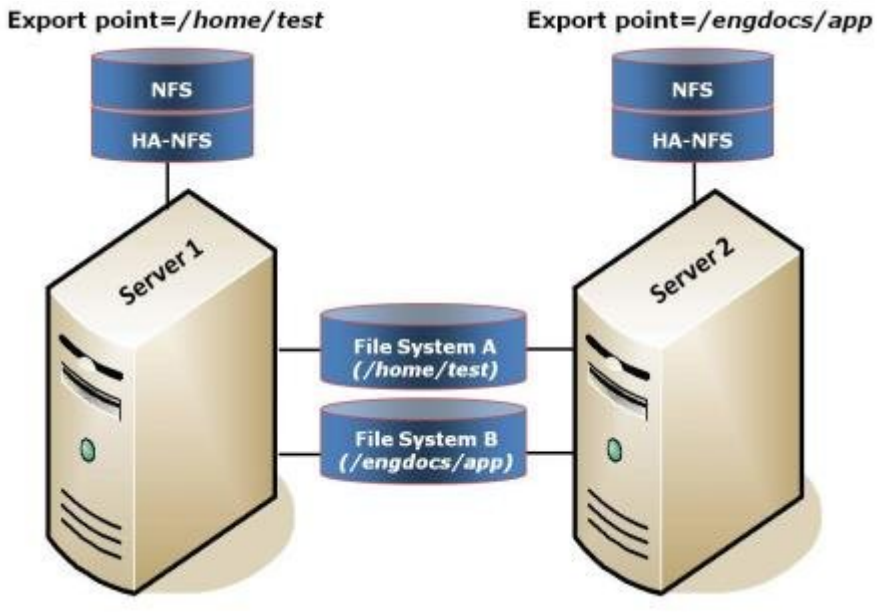
Related

- Configuration Considerations

6.10.3.3.1. Active – Active – NFS v2-v3

An example of **Active/Active for NFS v2/v3** consists of two or more systems actively running NFS and exporting file systems.

Active/Active Configuration Example for NFS v2/v3



Configuration Notes:

- The NFS software must be installed on both servers.
- Initially, *Server 1* exports */home/test* and *Server 2* exports */engdocs/app*. In a switchover situation, one system can export both file systems.
- File System A is the underlying file system for export point */home/test*. File System B is the underlying file system for export point */engdocs/app*.
- The underlying file systems are on different shared disks.

Creating the First Resource Hierarchy on *Server 1*:

Server:	*Server1*
Export Point:	*/home/test*
IP tag:	ip-172.17.100.202
NFS Tag:	*nfs-/home/test*

Extending the First Resource Hierarchy to *Server 2*:

Template Server:	*Server1*
Tag to Extend:	*nfs-/home/test*
Target Server:	*Server2*
Target Priority:	10

Creating the Second Resource Hierarchy on *Server 2*:

Server:	*Server2*
Export Point:	*/engdocs/app*
IP Tag:	ip-172.17.100.203
NFS Tag:	*nfs-/engdocs/app*

Extending the Second Resource Hierarchy to *Server 1*:

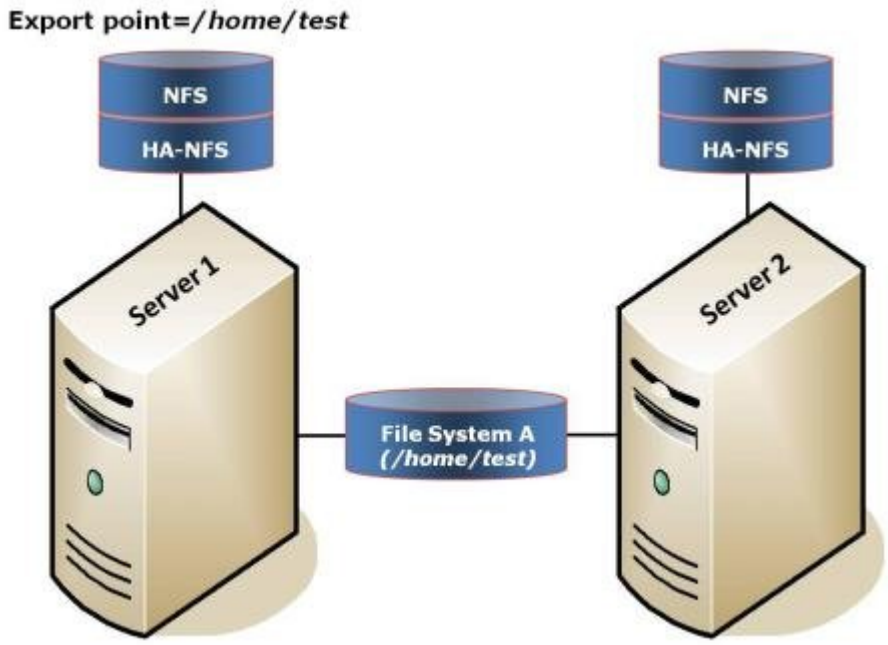
Template Server:	*Server2*
Tag to Extend:	*nfs-/engdocs/app*
Target Server:	*Server1*
Target Priority:	10

6.10.3.3.2. Active – Standby – NFS v2-v3

This section provides an example of **Active/Standby for NFS v2/v3**. In this configuration, *Server 1* is considered active because it is running NFS and exporting the file system. *Server 2* does other processing. If *Server 1* fails, *Server 2* gains access to the file system and uses the LifeKeeper secondary hierarchy to make it available to clients.

Note: In an active/standby configuration, *Server 2* might be running NFS but does not have any other NFS resources under LifeKeeper protection.

Active/Standby Configuration Example for NFS v2/v3



Configuration Notes:

- The NFS software must be installed on both servers.
- The underlying file system (*File System A*) must be on a shared (or replicated) disk.
- The NFS export point is */home/test*.
- The exported file system must have the same mount point on both the primary and backup servers.
- *Server 2* cannot access files and directories on the shared disk while *Server 1* is active.

Creating a Resource Hierarchy to *Server 1*:

Server:	*Server1*

Export Point:	*/home/test*
IP Tag:	ip-172.17.100.202
NFS Tag:	*nfs-/home/test*

Extending a Resource Hierarchy to *Server 2*:

Template Server:	*Server1*
Tag to Extend:	*nfs-/home/test*
Target Server:	*Server2*
Target Priority:	10

6.10.4. NFS Configuration Tasks

The following configuration tasks can be performed from the LifeKeeper GUI. These four tasks are described in this section as they are unique to an NFS Server resource instance and different for each Recovery Kit.

> **Note**: Throughout this section, we explain how to configure your Recovery Kit by selecting certain tasks from the **Edit** menu of the LifeKeeper GUI. You can also select each configuration task from the toolbar. You may also right-click on a global resource in the **Resource Hierarchy Tree** (left-hand pane) of the status display window to display the same drop-down menu choices as the **Edit** menu. This, of course, is only an option when a hierarchy already exists.

You can also right-click on a resource instance in the **Resource Hierarchy Table** (right-hand pane) of the status display window to perform all the configuration tasks, except **Creating a Resource Hierarchy**, depending on the state of the server and the particular resource.

- Create a Resource Hierarchy. Creates an application resource hierarchy in your LifeKeeper cluster.
- Delete a Resource Hierarchy. Deletes a resource hierarchy from all servers in your LifeKeeper cluster.
- Extend a Resource Hierarchy. Extends a resource hierarchy from the primary server to a backup server.
- Unextend a Resource Hierarchy. Unextends (removes) a resource hierarchy from a single server in the LifeKeeper cluster.
- Test Your Resource Hierarchy. Tests your NFS resource hierarchy by initiating a manual switchover.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.

- View / Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

6.10.4.1. Creating an NFS Resource Hierarchy

To create a resource instance from the primary server, you should complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy**.

   To change a selection already entered or if an error message is encountered during any step in the creation of your NFS resource hierarchy, use the **Back** button to change your selection or make corrections (assuming the **Back** button is enabled).

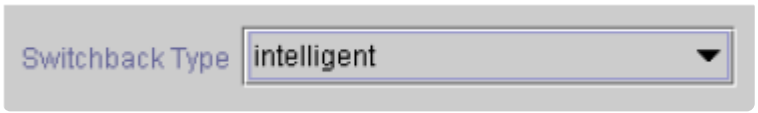
   A dialog box will appear with a drop-down menu listing all recognized Recovery Kits installed within the cluster. Select **NFS** from the drop-down menu.

   Click **Next** to proceed to the next dialog box.

> **Note**: If you click the **Cancel** button at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

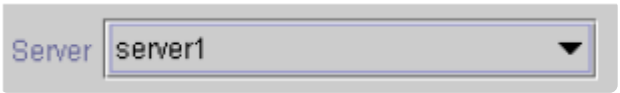
2. Select the **Switchback Type**. This dictates how the NFS instance will be switched back to the primary server when it comes back into service after a failover to the backup server. Choose either **Intelligent** or **Automatic**. **Intelligent switchback** requires administrative intervention to switch the instance back to the primary/original server. **Automatic switchback** means the switchback will occur as soon as the primary server comes back on line and is under LifeKeeper protection.

   The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box.

   Click **Next** to proceed to the next dialog box.

3. Select the **Server** where you want to create the NFS resource (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop-down menu.

Click **Next** to proceed to the next dialog box.

4. The **Export Point** dialog displays a drop-down list of export points for NFS file systems that meet the following criteria:

- ◦ The export point has been exported by NFS.
  - ◦ The export point is on a shared drive.
  - ◦ If the underlying file system is LifeKeeper-protected, it must be in service and have the highest priority on the server selected on the **Server** dialog.
  - ◦ NFSv4 criteria:
    - º For

NFS v4 root export with bind mounts, bind mounts must be on a shared drive just like the export, and if the file system is LifeKeeper-protected, it must be in service and have the highest priority on the server selected on the Server dialog.

- º If an

NFS v4 root export is already being protected, no choices will be provided (there should only be one v4 and a mixture of V2/v3 with v4 cannot be protected).

- º If an

NFS v2/v3 is already being protected, no NFS v4 will be listed in the choices.

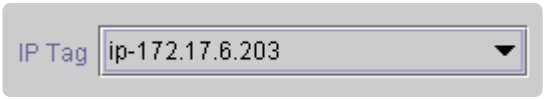
- º If nothing is protected, then the list could contain both v2/v3 and v4.

Select the NFS export point to be protected from the drop-down list.

Click **Next** to proceed to the next dialog box.

5. The **IP Tag** dialog displays a drop-down list of tags corresponding to virtual IP addresses currently under LifeKeeper protection and in service on the server where the NFS resource is being created.

Select the **tag** for the virtual IP address used by clients to access the protected NFS file system.



Note: At this point, LifeKeeper will check to ensure that there is a protected IP resource available. It will also validate that you have provided valid data to create your NFS resource hierarchy. If LifeKeeper detects a problem with either of these validations, an

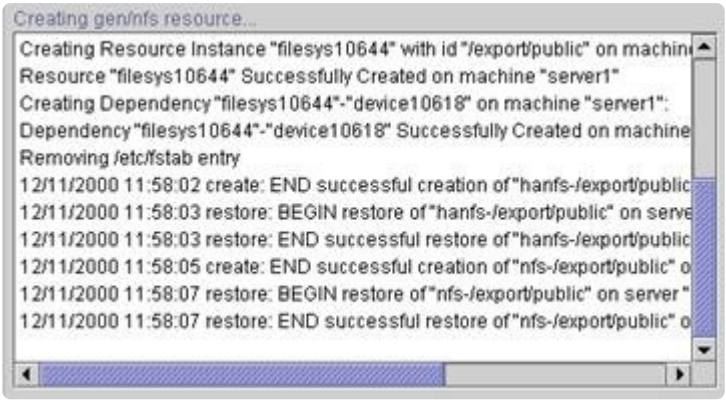
ERROR box will appear on the screen. If the directory paths are valid but there are errors with the NFS configuration itself, you may pause to correct these errors and continue with the hierarchy creation. You may even pause to create any LifeKeeper IP resources that are required.

Note: If you are using other LifeKeeper Recovery Kits that have virtual IP address dependencies, you might want to create a different virtual IP address for the NFS resource. Otherwise, if the virtual IP resource fails over to a backup server, all of the resources that depend on that IP resource will fail over at the same time.

Click **Next** to proceed to the next dialog box.

6. Select or enter the **NFS Tag**. This is a tag name given to the NFS hierarchy. You can select the default or enter your own tag name.

When you click the **Create** button, the ***Create Resource Wizard*** will create your NFS resource.



When you click **Next**, LifeKeeper will launch the ***Pre-Extend Wizard*** that is discussed in Extending Your Hierarchy

Note: The NFS resource hierarchy should be created successfully at this point. However, error messages may be encountered indicating that the new NFS instance has failed to start correctly. Note that the new NFS hierarchy must be started (In Service) before it can be extended to another system. A failure to start may remove the hierarchy, but if not, you may pause at this point and correct the problem based on the error message displayed. If the errors are not correctable, you will only be given the choice to cancel which cancels the resource create.

Bring the new hierarchy In Service before proceeding with extending your hierarchy.

Note: Please disable automatic startup of nfs-server.service after creating NFS resources on RHEL 7.1 or later and SLES12 SP1 or later. Since it is necessary for rpcbind.service to be running at the startup of NFS resources, please configure rpcbind.service to start automatically.

6.10.4.2. Deleting an NFS Resource Hierarchy

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop-down menu, select **Delete Resource Hierarchy**.

2. Select the name of the **Target Server** where you will be deleting your NFS resource hierarchy.

> **Note**: This dialog box does not appear if you select the **Delete Resource** task by right-clicking from either of the following:

- The right pane on an individual resource instance
- The left pane on a global resource when the resource is on only one server

Click **Next** to proceed to the next dialog box.

3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete and highlight it.

> **Note**: If you selected the Delete Resource task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

Click **Next** to proceed to the next dialog box.

4. An information box appears confirming your selection of the target server and the hierarchy you

have selected to delete.

Click **Delete** to proceed to the next dialog box.

5. Another information box appears confirming that the NFS resource was deleted successfully.

6. Click **Done** to exit out of the **Delete Resource Hierarchy** menu selection.

6.10.4.3. Extending Your NFS Hierarchy

After you have created a hierarchy, you will want to extend that hierarchy to another server in the cluster. There are two possible scenarios to extend your resource instance from the template server to a target server. The first scenario is when you "Continue" from creating the resource into extending that resource to another server. The other scenario is when you enter the Extend Resource Hierarchy task from the edit menu as shown below. Both scenarios take you through the same dialog boxes (with a few exceptions, which are detailed below).

1. If you are entering the ***Extend Wizard*** from the **LifeKeeper GUI** menu, select **Edit**, then **Resource**. From the drop-down menu, select **Extend Resource Hierarchy**. This will launch the **Extend Resource Hierarchy** wizard.

2. The first dialog box to appear will ask you select the **Template Server** where your NFS resource hierarchy is currently in service. It is important to remember that the Template Server you select now and the **Tag to Extend** that you select in the next dialog box represent an in service resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you selected. The drop-down box in this dialog provides the names of all the servers in your cluster.

Note: If you are entering the **Extend Resource Hierarchy** task immediately following the creation of a NFS resource hierarchy, this dialog box will not appear, since the wizard has already identified the template server in the create stage. This is also the case when you right click on either the NFS resource icon in the left hand pane or right-click on the NFS resource box in the right hand pane on the of the GUI window and choose **Extend Resource Hierarchy**.

Note: If you click the **Cancel** button at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extension process to that particular server. LifeKeeper will also unextend any dependent resources in the hierarchy (IP address or file system) that are currently extended past the cancellation point. However, if you have already extended the NFS resource hierarchy to another server, that instance will continue to be in effect until you specifically unextend it.

For example, let's say you have created your resource on Server 1 and extended that resource to Server 2. In the middle of extending the same resource to Server 3, you change your mind and click on the Cancel button inside one of the dialog boxes. This will cancel only your action to extend the resource to Server 3, not the extension you created to Server 2. If you want to remove Server 2 from this hierarchy, you must unextend the resource from Server 2.

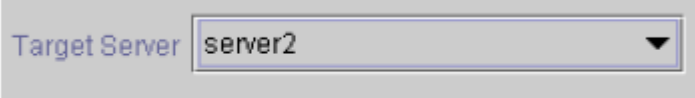
Click **Next** to proceed to the next dialog box.

3. Select the **Tag to Extend**. This is the name of the NFS instance you wish to extend from the template server to the target server. The wizard will list in the drop down menu all the resources that you have created on the template server, which you selected in the previous dialog box.

> **Note**: Once again, if you are entering the **Extend Resource Hierarchy** task immediately following the creation of an NFS resource hierarchy, this dialog box will not appear, since the wizard has already identified the tag name of your NFS resource in the create stage. This is also the case when you right-click on either the NFS resource icon in the left hand pane or on the NFS resource box in the right hand pane of the GUI window and choose **Extend Resource Hierarchy**.

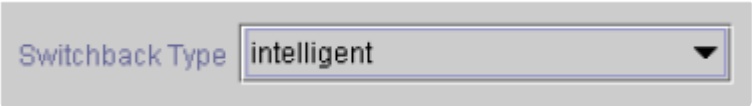
Click **Next** to proceed to the next dialog box.

4. Select the **Target Server** where you are extending your NFS resource hierarchy. The drop-down box provides the names of the servers in your cluster that are not already in the selected hierarchy.



Click **Next** to proceed to the next dialog box.

5. Select the **Switchback Type**. This dictates how the NFS instance will be switched back to the primary server when it comes back into service after a failover to the backup server. You can choose either intelligent or automatic. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and is under LifeKeeper protection.



The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box.

Click **Next** to proceed to the next dialog box.

6. Select or enter a **Template Priority**. This is the priority for the NFS hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. Note: This selection

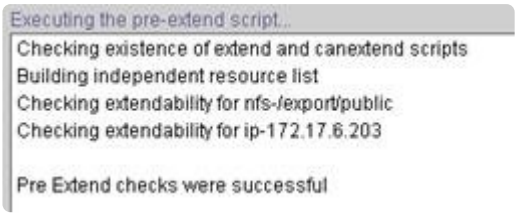
will appear only for the initial extend of the hierarchy.

Click **Next**.

7. Select or enter the **Target Priority**. This is the priority for the new extended NFS hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

Click **Next**.

8. An information box will appear explaining that LifeKeeper has successfully checked your environment and that all the requirements for extending this NFS resource have been met. If there were some requirements that had not been met, LifeKeeper would not allow you to select Next, and the **Back** button would be enabled.



If you click **Back**, you can make changes to your resource extension according to any error messages that may appear in the information box.

If you click **Cancel** now, you will need to come back and extend your NFS resource hierarchy to another server at some other time to put it under LifeKeeper protection.

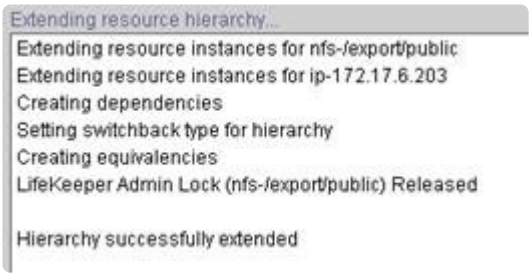
When you click **Next**, LifeKeeper will launch you into the **Extend Resource Hierarchy** configuration task and the **NFS Tag** dialog box will display.

9. This screen provides information about the **Template Server**, **Tag to Extend**, **Target Server** and the default **NFS Tag**. The **NFS Tag** is a tag name given to the NFS hierarchy extension. You can select the default or enter your own tag name.

Click **Next** to proceed to the next dialog box.

10. An information box will appear verifying that the extension is being performed.

Note: If you have not already extended the IP resource to the target server, the NFS Server Recovery Kit extends it in the process of extending the NFS resource. Before displaying the extension verification information box, the Recovery Kit displays several additional dialog boxes related to the extension of the IP resource.



Click **Next Server** if you want to extend the same NFS resource instance to another server in your cluster. This will repeat the Extend Resource Hierarchy operation.

If you click **Finish**, LifeKeeper will verify that the extension of the NFS resource was completed successfully.

11. If you click **Finish**, the following screen appears.

12. Click **Done** to exit.

Note: Be sure to test the functionality of the new instance on both servers.

6.10.4.4. Testing Your NFS Hierarchy

Before testing your NFS resource hierarchy, you should validate your client setup as described below. You can then test your NFS resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Validating the Client Setup

In general, clients must mount the file system using the LifeKeeper-protected IP address you selected during the Create NFS Resource Hierarchy task. There is no client-side checking to ensure that you select the correct IP address, so you must carefully follow the validation steps below to ensure the client is using the correct IP number for the file system.

To validate the client setup, do the following:

1. Verify that no NFS instances are in service on the secondary system.
2. Mount the file system on the client using the correct LifeKeeper-protected IP address.
3. Perform a manual switchover to the secondary system and ensure that the NFS instance you just switched over is the only NFS instance currently in service on the secondary.
4. When the switchover has completed, ensure that the client can still access the file system.

Performing a Manual Switchover from the GUI

After you define the dependencies, LifeKeeper automatically controls the starting and stopping of the application whenever it detects faults, which initiate failover recovery. You can also manually initiate a switchover for administrative reasons, such as maintenance.

You can initiate a manual switchover from the LifeKeeper GUI by selecting Edit, Resource, and In Service from the drop-down menu. For example, an in service request executed on a backup server causes the application hierarchy to be placed in service on the backup server and taken out of service on the primary server.

If you execute the Out of Service request, the application is taken out of service without bringing it in service on the other server.

Note: To take an NFS resource out of service, you must take both the NFS resource and the associated HA-NFS resource out of service.

For activities within the application, all actions are those defined in the application's documentation. LifeKeeper does not regulate or control internal operations such as rollbacks and backing-up archives. Tape archiving and restoration are the responsibility of the application administrator.

Recovery Operations

When the primary server fails, the NFS Server Recovery Kit software performs the following tasks:

- Starts the NFS daemons if they are not running.
- Exports the NFS file system.

6.10.4.5. Unextending Your NFS Hierarchy

Perform the following steps to unextend a resource hierarchy:

1. From the **LifeKeeper GUI menu**, select **Edit** and **Resource**. From the drop-down menu, select **Unextend Resource Hierarchy**.

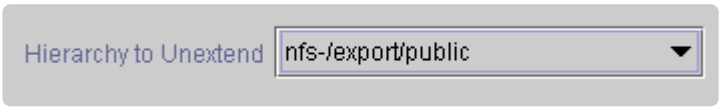
2. Select the **Target Server** where you want to unextend the NFS resource. It cannot be the server where the NFS resource is currently in service.

> **Note**: If you selected the **Unextend** task by right-clicking from the right pane on an individual resource instance this dialog box will not appear.

Click **Next** to proceed to the next dialog box.

3. Select the **NFS Hierarchy to Unextend**.

> **Note**: If you selected the **Unextend** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

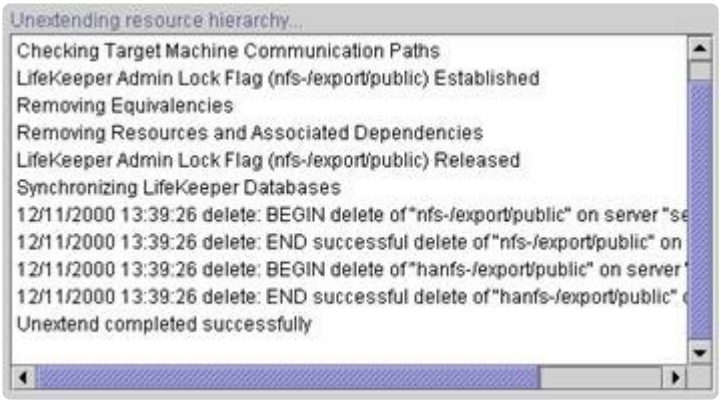


Click **Next** to proceed to the next dialog box.

4. An information box appears confirming the target server and the NFS resource hierarchy you have chosen to unextend.

Click **Unextend**.

5. Another information box appears confirming that the NFS resource was unextended successfully.



6. Click **Done** to exit the **Unextend Resource Hierarchy** menu selection and return to the LifeKeeper GUI.

   You will receive the warning **One or More Resources Unprotected** if the hierarchy is unextended down to one server.

6.10.5. NFS Troubleshooting

This section provides a list of messages that you may encounter while creating and extending a LifeKeeper NFS resource hierarchy or removing and restoring a resource. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Messages from other SPS components are also possible. In these cases, please refer to the Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

Click the following topics for Troubleshooting help.

- HA nfs-utils Installation and Configuration
- NFS Hierarchy Creation Errors
- NFS Extend Hierarchy Errors
- Hierarchy Restore, Remove and Recover Messages and Errors
- Hierarchy Delete Messages and Errors

6.10.5.1. HA nfs-utils Installation and Configuration

Version 5.0.0 and greater of the LifeKeeper NFS Server Recovery Kit requires the installation and configuration of a high availability enabled nfs-utils package (on some OS distribution versions, nfs-utils is provided via another package). The Recovery Kit will attempt to verify the presence of this HA enabled nfs-utils package. If it fails to detect a correctly configured nfs-utils package, the LifeKeeper Installation Support setup script may need to be rerun or the server may need to be rebooted.

The LifeKeeper Installation Support setup script will configure `nfs-utils` for use in an HA environment and will also perform setup configuration for **NFS v4** exports. You must answer **YES** to this HA NFS question for any NFS version that will be used.

The configuration needed for **NFS v4** requires the movement of *rpc_pipefs* from */var/lib/nfs* to */var/lib*. To do this may require the unloading of kernel modules and the addition or modification of configuration and boot time scripts. A system reboot may be required if Installation Support is unable to unload and reload kernel modules after the change. If this should occur, the user will be notified of the need for a system reboot. Completing the `rpc_pipefs` setup including a system reboot is required for successful operation of LifeKeeper.

6.10.5.2. Hierarchy Delete Messages and Errors

Error Number	Error Message	Description
106015	Unable to restore the entry for export point "EXPORT POINT" in */etc/exports* on server "SERVER"	Restore the entry manually.
106021	An entry for export point "EXPORT POINT" already exists in */etc/exports*. The entry that was being used by the NFS Server Recovery Kit has been placed in the file "FILENAME"	Verify that */etc/exports* has the correct export entry.
106049	Restore *statedir* from */var/lib/.nfs.LK* to */var/lib/nfs* failed on server SERVER.	Restoring the NFS state directory */var/lib/nfs* failed. Try manually restoring the directory by moving */var/lib/.nfs.LK* to */var/lib/nfs*.

6.10.5.3. Hierarchy Restore, Remove and Recover Messages

Bringing an NFS Resource In-Service (Restore)

Error Number	Error Message	Description
106007	Cannot bring NFS or HANFS resource “TAG” in service on server "SERVER"	Review other error messages to determine the action to take. After correcting the problem, try bringing the resource in service manually.
106010	NFS is not running on server “SERVER”. LifeKeeper will attempt to restart NFS.	This message is for information only. LifeKeeper will try to restart the NFS daemons automatically. If LifeKeeper encounters problems while restarting one of the daemons, you will receive a message that starting NFS failed.
106011	Starting NFS on server "SERVER" failed	LifeKeeper encountered a problem while restarting the NFS daemons. Try manually restarting NFS.
106012	The export point "EXPORT POINT" is not exported on server "SERVER". LifeKeeper will attempt to export the entry.	LifeKeeper has detected that the export point is no longer exported, and will try to export it.
106013	Unable to export "EXPORT POINT" on server "SERVER"	Try manually exporting the file system.
106014	Usage: USAGE STRING	Usage of command run with incorrect arguments. *Command line only*.
106019	Executing command: "COMMAND"	This message is displayed when LifeKeeper restarts an NFS daemon or exports/unexports an export point. It provides additional information that can be useful if there is a problem.
106024	Unable to stop and restart rpc.mountd on "SERVER!"	During a hierarchy restore the rpc.mountd daemon process needed to be restarted and this process failed. Manually attempt to stop and restart the process to determine the error and the action to take.
106027	Open of "ABC" on server "SERVER" failed: "File not found"	The attempted open failed for the reason listed.
106028	Mount of /proc/fs/nfsd failed on server "SERVER!"	In 2.6 and later kernels */proc/fs/nfsd* is used for client authentication and an attempt to mount it failed. Manually attempt to mount */proc/fs/nfsd* to determine the failure.
106029	Unable to get exclusive lock on "/var/lib/nfs/rmtab" on server "SERVER"	Unable to place an exclusive lock on /var/lib/nfs/rmtab for update after 20 seconds indicating a problem with the file.
106030	Unable to restore client info for "123.45.678.90" on server "SERVER!": "ABC"	Attempts to failover client 123.45.678.90 locks to server SERVER1 failed for the listed reason. Correct the failure condition and attempt to restore the hierarchy again.
106037	Attempts to get exclusive lock on "/var/lib/nfs/rmtab" on server "SEVER" failed: "ABC"	Unable to place an exclusive lock on /var/lib/nfs/rmtab for updating. See the error message for the cause.
106039	Open of "FILE" on server	An attempt to open or obtain an exclusive lock on a file has

	"SERVER" failed: "ABC" or Attempt to get exclusive lock on "FILE" on server "SERVER" failed: "ABC"	failed. See the error message for the cause.
106040	Multiple virtual IP addresses detected. In this release NFS lock failover only supports one virtual IP address.	Recreate the NFS resource hierarchies to use only one virtual IP address or set FAILOVERNFSLOCKS to false in the LifeKeeper defaults file.
106052	Unable to mount *rpc_pipefs* on "SERVER". Reason: "REASON".	*rpc_pipefs* was not mounted on SERVER and the mount attempt failed for REASON.
106053	*rpc_pipefs* successfully mounted on "SERVER"	*rpc_pipefs* was successfully mounted on SERVER.

Taking an NFS Resource Out of Service (Remove)

Error Number	Error Message	Description
106008	Unable to unexport the export point "EXPORT POINT" on server "SERVER"	Use the **exportfs(8)** command to unexport it.
106014	Usage: USAGE STRING	Usage of command run with incorrect arguments. *Command line only*.
106019	Executing command: "COMMAND"	This message is displayed when LifeKeeper restarts an NFS daemon or exports/unexports an export point. It provides additional information that can be useful if there is a problem.

Bringing an NFS Resource Back In Service (Recover)

The LifeKeeper core periodically checks the health of every NFS instance In Service on the local server by running an NFS “quickCheck” script. This script verifies the following:

- The file system is exported
- The NFS/HA-NFS daemons are running

If the instance is not fully functional, a "recover" script is invoked to attempt to restart the instance. This simply logs an error message, invokes "restore," prints the final error or success message – depending on error or success of the "restore" script – and returns the same result as "restore." If restore/recover fails, this instance is failed over to another server.

6.10.5.4. NFS Extend Hierarchy Errors

The error messages that might be displayed during NFS hierarchy extension are listed below, along with a suggested explanation for each. Note that these error messages appear when the GUI indicates it is "Executing the pre-extend script…." to validate the hierarchy prior to extending it to the new system.

During NFS Resource Hierarchy Creation on Target Server

Error Number	Error Message	Description
106016	"REQUIRED SOFTWARE" cannot be found or does not have the expected permissions on server "SERVER"	NFS must be installed on the primary server and all backup servers. Verify that the *nfs-utils* has been installed.
106017	The file system "FILE SYSTEM" on template server "SERVER" has a different mount point "MOUNT POINT" on server "SERVER"	The resources must be created with the same mount point on each server. Either unextend the file system hierarchy from the target server or recreate it with the same mount point on the template and target servers.
106018	Unable to copy the file "FILENAME" from server "SERVER" to server "SERVER"	Possible causes: • • Communication path between the servers is down • • File system is full of the target server • • File system problems – steeleye-lkNFS package needs to be reinstalled
106020	The generated id "nfs-/export" conflicts with an existing resource id	The internally generated resource ID for the nfs or hanfs resource has produced a duplicate.
106022	The export point "EXPORT POINT" is in */etc/exportson* the target server "SERVER"	Remove the export point from the */etc/exports* file on the target server before trying to extend the resource.
106023	The export point "EXPORT POINT" is exported on the target server "SERVER"	Unexport the export point on the target server before trying to extend the resource.
106051	Unable to create active/active configurations with NFS v4 exports. Either "TEMPLATE SERVER" or "TARGET SERVER" currently protects an NFS v4 root export.	Unable to extend the NFS resource from the TEMPLATE SERVER to the TARGET SERVER as one or both of the servers already protects an NFS v4 export and active/active configurations with NFS v4 exports is not supported.

6.10.5.5. NFS Hierarchy Creation Errors

The error messages that might be displayed during the NFS hierarchy creation are listed below, along with a suggested explanation for each. The messages listed cover both the creation of the nfs and hanfs resources. Error messages displayed by the LifeKeeper core and by other recovery kits are not listed in this guide. Note that you may stop to correct any problem described here, and then continue with hierarchy creation from the point where you left off – including creating any new LifeKeeper resources you might need for your NFS configuration.

Note: In the following error messages, *Command line* only indicates that you can only receive the message if you are entering commands on the command line; you cannot receive it if you are using the LifeKeeper GUI. Additionally, at the end of hierarchy create a resource restore is initiated. See Hierarchy Restore, Remove and Recover Messages and Errors for an explanation of messages and errors that can occur during that process.

Error Number	Error Message	Description
106000	Export point not specified	You must specify the export point for the NFS file system when you create the resource hierarchy. *Command line only*.
106001	The path "EXPORT POINT" is not exported by NFS	The export point you specified is not currently exported by NFS. Use **exportfs(8)** to export the path and verify the path is in the */var/lib/nfs/etabfile*. *Command line only*.
106002	create: The export point "EXPORTPOINT" on "server1" for client "CLIENT" either does not contain an FSID export option or the value is not unique. A unique FSID will be generated and "EXPORTPOINT" will be re-exported using the new FSID value.	All export point under LifeKeeper protection must use a unique fsid= export option for high availability NFS. The selected export did not meet this requirement so a unique value was generated followed by a re-export for the selected export point. Note: a client of "*" or "world" indicates the export point is available to all clients.
106003	Unable to create the export entry file for "EXPORT POINT" in LifeKeeper	Possible causes: • • The file system is full on the target server. • • File system problems – Steeleye-lkNFS package needs to be reinstalled.
106004	The export point "EXPORT POINT" is not on a shared file system on server "SERVER"	Make sure that the export point is for a shared file system. *Command line only*.
106005	Unable to create the HA-NFS hierarchy "TAG" with child resource "TAG" on server "SERVER"	Review the other error messages to determine the action to take.

106006	Unable to remove entry for export point "EXPORT POINT" from */etc/exportson* server "SERVER"	Verify that the */etc/exports* file exists and is readable.
106014	Usage: USAGE STRING	Usage of command run with incorrect arguments. *Command line only*.
106016	"REQUIRED SOFTWARE" cannot be found or does not have the expected permissions on server "SERVER"	NFS must be installed on the primary server and all backup servers. Verify that the **nfs-utils** has been installed
106019	Executing command: "COMMAND"	This message is displayed when LifeKeeper restarts an NFS daemon or exports/unexports an export point. It provides additional information that can be useful if there is a problem.
106020	The generated id "nfs-/export" conflicts with an existing resource id	The internally generated resource ID for the nfs or hanfs resource has produced a duplicate.
106025	An unknown error has occurred while running "newtag" on server "Server1"	An unexpected error occurred while running the command newtag to generate a tag for the nfs for hanfs resource.
106026	Adding dependency between HA-NFS resource "hanfs-/export" and filesysresource "export" on server &#34server1" failed.	Dependency creation between the selected hanfs resource and the filesys resource has failed for unknown reasons. See output for more information.
106027	Open of "ABC" on server "SERVER" failed: "File not found"	The attempted open failed for the reason listed.
106029	Unable to get exclusive lock on "/var/lib/nfs/rmtab" on server "SERVER"	Unable to place an exclusive lock on /var/lib/nfs/rmtab for update after 20 seconds indicating a problem with the file.
106031	Re-export of "/export" to add FSID option failed on server "SERVER"	The export point "/export" did not contain a fsid argument and the re-export after one was generated failed. Manually add a fsid argument to the /etc/exports entry and re-export to determine the failure.
106032	Dependent IP resource tag name not specified	You must specify the resource tag for the protected IP address when you create the NFS resource hierarchy. Command line only.
106033	Selected IP resource "TAG" does not exist on server "SERVER"	You must create the IP resource on the specified server before you can create the NFS resource. Also, make sure that you typed the IP resource correctly when you entered the command. Command line only.
106034	Adding dependency between NFS resource "TAG" and IP resource "TAG" on server "SERVER" failed	Verify the IP resource is in-service on the server where the NFS resource is being created. Command line only.
106035	Creation of HA-NFS resource "TAG" on server "SERVER" failed	Review other error messages to determine the action to take.
106036	Adding dependency between IP resource "TAG" and HA-NFS resource "TAG" on server "SERVER" failed	Verify that the IP resource is in-service on the server where the HA-NFS resource is being created. Command line only.
106037	Attempts to get exclusive lock on "/var/lib/nfs/rmtab" on server "SEVER" failed: "ABC"	Unable to place an exclusive lock on /var/lib/nfs/rmtab for updating. See the error message for the cause.

106038	Unable to create directory "ABC" on server "SERVER": "ABC"	An attempt to create a directory on the exported file system has failed. See the error message for the cause.
106039	Open of "FILE" on server "SERVER" failed: "ABC" or Attempt to get exclusive lock on "FILE" on server "SERVER" failed: "ABC"	An attempt to open or obtain an exclusive lock on a file has failed. See the error message for the cause.
106041	The selected IP resource “ip-123.45.678.90” is not ISP on server "SERVER"	The selected IP resource does exist on the server but is not currently in service. Bring the IP resource in service on the server and then re-attempt the creation. Command line only.
106048	Multiple NFS v4 root exports found on "SERVER".	Multiple NFS v4 psuedo file systems found on SERVER where only one is supported.
106050	Unable to protect more than one NFS v4 export or a combination of NFS v4 and NFS v3 exports.	Attempting to protect a mix of NFS v2/v3 exports with NFS v4 which is not supported.

6.11. Oracle Recovery Kit Administration Guide

The SIOS Protection Suite for Linux Oracle® Recovery Kit provides fault resilience for Oracle software in an SPS environment. The Recovery Kit software furnishes a mechanism to tie the data integrity of Oracle databases to the increased availability provided by SIOS Protection Suite.

Document Contents

This documentation contains the following topics:

SPS for Linux Technical Documentation (also available from the Help menu within the LifeKeeper GUI). A list of all the SPS for Linux documentation and where the information is available.

Requirements. Before you can install and set up the recovery software, your server must meet certain hardware and software requirements. You should refer to the SIOS Protection Suite Installation Guide for specific instructions on how to install or remove the LifeKeeper Oracle Recovery Kit.

Configuring Your Recovery Kit. To ensure that your SPS configuration provides the protection and flexibility you require, you need to be aware of the configuration rules. To appropriately plan your configuration, you must understand your network configuration, interface selection, user system setup, hierarchy options and the Oracle configuration tasks. In addition to planning your configuration, this section also includes configuration examples and the specific tasks required to configure your Recovery Kit.

Troubleshooting. This section provides a list of informational and error messages with recommended solutions.

6.11.1. Oracle Recovery Kit Hardware and Software Requirements

Before attempting to install or remove the LifeKeeper Oracle Recovery Kit, you must understand the hardware and software requirements and the installation and removal procedures.

Kit Hardware and Software Requirements

Be sure that your configuration meets the following requirements:

Servers. The Recovery Kit requires two or more supported computers configured in accordance with LifeKeeper requirements described in the SPS Technical Documentation and the SPS Release Notes.

Shared Storage. Oracle databases must reside on a shared disk in an SPS environment. Depending on your shared storage architecture, the appropriate LifeKeeper shared storage or multipath storage kit will need to be installed on each node in your cluster. In the example of NFS backed database storage, installation of the LifeKeeper NAS Kit is necessary.

LifeKeeper Software. You must install the same version of LifeKeeper software and any patches on each server. Please refer to the SPS Release Notes and SPS Technical Documentation for specific LifeKeeper requirements.

LifeKeeper IP Recovery Kit. This Recovery Kit is required if remote clients will be accessing the Oracle Database. You must have the same version of this Recovery Kit on each server.

IP Network Interface. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

Note: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation, and so forth.

TCP/IP Software. Each server also requires the TCP/IP software.

Oracle Software. Each server must have the Oracle software installed and configured before you can configure LifeKeeper and the LifeKeeper Oracle Recovery Kit. The same version should be installed on each server. Consult the SPS Release Notes or your sales representative for the latest release compatibility and ordering information.

You should refer to the SPS Installation Guide for specific instructions on how to install or remove the LifeKeeper Oracle Recovery Kit.

6.11.2. Configuring Oracle with LifeKeeper

This section contains information you should consider before you start to configure Oracle and examples of typical Oracle configurations.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View Properties. View the properties of a resource hierarchy on a specific server.
- Edit Properties. Edit the properties of a resource hierarchy on a specific server.

For instructions on installing Oracle on Linux distributions using the 2.6 kernel, please see your Linux distribution's website.

Also, please refer to your SPS for Linux Technical Documentation located on the SIOS Technology website for instructions on configuring your LifeKeeper Core resource hierarchies.

6.11.2.1. Specific Configuration Considerations for Oracle

Note: If you plan to use Oracle with Raw I/O, the Raw I/O devices must be properly set up prior to use. See the Appendix for instructions. (Raw I/O is not an option for LifeKeeper Single Server Protection.)

Before configuring the Oracle Recovery Kit, complete the following preparatory steps to ensure that file systems and disk partitions used by Oracle will be accessible from each server.

1.
2. **Remove personal initialization file prompts.** For the Oracle Recovery Kit to work properly, you must remove (or comment out) all prompts in the personal initialization file (i.e., .profile, .bash_profile) for the Oracle user. This file is specific to the shell that is being used by the Oracle user. The file cannot be interactive.

   Note: If “stty” statements are going to be in the personal initialization file, they must be in an “if” statement that verifies that an interactive terminal is being used.

3. **Configure Kernel Parameters.** Please refer to the Oracle documentation for information on how linux kernel parameters such as shared memory and other kernel resources should be configured. An example of how to set these parameters is below.

   On *each server* in the cluster:

   a. Set the following ipcs limits in `/etc/sysctl.conf` before configuring LifeKeeper.

   ```
   # changes for Oracle
   kernel.shmmax = <value>
   kernel.shmmni = <value>
   kernel.shmall = <value>
   kernel.sem = <value>
   ```

   b. Run `sysctl -p` to set the above changes in the kernel.

   c. On certain distributions you may need to add `sysctl -p` to the system initialization file (i.e. boot.local or rc.local) so that these kernel changes are set after each reboot.

4. **$ORACLE_HOME directory.** When you configure the *$ORACLE_HOME* directory and associated files on local disks, be sure that the *$ORACLE_HOME* directory and files are identical on all servers. Use the standard Linux utilities to create and copy directories and files to the set of servers. **Note:** In certain active/active configurations, the location of *$ORACLE_HOME* are different.

5. **Location.** The *$ORACLE_HOME* directory can be on shared or non-shared disks. The advantage to having the directory on shared media is that you only need to configure files such as the

parameter file `Oracle_HOME/dbs/<initSID.ora or spfileSID.ora>` once, if the same shared disk is used for *$ORACLE_HOME* (e.g. in an active/standby configuration). The disadvantage to the shared directory is that direct access to the file system is available to only one server at a time. SCSI reservations permit only one server at a time access to a LifeKeeper protected shared drive. If creating an active-active cluster configuration where two or more Oracle instances (SID) will be protected independently in the cluster, *$ORACLE_HOME* must be installed on local, non-shared storage.

6. **User and Group ID.** An oracle user (oracle) and group (dba) should be created on all servers. The user ID and group ID numbers must be the same on all servers.

7. **Databases, archive files, log files and control files.** All databases, archive files, log files, and control files must be created on shared file systems or disk partitions. These locations are set in the Oracle parameter file `init<SID>.ora` or `spfileSID.ora`. Please refer to the Oracle documentation for information on editing database parameters. The pathnames must be the same for all servers. Oracle internally keeps this information in its control file; therefore, SYSTEMS database space and paths cannot be changed unless Oracle is running.

   Note: Oracle log archiving is not enabled by default. If it is enabled prior to the creation of the LifeKeeper Oracle hierarchy, LifeKeeper will detect the location of the archive files and create a separate file system hierarchy if necessary. But if log archiving is enabled after the LifeKeeper Oracle hierarchy has been created, you must manually create and extend a file system hierarchy to protect the shared archive location, and create a dependency from the Oracle resource to this new file system hierarchy.

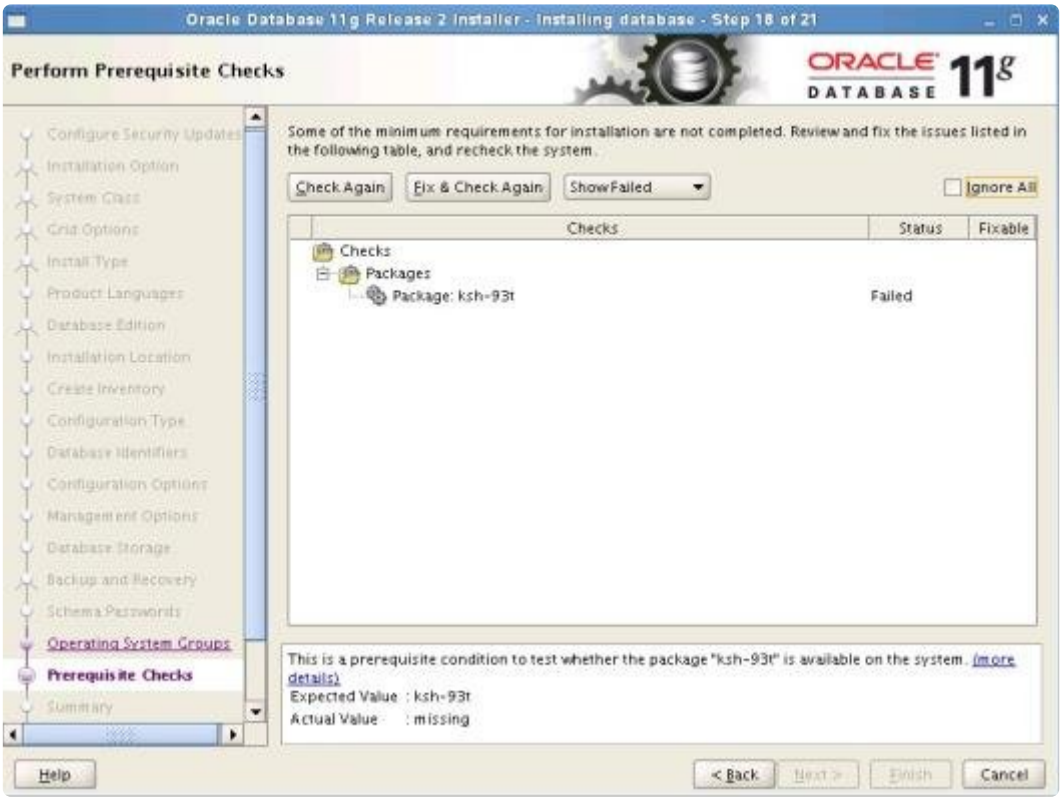
   Note: When using storage applications with locking and following recommendations for the NFS mount options, SPS requires the additional `nolock` option be set, e.g. `rw,nolock,bg,hard,nointr,tcp,nfsvers=3,timeo=600,rsize=32768,wsize=32768,actimeo`

8. On a new installation of Oracle, the final configuration of the database instance is easier if the database installation program is not allowed to create a database. When the installer asks if you want to create a database, select **No**. After the installation is complete, run the **Oracle Database Creation Assistant** (dbca). dbca provides much better control of where database components get created. When running dbca, specify that the Flash Recovery Area gets created on LifeKeeper protected storage (this applies to Oracle 11g).

   Important: The Flash Recovery Destination must be located on a shared drive.

   If runInstaller is allowed to create a database, the Flash Recovery Area will have to be relocated manually. (**Note:** Allowing `runInstaller` to create a database is not recommended.)

9. During the installation of Oracle using the "`runInstaller`" utility, there will be a point where the installer verifies the packages and configuration of Linux before proceeding with the Oracle database installation. If LifeKeeper 7.2 (or higher version) has already been installed, a message complaining about a missing ksh package will appear.



If this message is displayed, check the box in the upper righthand corner, **Ignore All**. The installation of LifeKeeper has removed the ksh package and replaced it with the Public Domain Korn Shell, `pdksh`. Oracle should install fine using `pdksh`.

Note: Beginning with Version 8.0, the Oracle ARK no longer requires `pdksh`; however, `pdksh` is still required by the LifeKeeper core and therefore still requires checking the Ignore All setting.

Note: Beginning with Version 8.1, LifeKeeper provides its own private `pdksh` package and therefore does not conflict with Oracle's `ksh` requirements.

10. **Tune the database engine.** Refer to Oracle documentation for guidelines on tuning the database engine for data integrity and performance. In particular, the tuning for memory caching and checkpointing frequency is critical to optimizing the application for fault resilience. The checkpointing interval determines the number of database transactions that have not been committed to disk and therefore would be lost during a system failure.

11. **Database entry in oratab file.** The `/etc/oratab` file must contain an entry for the database. The LifeKeeper configuration routines use the contents of this file to relate *$ORACLE_HOME* and *$ORACLE_SID* values. Usually, the Oracle installation program creates the required entry. In a configuration in which the Oracle software is installed to a shared file system, however, you must copy the `oratab` file from the server where the Oracle installation was performed to the `/etc` directory of the other servers so that it is available to all the servers.

    Note: The configuration can have only one oratab per server. Refer to the Oracle Product Manual for information on the file format.

Note: The `oratab` file can be accommodated in other locations besides `/etc`. By default, the Oracle ARK looks for the `oratab` file in `/etc` followed by `/var/opt/oracle`. If the `oratab` file is not located in one of these default locations, then `ORACLE_ORATABLOC` must be set in `/etc/default/LifeKeeper` to the directory containing `oratab`.

12. **Disable automatic start-ups.** Since LifeKeeper is responsible for starting the databases it controls, be sure to disable any automatic start-up actions. LifeKeeper disables automatic start-up when a hierarchy is created. This is accomplished by modifying the `oratab` file.

13. The Listener configuration file, `listener.ora`. New lines should not be embedded in the entries (e.g., SID_NAME=xx should be on one line).

14. **Oracle Database Username and Password.** LifeKeeper will use local session and OS Authentication to control Oracle Database. If you would like to turn off local OS Authentication for security reasons, LifeKeeper can use the specified username and password. The Oracle Database user must be able to connect as sysdba authority to the database to be protected, and each server's Oracle Database must have the same username and password. If this configuration is skipped during resource creation, then LifeKeeper will not use username and password to control the Oracle Database resource. This parameter can be added, changed or removed any time after creating the resource.

    Once under LifeKeeper protection, the LifeKeeper and database user privileges can be lowered from sysdba to sysoper. See Changing Username / Password for the Oracle Database Account for more information.

    Tips for Creating the Oracle Username and Password.

    a. On the node where the Oracle database is running, log in to Linux with a user that is part of the dba group. (The "oracle" account is most common.) Using the sqlplus utility, connect to the database as the administrative user by issuing the following command:

    ```
    $ sqlplus / as sysdba
    ```

    b. Create a new user for this function:

    ```
    SQL> CREATE USER lkdba IDENTIFIED BY "password";
    ```

    c. Then grant this user SYSDBA privileges:

    ```
    SQL> GRANT SYSDBA to lkdba;
    ```

    d. If Oracle has been configured so that each node in the LifeKeeper cluster has a local copy of *$ORACLE_HOME*, execute these commands on each node in the cluster. After creating the LifeKeeper Oracle hierarchy, bring the database in service on the node and then execute the `CREATE` and `GRANT` commands (above) to set up the user in Oracle.

CAUTION: Avoid configuring two databases on the same file system. If you must configure two databases on the same file system, exercise great care. In this situation, both databases must be placed under LifeKeeper protection and both hierarchies must have the same primary and backup servers.

6.11.2.2. Configuring the Oracle Net Listener for LifeKeeper Protection

If your Oracle database will have remote client connections, you will want to protect the Oracle Listener in addition to the Oracle database server. Please refer to the Oracle documentation for information on using Oracle network configuration utilities to create Oracle network configuration files such as `listener.ora` and `tnsnames.ora`.

Note: Refer to the Creating a Shared Oracle Listener for Multiple Resources section in the appendix in this document for instructions on how to create a shared Oracle Listener for multiple resources.

Listener Configuration

1.
2. You need to choose a switchable IP address for clients to make connections to. You may want to put this address in DNS. (Refer to the LifeKeeper IP Recovery Kit Documentation for details on creating an IP resource hierarchy. Refer to the topic Creating a Resource Dependency under GUI Administration Tasks for details on creating a resource dependency).

3. In the `listener.ora` file, specify this switchable IP address as the HOST for the database service name. (See the Oracle documentation for details about the `listener.ora` file.) Although the DNS name can be used in place of the switchable IP address for the HOST database service name, LifeKeeper best practices does not recommend this. Using the switchable IP address will prevent DNS lookup issues from impacting LifeKeeper's ability to determine the status of a running listener during quickCheck, restore or remove processing. Additionaly, a SID_LIST_LISTENER stanza must be defined, even though you may have only one listener defined.

   Sample format of a `listener.ora`:

```
.
.
.
SID_LIST_LISTENER =
     (SID_LIST =
     (SID_DESC =
              (SID_NAME = <SID Name>)
       )
)
.
.
.
<listener name>=
     (DESCRIPTION_LIST =
```

```
         (DESCRIPTION =
              (ADDRESS = (PROTOCOL = TCP)(HOST = <switchable IP>)(PORT
= <port number>))
       )
)
.
.
.
```

4. Specify the switchable IP address as the HOST in the `tnsnames.ora` file or Oracle Names. (See the Oracle documentation for details about the `tnsnames.ora` file.) Although the DNS name can be used in place of the switchable IP address for the HOST database service name, LifeKeeper best practices does not recommend this. Using the switchable IP address will prevent DNS lookup issues from impacting LifeKeeper's ability to determine the status of a running listener during quickCheck, restore or remove processing:

```
    .
    .
    .
    <SID Name>=
       (DESCRIPTION =
          (ADDRESS_LIST =
             (ADDRESS = (PROTOCOL = TCP)(HOST = <switchable IP>)(PORT
= <port number>))
             )
             (CONNECT_DATA =
             (SID = <SID Name>)
          )
       )
```

These sample files should work with both Oracle 10g and 11g:

`listener.ora`

```
SID_LIST_LISTENER =
   (SID_LIST =
   (SID_DESC =
      (SID_NAME = ORA11A)
   )
)
LISTENER =
   (DESCRIPTION_LIST =
      (DESCRIPTION =
         (ADDRESS = (PROTOCOL = TCP)(HOST = 192.0.2.0)(PORT = 1521))
```

```
      )
   )
```

tnsnames.ora

```
ORA01 =
   (DESCRIPTION =
      (ADDRESS_LIST =
         (ADDRESS = (PROTOCOL = TCP)(HOST = 192.0.2.0)(PORT = 1521))
      )
         (CONNECT_DATA =
         (SID = ORA01)
      )
   )
```

The normal location of `listener.ora` is in *$ORACLE_HOME/network/admin*. The most common port number is 1521. The global name of the database was defined at creation time. Also keep in mind, if the *$ORACLE_HOME* directory is installed on non-shared storage, a copy of `listener.ora` will need to be on both systems.

Note: Oracle Net provides the option of automatically failing over client connections to another listener if the listener for a service should fail. To take advantage of this feature, set the `FAILOVER` parameter to "**ON**" in the `tnsnames.ora` file. If the listener for the LifeKeeper-protected Oracle SID should fail, this allows client connections to continue through another listener until LifeKeeper recovers the protected listener.

6.11.2.3. Configuring Transparent Application Failover with LifeKeeper

When a server failover or an Oracle database failure occurs, users can be severely disrupted. Typically the user's connections to the database will be lost along with most work in progress. Upon the completion of the failover (or recovery of the Oracle database), clients will have to restart their application and reconnect to the database. With the Transparent Application Failover (TAF) feature of Oracle, this disruption can be reduced or eliminated by masking some types of failures. To configure TAF in a LifeKeeper environment, there are tasks that must be performed on both the LifeKeeper server side and the Oracle client side.

For clients to effectively take advantage of the TAF feature, the client application must use failover-aware API calls from the Oracle Call Interface (OCI). The clients must also configure the appropriate TAF support using the Oracle Net parameters in the `tnsnames.ora` file. TAF mode can be configured by including a FAILOVER_MODE parameter under the CONNECT_DATA section of the `tnsnames.ora` connect descriptor. The TAF mechanism supports several sub-parameters to control and affect the behavior of a client connection during failover. The LifeKeeper for Linux Oracle Recovery Kit supports the following TAF configuration sub-parameters:

TYPE= (SELECT or SESSION).

This value determines how TAF will handle client connection failover. When the type is set to **SELECT**, Oracle keeps track of all select statements issued during transition. Upon establishment of a new connection, the select statements are re-executed, and the cursors repositioned so clients can continue to fetch rows. When type is set to SESSION only a new connection is created; work in progress may be lost.

METHOD= (BASIC).

With this method TAF will attempt a reconnect only after the primary connection fails. The alternative method is **PRECONNECT**, LifeKeeper does not currently support the use of PRECONNECT as a method.

DELAY= (#sec).

This value is the number of seconds that TAF will wait between attempts to connect following a failure. This value should be carefully determined for your client application and environment.

RETRIES= (#number of tries).

This value is the number of times that TAF will attempt to retry a failed connection before giving up. The combination of **DELAY** and **RETRIES** must allow enough time for a complete recovery of Oracle in the event of a server failure. This will give TAF enough time to restart after the server failover has completed.

An excerpt from a sample tnsnames.ora file for a client system is included below.

```
LKproDB=
    (DESCRIPTION=
        (ADDRESS_LIST=
        (ADDRESS = (PROTOCOL=TCP) (HOST=<switchableIP>) (PORT=<port number>))
        )
        (CONNECT_DATA=
            (SID=LKroDB)
            (SERVER=DEDICATED)
            (FAILOVER_MODE=
                (TYPE=SELECT)
                (METHOD=BASIC)
                (DELAY=5)
                (RETRIES=30)
            )
        )
    )
```

The normal location of `tnsnames.ora` is in *$ORACLE_HOME/network/admin*. The most common port number is 1521. The `tnsnames.ora` files can also be located in user's home directories as well. Also, keep in mind, if the *$ORACLE_HOME* directory has been installed on non-shared storage, a copy of `listener.ora` and `tnsnames.ora` will need to be on both systems.

On the LifeKeeper server protecting the Oracle database, the listener should be configured using a LifeKeeper-protected switchable IP address. Refer to the Configuring the Oracle Net Listener for LifeKeeper Protection section above for details on configuring Oracle Net and listener support.

6.11.2.4. Configuring a Pluggable Database with Oracle Multitenant

LifeKeeper can protect the pluggable database (“PDB”) as well as the Oracle database server provided that the Oracle database supports the Oracle Multitenant architecture and protects the container database (“CDB”),

Checking CDB and PDB

1. Oracle resources must be created to protect the PDB. Also, the protected Oracle resource must be a CDB. You can check whether it is a CDB or not after connecting to the database using the following command.

   ```
   SQL> select CDB from V$DATABASE;
   ```

2. To protect the PDB, the PDB must be mounted inside the CDB. You can check whether the PDB is mounted by using the following command after connecting to the database.

   ```
   SQL> show pdbs;
   ```

Creating Oracle PDB Resources

1. From the LifeKeeper GUI menu, select **Edit**, then select **Server**. From the drop-down menu, select **Create Resource Hierarchy**.

Important: When you create a resource, the Oracle resource must have been created and be in service.

A dialog box appears displaying all the recognized Recovery Kits installed in the cluster in a dropdown list. Select **Oracle Pluggable Database** from the dropdown list. Click **Next** to proceed to the next dialog box.

Note: If the Back **button** is active in a dialog box, you can return to the previous dialog box by clicking it. This is especially useful when you encounter errors and need to correct the information you entered earlier.

At any stage of the hierarchy creation process, clicking **Cancel** will cancel the entire creation process.

2. You will be prompted to enter the following information. If the **Back** button is active in a dialog box, you can return to the previous dialog box. This is useful when you encounter errors and need to correct the information you entered earlier. You can click **Cancel** at any time to cancel the entire creation process.

Field	Description
Server	Select the LifeKeeper server on which Oracle PDB is created.
Switchback Type	Select intelligent or automatic. After the failover, when the Oracle PDB resource is brought back in service (active) on the backup server, how it is switched back to the primary server is determined. Intelligent switchback (intelligent) requires administrator intervention to switch resources back to the primary server, while automatic switchback (automatic) brings the primary server back online and a switchback is performed as soon as the LifeKeeper communication path is reestablished. **Note:** The switchback method must be the same as the one for dependent resources used by the Oracle PDB resource.
ORACLE_SID	Specify the SID of the protected Oracle database.
Oracle PDBs	Specify the PDB to protect. This field allows multiple selections.
PDB Tag	A unique tag name for the new Oracle PDB resource on the primary server. The default tag name is “pdb-<ORACLE_SID>”. You can also use another unique tag name. You can use letters, numbers, and the special symbols (“:”, “-“, “_”, “.”, “/”) for tag names.

3. Click **Next**. **Create Resource Wizard** appears and the Oracle PDB resource hierarchy is created. LifeKeeper verifies the input data. If a problem is detected, an error message appears in the information box.

4. A message saying that the Oracle PDB resource hierarchy has been successfully created and that the hierarchy must be extended to another server in the cluster to provide failover protection is displayed. Click **Next**.

5. Click **Continue**. The **Pre-extend** Wizard is launched. See Step 2 in “Extending the Oracle PDB Resource Hierarchy” for details on extending the resource hierarchy to another server.

Extending Oracle PDB Resources

1. Select **Extend Resource Hierarchy** from **Resource** in the **Edit** menu. The **Pre-Extend Wizard** is displayed. If you are not familiar with advanced operations, click **Next**. If you understand the default values for extending the LifeKeeper resource hierarchy and do not need to enter and confirm them, click **Accept Defaults**.

2. Enter the following details in the **Pre-Extend Wizard**.

   Note: The first two fields appear only when you start the operations from **Extend** in the **Edit** menu.

Field	Description
Template Server	Select the server where the Oracle PDB resource is currently in service.
Tag to Extend	Select the Oracle PDB resource to extend.

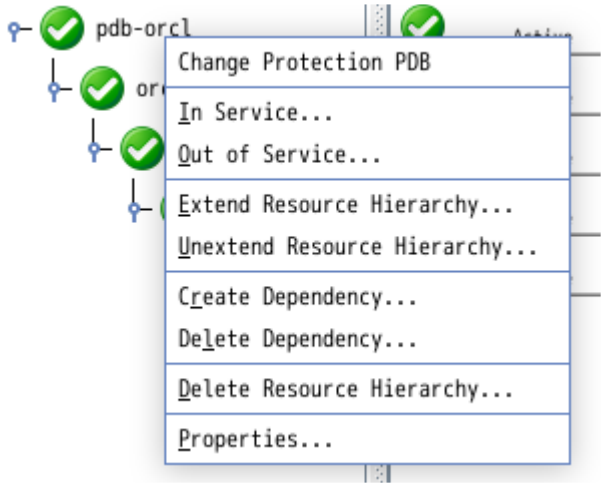
Target Server	Enter or select the target server.
Switchback Type	After the failover, when the Oracle PDB is brought in service (active) on the backup server, how it is switched back to the primary server is determined. You can choose intelligent or automatic. The switchback type can be changed later on the **General** tab of the **Resource Properties** dialog box when necessary. **Note:** The switchback method must be the same as the one for the dependent resource used by the Oracle PDB resource.
Template Priority	Select or enter a **template priority**. This is the priority of the Oracle PDB hierarchy which has currently been in service on the server. You can use any unused number between 1 and 999 for the priority, with lower numbers having higher priority (number 1 is the highest priority). During the extension process, priorities that are already in use by another system cannot be specified for this hierarchy. SIOS recommends the default value. **Note:** This field appears only when you extend the hierarchy for the first time.
Target Priority	This is the relative priority which is owned by the newly extending Oracle PDB hierarchy over the equivalent hierarchies on other servers. Any unused priority number between 1 and 999 is available and indicates the server's priority for the resource cascading failover sequence. Note that LifeKeeper assigns "1" by default to the server on which the hierarchy was created. The priorities do not need to be consecutive but two servers cannot have the same priority for a particular resource.

3. When the pre-extending checking is successful message is displayed, click **Next**.
4. Depending on the hierarchy to extend, a series of information boxes will be displayed showing the resource tags to be extended (some cannot be edited).
5. Confirm that the tag name is correct in **Extend Wizard** and click **Extend**.
6. When the message "Hierarchy extend operations completed" is displayed, click **Next Server** if you want to extend the hierarchy to another server, or click **Finish**.
7. When the message "Hierarchy Verification Finished" is displayed, click **Done**.

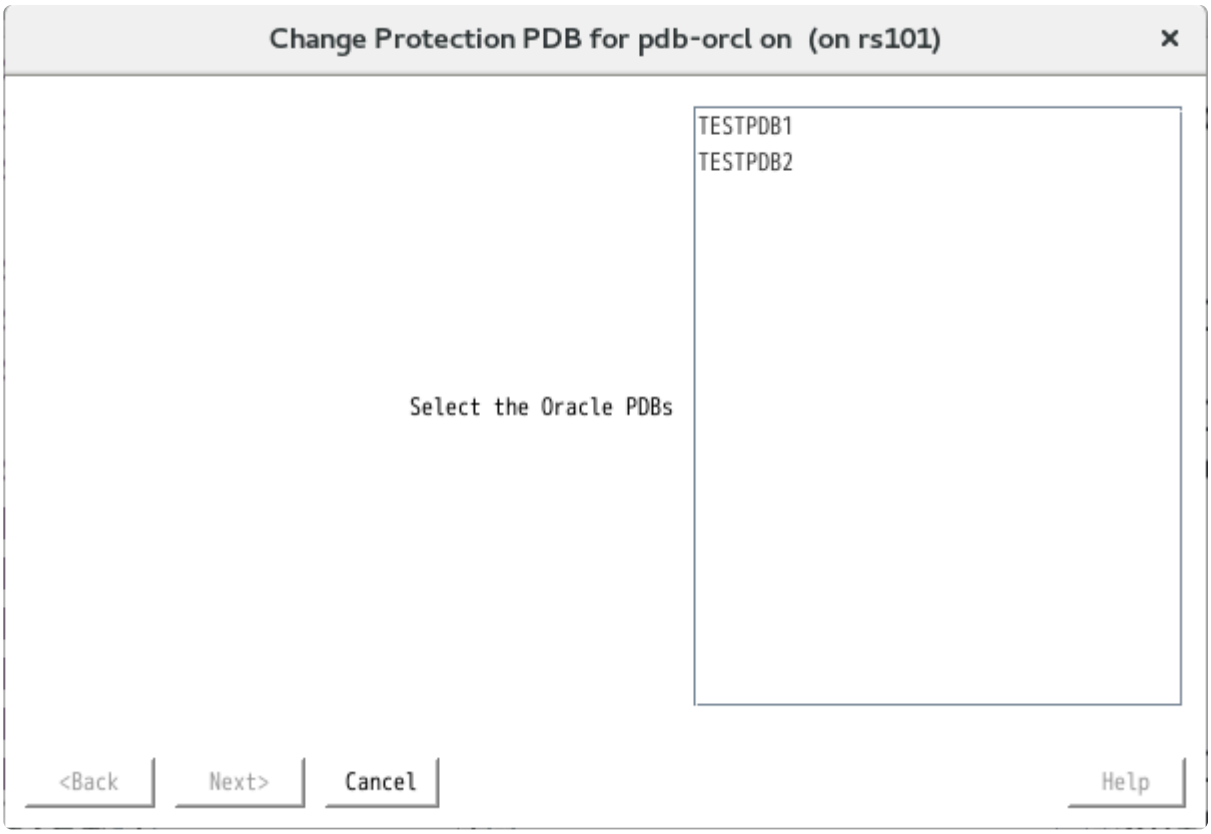
Changing the PDB to Protect

After creating the hierarchy, change the PDB to protect using the following steps.

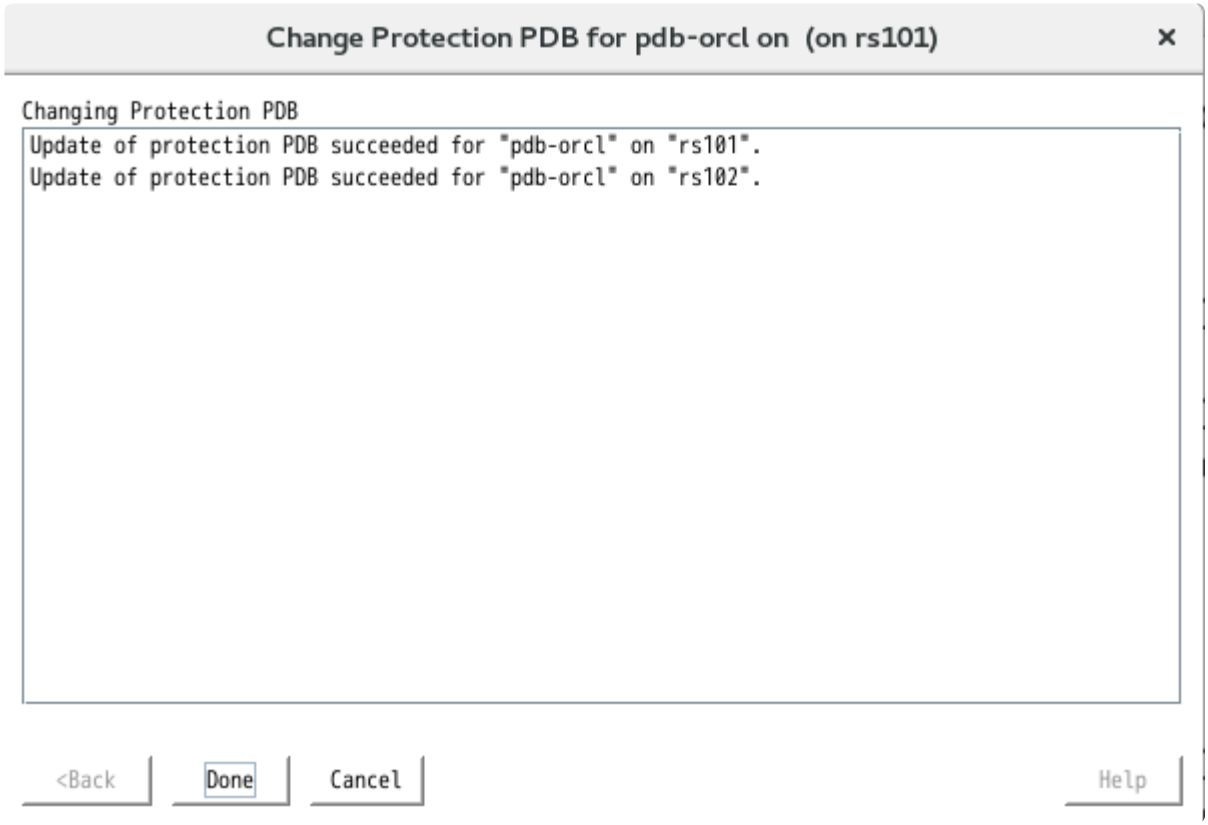
1. From the LifeKeeper GUI, right-click the Oracle PDB resource hierarchy and select **Change Protection PDB**.



2. Select the PDB you want to protect (you can select more than one PDB).



3. Click **Next** to change the settings.



4. Click **Done** to finish.

6.11.2.5. Oracle Configuration Examples

The following figures illustrate examples of both active/standby and active/active Oracle configurations in an SPS environment.

The examples in this section show how Oracle database instances can be configured on local and shared disks. Each diagram shows the relationship between the type of configuration and the Oracle parameters. Each configuration also adheres to the configuration rules and requirements described in this administration guide that ensure compatibility between the Oracle configuration and the LifeKeeper software.

This section first describes the configuration requirements and then provides these configuration examples:

- Active/Standby
- Active/Active

The examples in this section are only a sample of the configurations you could establish, but understanding these configurations and adhering to the configuration rules helps you define and set up workable solutions for your computing environment.

6.11.2.5.1. Oracle Configuration Requirements

Each of the examples involves one or two databases: **databaseA** and **databaseB**. By default, LifeKeeper offers a tag name matching the Oracle database system identifier (SID). However, the screen examples in the following pages use tag names consisting of the SID and server name such as databaseA-on-server1.

To understand the configuration examples, keep these configuration requirements in mind:

- **LifeKeeper hierarchy.** When performing LifeKeeper administration, the primary server refers to the location where the Oracle instance is currently running. System administration takes place on this server when creating a LifeKeeper hierarchy. For the configuration examples, the primary server is Server 1 and the backup or alternate server is Server 2.

- **Shared disk locked by one server.** When shared storage resources are under LifeKeeper protection, they can only be accessed by one server at a time. If the shared device is a disk array, an entire LUN is protected. If a shared device is a disk, then the entire disk is protected. This prevents inadvertent corruption of the data by other servers in the cluster. When a server fails, the highest priority backup server establishes its own protection, locking out all other servers.

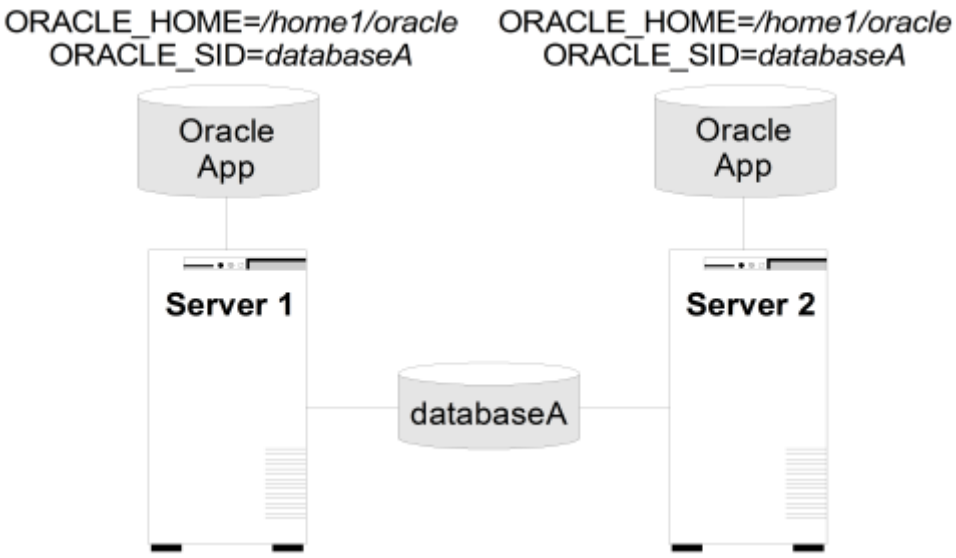
- **Database on shared disk.** In order for the LifeKeeper Oracle Recovery Kit to function properly, the database must always be on a shared device. The database may be on one or more file systems and/or disks.

Note: The tables associated with each configuration illustration provide examples of the appropriate information that would be entered in the Create Resource Hierarchy wizard on the primary server (Server 1) and Extend Resource Hierarchy wizard to the backup server (Server 2). For additional detail on what information to enter into the wizards, refer to the “LifeKeeper Configuration Tasks” section. These tables can be a helpful reference when configuring your Recovery Kit.

6.11.2.5.2. Oracle Active/Standby Configurations

This section provides two active/standby configuration examples, shown in Figure 1and Figure 2. In these configurations, Server 1 is considered active because it has exclusive access to the database. Server 2 does other processing. If Server 1 fails, Server 2 gains access to the database and LifeKeeper re-establishes the database operations.

Figure 1. Active/Standby Configuration, Example 1



Configuration Notes:

Each server has its own $ORACLE_HOME directory on a non-shared disk. Each server has the same version of the Oracle application.

The $ORACLE_HOME path is the same on both servers.

The database, databaseA, is on a shared disk.

Creating a resource hierarchy on Server 1:

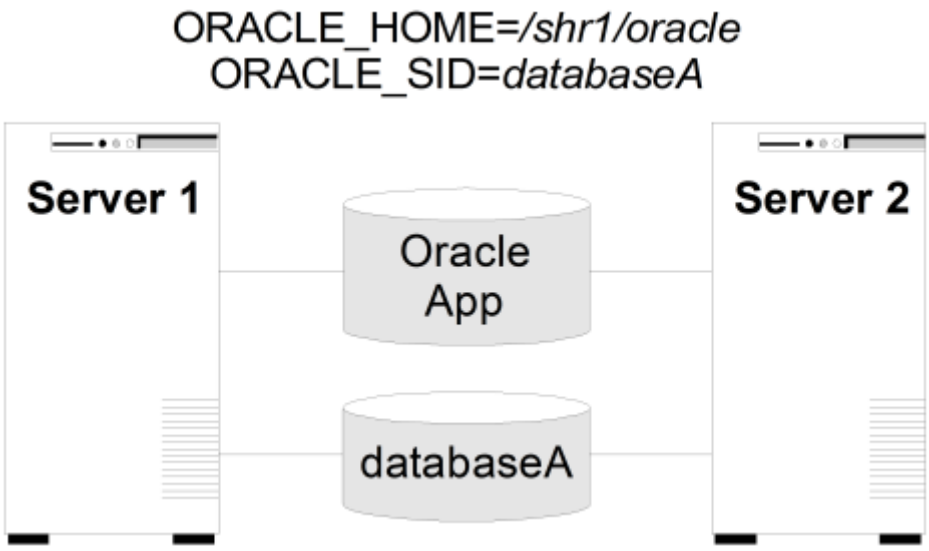
Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database:	system
Password for Username:	**********
ORACLE_SID for Database:	/home1/oracle
Database Tag:	databaseA-on-server1

Extending the resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend:	databaseA-on-server1

Target Server:	Server2
Target Priority:	10
Database Tag:	databaseA-on-server1

Figure 2. Active/Standby Configuration, Example 2



Configuration Notes:

1.
2. Both servers use the $ORACLE_HOME directory on a shared disk.
3. The $ORACLE_HOME path is the same on both servers.
4. The database, databaseA, is on a shared disk.
5. Server 2 can not access files and directories on the shared disk while Server 1 is active.
6. $ORACLE HOME can be on the same shared disk as the database or on separate disks.

Creating a resource hierarchy on Server 1:

Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database:	system
Password for Username:	**********
ORACLE_HOME for Database:	/shr1/oracle
Database Tag:	databaseA-on-server1

Extending the resource hierarchy to Server 2:

Template Server:	Server1

Tag to Extend:	databaseA-on-server1
Target Server:	Server2
Target Priority:	10
Database Tag:	databaseA-on-server1

6.11.2.5.3. Oracle Active/Active Configurations

An active/active configuration consists of at least two servers, each running a different database instance. The databases must be on different shared disks.

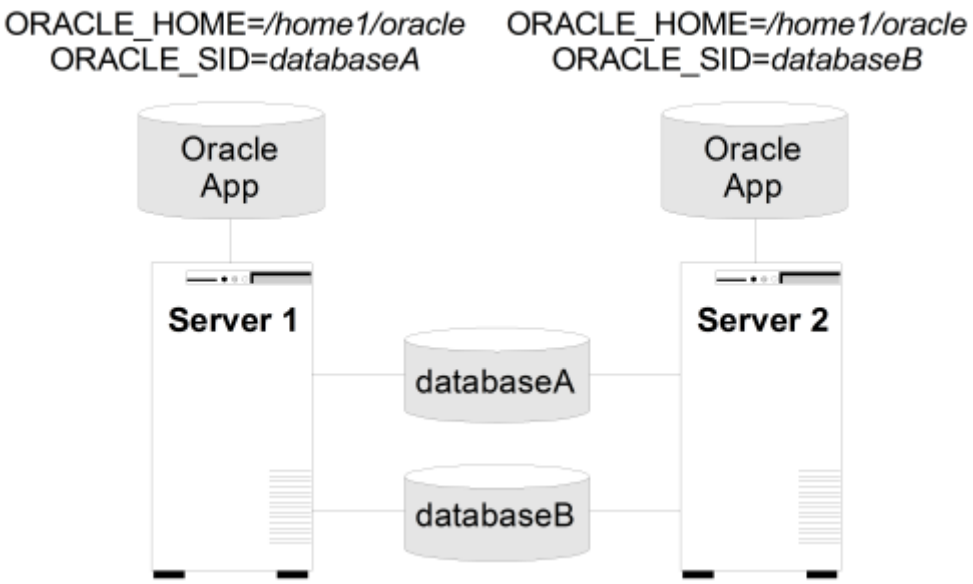
$ORACLE_HOME can be on non-shared or on shared disks depending upon the configuration requirements. For example, multiple database instances on any of the servers using a common $ORACLE_HOME require $ORACLE_HOME to be on non-shared disks. If the $ORACLE_HOME directories are on shared disk, they must be on separate shared disks.

This section provides two active/active configuration examples, shown in Figure 3 and Figure 4:

- Databases on shared resources and a common $ORACLE_HOME on non-shared resources.
- Databases on shared resources and the appropriate $ORACLE_HOME instance on the same shared resource.

Note: Multiple database instances on one server using multiple instances of $ORACLE_HOME on non-shared resources are not illustrated.

Figure 3. Active/Active Configuration, Example 1



Configuration Notes:

1.
2. The server has its own $ORACLE_HOME directory on a non-shared disk. Each server has the same version of the Oracle application.
3. The $ORACLE_HOME path is the same on both servers.

4. The databases, databaseA and databaseB, are on shared disks.

5. The oratab file exists in /etc/ on both servers, containing entries for both Oracle instances.

6. Initially, Server 1 runs databaseA and Server 2 runs databaseB. In a switchover situation, one system can run both databases.

7. See Creating Oracle Database Hierarchy After Installing Oracle Binaries on Local Storage for further information.

Creating the first resource hierarchy on Server 1:

Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database:	system
Password for Username:	**********
ORACLE_HOME for Database:	/home1/oracle
Database Tag:	databaseA-on-server1

Extending the first resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend:	databaseA-on-server1
Target Server:	Server2
Target Priority:	10
Database Tag:	databaseA-on-server1

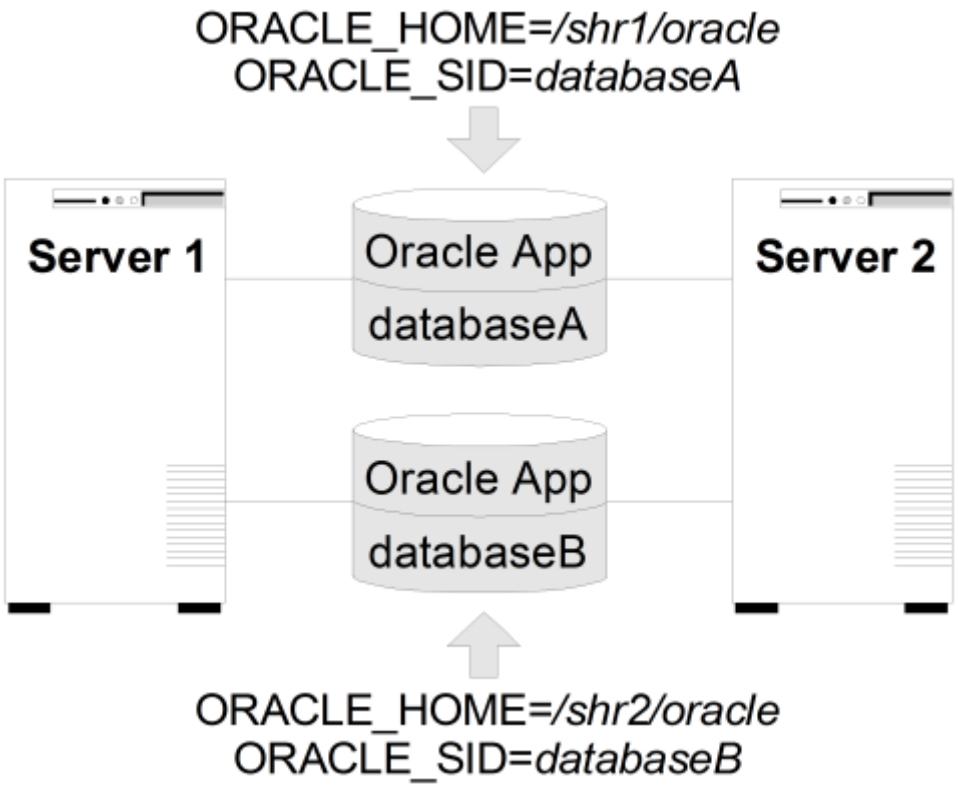
Creating a second resource hierarchy on Server 2:

Server:	Server2
ORACLE_SID for Database:	databaseB
Username for Database:	system
Password for Username:	**********
ORACLE_HOME for Database:	/home1/oracle
Database Tag:	databaseB-on-server2

Extending the second resource hierarchy to Server 1::

Template Server:	Server2
Tag to Extend:	databaseB-on-server2
Target Server:	Server1
Target Priority:	10
Database Tag:	databaseB-on-server2

Figure 4. Active/Active Configuration, Example 2



Configuration Notes:

1.
2. Both servers use an $ORACLE_HOME directory on different shared disks.

3. The Oracle application is the same on both servers. The $ORACLE_HOME is different for each instance defined on the server.

4. The databases, databaseA and databaseB, are on shared disks.

5. The oratab file exists in /etc/, containing entries for both Oracle instances.

6. A unique login is required for each Oracle instance. The id and gid for each login should be the same on Server 1 and Server 2.

7. Initially, Server 1 runs databaseA and Server 2 runs databaseB. In a switchover situation, one system can run both databases.

Creating the first resource hierarchy on Server 1:

Server:	Server1
ORACLE_SID for Database:	databaseA
Username for Database	system
Password for Username	**********
ORACLE_HOME for Database:	/shr1/oracle

Database Tag:	databaseA-on-server1

Extending the first resource hierarchy to Server 2:

Template Server:	Server1
Tag to Extend	databaseA-on-server1
Target Server	Server2
Target Priority:	10
Database Tag:	databaseA-on-server1

Creating a second resource hierarchy on Server 2:

Server:	Server2
ORACLE_SID for Database:	databaseB
Username for Database	system
Password for Username	**********
ORACLE_HOME for Database:	/shr2/oracle
Database Tag:	databaseB-on-server2

Extending the second resource hierarchy to Server 1:

Template Server:	Server2
Tag to Extend	databaseB-on-server2
Target Server	Server1
Target Priority:	10
Database Tag:	databaseB-on-server2

6.11.3. LifeKeeper Configuration Tasks for Oracle

You can perform the following configuration tasks from the LifeKeeper GUI. The following four tasks are described in this guide, as they are unique to an Oracle resource instance and different for each Recovery Kit.

- Create a Resource Hierarchy. Creates an application resource hierarchy in your LifeKeeper cluster.
- Delete a Resource Hierarchy. Deletes a resource hierarchy from all servers in your LifeKeeper cluster.
- Extend a Resource Hierarchy. Extends a resource hierarchy from the primary server to a backup server.
- Unextend a Resource Hierarchy. Unextends (removes) a resource hierarchy from a single server in the LifeKeeper cluster.
- View Oracle Configuration Settings – Allows viewing of the Resource Properties dialog.
- Change Username / Password. Change the Username and Password to login to protect Oracle Database.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View Properties. View the properties of a resource hierarchy on a specific server.
- Edit Properties. Edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, we explain how to configure your Recovery Kit by selecting certain tasks from the **Edit** menu of the LifeKeeper GUI. You can also select each configuration task from the toolbar. You may also right-click on a global resource in the Resource Hierarchy Tree (left-hand

pane) of the status display window to display the same drop down menu choices as the Edit menu. This, of course, is only an option when a hierarchy already exists.

You can also right-click on a resource instance in the Resource Hierarchy Table (right-hand pane) of the status display window to perform all the configuration tasks, except *Creating a Resource Hierarchy*, depending on the state of the server and the particular resource.

6.11.3.1. Creating an Oracle Resource Hierarchy

Note: In order to take advantage of Oracle Net remote client access, the IP address used for client connectivity must be under LifeKeeper protection as a dependent of the Oracle hierarchy. (Refer to the section Configuring the Oracle Net Listener for LifeKeeper Protection for details.)

To create a resource instance from the primary server, you should complete the following steps:

1.
2. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop down menu, select **Create Resource Hierarchy**.

   Important: The Oracle Application must be running when you create the resource.

   A dialog box will appear with a drop down list box with all recognized Recovery Kits installed within the cluster. Select **Oracle Database** from the drop down listing. Click **Next** to proceed to the next dialog box.

   Note: When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information.

   If you click **Cancel** at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

3. Select the **Switchback Type**. This dictates how the Oracle instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either intelligent or automatic. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths.

   The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box. Click **Next** to proceed to the next dialog box.

4. Select the **Server** where you want to place the Oracle Database (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop down box. Click **Next** to proceed to the next dialog box.

5. Select the **ORACLE_SID** for the Database ID. This is the tag name that specifies the Oracle system identifier of the database being configured. An entry for this database must exist in */etc/oratab*. Click **Next** to proceed to the next dialog box.

6. Input the **Username for ORACLE_SID**. This is the Oracle Database Username specified during

login to ORACLE_SID. This username must be able to connect as sysdba authority to the database to gain full control. Click **Next** to proceed to the next dialog box. (This field can be left empty. If left empty, LifeKeeper will not use Username and Password to control the Oracle Database resource, and the next step, **Input Password**, will be skipped.)

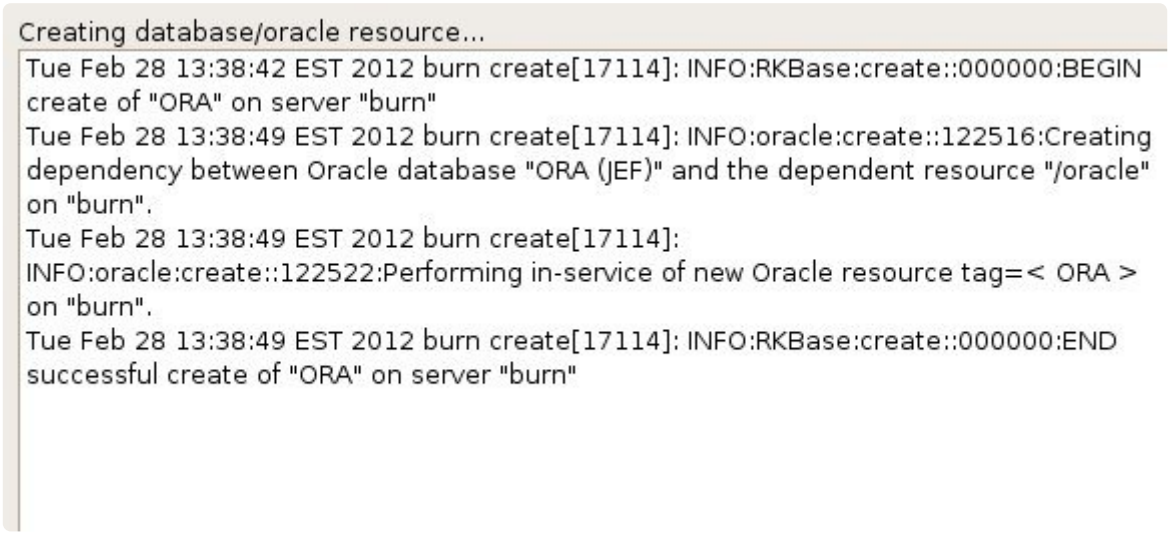
7. Input **Password**. This is the password specified during login to ORACLE_SID. The password will be saved by LifeKeeper with encrypting. Click Next to proceed to the next dialog box.

8. Select the **tag name** of the Listener to be included as a dependency of the Oracle resource. The list displays all the currently protected Listener resource(s) on the server. Select the Listener resource tag that corresponds to the required listener(s) for the Oracle SID. Select **None** if no Listener resource exists.

9. Select or enter the **Database Tag**. This is a tag name that LifeKeeper gives to the Oracle hierarchy. You can select the default or enter your own tag name.

   When you click **Create**, the ***Create Resource Wizard*** will create your Oracle resource.

10. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your Oracle resource hierarchy. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created.



   Click **Next** to proceed to the **Pre-extend dialog box** which is explained later in this documentation. You must extend the hierarchy to another server in your cluster in order to place it under LifeKeeper protection.

6.11.3.2. Deleting an Oracle Resource Hierarchy

To delete a resource hierarchy from all the servers in your SPS environment, complete the following steps:

1.
2. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop down menu, select **Delete Resource Hierarchy**.

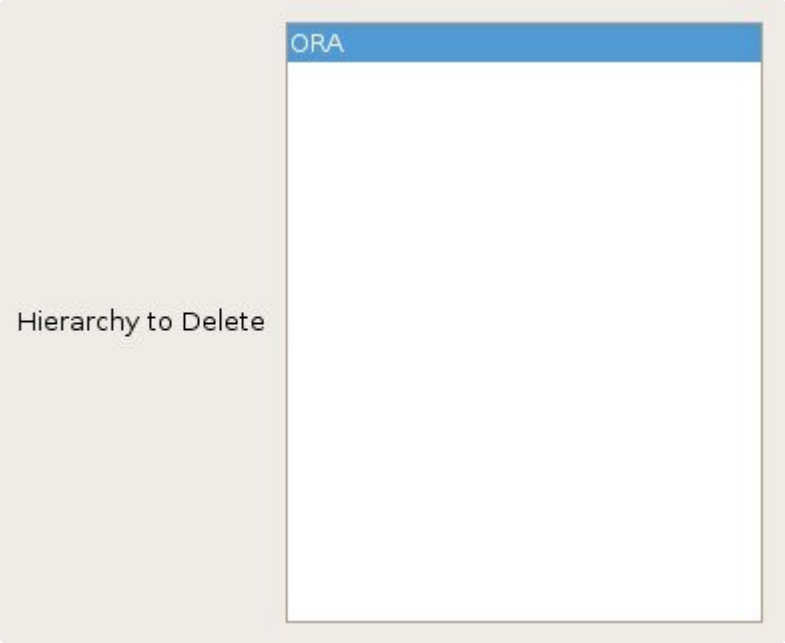
3. Select the name of the **Target Server** where you will be deleting your Oracle resource hierarchy.

   Note: If you selected the Delete Resource task by right-clicking from the right pane on an individual resource instance, or from the left pane on a global resource where the resource is on only one server this dialog box will not appear.

   Click **Next** to proceed to the next dialog box.

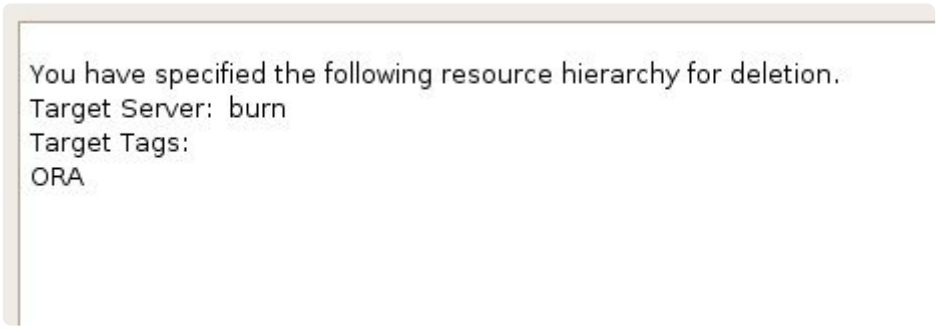
4. Select the Hierarchy to Delete. Identify the resource hierarchy you wish to delete, and highlight it.

   Note: If you selected the Delete Resource task by right clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.



   Click **Next** to proceed to the next dialog box.

5. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete.



Click **Delete** to delete your resource and proceed to the final dialog box.

6. Another information box appears confirming that the Oracle resource was deleted successfully.

7. Click **Done** to exit out of the **Delete Resource Hierarchy** menu selection.

Note: Refer to the Creating a Shared Oracle Listener for Multiple Resources section in the appendix of this document for instructions on how to create a shared Oracle Listener for multiple resources.

6.11.3.3. Extending Your Oracle Hierarchy

After you have created a hierarchy, you will want to extend that hierarchy to another server in the cluster. There are three possible scenarios to extend your resource instance from the template server to a target server. The first scenario is when you "**Continue**" from creating the resource into extending that resource to another server by clicking **Next** on the information dialog box displayed at the completion of the create. The second scenario is when you enter the **Extend Resource Hierarchy** task from the edit menu as shown below. The third scenario is when you right-click on an unextended hierarchy in either the left- or right-hand pane. Each scenario takes you through the same dialog boxes (with a few exceptions, which are clearly detailed below).

1.
2. If you are entering the **Extend Wizard** from the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the drop-down menu, select **Extend Resource Hierarchy**. This will launch the ***Pre-Extend Resource Hierarchy Wizard***.

3. The first dialog box to appear will ask you to select the **Template Server** where your Oracle resource hierarchy is currently in service. It is important to remember that the **Template Server** you select now and the **Tag to Extend** that you select in the next dialog box represent an in service resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you selected. The drop-down box in this dialog provides the names of all the servers in your cluster.

   Note: If you are entering the **Pre-Extend Resource Hierarchy** task immediately following the creation of an Oracle resource hierarchy, this dialog box will not appear, since the wizard has already identified the template server in the create stage. This is also the case when you right-click on either the Oracle resource icon in the left hand pane or right-click on the Oracle resource box in the right hand pane the of the GUI window and choose **Extend Resource Hierarchy**.

   It should be noted that if you click the **Cancel** button at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extension process to that particular server. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

   For example, let's say you have created your resource on Server 1 and extended that resource to Server 2. In the middle of extending the same resource to Server 3, you change your mind and click on the **Cancel** button inside one of the dialog boxes. This will cancel only your action to extend the resource to Server 3, not the extension you created to Server 2. If you want to remove Server 2 from this hierarchy, you must unextend the resource from Server 2.

   Click **Next** to proceed to the next dialog box.

4. Select the **Tag to Extend**. This is the name of the Oracle instance you wish to extend from the template server to the target server. The wizard will list in the drop-down box all the resources that you have created on the template server, which you selected in the previous dialog box.

   Note: Once again, if you are entering the Pre-Extend Resource Hierarchy task immediately following the creation of an Oracle resource hierarchy, **this dialog box will not appear,** since the wizard has already identified the tag name of your Oracle resource in the create stage. This is also the case when you right-click on either the Oracle resource icon in the left hand pane or on the Oracle resource box in the right hand pane the of the GUI window and choose **Extend Resource Hierarchy**.

   Click **Next** to proceed to the next dialog box.

5. Select the **Target Server** where you are extending your Oracle resource hierarchy. The drop-down box provides the names of the servers in your cluster that are not already in the selected hierarchy.

   Click **Next** to proceed to the next dialog box.

6. Select the **Switchback Type**. This dictates how the Oracle instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either *intelligent* or *automatic*. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths.

   The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box.

   Click **Next** to proceed to the next dialog box.

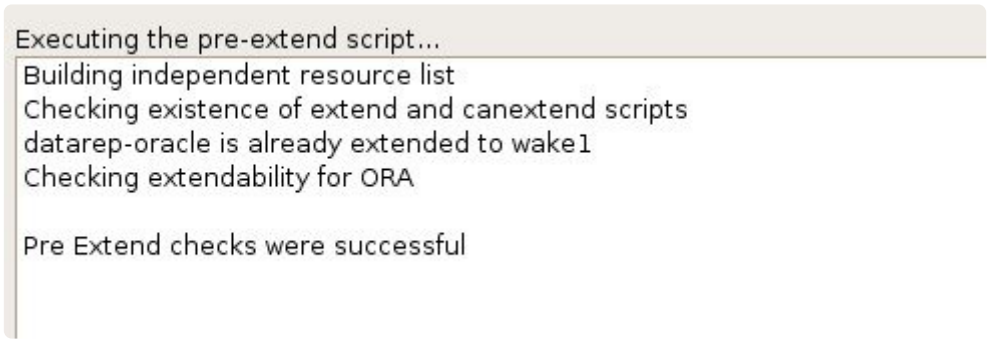
7. Select or enter a **Template Priority**. This is the priority for the Oracle hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. Note: This

selection will appear only for the initial extend of the hierarchy.

8. Select or enter the **Target Priority**. This is the priority for the new extended Oracle hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

   Click **Next**.

9. An information box will appear explaining that LifeKeeper has successfully checked your environment and that all the requirements for extending this Oracle resource have been met. If there were some requirements that had not been met, LifeKeeper would not allow you to select the Next button, and the Back button would be enabled.



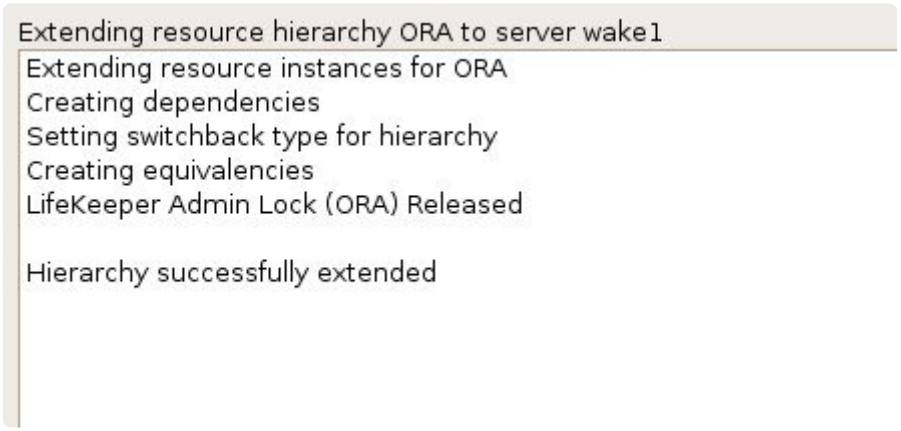
   If you click **Back**, you can make changes to your resource extension according to any error messages that may appear in the information box.

   If you click **Cancel** now, you will need to come back and extend your Oracle resource hierarchy to another server at some other time to put it under LifeKeeper protection.

   When you click **Next**, LifeKeeper will launch you into the Extend Resource Hierarchy configuration task.

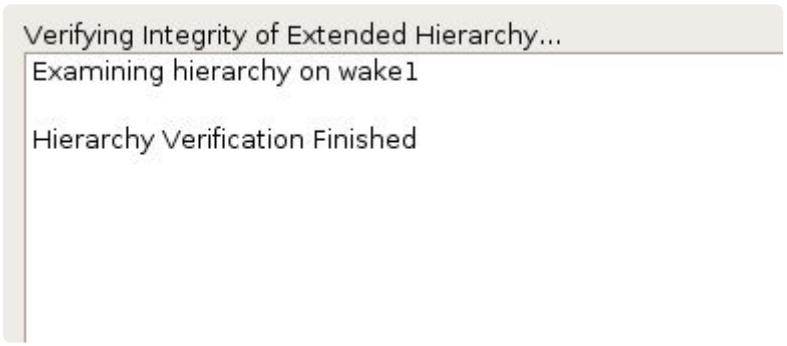
10. Once the Database Tag displays, click **Extend**.

11. An information box will appear verifying that the extension is being performed.



Click **Next Server** if you want to extend the same Oracle resource instance to another server in your cluster. This will repeat the **Extend Resource Hierarchy** operation.

If you click **Finish**, another dialog box will appear confirming LifeKeeper has successfully extended your Oracle resource.



12. Click **Done** to exit from the **Extend Resources Hierarchy** menu selection.

    Note: Be sure to test the functionality of the new instance on both servers.

6.11.3.4. Unextending Your Oracle Hierarchy

1.
2. From the **LifeKeeper GUI** menu, select **Edit**, then **Resource**. From the drop-down menu, select **Unextend Resource Hierarchy**.

3. Select the **Target Server** where you want to unextend the Oracle resource. It cannot be the server where Oracle is currently in service.

   Note: If you selected the **Unextend** task by right-clicking from the right pane on an individual resource instance, this dialog box will not appear.

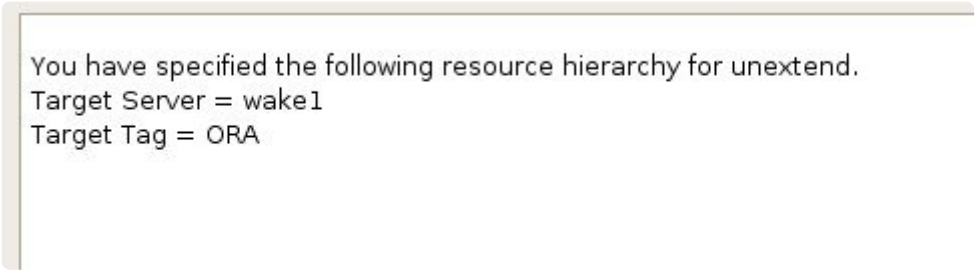
   Click **Next** to proceed to the next dialog box.

4. Select the **Oracle Hierarchy to Unextend**. Note: If you selected the **Unextend** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

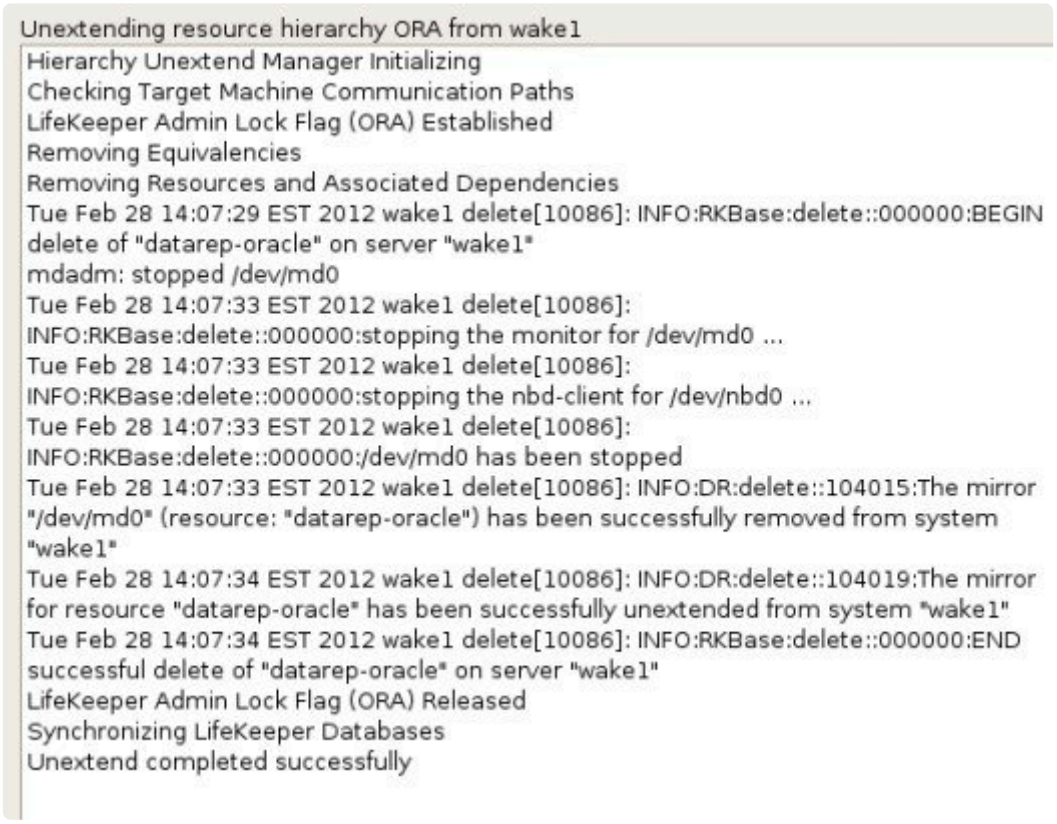
   Click **Next** to proceed to the next dialog box.

5. An information box appears confirming the target server and the Oracle resource hierarchy you have chosen to unextend.

   Click **Unextend**.

6. Another information box appears confirming that the Oracle resource was unextended successfully.



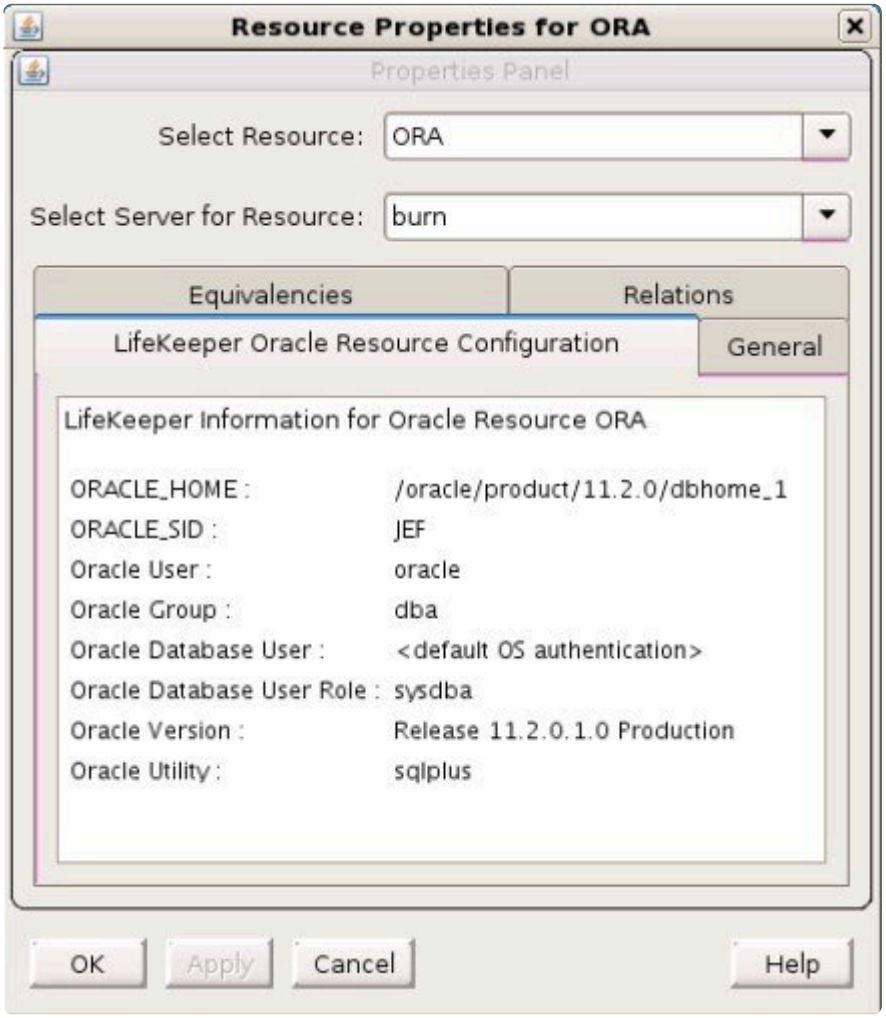
7. Click **Done** to exit out of the **Unextend Resource Hierarchy** menu selection.

6.11.3.5. Viewing Oracle Configuration Settings

The **Resource Properties** dialog is available from the **Edit** menu or from a resource context menu. This dialog displays the properties for a particular resource on a server. When accessed from the **Edit** menu, you can select the resource and the server. When accessed from a resource context menu, you can select the server.

From the **Configuration** tab, you can view the following Oracle settings:

- ORACLE_HOME
- ORACLE_SID
- Oracle User
- Oracle Group
- Oracle Database User
- Oracle Database User Role
- Oracle Version
- Oracle Utility



6.11.3.6. Changing Username / Password for the Oracle Database Account

After a hierarchy has been created, change the Username and Password using one of the following procedures.

If $ORACLE_HOME is on shared (or replicated) storage (common in active-passive configurations):

1.
2. On the system where the Oracle database resource is operational, edit the LifeKeeper configuration file */etc/default/LifeKeeper* and add the following line to the file:

   ```
   LK_ORA_NICE=1
   ```

   Do exactly the same on each system in the cluster that has the Oracle resource defined.

3. Use sqlplus to change the Oracle user's password.

   ```
   SQL> ALTER USER {username} IDENTIFIED BY {newpassword};
   ```

4. From the LifeKeeper GUI, right-click on the Oracle Database resource hierarchy, then select **Change Username / Password**.

5. Input **Username** and select **Next**.

6. Input **Password** and select **Next**.

7. Select the **database user role** and click **Apply**. Username and Password will be updated after validating.

8. Select **Done**.

9. Edit the LifeKeeper configuration file on all cluster nodes and make the following change:

   ```
   LK_ORA_NICE=0
   ```

If $ORACLE_HOME is on local storage and each node in the cluster has its own copy of $ORACLE_HOME (common in active-active configurations):

1.
2. On the system where the Oracle database resource is operational, edit the LifeKeeper configuration file /etc/default/LifeKeeper and add the following line to the file:

   ```
   LK_ORA_NICE=1
   ```

   Do exactly the same on each system in the cluster that has the Oracle resource defined.

3. Use `sqlplus` to change the Oracle user's password.

   ```
   SQL> ALTER USER {username} IDENTIFIED BY {newpassword};
   ```

4. From the LifeKeeper GUI, right-click on the Oracle Database resource hierarchy, then select **Change Username / Password**.

5. Input user name to use instead temporarily such as sys into **Username** and select **Next**.

6. Input Password and select Next.

7. Select the database user role and click Apply. Username and Password will be updated after validating.

8. Select **Done**.

9. Put the Oracle database resource "In Service" on one of the backup systems.

10. Once the database is running on the backup system, use `sqlplus` to change the Oracle account password.

    ```
    SQL> ALTER USER {username} IDENTIFIED BY {newpassword};
    ```

    When making this password change, use the new password that was set in Step 2. This process resets the security tokens in $ORACLE_HOME.

11. Put the database "In Service" on each node in the cluster and repeat Step 8.

12. Once the passwords have been changed on all cluster nodes, put the Oracle database back "In Service" on the desired node.

13. From the LifeKeeper GUI, right-click on the Oracle Database resource hierarchy, then select **Change Username / Password**.

14. Input **Username** and select **Next**.

15. Input **Password** and select **Next**.

16. Select the database user role and click **Apply**. Username and Password will be updated after validating.

17. Select **Done**.

18. Edit the LifeKeeper configuration file on all cluster nodes and make the following change:

    ```
    LK_ORA_NICE=0
    ```

6.11.3.7. Testing Your Oracle Resource Hierarchy

You can test your Oracle resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Performing a Manual Switchover from the GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting Edit, then Resource, then finally In Service from the drop down menu. For example, an in service request executed on a backup server causes the application hierarchy to be placed in service on the backup server and taken out of service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the Out of Service request, the application is taken out of service without bringing it in service on the other server.

Recovery Operations

When the primary server fails, the Oracle Recovery Kit software performs the following tasks:

- Brings Oracle into service on the backup server by bringing in-service the logical interface on one of that server's physical network interfaces. (Note: This occurs only when there is an IP resource instance defined as a dependency of the Oracle hierarchy.)
- Mounts the file system on the shared disk on that server.
- Starts the daemon processes related to Oracle.

Since session context is lost following recovery, after the recovery, Oracle users must reconnect using exactly the same procedures they used to connect originally.

6.11.4. Oracle Troubleshooting

The Oracle Recovery Message Catalog below contains listings of all messages that may be encountered while utilizing the Oracle kit.

The Combined Message Catalog provides a listing of all messages that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. This full listing may be searched for any error code received.

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Oracle Message Catalog

Troubleshooting

The Message Catalog provides a listing of all messages that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. This full listing may be searched for any error code received, or you may go directly to the Oracle Kit Message Catalog or the Oracle Listener Message Catalog which contain listings of all messages that may be encountered while utilizing the Oracle Recovery Kit.

- Oracle Known Issues and Restrictions

6.11.4.1. Oracle Known Issues and Restrictions

Control File Switchover Failu

If the $ORACLE_HOME directory does not recover, the database control files may not be set up properly. For automatic switchover, the control files need to be configured on a shared device during the database creation. If you keep the control files on separate servers, you must manually update both servers when you need to implement changes.

Truncated Output

Some versions of Oracle truncate the output when executing the show parameters control_files in sqldba mode. If your version of Oracle exhibits this behavior, verify the following:

- controlfile parameter. Verify that the controlfile parameter resides in the *$Oracle_HOME/dbs/init#SID.ora* file.
- controlfile devices. Verify that the controlfile devices are on a continuous line, with no new lines, and with each device being separated by a comma.

If an Oracle-related device does not configure properly, then the device can be configured manually using the file system applications available under LifeKeeper Application management.

Flash Recovery Destination Located on a Shared Drive

As noted in the configuration section of this document, it is important that the Flash Recovery destination is located on a shared drive. To see where Oracle believes the Flash Recovery Area is, issue the following query (as `SYSDBA`):

```
SQL> SELECT substr(Name,1,30) Name,

  (SPACE_LIMIT/1024/1024/1024) Space_Limit_GB,

       SPACE_USED/1024/1024/1024 Space_Used_GB,

       SPACE_RECLAIMABLE, NUMBER_OF_FILES
```

FROM V$RECOVERY_FILE_DEST;

NAME SPACE_LIMIT_GB SPACE_USED_GB
SPACE_RECLAIMABLE

———————————— ———————————— ————————————- ————————————- ———————————— ————————————-

NUMBER_OF_FILES

———————-

/U01/flash_recovery_area 3.76171875 .156448364 0

4

Following is an example of how to make a change in *$ORACLE_HOME/dbs/spfile<sid>* to complete this task:

```
SQL> ALTER SYSTEM SET
 DB_RECOVERY_FILE_DEST='/oracledb/oracle/flash_recovery_area' scope=both;
```

System altered.

SQL> show parameter DB_RECOVERY_FILE_DEST;

NAME TYPE VALUE

————————————————————————————

db_recovery_file_dest string /oracledb/oracle/flash_recovery_area

db_recovery_file_dest_size big integer 2G

SQL> commit;

Prevent Failover When Unable to Connect to the Database

When resource health checks are performed, the Oracle ARK checks for running database processes and attempts to connect to the database. To prevent a health check failure caused by reaching the maximum allowed connections, set the following in /etc/default/LifeKeeper:

```
LK_ORA_NICE=1
```

Note: Setting `LK_ORA_NICE` can mask other types of connection errors caused by a non-functioning database. Use this setting with caution.

Non-Traditonal Location for *oratab*

By default, the Oracle ARK looks for the oratab file in */etc* followed by */var/opt/oracle*. If the *oratab* file is not located in one of these default locations, then `ORACLE_ORATABLOC` must be set in */etc/default/LifeKeeper* to the directory containing *oratab*.

NFS Version 4 Not Supported

The Oracle Recovery Kit supports NFSv3 for shared database storage. NFSv4 is not supported at this time due to NFSv4 file locking mechanisms.

Creating Oracle Database Hierarchy After Installing Oracle Binaries on Local Storage

If you have elected to install the Oracle binaries (***$ORACLE_BASE***) on local storage on each of your LifeKeeper cluster nodes, you will see a message similar to the following when you create your Oracle database hierarchy in the LifeKeeper GUI.

```
BEGIN create of <SID> on server <server1>
Creating resource instance <SID> on server <server1>
Setting resource state for <SID> on server <server1> to "ISP".

ORACLE_HOME "/opt/oracle/app/oracle/product/11.2.0/dbhome_1" does not
reside on a shared file system. Please be sure that the ORACLE_HOME
directory and associated files are identical on all servers. Refer to the
LifeKeeper Oracle Recovery Kit documentation for more information.
Creating dependent file system resource "/u00" on <server1>.
Creating dependency between Oracle database "SID (SID)" and the dependent
resource "/u00" on <server1>.
Creating dependency between Oracle database "SID (SID)" and the listener
resource "LSNR.LISTENER" on <server1>.
Performing in-service of new Oracle resource tag=< SID > on <server1>.
END successful create of on server <server1>
```

You will also find a similar warning in the LifeKeeper log. If this warning is not heeded and you continue on by extending your hierarchy and then try to bring the database resource in service on another node, you will get a message similar to the following in the LifeKeeper GUI dialog:

```
Put resource "OST" in-service
BEGIN restore of "OST" on server "cae-qa-v39"
Begin the "start [ start.normal ]" of the database "OST" on "cae-qa-v39".
The "start [ start.normal ]" attempt of the database "OST" appears to
have failed on "cae-qa-v39".
ORA-01078: failure in processing system parameters
LRM-00109: could not open parameter file '/opt/oracle/app/oracle/product/
11.2.0/dbhome_1/dbs/initOST.ora'
Begin the "start [ start.force ]" of the database "OST" on "cae-qa-v39".
The "start [ start.force ]" attempt of the database "OST" appears to have
failed on "cae-qa-v39".
```

```
select 'alter database datafile '||file#||' end backup;' from v\$backup
where status = 'ACTIVE'
```

It is also possible to get a message similar to the following in the dialog box:

```
Put resource "OST" in-service
BEGIN restore of "OST" on server "cae-qa-v39"
Begin the "start [ start.normal ]" of the database "OST" on "cae-qa-v39".
Initial inspection of "start.normal" failed, verifying failure or success
of received output.
Logon failed with "" for "OST" on "cae-qa-v39". Please check username/
password and privileges.
The "start [ start.normal ]" attempt of the database "OST" appears to
have failed on "cae-qa-v39".
ERROR:

ORA-01031: insufficient privileges

Enter password:

ERROR:

ORA-01005: null password given; logon denied
```

To solve this issue, copy ***$ORACLE_BASE/admin*** from the primary system where the database instance was created to the backup system (where the hierarchy was extended to) ***$ORACLE_BASE/admin***. Also change ownership of this directory to your Oracle username and Oracle group (typically **oracle:oinstall**).

Also copy all ***{$ORACLE_SID}*** (**OST** in this example) files from the primary system in **$ORACLE_BASE/product/11.2.0/dbhome_1/dbs** to **ORACLE_BASE/product/11.2.0/dbhome_1/dbs** on the backup system.

For example, these were the files that were copied from a primary system to the backup, and the ORACLE SID was OST.

```
-rw-r—— 1 oracle oinstall 1544 2012-05-09 11:02 hc_OST.dat
-rw-r—— 1 oracle oinstall 24 2012-01-31 10:22 lkOST
-rw-r—— 1 oracle oinstall 1536 2012-03-05 09:02 orapwOST
-rw-r—— 1 oracle oinstall 2560 2012-05-09 10:58
spfileOST.ora
```

In another example, where the SID was ORA01, the following files were copied:

```
-rw-r—— 1 oracle oinstall 1536 2010-09-08 18:25 orapwORA01
```

```
-rw-r---- 1 oracle oinstall 24 2010-09-08 18:25 lkORA01
-rw-r---- 1 oracle oinstall 2560 2010-09-08 18:30 spfileORA01.ora
-rw-r---- 1 oracle oinstall 1544 2010-09-08 18:30
hc_ORA01.dat
```

and a directory

```
peshm_ORA01_0/:
```

Oracle Listener Stays in Service on Primary Server After Failover

Network failures may result in the listener process remaining active on the primary server after an application failover to the backup server.

6.11.4.1.1. Oracle Database Creation Problems

Problem:	During DataBase creation using dbca, the following message is received: "`ORA-00439 feature not enabled: string`"
Action:	Check the value of the environment variable `$ORACLE_SID`. Make sure that is the same as the SID that is being created.
Problem:	During Database creation from scripts, the following message is received: "**`ORA-01092 ORACLE instance terminated. Disconnection forced`**"
Action:	See the alert in the bdump directory. If you see the message "`ORA-12714 invalid national character set specified`", then check the value of the environment variable`$ORA_NLS33`. Make sure that it is set to the correct location.
Problem:	If you encounter problems creating the database from the script generated from dbca, then do the following:
Action:	1. Be sure to create the following directories if they do not already exist: `bdump` `cdump` `udump` `<oracle data base directory>/oradata sid` `<oracle data base directory>/dbs` `<oracle data base directory>/admin/<SID>` If you need to determine the path to your *bdump* and *udump* directories, you can look in the initialization file (*init<SID>.ora*). 2. Make sure the file *$ORACLE_HOME/dbs/orapw* exists; if not, use the `orapwd` utility to create it.

6.11.4.1.2. Oracle Database Startup Problems

Problem:	During DataBase start-up using sqlplus, the following message is received: "`ORA-03113 end-of-file on communication channel`"
Action:	Make sure the initialization file (*`init<SID>.ora`*) and the password file (*`orapw<SID>`*) are in the directory *`$ORACLE_HOME/dbs`*.
Problem:	During Database startup, the following message is received: "`ORA-01092 ORACLE instance terminated. Disconnection forced`"
Action:	See the alert in the *bdump* directory. If you see the message "`ORA-12701 CREATE DATABASE character set is not known`", then check the value of the environment variable `$ORA_NLS33`. Make sure that it is set to the correct location.

6.11.4.1.3. inqfail error in the LifeKeeper Log

If an inqfail error appears in your LifeKeeper error log following a failover, you will need to change the `filesystemio` setting.

Note: The disk id and server name will be different for each configuration.

To resolve this problem, you will need to change the setting `filesystemio="SETALL" to filesystemio="ASYNCH"`

To locate this setting, query the option with the following SQL command:

1. `SQL> show parameter filesystemio;`
2. Use the following commands to change the settings:

```
SQL> alter system set filesystemio_options=<XXXXXXX> scope=spfile;

     <XXXXXXX> can be set to

     <XXXXXXX> = {none | setall | direction | asynch}

     NONE - no optimization

     ASYNC – enable asynchronous I/O

     DIRECTIO - enable direct I/O

     SETALL - enables all available features
```

IMPORTANT: Oracle needs to be restarted after resetting the parameter.

6.11.5. Oracle Appendix

6.11.5.1. Setting up Oracle to Use Raw I/O

Use the following steps to create an Oracle database that uses shared Raw I/O devices instead of files.

1. Determine the minimum number and sizes of files that you will need to create your database, including control files, tablespaces and redologs. You can create a mixed setup with some of those items as files and others on Raw I/O devices. All of the Raw I/O devices must use shared disk partitions.

2. Create a Raw I/O setup with the necessary number of Raw I/O devices.

   a. Create the raw devices with the same size or larger than you are going to specify for the Oracle database creation.

   b. Create raw device mappings in the system initialization file(i.e. `boot.local` or `rc.local`) using the `raw` command. You should add meaningful comments to identify which raw device represents which Oracle file. This is done so that the mapping can be re-established in the case of a re-boot of the system. These mappings should be removed from the file manually once the Raw I/O device is under LifeKeeper protection.

3. Make the raw devices writable for the Oracle database using the following command:
   `chown oracle:dba /dev/raw[0-9]*`

   where the owner and group are specific to your Oracle instance's configuration.

4. Activate the raw device settings by executing the file that contains the mappings.

5. If you already have a database creation script, go directly to Step 6. If not, you may use one of the Oracle Java GUI tools, `dbassist` or `dbca`, to generate your database creation scripts. Using either tool, you must choose to "**Save As Script**". Do not choose to create the database.

   Notes:

   - In dbca, the "**New Database**" template must be selected to generate scripts. Change filenames to shared devices and adjust the values for your configuration if necessary.

   - The DB creation process should not be started at this point! The `dbassist` tool checks to see if the file specified for each tablespace already exists and will not proceed if it does. The dbca tool prompts to confirm that it will overwrite the files but fails on raw devices. In either case, you are unable to use raw devices directly from these tools.

6. The database creation scripts (either the existing ones or those created by `dbassist` or `dbca`) must be edited. The desired filename (including the path) must be replaced with the full path name of the Raw I/O device. The affected files should include (at minimum) the file's database creation file (for the `CREATE DATABASE` command) and tablespace creation file (for the `CREATE`

TABLESPACE command). Depending on what options you selected in dbassist or dbca, there may more files to be edited. Also, edit the initialization file to change the control files to Raw I/O devices, if desired. The initialization file is located in the directory with the creation script. The result looks like this for the data file:

```
. . .

CREATE DATABASE "LK"

     maxdatafiles 254

     maxinstances 8

     maxlogfiles 32

     character set US7ASCII

     national character set US7ASCII

DATAFILE '/dev/raw/raw1' SIZE 260M AUTOEXTEND ON NEXT 10240K

logfile '/ora/LK/redo01.log' SIZE 500K,

       '/ora/LK/redo02.log' SIZE 500K,

       '/ora/LK/redo03.log' SIZE 500K;

. . .
```

The Raw I/O device must be the minimum size required by Oracle for the data that will be stored.

7. Now create the database by running the script that you created in step #5.
8. Be sure to check the create log for any database or tablespace errors that may have occurred.
9. If you have trouble creating the database with the creation scripts, or you want to add tablespaces on raw devices later, you must create the database with the applicable tool (i.e. dbassist or dbca). Then, add the Raw I/O device data files by executing a command similar to the following from the sql utility:

   ```
   tablespace RAWTS
   DATAFILE '/dev/raw/raw217' SIZE 50M REUSE
   DEFAULT STORAGE (INITIAL 50K NEXT 50K
   MINEXTENTS 1 MAXEXTENTS 4) ONLINE
   ```

10. Add udev rules to make the raw device permissions and ownership persistent across reboots,

switchovers and failovers.

In the Linux 2.6 kernel, the udev system is the default method through which the kernel controls the creation of the special files such as raw devices. When used by Oracle, raw devices require specific ownership and permission settings. These specific settings conflict with the kernel default settings. Addressing the specific settings requires the use of udev rules to set the ownership of the raw device to the Oracle user and Oracle group used with the LifeKeeper protected Oracle SID.

Note: In some OS distributions, rules for creating devices and rules for setting device permissions must be separate. Check your OS distribution udev documentation for more information.

The following are example udev rules that may work for your OS distribution:

```
KERNEL==“raw10”, RUN+=”/bin/chown oracle:oinstall /dev/raw/raw10”

KERNEL==“raw[3-5]*”, OWNER=“oracle”, GROUP=“oinstall”, MODE=“660”
```

The udev rules created must be applied to all nodes in the LifeKeeper cluster prior to bringing the resource hierarchy into service.

6.11.5.1.1. Adding a Tablespace After Creating Hierarchy

If a tablespace is added on a Raw I/O device after the Oracle hierarchy has been created in LifeKeeper, you must create a LifeKeeper Raw I/O hierarchy via the GUI and manually create a dependency between the Oracle resource (as parent) and the Raw I/O resource (as child).

6.11.5.2. Creating an Oracle Listener for Multiple Resources

You may want to create an Oracle Listener if any of the following statements are true for your system configuration:

- Multiple Listeners are defined for Multiple Oracle SIDs
- The Oracle Listener is a critical component in your configuration
- A Single Listener is defined for Multiple Oracle SIDs

> **NOTE:** If multiple listeners are intended to be protected with SPS, the collection of `LISTENER/SID_LIST_LISTENER` stanzas must all be unique. (The 11g installation's *`listener.ora`* file would contain the stanza `LISTENER_11G_1/SID_LIST_LISTENER_11G_1`.)

This process will allow protection of listener(s) within LifeKeeper to accommodate various listener(s) and SIDs combinations.

If you are creating a Listener for multiple resources, follow these procedures.

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy**.

   IMPORTANT: The Oracle Application must be running when you create the resource

2. A dialog box will appear with a drop down list box with all recognized Recovery Kits installed within the cluster. Select **Oracle Database Listener** from the drop down listing. Click **Next** to proceed to the next dialog box.

3. You will be prompted to enter the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information. If you click **Cancel** at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

Field	Tips
Switchback Type	Choose either intelligent or automatic. This dictates how the Listener resource will be switched back to this server when the server comes back up after a failover. If using data replication, choose intelligent as the switchback type.

	Note: The switchback type must match that of the dependent resources (IP and volume resources) used by the Listener resource, or else the create will fail.
Server	Select the server on which you want to create the hierarchy.
Listener Configuration File Path	Select the full path to the Oracle listener configuration file.
Listener Names(s)	Select the name(s) of the Oracle Listener(s) to provide protection for with this resource instance.
Listener Executable(s)	Select the path to the Oracle listener executable. The listener executable is required to start, stop, monitor and recover the specified Oracle listener(s).
Listener Protection Level	Select one of the following levels: Full Control (Start, Stop, Monitor and Recover) Intermediate Control (Start, Monitor and Recover) Minimal Control (Start and Monitor Only)
Listener Recovery Level	Select the level of recovery for the specified listener(s): Standard (On) – Enable standard LifeKeeper recovery. If Standard (On) is selected, all listener failures will be tried locally, and if necessary, trigger a failover to an available backup server. Optional (off) – Enable optional LifeKeeper recovery. If Optional (Off) is selected, all listener failures will be tried locally, but will not cause a failover to an available backup server. **Note**: Local recovery is performed for both recovery levels when a listener error occurs; however, execution of failover depends on the recovery level.
IP Address Name(s)	Select the IP Address resource name that will be protected as dependents of this resource hierarchy. IP Address associated with the selected listener(s) are displayed in the choice list. Select None if no IP resources are required for this configuration,
Listener Tag	Enter a unique name for the resource on the server. The valid characters allowed for the name are letters, digits, and the following special characters: – _ . /

4. Select the **Create** button to start the hierarchy creation. An information box appears and LifeKeeper will validate that you have provided valid data to create your database listener resource hierarchy. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created.

5. The ***Pre-Extend Wizard*** dialog will appear stating that you have successfully created the resource hierarchy and you will be prompted to select the following information. If you are unfamiliar with the **Extend** operation, click **Next** after making a selection in each dialog box. If you are familiar with the **LifeKeeper Extend Resource Hierarchy** defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults**.

Field	Tips
Target Server	Select a Target Server to which the hierarchy will be extended. If you select Cancel before extending the resource hierarchy to at least one other server, LifeKeeper will provide no protection for the applications in the hierarchy.
Switchback Type	This dictates how the Oracle Listener instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either intelligent or automatic. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. Note: Remember that the switchback strategy must match that of the dependent resources to be used by the Oracle Listener resource.
Template Priority	This field appears only if you did NOT extend directly from the Create function.) Enter a number between 1 and 999 to specify the template server's priority in the cascading failover sequence for this resource. A lower number means a higher priority. LifeKeeper assigns the number "1" to the server on which the hierarchy was created. No two servers can have the same priority for a given resource.
Target Priority	Enter a number between 1 and 999 to specify the target server's priority in the cascading failover sequence for this resource. A lower number means a higher priority. LifeKeeper offers a default of 10 for the first server to which a hierarchy is extended.

6. After receiving the message that the pre-extend checks were successful, click **Next** and enter the following information.

Field	Tips
Listener Configuration File Path	Select the full path to the Oracle Listener configuration file.
Listener Executable(s) Path	Select the path to the Oracle Listener executables. The listener executables are required to start, stop, monitor and recover the specified Oracle listener(s).
Listener Tag	This field is automatically populated with a unique name for the new Oracle Listener resource instance on the primary server. The default naming pattern will be displayed for you. You may type in another unique name. The valid characters allowed for the Listener tag are letters, digits and the following special characters – : . /

7. Click **Extend**. The **Hierarchy Integrity Verification** window displays with the following message, **Hierarchy Verification Finished**. Click **Next Server** or **Finish**.

6.11.5.2.1. Updating the Oracle Listener Protection Level

1. Select a **resource** and then the  button from the **Resource** toolbar to update the protection level of the resource.

2. Enter the following information.

Field	Tips
Listener Protection Level	Select one of the following: Full Control (Start, Stop, Monitor and Recover) Intermediate Control (Start, Monitor and Recover) Minimal Control (Start and Monitor Only)

3. Click **Update** to change the **Protection Level** from the current state to the new state. Select **Cancel** to leave the value unchanged.

6.11.5.2.2. Updating the Oracle Listener Recovery Level

1. Select a **listener** and then the  button from the **Resource** toolbar to update the recovery level of the resource.

2. Enter the following information.

Field	Tips
Listener Recovery Level	Select the level of recovery for the specified listener(s). Standard (On_ – enables a standard LifeKeeper recovery. If Standard, (On) is selected, all listener failures will be tried locally and if necessary trigger a fail over to an available backup server. Optional (Off) – enables optional LifeKeeper recovery. If Optional (Off) is selected, all listener failures will be tried locally, but will not cause a fail over to an available backup server. **Note**: Local recovery is performed for both recovery levels when a listener error occurs; however, execution of failover depends on the recovery level.
Update Confirmation	Select the Update button to change the Recovery Level from the current state to the new state. Select Cancel to leave the current value unchanged.

3. Click **Update** to change the Recovery Level from the current state to the new state. Select **Cancel** to leave the current value unchanged.

6.11.5.2.3. Updating the Oracle Protected Listener(s)

1. Select a **listener** and then the  button from the **Resource** toolbar to update your protected listener(s).

2. Enter the following information.

Field	Tips
Listener Name(s)	Select the name or names of the Oracle Listener(s) to provide protection for with this resource instance.
IP Address Name(s)	Select the IP Address resource name(s) that will be added as a dependent of this resource hierarchy. Select None if no additional IP Resources are required for this configuration.

3. Click **Update** to change the **Protected Listener(s)** and **IP assignment** from the current state to the new state. Select **Cancel** to leave the current value unchanged.

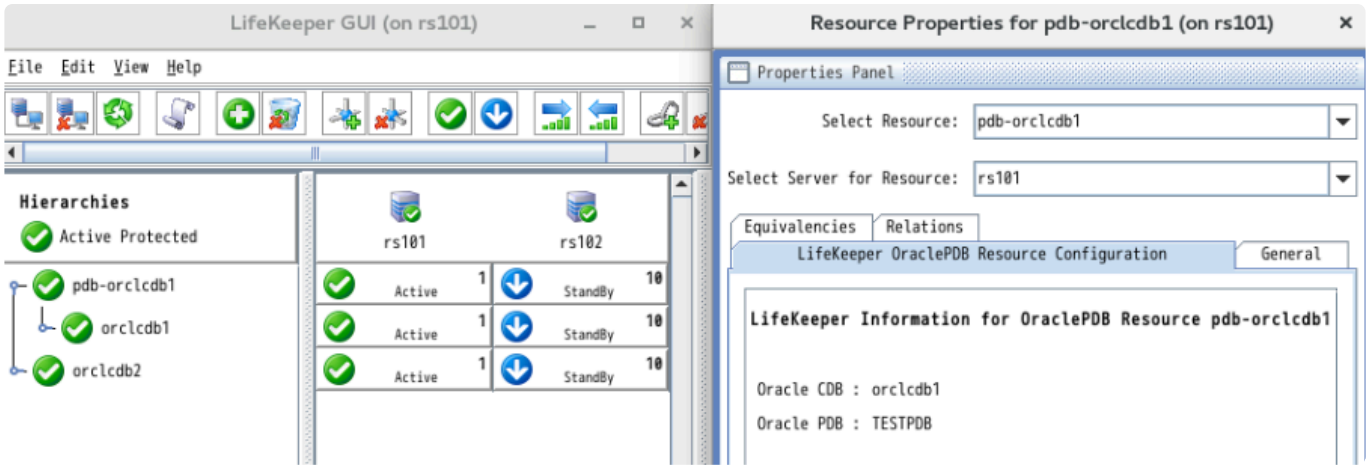
   If you select **Update**, a dialog displays stating that the Protected Listeners for the specific resource is being updated. Click **Finish**.

6.11.5.3. Migrating a Pluggable Database

This section describes the procedure for migrating a pluggable database (PDB) between container databases (CDB) protected by LifeKeeper. We recommend to make a backup of the database in advance.

How to Migrate

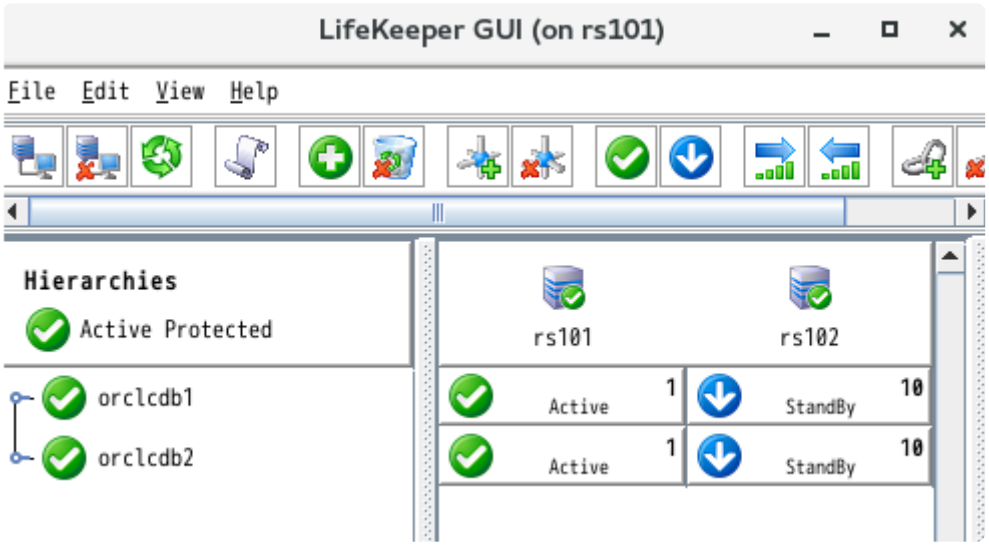
This section describes how to migrate a PDB using the PDB plug/unplug method.The figure below shows an example of a configuration for migrating a PDB (TESTPDB) from the source CDB (ORCLCDB1) to the target CDB (ORCLCDB2).



1. Before migration, connect to the source CDB and check the information of the PDB to migrate.

```
SQL> COLUMN NAME FORMAT A8

     SELECT NAME, DBID, GUID FROM V$CONTAINERS WHERE NAME='<PDB>';
```

2. Bring the Oracle PDB resource out of service and delete the resource. If you have an Oracle PDB resource that protects more than one PDB, remove the PDB from protection from the resource setting Change Protection PDB (see Changing the PDB to Protect).



3. Connect to the source CDB and unplug the PDB. (If the PDB is not stopped, stop the PDB.)

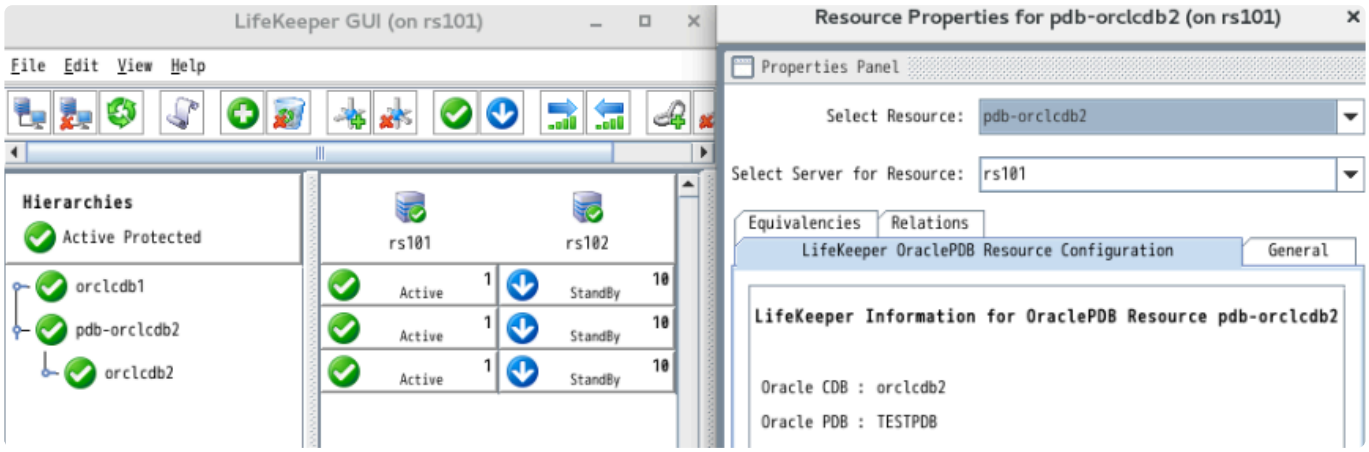
```
SQL> ALTER PLUGGABLE DATABASE <PDB> UNPLUG INTO '/home/
oracle/<PDB>.xml';

SQL> DROP PLUGGABLE DATABASE <PDB>;
```

4. Connect to the target CDB, plug in the PDB and start it.

```
SQL> CREATE PLUGGABLE DATABASE <PDB> USING '/home/oracle/<PDB>.xml'
<COPY|NOCOPY>;

SQL> ALTER PLUGGABLE DATABASE <PDB> OPEN;
```

5. Verify that the PDB information matches before and after the migration.

```
SQL> COLUMN NAME FORMAT A8

    SELECT NAME, DBID, GUID FROM V$CONTAINERS WHERE NAME='<PDB>';
```



6.12. PostgreSQL Recovery Kit Administration Guide

The SIOS Protection Suite for Linux PostgreSQL Recovery Kit is an SQL compliant, object-relational database management system (ORDBMS) based on POSTGRES. Since its inception, PostgreSQL has become one of the most advanced open source relational database management systems.

The SPS for Linux PostgreSQL Recovery Kit provides a mechanism for protecting PostgreSQL instances within LifeKeeper. The PostgreSQL software, LifeKeeper Core and PostgreSQL Recovery Kit are installed on two or more servers in a cluster. Once the PostgreSQL database instance is under LifeKeeper protection, clients connect to the database using a LifeKeeper protected IP address. The LifeKeeper protected IP address must be created separately and a dependency made manually between the parent PostgreSQL resource instance and the child IP address resource. In the event that the PostgreSQL server fails, LifeKeeper will first attempt to recover it on the local server. If the local recovery fails, then LifeKeeper will fail over to a backup server.

SIOS Protection Suite Documentation

The following SIOS Protection Suite product documentation is available from the SIOS Technology Corp. website:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- Optional Recovery Kit Documentation

PostgreSQL Documentation

You can find the PostgreSQL documentation, including the *Administration Guide, User Guide and Reference Guide* at the following location on the web:

http://www.postgresql.org/docs

6.12.1. PostgreSQL Resource Hierarchy

The following example shows a typical PostgreSQL resource hierarchy:

The dependencies in the above example correspond to the following protected resources:

Resource	PostgreSQL Software Component
LKIP.EXAMPLE.COM	Protects the switchable IP address used for client connections
var/lib/pgsql/data	Protects the database data directory (PGDATA)
var/lib/pgsql/exec	Protects the PostgreSQL server and client executables (when executables are installed on a shared file system)
var/lib/pgsql/log	Protects the database log file directory (when the log path is located on a shared file system)
var/lib/pgsql/pg_xlog	Protects the database transaction log directory (PGDATA/pg_xlog) The transaction log directory is also referred to as Write-Ahead-Log directory.
var/lib/pgsql/ socket_path	Protects the database socket directory (when the socket path is located on a shared file system).

In the event of failover, LifeKeeper will bring the file system, IP address and database resources (including all the resource dependencies) in service on a backup server. Clients will be disconnected and will need to re-connect to the server. Any SQL statement that has not been committed will need to be re-entered.

6.12.2. PostgreSQL Hardware and Software Requirements

Your LifeKeeper configuration must meet the following requirements prior to the installation of LifeKeeper for Linux PostgreSQL Recovery Kit. Please refer to the SPS for Linux Installation Guide for specific instructions regarding the installation and configuration of your LifeKeeper hardware and software.

Hardware Requirements

- **Servers** – Servers should be configured in accordance with the requirements described in the SPS for Linux Technical Documentation and the SPS for Linux Release Notes.

- **IP Network Interface Cards** – Each server requires at least one Ethernet TCP/IP-supported network interface card. Remember, however, that best practice is for a LifeKeeper cluster to have at least two communication paths. Two separate LAN-based communication paths using dual independent sub-nets are recommended for heartbeats, and at least one of these should be configured as a private network. Using a combination of TCP and TTY heartbeats is also supported.

Software Requirements

- **TCP/IP Software** – Each server in your LifeKeeper configuration requires TCP/IP Software.

- **PostgreSQL Software** – The same version of the PostgreSQL software must be installed on all servers in the cluster. The PostgreSQL software can be downloaded from one of the mirrors available at http://www.postgresql.org/download.

- **LifeKeeper software** – It is imperative that you install the same version of the LifeKeeper software and apply the same versions of the LifeKeeper software patches to each server in your cluster.

- **LifeKeeper for Linux PostgreSQL Recovery Kit** – The PostgreSQL Recovery Kit is provided on the SPS for Linux Installation Image File (`sps.img`) via ftp download. It is packaged, installed and removed via Red Hat Package Manager, rpm:

    steeleye-lkPGSQL

6.12.3. PostgreSQL Configuration Considerations

This section contains information that you should consider before you start to configure and administer the PostgreSQL Recovery Kit.

Using Mirrored File Systems with DataKeeper

Protecting PostgreSQL: Best Practices

6.12.3.1. Protecting PostgreSQL Best Practices

In an Active/Standby configuration, the backup server is not actively running the PostgreSQL, but stands by in case the primary server experiences a failure. In an Active/Active configuration, each server is actively running a PostgreSQL instance, while acting as a backup for the other server in case of failure. The following list provides requirements that should be adhered to when protecting a PostgreSQL resource instance in an active/standby or active/active configuration.

1. The PostgreSQL DataDir and Write-Ahead-LogPath (*PGDATA/pg_xlog*) must be installed on one or more shared file systems. The paths *DataDir* and *WAL-Path* must be shared between all servers that will protect the resource instance.

- ◦ The PostgreSQL Operating System User must own the data directory and directory containing the Write-Ahead-LogPath.

- ◦ The PostgreSQL database must have been created using the utility `initdb`. The `initdb` utility must be run as the PostgreSQL owner using the `-D <datadir>` option.

- ◦ The automatic startup of the default PostgreSQL instance must either be disabled or the default PostgreSQL instance must be restricted to running on a port other than those intended for use with LifeKeeper.

- ◦ The automatic startup of the PostgreSQL instance to be protected by LifeKeeper must be disabled. LifeKeeper will control the starting and stopping of the protected instance.

- ◦ The PostgreSQL instance must be started manually prior to hierarchy creation. It is required that the instance be started with the backend option `-o "-p <port>"` specified to the `pg_ctl` utility.

2. The `StartupLogPath`, `SocketPath` and the `ExecutablePath` can be installed to optional shared file systems on the primary server or each local node file system.

- ◦ The PostgreSQL Operating System User must own the directory containing the socket path.

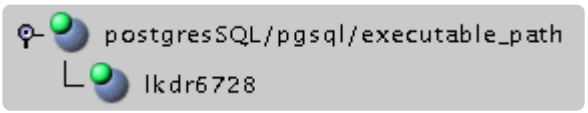
- ◦ The PostgreSQL Operating System User must have write permissions on the directory containing the StartupLogPath.

3. It is recommended that each instance use a unique port and socket path when running multiple instances in either an Active/Standby or Active/Active scenario.

6.12.3.2. Using Mirrored File Systems with DataKeeper

The PostgreSQL Recovery Kit supports the use of SIOS DataKeeper as a shared file system. The mirrored file systems can be used for the PostgreSQL installation path, log path, the data directory and the executable path.

For example, a dependent file system for a PostgreSQL resource would look similar to the following, which shows a file system for the data directory and its dependency, the DataKeeper resource mirror.



6.12.4. PostgreSQL Installation

Installing/Configuring PostgreSQL with LifeKeeper

The following sequence is recommended for installing and configuring the PostgreSQL database and LifeKeeper software. Each of these steps links to detailed tasks.

1. Install the PostgreSQL software.
2. Create the PostgreSQL database.
3. Install the LifeKeeper Core and PostgreSQL Recovery Kit.
4. Configure LifeKeeper Tunable Settings for PostgreSQL Resources.

After you have performed these tasks, you will be ready to create the LifeKeeper resource hierarchy to protect your PostgreSQL database.

Resource Configuration Tasks

Once you have completed the setup tasks described in the previous section, you are ready to create and extend your PostgreSQL resource hierarchies.

The following tasks are available for configuring the LifeKeeper for Linux PostgreSQL Recovery Kit:

- Create Resource Hierarchy – Creates a PostgreSQL resource hierarchy.
- Delete Resource Hierarchy – Deletes a PostgreSQL resource hierarchy.
- Extend Resource Hierarchy – Extends a PostgreSQL resource hierarchy from the primary server to the backup server.
- Unextend Resource Hierarchy – Unextends (removes) a PostgreSQL resource hierarchy from a single server in the LifeKeeper cluster.
- Viewing PostgreSQL Configuration Settings – Allows viewing of the Resource Properties dialog.

Refer to the GUI Administrative Tasks section of the SPS for Linux Technical Documentation for instructions on configuring LifeKeeper Core resource hierarchies, for instance, file system and IP resources.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all

applicable servers in the cluster.

- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View Properties. View the properties of a resource hierarchy on a specific server.
- Edit Properties. Edit the properties of a resource hierarchy on a specific server.

Note: The configuration tasks throughout this section are performed using the Edit menu. You may also perform most of the tasks:

- from the toolbar.
- by right-clicking on a global resource in the left pane of the status display.
- by right-clicking on a resource in the right pane of the status display.

Using the right-click method allows you to avoid entering information that is required using the **Edit** menu.

Upgrading

Upgrading From Previous Version of the PostgreSQL Recovery Kit

6.12.4.1. Install the PostgreSQL Software

Install the PostgreSQL software on all servers in the cluster using identical parameters/settings. Refer to the PostgreSQL Administration Guide for details. The following are additional recommendations and reminders to ensure that LifeKeeper will work with PostgreSQL:

- The PostgreSQL client software packages must be installed. These packages must include the PostgreSQL `psql` client utility.
- The PostgreSQL server software packages must be installed. These packages must include the PostgreSQL `pg_ctl` and `initdb` utilities.
- The PostgreSQL client and server packages must be the same version on all servers.
- A PostgreSQL Operating System User must exist on all servers as follows:
    - º This PostgreSQL Operating System User should be designated as the owner of the PostgreSQL software installation and subdirectories.
    - º This PostgreSQL Operating System User must have authority to use the

`pg_ctl` utility. The PostgreSQL Operating System User must be able to start and stop the postmaster instance using the `pg_ctl` commands.
    - º The PostgreSQL Operating System User name should contain alphanumeric characters only.
    - º The user id and group id of this PostgreSQL Operating System User must be identical on all servers.
- A PostgreSQL Database Administrator User must exist within the PostgreSQL database for LifeKeeper client connections through the `psql` utility.
    - º This PostgreSQL Database Administrator User must have the ability to connect to the database (

template1), as well as obtain the listing of defined databases for the instance.
    - º This PostgreSQL Database Administrator User must have the ability to view system tables and make generalized queries.
    - º The PostgreSQL Database Administrator User is different from the PostgreSQL Operating System User, although they can have the same name.
    - º Example: PostgreSQL Operating System User=postgres, and PostgreSQL Database Administrator User=lkpostgres; or PostgreSQL Operating System User=postgres, and PostgreSQL Database Administrator User=postgres.

- Auto Startup at the time of system activation must be disabled because PostgreSQL server daemon is controlled by LifeKeeper

6.12.4.2. Creating a PostgreSQL Database

Follow the instructions in your PostgreSQL Administration Guide to create your database. In addition, please note the following recommendations:

- The PostgreSQL data directory should be initialized using the `initdb` utility, specifying the `-D <data dir>` option. The `initdb` command must be run as the PostgreSQL Operating System User.
- The PostgreSQL instance data directory must reside on a shared file system.
- The PostgreSQL transaction log directory must reside on a shared file system.
- The PostgreSQL database name should contain alphanumeric characters only.
- After creating your database, you should disable automatic startup of the PostgreSQL database instance. Once under LifeKeeper protection, LifeKeeper will handle the start and stop of the database.
- The PostgreSQL instance must be started manually prior to hierarchy creation. It is required that the instance be started with the backend option `-o "-p <port>"` specified to the `pg_ctl` utility.

No Password Protection (Instance is not Password Protected)

- If the PostgreSQL database instance will not be password protected or will not require a password for local client connections from the PostgreSQL Database Administrator User, then an entry must exist allowing local trust connections. The following is an example of a `pg_hba.conf` entry to enable local client connects for the PostgreSQL Database Administrator User:

```
====================
.
.
Local all postgres trust
.
.
======================
```

Enabling Password Protected (Instance requires a Password for Connections)

- Password Protected database instances require a password entry for the PostgreSQL Database Administrator User to exist in the `.pgpass` credentials file on each server in the cluster where the resource will be protected. The `.pgpass` file must contain a valid and tested entry for each PostgreSQL Database Administrator User requiring a password.
- The `.pgpass` file must be located in the home directory of the PostgreSQL Operating System User. Please set the appropriate file permissions to restrict access to the file.

- The following is an example of a valid *.pgpass* file with the format

  `<hostname>:<port>:<database>:<user>:<password>`

```
==================
*:5443:*:lifekeeper:jh43tmp2009
====================
```

Note: The *.pgpass* file is required for the utility `psql` for unattended (non-terminal or scripted) connections. The *.pgpass* file must exist on each server where the password protected instance will be protected.

6.12.4.3. Install the LifeKeeper Software

Once you have installed the PostgreSQL software and created your database, you are ready to install the LifeKeeper Core software and any required patches followed by the PostgreSQL Recovery Kit.

Refer to the SPS for Linux Installation Guide for details on installing the LifeKeeper packages.

6.12.4.4. LifeKeeper Tunable Settings for PostgreSQL

The PostgreSQL Recovery Kit provides tunable environment variables to help customize resource protection in certain scenarios. To change the values of these variables, edit the file */etc/default/LifeKeeper*. No processes need to be restarted for the new settings to take effect. The default values will work for most environments where the PostgreSQL Recovery Kit will be installed.

LKPGSQL_CONN_RETRIES

This tunable controls the amount of time the PostgreSQL Recovery Kit will wait for the database to start. The amount of time is calculated by the Recovery Kit using the following formula: (LKPGSQL_CONN_RETRIES* 5) = total time in seconds to wait for a database instance to start. The setting of this variable affects both the resource in-service requests and the resource local recovery.

LKPGSQL_DISCONNECT_CLIENT

This tunable controls whether active clients will be disconnected in the event of a postmaster crash. When the value is set to 1 (true), active clients will be disconnected while resource local recovery is in progress. When the value is set to 0 (false), active clients will not be disconnected while resource local recovery is in progress. This variable affects only the resource local recovery events and is only applicable during local recovery events where the postmaster process is not running.

LKPGSQL_SDIRS

This tunable controls the client disconnect behavior when the PostgreSQL database is shut down. This comma separated tunable must be added to the defaults file. By setting this option, the specified resource instance or instances corresponding to the protected data directory will not force clients to disconnect during shutdown.

```
LKPGSQL_SDIRS=/protected/pgsql-datadir
```

```
LKPGSQL_SDIRS=/protected/pgsql-datadir,/otherprotected/pgsql-datadir
```

Where */protected/pgsql-datadir* and */otherprotected/pgsql-datadir* are the PostgreSQL data directories under LifeKeeper protection.

Note: The options LKPGSQL_SDIRS and LKPGSQL_IDIRS are exclusive. The value placed in the LKPGSQL_SDIRS or LKPGSQL_IDIRS tunable must match exactly with the protected datadir value selected during hierarchy creation.

LKPGSQL_IDIRS

This tunable controls the client disconnect behavior when the PostgreSQL database is shut down. This comma separated tunable must be added to the defaults file. By setting this option, the specified resource instance or instances corresponding to the protected data directory will force clients to do an immediate disconnect during shutdown.

```
LKPGSQL_IDIRS=/protected/pgsql-datadir
```

```
LKPGSQL_IDIRS=/protected/pgsql-datadir,/otherprotected/pgsql-datadir
```

Where */protected/pgsql-datadir* and */otherprotected/pgsql-datadir* are the PostgreSQL data directories under LifeKeeper protection.

Note: The options LKPGSQL_SDIRS and LKPGSQL_IDIRS are exclusive. The value placed in the LKPGSQL_SDIRS or LKPGSQL_IDIRS tunable must match exactly with the protected datadir value selected during hierarchy creation.

6.12.4.5. Creating a PostgreSQL Resource Hierarchy

Perform the following steps on the primary server:

1.
2. On the **Edit** menu, select **Server**, then **Create Resource Hierarchy**.
   The **Create Resource Wizard** dialog will appear.

3. Select **PostgreSQL Database** from the drop-down list and click **Next**.

4. You will be prompted for the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter any error requiring you to correct the previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

Fields	Tips
Switchback Type	Choose either **intelligent** or **automatic**. This determines how the PostgreSQL resource will be switched back to the primary server after it comes in-service (active) on the backup server following a failover. Intelligent switchback requires administrative intervention to switch the resource back to the primary server, while automatic switchback occurs as soon as the primary server is back on line and re-establishes LifeKeeper communication paths. **Note:** The switchback strategy must match that of the dependent resources to be used by the PostgreSQL resource.
PostgreSQL Executable Location	This field is used to specify the directory path containing the PostgreSQL executables. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . /
PostgreSQL Client Executable Location	This field is used to specify the directory path containing the PostgreSQL executable `psql`. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . /
PostgreSQL Administration Executable Location	This field is used to specify the directory path containing the PostgreSQL executable `pg_ctl`. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . /
PostgreSQL Data Directory	This field is used to specify the location of the PostgreSQL data directory (*datadir*) that will be placed under LifeKeeper protection. The specified directory must exist and reside on a shared file system. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . /

PostgreSQL Port	This field is used to specify the TCP/IP port number on which the postmaster daemon is listening for connections from client applications.
PostgreSQL Socket Path	This field is used to specify the full path to the Unix-domain socket on which the postmaster daemon is listening for connections from client applications. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . /
PostgreSQL Database Administrator User	This field is used to specify a PostgreSQL Database Administrator User name for the specified database instance with connection and administrator privileges for the instance.
PostgreSQL Logfile	This field is used to specify the log file path used by the -l option of pg_ctl to start and stop PostgreSQL.
PostgreSQL Database Tag	This is a unique tag name for the new PostgreSQL database resource on the primary server. The default tag name consists of the word pgsql followed by the port number for the database instance. You may type in another unique tag name. The valid characters allowed for the tag are letters, digits and the following special characters: – _ . /

5. Click **Create**. The **Create Resource Wizard** will then create your PostgreSQL resource hierarchy. LifeKeeper will validate the data entered. If LifeKeeper detects a problem, an error message will appear in the information box.

6. You should see a message indicating that you have successfully created a PostgreSQL resource hierarchy, and you must extend that hierarchy to another server in your cluster to achieve failover protection. Click **Next**.

7. Click **Continue**. LifeKeeper will then launch the **Pre-extend Wizard**. Refer to **Step 2** in the topic Extending a PostgreSQL Resource Hierarchy for details on how to extend your resource hierarchy to another server.

6.12.4.6. Deleting a PostgreSQL Resource Hierarchy

To delete a PostgreSQL resource hierarchy from all servers in your LifeKeeper configuration, complete the following steps:

1.
2. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.
3. Select the name of the **Target Server** where you will be deleting your PostgreSQL resource hierarchy.

   Note: If you selected the **Delete Resource** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.
4. Select the **Hierarchy to Delete**. (This dialog will not appear if you selected the **Delete Resource** task by right-clicking on a resource instance in the left or right pane.) Click **Next**.
5. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Next**.
6. Another information box appears confirming that the PostgreSQL resource was deleted successfully.
7. Click **Done** to exit.

6.12.4.7. Extending a PostgreSQL Resource Hierarchy

This operation can be started from the **Edit** menu or initiated automatically upon completing the **Create Resource Hierarchy** option, in which case you should refer to **Step 2** below.

1.
2. On the **Edit** menu, select **Resource**, then **Extend Resource Hierarchy**. The ***Pre-Extend Wizard*** appears. If you are unfamiliar with the **Extend** operation, click **Next**. If you are familiar with the LifeKeeper Extend Resource Hierarchy defaults and want to bypass the prompts for input/ confirmation, click **Accept Defaults**.

3. The ***Pre-Extend Wizard*** will prompt you to enter the following information.

   Note: The first two fields appear only if you initiated the **Extend** from the **Edit** menu.

Field	Tips
Template Server	Select the server where your PostgreSQL resource is currently in service.
Tag to Extend	Select the PostgreSQL resource you wish to extend.
Target Server	Enter or select the server you are extending to.
Switchback Type	This determines how the PostgreSQL resource will be switched back to the primary server after it comes in service (active) on the backup server following a failover. You can choose either **intelligent** or **automatic**. The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box. **Note:** Remember that the switchback strategy must match that of the dependent resources to be used by the PostgreSQL resource.
• Template Priority*	Select or enter a **Template Priority**. This is the priority for the PostgreSQL hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note:** This selection will appear only for the initial extend of the hierarchy.
Target Priority	This is the priority for the new extended PostgreSQL hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid indicating a server's priority in the cascading failover sequence for the resource. Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be

	consecutive, but no two servers can have the same priority for a given resource.

4. After receiving the message that the pre-extend checks were successful, click **Next**.

5. Depending upon the hierarchy being extended, LifeKeeper will display a series of information boxes showing the Resource Tags to be extended, some of which cannot be edited.

6. The ***Extend Wizard*** will prompt you to enter the following information.

Field	Tips
PostgreSQL Executable Location	This field is used to specify the directory path containing the PostgreSQL executables. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . /
PostgreSQL Database Tag	This is a unique tag name for the new PostgreSQL database resource on the primary server. The default tag name consists of the word pgsql followed by the port number for the database instance. You may type in another unique tag name. The valid characters allowed for the tag are letters, digits and the following special characters: – _ . /

7. After receiving the message "`Hierarchy extend operations completed`", click **Next Server** to extend the hierarchy to another server, or click **Finish** if there are no other extend operations to perform.

8. After receiving the message "`Hierarchy Verification Finished`", click **Done**.

6.12.4.8. Unextending a PostgreSQL Resource Hierarchy

To remove a resource hierarchy from a single server in the LifeKeeper cluster, do the following:

1.
2. On the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.

3. Select the **Target Server** where you want to unextend the PostgreSQL resource. It cannot be the server where the resource is currently in service. (This dialog box will not appear if you selected the Unextend task by right-clicking on a resource instance in the right pane.) Click **Next**.

4. Select the PostgreSQL hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the Unextend task by right-clicking on a resource instance in either pane).

5. An information box appears confirming the target server and the PostgreSQL resource hierarchy you have chosen to unextend. Click **Unextend**.

6. Another information box appears confirming that the PostgreSQL resource was unextended successfully. Click **Done** to exit the **Unextend Resource Hierarchy** menu selection.

6.12.4.9. Viewing PostgreSQL Configuration Settings

The **Resource Properties** dialog is available from the **Edit** menu or from a resource context menu. This dialog displays the properties for a particular resource on a server. When accessed from the **Edit** menu, you can select the resource and the server. When accessed from a **Resource Context** menu, you can select the server.

From the **Configuration** tab, you can view the following PostgreSQL settings:

- Executable Path
- Client Executable Name
- Admin Executable Name
- Bind Setting
- Startup Log Location
- PostgreSQL Operating System User Name
- PostgreSQL Database Administrator User
- Version Number
- Data Directory
- Socket Location
- Port Number
- OS Daemon Name

6.12.4.10. Upgrading PostgreSQL

During an upgrade from a previous version of the SPS for Linux PostgreSQL software, the upgrade will make modifications to the existing SPS PostgreSQL resource instance. When the SPS software is updated on the server, the following stored values will be added to the internal SPS information field automatically.

- **Client Executable Location (*psql*)** – the location of the *psql* or equivalent client utility used for connecting to the protected database instance. After an upgrade, this value can be verified from the LifeKeeper GUI properties display. The value can also be verified from the LifeKeeper command line using the `set_value` utility.

  `set_value` is the name of a LifeKeeper utility provided for the LifeKeeper PostgreSQL Recovery Kit to update the internal resource information field values. The use of this utility should be limited to issues explained in this topic or at the request and instruction of the SIOS Technology Corp. Support team.

  Note: The `set_value` utility does not perform rigorous error checking and therefore is not intended for general use.

- **Administration Executable Location (*pg_ctl*)** – the location of the `pg_ctl` or equivalent administration utility used for starting, stopping and checking the status of the protected database instance. After an upgrade, this value can be verified from the LifeKeeper GUI properties display. The value can also be verified from the LifeKeeper command line using the `set_value` utility.

- **PostgreSQL Database Administrator User** – the PostgreSQL Database Administrator User for the LifeKeeper protected instance. This user must have connection and administrator privileges for the protected database instance. The default value used following an upgrade is the PostgreSQL Operating System User that owns the PostgreSQL data directory. After an upgrade, this value can be verified from the LifeKeeper GUI properties display. The value can also be verified from the LifeKeeper command line using the `set_value` utility.

- **PostgreSQL Daemon Name (postmaster)** – the name of the running backend daemon. This value is determined during the first status check of the database instance. The default value is postmaster. After an upgrade, this value can be verified from the LifeKeeper GUI properties display. The value can also be verified from the LifeKeeper command line using the `set_value` utility.

- **Default Test Database (template1)** – the database used by LifeKeeper during the database instance monitoring to verify basic connectivity. After an upgrade, the default test database will be set to **template1**.

- **PostgreSQL Maximum Monitoring Hangs (LKPGSQL_QCKHANG_MAX)** – the setting that provides protection against an unlimited number of connection hangs before a restorative or reparative failover action is initiated. A portion of PosgreSQL Recovery Kit's monitoring requires a

connection to the protected database. The number of connection hangs allowed is determined during resource creation by the setting LKPGSQL_QCKHANG_MAX. The default value previous to version 8.1.2 was 15. After upgrading to version 8.1.2 (or later), the default value is 2. Since this value is stored with the resource at create time, any resources created prior to upgrading to version 8.1.2 will remain at the default value of 15 unless updated by the user while any resources created after upgrading to 8.1.2 (or later) will contain a default value of 2. The value can also be verified from the SPS command line using the set_value utility.

IMPORTANT NOTE: Following the upgrade of the SPS for Linux PostgreSQL Recovery Kit software, you should test your PostgreSQL resource hierarchy by initiating a manual switchover that will simulate a failover of the resource instance from the primary server to a backup server.

Important Upgrade Considerations

If a resource does not come into service following the upgrade, check the following conditions:

- **Client Executable name is not found or incorrect**

  The value can be updated using the `set_value` utility. The syntax for the Client Executable update is as follows:

  ```
  /opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value <tag>
  'clientexe' <full path to the psql utility>.
  ```

  Example: `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value pgsql-5443 'clientexe' '/pgsql/clientutils/psql'.`

- **Administration Executable name is not found or incorrect**

  The value can be updated using the `set_value` utility. The syntax for the Administration Executable update is as follows:

  ```
  /opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value <tag>
  'osexe' <full path to the pg_ctl utility>.
  ```

  Example: `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value pgsql-5443 'osexe' '/pgsql/adminutils/pg_ctl'.`

Lowering the interval for recovering from multiple hang events (LKPGSQL_QCKHANG_MAX)

- **Maximum Monitoring Hangs value is too large in versions prior to 8.1.2**

  The value for **Maximum Monitoring Hangs** for existing PostgreSQL resource instances can viewed or set using the `set_value` utility.

The syntax for setting the value for the **Maximum Monitoring Hangs** (LKPGSQL_QCKHANG_MAX) is as follows:

```
/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value <tag>
'hangmax' <number>.
```

Example: `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value pgsql-5443 'hangmax' 3.`

Note: Include the -c argument to update the value on all nodes in the cluster (`set_value -c <tag>…`).

The syntax for **viewing** the value is as follows:

```
/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value -l <tag>
'hangmax'
```

Example:`/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value -l pgsql-5443 'hangmax'`

6.12.5. PostgreSQL Administration

Updating Database Administrator User

The Update User option allows the LifeKeeper administrator to change the current PostgreSQL Database Administrator User for the LifeKeeper PostgreSQL resource instance.

Testing Your PostgreSQL Resource Hierarchy

You can test your PostgreSQL resource hierarchy by initiating a manual switchover that will simulate a failover of the resource instance from the primary server to a backup server.

Performing a Manual Switchover from the LifeKeeper GUI

EnterpriseDB Postgres Plus Advanced Server Environments

Protecting EnterpriseDB Postgres Plus Advanced Server Resources

Symfoware Server/Enterprise Postgres Environments

Protecting Symfoware Server/Enterprise Postgres Resources

6.12.5.1. Performing a Manual Switchover from the LifeKeeper GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, **Resource**, and **In Service**. For example, an in-service request executed on a backup server causes the PostgreSQL resource hierarchy to be placed in service on the backup server and taken out-of-service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the Out of Service request, the resource hierarchy is taken out-of-service without bringing it in service on the other server.

Important: After bringing your resource hierarchy in service on the backup server, you should attempt to connect to the databases. With password protected instances, it is of particular importance that the *.pgpass* file is verified on the backup server. To verify the *.pgpass* file is valid, a client connection to the database should be made using both the `psql` utility and the PostgreSQL Database Administrator User. A valid *.pgpass* file exists if the connection succeeds without prompting for an interactive password.

6.12.5.2. Protecting EnterpriseDB Postgres Plus Advanced Server

No additional LifeKeeper configuration settings are needed to protect EnterpriseDB Postgres Plus Advanced Server Resources.

Issue	Solution
During the installation of EnterpriseDB Postgres Plus Advanced Server, if the option **PostgreSQL-compatible defaults and samples** is chosen in the **Configuration Mode** dialog, the '*edb*' database that is used by LifeKeeper is not created.	Manually add the '*edb*' database using the utility `'createdb'`. The command `'createdb -p <port> -h <socket path> edb'` should be executed as the PostgreSQL Operating System User. The following is an example: `su - postgres` `postgres@server1 ~>createdb -p 5435 -h /var/lib/postgres edb`

6.12.5.3. Protecting Symfoware Server/ Enterprise Postgres

The following table explains the supported features when protecting Symfoware Server/Enterprise Postgres.

The support range
• Support the compatible functions with PostgreSQL. • The mirroring functionality is not supported. Use DataKeeper instead. • The native interface of Symfoware Server is not supported. Use Open Interface (Symfoware 12.2), Postgres (Symfoware 12.3 or later). • WebAdmin is asymmetry. For the details, refer to Symfoware Server Cluster Operation Guide for Fujitsu Software. • Following functions are not supported: - WAL duplication - Encryption (encryption of stored data) - Data concealing - Parallel search - In-memory

Notes for the configuration
Set up the environment variable, LD_LIBRARY_PATH required to execute Symfoware commands (pg_ctl,psql, etc.) in the appropriate environment file (.bash_profile, etc.) for the DB Administration User (OS user) log-in.

6.12.5.4. Updating Database Administrator User

This **Update User** option will update the stored value for the PostgreSQL Database Administrator User on all systems where the resource is protected. The **Update User** option can be invoked from either the **LifeKeeper resource toolbar** or the **LifeKeeper resource context menu**.

To update the PostgreSQL Database Administrator User, perform the following steps on the primary server:

Note: The **Update User** menu and toolbar options will be disabled for any out-of-service resources.

1.
2. On the toolbar, select the **Update User** icon or select **Update User** from the resource context menu.

   The **Update User Wizard** dialog will appear.

3. You will be prompted for the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter any error requiring you to correct the previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

Field	Tips
Enter **PostgreSQL Database Administrator User**	This dialog requests a PostgreSQL Database Administrator User name for the specified database instance with connection and administrator privileges for the instance. **Note:** A validation script will verify connectivity using the value specified. A password protected instance will require a valid entry in the *.pgpass* file for the PostgreSQL Database Administrator User.
Confirm Update Action	This dialog requests confirmation of the update user change of the previous user value to the new user value.

4. Click **Update**. The PostgreSQL Database Administrator User will be updated on all servers where the resource is currently protected.

6.12.6. PostgreSQL Troubleshooting

This section provides a list of messages that you may encounter while creating and extending an SPS PostgreSQL resource hierarchy or removing and restoring a resource. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Messages from other SPS components are also possible. In these cases, please refer to the Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

- General Tips
- Tunables

6.12.6.1. PostgreSQL General Tips

The following error messages and conditions may be encountered while using the recovery kit.

Error	Solution
Unable to protect PostgreSQL database using the same port as another LK protected PostgreSQL database.	Verify the version of PostgreSQL includes a *postgresql.conf* file. In the *postgresql.conf* file, set the entry listen_address= to the IP address to be used with the database instance. **Note:** The format of the listen_address= in the *postgresql.conf* file is important as syntax errors can result in a failure to start the database server.
Unable to perform a manual switchover of version 8.X when clients are connected.	The default (smart) shutdown option fails to disconnect clients on a switchover. If shutdown continues to fail with connected clients, verify that the LKPGSQL_SDIRS tunable is not set. If the problem persists, set the LifeKeeper tunable LKPGSQL_IDIRS.
Unable to connect from a remote client to the database server.	To enable remote host login for PostgreSQL, refer to the *PostgreSQL Administration Guide* on configuring the *pg_hba.conf* file.
psql: connectDBStart() — connect() failed: No such file or directory. Is the postmaster running at 'localhost' and accepting connections on Unix socket '<port>'?"	Verify that the socket file exists and the instance is currently running. If the socket file resides in */tmp*, it may have been removed by a cron job that cleans up the */tmp* directory. Take the resource out of service and back in service. Then modify the cron job to leave PostgreSQL socket files.
PostgreSQL resource hierarchy fails to come in service but the database is running.	The database may have failed to respond to the LifeKeeper client request within the specified interval. Adjust the tunable LKPGSQL_CONN_RETRIES in */etc/default/LifeKeeper* to increase the number of seconds allowed for the recovery and restart of the PostgreSQL database instance.
PostgreSQL resource hierarchy fails local recovery following a postmaster crash with active client connections.	When a large number of active clients are connected to PostgreSQL, the database may be unable to properly restart until the client connections have terminated. In this scenario, it may be best to force client connections to terminate so that local recovery will be successful. The variable LKPGSQL_DISCONNECT_CLIENT can be set in */etc/default/LifeKeeper* to control the behavior of the PostgreSQL resource hierarchy in this scenario. When the value is set to 1(**true**), client processes will be sent a SIGTERM signal to force them to disconnect from the database. This action will only be taken if the postmaster process is not running during local recovery.
Unable to connect to database with error "WARNING: password file "*/home/<osuser>/.pgpass*" has world or group read access"	The *.pgpass* file permissions should be u=rw(0600). Change the permissions and owner of the *.pgpass* file.
FATAL: syntax error in file "*/<datadir>/postgresql.conf*" line 50, near token ".17"	The *postgresql.conf* file listen_address= entry does not contain proper quoting. Verify entries are valid and the entry is enclosed in proper quotes.

6.12.6.2. PostgreSQL Tunables

Error	Solution
LKPGSQL_KILLPID_TIME	Time to wait after a process id is killed before rechecking for this process.
LKPGSQL_CONN_RETRIES	Replaces LKPGSQLMAXCOUNT – number of times to try a client connection after an action (start or stop)
LKPGSQL_ACTION_RETRIES	Number of times to attempt start or stop action before failing the action command.
LKPGSQL_STATUS_TIME	Timeout for status command.
LKPGSQL_QCKHANG_MAX	Number of quickCheck script hangs allowed before a failover/sendevent is triggered for the database instance.
LKPGSQL_CUSTOM_DAEMON	Allows a user to specify additional aliases for the postgres daemons (default postmaster).
LKPGSQL_IDIRS	Replaces LKPGSQL_IPORTS – Contains datadir entries for instances that will be shutdown using the immediate option only.
LKPGSQL_SDIRS	Contains datadir entries for instances that will be shutdown using the smart option.
LKPGSQL_DISCONNECT_CLIENT	Controls the behavior the PostgreSQL resource hierarchy during a database failure scenario. When the value is set to 1(**true**), client processes will be sent a SIGTERM signal to force them to disconnect from the database. This action will only be taken if the postmaster process is not running during local recovery. **Note:** This parameter cannot be used for PostgreSQL 8.2 and later.
LKPGSQL_DISCONNECT_CLIENT_BYTAG	Similar to LKPGSQL_DISCONNECT_CLIENT, this setting limits the action to the comma separated list of tags specified by this tunable. **Note:** This parameter cannot be used for PostgreSQL 8.2 and later.
LKPGSQL_RESUME_PROC	Determines if process found in the stopped state (state = ~T) will be resumed when detected or ignored.
LKPGSQLDEBUG	Turns on debug for PostgreSQL database kit as well as for the postgres database. Valid entry range: 0 – 5. Larger numbers produce greater debug information. This tunable will be passed on to the postmaster database using the option –d <LKPGSQLDEBUG>.

6.13. Postfix Recovery Kit Administration Guide

Postfix plays a variety of roles, all critical to the proper flow of email. It listens on the network for incoming mail, transports mail messages to other servers, and delivers local mail to a local program.

The LifeKeeper for Linux Postfix Recovery Kit provides a mechanism to recover Postfix from a failed primary server to a backup server in a LifeKeeper environment. Both LifeKeeper and Postfix ensure data integrity throughout the course of the failover process without significant lost time or human intervention.

Document Contents

This guide contain the following topics:

- Documentation and References. Provides a list of LifeKeeper for Linux documentation and where to find them
- Requirements A description of the hardware and software necessary to properly setup, install, and operate the Postfix Recovery Kit. Refer to SIOS Protection Suite Installation Guide for specific instructions on how to install or remove LifeKeeper for Linux software.Postfix Recovery Kit.
- Configuring the LifeKeeper for Linux Postfix Recovery Kit. A description of the procedures required to properly configure the Postfix Recovery Kit.
- Postfix Configuring Validation. Provides steps for validating the Postfix configuration prior to creating the Postfix resource hierarchy.
- LifeKeeper Configuration Tasks. A description of the tasks for creating and managing your Postfix resource hierarchies using the LifeKeeper GUI.
- Create a Dependency with the Mailbox Spool Resource. Describes how to manually create a dependency between the Postfix resource and the Mailbox Spool file system resource.
- Testing Your Resource Hierarchy. Describes steps for testing your Postfix resource hierarchies using the LifeKeeper GUI and command-line interface.
- Troubleshooting. A list of LifeKeeper for Linux error messages including a description for each.

Documentation and References

The following LifeKeeper product documentation is available from SIOS Technology Corp.:

- SPS for Linux Release Notes

- SPS for Linux Technical Documentation
- SIOS Protection Suite Installation Guide
- Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is available on the SIOS Technology Corp. website at:

http://docs.us.sios.com

Postfix Documentation and References

The following is a list of reference documents associated with the Postfix application and the LifeKeeper Postfix Recovery Kit:

- Postfix Man Page
- Red Hat Postfix Reference Manual

6.13.1. Postfix Hardware and Software Requirements

Your LifeKeeper configuration must meet the following requirements **prior** to the installation of the LifeKeeper for Linux Postfix Recovery Kit. Please see SIOS Protection Suite Installation Guide for specific instructions regarding the configuration of your LifeKeeper hardware and software.

Hardware Requirements

- **Servers** – LifeKeeper for Linux supported servers configured in accordance with the requirements described in SIOS Protection Suite Installation Guide and SPS for Linux Release Notes.

- **Data Storage** – The Postfix Recovery Kit can be used in conjunction both with shared storage and with replicated storage provided by the LifeKeeper Data Replication product.

Software Requirements

- **TCP/IP** software. Each server also requires the TCP/IP software.

- **LifeKeeper software**. You must install the same version of LifeKeeper software and any patches on each server.

- **LifeKeeper for Linux IP Recovery Kit**. You must have the same version of the LifeKeeper IP Recovery Kit on each server.

- **IP network interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

  Note: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation, and so forth.

- **Postfix software**. Each server must have the Linux distribution version of the Postfix software installed and configured before you can configure LifeKeeper and the Postfix Recovery Kit. The same version should be installed on each server. Consult the SPS for Linux Release Notes or your sales representative for the latest release compatibility and ordering information.

6.13.1.1. Postfix Recovery Kit Installation

Please refer to the SIOS Protection Suite Installation Guide for specific instructions on the installation and removal of the LifeKeeper for Linux software, including recovery kits.

6.13.2. Configuring the LifeKeeper for Linux Postfix Recovery Kit

This section describes the LifeKeeper for Linux Postfix Recovery Kit configuration details. It also contains information you should consider before you start to configure and administer the Postfix Recovery Kit. Please refer to SPS for Linux Technical Documentation for instructions on configuring LifeKeeper Core resource hierarchies.

6.13.2.1. Postfix Protection Objects

The Postfix Recovery Kit protects the following objects:

- Postfix service daemon
- Network socket of Postfix

  Create one or more virtual IP addresses for monitoring mail and assign them to the inet_interfaces parameter in the Postfix configuration file. If “all” is specified for the inet_interfaces parameter, then the local loopback address is used for monitoring. The supported SMTP / SMTPS service ports supported by the Postfix smtpd daemon are 25 and 465 respectively.

- The queue directory (filesystem) Postfix uses

  If you need the mailbox spool area on another file system and need to protect it, you must create the file system hierarchy for it and create a dependency between the Postfix resource and this resource. Please refer to Create Dependency with Mailbox Spool Resource.

6.13.2.2. Postfix Configuration Requirements

- If the IP address used by the SMTP service is specified, the IP address should be a virtual IP address that is protected by Lifekeeper.
- main.cf

  The Postfix Recovery Kit refers to the value of the following parameters:

  - º mail_owner
  - º setgid_group
  - º daemon_directory
  - º command_directory
  - º process_id_directory
  - º inet_interfaces

    Specify the virtual IP addresses to be monitored. One or more may be specified. Use “all” to specify all IP addresses.
  - º queue_directory
  - º mail_spool_directory
- master.cf

  You must specify the following:

  - º A smtp(s) service entry to start smtpd.
- The directory specified for the queue_directory value must be on shared storage. This is necessary so that the file system of this directory can be LifeKeeper protected.
- If the system has a mailbox spool, the directory specified for the mail_spool_directory value has to be on shared storage.
- Owner id of postfix has to be the same id on all cluster servers.
- Group id of postdrop (setgid_group) has to be the same id on all cluster servers.
- Auto startup at the time of the system activation must be disabled because Postfix service is

controlled by LifeKeeper.

6.13.2.3. Port and TCP Interface Definition and the Postfix Recovery Kit

The Postfix Recovery Kit listens to the port specified in the SMTP entry in the Postfix configuration file (master.cf). If the port is specified as a service name (e.g., smtp) then the port number is looked up in the /etc/services file (smtp is ”25” and smtps is ”465”).

```
smtp      inet  n       -       n       -       -       smtpd
```

6.13.2.4. DNS, Postfix and LifeKeeper

DNS offers a mechanism (MX Records) for specifying backup or alternate hosts for mail delivery. This mechanism also allows hosts to assume mail-handling responsibilities for other hosts that are not configured to accept mail, such as a null client. MX records also provide a mechanism of forcing all mail to go to the hub machine or mail server. MX records specify a mail exchanger for a domain name (i.e. a host that will process and/or forward mail for the specified hostname). As an example, this is done by adding entries into the DNS server as follows:

```
himalaya.sc.steeleye.com IN      MX      10  relay.steeleye.com.
```

In the example, the server himalaya.sc.steeleye.com has an MX record that will cause mail for this server to be delivered to relay.steeleye.com. The server which is to be LifeKeeper protected should not have any MX records. The LifeKeeper protected alias IP address that is used during the Postfix resource hierarchy creation should be used for MX records instead.

6.13.2.5. Postfix Configuration Examples

Configuration 1: Active/Standby Configuration Example

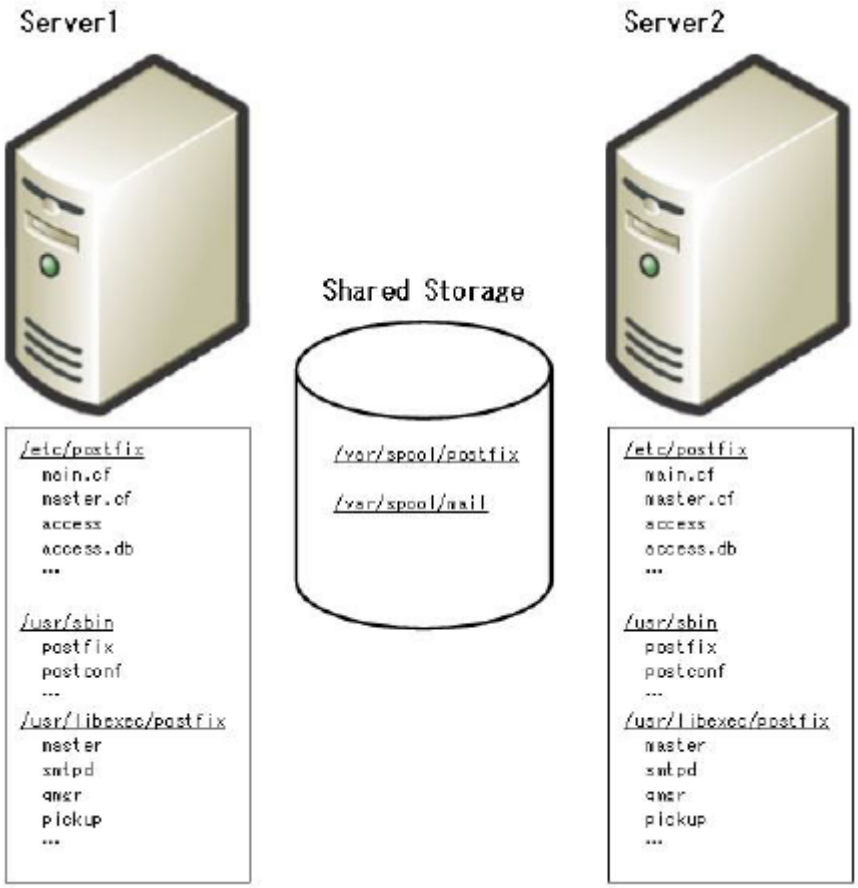


Figure 1: Typical LifeKeeper Active/Standby Postfix Environment 1

- The Postfix configuration files are on both servers
- The Postfix executable files are on both servers.
- The queue area (e.g. /var/spool/postfix) is on shared storage.
- The spool area (e.g. /var/spool/mail) is on shared storage.

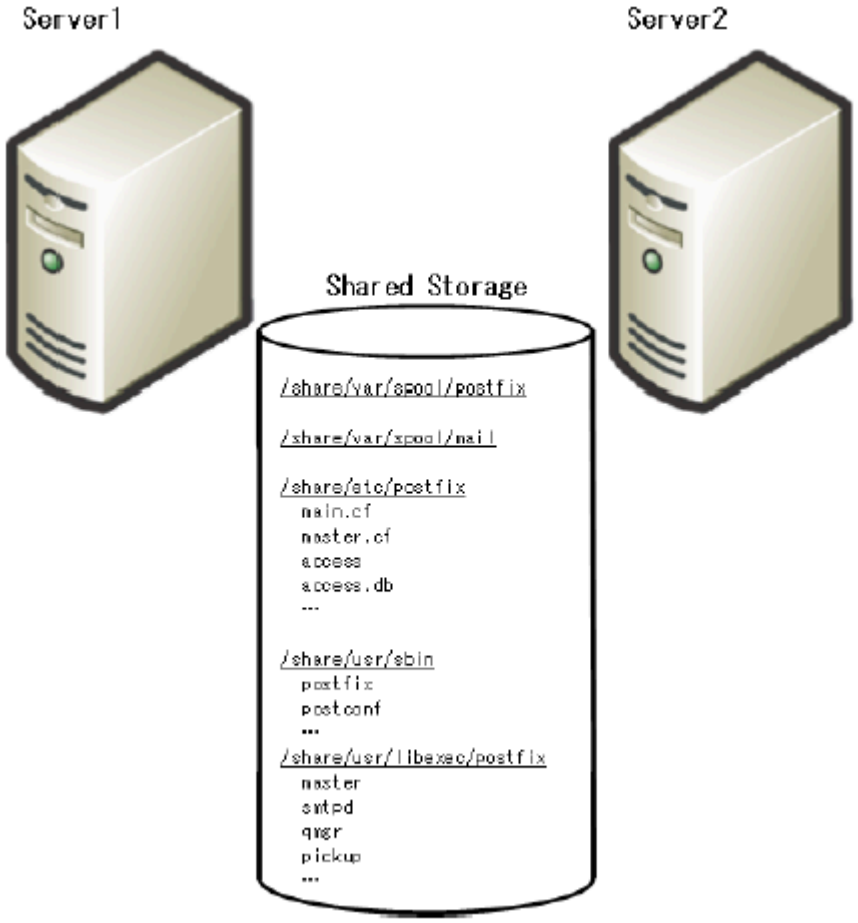


Figure 2: Typical LifeKeeper Active/Standy Postfix Environment 2

- The Postfix configuration files are on shared file system.
- The Postfix executable files are on shared file system.
- The queue area (e.g. /var/spool/postfix) is on shared storage.
- The spool area (e.g. /var/spool/mail) is on shared storage.

Configuration 2: Active/Active Configuration Example

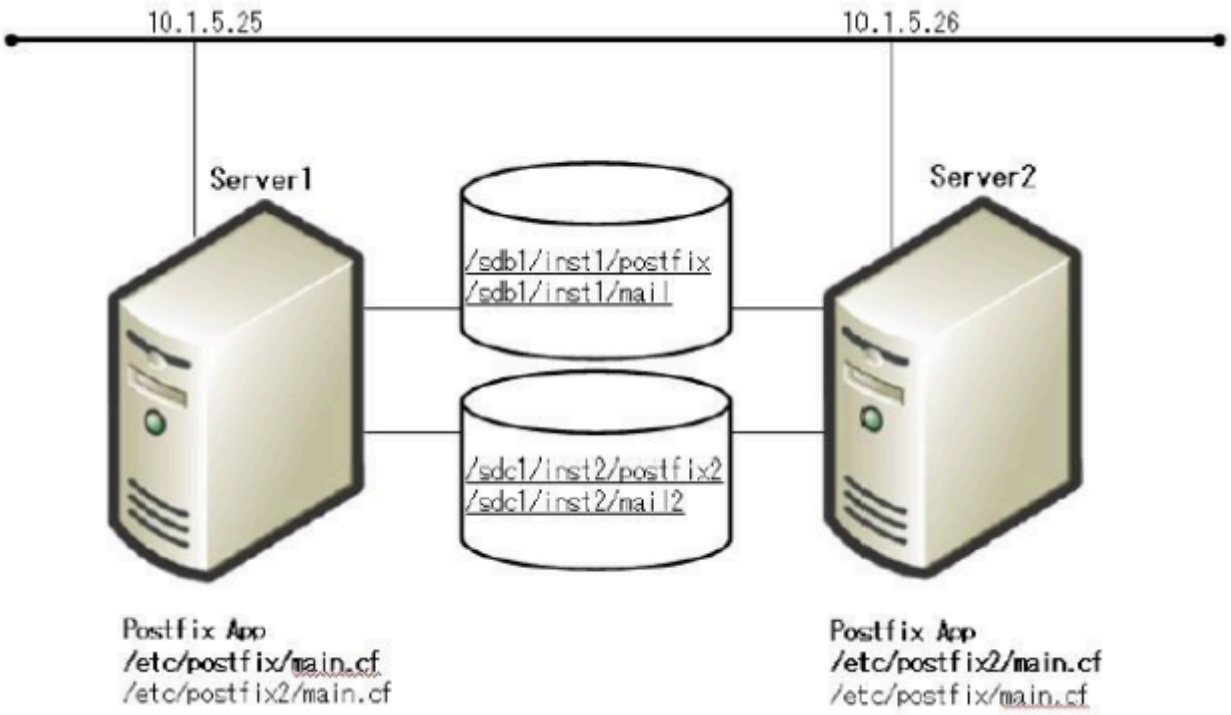


Figure 3: Typical LifeKeeper Active/Active Postfix Environment

[Server1 (Instance 1 is active)]

The Postfix configuration file: /etc/postfix

The Postfix executable files: /usr/sbin

The queue area: /sdb1/inst1/postfix

The spool area: /sdb1/inst1/mail

<main.cf>

inet_interfaces = 10.1.5.25, localhost

[Server2 (Instance 2 is active)]

The Postfix configuration file: /etc/postfix2

The Postfix executable files: /usr/sbin

The queue area: /sdc1/inst2/postfix2

The spool area: /sdc1/inst2/mail2

<main.cf>

inet_interfaces = 10.1.5.26

alternate_config_directories = /etc/postfix2

The following figure shows the Postfix resource hierarchies displayed in the LifeKeeper GUI:

6.13.3. Postfix Configuration Validation

This section shows a method to check the systems by using the Typical LifeKeeper Postfix Environment 1 as an example before you start to create resources in LifeKeeper.

Postfix Configuration Validation Steps

1. Postfix Configuration

   The Postfix configuration files are on both servers.

```
main.cf (extract)

daemon_directory = /usr/libexec/postfix
command_directory = /usr/sbin
process_id_directory = pid
inet_interfaces = localhost , 192.168.0.10
mail_spool_directory = /var/spool/mail
queue_directory = /var/spool/postfix
```

```
master.cf (extract)

smtp      inet  n       -       n       -       -       smptd
```

2. Bring up virtual IP address for SMTP

   You must bring up virtual IP address for SMTP. You can configure it by using the “ifconfig” command or creating a LifeKeeper IP resource.

```
# ifconfig eth0:1 192.168.0.10 netmask 255.255.255.0 up
```

3. Mount the shared filesystem for queue area

```
# mkfs.ext3 /dev/sda1
# mount -t ext3 /dev/sda1 /mnt/queue
# mkdir -p /mnt/queue/postfix
```

```
# cp -rp /var/spool/postfix/* /mnt/queue/postfix/
# mv /var/spool/postfix /var/spool/postfix.org
# ln -s /mnt/queue/postfix /var/spool/postfix
# postfix check
```

4. Mount the shared filesystem for spool area

```
# mkfs.ext3 /dev/sdb1
# mv /var/spool/mail /var/spool/mail.org
# mkdir -p /var/spool/mail
# mount -t ext3 /dev/sdb1 /var/spool/mail
```

5. Start Postfix

```
# postfix –c /etc/postfix start
postfix/postfix-script: starting the Postfix mail system
```

6. Verify processes and socket for Postfix

```
# netstat -pltun | grep master
tcp    0   0   127.0.0.1:25        0.0.0.0:*       LISTEN      15931/master
tcp    0   0   192.168.0.10:25     0.0.0.0:*       LISTEN      15931/master

# ps -ef | grep -v grep | grep postfix
root      15931      1  0  16:11  ?  00:00:00  /usr/libexec/postfix/master
postfix   15932  15931  0  16:11  ?  00:00:00  pickup -l -t fifo -u
postfix   15933  15931  0  16:11  ?  00:00:00  qmgr -l -t fifo -u
```

7. Stop Postfix

```
# postfix -c /etc/postfix stop
postfix/postfix-script: stopping the Postfix mail system
```

If you cannot start or stop Postfix in steps 5-7, please check the Postfix error messages. Once there are no error messages in the log file, the configuration is correct. Next, repeat steps 1-7 on all systems in the cluster and confirm that the configuration is correct.

6.13.4. LifeKeeper Configuration Tasks for Postfix

You can perform all LifeKeeper for Linux Postfix Recovery Kit administrative tasks via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer, and monitor Postfix resources.

The following tasks are available for configuring the LifeKeeper for Linux Postfix Recovery Kit:

- Create a Resource Hierarchy – Creates a Postfix resource hierarchy.
- Delete a Resource Hierarchy – Deletes a Postfix resource hierarchy.
- Extend a Resource Hierarchy – Extends a Postfix resource hierarchy from the primary server to the backup server.
- Unextend a Resource Hierarchy – Unextends (removes) a Postfix resource hierarchy from a single server in the LifeKeeper cluster.
- Create Dependency – Creates a child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete Dependency – Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service – Activates a resource hierarchy.
- Out of Service – Deactivates a resource hierarchy.
- View /Edit Properties – View or edit the properties of a resource hierarchy.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You can also perform most of these tasks:

1. From the toolbar, right – click on a global resource in the left pane of the status display.
2. Right – click on a resource instance in the right pane of the status display.

   Note: Using the right-click method allows you to avoid entering information that is required when using the **Edit** menu.

6.13.4.1. Creating a Postfix Resource Hierarchy

After you have completed the necessary setup tasks, use the following steps to define the Postfix resource hierarchy.

IMPORTANT: The alias IP address should be under LifeKeeper protection before creating the Postfix resource instance.

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the menu, select **Create Resource Hierarchy**.

   The **Create Resource Wizard** dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select **Postfix Mail Server** and click **Next**.

3. You will be prompted to enter the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter an error requiring you to correct previously entered information. You may click Cancel at any time to cancel the entire creation process.

Field	Tips
Switchback Type	Choose either *intelligent* or *automatic*. This dictates how the Postfix instance will be switched back to this server when the server comes back up after a failover. The switchback type can be changed later from the General tab of the Resource Properties dialog box. **Note**: The switchback strategy should match that of the IP or File System resource to be used by the Postfix resource. If they do not match the Postfix resource, creation will attempt to reset them to match the setting selected for the Postfix resource.
Server	Select the **Server** on which you want to create the hierarchy.
Postfix Binary Location	Enter the directory path name where the Postfix daemon resides.
Postfix server Config File Location	Enter the directory path name where the Postfix configuration file (main.cf) resides.
Queue Root Directory	Enter the directory path name of the Postfix queue directory. The default is decided from the configuration file, which you selected in the previous dialog box. The Postfix queue directory must be on a shared disk. If the Postfix queue directory is a symbolic link, the dialog box will show the root directory of the symbolic link pointing to the directory's original location.

Root Tag	Either select the default root tag offered by LifeKeeper, or enter a unique name for the resource instance on this server. The default is postfix-on- *<queue directory path>*. You may use letters, numbers and the following special characters: – _ . /

4. Click **Create**. The Create Resource Wizard will then create your Postfix resource hierarchy. LifeKeeper will validate the data entered. If LifeKeeper detects a problem, an error message will appear in the information box.

5. An information box will appear indicating that you have successfully created a Postfix resource hierarchy, and you must extend that hierarchy to another server in your cluster in order to achieve failover protection. Click **Next**.

6. Click **Continue**. LifeKeeper will then launch the *Pre-Extend Wizard*. Refer to Step 2 under Extending Your Hierarchy (below) for details on how to extend your resource hierarchy to another server.

6.13.4.2. Extending a Postfix Resource Hierarchy

This operation can be started from the **Edit** menu, or initiated automatically upon completing the **Create Resource Hierarchy** option, in which case you should refer to Step 2 below.

1. On the **Edit** menu, select Resource, then Extend Resource Hierarchy. The *Pre-Extend Wizard* appears. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the LifeKeeper **Extend Resource Hierarchy** defaults and want to bypass the prompts for input/ confirmation, click **Accept Defaults**.

2. The *Pre-Extend Wizard* will prompt you to enter the following information.

   Note: The first two fields appear only if you initiated the Extend from the **Edit** menu.

Field	Tips
Template Server	Enter the server where your Postfix resource is currently in service.
Tag to Extend	Select the Postfix resource you wish to extend.
Target Server	Enter or select the server you are extending to.
Switchback Type	Select either *intelligent* or *automatic*. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. **Note**: Remember that the switchback strategy must match that of the dependent resources to be used by the Postfix resource.
Template Priority	Select or enter a priority for the template hierarchy. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (the number 1 indicates the highest priority). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note**: This selection will appear only for the initial extend of the hierarchy.
Target Priority	Either select or enter the priority of the hierarchy for the target server.

3. After receiving the message that the pre-extend checks were successful, click **Next**.

4. LifeKeeper will display a series of dialog boxes for the Postfix resource to be extended, some of which cannot be edited.

Field	Tips

Root Tag	LifeKeeper will provide a default tag name for the new Postfix resource instance on the target server. The default tag name is the same as the tag name for this resource on the template server. If you enter a new name, be sure it is unique on the target server. You may use letters, numbers and the following special characters: – _ . /
Binary Directory (Information Only)	This dialog box is for informational purposes only. You cannot change the Binary Directory that appears in the box.
Configuration Directory (Information Only)	This dialog box is for informational purposes only. You cannot change the Configuration Directory that appears in the box.

If the IP and Filesystem dependent resource are also being extended, LifeKeeper will display a series of dialog box for the resources, some of which cannot be edited.

Click **Extend**

5. After receiving the message “Hierarchy extend operations completed” click **Next Server** to extend the hierarchy to another server, or click **Finish** if there are no other extend operations to perform.

6. After receiving the message “Hierarchy Verification Finished”, click **Done**.

6.13.4.3. Unextending a Postfix Resource Hierarchy

To remove a resource hierarchy from a single server in the LifeKeeper cluster, do the following:

1. On the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.

2. Select the **Target Server** where you want to unextend the Postfix resource. It cannot be the server where the Postfix resource is currently in service. (This dialog box will not appear if you selected the Unextend task by right clicking on a resource instance in the right pane.) **Click Next**.

3. Select the Postfix hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the Unextend task by right clicking on a resource instance in either pane).

4. An information box appears confirming the target server and the Postfix resource hierarchy you have chosen to unextend. Click **Unextend**.

5. Another information box appears confirming that the Postfix resource was unextended successfully. Click **Done** to exit the Unextend Resource Hierarchy menu selection.

6.13.4.4. Deleting a Postfix Resource Hierarchy

It is important to understand what happens to dependencies and protected services when a Postfix hierarchy is deleted.

- **Dependencies**: When you choose to delete the Postfix hierarchy, only the Postfix resource will be deleted. Dependent IP and file system resources will not be removed.

- **Protected Services**: If you take the Postfix resource hierarchy out of service before deleting it, the Postfix daemons will be stopped. If you delete a hierarchy while it is in service, the Postfix daemons will continue running and offering services (without LifeKeeper protection) after the hierarchy is deleted.

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.

2. Select the **Target Server** where you will be deleting your Postfix resource hierarchy and click **Next**. (This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in either pane.)

3. Select the **Hierarchy to Delete**. (This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in the left or right pane.) Click **Next**.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete**.

5. Another information box appears confirming that the Postfix resource was deleted successfully.

6. Click **Done** to exit.

6.13.4.5. Create Dependency with Mailbox Spool Resource

If the Postfix queue directory and Mailbox Spool directory are on the same file system (LUN) on the shared disk, both directories are protected by creating the Postfix resource hierarchy and extending the Postfix resource hierarchy to another server in your cluster. If your spool directory is on another file system (LUN), you must create a file system resource for Mailbox Spool and create a dependency for the resource.

To create a resource instances and create dependencies for your Mailbox Spool directory, you should complete the following step:

1. Mount file system for your Mailbox Spool Directory.
2. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop down menu, select **Create Resource Hierarchy**.
3. Select File System from the drop down listing.
4. Select Switchback Type.
5. Select the Primary Server.
6. Select the Mount Point for the file system resource hierarchy.
7. Select or enter Root Tag.

Through this process, the file system resource is created on the primary server, and you must extend it to backup servers. Next, create dependencies for each file system resources to the Postfix resource. You should refer Creating Resource Dependency section of LifeKeeper for Linux Technical Documentation for specific instructions on how to create dependencies.

6.13.4.6. Testing Your Postfix Resource Hierarchy

You can test your Postfix resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Performing a Manual Switchover from the GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, then **Resource**, then finally **In Service** from the drop down menu. For example, an in service request executed on a backup server causes the application hierarchy to be placed in service on the backup server and taken out of service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out of Service** request, the application is taken out of service without bringing it in service on the other server.

Performing a Manual Switchover from the Command-Line Interface

You can initiate a manual switchover from the LifeKeeper command-line interface by the following steps on the server:

```
# /opt/LifeKeeper/bin/perform_action -t [tag-name] -a [restore|remove]
```

- **-t**
- This specifies the last resource instance that the action will be performed on. “tag-name” are the information elements that may be used to describe the resources in the hierarchy, the name can be checked from LifeKeeper GUI, or “lcdstatus” command.

- **-a**

  This specifies the resource action that will be performed. To bring the resource instance into service, specify restore, to take a resource out of service, specify remove.

Please refer to man pages of *perform action* for more details.

Recovery Operations

When the following failure occurs on the in service server, the Postfix Recovery Kit software performs Recovery:

- Failure in the Postfix resource

- Failure in IP resource relative to the Postfix resource

- Failure in file system resource relative to the Postfix resource
- Node Failure

When the primary server fails, the Postfix Recovery Kit software performs the following tasks:

- Brings the alias IP address into service on the backup server by bringing *in service* a logical interface on one of that server's physical network interfaces
- Mounts the file system(s) on the shared disk on that server
- Starts the daemon processes related to Postfix

Since session context is lost following recovery, after the recovery, Postfix users must reconnect using exactly the same procedures they used to connect originally.

6.13.5. Postfix Troubleshooting

This section provides a list of messages that you may encounter during the process of creating, extending, removing and restoring a LifeKeeper Postfix hierarchy, and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. Other messages from other LifeKeeper scripts and utilities are also possible. In these cases, please refer to the documentation for the specific script or utility. Messages in this section fall under these categories:

Hierarchy Creation

Hierarchy Extend

Hierarchy Remove, Restore and Recovery Error Messages

6.13.5.1. Postfix Hierarchy Creation Error Messages

Error	Error Message
No config path	The Postfix configuration path was not found. Please enter the configuration path.
main.cf not found in the configuration path	The file main.cf does not exist in the path specified. Please enter the correct path.
master.cf not found in the configuration path	The file master.cf does not exist in the configuration path. Please enter the correct path.
A value of inet_interfaces must be IPv4 or “all”	Please specify an IPv4 address or “all” for the inet_interfaces parameter in the main.cf file.
No execute path	Must specify the absolute path to the Postfix executables. Please enter the correct path.
Postfix command invalid	The Postfix command is invalid. Please verify the Postfix installation or command and enter the correct command.
<*queue directory*> is not found. This directory must exist on a shared filesystem	The mail queue directory(s) must be located on a shared filesystem. Please make sure your configuration is correct.
<*tag name*> not in service on the server	The tag name is not in service. Please create the IP resource and verify that the virtual IP address is active on the server.
Could not find IP resouce for “<*IP address*>”	The LifeKeeper IP resource for the IP address specified for the inet_interfaces parameter in main.cf is missing. Please create the LifeKeeper IP resource.

6.13.5.2. Postfix Hierarchy Extend Error Messages

Error	Error Message
postfix id does not match between servers	The Postfix uid does not match on the servers in the cluster. Please set the same uid for the user “postfix” on the cluster servers.
postdrop gid does not match between servers	The Postfix postdrop gid does not match on the servers in the cluster. Please set the same gid for the group “postdrop” on the cluster servers.

6.13.5.3. Postfix Resource In-Service / Out-of-Service / Health Monitoring Error Messages

Error	Error Message
Master process of postfix is not running	The master process of Postfix is not running. Please check the Postfix error log.
Failed in a check by postfix command	Postfix command check option has failed. Please check the Postfix configuration file or Postfix environment.
Couldn't start postfix resource	The Postfix resource could not start. Please check the Postfix error log.
Failed in a stop process by kill command	The kill command failed to stop Postfix. Please check the Postfix error log.
PID <*pid*> does not exist. postfix may have already stopped	The Postfix pid does not exist. Please check the Postfix error log and Postfix processes. The Postfix process may have been stopped and then restarted and assigned another pid.
Check script was not able to be connected to a socket (*vip:port*)	The check script was not able to connect to the socket for service. Please check the Postfix configuration file and the Postfix owner.
Execute files (postfix or postconf command) is not an executable file	The files postfix or postconf does not exist or are not executable. The files are located in the executable path that was specified when the resource was created. Please check these files.
Configuration files (main.cf or master.cf) does not exist	The Postfix configuration files main.cf or master.cf does not exist or is not readable. The files and located in the configuration path that was specified when the resource was created. Please check these files.
The postfix owner <*owner name*> does not exist	The Postfix owner does not exist. Please check the Postfix configuration and Postfix owner.
The postdrop group id does not match and attribute of queue directory	The postdrop group id does not match the group id associates with the files in the mail queue directory. Please check the Postfix configuration file.

6.14. Route53 Recovery Kit Administration Guide

Route53 Recovery Kit provides a mechanism for updating Amazon Route 53 DNS information corresponding to a virtual IP address and an actual IP address information of IP resources that are in dependency relation when switching to a failed primary server to a backup server

SIOS Protection Suite Documentation

The following is a list of SIOS Protection Suite for Linux related information available from SIOS Technology Corp.

- SPS for Linux Technical Documentation
- SPS for Linux Release Note
- SIOS Technology Corp. Documentation

For the details, please refer to Amazon Route 53 Documentation.

Note: Amazon Web Services, the “Powered by Amazon Web Services” logo, “AWS”, “Amazon EC2”, “EC2”, “Amazon Elastic Compute Cloud”, “Amazon Route 53”, “Amazon Virtual Private Cloud”, and “Amazon VPC” are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries.

6.14.1. Route53 Recovery Kit Requirements

Prior to installing and configuring the Route53 Recovery Kit, be sure your configuration meets the following requirements.

- AWS Command Line Interface (AWS CLI) must be installed on all EC 2 instances. Refer to Installing AWS Command Line Interface" for installation information.
- Instances need to have an access to Amazon Route 53 service endpoint, route53.amazonaws.com, with HTTPS protocol. Please configure EC2 and the OS properly.
- Register an appropriate domain name for Amazon Route 53.
- In order for LifeKeeper to operate AWS, an IAM user or an IAM role with the following access privilege is required. Please configure IAM roles for Amazon EC2 or the AWS CLI appropriately so that it can be accessed from root user of the Amazon EC2 instance.
    - ◦ route53:GetChange
    - ◦ route53:ListHostedZones
    - ◦ route53:ChangeResourceRecordSets
    - ◦ route53:ListResourceRecordSets

Note: If the path name of AWS CLI executable files is not specified on the "PATH" parameter in the LifeKeeper defaults file */etc/default/LifeKeeper*, you must append the path name of AWS CLI executable files to the "PATH" parameter.

LifeKeeper Software:

You need to install the same version of LifeKeeper software and patches on each server. For the specific LifeKeeper requirements, please refer to Technical Documentation or SPS for Linux release note

6.14.2. Route 53 Configuration

To configure LifeKeeper to provide the required protection capability and flexibility, you need to know the configuration requirements. You also need to understand Amazon, Amazon Virtual Private Cloud (VPC), Amazon Elastic Compute Cloud (EC2), Amazon Route 53 and hierarchy configuration options of the user system. In addition to the configuration planning, this section also describes the specific tasks required to set up the Recovery Kit.

Specific Configuration Considerations for Route53 Resources

The following configuration tasks for Route53 resources are described in this section. They are unique to a Route53 resource instance and different for each recovery kit.

- Creating a Resource Hierarchy: Creates an application resource hierarchy in your LifeKeeper cluster.
- Deleting a Resource Hierarchy: Deletes a resource hierarchy from all servers in your LifeKeeper cluster.
- Extending Your Hierarchy: Extends a resource hierarchy from the primary server to a backup server.
- Unextending Your Hierarchy: Unextends (removes) a resource hierarchy from a single server in your LifeKeeper cluster.
- Adjusting Route53 Recovery Kit Tunable Values: Tunes characteristics of the overall behavior of the Route53 Recovery Kit.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation. They are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency: Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency: Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service: Brings a resource hierarchy into service on a specific server.
- Out of Service: Takes a resource hierarchy out of service on a specific server.
- View Properties / Edit Properties: View or edit the properties of a resource hierarchy on a specific server.

6.14.2.1. Creating a Route53 Resource Hierarchy

To create a resource instance from the primary server, complete the following steps.

1. From the LifeKeeper GUI menu, select **Edit** then **Server** From the drop down menu, select **Create Resource Hierarchy**.

2. A dialog box will appear with a drop down list showing all of the recognized recovery kits installed within the cluster. Select **Amazon Route53** from the drop down list and click **[Next]**

3. You will be prompted to enter the following information. (When the Back button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful in the event that you need to correct previously entered information.)

   Note: you click the Cancel button at any time when creating your hierarchy, LifeKeeper will cancel the entire creation process.

Field	Tips
Switchback Type	This dictates how the Route53 instance will be switched back to this server when the server recovers after a failover. You can choose either intelligent or automatic. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths. **Note**:The switchback type can be changed later from the General tab of the Resource Properties dialog box.
Server	Select the Server for the Route53 resource (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop down list.
Domain name (Route53 hosted zone)	Route53 hosted zones are listed in the drop down list. Select the domain name to use.
Host Name (Not FQDN)	Enter the host name.
IP resource	Select the IP resource. This is the virtual IP address or the actual IP address that is protected by LifeKeeper.
Route53 Resource Tag	Select or enter a unique Route53 Resource Tag name for the Route53 resource instance you are creating. This field is populated automatically with a default tag name, route53-<host name>.

4. Click **Create**. The Create Resource Wizard will then create your Route53 resource

5. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your Route53 resource hierarchy. If LifeKeeper detects a problem an ERROR will appear in the information box. If the validation is successful your resource will be created. Click **Next**

Another information box will appear confirming that you have successfully created a Route53 resource hierarchy. You must extend that hierarchy to another server in your cluster in order to place it under LifeKeeper protection

When you click **Continue**, LifeKeeper will launch the Pre-Extend configuration task. Refer to Extending Your Hierarchy for details on how to extend your resource hierarchy to another server.

If you click **[Cancel]** now, another dialog box will appear alerting you that you will need to manually extend your Route53 resource hierarchy to another server at some other time to put it under LifeKeeper protection.

6.14.2.2. Deleting a Route53 Resource Hierarchy

To delete a resource hierarchy from all of the servers in your LifeKeeper environment, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resouce**. From the dropdown menu, select **Delete Resource Hierarchy**.

2. Select the name of the Target Server that you are deleting from your Route53 resource hierarchy and click **Next**

   Note:This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in either pane.

3. Select the Hierarchy to Delete. Identify the resource hierarchy you wish to delete, highlight it then click **Next**

   Note:This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in the left or right pane.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to **Delete** to proceed.

5. An information box appears confirming that the Route53 resource was deleted successfully.

6. Click **Done** to exit.

6.14.2.3. Extending Your Route53 Resource Hierarchy

After you have created a hierarchy, you must extend that hierarchy to another server in the cluster. There are three possible scenarios to extend your resource instance from the template server to a target server.

- Continue from creating the resource into extending that resource to another server.
- Enter the Extend Resource Hierarchy task from the edit menu as shown below.
- Right click on an unextended hierarchy in either the left or right hand pane.

Each scenario takes you through the same dialog boxes (with a few exceptions, detailed below).

1. If you are entering the Extend wizard from the LifeKeeper GUI menu, select **Edit** , then **Resource** . From the drop down menu, select **Extend Resource Hierarchy**. This will launch the Extend Resource Hierarchy wizard. If you are unfamiliar with the Extend operation, click **Next** . If you are familiar with the LifeKeeper Extend Resource Hierarchy defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults**.

2. The Pre-Extend Wizard will prompt you to enter the following information

   Note: The first two fields appear only if you initiated the Extend from the Edit menu. It should be noted that if you click Cancel at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extension process to that particular server. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

Field	Tips
Switchback Type	Select the Switchback Type. This dictates how the Route53 instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either intelligent or automatic. * Intelligent switchback requires administrative intervention to switchback the instance to the primary/original server. * Automatic switchback means the switchback will occur as soon as the primary server comes back on line and reestablishes LifeKeeper communication paths. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box.

Template Priority	Select or enter a Template Priority. This is the priority for the Route53 hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note**: This selection will appear only for the initial extend of the hierarchy.
Target Priority	Select or enter the Target Priority. This is the priority for the new extended Route53 hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). **Note**: LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities do not need to be consecutive and no two servers can have the same priority for a given resource.

3. An information box will appear explaining that LifeKeeper has successfully checked your environment and that all the requirements for extending this Route53 resource have been met. If there were some requirements that have not been met, LifeKeeper will not allow you to select the **Next** button, and the **Back** button will be enabled. If you click **Back**, you can make changes to your resource extension according to any error messages that may appear in the information box. If you click **Cancel** now, you will need to manually extend your Route53 resource hierarchy to another server to put it under LifeKeeper protection. When you click **Next** , LifeKeeper will launch you into the Extend Resource Hierarchy configuration task.

4. The Extend Resource Hierarchy configuration task will prompt you to enter the following information

Field	Tips
Route53 Resource Tag	Select or enter the Route53 Resource Tag. This is the resource tag name to be used by the Route53 resource being extended to the target server. **Note**:The field in not editable.

5. An information box will appear verifying that the extension is being performed. Click **Next Server** if you want to extend the same Route53 resource instance to another server in your cluster. This will repeat the Extend Resource Hierarchy operation. If you click **Finish**, LifeKeeper will verify that the extension of the Route53 resource was completed successfully.

6. Click **Done** to exit from the Extend Resources Hierarchy menu selection.

   Note: Be sure to test the functionality of the new instance on all servers.

6.14.2.4. Unextending Your Route53 Resource Hierarchy

To unextend a hierarchy complete the following steps:

1. From the **LifeKeeper GUI menu**, select **Edit**, then **Resource**. From the dropdown menu, select **Unextend Resource Hierarchy**.

2. Select the Target Server that you are unextending from the Route53 resource. It cannot be the server that the Route53 resource is currently in service on. Click **Next**.

   Note: If you selected the **Unextend** task by right-clicking from the right pane on an individual resource instance, the dialog box will not appear.

3. Select the Route53 Hierarchy to unextend. Click **Next**

   Note: If you selected the **Unextend** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, the dialog will not appear.

4. An information box will appear confirming the target server and the Route53 resource hierarchy you have chosen to unextend. Click **Unextend**.

5. An information box will appear confirming the Route53 resource hierarchy you have chosen to unextend.

6. Click **Done** to exit.

6.14.2.5. Adjusting Route53 Recovery Kit Tunable Values

For the parameters can be configured in the Route53 Recovery Kit, refer to Parameter Lists.

6.14.2.6. Route53 Resource Monitoring and Recovery

The Route53 resource monitors the normality of the retrieval of the DNS A record registered at the time of creation and the association with the virtual IP address. The monitoring process is as follows.

1. Obtain the address set in the Route 53 A record with API. If it fails to obtain the record, it will retry 3 additional times waiting 2 seconds between attempts (by default). After the third unsuccessful attempt it will stop the monitoring and record the failure in the log.

2. Obtain an IP address from the dependent IP resource and compare it with the IP address in the DNS A record information. If the IP address information matches, then exit with a success as no errors exist. If the IP addresses do not match then exit with a failure to initiate a local recovery.

6.14.2.7. Route53 User System Setup

Ran IP resource that required for creating the Route53 resource can be either the virtual IP resource or the actual IP resource (resource for the primary IP address that is configured for NIC).

When Using the Virtual IP Resource

When using the virtual IP resource for a child resource of the Route53 resource, you need to reconfigure the route table so that the communication with the virtual IP address to the backup server is enabled when switching over the resource. Please use the Recovery Kit for EC2 along with the Route53 Recovery Kit. For details, please refer to the Recovery Kit for EC2 document

When Using the Actual IP Resource

No additional information needs to be configured when using the actual IP resource for a child resource for the Route53 resource. However, because the destination IP address will be changed every time the switch over occurs, please note that the connection should be established with the host name that is protected by the Route53 resource.

6.14.3. Route53 Troubleshooting

The Message Catalog provides a listing of all messages that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. This full listing may be searched for any error code received.

Updating the record associated with the Route53 resource startup may take time

Amazon provides the following information regarding the propagating speed of changes made to DNS record.

Amazon Route 53 FAQs
Q: How quickly will changes I make to my DNS settings on Amazon Route 53 propagate globally?
https://aws.amazon.com/route53/faqs/?nc1=h_ls

The Route53 resource checks the status of updates to the DNS record using the Route53 API. It considers that updates are completed when receiving INSYNC status, and retries the status checking when it receives PENDING status. As a result, the Route53 resource may result in a startup failure when it takes a long time to propagate updates to the DNS even though the record is updated successfully for the Route53 resource startup process.

If the startup of the Route53 resource fails, check the Route53 management console to make sure that the A record is updated correctly. If it is updated, updates to the relevant DNS service have been completed. Updates to LifeKeeper is required to propagate the updates to the DNS service. Restart the Route53 resource from LifeKeeper GUI

If you encounter the startup failure of the Route53 resource all the time due to the above mentioned reason, increase the number of the value of “ROUTE53_CHANGEID_TRY_COUNT” in /etc/default/LifeKeeper to 6 or 7 (the default value is 5). Restart of LifeKeeper or the OS is not required for this change.

Correctly set TTL value of the DNS record

An access from a client after a switchover or a failover uses the DNS information cache that each client holds until the time set as TTL is passed. If the longer TTL value is set, access attempts to the address before switching increase and unexpected problems may occur. If the shorter TTL value is set, DNS resolution often occurs and network load increases. Please set the TTL value as short as possible according to your environment.

Set the “ROUTE53_TTL” for the TTL value in /etc/default/LifeKeeper. The unit should be seconds

6.15. Samba Recovery Kit Administration Guide

The LifeKeeper for Linux Samba Recovery Kit provides fault resilient protection for Samba file and print shares on a Linux server existing in a heterogeneous network. This enables a failure on the primary Samba server to be recovered on a designated backup server without significant lost time or human intervention.

Document Contents

This guide contain the following topics:

- Requirements. A description of the hardware and software necessary to properly setup, install, and operate the Samba Recovery Kit. Refer to SIOS Protection Suite Installation Guide for specific instructions on how to install or remove LifeKeeper for Linux software.Samba Recovery Kit .
- Samba Recovery Kit Overview. Provides a brief description of the Samba Recovery Kit's features and functionality.
- Configuring the LifeKeeper for Linux Samba Recovery Kit. A description of the procedures required to properly configure the Samba Recovery Kit.
- LifeKeeper Configuration Tasks. A description of the tasks for creating and managing your Samba resource hierarchies using the LifeKeeper GUI.
- Samba Hierarchy Administration. Provides information about tasks that may be required after your Samba resources are created.
- Troubleshooting. A list of LifeKeeper for Linux error messages including a description for each.

Documentation and References

The following LifeKeeper product documentation is available from SIOS Technology Corp.:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- SIOS Protection Suite Installation Guide
- Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is available on the SIOS Technology Corp. website at:

http://docs.us.sios.com

6.15.1. Samba Recovery Kit Requirements

Your LifeKeeper configuration must meet the following requirements **prior** to the installation of the Samba Recovery Kit. Please see the SIOS Protection Suite Installation Guide for specific instructions regarding the configuration of your LifeKeeper hardware and software.

- **Servers**.The Recovery Kit requires two or more servers configured in accordance with the requirements described in the SPS for Linux Installation Guide. See the SPS for Linux Release Notes for supported Linux distributions.
- **LifeKeeper software**. You must install the same version of LifeKeeper software and any patches on each server.
- **LifeKeeper IP Recovery Kit**. You must have the same version of the LifeKeeper IP Recovery Kit on each server.
- **IP network interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

> **Note**: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation, and so forth.

- **TCP/IP Software**. Each server also requires the TCP/IP software.
- **Samba Software**. Samba is delivered with all Linux distributions that LifeKeeper for Linux supports. The Samba Recovery Kit has been tested on Red Hat, SUSE and Miracle Linux.

6.15.2. Samba Recovery Kit Installation

Please refer to the SPS for Linux Installation Guide for specific instructions on the installation and removal of the LifeKeeper for Linux software.

6.15.3. Samba Recovery Kit Overview

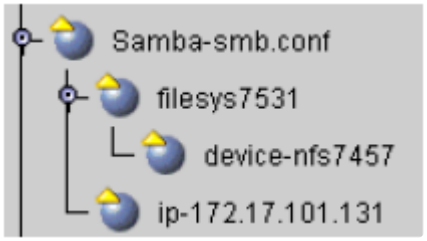
Samba is a suite of applications that speak the Server Message Block (SMB) protocol, allowing a Linux server to communicate in a heterogeneous network with servers and clients running Microsoft Windows products.

The Samba Recovery Kit enables LifeKeeper to protect Samba file and print shares on a Linux server. While Samba provides other services such as client authentication, Network Neighborhood browsing assistance and WINS name server resolution, this release of LifeKeeper does not protect these additional services. These other Samba services may coexist on a LifeKeeper server running as an unprotected instance of Samba as long as they adhere to the rules specified in the section Running Multiple Instances of Samba.

The Samba Recovery Kit provides a mechanism to recover protected Samba file and print shares from a failed primary server onto a backup server. LifeKeeper can detect failures either at the server level (via heartbeat) or resource level (by monitoring the Samba daemons) so that control of the Samba resources is transferred to a backup server.

Samba Resource Hierarchies

A typical Samba hierarchy will be comprised of a Samba resource, one or more file system resources, one or more IP resources, and possibly a print services resource. An example of a resource hierarchy protecting a Samba file share is shown below:



This *Samba-smb.conf* hierarchy protects one fileshare *filesys7531* (which is dependent upon the partition *device-nfs7457*), and one IP address 172.17.101.131. The following sections describes how the Samba resources are configured.

6.15.4. Configuring Samba with LifeKeeper

There are a number of Samba configuration considerations that need to be made before attempting to create LifeKeeper for Linux Samba resource hierarchies. Samba services on a Linux server are provided by two daemon processes, **smbd** and **nmbd**. These daemon processes are controlled by the values defined in the Samba configuration file which is described below.

6.15.4.1. The Samba Configuration File

While a Samba configuration file can contain many different directives, this description focuses on those aspects of the configuration file that affect your LifeKeeper configuration. Here are some key points about the configuration file:

- The configuration file is comprised of sections which correspond to the share (or service) they provide. Each section of the configuration file contains individual configuration options (or directives) unique to that share.
- The directives that are specified are sanity checked by the Samba Recovery Kit. Failure to set the directives properly will cause Samba resource creation to fail.
- The default configuration file for Samba is typically named *smb.conf* and resides in */etc* or */etc/samba* depending on the Linux distribution.
- Configuration file names must be unique within the cluster, or must reside in a different directory on each server for Active/Active configurations. The unique naming or location is required as the Samba Recovery Kit replicates a copy of the configuration file during extension to the same location on the backup server.
- Default set up can be used (and recommended) in the case to execute only one Samba daemon instance with active/standby set up. In this case, Samba daemon automatic startup must be disabled.
- If more than one version of Samba will be running in an Active/Standby configuration or if you use an Active/Active configuration, unique Samba configuration file names are required. See Running Multiple Instances of Samba for more requirements and information on running multiple versions of Samba.

The following sections of this document describe the sections of the configuration file, including the options required for LifeKeeper to protect a Samba share.

6.15.4.2. [Global] Section of the Configuration File

The [global] section is a special section in the configuration file that must appear in every configuration file used in a LifeKeeper Samba resource hierarchy. As the name implies, any options set in this section apply to all other sections unless that directive is called out specifically in the other sections. LifeKeeper requires that certain directives be defined in the [global] section. Some of these directives may not exist in a default configuration file and will therefore need to be added. They are:

- **netbios name** – The unique name given to the set of resources that comprise a LifeKeeper Samba hierarchy. This is the name used by clients to connect to the shares via the IP addresses defined in the interfaces directive (e.g. NetBIOS name = server1_print1).

- **interfaces** – The list of network addresses for the Linux Samba server to recognize and respond. Here are the requirements for properly configuring the interfaces directive:

    - º All subnets that are serviced by the Samba server must be listed. These must be LifeKeeper protected addresses and they must be unique within the cluster (no other Samba configuration file should use the same IP addresses).

    - º The interfaces directive can have multiple formats, IP addresses (dot version or host name), and network interface names and can make use of wild cards. However, the Samba Recovery Kit requires the use of the IP address in dot format (100.25.104.25) without wild cards.

    - º The subnet mask may be used in conjunction with the IP address but it is not used by LifeKeeper.

    - º LifeKeeper IP resources for the address specified in this directive must be created prior to the creation of the Samba resource hierarchy. Additionally, if the network mask is applied to the addresses in this directive it must match the mask used on the IP resource when it was created.

    - º Other non-protected instances of Samba should also use the interfaces directive, being sure to specify IP addresses different than those used by LifeKeeper.

      Note: Because of the use of the bind interfaces only directive discussed below, the interfaces directive may need to contain the localhost address of 127.0.0.1 to ensure proper operation of the utility **smbpasswd**. See Running Multiple Instances of Samba for information to help you determine whether the localhost address is needed.

- **lock directory** (or lock dir) – The name and location of a unique lock file location for the Samba instance on all servers. This directory must already exist on all servers in the cluster.

  Note: This directive is sometimes call **lock dir**. The Samba Recovery Kit will handle both directive

names.

- **bind interfaces only** – This directive tells **smbd** and **nmbd** processes to serve SMB requests on the addresses defined in the interfaces directive only. It must be set to *Yes*. Other non-protected instances of Samba running on the system must also have this directive set to *yes*. When set to yes, Samba will not service requests on subnets that are not listed in the interfaces directive nor will it service requests for other instances of Samba that may be running on the server.

6.15.4.3. [Homes] Section of the Configuration File

The [homes] section is a special section in the configuration file to handle connection attempts to a user's home directory on a Samba server if it is not specifically defined as a share. LifeKeeper does not protect users' home directories via this special share; therefore **it should be removed or commented out**. In order for the LifeKeeper Samba Recovery kit to protect a Samba share it must have a path directive specified. The path directive is used to determine the file system that the LifeKeeper Samba hierarchy protects. The [homes] section does not have a path specified by default because the path is determined at the time a user makes a connection to the Samba server. It is for this reason that this special share must be removed or comment out.

6.15.4.4. [Printers] Section of the Configuration File

The [printers] section handles connection attempts to printers on a Samba server if it is not specifically defined as a share. LifeKeeper does not protect printer shares via this special section nor through the global directive load printers. Each LifeKeeper-protected printer share must be defined in its own share section in the configuration file.

6.15.4.5. Share Definition Sections of the Configuration File

All other sections in the configuration file define the file and/or print shares that clients can attempt to access for this instance. A configuration file must have one or more shares defined. The Samba configuration file can contain file shares only, print shares only or a combination of both file and print shares. LifeKeeper does not limit the number of shares that can be defined, but one must realize that a failure relating to any one share could cause the entire hierarchy to be switched over to the backup server. The following directives must be defined for each share:

- **path** – This directive identifies the pathname at the root of the file or print share. The value determines the File System resource to be protected as part of the Samba hierarchy. If the LifeKeeper File System resource does not already exist when the Samba resource is created, LifeKeeper will create it for you.

  Note: This directive is sometimes called directory. The recovery kit will handle both directive names.

- **printable** – A *Yes* value indicates that the Samba share is used as a print spool repository for printing to Linux printers. If the share is to be a regular file share then set this directive to No or do not specify it, as it is *No* by default unless set to *Yes* in the [global] section. If this directive is set to Yes, then creation of a Samba hierarchy will require the existence of LifeKeeper Print Services resource that protects the printer defined via the printer name directive listed below.

  Note:This directive is sometimes called print ok. The recovery kit will handle both directive names.

- **printer name** – This directive defines the printer name used by the share and is used to find a Print Services instance that protects the named printer. The Print Services instance will become a child resource in the Samba hierarchy. If this directive is not defined for a printer share, the Samba Recovery Kit will use the share name as the printer name.

  Note: This directive is sometimes called printer. The kit will handle both directive names.

Note: The Samba configuration file allows the use of variable substitution for a number of directives. Variable substitution should not be used for any of the directives specified above unless the variable is resolved by the Samba utility *testparm*.

6.15.4.6. Running Multiple Instances of Samba

Running multiple instances of Samba in a LifeKeeper cluster introduces additional configuration requirements and restrictions. The following Samba configuration scenarios may involve multiple instances of Samba:

- Active/Standby configuration with multiple LifeKeeper Samba instances on one server
- Active/Active configuration with multiple LifeKeeper Samba instances on more than one server

Either of these configurations could include a non-LifeKeeper protected version of Samba.

As previously noted in Configuring the LifeKeeper for Linux Samba Recovery Kit, when running multiple instances of Samba each version must have a uniquely named configuration file, or the files must reside in different directories. Within each configuration file a number of directives are required and must be unique – in particular, netbios name, lock directory, pid directory, interfaces and log file. If these directives are not unique, Samba may not startup and therefore will not be available for client connections. Additionally, the lock, log file, and pid directories specified for each instance must exist on all servers in the cluster.

smbpasswd Utility and Multiple Instances of Samba

Although not required by LifeKeeper, some Samba utilities used by the Samba Recovery Kit expect to be able to open *smb.conf* in its default location. The Recovery Kit uses the **smbclient** and **nmblookup** utilities to connect to **smbd** and **nmbd** (respectively) in order to determine the health of the daemon processes while under LifeKeeper protection. These two utilities will not error out if they do not find *smb.conf* in its default location. However, *smb.conf* is required by the **smbpasswd** utility to be in its default location.

smbpasswd is used to maintain the *smbpasswd* file for authentication of users on client connection requests when the security level is set to share or user. If the default configuration file is missing, any attempt to change Samba passwords will fail. To avoid this problem, one of the instances of Samba must use the default configuration file if the security level is set to share or user, or if the server is acting as the **smbpasswd** server for those systems with Samba security level set to server. The reason for this is that **smbpasswd** uses the default configuration file to obtain the location of the *smbpasswd* file. Because of this requirement only one location for the *smbpasswd* file can exist within the LifeKeeper cluster. The configuration files for all instances of Samba in the cluster must have the directive smb passwd file set to the same value. Additionally, the *smbpasswd* file must be kept in sync on all servers in the cluster.

The **smbpasswd** utility is also affected by the use of the bind interfaces only directive, which is required by the LifeKeeper Samba Recovery Kit. With the bind interfaces only directive set to Yes, a regular user changing his Samba password will attempt to connect to a **smbd** daemon process using the localhost address of 127.0.0.1. If that address has been added to the interfaces directive in the configuration file

used by the **smbd** daemon, and if **smbd** has connected to and is listening on that address, then the password change will be successful. If the daemon does not have that address in its configuration file interfaces directive, then the password change will fail. In a multiple instance environment, if the localhost is specified in more than one configuration file, only one instance will be able to start up and run. Using the –r netbios_name option to **smbpasswd** will work in place of adding the localhost address to the interfaces list (for example: **smbpasswd -r server1 print1…**).

> **Note**: As previously stated, non-protected Samba instances running on a LifeKeeper server with protected Samba instances must also have the bind interfaces only directive set to “Yes”.

Samba and User Authentication Considerations

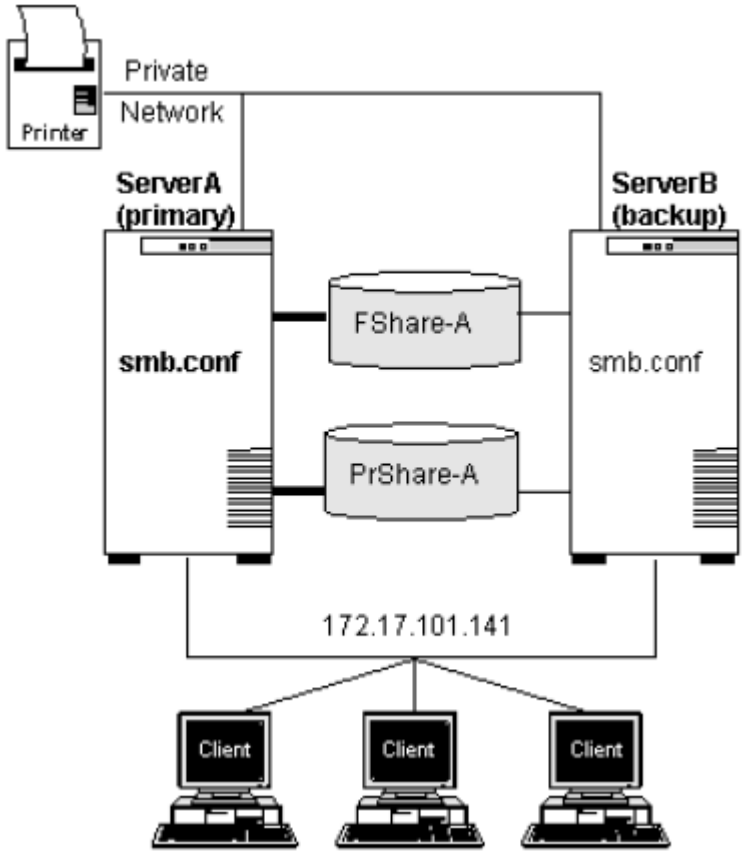
Samba supports several methods for user authentication via the security parameter (e.g. share, user, domain, …) which must be considered when protecting Samba via LifeKeeper to ensure data files such as /etc/samba/smbpasswd or /etc/samba/secrets.tdb are kept in sync on all servers in the cluster. So when using security methods such as user, you must ensure that the smbpasswd file is kept in sync on all servers in the cluster. Additionally, security methods such as domain require synchronization of the secrets.tdb file. A LifeKeeper active/active configuration with the secrets.tdb file requires the use of the private dir parameter to specify the location of the file. The value for this parameter must be unique for each LifeKeeper Samba instance.

6.15.4.7. Samba Configuration Examples

This section contains definitions and examples of typical Samba configurations. Each example includes the configuration file entries that apply to LifeKeeper.

Configuration 1: Active/Standby Configuration

In the Active/Standby configuration, ServerA is the primary LifeKeeper server. It exports the file and print shares that reside on a shared storage device. While ServerB may be handling other applications/ services, it acts only as a backup for the Samba resources in LifeKeeper's context.



Configuration Notes:

- The clients connect to the Samba servers using the NetBIOS name LKServerA over the protected IP address (172.17.101.141), which is defined by the interfaces directive of the configuration file.

- The configuration file smb.conf has been copied to ServerB upon extension of the Samba resource hierarchy. It contains the following directives:

  [global] netbios name = LKServerA bind interfaces only = yes

  lock directory = /var/lock/samba interfaces = 172.17.101.141 127.0.0.1

  log file = /var/log/sambaServA/log

[FShare-A]

path = /FShare-A

read only = no

public = yes

valid users =

printable = no

create mode = 0664

directory mode = 0775

[PRShare-A]

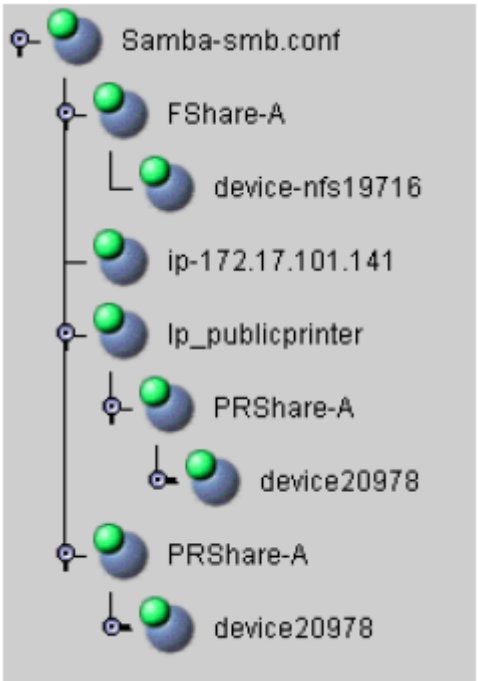
path = /PRShare-A

printer = publicprinter

printable = yes

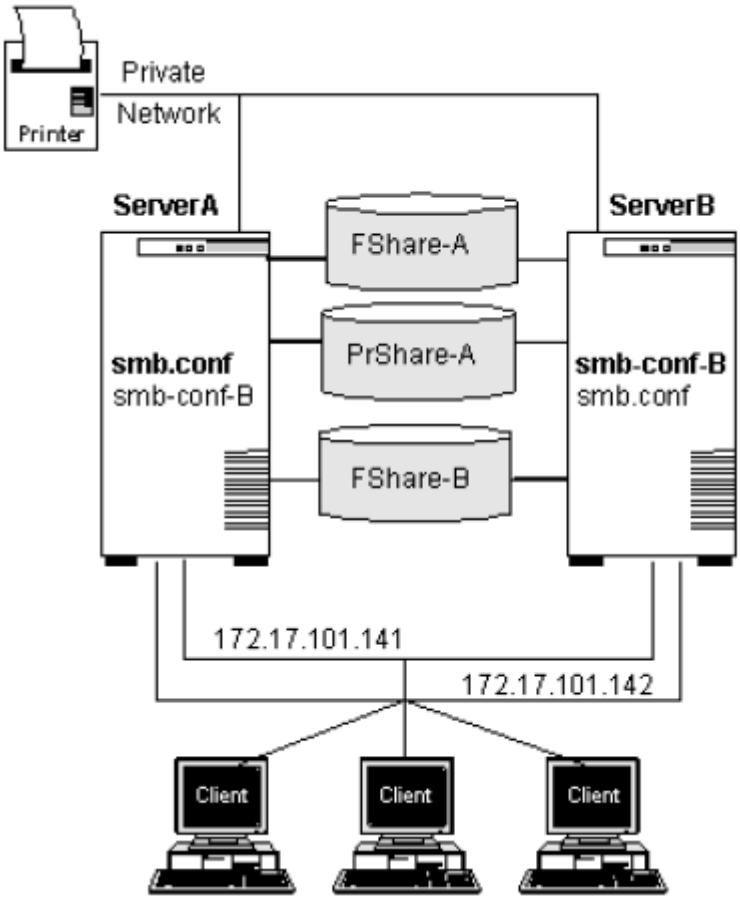
browseable = no

- The Samba resource hierarchy would look like the following:



Configuration 2: Active/Active Configuration

In the Active/Active configuration below, both ServerA and ServerB are primary LifeKeeper servers for Samba resources. Each server is also the backup server for the other.



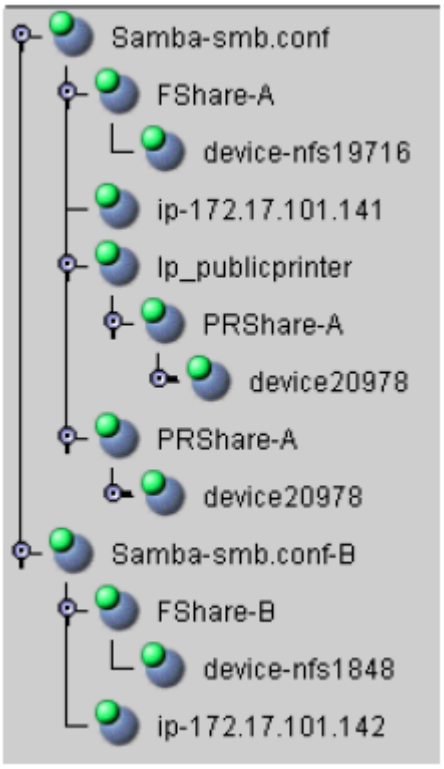
Configuration Notes:

- The clients connect to the Samba servers using the NetBIOS name LKServerA and LKServerB over the protected IP addresses (172.17.101.141 and 172.17.101.142 respectively), which are defined by the interfaces directive of the configuration files.
- The configuration file *smb.conf* was copied to ServerB upon extension of the Samba resource hierarchy. Likewise, the configuration file *smb.conf-B* was copied to ServerA upon extension of the Samba resource hierarchy.
- ServerA protects the file share */Fshare-A*; ServerB protects the file share */Fshare-B*.
- ServerA protects the print share */publicprinter*; ServerB does not protect a print share.
- The two configuration files contain the following directives:

smb.conf	smb-conf-B

[global] netbios name = LKServerA bind interfaces only = yes lock directory = /var/lock/sambaServA pid directory = /var/run/sambaServA interfaces = 172.17.101.141 127.0.0.1 log file = /var/log/sambaServA/log [FShare-A] path = /FShare-A read only = no public = yes valid users = printable = no create mode = 0664 directory mode = 0775 PRShare-A path = /PRShare-A printer = publicprinter printable = yes browseable = no	[global] netbios name = LKServerB bind interfaces only = yes lock directory = /var/lock/ pid directory = /var/run/sambaServB sambaServB interfaces = 172.17.101.142 log file = /var/log/sambaServB/log [FShare-B] path = /FShare-B read only = no public = yes valid users = printable = no create mode = 0664 directory mode = 0775

- The Samba resource hierarchies would look like the following:


Samba-smb.conf
FShare-A
device-nfs19716
ip-172.17.101.141
lp_publicprinter
PRShare-A
device20978
PRShare-A
device20978
Samba-smb.conf-B
FShare-B
device-nfs1848
ip-172.17.101.142

6.15.5. Samba Configuration Steps

This section provides steps that you should take to configure your Samba resources.

1. Plan your Samba configuration. This includes the following:

    - º NetBIOS name(s) to be used
    - º The interfaces that will be protected and allowed access to the shares
    - º The file systems to be used for the Samba shares and thus protected
    - º The location of the lock and log directory (or directories)

    Consideration should be given to the number of configuration files to be used and the type of configuration (Active/Standby vs. Active/Active). For example, if you have four Samba shares to protect, you could list all four shares in one configuration file, with the disadvantage that a failure of any one file system will cause the failover of the entire Samba hierarchy, including all four file shares. Alternatively, you could create four separate configuration files, each protecting one file share, which requires that four NetBIOS names be defined and managed.

2. Setup your Samba configuration file(s) based on the plan made in step 1. This includes the required directives in the [global] section as well as those for the file and print shares to be used. See Configuring the LifeKeeper for Linux Samba Recovery Kit for a discussion of the global and share directives required for LifeKeeper Samba hierarchies.

3. Create protected IP addresses under LifeKeeper, which will be used for client connections to the Samba server via the NetBIOS name. The protected IP address(es) should match the value(s) placed in the interface directive in the configuration file. (Refer to the SPS for Linux IP Recovery Kit Administration Guide for details on setting up IP resources.) Test the protected IP addresses by pinging them from all clients and other cluster servers. A protected IP resource for the local host (127.0.0.1) is not required.

4. Start the Samba daemons and test client connections.

    a. The commands to start the daemons are as follows:

    s/nmbd –D –s ConfigurationFile

    b. Use the Samba utility smbclient to test connections to the smbd daemon as follows. This should be done for each address defined in the interfaces directive.

    smbclient –L netbios_name –U% -I Protected_IP_Address

    c. Use **nmblookup** to test connection to the nmbd daemon process. This should be done for each broadcast address. Use the associated broadcast address for each address defined in

the interfaces directive. (The broadcast address can be obtained by running **ifconfig**).

nmblookup –B broadcast_address netbios_name

5. Stop the Samba daemons started in the previous step. This is accomplished via the **kill** command. Find the running daemon processes via the **ps** command and issue a **kill pid** which will cause them to exit.

6. Create protected file system(s) under LifeKeeper that will host the Samba file and print shares as defined in the above steps. (Refer to Creating a file system resource hierarchy in the SPS for Linux Technical Documentation for information on creating a File System resource hierarchy.) This step may be skipped since File System resources will be created automatically when creating a Print Services resource or Samba resource.

7. Create directories on the protected file systems for the shares should one file system be used for multiple Samba shares.

8. Create protected Print Services hierarchies under LifeKeeper, which will be used for client printing should any printer shares be defined in the configuration file.

9. Create the Samba resource hierarchy in LifeKeeper and extend it to at least one backup server (see the LifeKeeper Configuration Tasks section below). The extend script will copy the Samba configuration file from the template server to the same location on the target server.

10. On the primary server, test client connections to the shares that are protected by the Samba hierarchy which is in service. For instance, map the shared directory from a Windows client and ensure that it can access files on the share. Repeat the test for all servers in the cluster. You should also test your Samba resource by performing a manual switchover to a backup server. (See Testing Your Resource Hierarchy.)

11. Automatic startup must be disabled at the time of system boot because Samba daemon protected by LifeKeeper is controlled by LifeKeeper.

6.15.6. LifeKeeper Configuration Tasks for Samba

You can perform all LifeKeeper for Linux Samba Recovery Kit administrative tasks via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer, and monitor Samba resources.

The following tasks are available for configuring the LifeKeeper for Linux Samba Recovery Kit:

- Create a Resource Hierarchy – Creates a Samba resource hierarchy.
- Delete a Resource Hierarchy – Deletes a Samba resource hierarchy.
- Extend a Resource Hierarchy – Extends a Samba resource hierarchy from the primary server to the backup server.
- Unextend a Resource Hierarchy – Unextends (removes) a Samba resource hierarchy from a single server in the LifeKeeper cluster.
- Create Dependency – Creates a child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete Dependency – Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service – Activates a resource hierarchy.
- Out of Service – Deactivates a resource hierarchy.
- View / Edit Properties – View or edit the properties of a resource hierarchy.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You can also perform most of these tasks:

1. From the toolbar, right – click on a global resource in the left pane of the status display.
2. Right – click on a resource instance in the right pane of the status display.

Note: Using the right-click method allows you to avoid entering information that is required when using the **Edit** menu.

6.15.6.1. Creating a Samba Resource Hierarchy

After you have completed the necessary setup tasks, use the following steps to define the Samba resource hierarchy.

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the menu, select **Create Resource Hierarchy**.

   The *Create Resource Wizard* dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select Samba Share and click **Next**.

3. You will be prompted to enter the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter an error requiring you to correct previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

Field	Tips
Switchback Type	Choose either *intelligent* or *automatic* This dictates how the Samba instance will be switched back to this server when the server comes back up after a failover. The switchback type can be changed later from the General tab of the Resource Properties dialog box. **Note**:The switchback strategy should match that of the Print Server, IP or File System resource to be used by the Samba resource. If they do not match the Samba resource creation will attempt to reset them to match the setting selected for the Samba resource.
Server	Select the Server on which you want to create the hierarchy.
Location of Configuration File	Select the directory where the Samba configuration file is located.
Config File Name	Enter the name of the Samba configuration file to be used for this resource creation. The default is *smb.conf*. **Note**: LifeKeeper will read the selected configuration file, and if the file does not specify the required directives, LifeKeeper will generate an error message. It does minimal checking of the configuration file (to verify that shares exist, that they have a path directive, that a lock directory has been specified, and that the directory exists). Additional checking is done during the creation process.

Root Tag	Either select the default root tag offered by LifeKeeper, or enter a unique name for the resource instance on this server. The default is Samba- *configfilename*, where *configfilename* is the name of the associated configuration file. You may use letters, numbers and the following special characters: – _ . /

4. Click **Create**. The Create Resource Wizard will then create your Samba resource hierarchy. LifeKeeper will validate the data entered. If LifeKeeper detects a problem, an error message will appear in the information box.

5. An information box will appear indicating that you have successfully created a Samba resource hierarchy, and you must Extend that hierarchy to another server in your cluster in order to achieve failover protection. Click **Next**.

6. Click Continue. LifeKeeper will then launch the Pre-Extend Wizard. Refer to Step 2 under Extending a Resource Hierarchy (below) for details on how to extend your resource hierarchy to another server.

Note: See Failure Restoring Samba Hierarchy in the Samba Troubleshooting section for tips to follow in the case that the creation of the Samba hierarchy fails.

6.15.6.2. Extending Your Samba Resource Hierarchy

This operation can be started from the **Edit** menu, or initiated automatically upon completing the **Create Resource Hierarchy** option, in which case you should refer to Step 2 below.

1. On the **Edit** menu, select **Resource**, then Extend Resource Hierarchy. The Pre-Extend Wizard appears. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the LifeKeeper **Extend Resource Hierarchy** defaults and want to bypass the prompts for input/ confirmation, click **Accept Defaults**.

2. The *Pre-Extend Wizard* will prompt you to enter the following information.

Note: The first two fields appear only if you initiated the Extend from the **Edit** menu.

Field	Tips
Template Server	Enter the server where your Samba resource is currently in service.
Tag to Extend	Select the Samba resource you wish to extend.
Target Server	Enter or select the server you are extending to.
Switchback Type	Select either intelligent or automatic. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. **Note**: Remember that the switchback strategy must match that of the dependent resources to be used by the Samba resource.
Template Priority	Select or enter a priority for the template hierarchy. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (the number 1 indicates the highest priority). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note**:This selection will appear only for the initial extend of the hierarchy.
Target Priority	Either select or enter the priority of the hierarchy for the target server.
Root Tag	LifeKeeper will provide a default tag name for the new Samba resource instance on the target server. The default tag name is the same as the tag name for this resource on the template server. If you enter a new name, be sure it is unique on the target server. You may use letters, numbers and the following special characters: – _ . /

3. After receiving the message that the pre-extend checks were successful, click **Next**.

4. Depending upon the hierarchy being extended, LifeKeeper will display a series of information box showing the Resource Tags to be extended, which cannot be edited. Click **Extend**.

5. After receiving the message "Hierarchy extend operations completed" click **Next Server** to extend the hierarchy to another server, or click Finish if there are no other extend operations to perform.

6. After receiving the message "Hierarchy Verification Finished", click **Done**.

6.15.6.3. Unextending Your Samba Resource Hierarchy

To remove a resource hierarchy from a single server in the LifeKeeper cluster, do the following:

1. On the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.
2. Select the **Target Server** where you want to unextend the Samba resource. It cannot be the server where the Samba resource is currently in service. (This dialog box will not appear if you selected the Unextend task by right clicking on a resource instance in the right pane.) Click **Next**.
3. Select the Samba hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the Unextend task by right clicking on a resource instance in either pane)
4. An information box appears confirming the target server and the Samba resource hierarchy you have chosen to unextend. Click **Unextend**.
5. Another information box appears confirming that the Samba resource was unextended successfully. Click **Done** to exit the Unextend Resource Hierarchy menu selection.

6.15.6.4. Deleting a Samba Resource Hierarchy

It is important to understand what happens to dependencies and protected services when a Samba hierarchy is deleted.

- **Dependencies**: Before removing a resource hierarchy, you may wish to remove the dependencies. Dependent file systems will be removed unless they are used in another hierarchy. Dependent IP and Print Services resources will not be removed as long as the delete is done via the LifeKeeper GUI or the Samba delete script.

- **Protected Services**: If you take the Samba resource hierarchy out of service before deleting it, the Samba daemons will be stopped. If you delete a hierarchy while it is in service, the Samba daemons will continue running and offering services (without LifeKeeper protection) after the hierarchy is deleted.

To delete a resource hierarchy from **all** the servers in your LifeKeeper environment, complete the following steps:

1. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.

2. Select the **Target Server** where you will be deleting your Samba resource hierarchy and click **Next**. (This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in either pane.)

3. Select the **Hierarchy to Delete**. (This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in the left or right pane.) Click **Next**.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete**.

5. Another information box appears confirming that the Samba resource was deleted successfully.

6. Click **Done** to exit.

6.15.6.5. Testing Your Samba Resource Hierarchy

You can test your Samba resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Selecting **Edit**, then **Resource**, then **In Service**. For example, an In Service request executed on a backup server causes the application hierarchy to be taken out of service on the primary server and placed in service on the backup server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the *Out of Service* request, the application is taken out of service without bringing it in service on the other server.

6.15.7. Samba Hierarchy Administration

Once your Samba resource hierarchies are created, follow these guidelines for ongoing administration of your Samba resources.

6.15.7.1. Modifying the Samba Configuration File

When changes are required to a Samba configuration file that is used in a LifeKeeper Samba instance, perform these procedures on the server that is In Service, Protected (ISP). There are three types of configuration file changes:

- Those that do not directly impact the Samba hierarchy
- Those that directly impact the hierarchy but do not require a delete and recreation of hierarchy
- Those that directly impact the hierarchy and require a delete and recreation of the hierarchy

Modifications that do not directly impact the Samba Hierarchy

Any changes to configuration file directives not used by LifeKeeper fall into this category. (See Configuring the LifeKeeper for Linux Samba Recovery Kit for a list the directives used by the kit.) Example directives not used by LifeKeeper would include security, hosts allow, hosts deny and valid users to name a few. The procedures are as follows:

1. Take the Samba resource for the configuration file out of service. This step is required to stop the Samba daemons.
2. Make the necessary updates to the Samba configuration file.
3. Synchronize the configuration within the cluster. Use the utility synccfg to perform this task:

   LKROOT/lkadm/subsys/gen/samba/bin/synccfg –t TargetSys –c ConfigFile

   where *LKROOT* is the install location of LifeKeeper (*/opt/LifeKeeper* by default), *TargetSy* is the node to update and *ConfigFile* is the full path to the configuration file to copy.
4. Repeat the previous step for all servers in the hierarchy.
5. Bring the hierarchy back in service to restart the Samba daemons.

Modifications that directly impact the Samba Hierarchy

Any changes to configuration file directives used by LifeKeeper (see Configuring the LifeKeeper for Linux Samba Recovery Kit for a list), with the exception of the netbios name or the physical movement of the configuration file, fall into this category. Depending on the extent of the changes, it may be quicker and easier to proceed to the third category and just recreate the hierarchy. The typical types of changes expected in this category include the addition of new file and print shares, removal of file and print

shares or the addition or removal of IP interfaces.

1. Take the Samba resource for the configuration file out of service. This step is required to stop the Samba daemons.

2. Make the necessary updates to the Samba configuration file.

3. Make the necessary updates to the Samba hierarchy. This varies depending on the type of change made to the configuration file. For example:

    - º If an additional IP address has been added to the interfaces directive, then a new IP resource needs to be created, extended and then added as a dependent child to the Samba resource hierarchy. See Creating a Resource Dependency in the

SPS for Linux Technical Documentation for information on how to create dependencies.

    - º If a new file share has been added to the configuration file, then a File System resource may need to be created, extended and added as a dependent child to the Samba resource hierarchy. If the File System resource already exists as a child in the hierarchy (e.g. the path directive defined for the new share has the same file system mount point as another file or print share) then it does not need to be created and added as a dependent child.

    - º If a new print share is added, then File System and Print Services resources need to be created, extended and added as dependent children in the Samba hierarchy. If a print services resource does not exist that protects the printer as defined by the print share name or printer/ printer name directive, then one must be created. See file share above to determine if a file system resource needs to be added.

    - º If a file or print share is removed, or if an IP address is removed from the interfaces directives, delete the dependency in the Samba hierarchy and then delete the individual resource.

    - º If a print share name is changed, follow the delete of print share followed by the addition of new print share.

4. Synchronize the configuration within the cluster. Use the utility **synccfg** to perform this task:

    LKROOT/lkadm/subsys/gen/samba/bin/synccfg –t TargetSys –c ConfigFile

    where *LKROOT* is the install location of LifeKeeper *, _TargetSys* is the server to update and *ConfigFile* is the full path to the configuration file to copy.

5. Repeat the previous step for all servers in the hierarchy.

6. Bring the hierarchy back in service to restart the Samba daemons.

    Note: If you are making a number of changes that require numerous resource creations and dependency additions or deletions, you may wish to create all the new resources before you take the Samba hierarchy out of service so that downtime is minimized.

Modifications that directly impact the Samba Hierarchy, requiring a deletion and recreation of the Hierarchy

If the netbios name directive is changed or the physical location of the configuration file is changed, then you must:

1. Delete the hierarchy. (See Deleting a Resource Hierarchy for details.)
2. Change the NetBIOS name or move the configuration file.
3. Create a new Samba hierarchy and extend to all backup servers.

6.15.7.2. Maintaining the smvpasswd File

Samba provides four different authentication methods via the security directive. The share and user security settings both require access to the local *smbpasswd* file to determine if access will be granted. As noted in the section Running Multiple Instances of Samba there can only be one *smbpasswd* file, and this presents a potential administration problem in a LifeKeeper cluster. If share or user level security is selected, the file must be kept in sync on all servers so that authentication will succeed after a failover.

In a cluster with only one Samba hierarchy, the use of share or user level security can be accomplished by placing the *smbpasswd* file on a file share defined in the configuration file. The access to this share should be such that only administrators have access. In a multiple instances scenario, either server level or domain level security is suggested.

Note: If firewalls are in use, ensure that the firewall will allow connections to the **smbd** daemon, and that the **nmblookup** will work.

6.15.8. Samba Troubleshooting

Failure restoring Samba hierarchy

Failure of a Samba hierarchy restore can leave the daemon processes **smbd** and **nmbd** running. (The restore operation is initiated via the completion of a create, failover, or manual switchover, or via a local recovery caused by a connection failure to **smbd** and **nmbd**.) If the daemons are not stopped and restarted after the problem is corrected, and the restore is attempted again, the restore could fail again.

Suggested Action:

Correct the cause of the connection failure (for instance, an incorrect mask setup for the interfaces directive). Next, manually stop the **smbd** and/or **nmbd** daemons. Then bring the hierarchy in service, or re-create the hierarchy if the failure occurred during creation.

Stopping the daemons ensures that a re-read of the configuration file occurs during the restore.

6.15.8.1. Common Samba Error Messages

This section provides a list of messages that you may encounter with the use of the SPS Samba Recovery Kit. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Because the Samba Recovery Kit relies on other SPS components to drive the creation and extension of hierarchies, messages from these other components are also possible. In these cases, please refer to the **Message Catalog** (located on our Technical Documentation site under **Search for an Error Code**) which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

Common Error Messages

Error Number	Error Message	Description
109009	Error getting netbios name from the instance information field for tag "Samba-smb.conf" on server "ServerA".	Extracting the NetBIOS name from the instance info failed. Check that the info field contains the configuration file and NetBIOS name.
109015	The Samba utility testparm failed. Unable to parse Samba configuration file.	The Samba utility **testparm** that is used to parse the configuration file failed. Run **testparm** from the command line specifying the configuration file used for the hierarchy to determine the failure.
109019	Failed to initialize for reading of the Configuration File "/tmp/smb.ini.1234".	Attempts to read the generated output of the configuration file failed. The utility **testparm** generated a bad file.
109022	Error getting configuration name from the instance information field for tag "Samba- smb.conf" on server "ServerA".	Extracting the NetBIOS name from the instance info failed. Check that the info field contains the configuration file and NetBIOS name.
109030	Failure opening "/var/lock/samba/smbd.pid" on server "ServerA": "File Not Found"	The attempted open of the daemon process ID file failed for the listed reason. Correct the problem based on the listed error code.
109050	Open of the testparm output file failed.	The open of output file created by running testparm failed because the file does not exist or does not contain any data. Run **testparm** from the command line specifying the configuration file used for the hierarchy to determine the failure.

6.15.8.2. Hierarchy Creation

Error Number	Error Message	Description
109001	Usage: "valid_cf"CfgPath CfgName TemplateSys	The **valid_cf** script requires three arguments, the directory containing the configuration file, the name of the configuration file and the name of the template system on which to validate the configuration file. You must specify all three.
109002	Must specify an absolute path to the "smb.conf" configuration file.	You must specify the absolute path to the configuration file when running the scripts **choice_cf** and **valid_cf**.
109003	The file "smb.conf" does not exist in "/etc".	You must specify the correct path to the configuration file when running **valid_cf** to validate the select configuration file.
109004	The path "/export/fs" in "/etc/samba/smb.conf" does not reside on a shared file system.	The Samba configuration shares must contain a path directive that can be protected by LifeKeeper via a File System resource. Edit the configuration file and change the path to a file system that LifeKeeper can protect.
109005	Usage: "choices_cf" CfgFile	The **choice_cf** script requires the full path to an existing configuration file. Please specify the correct path.
109006	Samba Configuration file not specified.	No configuration file was specified for the creation of the Samba resource hierarchy.
109007	Cannot bring hierarchy "Samba-smb.conf" in service on server "ServerA". **Action**: After correcting the problem, try bringing the hierarchy in service manually.	The in service attempt at the end of creation failed. View the log file for possible reason for the failure.
109010	The Samba configuration file does not have the interfaces directive defined. This directive is required to create Samba File Share hierarchies.	The configuration file is missing the interfaces directive or the directive does not contain any IP addresses other than the localhost (127.0.0.1).
109011	The Samba configuration file does not have a correctly formatted interfaces directive. The interfaces directive must be in full dotted decimal IP address format with or without the mask parameter.	The interfaces directive must contain one or more IP addresses in the format of aaa.bbb.ccc.ddd or aaa.bbb.ccc.ddd/mask separated by a space.
109012	The Samba configuration file section "FileShare1" does not have a path directive defined. All Samba shares must have a path directive.	The specified share in the configuration file does not have a path directive or the directive does not contain a value.

109014	No IP resources defined on server "ServerA". **Action**: Create IP resources for the IP addresses defined in the Samba configuration file interfaces directive.	The specified server does not contain any LifeKeeper protected IP resources needed for the creation of the Samba resource hierarchy.
109016	The IP(s) "100.25.104.25,100.35.104.26" defined in the interfaces directive are not under LifeKeeper protection. **Action**: Create IP resources for the unprotected addresses defined in the Samba configuration file interfaces directive.	LifeKeeper protected IP resources must exist for all of the IP addresses listed in the interfaces directive. Those listed are not protected by LifeKeeper.
109017	Missing configuration file name.	No configuration file exists when attempting to run **testparm** during Samba resource creation.
109018	No Samba shares found in "/etc/samba/smb.conf".	The Samba configuration specified for the resource must contain at least one file or print share.
109020	Bad configuration file. No section information found in the file "/tmp/smb.ini.1234".	The Samba configuration specified for the resource must contain at least one file or print share.
109021	Creation of Samba hierarchy with tag "Samba-smb.conf" on server "ServerA" failed.	The create of the Samba hierarchy failed. Examine the other error messages to determine the cause of the failure.
109023	The file system resource "filesys1328" is not in-service on server "ServerA".	The File System resource needed as a dependent child in the Samba hierarchy is not in service on the template server. Bring the resource in service and retry the resource creation.
109024	Selected IP resource "ip-100.25.104.26" does not exist on server "ServerA". **Action**: Retry the operation.	The specified IP resource tag no longer exists and is needed for the creation of the Samba hierarchy. Recreate the IP resource and retry the resource creation.
109025	LifeKeeper was unable to create a dependency between the Samba hierarchy "Samba-smb.conf" and the IP resource "ip-100.25.104.26" on server "ServerA".	The dependency creation attempt between the Samba resource and the IP resource failed. Examine the other messages to determine the cause of the failure.
109026	The Samba configuration file does not have a netbios name directive defined.	All configuration files must contain a NetBIOS name. Add a NetBIOS name directive to the configuration file.

	Action: Add a netbios name directive in the global section of the configuration file.	
109029	The Samba configuration file "%s" directive defines a directory that does not exist. The %s can contain pid directory,lock dir, or lock directory.	All configuration files must contain an existing directory as specified by the directive. Add the missing directory.
109035	The Samba directive "bind interfaces only" must be set to "Yes". **Action**: Change "bind interfaces only" to "Yes" and recreate the hierarchy.	All configuration files must have the “bind interfaces only” directive set to Yes. Correctly set the directive to Yes.
109036	Selected Printer resource "lp-admin" does not exist on server "ServerA". **Action**: Retry the operation.	The specified Print Services resource tag no longer exists and is needed for the creation of the Samba hierarchy. Recreate the Print Services resource and retry the resource creation.
109037	LifeKeeper was unable to create a dependency between the Samba hierarchy "Samba-smb.conf" and the Printer resource "lp-admin" on server "ServerA".	The dependency creation attempt between the Samba resource and the Print Services resource failed. Examine the other messages to determine the cause of the failure.
109038	The Printers(s) "lpadmin" defined in the configuration file are not under LifeKeeper protection. **Action**: Create Printer instances for the unprotected printers defined in the Samba configuration file.	LifeKeeper protected Print Services resources must exist for all of the printers defined in the configuration file. Those listed are not LifeKeeper protected.
109041	The selected configuration file "/etc/samba/smb.conf" is in use by Samba resource "Samba-smb.conf". Or The selected netbios name "LKServer" is in use by Samba resource "Samba-smb.conf".	A Samba configuration file or netbios name can only be protected once in the cluster. Rename the configuration file or select a new NetBIOS name.

6.15.8.3. Hierarchy Extension

Error Number	Error Message	Description
109008	Replication of config file to target server "ServerB" failed. The "mkdir" of "/etc/samba/config_files" failed. Or Replication of config file to target server "ServerB" failed. The "/etc/samba/config_files" failed.	The attempted copy of the Samba configuration file on the extend failed. Either the **mkdir** or remote copy failed.
109042	WARNING: The configuration file "/etc/samba/smb.conf" currently exists on server "ServerB" and will be overwritten if this resource is extended.	The configuration file used for the resource hierarchy already exists on the backup server and will be overwritten. Cancel the extension to abort the overwriting of the file.

6.15.8.4. Restore

Error Number	Error Message	Description
109027	Failed start of smbd as daemon process.	The attempt to start **smbd** as a daemon failed. Check the Samba log files for additional information.
109028	Failed start of nmbd as daemon process.	The attempt to start **nmbd** as a daemon failed. Check the Samba log files for additional information.

6.15.8.5. Remove

Error Number	Error Message	Description
109033	Failed start of smbd as daemon process. Attempting to stop via SIGKILL.	The normal termination of the **smbd** daemon process failed so the daemon will be forcibly terminated.
109034	Failed to stop nmbd daemon process. Attempting to stop via SIGKILL.	The normal termination of the **nmbd** daemon process failed so the daemon will be forcibly terminated.

6.15.8.6. Resource Monitoring

Error Number	Error Message	Description
109031	Connection attempt to smbd daemon on address "100.25.104.26" failed.	The health check of the **smbd** daemon process failed when a connection attempt on the listed IP address failed. A local recovery will be attempted. If the local recovery fails, the hierarchy will be switch over to the backup server.
109032	Connection attempt to **nmbd** daemon failed for broadcast address "100.25.107.255" using netbios name "FILESERV1".	The health check of the **nmbd** daemon process failed when a connection attempt on the listed broadcast address failed. A local recovery will be attempted. If the local recovery fails, the hierarchy will be switched over to the backup server.
109039	No dependent IP resources were found for tag "Samba-smb.conf" on server "ServerA".	No IP children were found for the Samba hierarchy when attempting to ascertain the health of the **nmbd** daemon. Examine the logs to determine the failure.
109040	Unable to determine IP address and/or mask for IP resource "ip-100.25.104.25".	Extracting the IP address and Mask from the info for the IP resource failed. Check that the info field contains the IP address and mask.

6.15.8.7. Configuration File Synchronization Utility

Error Number	Error Message	Description
109013	The hierarchy for the specified configuration file does not have an equivalency with the target system. Select another target system or configuration file.	The -t TargetSys argument to the **synccfg** utility specified a system that does not contain an equivalency for the Samba hierarchy on the local system.
109044	Usage: synccfg -t TargetSys -c ConfigFile	The **synccfg** utility requires two arguments,-t TargetSys where the configuration file will be copied and –c ConfigFile for the name of the configuration to copy. You must specify both arguments.
109045	Specified Configuration file does not exist on this server.	The –c ConfigFile argument to the **synccfg** utility specified a configuration that does not exist on this system.
109046	The target system specified for updating is the same as the template system. Select a new target system.	The –t TargetSys argument to the **synccfg** utility specified the local server as the target of the copy. The target server must not be the local system.
109047	The specified configuration file is not used in any Samba instances. Select another configuration file.	The –c ConfigFile argument to the **synccfg** utility specified a configuration file that is not used in any LifeKeeper Samba instance. Only configuration file protected by LikeKeeper can be copied with this utility.
109048	The syncronization of the Samba configuration file failed with a status of "%s".	The attempted synchronization of the configuration files between the local system and the target system failed. The status provides the reason for the failure.
109049	The hierarchy for the specified configuration file is ISP on the target node. Select another target system or configuration file.	The -t TargetSys argument to the **synccfg** utility specified a system where the Samba hierarchy is currently ISP. The target system must not be ISP.

6.16. SAP Recovery Kit Administration Guide

The SIOS Protection Suite for Linux SAP Recovery Kit provides a mechanism to recover SAP NetWeaver from a failed primary server onto a backup server in a LifeKeeper environment. The SAP Recovery Kit works in conjunction with other SIOS Protection Suite Recovery Kits (the IP Recovery Kit, NFS Server Recovery Kit, NAS Recovery Kit and a database recovery kit, e.g. Oracle Recovery Kit) to provide comprehensive failover protection.

This documentation provides information critical for configuring and administering your SAP resources. You should follow the configuration instructions carefully to ensure a successful SPS implementation. You should also refer to the documentation for the related recovery kits.

SAP Recovery Kit Overview

Documentation

The following is a list of LifeKeeper related information available from SIOS Technology Corp.:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation (also available from the Help menu within the LifeKeeper GUI)
- SIOS Technology Corp. Documentation and Support
- Abbreviations and Definitions – Contains a list of abbreviations and terms that are used throughout this documentation along with their meaning.
- LifeKeeper/SAP Icons – Contains a list of icons being used and their meanings.

Reference Documents

The following are documents associated with SAP that are referenced throughout this documentation:

- *SAP R/3 in Switchover Environments* (SAP document 50020596)
- *R/3 Installation on UNIX:* (Database specific)
- *SAP Web Application Server in Switchover Environments*
- *Component Installation Guide SAP Web Application Server* (Database Specific)
- *SAP Notes 7316, 14838, 201144, 27517, 31238, 34998 and 63748*

6.16.1. SAP Abbreviations and Definitions

The following abbreviations are used throughout this documentation:

Abbreviation	Meaning
AS	SAP Application Server. Although AS typically refers to any application server, within the context of this document, it is intended to mean a non-CI, redundant application server. Thus, the application server is not required for protection by LifeKeeper.
ASCS	ABAP SAP Central Services Instance. This is the SAP instance that contains the Message and Enqueue services for the NetWeaver ABAP environment. This instance is a single point of failure and must be protected by LifeKeeper.
(ASCS)	The backup ABAP SAP Central Services Instance server. This is the server that hosts the ASCS when the primary ASCS server fails.
DB	The SAP Database instance. This database may be Oracle or any other database supported by SAP. This instance is a single point of failure and must be protected by LifeKeeper. Note that the CI and DB may be located on the same server or different servers. DB is also used to denote the Primary DB Server.
(DB)	The backup Database server. This is the server that hosts the DB when the primary DB server fails. Note that a single server might be a backup for both the Database and Central Instance.
ENSAv1	Standalone Enqueue Server Version 1. This is the version of the enqueue server available in SAP kernel versions prior to 7.51.
ENSAv2	Standalone Enqueue Server Version 2. This version of the enqueue server is available in SAP kernel versions 7.51 and later.
ERS	Enqueue Replication Server.
ERSv1	Enqueue Replication Server Version 1. This is the version of the enqueue replication server available in SAP kernel versions prior to 7.51.
ERSv2	Enqueue Replication Server Version 2. This version of the enqueue replication server is available in SAP kernel versions 7.51 and later.
HA	Highly Available; High Availability.
ID or **<ID>**	Two digit numerical identifier for an SAP instance.
<INST>	Directory for an SAP instance whose name is derived from the services included in the instance and the instance number, for example a CI <INST> might be *DVEBMGS00*.
PAS	Primary Application Server Instance.
SAP Instance	A group of processes that are started and stopped at the same time.
SAP System	A group of SAP Instances.
<sapmnt>	SAP home directory which is */sapmnt* by default but may be changed by the user during installation.
SCS	SAP Central Services Instance. This is the SAP instance that contains the Message and Enqueue services for the NetWeaver Java environment. This instance is a single point of failure and must be protected by LifeKeeper.
(SCS)	The backup SAP Central Services Instance server. This is the server that hosts the SCS when the primary SCS server fails.

SID or **<SID>**	System ID.
sid or **<sid>**	Lower case version of SID.
SPOF	Single Point of Failure.

6.16.2. LifeKeeper – SAP Icons

The following icons are significant on how to interpret the status of SAP resources in a LifeKeeper environment. These icons will show up in the LifeKeeper UI.

	**Active** – Resource is active and in service (Normal state).
	**Standby** – Resource is on the backup node and is ready to take over if the primary resource fails (Normal state).
	**Failed** – Resource has failed; you can try to put the resource in service (right-click on the resource and scroll down to “In Service” and enter). If the resource fails again, then recovery has failed (Failure state).
	**Attention needed** – SAP resource has failed or is in a caution state. If it has failed and automatic recovery is enabled (Protection Level Full or Standard), then LifeKeeper will try to automatically recover the resource. Right-click on the SAP resource and choose **Properties**. This will show which resource has a caution state. An SAP state of **Yellow** may be normal, but it signifies that SAP resources are running slow or have performance bottlenecks.
	**Update protection level** – Action icon to allow changing the SAP resource protection level.
	**Update recovery level** – Action icon to allow changing the SAP resource recovery level.
	**Handle warnings** – Action icon to set failure on warnings.
	Action icon to select different actions, such as **start**, **stop**, **migrate** or **setting maintenance mode**.

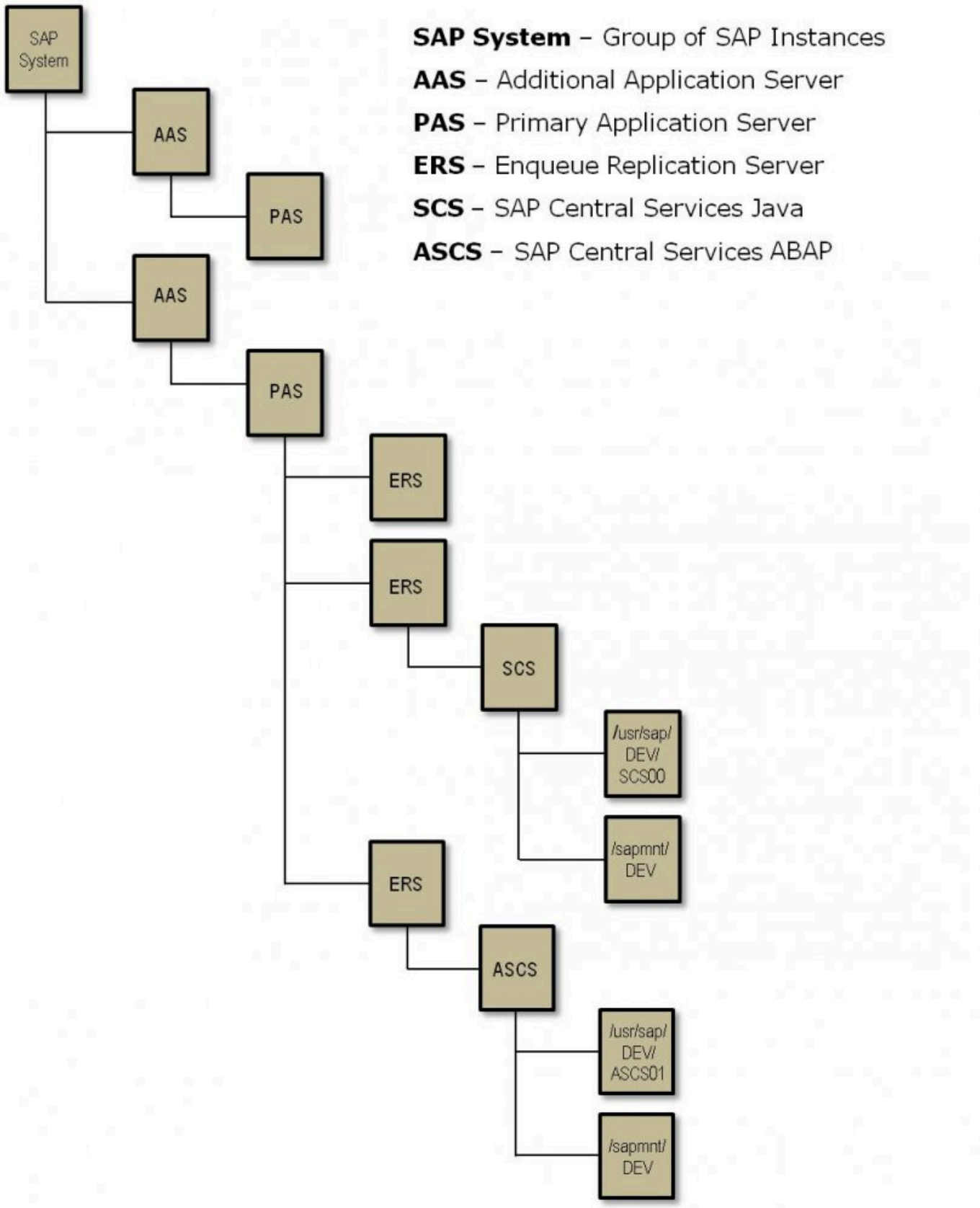
6.16.3. SAP Recovery Kit Overview

There are some services in the SAP NetWeaver framework that cannot be replicated. They cannot exist more than once for the same SAP system, therefore, they are single points of failure. The LifeKeeper SAP Recovery Kit provides protection for these single points of failure with standard LifeKeeper functionality. In addition, the kit provides the ability to protect, at various levels, the additional pieces of the SAP infrastructure. The protection of each infrastructure component will be represented in a single resource within the hierarchy.

The SAP Recovery Kit provides monitoring and switchover for different SAP instances; the SAP Primary Application Server (PAS) Instance, the ABAP SAP Central Service (ASCS) Instance and the SAP Central Services (SCS) Instance (the Central Service Instances protect the enqueue and message servers). The SAP Recovery Kit works in conjunction with the appropriate database recovery kit to protect the database, and with the Network File System (NFS) Server Recovery Kit to protect the NFS mounts. The IP Recovery Kit is also used to provide a virtual IP address that can be moved between network cards in the cluster as needed. The Network Attached Storage (NAS) Recovery Kit can be used to protect the local NFS mounts. The various recovery kits are used to build the SAP resource hierarchy which provides protection for all of the components of the application environment.

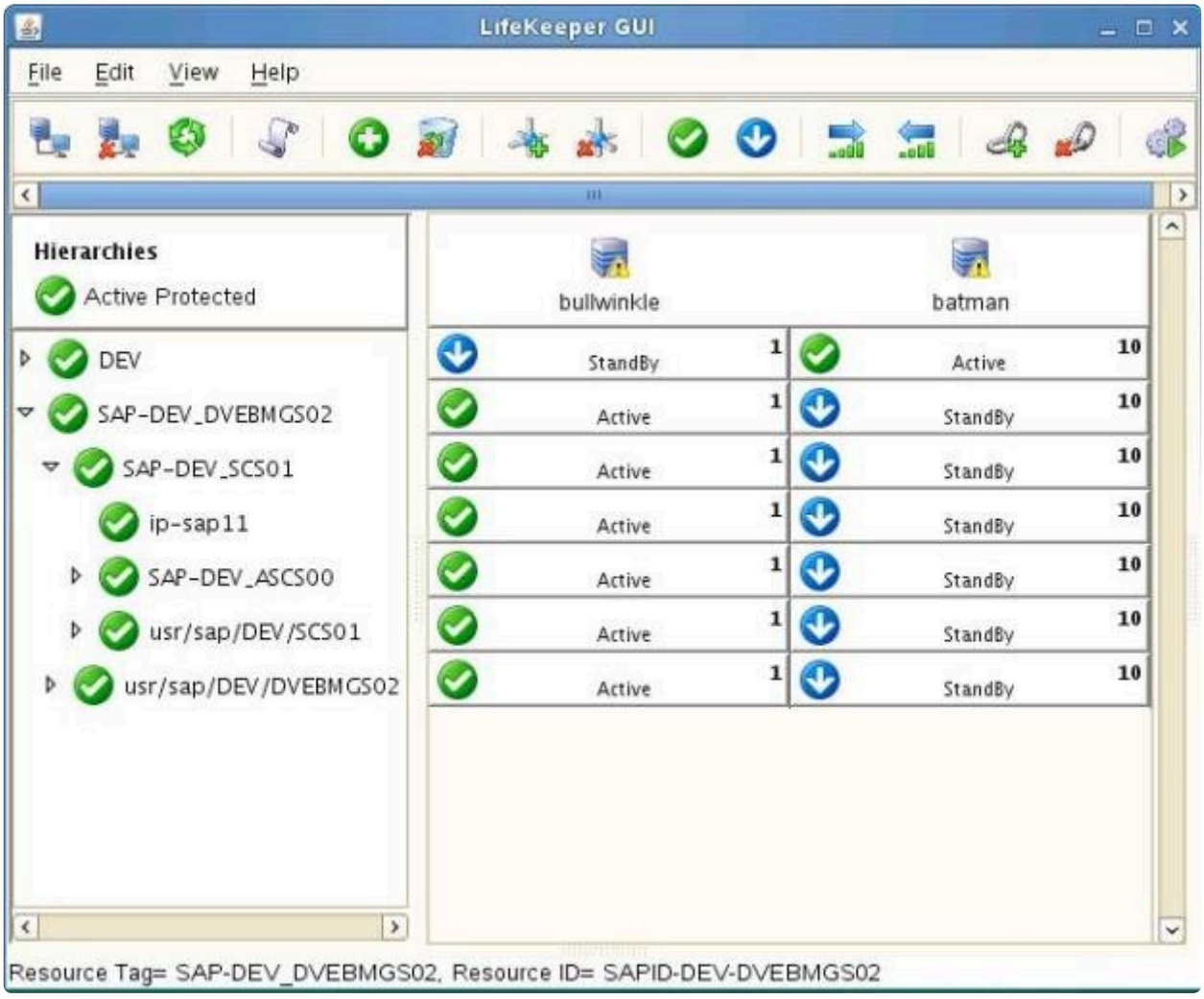
Each recovery kit monitors the health of the application under protection and is able to stop and restart the application both locally and on another cluster server.

Map of SAP System Hierarchy



A typical SAP resource hierarchy as it appears in the LifeKeeper GUI is shown below.

SAP Resource Hierarchy



Note: The directory */usr/sap/trans* is optional in SAP environments. The directory does not exist in the SAP NetWeaver Java only environments.

Note: An ERS resource created in SPS-L v9.4.0 or later will operate in its own independent hierarchy. See ERS Resource Types in LifeKeeper for more details.

6.16.4. SIOS Protection Suite for SAP Solution Page

SAP High Availability Interface 7.73

Package	Version	OS/Application Version Support
LifeKeeper Core	9.5.0	Red Hat Enterprise Linux 6 (Up to 6.10) Red Hat Enterprise Linux 7 (Up to 7.6) Red Hat Enterprise Linux 8 SUSE LINUX Enterprise Server (SLES) 11 (Up to SP4) SUSE LINUX Enterprise Server (SLES) 12 (SP1 to SP4) SUSE LINUX Enterprise Server (SLES) 15 (Up to SP1) CentOS 6.0 to 6.10 CentOS 7.0 to 7.6 Oracle Linux 6.3 to 6.10(including UEK R3) Oracle Linux 7.0 to 7.6(including UEK R4)
LifeKeeper SAP Recovery Kit	9.5.0	SAP NetWeaver 7.3 including Enhancement Package 1 SAP NetWeaver 7.4 SAP NetWeaver 7.5 SAP NetWeaver AS for ABAP 7.51 innovation package SAP NetWeaver AS for ABAP 7.52 innovation package SAP S/4HANA 1809 Platform SAP S/4HANA 1909 Platform
LifeKeeper NFS Server Recovery Kit	9.5.0	NFS exported file systems on Linux distributions with a kernel version of 2.6 or later
LifeKeeper Supported Databases for SAP	9.5.0	Oracle, SAP ASE (Sybase), SAP HANA, SAP DB (MaxDB), IBM DB2
SAP HA Interface Connector	7.73	SAP High Availability Interface 7.73

✻ Beginning in v9.5.0 SIOS has released the new SAP HANA Application Recovery Kit. SIOS will continue to support the SAP HANA gen/app based Recovery Kit with the 9.4.x releases until **March 31, 2022**. If you are using SIOS Protection Suite for Linux v9.5 or later you must use the new (built-in) SAP HANA Application Recovery Kit.

! The existing SAP HANA gen/app based Recovery Kit is **not** supported with v9.5.0. Users who wish to upgrade to the SIOS Protection Suite for Linux v9.5.0 **must** convert their existing SAP HANA gen/app based Recovery Kit to the new SAP HANA Recovery Kit. Refer to Upgrading from the SAP HANA Gen/App to the SAP HANA Recovery Kit for details.

✻ **NOTE:** Operating system versions built for enhanced SAP support (such as Red Hat Enterprise Linux for SAP Business Applications, Red Hat Enterprise Linux for SAP Solutions, and SUSE Linux Enterprise Server for SAP Applications) are also supported as long as the running Linux kernel version is the same as one of the supported OS

versions listed above.

6.16.5. SAP Hardware and Software Requirements

Before installing and configuring the SPS SAP Recovery Kit, be sure that your configuration meets the following requirements:

- **Servers**. This recovery kit requires two or more computers configured in accordance with the requirements described in the SPS for Linux Technical Documentation and the SPS for Linux Release Notes.

- **Shared Storage**. SAP Primary Application Server (PAS) Instance, ABAP SAP Central Service (ASCS) Instance, SAP Central Services (SCS) Instance and program files must reside on shared disk(s) in an SPS environment. The file system for the ERS instance may also reside on a shared disk in the case of an ERSv2 configuration.

- **IP Network Interface**. Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.

  Note: Even though each server requires only a single network interface, multiple interfaces should be used for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation and so forth. (See IP Local Recovery and Configuration Considerations for additional information.)

- **Operating System**. Linux operating system. (See the SPS for Linux Release Notes for a list of supported distributions and kernel versions.)

- **TCP/IP software**. Each server requires the TCP/IP software.

- **SAP Software**. Each server must have the SAP software installed and configured before configuring SPS and the SPS SAP Recovery Kit. The same version should be installed on each server. Consult the SPS for Linux Release Notes or your sales representative for the latest release compatibility and ordering information.

- **SPS software**. You must install the same version of SPS software and any patches on each server. Please refer to the SPS for Linux Release Notes for specific LifeKeeper requirements.

- **SPS IP Recovery Kit**. This recovery kit is required if remote clients will be accessing the SAP PAS, ASCS or SCS instance. You must use the same version of this recovery kit on each server.

- **SPS for Linux NFS Server Recovery Kit**. This recovery kit is required for most configurations. You must use the same version of this recovery kit on each server.

- **SPS for Linux Network Attached Storage (NAS) Recovery Kit**. This recovery kit is required for some configurations. You must use the same version of this recovery kit on each server.

- **SPS for Linux Database Recovery Kit**. The SPS recovery kit for the database being used with SAP must be installed on each database server. Please refer to the SPS for Linux Release Notes for information on supported databases. A LifeKeeper database hierarchy must be created for the SAP PAS, ASCS or SCS Instance prior to configuring SAP.

Important Notes:

- If running an SAP version prior to Version 7.3, please consult your SAP documentation and notes on how to download and install **SAPHOSTAGENT** (see Important Note in the Plan Your Configuration topic).

- Refer to the SPS for Linux Installation Guide for instructions on how to install or remove the Core software and the SAP Recovery Kit.

- The installation steps should be performed in the order recommended. The SAP installation will fail if LifeKeeper is installed first.

- For details on configuring each of the required SPS Recovery Kits, you should refer to the documentation for each kit (IP, NFS Server, NAS, and Database Recovery Kits).

- Please refer to SAP installation documentation for further installation requirements, such as swap space and memory requirements.

6.16.6. SAP Configuration Considerations

This section contains information to consider before starting to configure SAP and contains examples of typical SAP configurations. It also includes a step-by-step process for configuring and protecting SAP with LifeKeeper.

For instructions on installing SAP on Linux distributions supported by LifeKeeper using the 2.4 or 2.6 kernel, please see the database-specific SAP installation guide available from SAP.

Also, refer to your SPS for Linux Technical Documentation for instructions on configuring your SPS Core resources (for example, file system resources).

Supported Configurations

There are many possible configurations of database and application servers in an SAP Highly Available (HA) environment. The specific steps involved in setting up SAP for LifeKeeper protection are different for each configuration, so it is important to recognize the configuration that most closely fits your requirements. Some supported configuration examples are:

ABAP+Java Configuration (ASCS and SCS) **Note:** An ERS resource created in SPS-L 9.4.0 or later will operate in its own independent hierarchy. See ERS Resource Types in LifeKeeper for more details.

ABAP Only Configuration (ASCS) **Note:** An ERS resource created in SPS-L 9.4.0 or later will operate in its own independent hierarchy. See ERS Resource Types in LifeKeeper for more details.

Java Only Configuration (SCS) **Note:** An ERS resource created in SPS-L 9.4.0 or later will operate in its own independent hierarchy. See ERS Resource Types in LifeKeeper for more details.

The configurations pictured in the above examples consist of two servers hosting the Central Services Instance(s) with an ERS Instance, Database Instance, Primary Application Server Instance and zero or more additional redundant Application Server Instances (AS). Although it is possible to configure SAP with no redundant Application Servers, this would require users to log in to the ASCS Instance or SCS Instance which is not recommended by SAP. The ASCS Instance, SCS Instance and Database servers have access to shared file storage for database and application files.

While Central Services do not use a lot of resources and can be switched over very fast, Databases have a significant impact on switchover speeds. For this reason, it is recommended that the Database Instances and Central Services Instances (ASCS and SCS) be protected through two distinct LifeKeeper hierarchies. They can be run on separate servers or on the same server.

Configuration Notes

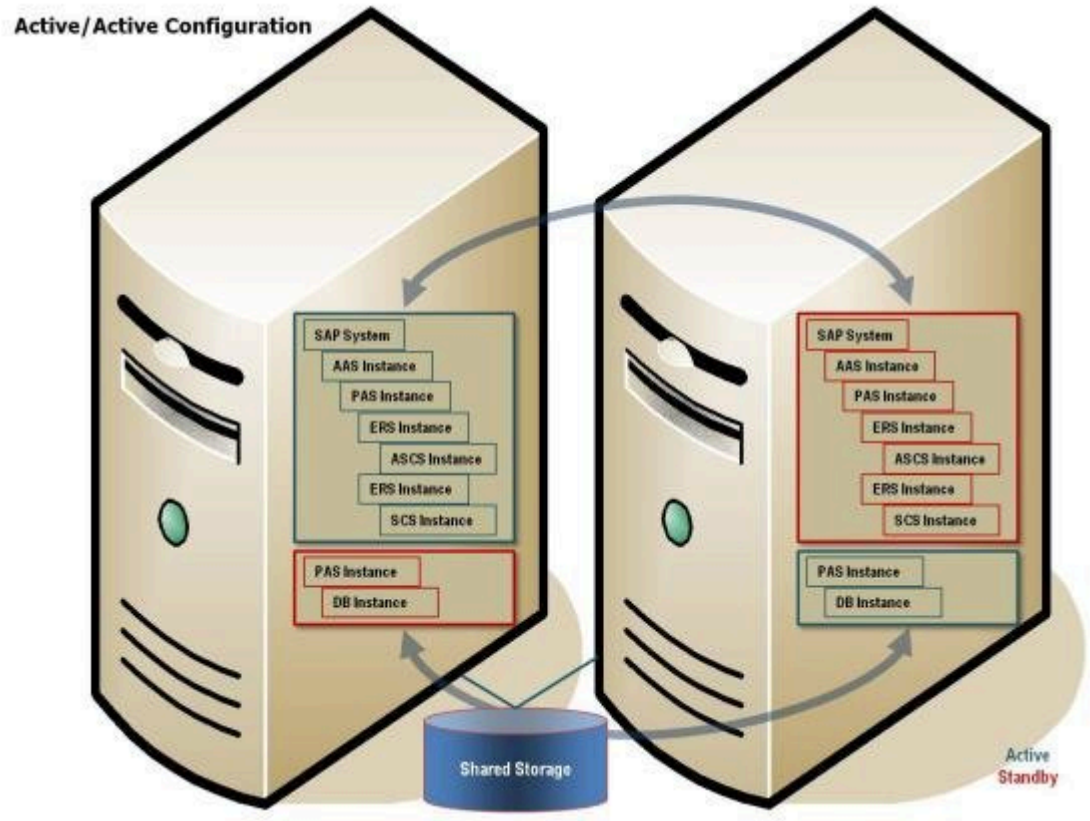
The following are technical notes related to configuring SAP to run in an HA environment. Please see subsequent topics for step-by-step instructions on protecting SAP with LifeKeeper.

6.16.6.1. ABAP+Java Configuration (ASCS and SCS)

The ABAP+Java Configuration comprises the installation of:

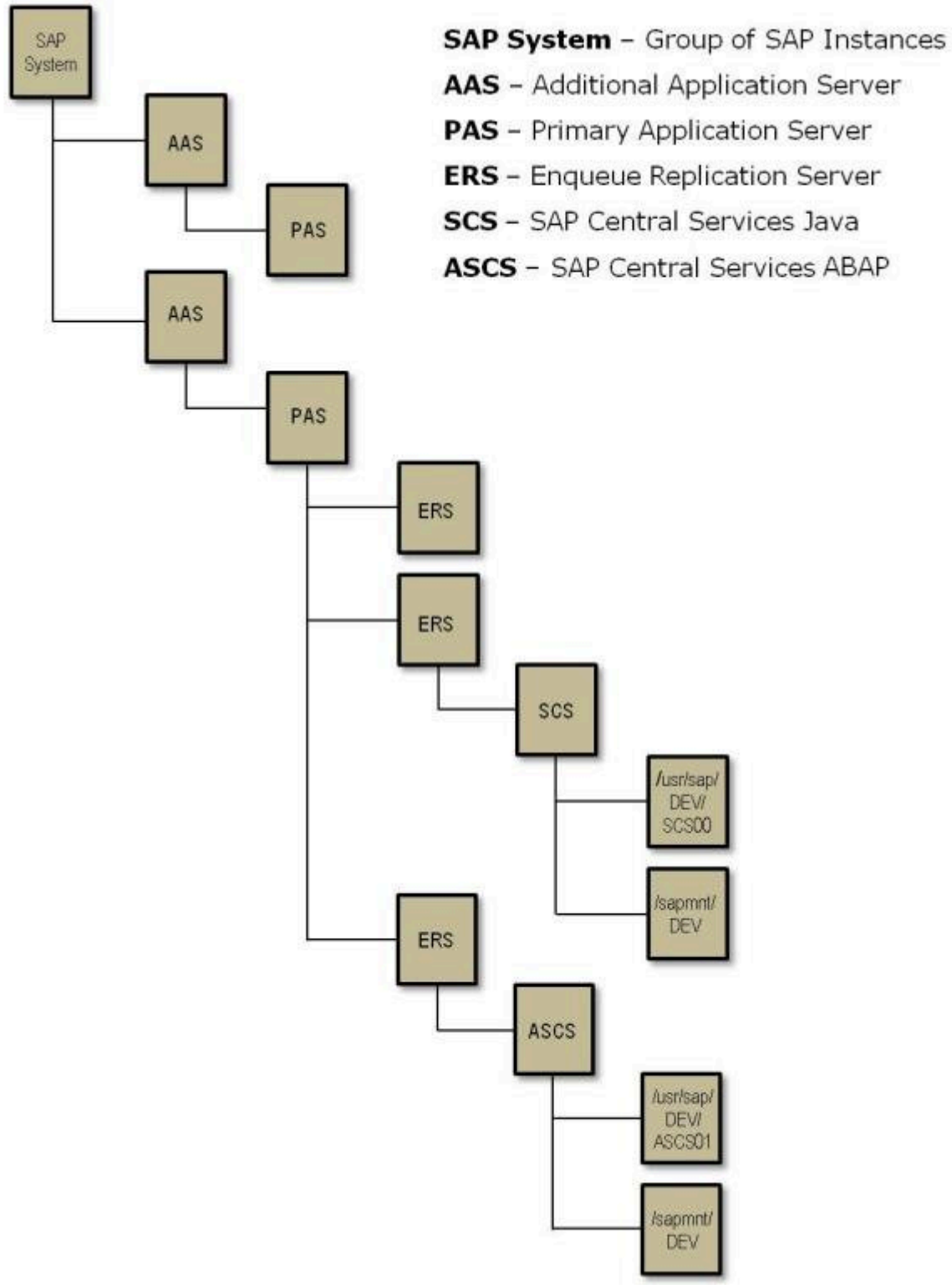
- Central Services Instance for ABAP (ASCS Instance)
- Enqueue Replication Server Instance (ERS instance) for the ASCS Instance (optional) (Both the ASCS Instance and the SCS Instance must each have their own ERS instance)
- Central Services Instance for Java (SCS Instance)
- Enqueue Replication Server Instance (ERS Instance) for the SCS Instance (optional)
- Database Instance (DB Instance) – The ABAP stack and the Java stack use their own database schema in the same database
- Primary Application Server Instance (PAS)
- Additional Application Server Instances (AAS) – It is recommended that Additional Application Server Instances (AAS) be installed on hosts different from the Primary Application Server Instance (PAS) Host

Switchover Cluster for an SAP Dual-stack (ABAP+Java) System



In the above example, ASCS and SCS are in a separate LifeKeeper hierarchy from the Database and these Central Services Instances are active on a separate server from the Database.

Example SAP Hierarchy



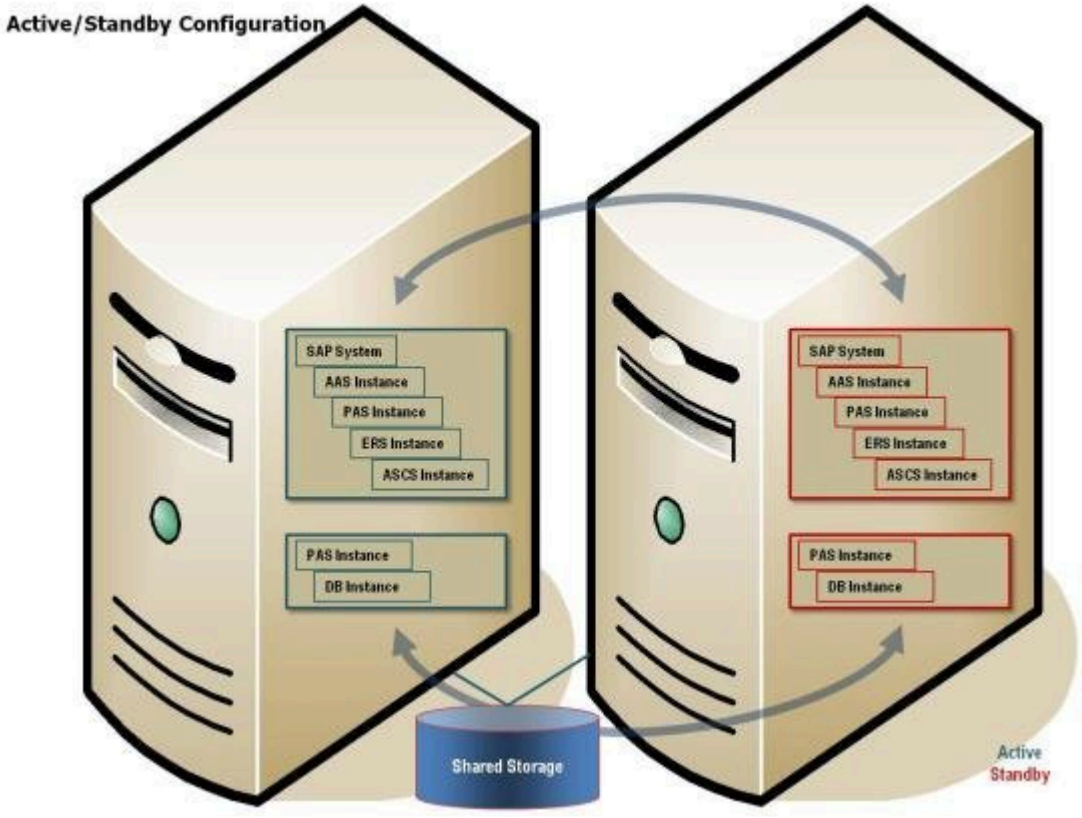
Note: An ERS resource created in SPS-L v9.4.0 or later will operate in its own independent hierarchy. See ERS Resource Types in LifeKeeper for more details.

6.16.6.2. ABAP SCS (ASCS)

The ABAP Only Configuration comprises the installation of:

- Central Services Instance for ABAP (ASCS Instance)
- Enqueue Replication Server Instance (ERS instance) for the ASCS Instance (optional)
- Database Instance (DB Instance)
- Primary Application Server Instance (PAS)
- Additional Application Server Instances (AAS) – It is recommended that Additional Application Server Instances (AAS) be installed on hosts different from the Primary Application Server Instance (PAS) Host

Switchover Cluster for an SAP ABAP Only (ASCS) System



In the above example, ASCS is in a separate LifeKeeper hierarchy from the Database. Although it is active on the same server as the Database, they can fail over separately.

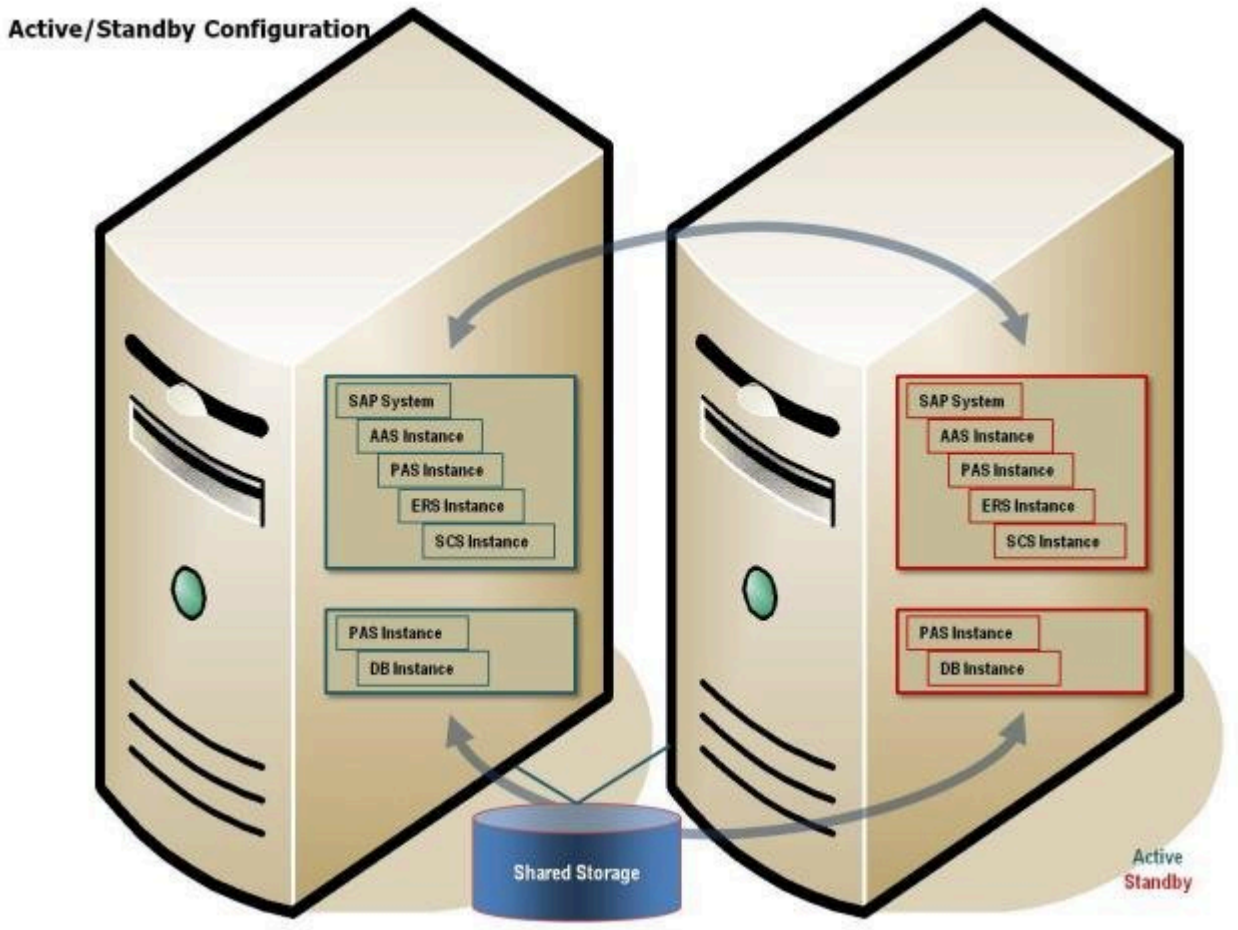
Note: An ERS resource created in SPS-L v9.4.0 or later will operate in its own independent hierarchy. See ERS Resource Types in LifeKeeper for more details.

6.16.6.3. Java Only Configuration (SCS)

The Java Only Configuration comprises the installation of:

- Central Services Instance for Java (SCS Instance)
- Enqueue Replication Server Instance (ERS Instance) for the SCS Instance (optional)
- Database Instance (DB Instance)
- Primary Application Server Instance (PAS)
- Additional Application Server Instances (AAS) – It is recommended that Additional Application Server Instances (AAS) be installed on hosts different from the Primary Application Server Instance (PAS) Host

Switchover Cluster for a Java Only System (SCS)



In the above example, SCS is in a separate LifeKeeper hierarchy from the Database. Although it is active on the same server as the Database, they can fail over separately.

Note: An ERS resource created in SPS-L v9.4.0 or later will operate in its own independent hierarchy. See ERS Resource Types in LifeKeeper for more details.

6.16.6.4. SAP Directory Structure

The directory structure for the database will be different for each database management system that is used with the SAP system. Please consult the SAP installation guide specific to the database management system for details on the directory structure for the database. All database files must be located on shared disks to be protected by the LifeKeeper Recovery Kit for the database. Consult the database specific Recovery Kit Documentation for additional information on protecting the database.

See the Directory Structure Diagram below for a graphical depiction of the SAP directories described in this section.

The following types of directories are created during installation:

Physically shared directories (reside on global host and shared by NFS):

> */<sapmnt>/<SAPSID>* – Software and data for one SAP system (should be mounted for all hosts belonging to the same SAP system)
>
> */usr/sap/trans* – Global transport directory (has to have an export point)

Logically shared directories that are bound to a node such as */usr/sap* with the following local directories (reside on the local host with symbolic links to the global host):

> */usr/sap/<SAPSID>*
>
> */usr/sap/<SAPSID>/SYS*
>
> */usr/sap/hostctrl*

Local directories (reside on the local host and shared) that contain the SAP instances such as:

> */usr/sap/<**SAPSID**>/DVEBMGS<**No.**>* — Primary application server instance directory
>
> */usr/sap/<**SAPSID**>/D<**No.**>* — Additional application server instance directory
>
> */usr/sap/<**SAPSID**>/ASCS<**No.**>* — ABAP central services instance (ASCS) directory
>
> */usr/sap/<**SAPSID**>/SCS<**No.**>* — Java central services instance (SCS) directory
>
> */usr/sap/<**SAPSID**>/ERS<**No.**>* — Enqueue replication server instance (ERS) directory for the ASCS and SCS

The SAP directories: */sapmnt/<**SAPSID**>* and */usr/sap/trans* are mounted from NFS; however, SAP instance directories (*/usr/sap/<**SAPSID**>/<**INSTTYPE**><**No.**>*) should always be mounted on the cluster node currently running the instance. Do not mount such directories with NFS. The required directory structure depends on the chosen configuration. There are several issues that dictate the required

directory structure.

NFS Mount Points and Inodes

LifeKeeper maintains NFS share information using inodes; therefore, every NFS share is required to have a unique inode. Since every file system root directory has the same inode, NFS shares must be at least one directory level down from root in order to be protected by LifeKeeper. For example, referring to the information above, if the */usr/sap/trans* directory is NFS shared on the SAP server, the */trans* directory is created on the shared storage device which would require mounting the shared storage device as */usr/sap*. It is not necessarily desirable, however, to place all files under */usr/sap* on shared storage which would be required with this arrangement. To circumvent this problem, it is recommended that you create an */exports* directory tree for mounting all shared file systems containing directories that are NFS shared and then create a soft link between the SAP directories and the /exports directories, or alternately, locally NFS mount the NFS shared directory. (**Note**: The name of the directory that we refer to as */exports* can vary according to user preference; however, for simplicity, we will refer to this directory as */exports* throughout this documentation.) For example, the following directories and links/ mounts for our example on the SAP Primary Server would be:

For the */usr/sap/trans* share	
Directory	**Notes**
/trans	created on shared file system and shared through NFS
/exports/usr/sap	mounted to / (on shared file system)
/usr/sap/trans	soft linked to */exports/usr/sap/trans*

Likewise, the following directories and links for the *<sapmnt>/<SAPSID>* share would be:

For the *<sapmnt>/<SAPSID>* share	
Directory	**Notes**
/<SAPSID>	created on shared file system and shared through NFS
/exports/sapmnt	mounted to / (on shared file system)
<sapmnt>/<SAPSID>	NFS mounted to *<virtual SAP server>:/exports/sapmnt/<SAPSID>*

Detailed instructions are given for creating all directory structures and links in the configuration steps later in this documentation. See the NFS Server Recovery Kit Documentation for additional information on inode conflicts and for information on using the new features in NFSv4.

Local NFS Mounts

The recommended directory structure for SAP in a LifeKeeper environment requires a locally mounted NFS share for one or more SAP system directories. If the NFS export point for any of the locally mounted NFS shares becomes unavailable, the system may hang while waiting for the export point to become available again. Many system operations will not work correctly, including a system reboot. You

should be aware that the NFS server for the SAP cluster should be protected by LifeKeeper and should not be manually taken out of service while local mount points exist.

To avoid accidentally causing your cluster to hang by inadvertently stopping the NFS server, please follow the recommendations listed in the NFS Considerations topic. It is additionally helpful to mount all NFS shares using the 'intr' mount option so that hung processes resulting from inaccessible NFS shares can be killed.

When NFS shares are not accessible the unmount can fail. LifeKeeper will attempt to unmount the filesystem multiple times. These multiple attempts will typically succeed in eventually taking the resource out of service. However, this will cause delays in taking the resource out of service. To avoid these retries, use 'nfsvers=3, proto=udp' mount options.

> ! Note the usage of udp; this is important for failover and recovery.

NFS Mounts and su

LifeKeeper accomplishes many database and SAP tasks by executing database and SAP operations using the `su - <admin name> -c <command>` command syntax. The `su` command, when called in this way, causes the login scripts in the administrator's home directory to be executed. These login scripts set environment variables to various SAP paths, some of which may reside on NFS mounted shares. If these NFS shares are not available for some reason, the su calls will hang, waiting for the NFS shares to become available again.

Since hung scripts can prevent LifeKeeper from functioning properly, it is desirable to configure your servers to account for this potential problem. The LifeKeeper scripts that handle SAP resource remove, restore and monitoring operations have a built-in timer that prevents these scripts from hanging indefinitely. No configuration actions are therefore required to handle NFS hangs for the SAP Application Recovery Kit.

Note that there are many manual operations that unavailable NFS shares will still affect. You should always ensure that all NFS shares are available prior to executing manual LifeKeeper operations.

> ✻ To avoid any delay in handling inaccessible NFS shares please follow the recommendations listed in NFS Considerations.

Location of *<INST>* directories

Since the */usr/sap/<SAPSID>* path is not NFS shared, it can be mounted to the root directory of the file system. The */usr/sap/<SAPSID>* path contains the *SYS* subdirectory and an *<INST>* subdirectory for each SAP instance that can run on the server. For certain configurations, there may only be one *<INST>* directory, so it is acceptable for it to be located under */usr/sap/<SAPSID>* on the shared file system. For other configurations, however, the backup server may also contain a local AS instance whose *<INST>*

directory should not be on a shared file system since it will not always be available. To solve this problem, it is recommended that for certain configurations, the PAS's, ASCS's or SCS's */usr/sap/<SAPSID>/<INST>, /usr/sap/<SAPSID>/<ASCS-INST>* or */usr/sap/<SAPSID>/<SCS-INST>* directories should be mounted to the shared file system instead of */usr/sap/<SAPSID>* and the */usr/sap/<SAPSID>/SYS* and */usr/sap/<SAPSID>/<AS-INST>* for the AS should be located on the local server.

> ✻ The ERS filesystem should be mounted locally in the case of ERSv1 and mounted on shared storage in the case of ERSv2.

For example, the following directories and mount points should be created for the ABAP+Java Configuration:

Directory	Notes
*/usr/sap/<SAPSID>/DVEBMGS<**No**.>*	mounted to / (on shared file system)
*/usr/sap/<SAPSID>/SCS<**No**.>*	mounted to / (on shared file system)
*/usr/sap/<SAPSID>/ERS<**No**.> (for SCS instance)*	should be locally mounted on all cluster nodes or mounted from a NAS share (*should not be mounted on shared storage*)
*/usr/sap/<SAPSID>/ASCS<**Instance No**.>*	mounted to / (on shared file system)
*/usr/sap/<SAPSID>/ERS<**No**.>* *(for ASCS instance)*	should be locally mounted on all cluster nodes or mounted from a NAS share (*should not be mounted on shared storage*)
*/usr/sap/<SAPSID>/AS<**Instance No**.>*	created for AS on backup server

Note: The Enqueue Replication Server (ERS) resource will be in-service (ISP) on the primary node in your cluster. However, the architecture and function of the ERS requires that the actual processes for the instance run on the backup node. This allows the standby server to hold a complete copy of the lock table information for the primary server and primary enqueue server instance. When the primary server running the enqueue server fails, it will be restarted by SIOS Protection Suite on the backup server on which the ERS process is currently running. The lock table (replication table) stored on the ERS is transferred to the enqueue server process being recovered and the new lock table is created from it. Once this process is complete, the active replication server is then deactivated (it closes the connection to the enqueue server and deletes the replication table). SIOS Protection Suite will then restart the ERS processes on the new current backup node (formerly the primary) which has been inactive until now. Once the ERS process becomes active, it connects to the enqueue server and creates a replication table. For more information on the ERS process and SAP architecture features, visit http://help.sap.com and search for **Enqueue Replication Service**.

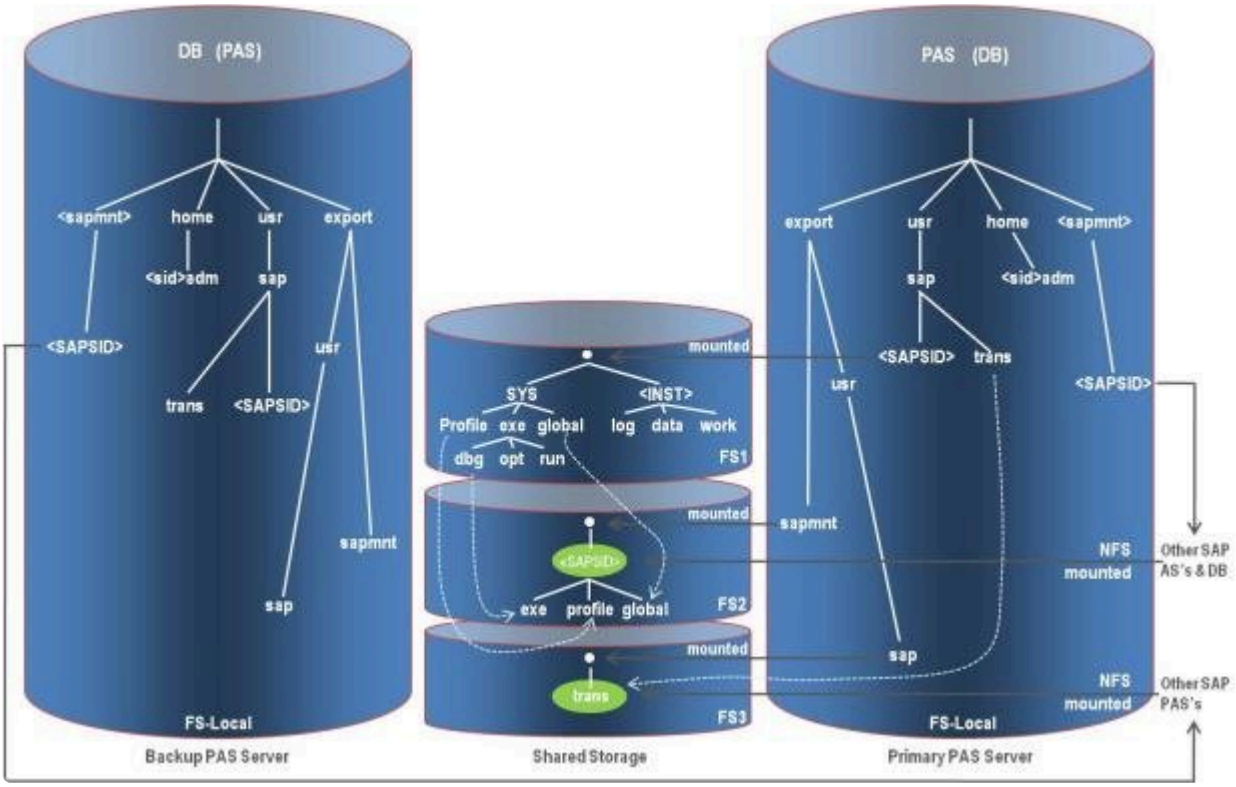
Since the replication server is always active on the backup node, it cannot reside on a SIOS Protection

Suite protected file system as the file system would be active on the primary node while the replication server process would be active on the backup node. Therefore, the file systems that ERS uses should be locally mounted on all cluster nodes or mounted from a NAS share.

Directory Structure Diagram

The directory structure required for LifeKeeper protection of ABAP only environments is shown graphically in the figure below. See the Abbreviations and Definitions section for a description of the abbreviations used in the figure.

Directory Structure Example



Legend

	soft link
	mount point
	NFS shared file system
PAS	Primary PAS
(PAS)	Backup PAS
AS	Application Server
DB	Primary DB Server
(DB)	Backup DB Server
FSn	File System on shared storage
FS-local	File System on local disk

Directory Structure Options

The configuration steps presented in this documentation are based on the directory structure and diagrams described above. This is the recommended directory structure as tested and certified by SIOS Technology Corp.

There are other directory structure variations that should also work with the SAP Recovery Kit, although not all of them have been tested. For configurations with directory structure variations, you should follow the guidelines below.

- The */usr/sap/trans* directory can be hosted on any server accessible on the network and does not have to be the PAS server. If you locate the */usr/sap/trans* directory remotely from the PAS, you will need to decide whether access to this directory is mission critical. If it is, then you may want to protect it with LifeKeeper. This will require that it be hosted on a shared or replicated file system and protected by the NFS Server Recovery Kit. If you have other methods of making the */usr/sap/trans* directory available to all of the SAP instances without NFS, this is also acceptable.

- The */usr/sap/trans* directory does not have to be NFS shared regardless of whether it is located on the PAS server.

- The */usr/sap/trans* directory does not have to be on a shared file system if it is not NFS shared or protected by LifeKeeper.

- The directory structure and path names used to export NFS file systems shown in the diagrams are examples only. The path */exports/usr/sap* could also be */exports/sap* or just */sap*.

- The */usr/sap/<SAPSID>/<INST>* path needs to be on a shared file system at some level (except in the case of an ERSv1 instance). It does not matter which part of this path is the mount point for the file system. It could be */usr, /usr/sap, /usr/sap/<SAPSID>* or */usr/sap/<SAPSID>/<INST>*.

- The */sapmnt/<SAPSID>* path needs to be on a shared file system at some level. The configuration diagrams show this path as NFS mounted, although this is an SAP requirement and not a LifeKeeper requirement.

6.16.6.5. SAP Virtual Server Name

SAP Application Servers and SAP clients communicate with the SAP Primary Application Server (PAS) using the name of the server where the PAS Instance is running. Likewise, the SAP PAS communicates with the Database (DB) using the name of the DB server. In a high availability (HA) environment, the PAS may be running on either the Primary Server or Backup Server at any given time. In order for other servers and clients to maintain a seamless connection to the PAS regardless of which server it is active on, a virtual server name is used for all communication with the PAS. This virtual server name is also mapped to a switchable IP address that can be active on whichever server the PAS is running on.

The switchable IP address is created and handled by LifeKeeper using the IP Recovery Kit. The virtual server name is configured manually by adding a virtual server name/switchable IP address mapping in DNS and/or in all of the servers' and clients' host files. See the IP Local Recovery topic for additional information on how this works.

Additionally, SAP configuration files must be modified so that the virtual server name is substituted for the physical server name. This is covered in detail in the Installation section where additional instructions are given for configuring SAP with LifeKeeper.

Note: A separate switchable IP address is recommended for use with SAP Application Server hierarchies and the NFS Server hierachies. This allows the IP address used for NFS clients to remain separate from the IP used for SAP clients.

6.16.6.6. SAP Health Monitoring

LifeKeeper monitors the health of the Primary Application Server (PAS) Instance and initiates a recovery operation if it determines that SAP is not functioning correctly.

The status will be returned to the user, via the GUI Properties Panel and CLI, as **Gray** (unknown/ inactive/offline), **Red** (failed), **Yellow** (issue) or **Green** (healthy).

If the status of the instance is **Gray**, state is unknown; no information is available.

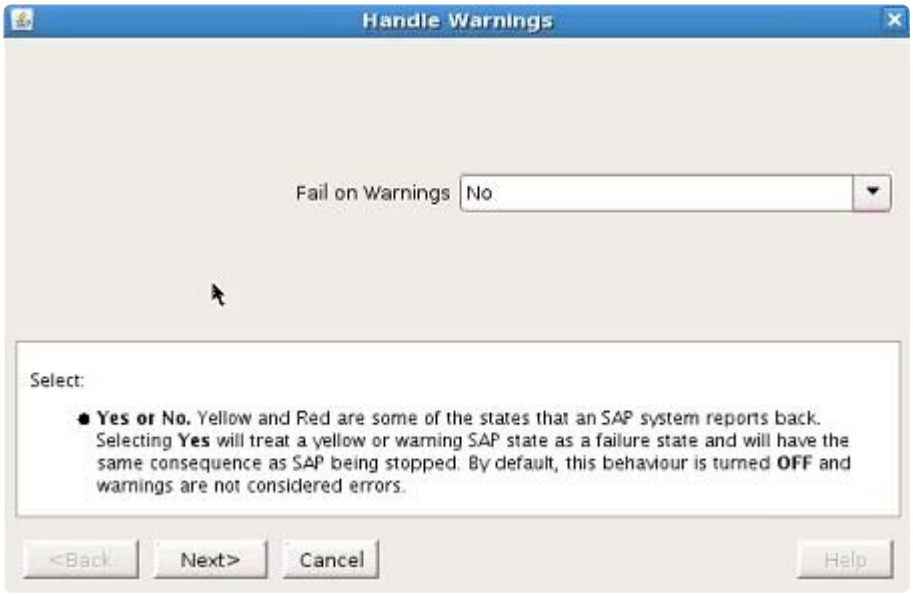
If the status of the instance is **Red**, the resource will be considered in a failed state and LifeKeeper will initiate the appropriate recovery handling operations.

If the status of the instance is **Yellow**, it indicates that there may be an issue with the SAP processes for the defined Instance. The default behavior for a yellow status is to continue monitoring without initiating recovery.

This default behavior can be changed by configuring this option via the GUI resource menu.

1. Right-click the **Instance**.
2. Select **Handle Warnings**.

3. The following screen will appear, prompting you to select whether to **Fail on Warnings**.



Selecting **Yes** will cause a **Yellow Warning** to be treated as an error and will initiate recovery.

Note: It is highly recommended that this setting be left on the default selection of **No** as Yellow is a transient state that most often does not indicate a failure.

6.16.6.7. SAP License

In a high availability (HA) environment, SAP is configured to run on both a Primary and a Backup Server. Since the SAP licensing scheme is hardware dependent, a separate license is required for each server where SAP is configured to run. It will, therefore, be necessary to obtain and install an SAP license for both the Primary and Backup Servers.

6.16.6.8. SAP Automatic Switchback

In Active/Active configurations, the SAP Primary Application Server Instance (PAS), ABAP SAP Central Services Instance (ASCS) or SAP Central Services Instance (SCS) and Database (DB) hierarchies are separate and are in service on different servers during normal operation. There are times, however, when both hierarchies will be in service on the same server such as when one of the servers is being taken down for maintenance. If both hierarchies are in service on one of the servers and both servers go down, then when the servers come back up, it is important that the database hierarchy come in service before the SAP hierarchy in-service operation times out. Since LifeKeeper brings hierarchies in service during startup serially, if it chooses to bring SAP up first, the database in-service operation will wait on the SAP in-service operation to complete and the SAP in-service operation will wait on the database to become available, which will never happen because the DB restore operation can only begin after the PAS, ASCS or SCS restore completes. This deadlock condition will exist until the PAS, ASCS or SCS restore operation times out. (**Note**: SAP will time out and fail after 10 minutes.)

To prevent this deadlock scenario, it is important for this configuration to set the switchback flag for both hierarchies to **Automatic Switchback**. This will force LifeKeeper to restore each hierarchy on its highest priority server during LifeKeeper startup, which in this case is two different servers. Since LifeKeeper restore operations on different servers can occur in parallel, the deadlock condition is prevented.

6.16.6.9. Notes – Special Configuration Steps

The following items require special configuration steps in a high availability (HA) environment. Please consult the document *SAP Web Application Server in Switchover Environments* for additional information on configuration requirements for each:

- Login Groups
- SAP Spoolers
- Batch Jobs
- SAP Router
- SAP System Upgrades

6.16.7. SAP Installation

Configuration/Installation

Before using LifeKeeper to create an SAP resource hierarchy, perform the following tasks **in the order recommended** below. Note that there are additional non-HA specific configuration tasks that must be performed that are not listed below. Consult the appropriate SAP installation guide for additional details.

The following tasks refer to the "**SAP Primary Server**" and "**SAP Backup Server**." The **SAP Primary Server** is the server on which the Central Services will run during normal operation, and the **SAP Backup Server** is the server on which the Central Services will run if the SAP Primary Server fails.

Although it is not necessarily required, the steps below include the recommended procedure of protecting all shared file systems with LifeKeeper prior to using them. Prior to LifeKeeper protection, a shared file system is accessible from both servers and is susceptible to data corruption. Using LifeKeeper to protect the file systems preserves single server access to the data.

Before Installing SAP

The tasks in the following topic are required before installing your SAP software. Perform these tasks in the order given. Please also refer to the SAP document *SAP Web Application Server in Switchover Environments* when planning your installation in NetWeaver Environments.

Plan Your Configuration

Installing SAP Software

These tasks are required to install your SAP software for high availability. Perform the tasks below in the order given. Click on each task for details. Please refer to the appropriate SAP Installation Guide for further SAP installation instructions.

Primary Server Installation

Install the Core Services, ABAP and Java Central Services

Install the Database

Install the Primary Application Server Instance

Install Additional Application Server Instances

Backup Server Installation

Install on the Backup Server

Installing LifeKeeper

Configuring SAP with LifeKeeper

Resource Configuration Tasks

The following tasks explain how to configure your recovery kit by selecting certain options from the **Edit** menu of the LifeKeeper GUI. Each configuration task can also be selected from the toolbar or you may right-click on a global resource in the **Resource Hierarchy Tree** (left-hand pane) of the status display window to display the same drop down menu choices as the **Edit** menu. This, of course, is only an option when a hierarchy already exists.

Alternatively, right-click on a resource instance in the **Resource Hierarchy Table** (right-hand pane) of the status display window to perform all the configuration tasks, except creating a resource hierarchy, depending on the state of the server and the particular resource.

To enable the SAP SIOS HA Cluster Connector for an SAP instance, see Activating the SAP SIOS HA Cluster Connector (SSHCC).

For proper administration of the ERS instance in LifeKeeper, the ERS profile must use the Start_Program parameter instead of Restart_Program for starting the ERS process. See the ASCS + ERS Restart_Program Parameter page for details on how to modify this parameter in the ERS instance profile.

Test the SAP Resource Hierarchy

You should thoroughly test the SAP hierarchy after establishing LifeKeeper protection for your SAP software. Perform the tasks in the order given.

6.16.7.1. Plan Your SAP Configuration

1. Determine which configuration you wish to use. The required tasks vary depending on the configuration.

2. Determine whether the SAP system-wide */usr/sap/trans* directory will be hosted on the SAP Primary Application Server or on a file server. It can be hosted either place as long as it is NFS shared and fully accessible. If it is hosted on the SAP Primary Application Server and located on a shared file system, it should be protected by LifeKeeper and included in the SAP hierarchy.

3. Consider the storage requirements for SAP and DB as listed in the *SAP Installation Guide*. Most of the SAP files will have to be installed on shared storage. Consult the SPS for Linux Technical Documentation for the database-specific recovery kit for information on which database files are installed on shared storage and which are installed locally. Note that in an SAP environment, SAP requires local access to the database binaries, so they will have to be installed locally. Determine how to best use your shared storage to meet these requirements.

   Also note that when shared storage resources are under LifeKeeper protection, they can only be accessed by one server at a time. If the shared device is a disk array, an entire LUN is protected. If a shared device is a disk, then the entire disk is protected. All file systems located on a single volume will therefore be controlled by LifeKeeper together. This means that you must have at least two logical volumes (LUNs), one for the database and one for SAP.

4. Virtual host names will be needed in order to identify your systems for failover. A new IP address is required for each virtual host name used. Make sure that the virtual host name can be correctly resolved in your Domain Name System (DNS) setup, then proceed as follows:

   a. Create the new virtual ip addresses by using the command:

   ```
   ifconfig eth0:1 {IPADDRESS} netmask {255.255.252.0} (use the right
   netmask for your configuration)
   ```

   Note: To verify these new virtual ip addresses, either the ifconfig or ip addr show command can be used if using LifeKeeper 7.3 or earlier. However, starting with LifeKeeper 7.4, the ip addr show command should be used.

   b. Repeat with eth0:2 for the database virtual IP.

   c. A separate virtual IP will also be needed for the ERS instance if using ERSv2.

   In order to associate the switchable IP addresses with the virtual server name, edit /etc/hosts and add the new virtual ip addresses.

   Note: This step is optional if the Primary Application Server and the Database are always running on the same server and communication between them is always local. But it is advisable to have separate switchable IP addresses and virtual server names for the Primary Application Server and

the Database in case you ever want to run them on different servers.

5. Stop the caching daemon on both machines.

```
rcnscd stop
```

6. Mount the software.

```
mount //{path of software} (no password needed)
```

7. Run an X session (either an ssh -X or a VNC session — for Microsoft Windows users, Hummingbird Exceed X Windows can be used).

   Note: When `sapinst` is run, the directory will be extracted under */tmp*

8. When installing SAP (specifically ASCS or SCS) you need to specify the SAPINST_USE_HOSTNAME=vip option. This is also required for the ERS instance if using ERSv2, but not if using ERSv1.

   Note: Specify `sapinst SAPINST_USE_HOSTNAME=vip` where **vip** is the virtual IP that will float between the nodes.

Important Note

The LifeKeeper SAP Recovery Kit relies on the SAP Host agent being installed. If this software is not installed, then the LifeKeeper SAP kit will not install. With SAP Netweaver Version 7.3 and higher, this host agent is supplied; however, prior versions require a download from SAP. It is recommended that you consult your SAP help notes for your specific version. You can also refer to the Help Forum (help.sap.com) for further documentation.

- The saphostexec module, either in RPM or SAR format, can be downloaded from SAP.
- To make sure that the modules are installed properly, there are a few modules to search for (*saposcol, saphostexec, saphostctrl*). These modules are typically found where SAP is installed (typically */usr/sap directory*).

6.16.7.2. Installation of the Core Services

Before installing software, make sure that the date/time is synchronized on all servers. This is important for both LifeKeeper and SAP.

The Core Services, ABAP and Java Central Services (ASCS and SCS), are single points of failure (SPOFs) and therefore must be protected by LifeKeeper. Install these core services on the SAP Primary Server using the appropriate SAP Installation Guide.

Installation Notes

- To be able to use the required virtual host names that were created in the Plan Your Configuration topic, set the `SAPinst` property `SAPINST_USE_HOSTNAME` to specify the required virtual host names before starting `SAPinst`. (**Note**: Document the SAPINST_USE_HOSTNAME virtual IP address as it will be used later during creation of the SAP resources in LifeKeeper.)

  Run `./sapinst SAPINST_USE_HOSTNAME={hostname}`

- In seven phases, the **Core Services** should be created and started. If permission errors occur on `jdbcconnect.jar`, go to */sapmnt/STC/exe/uc/linuxx86_64* and make that directory as well as file `jdbcconnect.jar` writeable (chmod 777 —-).

- Installation completes with a success message.

6.16.7.3. Installation of the Database

1. Note the group id for dba and oinstall as this will be needed for the backup machine.
2. Change to the software directory and run the following:

```
./sapinst SAPINST_USE_HOSTNAME={database connectivity ip address}
```

3. Run SAPinst to install the Database Instance using the appropriate SAP Installation Guide.

Installation Notes

- SIOS recommends removing the orarun package, if it is already installed, prior to installation of the Database Instance (see **SAP Note 1257556**).
- The database installation option in the SAPinst window assumes that the database software is already installed, except for Oracle. For Oracle databases, SAPinst stops the installation and prompts you to install the database software.
- The <DBSID> identifies the database instance. `SAPinst` prompts you for the <DBSID> when you are installing the database instance. The <DBSID> can be the same as the <SAPSID>.
- If you install a database on a host other than the SAP Global host, you must mount global directories from the SAP Global host.
- If you run into an issue where the Listener was started, kill it using the command

```
(ps -ef | grep lsnrctl)
```

- To reset passwords for `SAPR3` and `SAPR3DB` userids, use the command

```
brtools
```

After Database installation is complete, close the original dialog and continue with SAP installation, Installing Application Services.

6.16.7.4. Installation of the Primary Application Server Instance

1. To install the Primary Application Server instance, rerun sapinst from the previously mentioned directory.

   ./sapinst SAPINST_USE_HOSTNAME=<vip>

2. When prompted, Select **Primary Application Server Instance** and continue with installation using the appropriate SAP Installation Guides.

Installation Notes

- The Primary Application Server Instance does not need to be part of the cluster because it is no longer a single point of failure (SPOF). The SPOF is now in the central services instances (SCS instance and ASCS instance), which are protected by the cluster.
- The directory of the Primary Application Server Instance is called DVEBMGS<*No*>, where <*No*> is the instance number.
- Installation of application server is complete when the **OK** message is received.
- When installing replicated enqueue on 7.1, run `sapinst` as-is.

6.16.7.5. Installation of Additional Application Server Instances

It is recommended that Additional Application Server Instances be installed to create redundancy. Application Server Instances are not SPOFs, therefore, they do not need to be included in the cluster.

On every additional application server instance host, do the following:

1. Run `SAPinst` to install the Additional Application Server Instance.

2. When prompted, select **Additional Application Server Instance** and continue with installation using the appropriate SAP Installation Guide.

6.16.7.6. Installation on the Backup Server

On the backup server, repeat the Installation procedures that were performed on the primary server:

1. Install the Core Services, ABAP and Java Central Services
2. Install the Database
3. Install the Application Services

6.16.7.7. Install SPS

On both the **Primary** and the **Backup** servers, LifeKeeper software will now be installed including the following recovery kits:

- SAP
- appropriate database (i.e. Oracle, SAP DB)
- IP
- NFS
- NAS

1. Stop **Oracle Listener** and **SAP** on both machines.

   For Example, if the Oracle user is *orastc*, the Oracle listener is *LISTENER_STC*, and the SAP user is *stcadm*:

   a. `su` to user `orastc` and run command `lsnrctl stop LISTENER_STC`

   b. `su` to user `stcadm` and run command `stopsap sap{No.}`

   c. From root user, make sure there are no SAP or Oracle user processes; if there are, enter "`killall sapstartsrv`"; even after this command, if there are processes, run "`ps -ef`" and kill each process

2. Go into */etc/hosts* on both machines and ensure that host and/or DNS entries are properly specified.

3. **Stop** and **remove** the IP addresses from the current interfaces. **Note**: This step is required before the IP addresses can be protected by the LifeKeeper IP Recovery Kit.

   ```
   ifconfig eth0:1 down

   ifconfig eth0:2 down
   ```

4. Verify the IP addresses have been removed by performing a connection attempt, for example using ping.

5. Following the steps in the SPS Installation Guide, install SPS on both the Primary server and the Backup server (DE, core, DataKeeper, LVM as well as the licenses). When prompted to select Recovery Kits, make sure you select the following:

   SAP, appropriate database (i.e. Oracle), IP, NFS and NAS

The installation script does certain checking and might fail if the environment is not set up correctly as shown in this example:

```
SAP Services file /usr/sap/sapservices not found

SAP Installation is not valid; please check environment and retry

error: %pre(steeleye-lkSAP-7.3.1-1.noarch) scriptlet failed, exit
status 2

error: install: %pre scriptlet failed (2), skipping steeleye-
lkSAP-7.3.1-1
```

In the above example, the expected SAP file, `/usr/sap/sapservices`, is missing. It is very important for the environment to be in the right state before installation can continue.

Refer to the Recovery Kit Documentation for additional information about installing the recovery kits and configuring the servers for protecting resources.

6.16.7.8. Create File Systems and Directory Structure

While there are many different configurations depending upon which database management system is being used, below is the basic layout that should be adhered to.

- Set up comm paths between the primary and secondary servers
- Add virtual ip resources to *etc/hosts*
- Create virtual ip resources for instance and database hosts
- Set up shared disks
- Create file systems for SAP (located on shared disk)
- Create file systems for database (located on shared disk)
- Mount the main SAP file systems
- Create mount points
- Mount the PAS, ASCS/SCS, and ERS directories (if applicable) as well as any additional Application Servers

Please consult the SAP installation guide specific to the database management system for details on the directory structure for the database. All database files must be located on shared disks to be protected by the LifeKeeper Recovery Kit for the database. Consult the database specific Recovery Kit Documentation for additional information on protecting the database.

The following example is only a sample of the many configurations than can be established, but understanding these configurations and adhering to the configuration rules will help define and set up workable solutions for your computing environment.

1. From the UI of the primary server, set up comm paths between the primary server and the secondary server.
2. Add an entry for the actual primary and secondary virtual ip addresses in */etc/hosts.*
3. Log in to LifeKeeper on the primary server and create virtual ip resources for your host and your database (ex. *ip-db10* and *ip-sap10*).
4. Set up shared disks between the two machines.

   Note: One lun for database and another for SAP data is recommended in order to enable

independent failover.

5. For certain configurations, the following tasks may need to be completed:

- Create the physical devices
- Create the volume group
- Create the logical volumes for SAP
- Create the logical volumes for Database

6. Create the file systems on shared storage for SAP (these are *sapmnt*, *saptrans*, *ASCS{No}*, *SCS{No}*, *DVEBMGS{No}*). **Note**: SAP must be stopped in order to get everything on shared storage.

7. Create all file systems required for your database (Example: *mirrlogA*, *mirrlogB*, *origlogA*, *origlogB*, *sapdata1*, *sapdata2*, *sapdata3*, *sapdata4*, *oraarch*, *saparch*, *sapreorg*, *saptrace*, *oraflash* — *mkfs -t ext3 /dev/oracle/mirrlogA*).

   Note: Consult the SPS for Linux Technical Documentation for the database-specific recovery kit and the *Component Installation Guide SAP Web Application Server* for additional information on which file systems need to be created and protected by LifeKeeper.

8. Create mount points for the main SAP file systems and then mount them (required). For additional information, see the NFS Mount Points and Inodes topic. (Note: */exports* directory was used to mount the file systems.)

```
mount /dev/sap/sapmnt /exports/sapmnt

mount /dev/sap/saptrans /exports/saptrans
```

9. Create temporary mount points using the following command.

```
mkdir /tmp/m{No}
```

10. Mount the three SAP directories (the following mount points are necessary for each Application Server present whether using external NFS or not).

```
mount /dev/sap/ASCS00 /tmp/m1

mount /dev/sap/SCS01 /tmp/m2

mount /dev/sap/DVEBMGS02 /tmp/m3
```

Proceed to Moving Data to Shared Disk and LifeKeeper.

6.16.7.9. Move Data to Shared Disk and LifeKeeper

The following steps are an example using Oracle. **Note:** In this example STC is being used as the DB ID as well as the SAP SID. All occurrences of “STC” or “stc” in these commands (e.g., /usr/sap/STC or user orastc) should be replaced with the actual DB ID or SAP SID being used in the user’s cluster configuration.

Before Beginning: Primary and backup have been specified for the two servers. At the end of this procedure, the roles will be reversed. It is recommended that you first read through the steps, plan out which machine will be the desired primary and which will be the intended backup. At the end of this procedure, the role of primary and backup should become interchangeable. Understanding that in certain environments some machines are intended to be primary and some the backups, it is important to understand how this is structured.

1. Change directory to */usr/sap/STC*, then change to each subdirectory and copy the data.
    - ◦ cd ASCS{No.}
    - ◦ cp –a * /tmp/m1
    - ◦ cd ../SCS{No.}
    - ◦ cp –a * /tmp/m2
    - ◦ cd ../DVEBMGS{No.}
    - ◦ cp –a * /tmp/m3
2. Change the temporary directories to the correct user permission.

```
chown stcadm:sapsys /tmp/m1 (repeat for m2 and m3)
```

3. Unmount the three temp directories using `umount /tmp/m1` and repeat for m2 and m3.
4. Re-mount the device over the old directories.

```
mount /dev/sap/ASCS{No.} /usr/sap/STC/ASCS{No.}

mount /dev/sap/SCS{No.} /usr/sap/STC/SCS{No.}

mount /dev/sap/DVEBMGS{No.} /usr/sap/STC/DVEBMGS{No.}
```

5. Mount the thirteen temp directories for Oracle.

```
mount /dev/oracle/sapdata1 /tmp/m1

mount /dev/oracle/sapdata2 /tmp/m2

mount /dev/oracle/sapdata3 /tmp/m3

mount /dev/oracle/sapdata4 /tmp/m4

mount /dev/oracle/mirrlogA /tmp/m5

mount /dev/oracle/mirrlogB /tmp/m6

mount /dev/oracle/origlogA /tmp/m7

mount /dev/oracle/origlogB /tmp/m8

mount /dev/oracle/saparch /tmp/m9

mount /dev/oracle/sapreorg /tmp/m10

mount /dev/oracle/saptrace /tmp/m11

mount /dev/oracle/oraarch /tmp/m12

mount /dev/oracle/oraflash /tmp/m13
```

6. Change the directory to */oracle/STC* and copy the data.

   a. Change to each subdirectory (cd `cd sapdata1` and perform `cp –a * /tmp/m1`)

7. Repeat this previous step for each subdirectory as shown in the relationship above.

8. Change the temporary directories to the correct user permission.

```
chown orastc:dba /tmp/m1 (repeat for m2 to m12)
```

9. Unmount all the temp directories.

```
umount /tmp/m*
```

10. Re-mount the device over the old directories.

```
mount /dev/oracle/sapdata1 /oracle/STC/sapdata1
```

11. Repeat the above for all the listed directories.

12. Edit the */etc/exports* file and insert the mount points for SAP's main directories.

```
/exports/sapmnt *(rw,sync,no_root_squash)

/exports/saptrans *(rw,sync,no_root_squash)
```

13. Start the NFS server using the `rcnfsserver start` command (this is for SLES; for Red Hat, perform `service nfs start or systemctl start nfs`). If the NFS server is already active, you may need to do an "`exportfs -va`" to export those mount points.

14. Execute the following mount commands (**note the usage of udp; this is important for failover and recovery**).

```
mount {virtual ip}:/exports/sapmnt/ /sapmnt/ -o rw,sync,bg,intr,udp

mount {virtual ip}:/exports/saptrans /usr/sap/trans -o
rw,sync,bg,intr,udp
```

15. Log in to Oracle and start Oracle (after su to orastc).

```
lsnrctl start LISTENER_STC

sqlplus / as sysdba

startup
```

16. Log in to SAP and start SAP (after su to stcadm).

```
startsap sap{No.}
```

17. Make sure all processes have started.

```
ps -ef | grep en.sap (2 processes)

ps -ef | grep ms.sap (2 processes)

ps -ef | grep dw.sap (17 processes)
```

    SAP MCC (Microsoft Management Console Snap-In for SAP) is an SAP-supplied Windows client that can be used to administer SAP instances. A corresponding version for Unix/Linux called SAP MC (SAP Management Console) is also available.

18. Stop SAP and the Oracle Listener. (**Note:** Note in Step c, we use the SQL*Plus utility from Oracle to log in to Oracle and shut down the database.)

a. su to stcadm and enter command "`stopsap sap{No.}`"

b. su to orastc and enter command "`lsnrctl stop LISTENER_STC`"

c. su to orastc and enter "`sqlplus sys as SYSDBA`" and enter "`shutdown`" at the command prompt

d. enter command "`stopsap sap{No.}`"

e. `killall sapstartsrv` as root

f. kill any leftover processes still associated to the stcadm and orastc users (e.g., `ps -u stcadm` and `ps -u orastc`)

19. Unmount all the file systems.

```
umount /usr/sap/trans

umount /sapmnt/STC

umount /oracle/STC/*

umount /usr/sap/STC/DVEBMGS{No.}

umount /usr/sap/STC/SCS{No.}

umount /usr/sap/STC/ASCS{No.}
```

20. Stop the NFS server (using the command `rcnfsserver stop` in SLES or `service nfs stop` or `systemctl stop nfs` in RHEL) and perform the unmounts.

```
umount /exports/sapmnt

umount /exports/saptrans
```

21. Copy */etc/exports* to the backup system.

```
scp /etc/exports (backup ip):/etc/exports
```

22. Deactivate the logical volumes on the primary.

```
lvchange -an oracle

lvchange -an sap
```

23. Create the corresponding SAP directories on the backup system.

```
mkdir –p /exports/sapmnt

mkdir –p /exports/saptrans
```

24. Activate the logical volumes on the backup system.

```
lvchange -ay oracle

lvchange -ay sap
```

Note: Problems may occur on this step if any rearranging of storage occurred on the primary when the volume groups were built. A reboot of the backup will clear this up.

25. Mount the directories on the backup machine.

```
mount /dev/sap/sapmnt /exports/sapmnt

mount /dev/sap/saptrans /export/saptrans

mount /dev/sap/ASCS00 /usr/sap/STC/ASCS{No.}

mount /dev/sap/SCS01 /usr/sap/STC/SCS{No.}

mount /dev/sap/DVEBMGS02 /usr/sap/STC/DVEBMGS{No.}

mount /dev/oracle/sapdata1 /oracle/STC/sapdata1

mount /dev/oracle/sapdata2 /oracle/STC/sapdata2

mount /dev/oracle/sapdata3 /oracle/STC/sapdata3

mount /dev/oracle/sapdata4 /oracle/STC/sapdata4

mount /dev/oracle/origlogA /oracle/STC/origlogA

mount /dev/oracle/origlogB /oracle/STC/origlogB

mount /dev/oracle/mirrlogA /oracle/STC/mirrlogA

mount /dev/oracle/mirrlogB /oracle/STC/mirrlogB

mount /dev/oracle/oraarch /oracle/STC/oraarch

mount /dev/oracle/saparch /oracle/STC/saparch

mount /dev/oracle/saptrace /oracle/STC/saptrace
```

```
mount /dev/oracle/sapreorg /oracle/STC/sapreorg
```

26. Switch over the IP addresses to the backup system via LifeKeeper.
27. Mount the NFS exports on the backup

```
mount sap{No.}:/exports/sapmnt/STC /sapmnt/STC

mount sap{No.}:/exports/saptrans/trans /usr/sap/trans
```

28. Log in to Oracle and start Oracle (after su to orastc).

```
lsnrctl start LISTENER_STC

sqlplus / as sysdba

startup
```

29. Log in to SAP and start SAP (after su to stcadm).

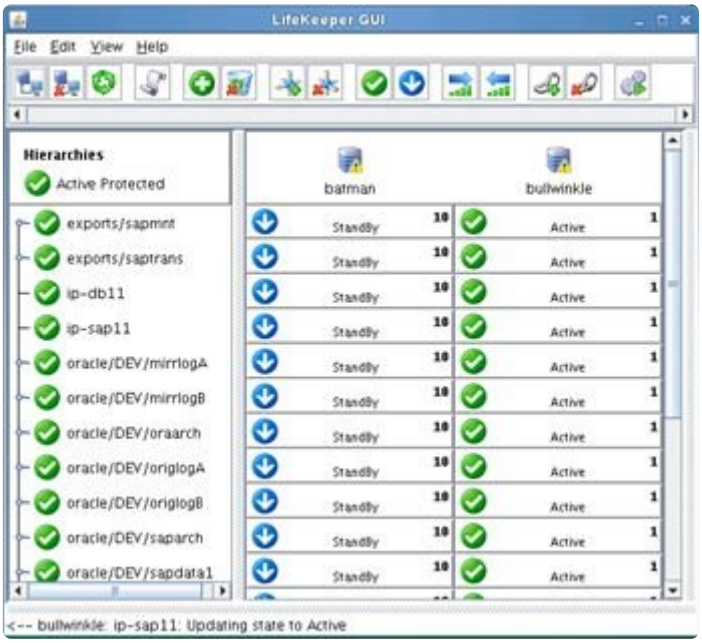
```
startsap sap{No.}
```

30. Log in to LifeKeeper and switch primary and backup priority instances (make backup higher priority).
31. On the original primary, save the original directories as such:

```
mv /exports /exports-save

mv /usr/sap/STC/DVEBMGS{No.} /usr/sap/STC/DVEBMGS{No.}-save (repeat
for SCS{No.} and ASCS{No.})

mv /oracle/STC/sapdata1 /oracle/STC/sapdata1-save (repeat for
sapdata2, sapdata3, sapdata4, mirrlogA, mirrlogB, origlogA,
origlogB, sapreorg, saptrace, saparch, oraarch)
```

32. Create “file system” resources, all the 17 mount points (5 for SAP and 12 for Oracle) one by one.
33. Extend to the original primary.

LifeKeeper resource hierarchy and SAP cluster are set up. (**Note**: This is a screen shot from the DEV instance.)


LifeKeeper GUI
File Edit View Help
Hierarchies
Active Protected
exports/sapmnt
exports/saptrans
ip-db11
ip-sap11
oracle/DEV/mirrlogA
oracle/DEV/mirrlogB
oracle/DEV/oraarch
oracle/DEV/origlogA
oracle/DEV/origlogB
oracle/DEV/saparch
oracle/DEV/sapdata1
batman
bullwinkle
Standby
Active
<-- bullwinkle: ip-sap11: Updating state to Active

6.16.7.10. Modify ASCS and ERS Instance Profile Settings

Change the Restart_Program parameter to Start_Program for the enqueue server and enqueue replicator processes (in the ASCS and ERS instance profiles, respectively) to prevent the sapstart utility from automatically restarting them. Also change the entry ‘Autostart = 0’ to ‘Autostart = 1’ in both instance profiles to prevent each instance from being automatically restarted when the system reboots.

Before performing the following steps, identify the locations of the active ASCS and ERS instance profiles. This can be found from the /usr/sap/sapservices file by looking at the path of the file provided to sapstartsrv as the ‘pf=’ parameter:

> **Note:** The following is only an example, where we are using ASCS instance number, 00 and ERS on 10, change the numbers accordingly for your environment. Also replace the <SID> to be the actual System ID.

```
LD_LIBRARY_PATH=/usr/sap/SID/ASCS00/exe:$LD_LIBRARY_PATH; export

LD_LIBRARY_PATH; /usr/sap/SID/ASCS00/exe/sapstartsrv

pf= /usr/sap/SID/SYS/profile/SID_ASCS00_sap1 -D -u SIDadm

LD_LIBRARY_PATH=/usr/sap/SID/ERS10/exe:$LD_LIBRARY_PATH; export

LD_LIBRARY_PATH; /usr/sap/SID/ERS10/exe/sapstartsrv

pf= /usr/sap/SID/SYS/profile/SID_ERS10_sap2 -D -u SIDadm
```

In this example, the active ASCS instance profile is located at /usr/sap/SID/SYS/profile/SID_ASCS00_sap1 and the active ERS instance profile is located at /usr/sap/SID/SYS/profile/SID_ERS10_sap2.

Steps

1. Edit the ASCS instance profile as follows:

   a. Change ‘Autostart = 1’ to ‘Autostart = 0’, or manually add the line ‘Autostart = 0’.

   b. The exact format of the line in the profile that starts the enqueue server process will vary depending on whether version 1 or 2 of the Standalone Enqueue Server Framework is being used. Modify the profile according to the following table:

If the following line appears…	Change it to…

Restart_Program_01 = local $(_EN) pf=$(_PF)	Start_Program_01 = local $(_EN) pf=$(_PF)
Restart_Program_01 = local $(_ES2) pf=$(_PF)	Start_Program_01 = local $(_ES2) pf=$(_PF)
Restart_Program_01 = local $(_ENQ) pf=$(_PF)	Start_Program_01 = local $(_ENQ) pf=$(_PF)

Note: The numbers xx in the Start_Program_xx or Restart_Program_xx entries may be different on your system. They do not need to be changed to match the lines shown above.

2. Verify that all entries are correct and save the changes to the ASCS instance profile.

3. Edit the ERS instance profile as follows:

   a. Change ‘Autostart = 1’ to ‘Autostart = 0’, or manually add the line ‘Autostart = 0’.

   b. The exact format of the line in the profile that starts the enqueue replicator process will vary depending on whether version 1 or 2 of the Standalone Enqueue Server Framework is being used. Modify the profile according to the following table:

If the following line appears…	Change it to…
Restart_Program_00 = local $(_ER) pf=$(_PFL) NR=$(SCSID)	Start_Program_00 = local $(_ER) pf=$(_PFL) NR=$(SCSID)
Restart_Program_00 = local $(_ER2) pf=$(_PF)	Start_Program_00 = local $(_ER2) pf=$(_PF)
Restart_Program_00 = local $(_ENQR) pf=$(_PF)	Start_Program_00 = local $(_ENQR) pf=$(_PF)

Note: The numbers xx in the Start_Program_xx or Restart_Program_xx entries may be different on your system. They do not need to be changed to match the lines shown above.

5. Verify that all entries are correct and save the changes to the ERS instance profile.

6. SIOS recommends restarting the system to ensure the updated profile is read and no caching is in effect.

7. SIOS also recommends to always verify changes in a test environment before applying to production workload.

Notes:

1. The entries ‘Autostart = 0’ and ‘Start_Program_xx …’ must be on separate lines in each instance profile.

2. The numbers xx in the Start_Program_xx or Restart_Program_xx entries may be different on your system. They do not need to be changed to match the numbers given in this solution.

3. In a Java-based or dual stack Java+ABAP deployment there will be an SCS central services instance. In this case, steps 1-3 also need to be performed for the SCS instance.

4. See SAP Note 768727 (Automatic restart functions in sapstart for processes) for more details on the differences between Start_Program and Restart_Program.

6.16.7.11. Upgrading from a Previous Version of the SAP Recovery Kit

To upgrade from a previous version of the SAP Recovery Kit, perform the following steps.

1. Prior to upgrading, please review the Plan Your Configuration topic to make sure you understand all the implications of the new software.

   Note: If running a version prior to SAP Netweaver 7.3, the SAPHOST agent will need to be installed. See the Important Note in the Plan Your Configuration topic for more information.

   It is recommended that you take a snapshot of your current hierarchy using the lkbackup utility.

2. Follow the instructions in the "Upgrading SPS" topic in the *SPS for Linux Installation Guide*.

   A backup will be performed of the existing hierarchy. The upgrade will then destroy the old hierarchy and recreate the new hierarchy. If there is a failure, see In Case of Failure below.

3. At the end of the upgrade, stop and restart the LifeKeeper GUI in order to load the updated GUI client.

   The LifeKeeper GUI server caches pages, so a restart is needed for it to refresh the new pages. As root, enter the command "`lkGUIserver restart`" which should stop and restart the GUI server. Exit all clients before attempting such a restart.

   Note: Restarting your entire LifeKeeper system is not necessary, but it would be advisable in a production setting to schedule some down time and go through an orderly system preparation time even though testing has not required a system recycle.

4. Log in to the LifeKeeper UI, note the hierarchy and make sure the hierarchy is correct.

In Case of Failure

It is possible to retry the upgrade. The upgrade script is kept intact in */tmp* directory (*lkcreatesaptmp*). This is a temporary file that is used during the upgrade. The commands are written here and can be executed to create the hierarchy.

If there is a failure, an error or you suspect the hierarchy is not correct, the following steps are recommended:

1. Stop LifeKeeper using `/etc/init.d/lifekeeper stop-nofailover`

2. Remove the new rpm "*rpm -e steeleye-lkSAP*".

3. Install the old rpm "*rpm -i steeleye-lkSAP-<previous version>.noarch.rpm*".

4. Restore the old hierarchy using `lkbackup -x`

5. Restart LifeKeeper.

6. Contact SIOS Support for help. Prior to contacting Support, please have on hand the logs, the previous snapshot of the hierarchy, the hierarchy that was created and failed and the error messages received during the upgrade.

6.16.7.12. SAP IP Resources

Before continuing to set up the LifeKeeper hierarchy, determine the IP address that the SAP resource will use for failover or switchover. This is typically the virtual IP address used during the installation of SAP using the parameter SAPINST_USE_HOSTNAME. This IP address is a virtual IP address that is shared between the nodes in a cluster and will be active on one node at a time. This IP address will be different than the IP address used to protect the database hierarchy. Please note these IP addresses so they can be utilized when creating the SAP resources.

6.16.7.13. Creating an SAP Resource Hierarchy

To protect the SAP System, an SAP Hierarchy will be needed. This SAP Hierarchy consists of the Core (Central Services) Resource, the ERS Resource, the Primary Resource and Secondary Resources. To create this hierarchy, perform the following tasks from the Primary Server. **Note:** The below example is meant to be a guideline for creating your hierarchy. Tasks will vary somewhat depending upon your configuration.

Create the Core Resource

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy.**

   A dialog box will appear with a drop-down list box with all recognized recovery kits installed within the cluster. Select **SAP** from the drop-down listing.

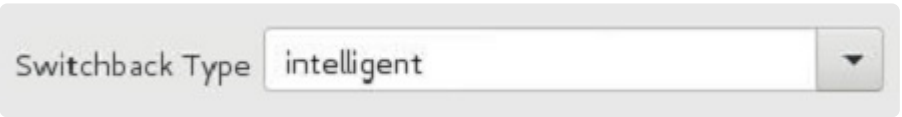
   

   Click **Next.**

   When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information.

   If you click **Cancel** at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

2. Select the **Switchback Type**. This dictates how the SAP instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either **intelligent** or **automatic**. **Intelligent switchback** requires administrative intervention to switch the instance back to the primary/original server. **Automatic switchback** means the switchback will occur as soon as the primary server comes back on line and re-establishes LifeKeeper communication paths.

   The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box.

   Click **Next**.

3. Select the Server where you want to place the SAP PAS, ASCS or SCS (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop-down list box.

Click **Next**

4. Select the **SAP SID**. This is the system identifier of the SAP PAS, ASCS or SCS system being protected.

Click **Next**.

5. Select the SAP Instance Name (ex. ASCS<No.>) (Core Instance first) for the SID being protected.

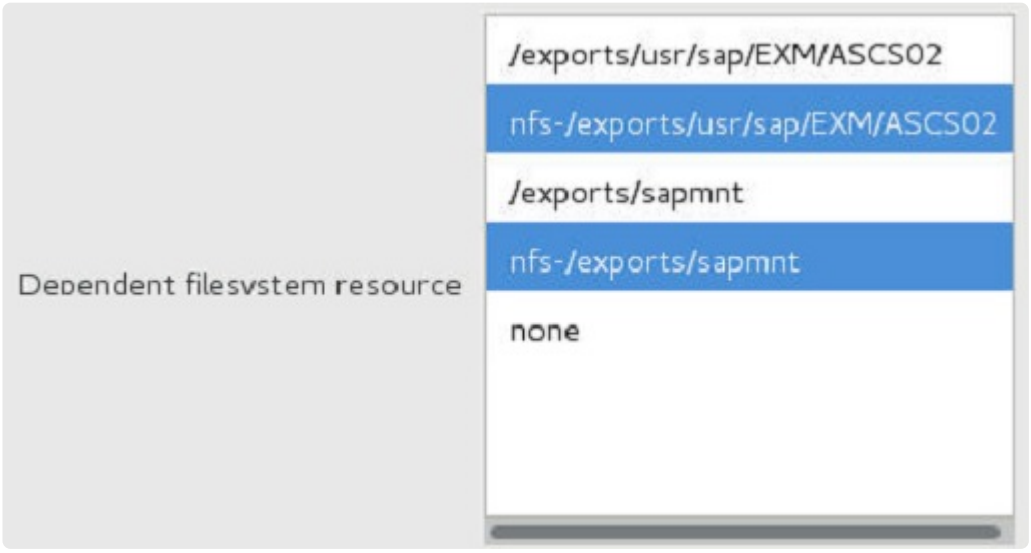
Note: Additional screens may appear related to customization of Protection and Recovery Levels.

6. Select the **IP Child Resource**. This is typically either the Virtual Host IP address noted during SAP installation (SAPINST_USE_HOSTNAME) or the IP address needed for failover.

a. Select whether LifeKeeper should attempt to **automate creation of dependent filesystems** for the instance. If **yes** is selected, LifeKeeper will attempt to create the necessary filesystem resources and add them as dependencies under the SAP resource. (Note that replicated filesystems cannot be created automatically.) If **no** is selected, the

following dialog box will prompt the user to select existing LifeKeeper filesystem resources to be added as dependencies.

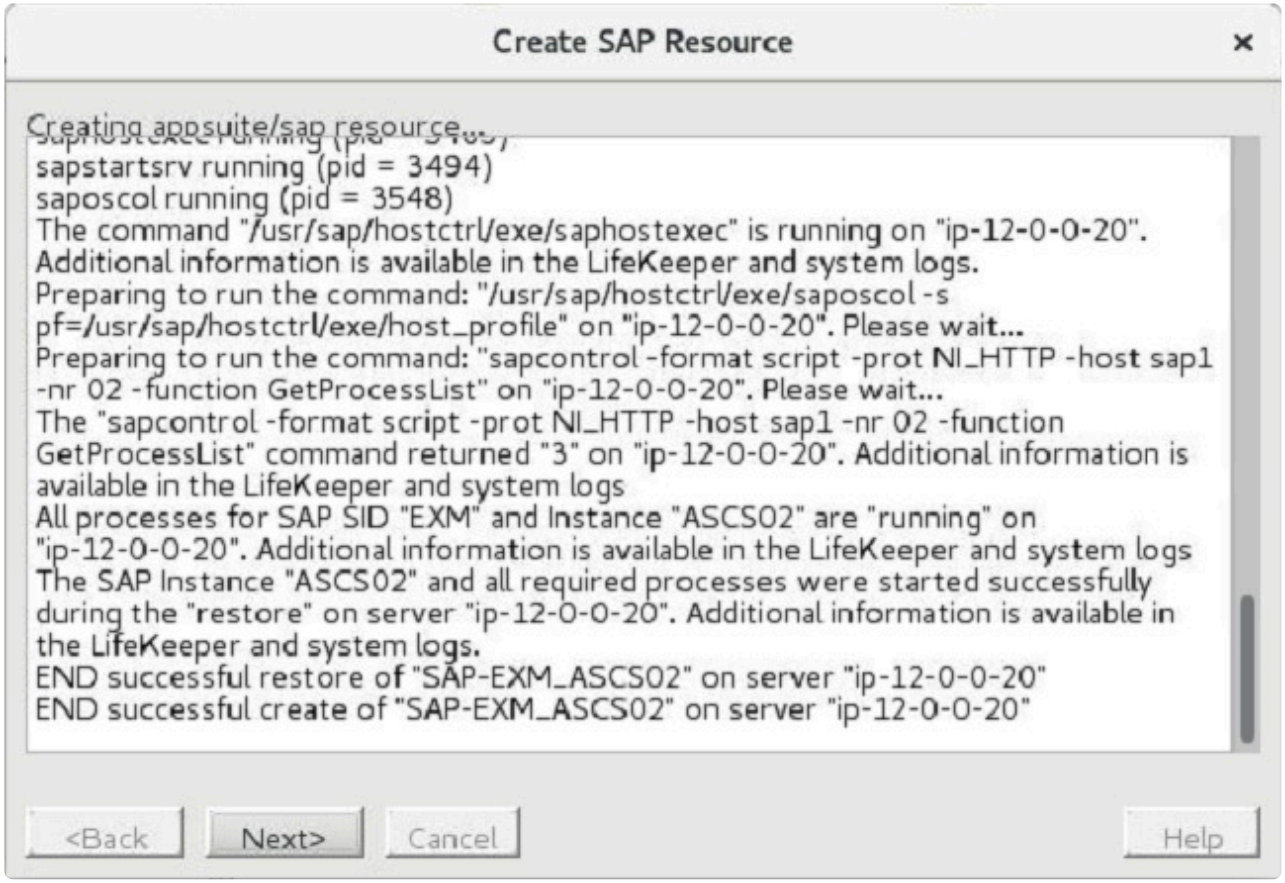
b. If **no** was chosen in the previous dialog, the option will be provided to select the filesystem resource(s) which should be added as a dependency under the SAP resource. Multiple resources can be selected by holding CTRL and clicking each resource separately. **Note:** Filesystem resources must be in-service (ISP) on this server in LifeKeeper in order to appear as choices in this dialog.



7. Select or enter the **SAP Tag**. This is a tag name that LifeKeeper gives to the SAP resource. You can select the default or enter your own tag name. The default tag is *SAP-<SID>_<INST>*.

When you click **Create**, the ***Create SAP Resource Wizard*** will create your SAP resource.

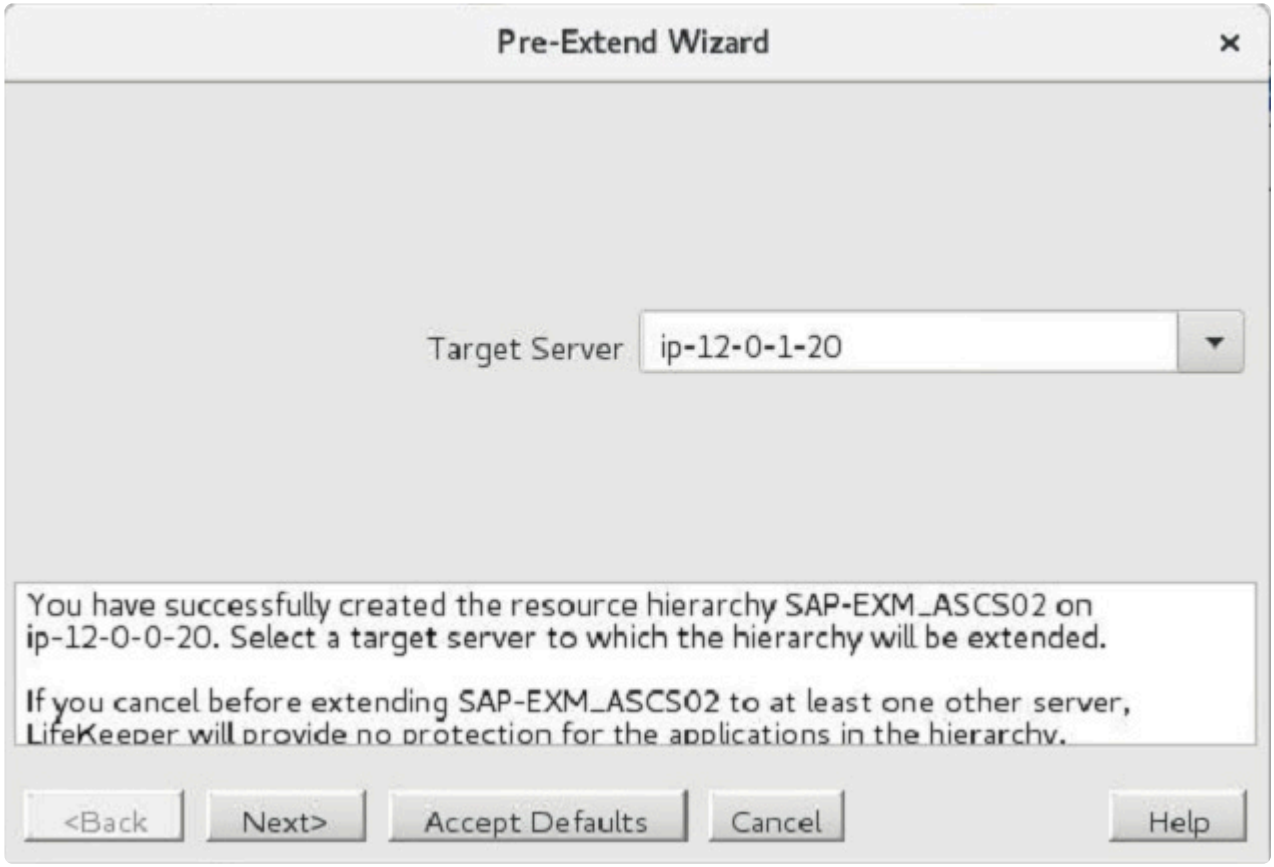
8. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your SAP resource hierarchy. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created. There may also be errors or messages output from the SAP startup scripts that are displayed in the information box.



Click **Next**

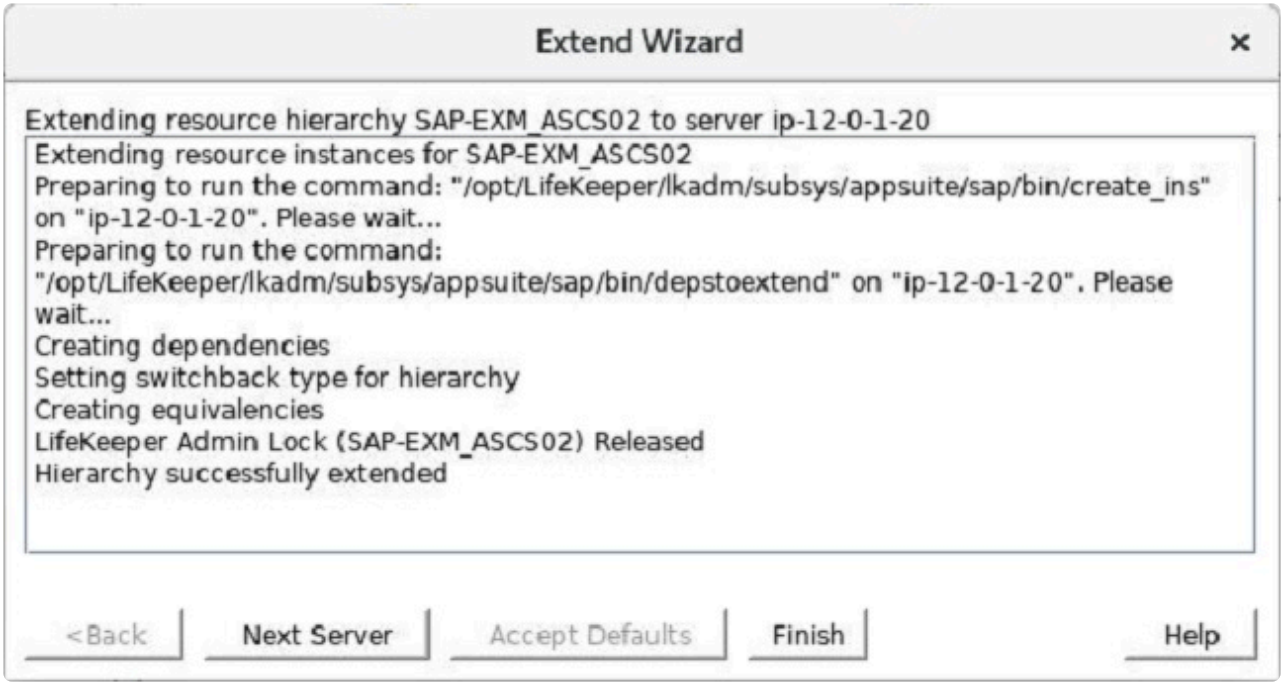
9. Another information box will appear explaining that you have successfully created an SAP resource hierarchy, and you must **Extend** that hierarchy to another server in your cluster in order to place it under LifeKeeper protection.

   When you click **Next**, LifeKeeper will launch the ***Pre-Extend Wizard*** that is explained later in this section.



If you click **Cancel** now, a dialog box will appear warning you that you will need to come back and extend your SAP resource hierarchy to another server at some other time to put it under LifeKeeper protection.

10. The ***Extend Wizard*** dialog will appear stating **Hierarchy successfully extended**. Click **Finish**.



11. The **Hierarchy Integrity Verification** dialog appears. Once Hiearchy Verification finishes, click **Done** to exit the **Create Resource Hierarchy** menu selection.

Hierarchy with the Core as the Top Level

Create the ERS Resource

The ERS resource provides additional protection against a single point of failure of a Core Instance (Central Services Instance) or enqueue server process. When a Core Instance (Central Services Instance) fails and is restarted, it will retrieve a backup copy of the enqueue lock table (i.e., the *replication table*) from the enqueue replication server. The result is that, in the event of the enqueue server failure, no transactions or updates are lost and the service for the SAP system continues.

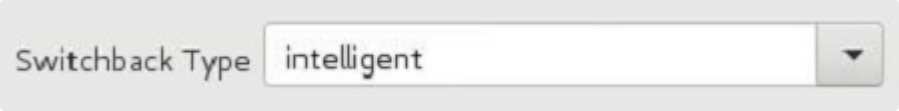
For a discussion of the differences in the implementation of ERS resources in SPS-L in versions 9.4.0 and later versus the implementation prior to version 9.4.0, see ERS Resource Types in LifeKeeper.

> **Important Note:** The creation and extension of the ERS hierarchy **must occur on a server where the corresponding ASCS/SCS instance is not in-service (ISP) in LifeKeeper**. In the case of ERSv2, the underlying ERS filesystem resources must first be brought in-service on the backup server where the resource will be created.

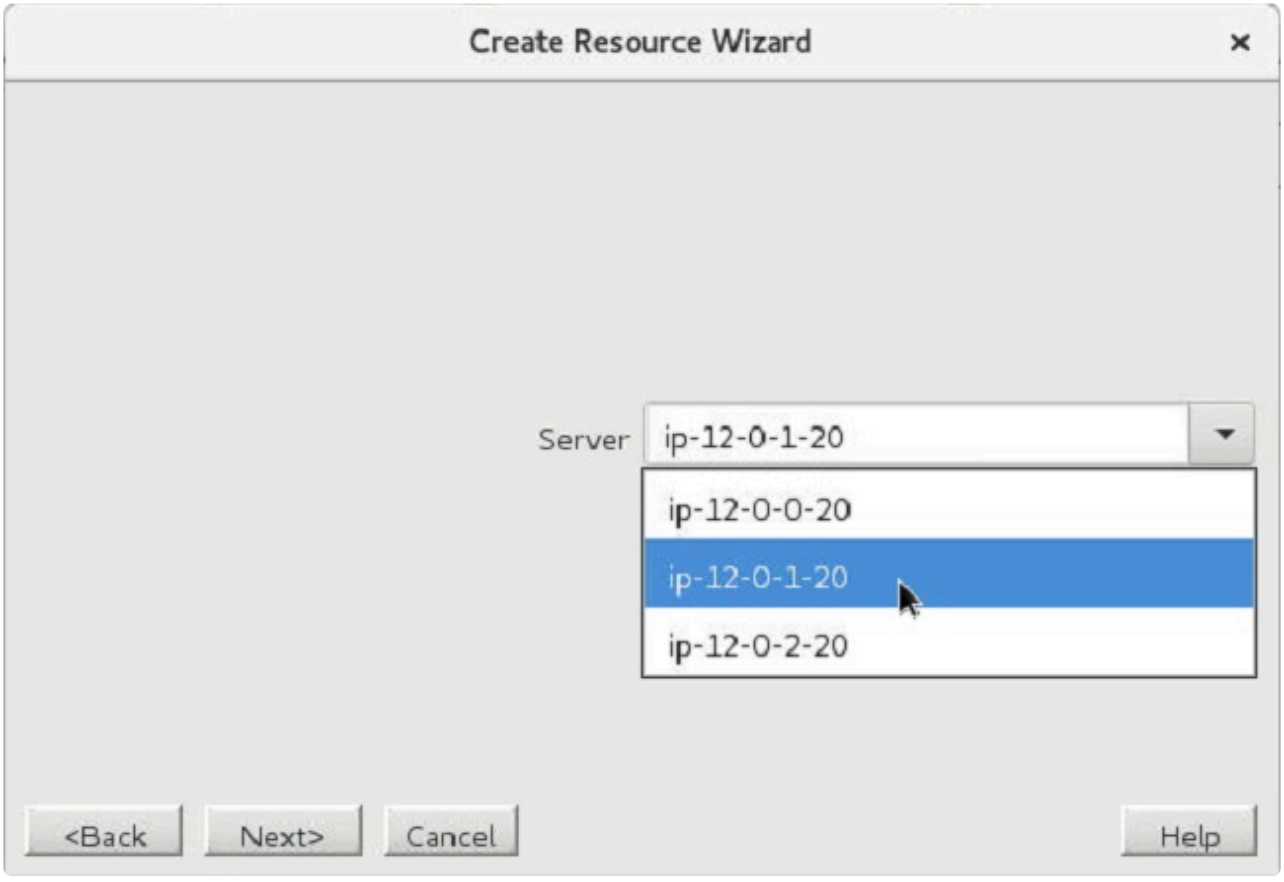
Perform the following steps to create the ERS Resource.

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy**. A dialog box will appear with a drop-down list box with all recognized recovery kits installed within the cluster. Select **SAP** from the drop-down listing. Click **Next**.

2. Select the **Switchback Type**. Click **Next**.



3. **Important:** Select a **Server** in the cluster where the corresponding ASCS/SCS instance is **not ISP**. Click **Next**.

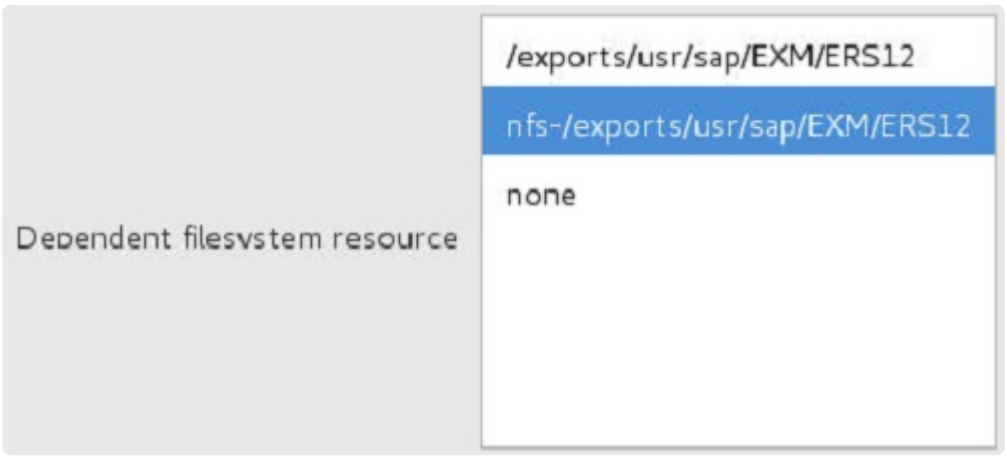


4. Select the **SAP SID** for the ERS instance. Click **Next**.

5. Select the **SAP Instance Name** (ex. ERS<No.>). Click **Next**.

6. If creating a resource to represent an ERSv2 instance, select the **IP Child Resource**. This choice will not appear when creating a resource to represent an ERSv1 instance.

7. If creating a resource to represent an ERSv2 instance, select the **Dependent Filesystem Resource** to be added as a dependency under the ERS resource. Note that filesystem resources must be in-service (ISP) on this server in LifeKeeper in order to appear on this list. This choice will not appear when creating a resource to represent an ERSv1 instance.



8. Select or enter the **SAP Tag**.

9. Follow prompts to **extend resource hierarchy**. **Note:** A resource representing an ERSv1 instance may only be extended to one backup server. The hierarchy extension will fail if attempting to extend an ERSv1 hierarchy to a third cluster node.

10. Once **Hierarchy Successfully Extended** displays, select **Finish**.

11. Select **Done**.

Separate ASCS and ERSv1 Hierarchies

Separate ASCS and ERSv2 Hierarchies

In the case where the ASCS instance is running ENSAv2, the ERS instance is running ERSv2, and the hierarchies have been extended to the same systems in a cluster with three or more nodes, the ASCS and ERS resource hierarchies can be forced to attempt to avoid each other on switchover and failover by using special “hierarchy avoidance” generic application resources. See Enforcing ASCS/ERS Avoidance Behavior When Using ENSAv2/ERSv2 for more details on how to create these resources and add them as dependencies in the ASCS and ERS hierarchies. **Note:** These hierarchy avoidance resources should not be used in an ENSAv1/ERSv1 configuration or in a two node cluster.

Create the Primary Application Server Resource

1. Again, for this same SAP SID, repeat the above steps to create the Primary Application Server Resource selecting **DVEBMGS{XX}** (where {XX} is the instance number) when prompted.

2. Select the **Level of Protection** when prompted (default is **FULL**). Click **Next**.

3. Select the **Level of Recovery** when promted (default is **FULL**). Click **Next**.

4. When prompted for **Dependent Instances**, select the “parent” instance, which would be the **ERS instance** created above.

5. Select the **IP Child Resource**.

6. Follow prompts to extend resource hierarchy.

7. Once **Hierarchy Successfully Extended** displays, select **Finish**.

8. Select **Done**.

Hierarchy with Primary Application Server as Top Level

Create the Secondary Application Server Resources

If necessary, create the Secondary Application Server Resources in the same manner as above.

Note: For command line instructions, see Setting Up SAP from the Command Line.

6.16.7.14. Deleting an SAP Resource Hierarchy

To delete a resource from all servers in your LifeKeeper configuration, complete the following steps.

Note: Each resource should be deleted separately in order to delete the entire hierarchy.

1. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.
2. Select the name of the **TargetServer** where you will be deleting your resource hierarchy.

   Note: If you selected the **Delete Resource** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next**.
3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete and highlight it.

   Note: This dialog will not appear if you selected the **Delete Resource** task by right-clicking on a resource instance in the left or right pane.

   Click **Next**.
4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete**.
5. Another information box appears confirming that the resource was deleted successfully. Click **Done** to exit.

6.16.7.15. Common SAP Recovery Kit Tasks

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View / Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

6.16.7.16. Setting Up SAP from the Command Line

You can set up the SAP Recovery Kit through the use of the command line.

Creating an SAP Resource from the Command Line

From the Primary Server, execute the following command:

```
$LKROOT/lkadm/subsys/appsuite/sap/bin/create <primary sys> <tag> <SAP SID> <SAP Instance> <switchback type> <IP Tag> <Protection Level> <Recovery Level> <Additional SAP Dependents>
```

Example:
$LKROOT/lkadm/subsys/appsuite/sap/bin/create liono SAP-STC_SCS00 STC SCS00 intelligent ip-sap10 Full Full none

Notes:

- **Switchback Type** – This dictates how the SAP instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either **Intelligent** or **Automatic**. Intelligent switchback requires administrative intervention to switch the instance back to the primary/original server. Automatic switchback means the switchback will occur as soon as the primary server comes back on line and re-establishes LifeKeeper communication paths.
- **IP Tag** – This represents the IP resource that will become a dependent of the SAP resource hierarchy.
- **Protection Level** – The **Protection Level** represents the actions that are allowed for each resource.
- **Recovery Level** – The **Recovery Level** provides instruction for the resource in the event of a failure.
- **Additional SAP Dependents** – This value represents the LifeKeeper SAP resource tag that will become a dependent of the current SAP resource being created.

Extending the SAP Resource from the Command Line

Extending the SAP Resource copies an existing hierarchy from one server and creates a similar hierarchy on another LifeKeeper server. To extend your resource via the command line, execute the following command:

```
system "$LKROOT/lkadm/bin/extmgrDoExtend.pl -p 1 -f, \"$tag\"
```

```
\”$backupnode\”
\”$priority\” \”$switchback\” \\\”$sapbundle\\\”“;
```

Example: Using a simple script for usability and ease.

```
#!/etc/default/LifeKeeper-perl
require “/etc/default/LifeKeeper.pl”;
my $lkroot=”$ENV{LKROOT}”;
my $tag=“SAP”;
my $backupnode=“snarf”;
my $switchback=“INTELLIGENT”;
my $priority=10;
$sapbundle = “\”$tag\”,\”$tag\””;
system “$lkroot/lkadm/bin/extmgrDoExtend.pl -p 1 -f,
\”$tag\” \”$backupnode\”
\”$priority\” \”$switchback\” \\\”$sapbundle\\\””;
```

6.16.7.17. Activating the SAP SIOS HA Cluster Connector (SSHCC)

The SAP SIOS HA Cluster Connector (SSHCC) provides an interface between the SAP Start Service (sapstartsrv) and LifeKeeper. While the HA Cluster Connector is active for an SAP instance, calls through sapcontrol which affect the state of the instance will be routed through LifeKeeper in order to keep the status of the resource in the cluster up-to-date.

In order to activate the SAP SIOS HA Cluster Connector for an SAP instance, follow these steps:

1. Identify the location of the active profile for the SAP instance. This can be found from the /usr/sap/sapservices file by looking for the line corresponding to the instance. For example, consider the following line corresponding to an ASCS instance:

```
LD_LIBRARY_PATH=/usr/sap/STC/ASCS00/exe:$LD_LIBRARY_PATH; export
LD_LIBRARY_PATH; /usr/sap/STC/ASCS00/exe/sapstartsrv pf=/usr/sap/STC/SYS/
profile/STC_ASCS00_sap1 -D -u stcadm
```

   In this example, the active ASCS instance profile is located at /usr/sap/STC/SYS/profile/STC_ASCS00_sap1.

2. Edit the instance profile found in step 1 and add the following lines to the bottom of the file:

```
#-----------------------------------------------------------------------#
```

- #

SAP SIOS High Availability Cluster Connector

```
#-----------------------------------------------------------------------#service/
halib = saphascriptco.soservice/halib_cluster_connector = /opt/LifeKeeper/lkadm/
subsys/appsuite/sap/bin/sap_sios_cluster_connectorservice/halib_debug_level = 1
```

3. Add the following line to the /etc/sudoers file on each system in the cluster to allow the SAP administrator user to run the sap_sios_cluster_connector script. Replace <sid> in the following line with the lower-case SAP SID for your system.

```
<sid>adm
```

ALL=NOPASSWD:/opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/sap_sios_cluster_connector-main

4. In order for the profile change to take effect, the sapstartsrv process for the instance must be restarted. This can be accomplished with the following command (replacing <sid> with the lower-case SAP SID and <SID> with the upper-case SAP SID):

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function RestartService <SID>"
```

5. To verify that the HA Cluster Connector was successfully activated for the SAP instance, run the following command:

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function HAGetFailoverConfig"
```

- If the HA Cluster Connector has been successfully configured for the instance, the output from this

command should show

HAActive: TRUE, as in the following sample output.

```
HAGetFailoverConfig
OK
HAActive: TRUE
HAProductVersion: "SIOS Protection Suite for Linux" steeleye-lk 9.3.2-6863
HASAPInterfaceVersion: "SIOS Protection Suite for Linux" steeleye-lkHACONNECTO
R-for-SAP 7.5.0-6855
HADocumentation: docs.us.sios.com/Linux/current/LK4L/SAP/index.htm
HAActiveNode: ip-12-0-0-20
HANodes: ip-12-0-0-20, ip-12-0-1-20, ip-12-0-2-20
```

- If the HA Cluster Connector did not initialize successfully, relevant error messages can be found in the sapstartsrv.log file for the instance, typically located at /usr/sap/<

SID>/<INST>/work/sapstartsrv.log.

6.16.7.18. SAP Test Preparation

1. Set up an SAP GUI on an SAP client to log in to SAP using the virtual SAP server name.

2. Set up an SAP GUI on an SAP client to log in to the redundant AS.

3. If desired, install additional AS's on other servers in the cluster and set up a login group among all application servers, excluding the PAS. For every AS installed, the profile file will have to be modified as previously described.

6.16.7.19. Perform SAP Tests

Perform the following series of tests. The test steps are different for each configuration. Some steps call for verifying that SAP is running correctly but do not call out specific tests to perform. For a list of possible tests to perform to verify that SAP is configured and running correctly, refer to the appendices of the SAP document, *SAP R/3 in Switchover Environments*.

Tests for Active/Active Configurations

1. When the SAP hierarchy is created, the SAP and DB will in-service on different servers. From an SAP GUI, log in to SAP. Verify that you can successfully log in and that SAP is running correctly.

2. Log out and re-log in through a redundant AS. Verify that you can successfully log in.

3. If you have set up a login group, verify that you can successfully log in through this group.

4. Using the LifeKeeper GUI, bring the SAP hierarchy in service on the SAP Backup Server. Both the SAP and DB will now be in service on the same server.

5. Again, verify that you can log in to SAP using the SAP virtual server name, a redundant AS and the login group. Verify that SAP is running correctly.

6. Using the LifeKeeper GUI, bring the DB hierarchy in service on the DB Backup Server. Each hierarchy will now be in service on its backup server.

7. Again, verify that you can log in to SAP using all login methods and that SAP is running correctly. If you execute transaction SM21, you should be able to see in the logs where the PAS lost then regained communication with the DB.

8. While logged in to SAP, shut down the SAP Backup server where SAP is currently in service by pushing the power supply switch. Verify that the SAP hierarchy comes in service on the SAP Primary Server, and that after the failover, you can again log in to the PAS, and that it is running correctly.

9. Restore power to the failed server. Using the LifeKeeper GUI, bring the DB hierarchy back in service on the DB Primary Server. Again, while logged in to SAP, shut down the DB Primary server where the DB is currently in service by pushing the power supply switch. Verify that the DB hierarchy comes in-service on the DB Backup Server and that, after the failover, you are still logged in to SAP and can execute transactions successfully.

10. Restore power to the failed server. Using the LifeKeeper GUI, bring the DB hierarchy back in service on the DB Primary Server.

Tests for Active/Standby Configurations

1. When the hierarchy is created, both the SAP and DB will be in service on the Primary Server. The redundant AS will be started on the Backup Server. From an SAP GUI, log in to SAP. Verify that you can successfully log in and that SAP is running correctly. Execute transaction SM51 to see the list of SAP servers. This list should include both the PAS or ASCS and AS.

2. Log out and re-log in through the redundant AS on the Backup Server. Verify that you can successfully log in.

3. If you have set up a login group, verify that you can successfully log in through this group.

4. Using the LifeKeeper GUI, bring the SAP/DB hierarchy in service on the Backup Server.

5. Again, verify that you can log in to SAP using the SAP virtual server name, a redundant AS and the login group. Verify that SAP is running correctly.

6. While logged in to SAP, shut down the SAP/DB Backup server where the hierarchy is currently in service by pushing the power supply switch. Verify that the SAP/DB hierarchy comes in service on the Primary Server, and after the failover, you can again log in to the PAS and that it is running correctly (you will lose your connection when the server goes down and will have to re-log in).

7. Restore power to the failed server. Again, while logged in to SAP, shut down the SAP/DB Primary server where the DB is currently in service by pushing the power supply switch. Verify that the SAP/DB hierarchy comes in service on the Backup Server and that, after the failover, you can again log in to SAP, and that it is running correctly.

8. Again, restore power to the failed server. Using the LifeKeeper GUI, bring the SAP/DB hierarchy in service on the Primary Server.

6.16.8. SAP Administration

This section provides tips and other information that may be helpful for administration and maintenance of certain configurations.

Oracle Database

6.16.8.1. NFS Considerations

As previously described in the Configuration Considerations topic, if the file system has been configured on either the PAS Primary or Backup server to locally mount NFS shares, an NFS hierarchy out-of-service operation will hang the system and prevent a clean reboot. To avoid causing your cluster to hang by inadvertently stopping the NFS server, we make the following recommendations:

- Do not take your NFS hierarchy out of service on a server that contains local NFS mount points to the protected NFS share. You may take your SAP resource in and out of service freely so long as the NFS child resources stay in service. You may also bring your NFS hierarchies in service on a different server prior to shutting a server down.

- If you must stop LifeKeeper on a server where the NFS hierarchy protecting locally mounted NFS shares is in service, always use the –f option. Stopping LifeKeeper using the command `lkstop -f` stops LifeKeeper without taking the hierarchies out of service, thereby preventing a server hang due to local NFS mounts. See the `lkstop` man page for additional information.

- If you must reboot a server where the NFS hierarchy protecting locally mounted NFS shares is in service, you should first stop LifeKeeper using the `-f` option as described above. A server reboot will cause the system to stop LifeKeeper without the `-f` option, thereby taking the NFS hierarchies out-of-service and hanging the system.

- If you need to uninstall the SAP package, do not do so when there are SAP hierarchies containing NFS resources that are in-service protected (ISP) on the server. Delete the SAP hierarchy prior to uninstalling the package.

- If you are upgrading SPS or if you need to run the SPS Installation setup scripts, it is recommended that you follow the upgrade instructions included in the SPS for Linux Installation Guide. This includes switching all applications away from the server to be upgraded before running the setup script on the SPS Installation image file and/or updating your SPS packages. Specifically, the setup script on the LifeKeeper Installation image file should not be run on a server where LifeKeeper is protecting active NFS shares, since upgrading the nfsd kernel module requires stopping NFS on that server which may cause the server to hang with locally mounted NFS file systems. For additional information, refer to the NFS Server Recovery Kit Documentation.

- Using TCP can lead to hangs during out-of-service operations during the forceumount call. When NFS shares are not accessible the unmount can fail. LifeKeeper will attempt to unmount the filesystem multiple times. These multiple attempts will typically succeed in eventually taking the resource out of service. However, this will cause delays in taking the resource out of service. To avoid these retries, use ‘nfsvers=3, proto=udp’ mount options.

> ! Note the usage of udp; this is important for failover and recovery.

- If the /sapmnt (or /sapmnt/<SID>) filesystem is shared via NFS,

'SAP_NFS_CHECK_DIRS=/sapmnt' should be added to /etc/default/LifeKeeper on each node in the cluster to help prevent hangs in SAP resource administration actions due to a loss of the NFS shares.

! SAP_NFS_CHECK_DIRS should not be used if the filesystems are being shared with EFS on AWS since the pingnfs check does not apply in that case.

6.16.8.2. SAP Client Reconnect

An SAP client can either be configured to log on to a specific SAP instance or a logon group. If configured to log on through a logon group, SAP determines which running instance the client actually connects to.

If the instance to which the client is connected goes down, the client connection is lost and the client must re-log on. If the database is temporarily lost, but the instance to which the client is connected stays up, the client will be temporarily unavailable until the database comes back up but the client does not have to re-log on.

For performance reasons, clients should log on to redundant Application instances and not the PAS, ASCS or SCS. Administrators may wish, however, to be able to log on to the PAS to view logs, etc. After protecting SAP with LifeKeeper, a client login can be configured using the virtual SAP server name so the client can log on regardless of whether the SAP Instance is active on the SAP Primary or Backup server.

6.16.8.3. Adjusting SAP Recovery Kit Tunable Values

Several of the SAP scripts have been written with a timeout feature to allow hung scripts to automatically kill themselves. This feature is required due to potential problems with unavailable NFS shares. This is explained in greater detail in the NFS Mounts and su topic. Each script equipped with this feature has a default timeout value in seconds that can be overridden if necessary. To reduce timeout wait times add required NFS file systems to 'SAP_NFS_CHECK_DIRS' as a comma-separated list. This enables ping to verify the file system is exported and if not it returns immediately.

`SAP_DEBUG` and `SAP_CREATE_NAS` can be enabled or disabled. The default for `SAP_DEBUG` is 0 (disabled). To enable, set this parameter to 1.

The default for `SAP_CREATE_NAS` is 1 (enabled). This is used for automatically including a NAS resource for NAS mounted file systems. To disable, set this parameter to 0.

The default for the tunable `SAP_CONFIG_REFRESH` is LKCHECKINTERVAL/2. The user can change the number of seconds between calls to refresh the properties panel for an SAP resource in the LifeKeeper GUI.

The table below shows the script names, variable names, and default values. To override a default value, simply add a line to the */etc/default/LifeKeeper* file with the desired value for that script. For example, to allow the remove script to run for a full minute before being killed, add the following line to */etc/default/LifeKeeper*:

```
SAP_REMOVE_TIMEOUT=60
```

Note: The script may actually run for slightly longer than the timeout value before being killed.

Note: It is not necessary to stop and restart LifeKeeper when changing these values.

Script Name	Variable Name	Default Value
remove	`SAP_REMOVE_TIMEOUT`	804 seconds
restore	`SAP_RESTORE_TIMEOUT`	516 seconds
recover	`SAP_RECOVER_TIMEOUT`	1320 seconds
quickCheck	`SAP_QUICKCHECK_TIMEOUT`	60 seconds
debug	`SAP_DEBUG`	0 (to enable, set to 1)
create NAS	`SAP_CREATE_NAS`	1 (to disable, set to 0)

GUI Properties Panel refresh	SAP_CONFIG_REFRESH	1/2 the value of LKCHECKINTERVAL
NFS shares to check	SAP_NFS_CHECK_DIRS	empty

Note: In a NetWeaver Java Only environment, if you choose to start the Java PAS in addition to the SCS Instance, you may need to increase the values for SAP_RESTORE_TIMEOUT and SAP_RECOVER_TIMEOUT.

6.16.8.4. Separation of SAP and NFS Hierarchies

Although the LifeKeeper SAP hierarchy described in this section implements the SAP NFS hierarchies as child dependencies to the SAP resource, it is possible to detach and maintain the NFS hierarchies separately after the SAP hierarchy is created. You should consider the advantages and disadvantages of maintaining these as separate hierarchies as described below prior to removing the dependency. Note that this is only possible if the NFS shares being detached are hosted on a logical volume (LUN) that is separate from other SAP filesystems.

To maintain these two hierarchies separately, simply create the SAP hierarchy as described in this documentation and then manually break the dependency between the SAP and NFS resources through the LifeKeeper GUI.

Advantage to maintaining SAP and NFS hierarchies separately: If there is a problem with NFS, the NFS hierarchy can fail over separately from SAP. In this situation, as long as SAP handles the temporary loss of NFS mounted directories transparently, an SAP failover will not occur. The NFS mounts need to be in both /etc/mtab and listed in the tunable value SAP_NFS_CHECK_DIRS in /etc/default/LifeKeeper in order to avoid LifeKeeper hanging while the NFS shares are unavailable.

Disadvantage to maintaining SAP and NFS hierarchies separately: NFS shares are not guaranteed to be hosted on the same server where the PAS, ASCS, SCS or ERS instance is running. **Note**: Consult your *SAP Installation Guide* for SAP's recommendations.

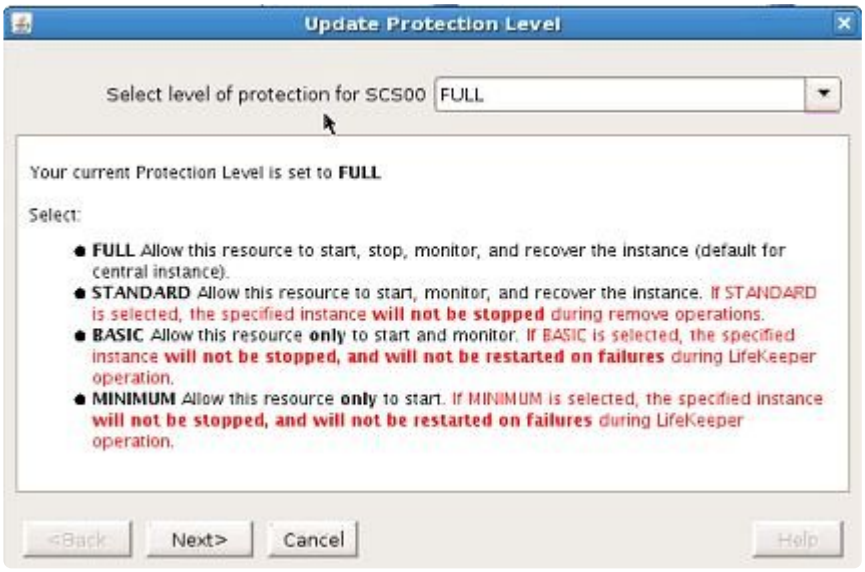
The diagram below shows the SAP and NFS hierarchies after the dependency has been deleted.

6.16.8.5. Update SAP Protection Level

The **Protection Level** represents the actions that are allowed for each resource. To find out what your current protection level is or to change this option, go to **Update Protection Level**. The level of protection can be set to **FULL**, **STANDARD**, **BASIC** or **MINIMUM**.

1. Right-click your instance.
2. Select **Update Protection Level**.

3. The following screen will appear, prompting you to select the Level of Protection.



FULL. This is the default level which provides full protection, allowing the instance to be started, stopped, monitored and recovered.

STANDARD. Selecting this level will allow the resource to start, monitor and recover the instance, but it will not be stopped during a remove operation.

BASIC. Selecting this level will allow the resource to start and monitor only. It will not be stopped or restarted on failures.

MINIMUM. Selecting this level will only allow the resource to start the instance. It will not be stopped or restarted on failures.

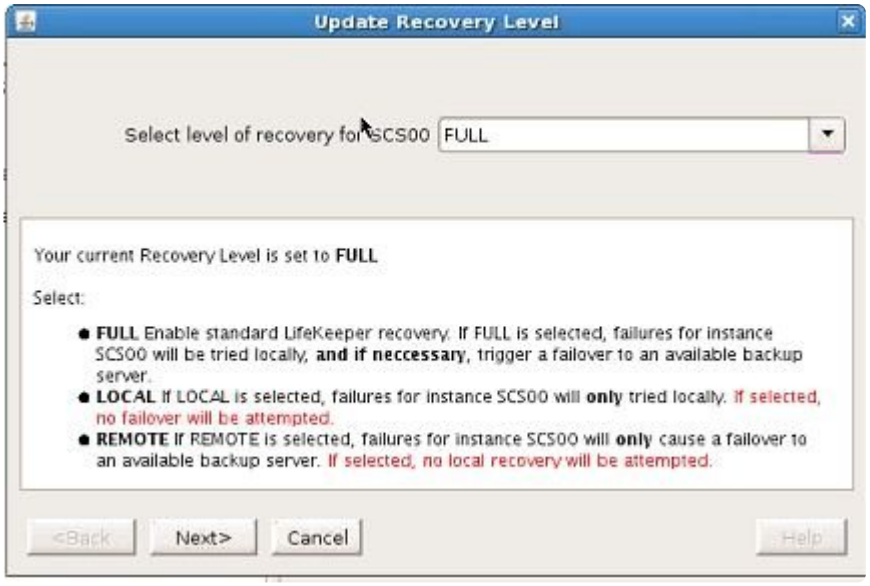
Note: The **BASIC** and **MINIMUM** Protection Levels are for placing the LifeKeeper protected application in a temporary maintenance mode. The use of **BASIC** or **MINIMUM** as an ongoing state for the Protection Level of a resource is not recommended. See Hierarchy Remove Errors in the Troubleshooting Section for further information.

6.16.8.6. Update SAP Recovery Level

The **Recovery Level** provides instruction for the resource in the event of a failure. To find out what your current recovery level is or to change this option, go to **Update Recovery Level**. The recovery level can be set to **FULL**, **LOCAL** or **REMOTE**.

1. Right-click your instance.
2. Select **Update Recovery Level**.

3. The following screen will appear, prompting you to select the **Level of Recovery**.



FULL. When recovery level is set to **FULL**, the resource will try to recover locally. If that fails, it will try to recover remotely until it is successful.

LOCAL. When recovery level is set to **LOCAL**, the resource will only try to restart locally; it will not fail over.

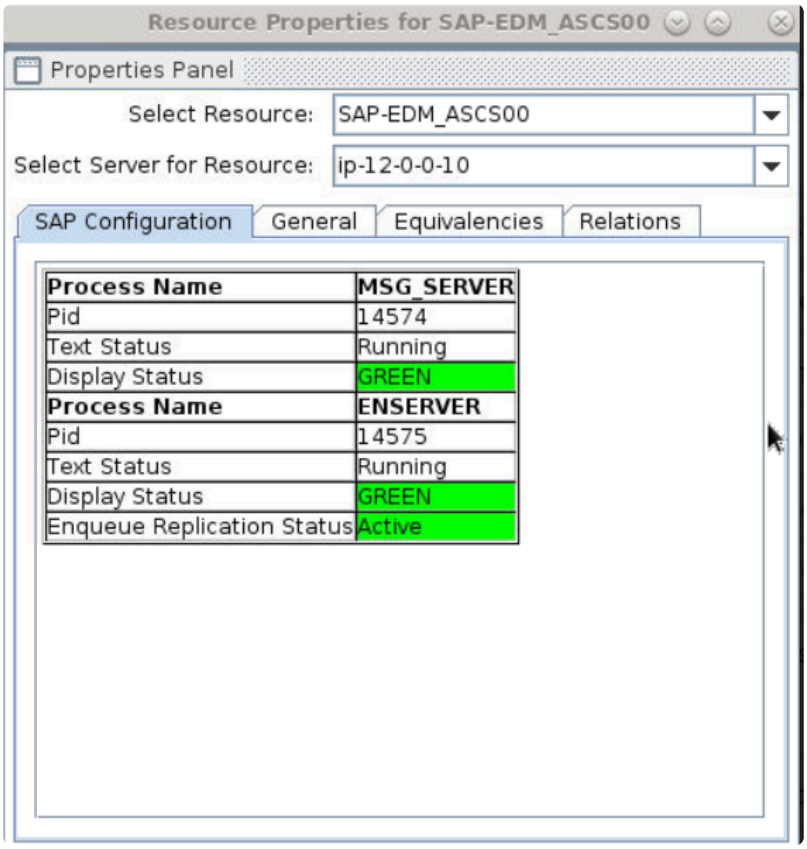
REMOTE. When recovery level is set to **REMOTE**, the resource will only try to restart remotely. It will not attempt to restart locally first.

6.16.8.7. View SAP Properties

The **Resource Properties** page allows you to view the configuration details for a specific SAP resource. To view the properties of a resource on a specific server or display the status of SAP processes, view the **Properties** Screen:

1. Right-click your instance.
2. Select **Properties**.

3. The following **Properties** screen will appear.



The resulting **Properties** page contains four tabs. The first of those tabs, labeled SAP Configuration, contains configuration information that is specific to SAP resources. The remaining three tabs are available for all LifeKeeper resource types.

6.16.8.8. Special Considerations for Oracle

Once the SAP processes are functioning on the systems in the LifeKeeper cluster, resources will need to be created in LifeKeeper for the major SAP functions. These include the ASCS system, the DVEBMSG system, the SCS system and the Oracle database.

This topic will discuss some special considerations for protecting Oracle in a LifeKeeper environment.

- Make sure that the LifeKeeper for Linux Oracle Application Recovery Kit is installed.
- Consult the Oracle Recovery Kit documentation.
- During the installation of SAP, the SAPinst process normally assumes that the database software has already been installed and configured. However, if Oracle is the database to be used with SAP, the SAPinst process will prompt the installer to start the Oracle installation tool (RUNINSTALLER) and complete the Oracle install.
- While installing Oracle during the installation of SAP, an Oracle SID was created. This SID is needed by the Oracle Recovery Kit, so be prepared to supply it when creating the Oracle resource in LifeKeeper.
- When creating a standard SAP installation with Oracle, thirteen separate file systems are created that the Oracle instance will use. Commonly, each of these file systems is built on top of an LVM logical volume and each may contain many separate physical volumes. For LifeKeeper to properly represent these file systems, a separate resource is created for each physical and logical volume and volume group. Since this large collection of resources needs to be assembled into a LifeKeeper hierarchy, it may take some time to complete the creation and extension of the Oracle hierarchy. Do not be surprised if it takes at least an hour for the creation process to complete, and another 10 to 20 minutes for the extension to complete.
- Building the necessary Oracle (and SAP) file systems on top of LVM is not required, and the SAP and Oracle recovery kits in LifeKeeper will work fine with standard Linux file systems.
- The LifeKeeper Oracle Recovery Kit can identify ten of the thirteen file systems the Oracle SAP installation uses as standard Oracle dependencies, and the kit will automatically create dependencies in the hierarchy for these file systems. The Oracle Recovery Kit does not recognize the *saptrace*, *sapreorg* and *saparch* file systems automatically. The administrator setting up LifeKeeper will need to manually create resource dependencies for these additional file systems.

6.16.8.9. SSHCC HA Actions

The SAP SIOS HA Cluster Connector (SSHCC) Actions provide a list of advanced configuration operations that work in conjunction with the SAP SIOS HA Cluster Connector. The following advanced configuration operations are available:

- **Start Instance** – performs an SAP SIOS HA Cluster Connector start action on the specified resource tag on the current node
- **Stop Instance** – performs an SAP SIOS HA Cluster Connector stop action on the specified resource tag on the current node
- **Migrate Instance** – performs an SAP SIOS HA Cluster Connector migrate action on the specified resource tag on the current node
- **Maintenance Mode** – performs an SAP SIOS HA Cluster Connector maintenance mode action on the specified resource tag on all cluster nodes

Note: These advanced configuration operations should not be used as a replacement for the standard SIOS Protection Suite for Linux in-service or out-of-service operations.

To perform these operations:

1. Right click on your instance
2. Select SSHCC HA Actions from the available menu
3. Select the desired operation from the provided choices
4. Confirm the selected SSHCC HA operation for the specified tag
5. Select Update to continue with the selected operation

6.16.8.10. ERS Resource Types in LifeKeeper

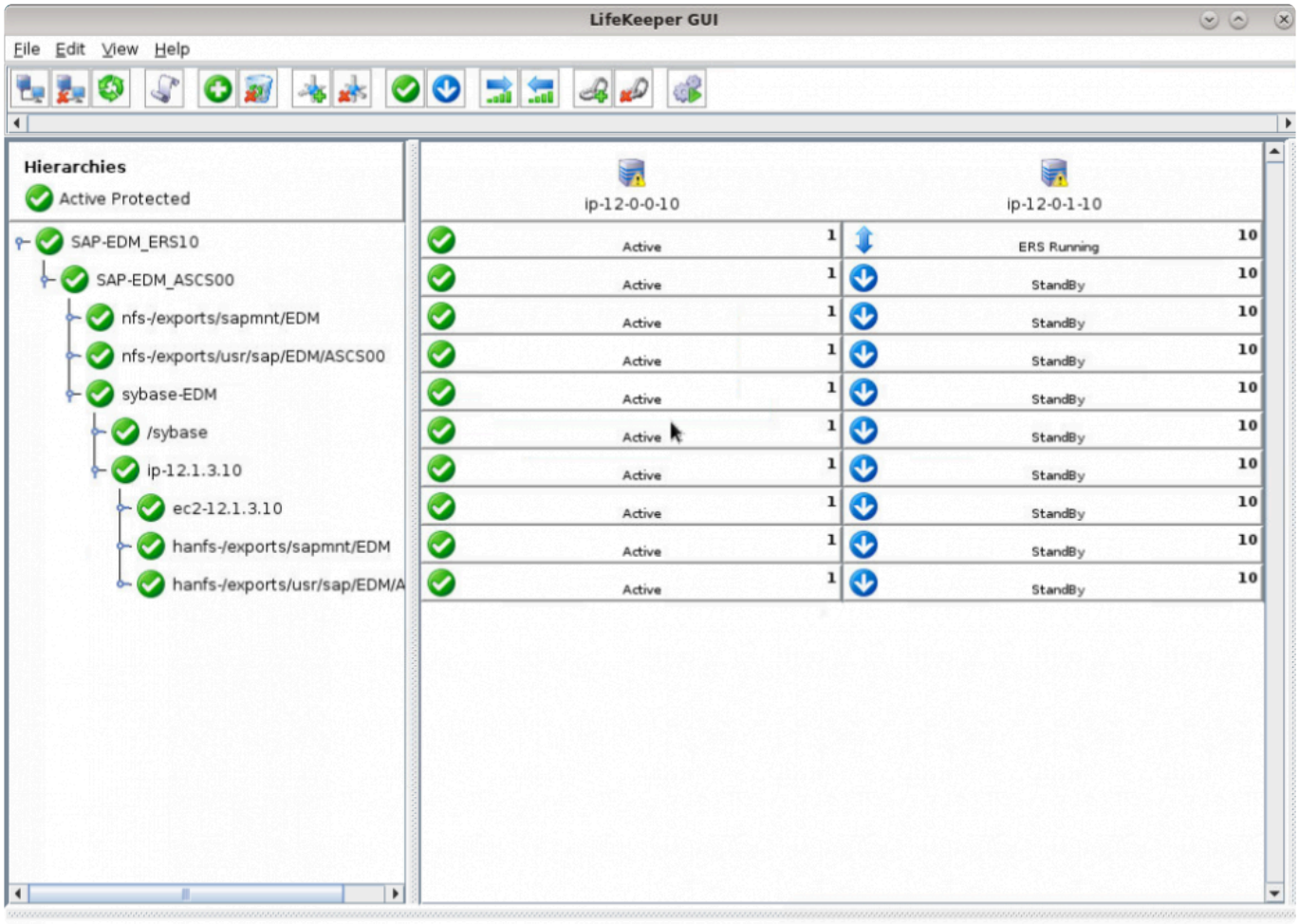
Important: Please see the SAP Recovery Kit – Known Issues / Restrictions

The design and implementation of the ERS resource type was modified in SPS-L 9.4.0. This page describes the differences between the implementation prior to version 9.4.0 and the implementation in versions 9.4.0 and later, how to determine which version exists on your system, and how to upgrade to the new resource type.

ERS Resources Prior to SPS-L 9.4.0

In versions of SPS-L prior to 9.4.0, the ERS resource was designed to sit at the top of an SAP hierarchy with a dependency on the Central Services (ASCS/SCS) resource that it provides lock table redundancy for.

The behavior of this resource was designed such that it would start the ERS instance on the backup node (where the ERS resource was listed as Standby in the LifeKeeper GUI). Upon switchover or failover of the SAP hierarchy, the ASCS/SCS instance would be started on the backup node and would obtain the backup copy of the lock table (i.e., the replication table) from shared memory on that system. Once the enqueue server successfully obtained the lock table, it would send a signal to notify the ERS instance to terminate itself. Once the ERS resource became Active (ISP) in LifeKeeper on the backup node, the ERS instance would be started on the original primary node when it became available. At that point the replication server would reconnect to the enqueue server and resume lock table replication. When the ERS instance is running on the backup node the LifeKeeper GUI status will change from ‘StandBy’ to ‘ERS Running’.



ERS Resources in SPS-L 9.4.0 and Later

In SPS-L 9.4.0 and later, the ERS resource was redesigned to operate in its own independent hierarchy.

ERSv1 Resource in an Independent Hierarchy

ERSv2 Resource in an Independent Hierarchy with Virtual IP and Highly Available Dependent Filesystem

This newer ERS resource design supports ERSv1 instances in two-node clusters and ERSv2 instances in clusters with any number of nodes. When using this resource type, the ERS instance will be started on the same node where the LifeKeeper is currently Active (ISP). The design change was made to facilitate the ability for an ERSv2 hierarchy (including its dependent virtual IP and filesystem resources) to failover independently of its corresponding Central Services resource hierarchy.

In order to attempt to keep the ERS resource from being Active (ISP) on the same node as its corresponding Central Services resource, this newer ERS resource type will check during each quickCheck interval (default: two minutes) whether:

1. The ERS resource is Active on the same node as its corresponding Central Services resource,
2. The lock table replication is in-sync between the enqueue server and the replication server, and
3. There is a different node available in the cluster that all resources in the ERS hierarchy could successfully relocate to.

If all three of these conditions are met, LifeKeeper will automatically relocate the ERS hierarchy to a different cluster node in order to provide redundancy of the enqueue server lock table data across cluster nodes. This automatic relocation behavior can be disabled by setting the flag ‘sap_no_ers_relocation_<ERS Tag>’ in LifeKeeper. This can be accomplished with a command similar to the following (where SAP-EXM_ERS12 is the example tag for the ERS resource):

```
/opt/LifeKeeper/bin/flg_create -f
"sap_no_ers_relocation_SAP-EXM_ERS12"
```

Creating this flag will not disable failover due to a failed local recovery of the ERS instance.

Note when using NFS: Since this design eliminates the LifeKeeper dependency of the ERS resource on the sapmnt filesystem, it is important to make appropriate use of the tunable value SAP_NFS_CHECK_DIRS to help prevent action scripts from hanging due to the sapmnt NFS share being inaccessible. See NFS Considerations for more details.

Which ERS Resource Type Do I Have in My Hierarchy?

If you are not sure in which version of LifeKeeper your existing ERS resource was created, run the following command (replacing <ERS Tag> with the tag name of your ERS resource):

```
/opt/LifeKeeper/bin/ins_list -f: -t <ERS Tag> | cut -d: -f6
```

The output of this command should be similar to:

EXMERS12sap2/sapmntFULLFULL5

- If the last digit in the output is 2, then this resource was created in SPS-L 9.3.2 or earlier. This type of resource may only be used to represent an ERSv1 instance in a two-node cluster and should sit at the top of the SAP hierarchy with a dependency on its corresponding Central Services (ASCS/SCS) instance below it. See the **Upgrading the ERS Resource Type** section below for instructions on how to switch to the newer design that operates in an independent hierarchy.

- If the last digit in the output is 5, then this resource was created in SPS-L 9.4.0 or later. This type of resource can be used to represent either an ERSv1 instance in a two-node cluster or an ERSv2 instance in a cluster with any number of nodes. It should operate in a hierarchy independent of its corresponding Central Services resource.

Upgrading the ERS Resource Type

To upgrade to the newer ERS resource type in SPS-L 9.4.0 or later, complete the following steps.

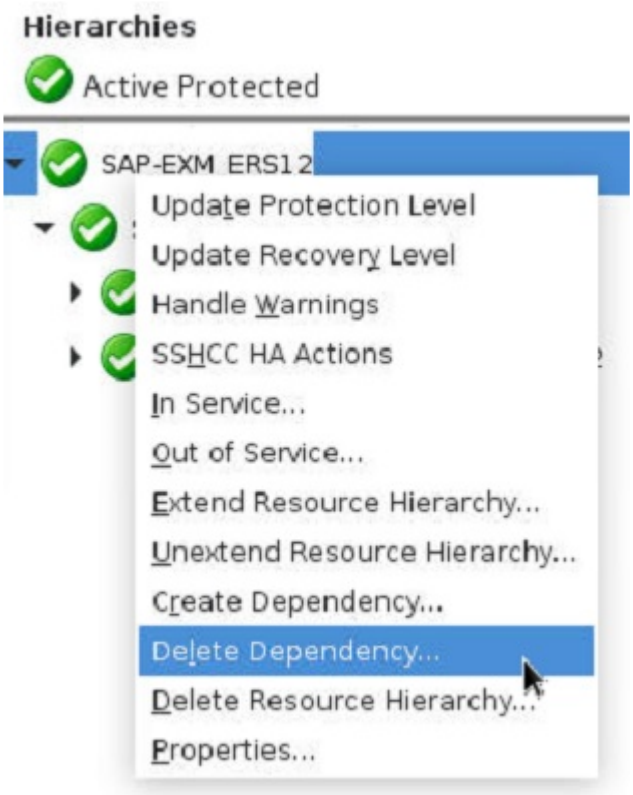
1. Before attempting the upgrade process, create a backup of your LifeKeeper hierarchies on all cluster nodes by running the command:

```
/opt/LifeKeeper/bin/lkbackup -c --cluster
```

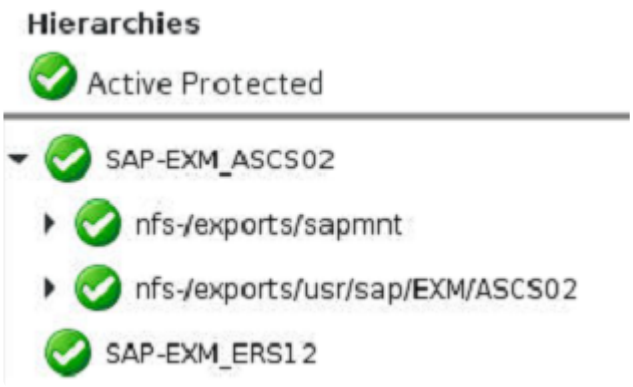
   - If you make a mistake at any point, the saved hierarchy configuration can be restored by stopping LifeKeeper on all nodes and running the command:

```
/opt/LifeKeeper/bin/lkbackup -x --cluster
```

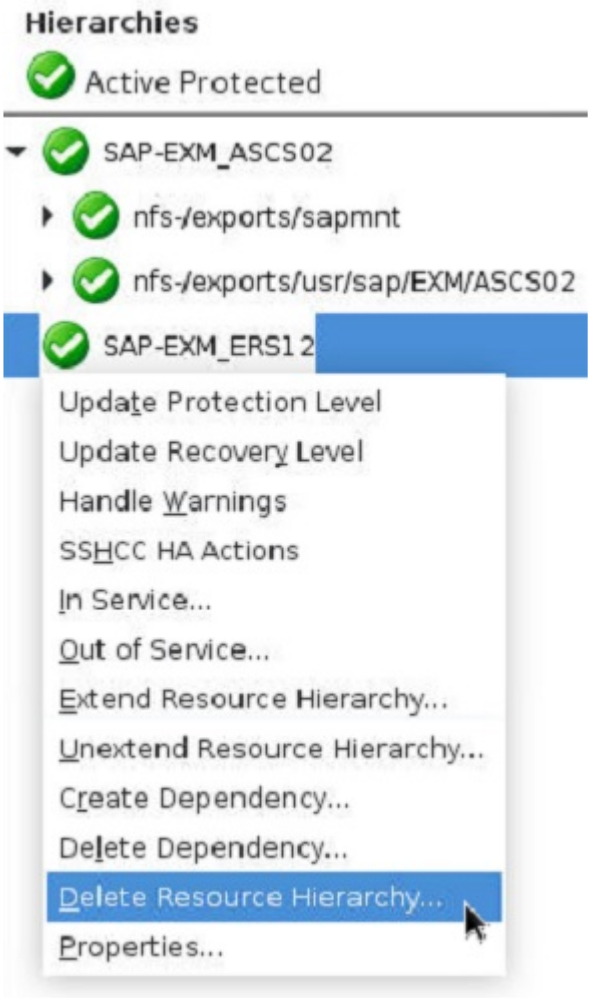
2. Right-click the ERS resource in the LifeKeeper hierarchy panel and select **Delete Dependency…**



3. Delete all dependencies of the ERS resource. You may need to select **Delete Dependency…** multiple times in order to delete all dependencies. When you are finished, the ERS resource should exist in a hierarchy by itself with no child dependencies and no other resources dependent on it, as shown in the following image.



4. Right-click the ERS resource in the LifeKeeper hierarchy panel and select **Delete Resource Hierarchy…** Select any server as the **Target Server** and click **Next**. Click **Delete** to delete the ERS resource on all nodes. **Warning:** If you did not successfully delete all dependencies between the ERS resource and other resources in the SAP hierarchy in the previous step, then this step **could delete your entire SAP hierarchy**.



5. Once the ERS resource has been successfully deleted on all nodes, follow the instructions in the “Create the ERS Resource” section in SAP Installation → Creating an SAP Hierarchy to create a new ERS resource and extend it to the desired cluster nodes.

6.16.8.11. Upgrading from ENSAv1 to ENSAv2

In order to upgrade from Standalone Enqueue Server version 1 to Standalone Enqueue Server version 2, first ensure that your SAP kernel version supports ENSAv2 then complete the following steps:

1. Set the following parameters in the default profile (typically located at /usr/sap/<SID>/SYS/profile/DEFAULT.PFL). These parameters must be the same for all instances:

   - enq/enable=TRUE
     enq/serverhost=<ASCS instance host>
     enq/serverinst=<ASCS instance number>
     enque/process_location=REMOTESA

2. In the ASCS instance profile (typically located at /usr/sap/<SID>/SYS/profile/<SID>_ASCS<No>_<VIP>), set the following parameters:

   - _ENQ = enq.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
     Execute_01 = local rm -f $(_ENQ)
     Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enq_server$(FT_EXE)
     Start_Program_01 = local $(_ENQ) pf=$(_PF)

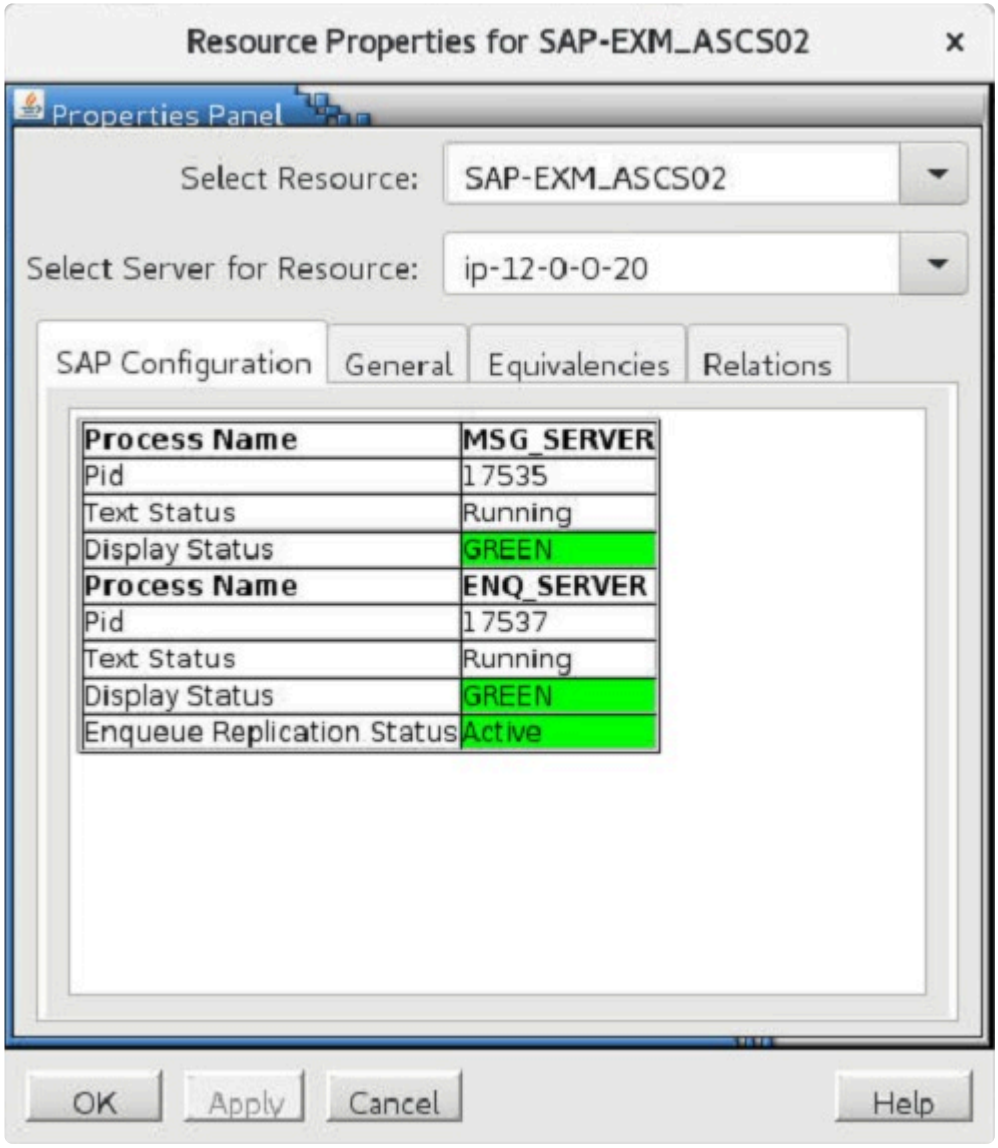
   Note: The number used for the Execute_* and the Start_Program_* parameters should be the first number not yet used for that parameter in this profile.

3. After setting the parameters in the default and ASCS instance profiles, restart the SAP Start Service for the ASCS instance by running the following command (replacing <sid> with your lower-case SAP SID and <SID> with your upper-case SAP SID):

   - su - <sid>adm -c “sapcontrol -nr <ASCS Inst#> -function RestartService <SID>”

4. **On all cluster nodes that the ASCS resource has been extended to**, edit the file /opt/LifeKeeper/subsys/appsuite/resources/sap/INFO_<ASCS Tag> and ensure that SAPENQ_VERSION=2. This may require you to add the line “SAPENQ_VERSION=2” if the INFO file does not yet contain a value for SAPENQ_VERSION.

5. Restart the SAP system. Once the ASCS instance is restarted, it will be using the enq_server (ENSAv2) process instead of the enserver (ENSAv1) process. This can be verified by right-clicking the ASCS resource in LifeKeeper and selecting **Properties…**

- 

Note: The Enqueue Replication Status on your system will not show Active until you have also completed the upgrade from ERSv1 to ERSv2 for the corresponding ERS resource. See Upgrading from ERSv1 to ERSv2 for more information.

6.16.8.12. Upgrading from ERSv1 to ERSv2

In order to upgrade from Enqueue Replication Server version 1 to Enqueue Replication Server version 2, first ensure that your SAP kernel version supports ERSv2 then complete the following steps:

1. Upgrade the ASCS instance to use ENSAv2 by following the instructions on Upgrading from ENSAv1 to ENSAv2. The same version of the Standalone Enqueue Server and Enqueue Replication Servers must be used since mixed version configurations are not supported by SAP. See the SAP documentation on ENSAv2/ERSv2 for more details.

2. Set up the virtual IP and shared file system for the ERS instance. Also create the corresponding virtual IP and filesystem resource hierarchies in LifeKeeper. They will be used in step 9 during the recreation of the ERS resource. **Note:** These LifeKeeper resources should be Active (ISP) on a node where the corresponding ASCS instance is currently Standby (OSU).

3. Set the following parameters in the default profile (typically located at /usr/sap/<SID>/SYS/profile/DEFAULT.PFL). These parameters must be the same for all instances:

   - enq/enable=TRUE
     enq/serverhost=<
   ASCS instance host>
   enq/serverinst=<ASCS instance number>
   enq/replicatorhost=<ERS instance host>
   enq/replicatorinst=<ERS instance number>
   enque/process_location=REMOTESA

4. In the ASCS instance profile (typically located at /usr/sap/<SID>/SYS/profile/<SID>_ASCS<No>_<VIP>), set the following parameters:

   - enq/server/replication/enable = true
     _ENQ = enq.sap$(
   SAPSYSTEMNAME)_$(INSTANCE_NAME)
   Execute_01 = local rm -f $(_ENQ)
   Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enq_server$(FT_EXE) $(_ENQ)
   Start_Program_01 = local $(_ENQ) pf=$(_PF)

Note: The number used for the Execute_* and the Start_Program_* parameters should be the first number not yet used for that parameter in this profile.

5. In the ERS instance profile (typically located at /usr/sap/<SID>/SYS/profile/<SID>_ERS<No>_<VIP>), set the following parameters:

   - _ENQR = enqr.sap$(

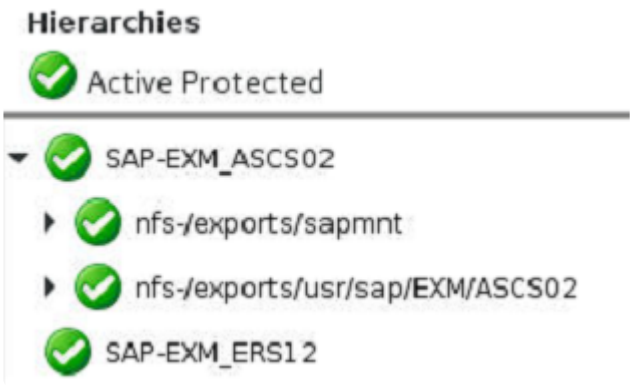
SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_01 = local rm -f $(_ENQR)
Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enq_replicator$(FT_EXE) $(_ENQR)
Start_Program_00 = local $(_ENQR) pf=$(_PF)

Note: The number used for the Execute_* and the Start_Program_* parameters should be the first number not yet used for that parameter in this profile.

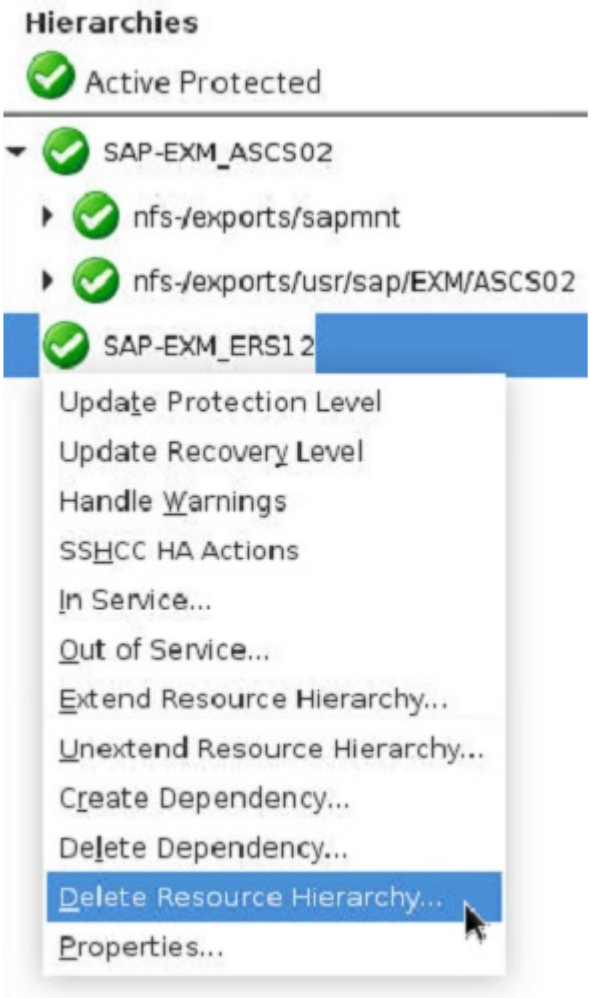
6. After setting the parameters in the default, ASCS, and ERS instance profiles, restart the SAP Start Service for the ASCS and ERS instances by running the following commands (replacing <sid> with your lower-case SAP SID and <SID> with your upper-case SAP SID):

   - su – <sid >adm -c “sapcontrol -host <
   ASCS VIP> -nr <ASCS Inst#> -function RestartService <SID>”
   su – <sid >adm -c “sapcontrol -host <ERS VIP> -nr <ERS Inst#> -function RestartService <SID>”

7. If you do not have an existing ERS resource in LifeKeeper that needs to be upgraded, skip to step 9. Otherwise, delete all dependencies of the ERS resource in LifeKeeper by right-clicking the resource and selecting **Delete Dependency…**. You may need to select **Delete Dependency…** multiple times in order to delete all dependencies. When you are finished, the ERS resource should exist in a hierarchy by itself with no child dependencies and no other resources dependent on it, as shown in the following image.

   - 

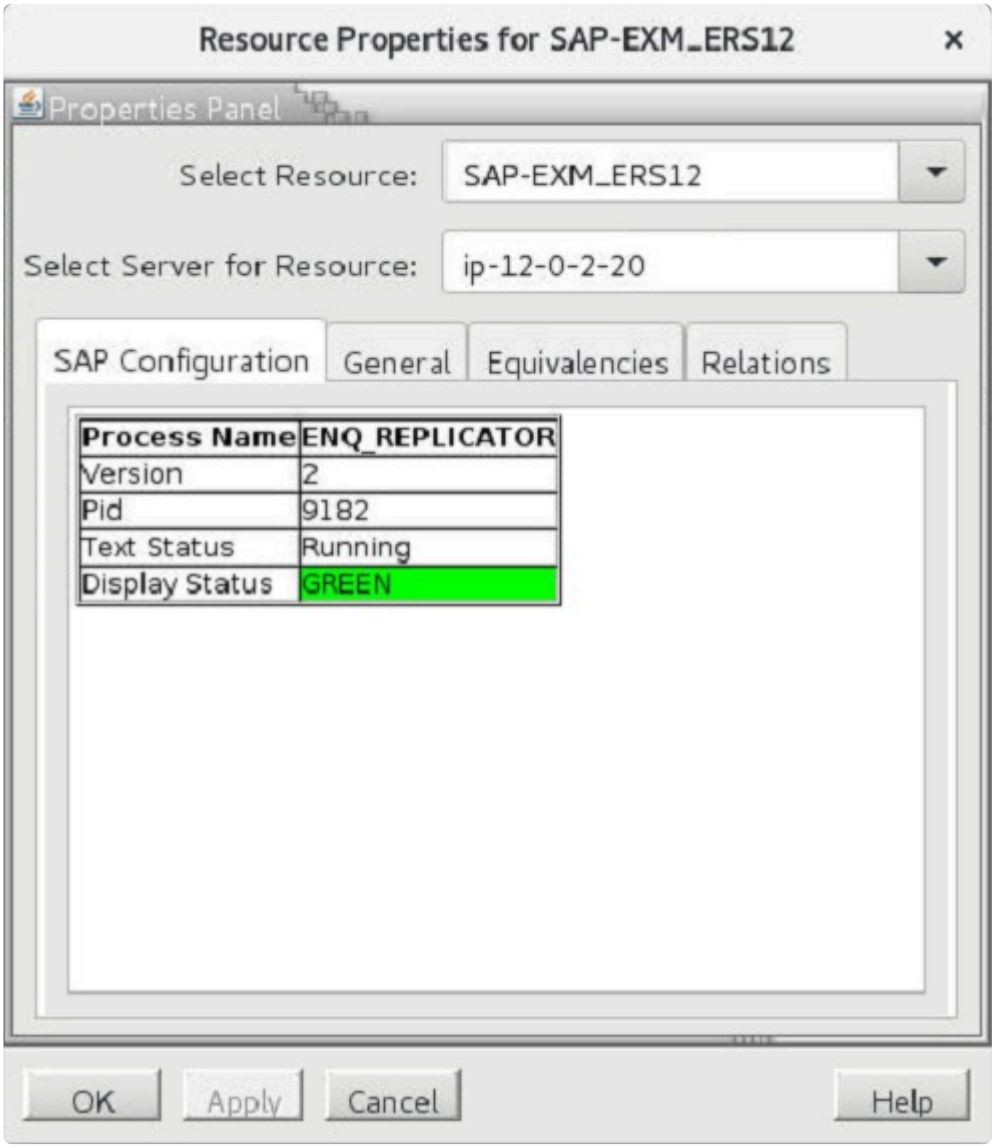
8. Right-click the ERS resource in the LifeKeeper hierarchy panel and select **Delete Resource Hierarchy…**. Select any server as the **Target Server** and click **Next**. Click **Delete** to delete the ERS resource on all nodes. **Warning:** If you did not successfully delete all dependencies between the ERS resource and other resources in the SAP hierarchy in the previous step, then this step **could delete your entire SAP hierarchy**.



9. Once the ERS resource has been successfully deleted on all nodes, follow the instructions in the “Create the ERS Resource” section in SAP Installation → Creating an SAP Hierarchy to create a new ERS resource and extend it to the desired cluster nodes. During the ERS resource creation, select the virtual IP and dependent filesystem resources created in step 2, if applicable. If selected, these resources will be automatically added as child dependencies in the ERS hierarchy.

10. **On all cluster nodes that the ERS resource has been extended to**, edit the file /opt/LifeKeeper/subsys/appsuite/resources/sap/INFO_<ERS Tag> and ensure that SAPENQREP_VERSION=2. This may require you to add the line “SAPENQREP_VERSION=2” if the INFO file does not yet contain a value for SAPENQREP_VERSION.

11. Restart the SAP system. Once the ERS instance is restarted, it will be using the enq_replicator (ERSv2) process instead of the enrepserver (ERSv1) process. This can be verified by right-clicking the ERS resource in LifeKeeper and selecting **Properties…**.



6.16.9. SAP Troubleshooting

This section provides a list of messages that you may encounter during the process of creating and extending an SPS SAP resource hierarchy, removing and restoring a resource and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition.

Messages from other SPS components are also possible. In these cases, please refer to the Message Catalog which provides a listing of all error codes, including operational, administrative and GUI, that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the error code and necessary action to resolve the issue. This full listing may be searched for any error code received, or you may go directly to one of the individual Message Catalogs for the appropriate SPS component.

Messages in this section fall under these topics:

- SPS SAP Messages
- Changing ERS Instances
- ASCS + ERS Restart_Program Parameter
- Hierarchy Remove Errors
- SAP Error Messages During Failover or In-Service
- SAP Installation Errors
- Troubleshooting sapinit
- ‘tset’ Errors Appear in the LifeKeeper Log File

6.16.9.1. SPS SAP Messages

This section references specific SPS Cause and Action messages as they relate to SAP. Refer to the collection of SPS SAP Messages to find the coded/named error message related to your issue.

112048 – alreadyprotected.ref

112022 – cannotfind.ref

112073 – cantcreateobject.ref

112071 – cantwrite.ref

112027 – checksummary.ref

112090 – cmdoutputempty.ref

112069 – commandnotfound.ref

112018 – commandReturned.ref

112033 – dbdown.ref

112023 – dbnotopen.ref

112032 – dbup.ref

112058 – depcreatefail.ref

112021 – disabled.ref

112092 – enqrepenabledprofileerror.ref

112080 – enqueuereplicationoutofsync.ref

112082 – enqueueversionmismatch.ref

112091 – enqversionprofileerror.ref

112049 – errorgetting.ref

112079 – ersrelocationextendfail.ref

112081 – ersrelocationstart.ref

112086 – ersremotelyisprestorefailure.ref

112085 – ersshouldmove.ref

112041 – exenotfound.ref

112066 – filemissing.ref

112057 – fscreatefailed.ref

112064 – gidnotequal.ref

112062 – homedir.ref

112043 – hung.ref

112065 – idnotequal.ref

112059 – inprogress.ref

112009 – instancenotrunning.ref

112010 – instancerunning.ref

112070 – invalidfile.ref

112067 – links.ref

112005 – lkinfoerror.ref

112004 – missingparam.ref

112088 – multcorrenqresources.ref

112035 – multimp.ref

112014 – multisap.ref

112053 – multisid.ref

112050 – multivip.ref

112039 – nfsdown.ref

112038 – nfsup.ref

112001 – nochildren.ref

112045 – noequiv.ref

112031 – nolkdbhost.ref

112056 – nonfsresource.ref

112024 – nonfs.ref

112026 – nopidnostatus.ref

112015 – nopid.ref

112040 – nosuchdir.ref

112013 – nosuchfile.ref

112006 – notrunning.ref

112036 – notshared.ref

112068 – objectinit.ref

112030 – pairdown.ref

112054 – pathnotmounted.ref

112060 – recoverfailed.ref

112046 – removefailed.ref

112047 – removesuccess.ref

112083 – resourcecanfailover.ref

112084 – resourcecannotfailover.ref

112002 – restorefailed.ref

112003 – restoresuccess.ref

112007 – running.ref

112052 – setupstatus.ref

112055 – sharedwarning.ref

112017 – sigwait.ref

112011 – startinstance.ref

112008 – start.ref

112025 – status.ref

112034 – stopfailed.ref

112029 – stopinstancefailed.ref

112028 – stopinstance.ref

112072 – stop.ref

112061 – targetandtemplate.ref

112044 – terminated.ref

112078 – threenodeersextendfail.ref

112093 – unabletokill.ref

112089 – unsupportedinstancetype.ref

112019 – updatefailed.ref

112020 – updatesuccess.ref

112000 – usage.ref

112063 – usernotfound.ref

112012 – userstatus.ref

112016 – usingkill.ref

112042 – validversion.ref

112037 – valueempty.ref

112051 – vipconfig.ref

6.16.9.1.1. 112048 – alreadyprotected.ref

Cause:

The SAP Instance “*{instance name}*” is already under LifeKeeper protection on server “*{server}*”.

Action:

Choose another SAP Instance to protect or specify the correct SAP Instance.

Return to SPS SAP Messages

6.16.9.1.2. 112022 – cannotfind.ref

Cause:

An error occurred trying to find the IP address “*{IP address}*” on “*{server}*”.

Action:

Verify the IP address or name exists in DNS or the hosts file.

Return to SPS SAP Messages

6.16.9.1.3. 112073 – cantcreateobject.ref

Cause:

Unable to create an internal object for the SAP instance using SID="*{SAP system id}*", Instance="*{instance name}*" and Tag="*{tag name}*" on "*{server}*".

Action:

The values specified for the object initialization (SID, Instance, Tag and System) were not valid.

Return to SPS SAP Messages

6.16.9.1.4. 112071 – cantwrite.ref

Cause:

The file “*{file name}*” exists, but was not read and write enabled on “*{server}*”.

Action:

Enable read and write permissions on the specified file.

Return to SPS SAP Messages

6.16.9.1.5. 112027 – checksummary.ref

Cause:

“{status check}” for “*{instance}*”: running processes=“*{number}*”, stopped processes=“*{number}*”, total expected=“*{number}*” on “*{server}*”.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.6. 112090 – cmdoutputempty.ref

Cause:

The command *“{command}”* did not return any output on “{system}”.

Action:

Please verify that the command is able to run successfully.

Return to SPS SAP Messages

6.16.9.1.7. 112069 – commandnotfound.ref

Cause:

The command *“{command}”* is not found in the *“{file}”* perl module (*“{module name}”*) on *“{server}”*.

Action:

Please check the command specified and retry the operation.

Return to SPS SAP Messages

6.16.9.1.8. 112018 – commandReturned.ref

Cause:

The *“{command}”* command returned *“{variable}”* on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.9. 112033 – dbdown.ref

Cause:

One or more of the database components for *“{db name}”* are down for *“{instance}”* on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.10. 112023 – dbnotopen.ref

Cause:

Database *“{DB name}”* is not open for SAP SID *“{SAP System ID}”* and Instance *“{instance}”* on *“{server}”*.

Action:

Information only. No action required.

Return to SPS SAP Messages

6.16.9.1.11. 112032 – dbup.ref

Cause:

All of the database components for *“{db name}”* are running for *“{instance}”* on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.12. 112058 – depcreatefail.ref

Cause:

Unable to create a dependency between parent tag *“{tag name}”* and child tag *“{tag name}”* on *“{server}”*.

Action:

Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.13. 112021 – disabled.ref

Cause:

The *“{recover action}”* (*“{script}”* action) has been disabled for the LifeKeeper resource *“{resource name}”* on *“{server}”*.

Action:

The desired action will need to be enabled for this resource.

Return to SPS SAP Messages

6.16.9.1.14. 112092 – enqrepenabledprofileerror.ref

Cause:

The profile *“{profile}”* either does not exist or cannot be read on *“{system}”*. Unable to determine whether enqueue replication is enabled for resource *“{resource tag}”*.

Action:

Please verify that the file exists and/or can be read.

Return to SPS SAP Messages

6.16.9.1.15. 112080 – enqueuereplicationoutofsync.ref

Cause:

Enqueue replication is enabled for resource *“{resource tag}”* but is currently inactive or out-of-sync. While the enqueue replication server is down, no redundancy is provided for the enqueue server lock table.

Action:

Please verify that the corresponding enqueue replication server is running.

Return to SPS SAP Messages

6.16.9.1.16. 112082 – enqueueversionmismatch.ref

Cause:

Instance *"{instance}"* is running a different version of the enqueue server than its corresponding enqueue replication server. This configuration is not supported by SAP and will lead to unexpected resource behavior. See SAP Note 2711036 – Usage of the Standalone Enqueue Server 2 in an HA Environment.

Action:

Please review the online documentation for instructions on how to modify the instance profiles for the enqueue server and enqueue replication server so that they use the same version.

Return to SPS SAP Messages

6.16.9.1.17. 112091 – enqversionprofileerror.ref

Cause:

The profile *“{profile}”* either does not exist or cannot be read on *“{system}”*. The enqueue version for resource *“{resource tag}”* will be set to 1 by default.

Action:

Please verify that the file exists and/or can be read.

Return to SPS SAP Messages

6.16.9.1.18. 112049 – errorgetting.ref

Cause:

Error getting SAP *“{variable}”* value from *“{file/path}”* on *“{server}”*.

Action:

Verify the value exists in the specified file.

Return to SPS SAP Messages

6.16.9.1.19. 112079 – ersrelocationextendfail.ref

Cause:

Instance *“{instance}”* is running concurrently with its corresponding central services instance on system *“{system}”* and is being automatically relocated to a backup system..

Action:

Please reattempt the extend operation after the ERS resource hierarchy has been relocated.

Return to SPS SAP Messages

6.16.9.1.20. 112081 – ersrelocationstart.ref

Cause:

Instance *“{instance}”* is running concurrently with its corresponding central services instance on system *“{system}”* and is being automatically relocated to a backup system..

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.21. 112086 – ersremotelyisprestorefailure.ref

Cause:

The ERS resource corresponding to resource *“{resource tag}”* is in-service and maintaining backup locks on a remote system. Bringing resource *“{resource tag}”* in-service on *“{system}”* would result in a loss of the backup lock table. Please bring resource *“{resource tag}”* in-service on the system where the corresponding ERS resource is currently in-service in order to maintain consistency of the lock table. In order to force resource *“{resource tag}”* in-service on *“{system}”*, either (i) run the command ‘/opt/LifeKeeper/bin/flg_create -f sap_cs_force_restore *“{resource tag}”* as root on *“{system}”* and reattempt the in-service operation or (ii) take the corresponding ERS resource out of service on the remote system. **Warning:** Both of these actions will result in a loss of the backup lock table.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.22. 112085 – ersshouldmove.ref

Cause:

Conditions are favorable to move the *“{ERS resource tag}”* resource hierarchy on *“{system}”*. The corresponding central services resource *“{A/SCS resource tag}”* is ISP on *“{system}”*, remote failover is allowed by the recovery level of resource *“{ERS resource tag}”* a viable backup node is alive for the resource hierarchy to failover to, and enqueue replication is either not configured or in-sync.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.23. 112041 – exenotfound.ref

Cause:

The required utility or executable *“{util name/exec name}”*, was not found or was not executable on *“{server}”*.

Action:

Verify the SAP installation and location of the required utility.

Return to SPS SAP Messages

6.16.9.1.24. 112066 – filemissing.ref

Cause:

The start and stop files are missing from *“{path name}”* on *“{server}”*.

Action:

Verify that SAP is installed correctly.

Return to SPS SAP Messages

6.16.9.1.25. 112057 – fscreatefailed.ref

Cause:

Unable to create a file system resource hierarchy for the file system *“{file system}”* on *“{server}”*.

Action:

Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.26. 112064 – gidnotequal.ref

Cause:

The group id for user *“{user name}”* is not the same on template server *“{server}”* and target server *“{server}”*.

Action:

Please correct the group id for the user so that it matches between the template and target servers.

Return to SPS SAP Messages

6.16.9.1.27. 112062 – homedir.ref

Cause:

Unable to find the home directory *“{directory name}”* for the SAP user *“{user name}”* on *“{server}”*.

Action:

Verify SAP is installed correctly.

Return to SPS SAP Messages

6.16.9.1.28. 112043 – hung.ref

Cause:

The command *“{command}”* with pid *“{pid}”* has hung. Forcibly terminating the command *“{command}”* on *“{server}”*.

Action:

Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.29. 112065 – idnotequal.ref

Cause:

The id for user *“{user name}”* is not the same on template server *“{server}”* and target server *“{server}”*.

Action:

Please correct the user id for the user so that it matches between the template and target servers.

Return to SPS SAP Messages

6.16.9.1.30. 112059 – inprogress.ref

Cause:

A check is already in progress for *“{tag}”*, exiting *“{command}”* on *“{server}”*.

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.31. 112009 – instancenotrunning.ref

Cause:

The SAP SID *“{SAP system ID}”* with instance *“{instance}”* is not running on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs

Return to SPS SAP Messages

6.16.9.1.32. 112010 – instancerunning.ref

Cause:

The SAP SID *"{SAP System ID}"* with instance *"{instance}"* is running on *"{server}"*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.33. 112070 – invalidfile.ref

Cause:

The file *“{file name}”* is not a valid file.The file does not contain any required definitions on *“{server}”*.

Action:

Please specify the correct file for this operation.

Return to SPS SAP Messages

6.16.9.1.34. 112067 – links.ref

Cause:

The LifeKeeper SAP environment is using links instead of NFS mounts on *“{server}”*.

Action:

Unable to verify the existence of the SAP startup and stop files.

Return to SPS SAP Messages

6.16.9.1.35. 112005 – lkinfoerror.ref

Cause:

Unable to find the resource information for the specified tag in function *“{function name}”* for *“{instance}”* on *“{server}”*.

Action:

Verify that the instance exists and is a valid SAP instance/resource.

Return to SPS SAP Messages

6.16.9.1.36. 112004 – missingparam.ref

Cause:

The *“{parameter}”* parameter was not specified for the *“{function name}”* function on *“{server}”*.

Action:

If this was a command line operation, specify the correct parameters; otherwise consult the Troubleshooting section.

Return to SPS SAP Messages

6.16.9.1.37. 112088 – multcorrenqresources.ref

Cause:

More than one corresponding enqueue resource was found for *“{resource tag}”* on *“{system}”*. Corresponding enqueue resources with the same SID were: *“{corresponding enqueue resource tag(s)}”*.

Action:

Verify that at most one central services instance and one enqueue replication server instance exists under the same SID on this system.

Return to SPS SAP Messages

6.16.9.1.38. 112035 – multimp.ref

Cause:

Detected multiple devices for the mount point *“{mount point}”* on *“{server}”*.

Action:

Verify the mounted file systems are correct.

Return to SPS SAP Messages

6.16.9.1.39. 112014 – multisap.ref

Cause:

Detected multiple SAP servers in the file *“{filename}”* for SAP SID *“{SAP System ID}”* and Instance *“{instance}”* on *“{server}”*.

Action:

Multiple SAP servers for the SID and Instance is not currently supported; remove the duplicate entry.

Return to SPS SAP Messages

6.16.9.1.40. 112053 – multisid.ref

Cause:

Detected multiple instance directories for the SAP SID *“{SAP System ID}”* with Instance ID *“{instance}”* in directory *“{directory name}”* on *“{server}”*.

Action:

The chosen SID is only allowed to have a single instance directory for a given ID. Multiple Instance directories with the same Instance ID is not a supported configuration.

Return to SPS SAP Messages

6.16.9.1.41. 112050 – multivip.ref

Cause:

Detected multiple Virtual IP addresses/Virtual Names for the Instance *“{instance name}”* on *“{server}”*.

Action:

Verify the configuration settings for the Instance are correct.

Return to SPS SAP Messages

6.16.9.1.42. 112039 – nfsdown.ref

Cause:

The NFS server “*{server}*” for SAP SID “*{SAP System ID}*” and Instance “*{instance}*” is not accessible on “*{server}*”.

Action:

Additional information available in the LifeKeeper and system logs. Please restart the required NFS server(s).

Return to SPS SAP Messages

6.16.9.1.43. 112038 – nfsup.ref

Cause:

The NFS server *“{server}”* for SAP SID *“{SAP System ID}”* and Instance *“{instance}”* is accessible on *“{server}”*.

Action:

Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.44. 112001 – nochildren.ref

Cause:

Warning: No children specified to extend.

Action:

Verify the dependency list is correct.

Return to SPS SAP Messages

6.16.9.1.45. 112045 – noequiv.ref

Cause:

There are no equivalent systems available to perform the *“{action}”* action for the Replicated Enqueue Instance *“{ERS instance}”* on *“{server}”*.

Action:

The resource must be extended to at most one server before this operation can complete.

Return to SPS SAP Messages

6.16.9.1.46. 112031 – nolkdbhost.ref

Cause:

The dbhost *"{dbhost name}"* is not on a LifeKeeper protected node paired with *"{server}"*.

Action:

Verify the *dbhost* is valid and functional for the protected instance(s).

Return to SPS SAP Messages

6.16.9.1.47. 112024 – nonfs.ref

Cause:

There was an error verifying the NFS connections for SAP related mount points on *“{server}”*.

Action:

One or more NFS servers is not operational and needs to be restarted.

Return to SPS SAP Messages

6.16.9.1.48. 112056 – nonfsresource.ref

Cause:

The NFS export for the path *“{path name}”* required by the instance *“{instance name}”* does not have an NFS hierarchy protecting it on *“{server}”*.

Action:

You must create an NFS hierarchy to protect the SAP NFS Exports before creating the SAP hierarchy.

Return to SPS SAP Messages

6.16.9.1.49. 112015 – nopid.ref

Cause:

Unable to find a running process id for the command/utility *“{command/utility}”* for SAP SID *“{SAP System ID}”* and Instance *“{instance}”* on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.50. 112026 – nopidnostatus.ref

Cause:

The process id was not found or the textstatus for *“{process name}”* was not set to running (textstatus=“*{state}*”) on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.51. 112040 – nosuchdir.ref

Cause:

The SAP Directory *“{directory name}”* (*“{directory path}”*) does not exist on *“{server}”*.

Action:

Verify the directory exists, or create the appropriate directory.

Return to SPS SAP Messages

6.16.9.1.52. 112013 – nosuchfile.ref

Cause:

The file *“{filename}”* does not exist or was not readable on *“{server}”*.

Action:

Verify that the specified file exists and/or has read permission set for the root user.

Return to SPS SAP Messages

6.16.9.1.53. 112006 – notrunning.ref

Cause:

The command *“{command name}”* is not running on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.54. 112036 – notshared.ref

Cause:

The path *“{path name}”* is not located on a shared file system or shared device on *“{server}”*.

The indicated path was found, but it does not appear to be located on a shared file system. This path is required to be on a shared file system or shared device.

Action:

Verify the path is correctly configured for HA protection. If it is a Network Attached Storage device, then the steeleye-lkNAS kit must be installed.

Return to SPS SAP Messages

6.16.9.1.55. 112068 – objectinit.ref

Cause:

“Getting the tag object for *“{tag}”* failed, retrying using the template information of *“{template system}”* on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs. Please wait…

Return to SPS SAP Messages

6.16.9.1.56. 112030 – pairdown.ref

Cause:

The clustered pair *“{server name}”* with equivalency to *“{tag name}”* is not alive on *“{server}”*.

Action:

The connection between the clustered pairs must be established before executing this action.

Return to SPS SAP Messages

6.16.9.1.57. 112054 – pathnotmounted.ref

Cause:

“{}”: The path *“{path}”* (*“{path name}”*) is not mounted or does not exist on *“{server}”*.

Action:

Verify the installation and mount points are correct on this server.

Return to SPS SAP Messages

6.16.9.1.58. 112060 – recoverfailed.ref

Cause:

All attempts at local recovery for the SAP resource *“{resource name}”* have failed on *“{server}”*.

Action:

A failover to the backup server will be attempted.

Return to SPS SAP Messages

6.16.9.1.59. 112046 – removefailed.ref

Cause:

The SAP Instance *“{instance name}”* and all required processes were not stopped successfully during the *“{action}”* on server *“{server}”*.

Action:

Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.60. 112047 – removesuccess.ref

Cause:

The SAP Instance *“{instance name}”* and all required processes were stopped successfully during the *“{action}”* on server *“{server}”*.

Action:

Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.61. 112083 – resourcecanfailover.ref

Cause:

System *“{system}”* is a viable failover target for the resource hierarchy containing *“{resource tag}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.62. 112084 – resourcecannotfailover.ref

Cause:

No system in the cluster is currently a viable failover target for the resource hierarchy containing *“{resource tag}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.63. 112002 – restorefailed.ref

Cause:

The SAP Instance *“{instance}”* and all required processes were not started successfully during the *“{action}”* on server *“{server}”*.

Action:

Please check the LifeKeeper and system logs for additional information and retry the operation.

Return to SPS SAP Messages

6.16.9.1.64. 112003 – restoresuccess.ref

Cause:

The SAP Instance *“{instance}”* and all required processes were started successfully during the *“{action}”* on server *“{server}”*.

Action:Reference Documents

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.65. 112007 – running.ref

Cause:

The command *“{command}”* is running on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.66. 112052 – setupstatus.ref

Cause:

Verifying the *“{}”* basics of the *“{instance name}”* installation on *“{server}”*.

Action:

Information only. No action required.

Return to SPS SAP Messages

6.16.9.1.67. 112055 – sharedwarning.ref

Cause:

This is a warning but will become a critical error if *“{path}”* is not shared on *“{server}”*.

Action:

The indicated path was found, but it does not appear to be located on a shared file system. This path is required to be on a shared file system. Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.68. 112017 – sigwait.ref

Cause:

Signal *“{signal}”* sent to process id *“{process id}”*, waiting for a recheck to occur on *“{server}”*.

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.69. 112011 – startinstance.ref

Cause:

Issuing a start/restart of the SAP SID *“{SAP System ID}”* with instance *“{instance}”* on *“{server}”*.

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.70. 112008 – start.ref

Cause:

Issuing a start/restart of the command *“{command}”* on *“{server}”*.

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.71. 112025 – status.ref

Cause:

All processes for SAP SID *“{SAP System ID}”* and Instance *“{instance}”* are *“{state}”* on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.72. 112034 – stopfailed.ref

Cause:

Unable to stop the sap process/utility *“{process/utility name}”* with command *“{command}”* on *“{server}”*.

Action:

Please check the LifeKeeper and system logs for additional information and retry the operation.

Return to SPS SAP Messages

6.16.9.1.73. 112029 – stopinstancefailed.ref

Cause:

The SAP Instance *“{instance}”* and all required processes were not stopped successfully on server *“{server}”*.

Action:

Please check the LifeKeeper and system logs for additional information and retry the operation.

Return to SPS SAP Messages

6.16.9.1.74. 112028 – stopinstance.ref

Cause:

Issuing a stop of the SAP SID *“{SAP System ID}”* with instance *“{instance}”* using command *“{command}”* on *“{server}”*.

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.75. 112072 – stop.ref

Cause:

Issuing a stop/kill of the command *“{command}”* on *“{server}”*.

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.76. 112061 – targetandtemplate.ref

Cause:

The values specified for the target and the template servers are the same.

Action:

Please specify the correct values for the target and template servers.

Return to SPS SAP Messages

6.16.9.1.77. 112044 – terminated.ref

Cause:

The command *“{command}”* with pid *“{pid}”* terminating due to signal *“{signal}”* on *“{server}”*.

Action:

Information only. No action required.

Return to SPS SAP Messages

6.16.9.1.78. 112078 – threenodeersextendfail.ref

Cause:

ERSv1 is only supported in two-node clusters. Resource *“{resource tag}”* is unable to be extended to system *“{target system}”*.

Action:

Upgrade to ERSv2 in order to extend the hierarchy to three or more nodes.

Return to SPS SAP Messages

6.16.9.1.79. 112093 – unabletokill.ref

Cause:

Unable to kill *“{processes}”* for SID *“{SID}”* and instance *“{instance}”* (PID(s): *“{process pids}”*).

Action:

Please kill the processes manually and attempt the operation again.

Return to SPS SAP Messages

6.16.9.1.80. 112089 – unsupportedinstancetype.ref

Cause:

“{Routine}” was called for a resource with unsupported instance type *“{instance type}”*. This method supports only SAP instance type(s) *“{supported instance type(s)}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.81. 112019 – updatefailed.ref

Cause:

The update of the resource information field for resource with tag *“{tag name}”* has failed on *“{server}”*.

Action:

View the resource properties manually using `ins_list -t <tag>` to verify the resource is functional.

Return to SPS SAP Messages

6.16.9.1.82. 112020 – updatesuccess.ref

Cause:

The update of the resource information field for resource with tag *“{tag name}”* was successful on *“{server}”*.

Action:

Additional information is available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.83. 112000 – usage.ref

Cause:

Usage: *“{command name}” “{command usage}”*

Action:

Specify the correct usage for the requested command.

Return to SPS SAP Messages

6.16.9.1.84. 112063 – usernotfound.ref

Cause:

The SAP user *“{user name}”* does not exist on *“{server}”*.

Action:

Verify the SAP installation or create the required SAP user specified.

Return to SPS SAP Messages

6.16.9.1.85. 112012 – userstatus.ref

Cause:

Preparing to run the command: *“{command}”* on *“{server}”*.

Action:

Information only. No action required.

Return to SPS SAP Messages

6.16.9.1.86. 112016 – usingkill.ref

Cause:

Stopping process id *“{process id}”* of *“{SAP command}”* for SAP SID *“{SAP System ID}”* and Instance *“{instance}”* with command *“{command}”* on *“{server}”*.

Action:

Please wait…

Return to SPS SAP Messages

6.16.9.1.87. 112042 – validversion.ref

Cause:

One or more SAP / LK validation checks has failed on *“{server}”*.

Action:

Please update the version of SAP on this host to include the SAPHOST and SAPCONTROL Packages.

Return to SPS SAP Messages

6.16.9.1.88. 112037 – valueempty.ref

Cause:

The internal object value *“{value}”* was empty. Unable to complete *“{function}”* on *“{server}”*.

Action:

Additional information available in the LifeKeeper and system logs.

Return to SPS SAP Messages

6.16.9.1.89. 112051 – vipconfig.ref

Cause:

The *“{value name}”* or *“{value name}”* value in the file *“{filename}”* is still set to the physical server name on *“{server}”*.

Action:

The value(s) must be set to a virtual server name. See Configuring SAP with LifeKeeper for information on how to configure SAP to work with LifeKeeper.

Return to SPS SAP Messages

6.16.9.2. Disable Autostart in ERS Profile

Symptom:

A status check of an ERS Instance causes a `sapstart` for the selected instance.

ERS instance is always running on both systems.

Cause:

When an ERS Instance has Autostart=1 set in the profile, certain `sapcontrol` calls will cause the instance to be started as a part of the running command.

Action:

Stop the running ERS Instances in the cluster and modify the profile for the ERS Instances and set *Autostart=0*.

This profile change will require a restart of sapstartsrv in order to take effect. This can be accomplished by running the following command (replacing <sid> by the lower-case SAP SID and <SID> by the upper-case SAP SID):

```
su - <sid>adm -c "sapcontrol -nr <ERS Inst#> -function RestartService <SID>"
```

6.16.9.3. ASCS + ERS Restart_Program Parameter

Symptom:

The enqueue server or enqueue replicator process is running on both the primary and backup servers at the same time.

Cause:

If the ERS instance profile is configured to start the instance with Restart_Program instead of Start_Program, the sapstart process will automatically restart the ERS instance when it is terminated for any reason. This will cause unexpected behavior and could lead to a loss of the enqueue server lock table during failover or switchover.

Action:

Modify the ASCS and ERS instance profile parameters to use Start_Program instead of Restart_Program when starting the enqueue server and enqueue replicator processes. See Modify ASCS and ERS Instance Profile Settings for more details.

Once the ASCS and ERS profiles have been updated to use Start_Program instead of Restart_Program and SAP Start Service has been restarted, SIOS recommends restarting the system to ensure the updated profile is read and no caching is in effect.

If successful, the ERS instance should only be running on the backup system where the ASCS/SCS instance is not currently running.

For more details refer to Installing SAP.

6.16.9.4. SAP Hierarchy Remove Errors

Symptom:

File system remove fails with file system in use.

Cause:

1. **Resource Protection Level** was set to **Basic** or **Minimum** after the create or extend. When the resource Protection Level is set to Basic or Minimum, the SAP resource hierarchy will not be stopped during the remove operation. This leaves the processes running for that instance when remove is called. If the processes are also accessing the protected file system, LifeKeeper may be unable to unmount the file system.

2. **Resource Protection Level** was set to **Standard** for a non-replicated enqueue resource. When the resource Protection Level is set to Standard, the SAP resource hierarchy will not be stopped during the remove operation. This leaves the processes running for that instance when remove is called. If the processes are also accessing the protected file system, LifeKeeper may be unable to unmount the file system.

Action:

1. The **Basic** and/or **Minimum** settings should be used to place a resource in a temporary maintenance mode. It should not be used as an ongoing Protection Level. If the resource in question will require Basic or Minimum as the ongoing Protection Level, the Instance should be configured to use local storage and/or the entire resource hierarchy should be configured without the use of the LifeKeeper NAS Recovery Kit for local NFS mounts.

2. The **Standard** setting should be used for replicated enqueue resources only. **Note: Standard** is used for ERS resources that were created in SPS-L 9.3.2 or earlier and reside at the top of the SAP hierarchy with a dependency on the corresponding central services resource. ERS instances created in SPS-L 9.4.0 or later that reside in a hierarchy independent of the central services resource should have their Protection Level set to **Full**.

6.16.9.5. SAP Error Messages During Failover or In-Service

After a failover of a SAP, there will be error messages in the SAP logs. Many of these error messages are normal and can be ignored.

On Failure of the DB

`BVx: Work Process is in reconnect status` – This error message simply states that a work progress has lost the connection to the database and is trying to reconnect.

`BVx: Work Process has left reconnect status` – This is not really an error, but states that the database is back up and the process has reconnected to it.

Other errors – There could be any number of other errors in the logs during the period of time that the database is down.

On Startup of the CI

`E15: Buffer SCSA Already Exists` – This error message is not really an error at all. It is simply telling you that a previously created shared memory area was found on the system which will be used by SAP.

`E07: Error 00000 : 3No such process in Module rslgsmcc (071)` – *See SAP Note 7316* – During the previous shutdown, a lock was not released properly. This error message can be ignored.

During a LifeKeeper In-Service Operation

The following messages may be displayed in the LifeKeeper In Service Dialog during an in-service operation:

```
error: permission denied on key 'net.unix.max_dgram_qlen'

error: permission denied on key 'kernel.cap-bound'
```

These errors occur when `saposcol` is started and can be ignored (see *SAP Note 201144*).

6.16.9.6. SAP Installation Errors

Incorrect Name in tnsnames.ora or listener.ora Files

Cause:

When using the Oracle database, if the SAP installation program complains about the incorrect server name being in the *tnsnames.ora* or *listener.ora* file when you do the PAS Backup Server installation, then you may not have installed the Oracle binaries on local file systems.

Action:

The Oracle binaries in */oracle//920*<32 or 64>_ must be installed on a local file system on each server for the configuration to work properly.

6.16.9.7. Troubleshooting sapinit

Symptom:

`sapstartsrv` processes and additional SAP instance processes started by init script fail or cause processes to run on LifeKeeper Standby Node.

Cause:

SAP provides an init script for automatically starting SAP instances on a local node. When a resource is added to LifeKeeper protection, the init script (`sapinit`) may attempt to start SAP Instance processes that should not be running on the current node.

Action:

Disable the `sapinit` script or modify `sapinit` to skip over LifeKeeper protected Instances. To disable this behavior, the user must stop `sapinit` (Example: `/etc/init.d/sapinit stop`). The `sapinit` script should also be disabled using `chkconfig` or similar tool (Example: `chkconfig sapinit off`).

6.16.9.8. tset Errors Appear in the LifeKeeper Log File

Cause:

The su commands used by the SAP and Database Recovery Kits cause a ‘tset’ error message to be output to the LK log that appears as follows:

```
tset: standard error: Invalid argument
```

This error comes from one of the profile files in the SAP administrator’s and Database user’s home directory and it is only in a non-interactive shell.

Action:

If using the c-shell for the Database user and SAP Administrator, add the following lines into the *.sapenv_<hostname>.sh* in the home directory for these users. This code should be added around the code that determines if ‘tset’ should be executed:

```
if ( $?prompt ) then
    tty -s
    if ( $status == 0) then
        .
        .
        .
    endif
endif
```

Note: The code from “tty –s” to the inner “endif” already exists in the file.

If using the bash shell for the Database user and SAP Administrator, add the following lines into the *.sapenv_<hostname>.sh* in the home directory for the users.

Before the code that determines if ‘tset’ should be executed add:

```
case $- in
```

```
*i*) INTERACTIVE =“yes”;;

          *) INTERACTIVE =“no”;;

     esac
```

Around the code the that determines if ‘tset’ should be executed add:

```
             if [ $INTERACTIVE == “yes” ]; then

             tty -s

             if [ $? -eq 0 ]; then

                    .

                    .

                    .

               fi

          fi
```

Note: The code from “tty –s” to the inner “endif” already exists in the file.

6.16.10. Maintenance Mode

Maintaining a LifeKeeper Protected System

When performing shutdown and maintenance on a LifeKeeper-protected server, you must put that system's resource hierarchies in service on the backup server before performing maintenance. This process stops all activity for shared disks on the system needing maintenance.

Perform these actions in the order specified, where *Server A* is the primary system in need of maintenance and *Server B* is the backup server:

1. **Bring hierarchies in service on *Server B***. On the backup, *Server B*, use the LifeKeeper GUI to bring in service any resource hierarchies that are currently in service on *Server A*. This will unmount any file systems currently mounted on *Server A* that reside on the shared disks under LifeKeeper protection. See Bringing a Resource In Service for instructions.

2. **Stop LifeKeeper on *Server A***. Use the command `/etc/init.d/lifekeeper stop-nofailover (or, LKSTOP_MODE=stop-nofailover systemctl stop lifekeeper)` to stop LifeKeeper. Your resources are now unprotected.

3. **Shut down Linux and power down *Server A***. Shut down the Linux operating system on *Server A*, then power off the server.

4. **Perform maintenance**. Perform the necessary maintenance on *Server A*.

5. **Power on *Server A* and restart Linux**. Power on *Server A*, then reboot the Linux operating system.

6. **Start LifeKeeper on *Server A***. Use the command `/etc/init.d/lifekeeper start (or, systemctl start lifekeeper)` to start LifeKeeper. Your resources are now protected.

7. **Bring hierarchies back in-service on *Server A*, if desired**. On *Server A*, use the LifeKeeper GUI to bring in service all resource hierarchies that were switched over to *Server B*.

6.16.10.1. SAP Maintenance Mode

Placing LifeKeeper Protected Resources in Maintenance Mode During SAP Software Update

LifeKeeper has the ability to place the resources in an SAP hierarchy into **maintenance mode** to allow the user to upgrade their SAP software in-place without having to first bring the hierarchy in-service on a backup server.

When maintenance mode is enabled for the SAP hierarchy:

- Resource health monitoring, local recovery, and failover are disabled for all resources in the hierarchy.

> **Note:** To minimize the potential for data corruption during the upgrade process, SCSI reservation errors will still be detected and acted on by LifeKeeper. This may cause a resource hierarchy failover or may halt the system in the case of a lost SCSI reservation. To modify the behavior of LifeKeeper when a SCSI device cannot be accessed, see the SCSIERROR parameter in the Core Parameters List.

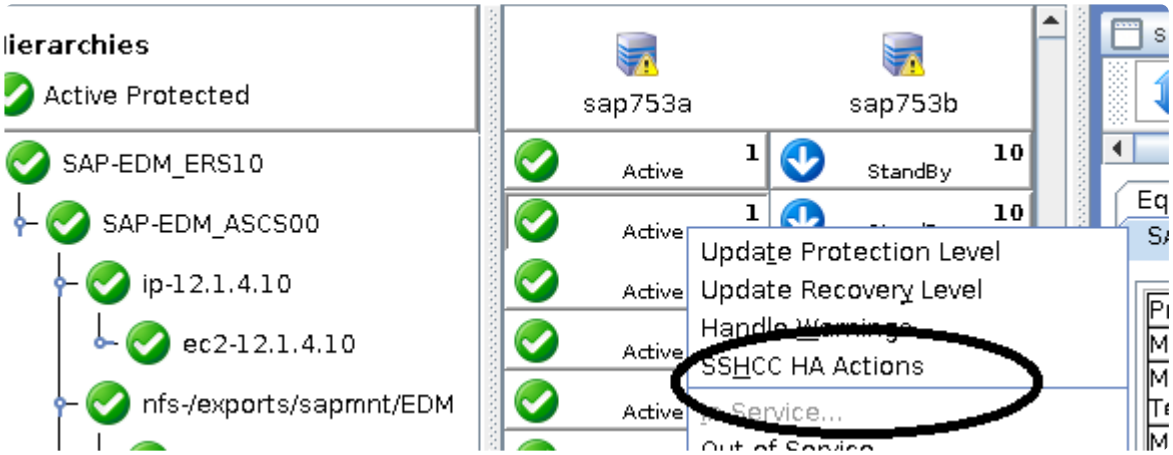
> **Note:** When using Quorum/Witness features, the default behavior of LifeKeeper is to halt the node upon loss of quorum. This is beneficial if the user has critical application hierarchies for which a loss of quorum needs to be acted on, but could possibly lead to an unexpected halt while performing maintenance activities if a network communication error occurs. Steps to manually disable/enable Quorum/Witness features can be found on the Quorum/Witness documentation page.

- Any DataKeeper mirrors in the hierarchy will be paused until either maintenance mode is disabled or the user manually resumes the mirror.

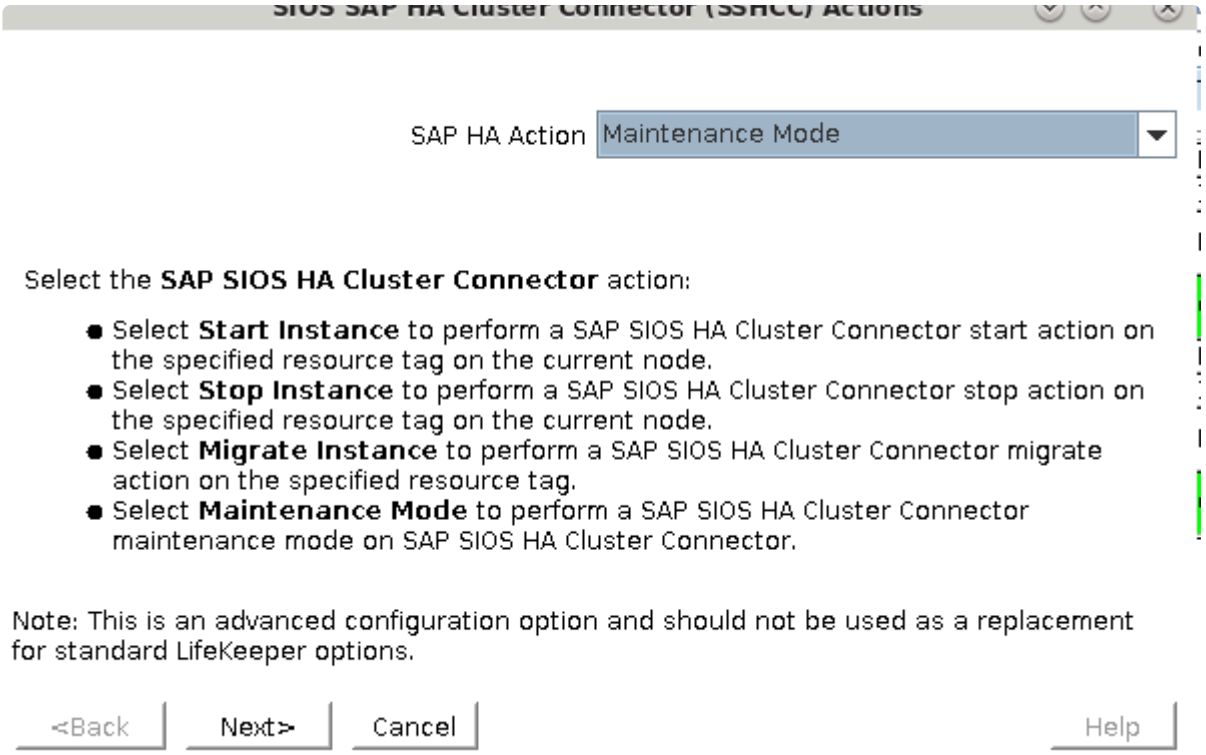
> **Note:** It is not recommended to manually resume any mirror while maintenance mode is enabled for the hierarchy. Since health monitoring, local recovery, and failover are disabled, any failure of the mirror will not be acted on by LifeKeeper.

To place the SAP hierarchies into maintenance mode:

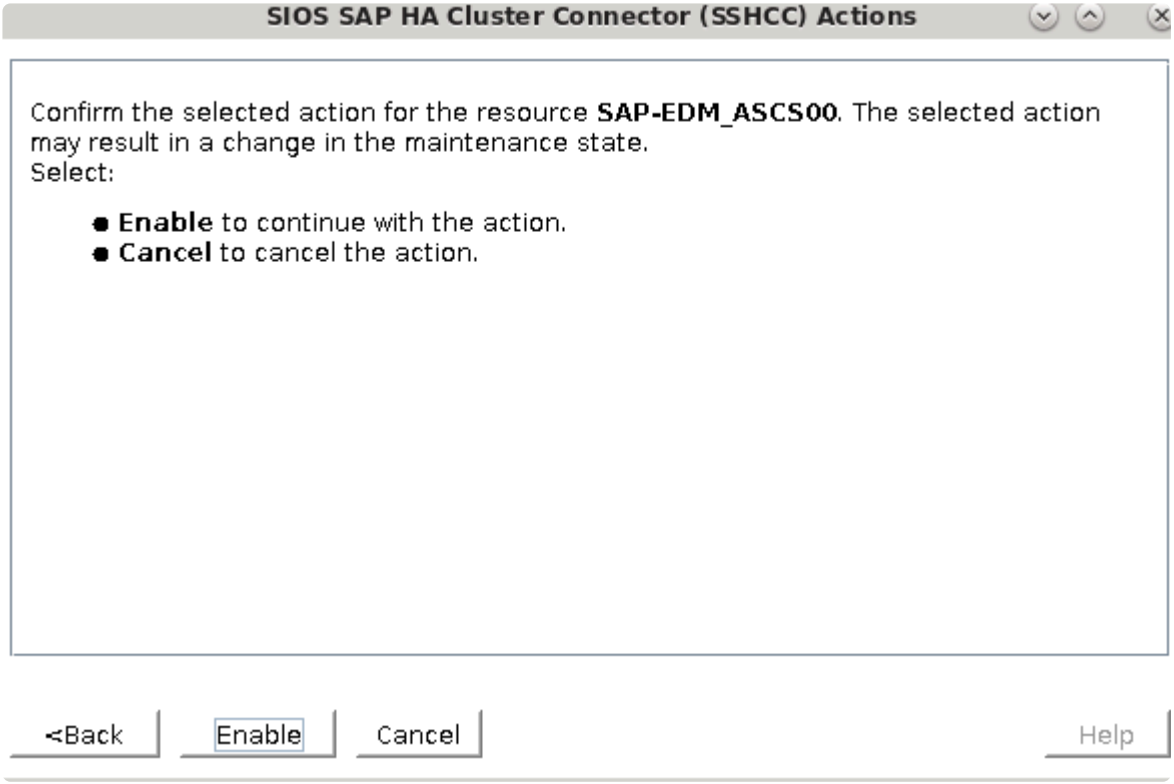
1. **Open** the **LifeKeeper GUI**.
2. **Right-click** the **SAP ASCS or SCS** resource in each hierarchy on the node where it is currently in-service and click **SSHCC HA Actions**.



3. Select **Maintenance Mode** in the drop-down box.



4. Select **Enable** or **Disable** maintenance mode for all resources in the chosen hierarchy. This action will be attempted on all nodes in the cluster.



Note: In SAP kernel 7.49 PL 200 and later, placing the SAP software into maintenance mode via SAP Update Manager will automatically place the corresponding LifeKeeper SAP hierarchy into maintenance mode via the HA connector function HASetMaintenanceMode as long as the SAP SIOS HA Cluster Connector is active for the SAP instance. If LifeKeeper resources are placed into maintenance mode manually via the LifeKeeper GUI, the user will need to refer to the documentation for SAP Update Manager to determine how to place the SAP software itself into maintenance mode before upgrading.

Checking Maintenance Mode Status for LifeKeeper Protected Resources

The LifeKeeper GUI does not currently show the maintenance mode status for each resource hierarchy. To check the maintenance mode status of an SAP hierarchy in LifeKeeper, run the following command from the command line:

```
sudo /opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/lk_maintenance_mode --mode=check --tag=<A/SCS Resource Tag> --cluster
```

where <A/SCS Resource Tag> is the LifeKeeper tag of the SAP ASCS or SCS resource (e.g., SAP-SID_ASCS00). The output will show whether maintenance mode is fully enabled, partially enabled, or fully disabled for the hierarchy containing the given resource.

6.16.10.2. Custom and Maintenance-Mode Behavior via Policies

> **Note:** As of SPS-L version 9.3.2, HA Maintenance Mode can be enabled for an SAP hierarchy via the SSHCC Actions menu in the LifeKeeper UI. See SAP Maintenance Mode for more details.

Resource Policy Management

Overview

Resource Policy Management in SIOS Protection Suite for Linux provides behavior management of resource local recovery and failover. Resource policies are managed with the **lkpolicy** command line tool (CLI).

SIOS Protection Suite

SIOS Protection Suite is designed to monitor individual applications and groups of related applications, periodically performing local recoveries or notifications when protected applications fail. Related applications, by example, are hierarchies where the primary application depends on lower-level storage or network resources. When an application or resource failure occurs, the default behavior is:

1. **Local Recovery**: First, attempt **local** recovery of the resource or application. An attempt will be made to restore the resource or application on the local server without external intervention. If local recovery is successful, then SIOS Protection Suite will not perform any additional action.

2. **Failover**: Second, if a local recovery attempt *fails* to restore the resource or application (or the *recovery kit* monitoring the resource has no support for local recovery), then a ***failover*** will be initiated. The failover action attempts to bring the application (and all dependent resources) into service on another server within the cluster.

Please see SIOS Protection Suite Fault Detection and Recovery Scenarios for more detailed information about our recovery behavior.

Custom and Maintenance-Mode Behavior via Policies

SIOS Protection Suite Version 7.5 and later supports the ability to set additional policies that modify the default recovery behavior. There are four policies that can be set for individual resources (see the section below about precautions regarding individual resource policies) *or* for an entire server. ***The recommended approach is to alter policies at the server level***.

The available policies are:

Standard Policies

- **Failover** This policy setting can be used to turn on/off resource failover. (**Note**: In order for reservations to be handled correctly, **Failover** cannot be turned off for individual scsi resources.)

- **LocalRecovery** – SIOS Protection Suite, by default, will attempt to recover protected resources by restarting the individual resource or the entire protected application prior to performing a failover. This policy setting can be used to turn on/off local recovery.

- **TemporalRecovery** – Normally, SIOS Protection Suite will perform local recovery of a failed resource. If local recovery fails, SIOS Protection Suite will perform a resource hierarchy failover to another node. If the local recovery succeeds, failover will not be performed.

  There may be cases where the local recovery succeeds, but due to some irregularity in the server, the local recovery is re-attempted within a short time; resulting in multiple, consecutive local recovery attempts. This may degrade availability for the affected application.

  To prevent this repetitive local recovery/failure cycle, you may set a temporal recovery policy. The temporal recovery policy allows an administrator to limit the number of local recovery attempts (successful or not) within a defined time period.

  > *Example*: If a user sets the policy definition to limit the resource to three local recovery attempts in a 30-minute time period, SIOS Protection Suite will fail over when a third local recovery attempt occurs *within* the 30-minute period.

  Defined temporal recovery policies may be turned *on* or *off*. When a temporal recovery policy is *off*, temporal recovery processing will continue to be done and notifications will appear in the log when the policy *would* have fired; however, no actions will be taken.

  Note: It is possible to disable failover and/or local recovery with a temporal recovery policy also in place. This state is illogical as the temporal recovery policy will **never** be acted upon if failover or local recovery are disabled.

Meta Policies

The “meta” policies are the ones that can affect more than one other policy at the same time. These policies are usually used as shortcuts for getting certain system behaviors that would otherwise require setting multiple standard policies.

- **NotificationOnly** – This mode allows administrators to put SIOS Protection Suite in a “monitoring only” state. ***Both*** local recovery ***and*** failover ***of a resource (or all resources in the case of a server-wide policy) are affected***. The user interface will indicate a Failure state if a failure is detected; *but no recovery or failover action will be taken*. **Note**: The administrator will need to correct the problem that caused the failure manually and then bring the affected resource(s) back in service to continue normal SIOS Protection Suite operations.

Important Considerations for Resource-Level Policies

Resource level policies are policies that apply to a specific resource only, as opposed to an entire resource hierarchy or server.

> *Example* :
>
> app
> - IP
> - file system
>
> In the above resource hierarchy, app depends on both IP and file system. A policy can be set to disable local recovery or failover of a specific resource. This means that, for example, if the IP resource's local recovery fails and a policy was set to *disable* failover of the IP resource, then the IP resource will not fail over or cause a failover of the other resources. However, if the file system resource's local recovery fails and the file system resource policy does not have failover disabled, then the entire hierarchy will fail over.
>
> **Note**: It is important to remember that resource level policies apply only to the specific resource for which they are set.

This is a simple example. Complex hierarchies can be configured, so care must be taken when setting resource-level policies.

The lkpolicy Tool

The **`lkpolicy`** tool is the command-line tool that allows management (querying, setting, removing) of policies on servers running SIOS Protection Suite for Linux. `lkpolicy` supports setting/modifying policies, removing policies and viewing all available policies and their current settings. In addition, defined policies can be set on or off, preserving resource/server settings while affecting recovery behavior.

The general usage is :

```
lkpolicy [—list-policies | —get-policies | —set-policy | —remove-policy] <name value pair data…>
```

> The `<name value pair data…>` differ depending on the operation and the policy being manipulated, particularly when setting policies. For *example*: Most on/off type policies only require `—on` or `—off` switch, but the temporal policy requires additional values to describe the threshold values.

Example lkpolicy Usage

Authenticating With Local and Remote Servers

The **`lkpolicy`** tool communicates with SIOS Protection Suite servers via an API that the servers expose. This API requires authentication from clients like the `lkpolicy` tool. The first time the `lkpolicy` tool is asked to access a SIOS Protection Suite server, if the credentials for that server are not known, it will ask the user for credentials for that server. These credentials are in the form of a username and password and:

1. Clients must have SIOS Protection Suite admin rights. This means the username must be in the *lkadmin group* according to the operating system's authentication configuration (via pam). It is ***not*** necessary to run as ***root***, but the root user can be used since it is in the appropriate group by default.

2. The credentials will be stored in the *credential store* so they do not have to be entered manually each time the tool is used to access this server.

See Configuring Credentials for SIOS Protection Suite for more information on the credential store and its management with the `credstore` utility.

An example session with `lkpolicy` might look like this:

```
[root@thor49 ~]# lkpolicy -l -d v6test4
Please enter your credentials for the system 'v6test4'.
Username: root
Password:
Confirm password:
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]# lkpolicy -l -d v6test4
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]#
```

Listing Policies

lkpolicy —list-policy-types

Showing Current Policies

lkpolicy —get-policies

lkpolicy —get-policies tag=*

lkpolicy —get-policies —verbose tag=mysql* # all resources starting with mysql

lkpolicy —get-policies tag=mytagonly

Setting Policies

lkpolicy —set-policy Failover —off

lkpolicy —set-policy Failover —on tag=myresource

lkpolicy —set-policy Failover —on tag=*

lkpolicy —set-policy LocalRecovery —off tag=myresource

lkpolicy —set-policy NotificationOnly —on

lkpolicy —set-policy TemporalRecovery —on recoverylimit=5 period=15

lkpolicy —set-policy TemporalRecovery —on —force recoverylimit=5 period=10

Removing Policies

lkpolicy —remove-policy Failover tag=steve

Note: *NotificationOnly is a policy alias. Enabling NotificationOnly is the equivalent of disabling the corresponding LocalRecovery and Failover policies*.

6.17. SAP HANA Recovery Kit Administration Guide

Beginning in v9.5.0 SIOS has released the new SAP HANA Application Recovery Kit. SIOS will continue to support the SAP HANA gen/app based Recovery Kit with the 9.4.x releases until **March 31, 2022**. If you are using SIOS Protection Suite for Linux v9.5 or later you must use the new (built-in) SAP HANA Application Recovery Kit.

The existing SAP HANA gen/app based Recovery Kit is **not** supported with v9.5.0. Users who wish to upgrade to the SIOS Protection Suite for Linux v9.5.0 **must** convert their existing SAP HANA gen/app based Recovery Kit to the new SAP HANA Recovery Kit. Refer to Upgrading from the SAP HANA Gen/App to the SAP HANA Recovery Kit for details.

SAP HANA is an enterprise-class in-memory database system that can be deployed for a wide array of purposes. It forms the basis for the SAP S/4HANA Enterprise Resource Planning platform. The SAP HANA Recovery Kit provides fault resilient protection for SAP HANA databases in a SIOS Protection Suite for Linux environment.

Document Contents

This guide includes the following topics to help you successfully create and manage your SAP HANA hierarchy:

- SAP HANA Recovery Kit Requirements. Lists the hardware and software necessary to properly set up, install and operate the SAP HANA Recovery Kit.
- Overview. Describes the SAP HANA Recovery Kit's features and functionality.
- Configuring SAP HANA with SPS. Provides instructions for installing and configuring the SAP HANA software.
- Resource Configuration Tasks. Describes the various functions you may perform on your hierarchies using the LifeKeeper GUI: create, extend, delete and unextend.
- Hierarchy Administration. Provides important recommendations for ongoing administration of the SAP HANA hierarchy.
- Troubleshooting. Lists and describes the error messages associated with the SAP HANA Recovery Kit.

SIOS Protection Suite Documentation

The following SIOS Protection Suite product documentation is available from the SIOS Technology Corp. website:

- SPS for Linux Release Notes

- SPS for Linux Technical Documentation
- SPS for Linux Installation Guide
- Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is provided on the SIOS Technology Corp. website at:

http://docs.us.sios.com/

and from the Help menu in the LifeKeeper GUI.

SAP HANA Documentation

Documentation for SAP HANA can be found at:

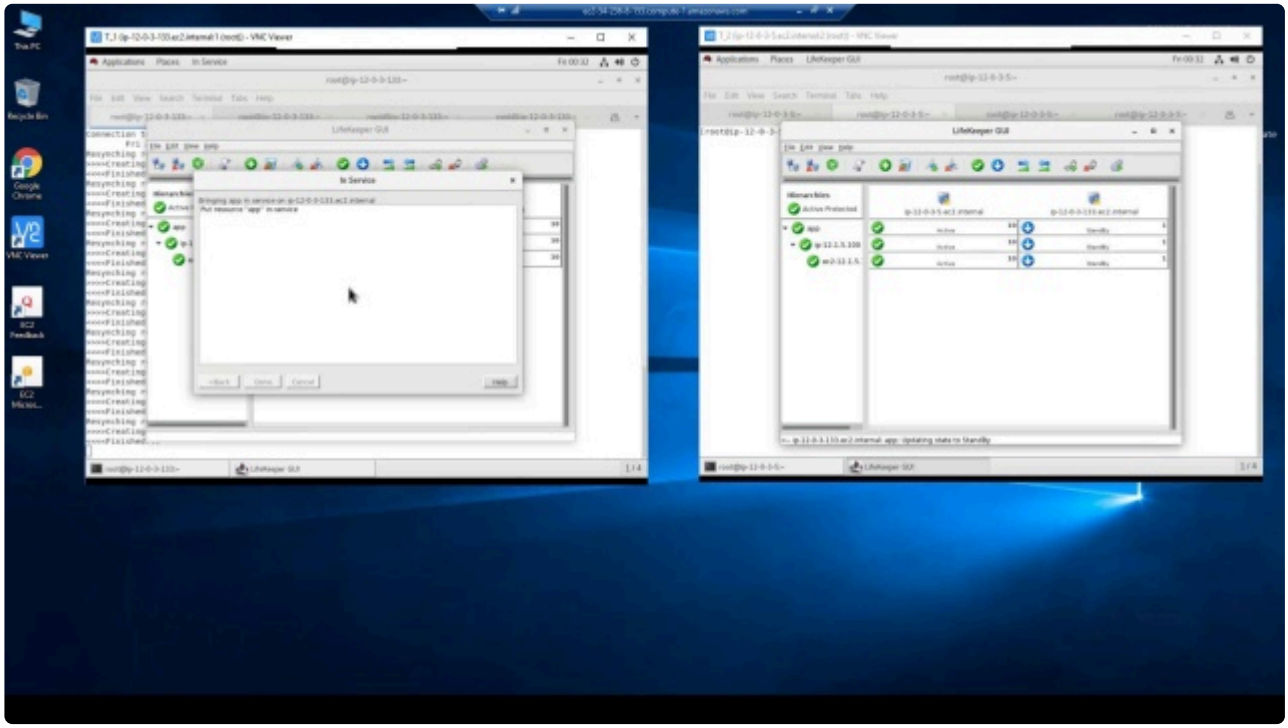
http://help.sap.com

6.17.1. Upgrading from the SAP HANA Gen/App to the SAP HANA Recovery Kit

✻ Beginning in v9.5.0 SIOS has released the new SAP HANA Application Recovery Kit. SIOS will continue to support the SAP HANA gen/app based Recovery Kit with the 9.4.x releases until **March 31, 2022**. If you are using SIOS Protection Suite for Linux v9.5 or later you must use the new (built-in) SAP HANA Application Recovery Kit.

! The existing SAP HANA gen/app based Recovery Kit is **not** supported with v9.5.0. Users who wish to upgrade to the SIOS Protection Suite for Linux v9.5.0 **must** convert their existing SAP HANA gen/app based Recovery Kit to the new SAP HANA Recovery Kit. Please follow the upgrade steps below.

✻ Please refer to the Support Matrix before upgrading.



Upgrade SIOS Protection Suite

1. Upgrade SIOS Protection Suite to the latest version on all nodes in the cluster (refer to Upgrading SPS).

2. Install the **SAP HANA ARK** and the **required** corresponding license on all nodes in the cluster.

Remove SAP HANA Gen/App Resources

1. LifeKeeper should be running on all nodes and the gen/app resource hierarchy should be in service.

2. Backup the LifeKeeper configuration in case the resource hierarchies need to be restored to the previous settings. To perform the backup execute the following command:

   ```
   /opt/LifeKeeper/bin/lkbackup -c --cluster
   ```

   By default the backup file is located at */opt/LifeKeeper/config/archive.<date –yyyymmddhhmm>.tar.gz*.

3. Right click the **SAP HANA gen/app** resource hierarchy in the left panel.

4. Choose **Delete Dependency…** if there are any dependencies attached to the gen/app resource (e.g., Virtual IP or AWS EC2 EIP).

5. After the dependencies are removed, right click the **SAP HANA gen/app** resource.

6. Choose **Delete Resource Hierarchy…** to remove the old SAP HANA gen/app resource.

> Please note that after the SAP HANA genapp resource is removed, there will be no SIOS Protection Suite Failover or Monitoring of the SAP HANA instance until a new SAP HANA resource is created. HANA System Replication will remain active unless changes outside of SPS cause it to terminate.

Create and Extend SAP HANA Resource

Create/Extend a SAP HANA resource using the newly installed SAP HANA ARK.

1. To create and extend a **SAP HANA resource**, follow the steps in Creating an SAP HANA Resource Hierarchy and Extending an SAP HANA Resource Hierarchy.

2. Once an **SAP HANA resource** hierarchy is created and extended, if you want to create or re-create any dependencies that were removed in Step 4 above, perform the following steps:

   a. Right click the **SAP HANA resource** hierarchy in the left panel.

   b. Choose **Create Dependency…** to create dependencies in the new SAP HANA resource (e.g., Virtual IP or AWS EC2 EIP) that were attached to the SAP HANA gen/app resource.

3. Verify/test the new SAP HANA resource hierarchy by performing the tests found here.

Remove HANA Gen/App Files

To remove the gen/app package from each system, execute the following command:

```
rpm -e steeleye-lkHOTFIX-HANA-SP1-9.1.0-6538.noarch
```

6.17.2. SAP HANA Recovery Kit Hardware and Software Requirements

✻ Beginning in v9.5.0 SIOS has released the new SAP HANA Application Recovery Kit. SIOS will continue to support the SAP HANA gen/app based Recovery Kit with the 9.4.x releases until **March 31, 2022**. If you are using SIOS Protection Suite for Linux v9.5 or later you must use the new (built-in) SAP HANA Application Recovery Kit.

! The existing SAP HANA gen/app based Recovery Kit is **not** supported with v9.5.0. Users who wish to upgrade to the SIOS Protection Suite for Linux v9.5.0 **must** convert their existing SAP HANA gen/app based Recovery Kit to the new SAP HANA Recovery Kit. Refer to Upgrading from the SAP HANA Gen/App to the SAP HANA Recovery Kit for details.

Hardware Requirements

- **Servers:** Servers should be configured in accordance with the requirements described in the SAP HANA Master Guide, SAP notes mentioned in this guide, SPS for Linux Documentation, and the SPS for Linux Release Notes.

- **Storage:** For SAP HANA databases utilizing SAP HANA System Replication, no shared storage is necessary. Special storage requirements are given in the SAP HANA Master Guide and the aforementioned SAP notes.

Software Requirements

- **SPS Software:** It is imperative that you install the same version of SPS software and apply the same versions of the SPS software patches to each server in your cluster.

- **SAP HANA Software:** SAP HANA Platform Edition 2.0 SP04 (or later) is required for use of the SAP HANA Recovery Kit. The same version of the SAP HANA software must be installed and configured on each server before configuring SPS and the SPS SAP HANA Recovery Kit. Since SAP HANA licenses are tied to hardware, each server will require its own license from SAP. SAP HANA System Replication must be enabled and fully configured and the database must be running on all cluster nodes before creating the SAP HANA resource hierarchy in LifeKeeper.

- **SPS SAP HANA Recovery Kit:** The SAP HANA Recovery Kit is bundled as an optional recovery kit with the core installation in SIOS Protection Suite for Linux v9.5.0 and later.

- **Witness Server:** SIOS recommends adding a 3rd node to a 2-node configuration as a witness server.

6.17.3. SAP HANA Recovery Kit Overview

The SAP HANA Recovery Kit is compatible with SAP HANA Platform 2 (SP04 or later).

SAP HANA provides three different mechanisms to increase availability.

- **Host Auto-Failover** – At least one standby node added to a SAP HANA system. These nodes are configured to work in standby mode. If the required processes or databases are not active, LifeKeeper will attempt to restart them. In case of an unsuccessful restart of the processes on the primary node, LifeKeeper will attempt to bring the database in-service on the backup node, register that node as primary master in SAP HANA System Replication, and register the previous primary node as the secondary replication site. If the previous primary node cannot be configured as the secondary SAP HANA System Replication site, the resource will be marked as Failed (OSF) on that node until the problem is corrected and it can be successfully registered. Once the previous primary node has been successfully registered as a secondary replication site, LifeKeeper will update the state of the SAP HANA resource on the node to Standby (OSU).

- **Storage Replication** – The storage used on the primary SAP HANA node replicates all data to another SAP HANA node. This replication works without a control process from the SAP HANA system. The storage replication is provided by hardware partners.

- **System Replication** – SAP HANA replicates all data from the primary SAP HANA node to a backup node by use of SAP's own built-in replication framework. Data is constantly pre-loaded on the secondary SAP HANA node.

With the SAP HANA Recovery Kit, SAP HANA systems, utilizing System Replication, can be protected and administered through SIOS LifeKeeper.

The Recovery Kit is able to start the SAP HANA system on all nodes and perform the takeover and replication site registration processes of SAP HANA System Replication. To ensure the functionality of the SAP HANA system, the following processes and states are continuously monitored:

- SAP Host Agent on all nodes
- SAP Start Service (sapstartsrv) of HDB instance on all nodes
- State of SAP HANA database on all nodes
- State of SAP HANA System Replication mode (primary on active node, sync|syncmem|async on secondary node)

If the required processes or databases are not active, LifeKeeper will attempt to restart them. In case of an unsuccessful restart of the processes on the primary node, LifeKeeper will attempt to bring the database in-service on the backup node, register that node as primary master in SAP HANA System Replication, and register the previous primary node as the secondary replication site. If the previous primary node cannot be configured as the secondary SAP HANA System Replication site, the resource will be marked as Failed (OSF) on that node until the problem is corrected and it can be successfully registered. Once the previous primary node has been successfully registered as a secondary replication

site, LifeKeeper will update the state of the SAP HANA resource on the node to Standby (OSU).

In case of an invalid state of the SAP HANA System Replication, the SAP HANA resource is also placed in the state “Out of Service – Faulty” (OSF). It has to be decided with the help of a database administrator whether a takeover is to be performed or how the SAP HANA System Replication mode should be corrected.

When carrying out the “Out of Service” action for an SAP HANA resource in LifeKeeper, only the database on the primary node is stopped by default. The database on the secondary node remains active and retains its SAP HANA System Replication mode.

6.17.3.1. SAP HANA GUI States

The active (ISP) resource can have the following states. Some of these warning and failure states are transient and may appear while LifeKeeper is attempting to recover required processes or before LifeKeeper initiates a failover of the HANA resource hierarchy.

GUI Text	GUI Properties State	Description	Icon
Active	Active	Sapstartsrv and the HDB instance are running properly. (State Name: ISP)	
Active – sapstartsrv Failure	Sapstartsrv is not running, HDB status unknown	Sapstartsrv is not running. The HDB instance status is unknown since sapstartsrv is used to check the status of HDB. (State Name: ISPSapStartSrvNotRunning)	
Active – HSR Disabled	HSR is running in replication mode 'none'	Sapstartsrv and HDB instance are running properly. The replication mode is 'none', indicating that SAP HANA Replication is disabled. (State Name: ISPHSRDisabled)	
Active – HDB Stopped	Sapstartsrv is running, HDB is not running properly	Sapstartsrv is running. The HDB instance is not running properly. (State Name: ISPHDBNotRunning)	
Active – Secondary	HSR is running in a secondary replication mode	Sapstartsrv and the HDB instance are running properly but the active node is registered as a secondary replication site. In normal operation, the active node is expected to be the primary replication site. (State Name: ISPSecondary)	
Active – Unknown Repl Mode	Replication mode cannot be determined	Sapstartsrv and HDB instance are running properly. The replication mode cannot be determined using 'hdbnsutil -sr_state' or it returns an unsupported mode. (State Name: ISPUnknownReplMode)	

The state shown in the GUI for the standby (OSU) resource is affected by the active (ISP) resource restore and quickCheck processes. The standby resource can take several minutes to get into its final state during the in-service operation, during which transient intermediate resource states may appear. This is normal and expected operation for SAP HANA resources. Some of the warning and failure states shown below are transient and may appear while LifeKeeper is attempting to recover required processes or re-register the standby server as a secondary replication site. If the SAP HANA database cannot be successfully started or registered as a secondary replication site on the backup server, the resource state on the backup server will transition to Failed (OSF). The standby resource can have the following states:

GUI Text	GUI Properties State	Description	Icon
Standby	No cluster HANA resource ISP,	This is the normal status for a resource that is out-of-service when an equivalent resource is not in-service on another node. (State Name: OSUStopped)	

	HDB is not running		
Standby – HDB Running	HDB is running, HSR monitoring inactive	Sapstartsrv and the HDB instance are running on the standby (OSU) node. There is either no active (ISP) node, or quickCheck has not determined the HSR status on the active node. (State Name: OSUHDBRunning)	
Standby – In Sync	HSR active and in-sync	Sapstartsrv and the HDB instance are running with HSR configured and reporting in-sync. (State Name: SecondaryActive)	
Standby – sapstartsrv Failure	sapstartsrv is not running, HDB status unknown	Sapstartsrv is not running. There is an active (ISP) node. The HDB instance status is unknown since sapstartsrv is used to check the status of HDB. (State Name: OSUSapStartSrvNotRunning)	
Standby – Unknown Repl. Mode	Replication mode cannot be determined	Sapstartsrv and the HDB instance are running. There is an active (ISP) node. The replication mode cannot be determined using ‘hdbnsutil -sr_state‘. (State Name: OSUUnknownReplMode)	
Standby – HSR Disabled	Replication mode is ‘none’	Sapstartsrv and the HDB instance are running. There is an active (ISP) node. Either the replication mode is ‘none’ according to ‘hdbnsutil -sr_state‘, or the HANA utility systemReplicationStatus.py returned ‘10’. (State Name: OSUHSRDisabled)	
Standby – HSR Error	Active node HSR reports error (11)	Sapstartsrv and the HDB instance are running. There is an active (ISP) node. The HANA utility systemReplicationStatus.py returned ‘11’ . (State Name: OSUHSRError)	
Standby – Initializing	HSR Initializing	Sapstartsrv and the HDB instance are running. There is an active (ISP) node. HANA utility systemReplicationStatus.py returned ‘13’ . (State Name: SecondaryInitializing)	
Standby – Syncing	HSR Synchronizing	Sapstartsrv and the HDB instance are running. There is an active (ISP) node. The HANA utility systemReplicationStatus.py returned ‘14’ . (State Name: SecondarySyncing)	
Standby – HDB Stopped	HANA resource ISP in cluster, HDB is not running	Sapstartsrv is running. The HDB instance is not running. There is an active (ISP) node. (State Name: OSUHDBNotRunning)	
Standby – Primary	HSR is running in replication mode ‘Primary’	Sapstartsrv and the HDB instance are running properly, but the standby node is registered as a primary replication site. In normal operation, the standby node is expected to be a secondary replication site. (State Name: OSUPrimary)	
Standby – Unknown HSR Status	Active node HSR reports error (12)	Sapstartsrv and the HDB instance are running. There is an active (ISP) node. The HANA utility systemReplicationStatus.py returned ‘12’ . (State Name: SecondaryUnknownHSRStatus)	
Standby – Unexpected HSR state	HSR status from ISP node not in expected range 10 – 15	Sapstartsrv and the HDB instance are running. There is an active (ISP) node. The HANA utility systemReplicationStatus.py returned a value that was not in the expected range of 10 to 15 . (State Name: OSUUnexpectedHSRState)	

6.17.3.2. SAP HANA Resource Hierarchy

The following example shows a typical SAP HANA resource hierarchy:

The child resource **vip-sps-hana** in this example is protecting the switchable IP address used for client connections to the database.

In the event of failover, SPS will bring the IP address and database resource in service on a backup server. Any database transaction which has not yet been committed will need to be run again.

6.17.4. Configuring SAP HANA with SPS

The following sequence is recommended for installing and configuring the SAP HANA database and SPS software. Each of these steps links to detailed tasks that follow.

- Install the SAP HANA Software
- Configure SAP HANA System Replication
- Modify the SAP HANA Instance Profile
- Install the SPS Software

After you have performed these tasks, you will be ready to create the SPS resource hierarchy to protect your SAP HANA database.

6.17.4.1. Install the SAP HANA Software

Install the SAP HANA software on all servers in the cluster using identical parameters/settings. In particular, the same SAP System ID (SID) and instance number must be used on all systems. Refer to the SAP HANA Master Guide for installation details.

6.17.4.2. Configure SAP HANA System Replication

Configure SAP HANA System Replication according to the instructions provided in the SAP HANA System Replication Guide (available at http://help.sap.com). Once SAP HANA System Replication has been successfully enabled on the intended primary replication site and the backup server has been successfully registered as a secondary replication site, continue with the rest of the steps in this guide.

6.17.4.3. Modify the SAP HANA Instance Profile

Disable Autostart for the HDB Instance

In order for LifeKeeper to successfully manage the SAP HANA resource during failover or system reboot, the Autostart feature which automatically starts the HDB instance on system boot must be disabled. In order to disable this feature, edit the instance profile for your HDB instance (typically located at /usr/sap/<SID>/SYS/profile/<SID>_HDB<##>_<HostName>) **on all cluster nodes** and ensure that the line

```
Autostart = 0
```

is present in the profile. Either add or modify this line, as necessary, and save the changes to the profile.

> ! Warning: If Autostart is not disabled for the HDB instance, then a machine failure of the primary SAP HANA System Replication site will result in a “System Replication split brain” scenario once the original primary node comes back online. In this scenario, the HDB instance is running and registered as primary master concurrently on multiple cluster nodes. As a result, LifeKeeper is unable to determine which site the user intends to be registered as the primary SAP HANA System Replication site. The SAP HANA resource is placed in the OSF (“Out of Service – Failed”) state on the original standby node and a warning message is broadcast to all open consoles until the situation is manually resolved by a database administrator. See Resolving Split Brain Scenarios for more details.

> * **Note:** In order to ensure continued successful management of the SAP HANA resource, LifeKeeper will monitor the value of the Autostart parameter for the HDB instance on all cluster nodes each quickCheck interval (defined by the LKCHECKINTERVAL parameter, default 2 minutes) and will automatically disable Autostart if necessary.

6.17.4.4. Install the SPS Software

Once you have installed the SAP HANA software and configured SAP HANA System Replication, you are ready to install the SPS Core software and any required patches followed by the SAP HANA Recovery Kit.

Refer to the SPS for Linux Installation Guide for details on installing the SPS packages.

6.17.5. SAP HANA Resource Configuration Tasks

Once you have completed the setup tasks described in the previous section, you are ready to create and extend your SAP HANA resource hierarchies.

The following tasks are available for configuring the SIOS Protection Suite for Linux SAP HANA Recovery Kit:

- **Create Resource Hierarchy** – Creates an SAP HANA resource hierarchy
- **Delete Resource Hierarchy** – Deletes an SAP HANA resource hierarchy
- **Extend Resource Hierarchy** – Extends an SAP HANA resource hierarchy from the primary server to the backup server
- **Unextend Resource Hierarchy** – Unextends (removes) an SAP HANA resource hierarchy from a single server in the SPS cluster

Please refer to your SPS for Linux Technical Documentation for additional instructions on configuring your LifeKeeper resource hierarchies.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View / Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu.

You may also perform most of the tasks:

- from the toolbar
- by right-clicking on a global resource in the left pane of the status display
- by right-clicking on a resource in the right pane of the status display

Using the right-click method allows you to avoid entering certain information that is required when using the **Edit** menu.

6.17.5.1. Creating an SAP HANA Resource Hierarchy

> **Important Note:** Before creating your SAP HANA resource hierarchy, SAP HANA System Replication must be enabled and fully configured and the database must be running on all servers in the cluster. See Configure SAP HANA System Replication for details.

Perform the following steps on the primary server:

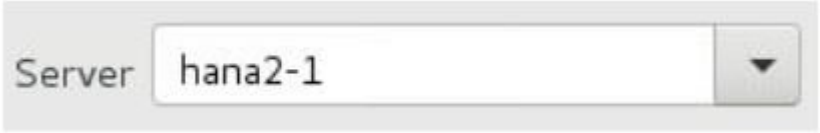
1. From the **Edit** menu, select **Server**, then **Create Resource Hierarchy**. The ***Create Resource Wizard*** dialog will appear.

2. Select **SAP HANA** from the drop-down list and click **Next**.

3. Select *intelligent* as the **Switchback Type** to be used for the SAP HANA resource. Click **Next**.

   Intelligent Switchback means that after a failover to the backup server, an administrator must manually switch the SAP HANA resource back to the primary server. CAUTION: This release of SPS does not support Automatic Switchback for SAP HANA resources. Additionally, the Automatic Switchback restriction is applicable for any other LifeKeeper resource with a dependency on a SAP HANA resource.

4. Select the **Server** on which the SAP HANA resource will be created. When creating the initial SAP HANA resource, the server chosen at this step must be the one on which the database is currently registered as primary master in SAP HANA System Replication. Click **Next**.



5. Select the **SAP HANA SID** under which the SAP HANA database is installed. Click **Next**.

6. Select the **SAP HANA** Instance to be protected by LifeKeeper. Click **Next**.

7. Select the **Dependent Virtual IP** resource to be protected by LifeKeeper. The IP resource must already exist and be in-service on the selected server in order to appear in the list. Select *none* if switching over of the virtual IP/host name on failover or switchover will be managed without using a LifeKeeper IP resource. Click **Next**.

8. Enter the **SAP HANA Resource Tag** to be used to identify the SAP HANA resource which will be created on the chosen server. The default tag name has the form HANA-<SID>_<HDB Instance>, but can be modified to use a different tag name as long as it is not currently being used to identify another LifeKeeper resource. Valid characters in tag names are letters, digits, and the following special characters: “-”, “_”, and “.”. Forward slashes (“/”) cannot be used in the tag name for a SAP HANA resource.

9. Click **Create**. The ***Create Resource Wizard*** will then create your SAP HANA resource hierarchy. SPS will validate the data entered as well as data obtained from the SAP HANA System Replication framework. If a problem is detected, an error message will appear in the information box. Click **Next**.

10. You should see a message indicating that you have successfully created an SAP HANA resource hierarchy and you must extend that hierarchy to another server in your cluster to achieve failover protection. Click **Next**.

11. Click **Continue**. SPS will then launch the ***Pre-Extend Wizard***. Refer to Extending an SAP HANA Resource Hierarchy for details on how to extend your resource hierarchy to another server.

6.17.5.2. Extending an SAP HANA Resource Hierarchy

> **Important Note:** Before extending your SAP HANA resource hierarchy, SAP HANA System Replication must be enabled and fully configured and the database must be running on all servers in the cluster. See Configure SAP HANA System Replication for details.

This operation can either be performed from the **Edit** menu or initiated automatically upon completing the **Create Resource Hierarchy** option, in which case you should begin with Step 2 below.

1. From the **Edit** menu, select **Resource**, then **Extend Resource Hierarchy**. The ***Pre-Extend Wizard*** appears. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the SPS Extend Resource Hierarchy defaults and want to bypass the prompts for input/ confirmation, click **Accept Defaults**.

2. Select the **Template Server** from which you want to extend an existing SAP HANA resource. (*This dialog box will not appear if you selected the **Extend** task by right-clicking on a resource instance in the right pane.*) Click **Next**.

3. Select the **Tag to Extend** for the SAP HANA resource that you would like to extend. (*This dialog box will not appear if you selected the **Extend** task by right-clicking on a resource instance in either pane.*) Click **Next**.

4. Select *intelligent* for the **Switchback Type** to be used for the SAP HANA resource. Click **Next**.

   Intelligent Switchback means that after a failover to the backup server, an administrator must manually switch the SAP HANA resource back to the primary server. CAUTION: This release of SPS does not support Automatic Switchback for SAP HANA resources. Additionally, the Automatic Switchback restriction is applicable for any other LifeKeeper resource with a dependency on a SAP HANA resource.

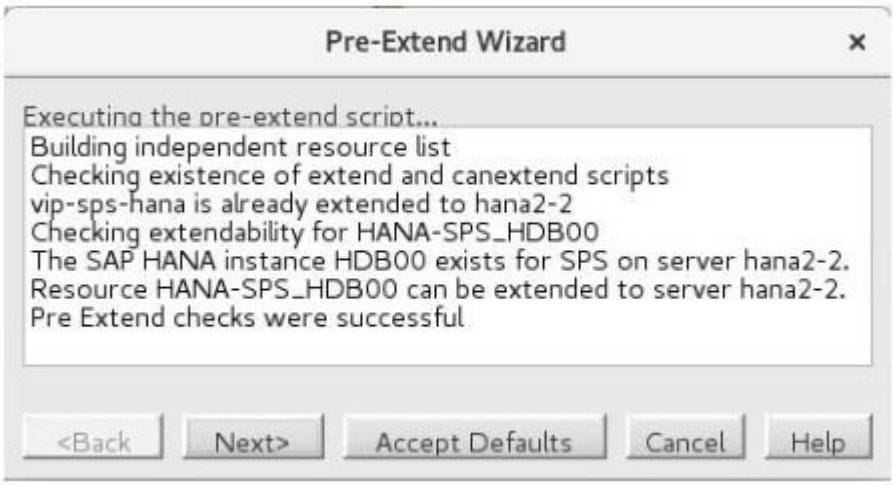
5. Select or enter the **Template Priority**. Click **Next**.

This is the priority for the SAP HANA hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note:** This selection will appear only for the initial extend of the hierarchy.

6. Select or enter the **Target Priority**. Click **Next**.

This is a priority for the newly extended SAP HANA hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. Note that SPS assigns priority 1 to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

7. After receiving the message that the pre-extend checks were successful, click **Next**.



8. The ***Extend Wizard*** will prompt you to enter the **Root Tag** for the SAP HANA resource. This is the unique name used by LifeKeeper to identify the equivalent SAP HANA resource being created on the target server. **Note:** The SAP HANA resource tag name is required to be the same across all cluster servers, so it cannot be edited in this dialog box. Also, forward slashes ("/") cannot be used in the tag name for a SAP HANA resource.

9. Click **Extend**. The ***Extend Wizard*** will then extend your SAP HANA resource hierarchy to the target server. If a problem is detected, an error message will appear in the information box.

10. After receiving the message "`Hierarchy successfully extended`", click **Next Server** to extend the hierarchy to another server, or click **Finish** if there are no other extend operations to perform.

11. After receiving the message "`Hierarchy Verification Finished`", click **Done**.

6.17.5.3. Unextending an SAP HANA Resource Hierarchy

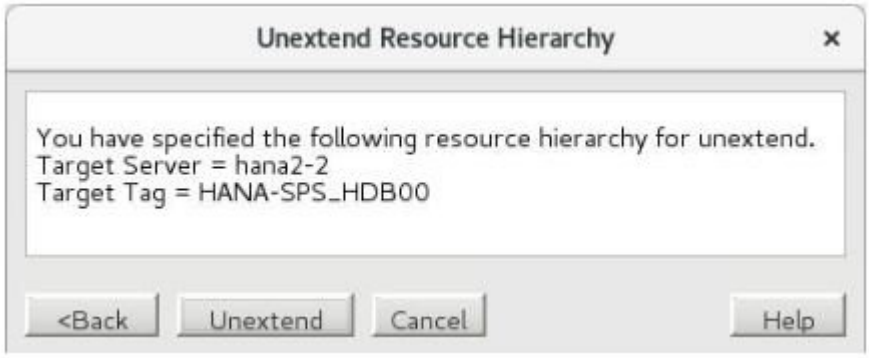
To remove a resource hierarchy from a single server in the SPS cluster, do the following:

1. From the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.

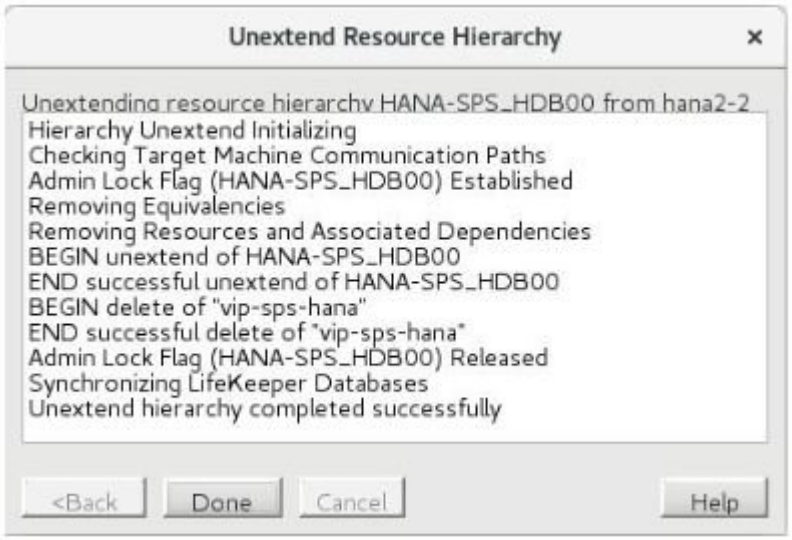
2. Select the **Target Server** where you want to unextend the SAP HANA resource. It cannot be the server where the resource is currently in service. (*This dialog box will not appear if you selected the **Unextend** task by right-clicking on a resource instance in the right pane*.) Click **Next**.

3. Select the SAP HANA hierarchy to unextend and click **Next**. (*This dialog box will not appear if you selected the **Unextend** task by right-clicking on a resource instance in either pane*).

4. An information box appears confirming the target server and the SAP HANA resource hierarchy you have chosen to unextend. Click **Unextend**.



5. Another information box appears confirming that the SAP HANA resource was unextended successfully. Click **Done** to exit the **Unextend Resource Hierarchy** menu selection.


Unextend Resource Hierarchy
Unextending resource hierarchy HANA-SPS_HDB00 from hana2-2
Hierarchy Unextend Initializing
Checking Target Machine Communication Paths
Admin Lock Flag (HANA-SPS_HDB00) Established
Removing Equivalencies
Removing Resources and Associated Dependencies
BEGIN unextend of HANA-SPS_HDB00
END successful unextend of HANA-SPS_HDB00
BEGIN delete of "vip-sps-hana"
END successful delete of "vip-sps-hana"
Admin Lock Flag (HANA-SPS_HDB00) Released
Synchronizing LifeKeeper Databases
Unextend hierarchy completed successfully
<Back
Done
Cancel
Help

6.17.5.4. Deleting an SAP HANA Resource Hierarchy

To delete SAP HANA resource from all servers in your SPS cluster environment, complete the following steps:

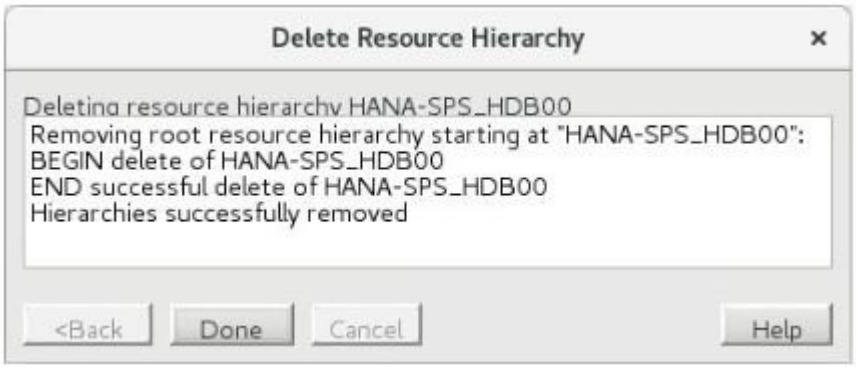
1. From the Edit menu, select **Resource**, then **Delete Resource Hierarchy**.

2. Select the name of the **Target Server** where you will be deleting your SAP HANA resource hierarchy. (*This dialog will not appear if you selected the **Delete Resource** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance*.)

3. Select the **Hierarchy to Delete**. (*This dialog will not appear if you selected the **Delete Resource** task by right-clicking on a resource instance in the left or right pane*.) Click **Next**.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Next**.

5. Another information box appears confirming that the SAP HANA resource was deleted successfully.



6. Click **Done** to exit.

6.17.5.5. Testing your SAP HANA Resource Hierarchy

Test Scenarios

To understand the behavior of the SAP HANA Recovery Kit, perform the following tests. The following prerequisites must be completed before performing any test:

- LifeKeeper and the SAP HANA database must be installed and configured according to the installation instructions provided by SIOS and SAP.
- SAP HANA System Replication must be enabled and active on all servers in the cluster, with the secondary replication site registered using one of the valid replication modes (sync, syncmem, or async) and operation modes (delta_datashipping, logreplay, or logreplay_readaccess). See Configure SAP HANA System Replication for more details.
- If managing the switchable IP address associated with the SAP HANA database with a LifeKeeper IP resource, there must exist a dependency of the SAP HANA resource on the IP resource. See Step 7 in Creating an SAP HANA Resource Hierarchy for more details.

Test Recovery of SAP Host Agent

Determine the status and the process ID's of the SAP Host Agent processes by using:

```
# /usr/sap/hostctrl/exe/saphostexec -status

saphostexec running (pid = 3818)

sapstartsrv running (pid = 3867)

saposcol running (pid = 3965)
```

Either manually kill one of the processes listed in the output or execute

```
/usr/sap/hostctrl/exe/saphostexec -stop
```

to impair the functionality of SAP Host Agent. The SAP HANA Recovery Kit will recognize that SAP Host Agent is not working properly and restart it on that node. The behavior can be observed by monitoring the LifeKeeper log with the following command:

```
tail -f /var/log/lifekeeper.log
```

During this recovery process, the SAP HANA resource does not change its state. After a successful recovery, SAP Host Agent is fully functional again. If the recovery kit is unable to restart SAP Host Agent, the HANA database and the resource remains in its current state. SAP Host Agent will be

checked again and if possible restarted later.

Test Recovery of sapstartsrv for the SAP HANA Instance

To test the recovery of the SAP Start Service (sapstartsrv) for the SAP HANA instance, the service must be stopped. One method to stop sapstartsrv is by executing the sapcontrol StopService webmethod:

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function StopService"
```

where <sid> is the lower-case SAP System ID for the HANA installation and <Inst#> is the HDB instance number. Another method is to kill the sapstartsrv process directly. In either case, sapstartsrv will be restarted by SAP HANA Recovery Kit. The resource does not change its state as long as sapstartsrv is able to be restarted successfully.

Test Recovery of the Secondary SAP HANA DB (Replication Target)

In the event of a failure of the secondary database instance (replication target) or if the secondary replication site is unregistered in SAP HANA System Replication, the recovery kit will re-register the secondary site with the previous replication and operation modes and restart the secondary database instance.

To induce such a failure, execute one of the following commands on the secondary replication site:

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function Stop"
```

```
su - <sid>adm -c "hdbnsutil -sr_unregister"
```

The behavior can be observed by monitoring the log file `/var/log/lifekeeper.log`. After the recovery, the state of the database instance and SAP HANA System Replication can be tested by running the following commands on the secondary replication site:

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function GetProcessList"
```

```
su - <sid>adm -c "hdbnsutil -sr_state"
```

In the event that the secondary database instance cannot be started by the recovery kit, the SAP HANA resource is flagged as Failed (OSF) on the corresponding node.

Once the cause of an unsuccessful start is fixed by an administrator, the SAP HANA Recovery Kit will start the database instance in the subsequent quickCheck cycle. Once started successfully, the resource

state will be updated to Standby (OSU) on the corresponding node.

Test Recovery of the Primary SAP HANA DB

In the event of a failure of the primary database instance (replication source), the replication mode of the database instance on the primary node is determined. If the replication mode is set to primary, the database instance will be started again. If the mode is not set to primary, the recovery kit will log a warning stating that the replication mode has been changed outside of LifeKeeper and suspend all monitoring of the SAP HANA resource until the issue is resolved. In the latter case, manual intervention is required to bring the HANA resource hierarchy in-service on the correct primary system. The behavior in this case can be observed in the LifeKeeper GUI, which will show the state “Active – HSR Disabled”, “Active – Unknown Repl Mode”, or “Active – Secondary” for the resource on the primary node, or by monitoring the log file `/var/log/lifekeeper.log`.

A failure of the primary database instance can be induced by running the following command on the primary replication site:

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function Stop"
```

After the recovery, the state of the database and the replication can be tested by using:

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function GetProcessList"
```

```
su - <sid>adm -c "hdbnsutil -sr_state"
```

In the event that the primary database instance cannot be started by the recovery kit on that node, LifeKeeper will initiate a failover of the entire hierarchy to the secondary node. On this node, the HANA Recovery Kit performs a takeover of SAP HANA System Replication and the previous secondary node becomes the new primary node for replication. LifeKeeper will attempt to re-register the faulty node as the secondary replication site using the previous replication and operation modes. If this is successful, the secondary database is also restarted. In the event that either the secondary node cannot be successfully registered as the secondary replication site or that the database cannot be successfully restarted on the secondary node, the HANA resource will be flagged as Failed (OSF) on the corresponding node. At this point, manual intervention is typically necessary to eliminate the cause of the failure. If the failover of the primary database instance failed, the resource is flagged as faulty Failed (OSF) and remains in this state until a manual in-service operation is performed by an administrator.

Test Machine Failure of the Secondary Node (reboot -f, power off)

If an error causes the secondary node to fail, the resource remains Active (ISP) on the primary node but SAP HANA System Replication is disrupted. Once the secondary node is restarted and LifeKeeper is active, the secondary database instance is automatically restarted as a replication target.

Test Machine Failure of the Primary Node (reboot -f, power off)

If an error causes the primary node to fail, a failover of the HANA resource hierarchy to the secondary

node is initiated. A takeover of SAP HANA System Replication is performed on the secondary node and the previous secondary replication site becomes the new primary replication site. Once the faulty node is restarted and LifeKeeper is active, the node is registered as a secondary replication site and the database instance is automatically restarted as a replication target.

6.17.6. SAP HANA Resource Hierarchy Administration

Important Note: Unless otherwise noted in this guide, all administrative tasks for a LifeKeeper-protected SAP HANA Database should be performed through LifeKeeper. Performing administrative actions such as stopping the database or disabling SAP HANA System Replication outside of LifeKeeper while the SAP HANA resource is Active/ISP in LifeKeeper will result in LifeKeeper taking action to place the database back into its expected running state.

Switchover of the SAP HANA Resource

When a switchover of the primary database instance is initiated, the SAP HANA Recovery Kit performs the following steps:

- The database instance is stopped on the previous primary node
- A takeover of SAP HANA System Replication is executed on the new primary node (i.e., the previous secondary node)
- The new secondary node (i.e., the previous primary node) is re-registered as the secondary SAP HANA System Replication site
- The database instance is started on the new secondary node

If a message similar to the following:

```
ERROR:hana:restore:HANA-SPS_HDB00:136266:The resource HANA-SPS_HDB00
protecting SAP HANA database HDB00 is not in sync. To protect the data
LifeKeeper will not restore the resource on $me. Please restore the resource
on the previous source server to allow the resync to complete.
```

is displayed while bringing the SAP HANA resource in-service, this means that SAP HANA System Replication was not in-sync when the primary database instance was stopped. Therefore data may exist on the primary database server which has not yet been replicated to the secondary database server. For this reason, LifeKeeper will not allow the secondary server to take over the primary replication role. The recommendation in this scenario is to bring the SAP HANA resource hierarchy back in-service on the previous primary server and allow the resynchronization to complete.

If the previous primary server is down and cannot be recovered, the SAP HANA resource can be forced online on server where the data is out-of-sync, but **this will result in a loss of all data that has not yet been replicated from the previous primary server**. If it is determined by a database administrator that this data loss is acceptable or unavoidable, the out-of-sync data flag can be manually removed with the following command:

```
/opt/LifeKeeper/bin/flg_remove -f '!HANA_DATA_OUT_OF_SYNC_<Tag>'
```

where <Tag> is the SAP HANA resource tag name in LifeKeeper (e.g., HANA-SPS_HDB00). After removing the out-of-sync data flag, reattempt the in-service operation for the HANA resource. Once brought in-service, the database will take over the primary SAP HANA System Replication role and all data on the previous primary server that has not been replicated will be lost. Once the previous primary server is repaired and brought back online, it will be registered as the secondary system replication site and the database will be restarted as the replication target.

Stopping the SAP HANA Database

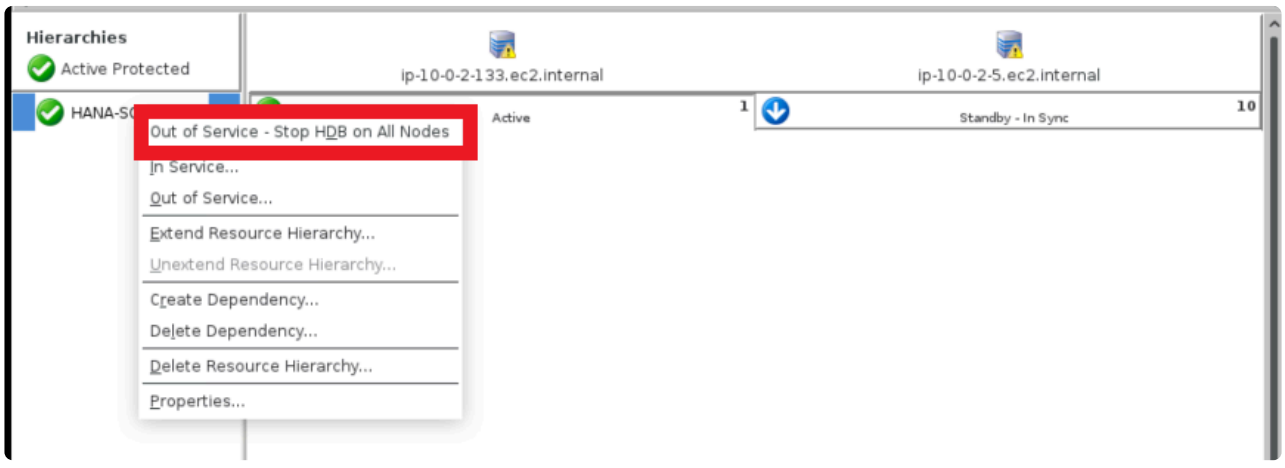
When the SAP HANA resource is taken out of service in LifeKeeper, only the primary database instance is stopped. The secondary database instance is kept running to minimize downtime during switchover or failover of the HANA resource hierarchy.

Stopping all SAP HANA Databases (Maintenance Mode)

When this option is chosen the primary HANA resource is taken out of service and all of the HANA database instances in a HANA resource cluster will be stopped. This option must be executed with utmost care, as it brings the possibility of a quick failover/switchover to the backup machine. **Note:** This option should only be chosen in the event that the secondary database instance must also be stopped (e.g. during the maintenance window).

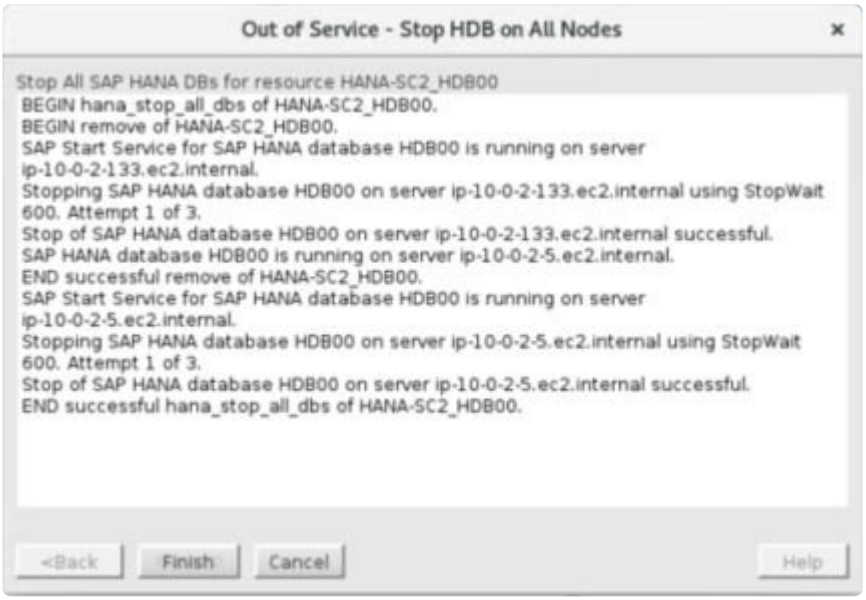
To use this option perform the following steps on HANA resource hierarchy:

1. Right click either on HANA resource under the left hand panel or an in-service server and choose the option Out of Service – Stop HDB on All Nodes.

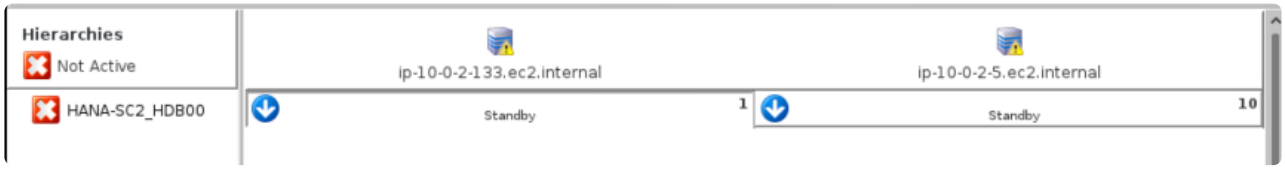


2. Verify the HANA resource and follow the instructions given in the dialog box. Click on **Stop All SAP HANA DBs** to start the process.

3. Once the process finishes, click **Finish** to complete the process.



4. The final state of SAP HANA resource will appear as shown below:



5. Once all of the maintenance activities are complete, bring the SAP HANA resource hierarchy in service on the last primary system.

6.17.6.1. Changing Replication and Operation Modes

Warning: While the replication and operation modes may be changed for the secondary replication site, the user should never perform a manual takeover or switchover of SAP HANA System Replication outside of LifeKeeper. Doing so will result in an error state in which the SAP HANA System Replication modes do not align with the modes that LifeKeeper is expecting on the Active/Standby servers. This error state is represented in the LifeKeeper GUI with the following resource states:

While in this state, all monitoring for the SAP HANA resource will be suspended until the resource is brought in-service in LifeKeeper on the server that is intended to be the primary replication site.

Changing the Replication Mode

The SAP HANA System Replication mode can be changed by using the “hdbnsutil -sr_changemode” command, even while the database is running on the secondary replication site.

1. Execute the following command on the secondary replication site:

   ```
   su - <sid>adm -c “hdbnsutil -sr_changemode --mode=[sync|syncmem|async]”
   ```

where `<sid>` is the lower-case SAP System ID for the SAP HANA installation and the desired replication mode (sync, syncmem, or async) is provided in the `–mode` option. The database can be running or stopped on the secondary site when this command is executed.

2. To confirm that the replication mode was changed successfully, execute the following command on the secondary replication site:

   ```
   su - <sid>adm -c “hdbnsutil -sr_state”
   ```

   and verify that the “mode” parameter has been updated to the new replication mode.

After the next quickCheck interval (default 2 minutes), LifeKeeper will automatically detect the change and update the info fields for all equivalent SAP HANA resources to contain the new replication mode. This can be verified by inspecting the info field for the HANA resource using the following command:

```
/opt/LifeKeeper/bin/ins_list -d <HostName> -t <Tag> -f: | cut -d: -f6 | tr '\002' ':'
```

where <HostName> is the host name of the server to obtain information on and <Tag> is the tag name of the SAP HANA resource in LifeKeeper (e.g., HANA-SPS_HDB00) on the given server.

The currently stored secondary replication mode is the third field in the output of this command:

```
SPS:HDB00:sync:shiba:logreplay
```

Changing the Operation Mode

Note: Changing the operation mode from logreplay or logreplay_readaccess to delta_datashipping will require a full data shipping from the primary site to the secondary site when replication resumes. This full data shipping will be performed automatically by SAP HANA when one of these operation mode changes is detected.

Changing the operation mode in SAP HANA System Replication typically requires stopping the database on the secondary site, re-registering it with the new operation mode, and restarting the database on the secondary site. In order to prevent LifeKeeper from automatically restarting the database on the secondary site during this process, it is recommended to suspend monitoring for the HANA resource until the database has been successfully re-registered with the new operation mode.

1. Suspend monitoring of the HANA resource by creating the corresponding nomonitor flag on the server where the resource is currently Active (ISP):

   ```
   /opt/LifeKeeper/bin/flg_create -f 'nomonitor_<Tag>'
   ```

   where <Tag> is the HANA resource tag in LifeKeeper on that server (e.g., HANA-SPS_HDB00). Once finished with this process, it is very important to remember to resume monitoring of the resource by removing the *nomonitor* flag with the command:

   ```
   /opt/LifeKeeper/bin/flg_remove -f 'nomonitor_<Tag>'
   ```

   Warning: Failure to remove this flag after performing the maintenance operations will cause any failures of the SAP HANA database to go undetected by LifeKeeper.

2. Ensure that the SAP HANA database is stopped on the backup server by executing the following command:

   ```
   su - <sid>adm -c "sapcontrol -nr <HDB Inst# > -function StopSystem HDB"
   ```

   where <HDB Inst#> is the instance number for the SAP HANA database instance (e.g., for an instance named HDB00, the instance number is 00).

3. Execute the following command on the backup server with the desired replication and operation modes to re-register it as a secondary replication site:

   ```
   su - <sid>adm -c "hdbnsutil -sr_register --name=<SecondarySiteName>
   ```

```
--remoteHost=<PrimaryHost> --remoteInstance=<PrimaryInst#>
--replicationMode=[sync|syncmem|async]
--operationMode=[delta_datashipping|logreplay|logreplay_readaccess]"
```

where `<SecondarySiteName>` is the alias to be used by SAP HANA System Replication to identify the secondary replication site, `<PrimaryHost>` is the host name of the server which is currently registered as the primary replication site, and `<PrimaryInst#>` is the instance number for the HDB instance on the primary replication site. **Note:** It is not necessary to unregister the secondary site with the "hdbnsutil -sr_unregister" command before re-registering it with a new replication or operation mode.

4. To verify that the backup server was successfully re-registered as a secondary replication site with the new operation mode, execute the following command on the backup server:

   ```
   su - <sid>adm -c "hdbnsutil -sr_state"
   ```

   and verify that "is secondary/consumer system" is true and that the parameter "operation mode" has been updated to the new operation mode.

5. Once the backup server has been successfully registered as a secondary replication site, start the database by executing the following command on the backup server:

   where `su - <sid>adm -c "sapcontrol -nr <HDB Inst#> -function StartSystem HDB"`

   where `<HDB Inst#>` is the instance number for the SAP HANA database instance.

6. Once the process is complete, remove the *nomonitor* flag that was created in Step 1 in order to resume LifeKeeper monitoring of the HANA resource:

   where `/opt/LifeKeeper/bin/flg_remove -f 'nomonitor_<Tag>'`

   Warning: Failure to remove this flag will cause any failures of the SAP HANA database to go undetected by LifeKeeper.

After the next quickCheck interval (default 2 minutes), LifeKeeper will automatically detect the change and update the info fields for all equivalent SAP HANA resources to contain the new operation mode. This can be verified by inspecting the info field for the HANA resource using the following command:

```
/opt/LifeKeeper/bin/ins_list -d <HostName> -t <Tag> -f: | cut -d: -f6 | tr '\002' ':'
```

where `<HostName>` is the host name of the server to obtain information on and `<Tag>` is the tag name of the SAP HANA resource in LifeKeeper (e.g., HANA-SPS_HDB00) on the given server.

The currently stored secondary replication mode is the fifth field in the output of this command:

```
SPS:HDB00:sync:shiba:logreplay
```

6.17.6.2. Resolving Split Brain Scenarios

A “split brain” scenario occurs when the SAP HANA database is running and configured as the primary SAP HANA Replication site on multiple cluster nodes. In this situation, LifeKeeper will suspend all monitoring of the HANA database until the issue is manually resolved by a database administrator.

There are two common types of split brain scenarios which may occur for an SAP HANA resource hierarchy.

- **LifeKeeper HANA Resource Split Brain**: The HANA resource is Active (ISP) in LifeKeeper on multiple cluster nodes. This situation is typically caused by a temporary network outage affecting the communication paths between cluster nodes.

- **SAP HANA System Replication Split Brain**: The HANA resource is Active (ISP) on the primary node and Standby (OSU) on the backup node in LifeKeeper, but the database is running and registered as the primary replication site on both nodes. This situation is typically caused by either a failure to stop the database on the previous primary node during failover, having Autostart enabled for the database, or a database administrator manually running “hdbnsutil -sr_takeover” on the secondary replication site outside of LifeKeeper.

Recommendations for resolving each type of split brain scenario are given below.

LifeKeeper HANA Resource Split Brain Resolution

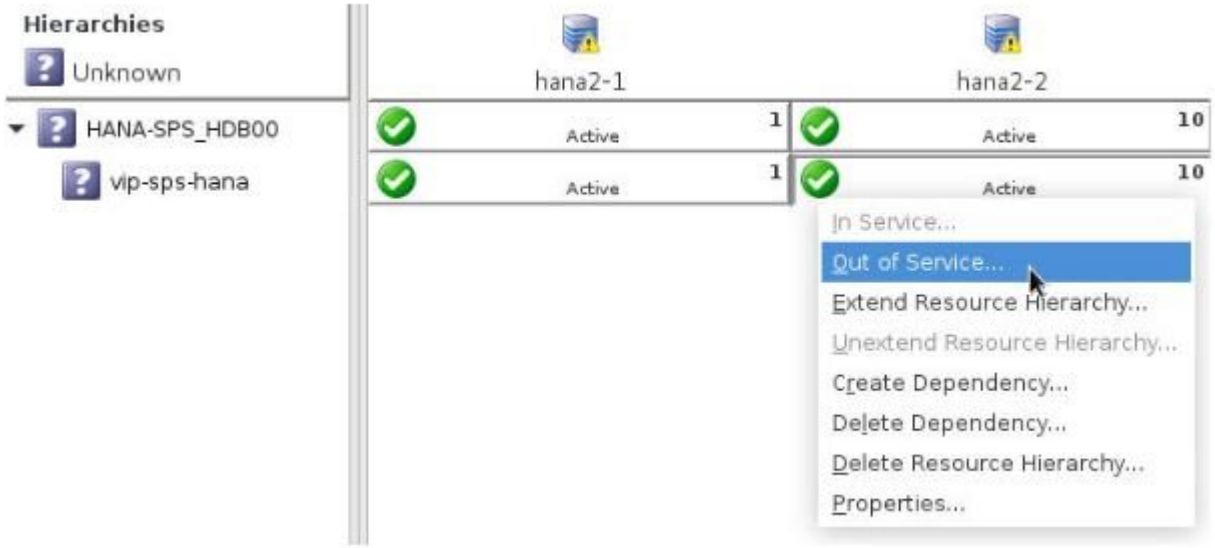
While in this split brain scenario, a message similar to the following is logged and broadcast to all open consoles every quickCheck interval (default 2 minutes) until the issue is resolved.

```
EMERG:hana:quickCheck:HANA-SPS_HDB00:136363:WARNING: A temporary communication failure has occurred between servers hana2-1 and hana2-2. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please take one of the following resource hierarchies out of service: HANA-SPS_HDB00 on hana2-1 or HANA-SPS_HDB00 on hana2-2. The server that the resource hierarchy is taken out of service on will become the secondary SAP HANA System Replication site.
```

Recommendations for resolution:

1. Investigate the database on each cluster node to determine which instance contains the most up-to-date or relevant data. This determination must be made by a qualified database administrator who is familiar with the data.

2. The HANA resource on the node containing the data that needs to be retained will remain Active (ISP) in LifeKeeper, and the HANA resource hierarchy on the node that will be re-registered as the secondary replication site will be taken entirely out of service in LifeKeeper. Right-click on each leaf resource in the HANA resource hierarchy on the node where the hierarchy should be taken out of service and click **Out of Service** …



> It is important in this step that the entire SAP HANA resource hierarchy (including the virtual IP resource, if one exists) is taken out of service on the node that will be re-registered as the secondary replication site.

3. Once the SAP HANA resource hierarchy has been successfully taken out of service, LifeKeeper will re-register the Standby node as the secondary replication site during the next quickCheck interval (default 2 minutes). Once replication resumes, any data on the Standby node which is not present on the Active node will be lost. Once the Standby node has been re-registered as the secondary replication site, the SAP HANA hierarchy has returned to a highly-available state.

SAP HANA System Replication Split Brain Resolution

While in this split brain scenario, a message similar to the following is logged and broadcast to all open consoles every quickCheck interval (default 2 minutes) until the issue is resolved.

```
EMERG:hana:quickCheck:HANA-SPS_HDB00:136364:WARNING: SAP HANA database HDB00
is running and registered as primary master on both hana2-1 and hana2-2.
Manual intervention is required in order to minimize the risk of data loss. To
resolve this situation, please stop database instance HDB00 on hana2-2 by
running the command 'su - spsadm -c "sapcontrol -nr 00 -function Stop"' on
that server. Once stopped, it will become the secondary SAP HANA System
Replication site.
```

Recommendations for resolution:

1. Investigate the database on each cluster node to determine whether important data exists on the Standby node which does not exist on the Active node. If important data has been committed to the database on the Standby node while in the split brain state, the data will need to be manually copied to the Active node. This determination must be made by a qualified database administrator who is familiar with the data.

2. Once any missing data has been copied from the database on the Standby node to the Active node, stop the database on the Standby node by running the command given in the LifeKeeper warning message:

   ```
   su - <sid>adm -c "sapcontrol -nr <Inst#> -function Stop"
   ```

   where <sid> is the lower-case SAP System ID for the HANA installation and <Inst#> is the instance number for the HDB instance (e.g., the instance number for instance HDB00 is 00).

3. Once the database has been successfully stopped, LifeKeeper will re-register the Standby node as the secondary replication site during the next quickCheck interval (default 2 minutes). Once replication resumes, any data on the Standby node which is not present on the Active node will be lost. Once the Standby node has been re-registered as the secondary replication site, the SAP HANA hierarchy has returned to a highly-available state.

6.17.7. SAP HANA Troubleshooting

The Message Catalog provides a list of all messages that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. This full listing may be searched for any error code received, or you may go directly to the SAP HANA Recovery Kit Message Catalog which contains a list of all messages that may be encountered while utilizing the SAP HANA Recovery Kit.

6.18. SAP MaxDB Recovery Kit Administration Guide

SAP DB / MaxDB (SAP DB) is a SQL-based, industrial-strength database system that can be deployed for a wide array of purposes. It is highly scalable, platform-independent and provides full transaction support. The database system was originally owned by SAP but has since been released to the Open Source community.

The SAP DB / MaxDB Recovery Kit provides fault resilient protection for SAP DB databases in an SPS for Linux environment.

Document Contents

This guide includes the following topics to help you successfully define and manage your SAP DB hierarchy:

- SAP DB / Max DB Recovery Kit Requirements. Lists the hardware and software necessary to properly set up, install and operate the SAP DB / Max DB Recovery Kit.
- Overview. Describes the SAP DB / Max DB Recovery Kit's features and functionality.
- Configuration Considerations. Contains information to consider before you install and configure the SAP DB / Max DB Recovery Kit.
- Configuring SAP DB with SAP. Provides instructions for installing and configuring the SAP DB software and SAP software.
- Resource Configuration Tasks. Describes the various functions you may perform on your hierarchies using the LifeKeeper GUI: create, extend, delete and unextend.
- Hierarchy Administration. Provides important recommendations for ongoing administration of the SAP DB hierarchy.
- Troubleshooting. Lists and describes the error messages associated with the SAP DB / Max DB Recovery Kit.
- Appendix. Provides requirements and instructions for setting up raw devices for use with the SAP DB / Max DB Recovery Kit.

SIOS Protection Suite Documentation

The following SPS product documentation is available from the SIOS Technology Corp. website:

SPS for Linux Release Notes

SPS for Linux Technical Documentation

SPS for Linux Installation Guide

Optional Recovery Kit Documentation

This documentation, along with documentation associated with optional LifeKeeper Recovery Kits, is provided on the SIOS Technology Corp. website at:

http://docs.us.sios.com/

and from the Help menu in the LifeKeeper GUI.

SAP DB Documentation

You can find the SAP DB/MaxDB documentation, including the Installation Guide, User Manual and Reference Manual, at the following locations on the web:

http://maxdb.sap.com/documentation/

6.18.1. SAP DB / MaxDB Recovery Kit Hardware and Software Requirements

Your SPS configuration must meet the following requirements prior to the installation of SPS for Linux SAP DB / MaxDB Recovery Kit. Please see the SPS for Linux Installation Guide for specific instructions regarding the installation and configuration of your SPS hardware and software.

Hardware Requirements

- **Servers** – Servers should be configured in accordance with the requirements described in the SPS for Linux Installation Guide and the SPS for Linux Release Notes.

- **IP Network Interface Cards** – Each server requires at least one Ethernet TCP/IP-supported network interface card. Remember, however, that an SPS cluster requires at least two communications paths. Two separate LAN-based communication paths using dual independent sub-nets are recommended for heartbeats, and at least one of these should be configured as a private network. Using a combination of TCP and TTY heartbeats is also supported.

Software Requirements

- **TCP/IP Software** – Each server in your SPS configuration requires TCP/IP Software.

- **SAP DB/MaxDB Software** – Supported versions of SAP DB/MaxDB Software listed in “SIOS Protection Suite for Linux Support Matrix” should be installed.

  Note: The same version of the SAP DB software must be installed on all servers in the cluster.

- **SPS Software** – It is imperative that you install the same version of the SPS software and apply the same versions of the SPS software patches to each server in your cluster.

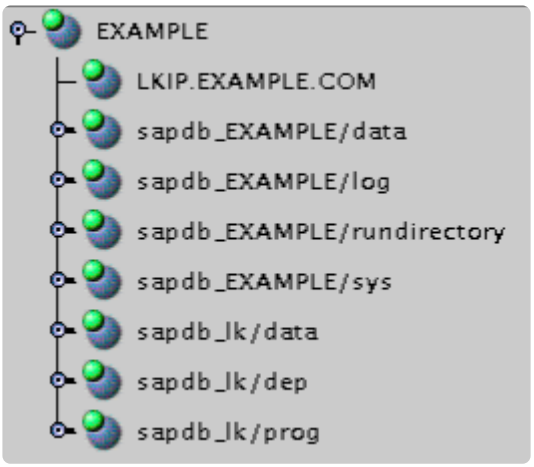
- **SPS for Linux SAP DB / Max DB Recovery Kit** – The SAP DB / Max DB Recovery Kit is provided on the SPS for Linux Installation Image File (Steeleye-lkSAPDB). It is packaged, installed and removed via Red Hat Package Manager, rpm.

6.18.2. SAP MaxDB Recovery Kit Overview

The SPS for Linux SAP DB / Max DB Recovery Kit provides a mechanism for protecting SAP DB instances within SPS. The SAP DB software, SPS Core and SAP DB / Max DB Recovery Kit are installed on two or more servers in a cluster. Once the SAP DB database instance is under SPS protection, clients connect to the database using an SPS protected IP address. The SPS protected IP address must be created separately and a dependency made manually between the parent SAP DB resource instance and the child IP address resource. In the event that the SAP DB server fails, SPS will first attempt to recover it on the local server. If the local recovery fails, then SPS will fail over to a backup server.

6.18.2.1. SAP DB / MaxDB Resource Hierarchy

The following example shows a typical SAP DB / MaxDB resource hierarchy:



The dependencies in the above example correspond to the following protected resources:

Resource	SAP DB Software Component
LKIP.EXAMPLE.COM	protects the switchable IP address used for client connections
sapdb_EXAMPLE /data	protects the database data device space for the EXAMPLE database
sapdb_ EXAMPLE /log	protects the database log device space for the EXAMPLE database
sapdb_ EXAMPLE /rundirectory	protects the database RUNDIRECTORY for the EXAMPLE database
sapdb_ EXAMPLE /sys	protects the database system device space for the EXAMPLE database
sapdb_lk/data	protects the independent data path
sapdb_lk/dep	protects the dependent path
sapdb_lk/prog	protects the independent program path

In the event of failover, SPS will bring the file system, IP address and database resources (including all the resource dependencies) in service on a backup server. Clients will be disconnected and will need to re-connect to the server. Any SQL statement that has not been committed will need to be re-entered.

6.18.3. SAP DB / MaxDB Configuration Considerations

This section contains information that you should consider before you start to configure and administer the SAP DB / Max DB Recovery Kit.

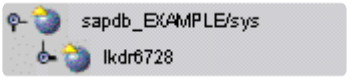
6.18.3.1. Using Raw I/O with SAP DB / MaxDB

If you plan to use SAP DB / MaxDB with raw devices, you must install the SPS Raw I/O Recovery Kit from the SPS Installation Image file. You must also properly set up the raw I/O devices prior to use. See the Appendix for instructions.

6.18.3.2. Using SAP DB / Max DB Mirrored File Systems with DataKeeper

The SAP DB / Max DB Recovery Kit supports the use of DataKeeper as a device space. In addition, the SAP DB / MaxDB software can be installed on mirrored file systems.

For example, a dependent file system for an SAP DB / MaxDB resource would look similar to the following, which shows a file system for the system device space and its dependency, the DataKeeper resource mirror.



6.18.3.3. SAP DB / MaxDB Active/Standby Considerations

In an Active/Standby configuration, the backup server is not actively running the SAP DB / MaxDB but stands by in case the primary server experiences a failure. The following scenarios provide specific requirements that must be adhered to when protecting an SAP DB resource instance in Active/Standby configurations.

Active/Standby Scenarios

The typical Active/Standby configurations are explained below in Scenarios 1 and 2.

Scenario 1

The SAP DB *IndepDataPath, IndepProgPath* and *DependPath* are installed to **one or more shared file systems on the primary server**.

- The paths *IndepDataPath, IndepProgPath,* and *DependPath* must be shared between all servers that will protect the resource instance.
- The directory structure under `/usr/spool/sql` must be replicated manually to each server in the cluster. This directory structure should not be located on shared storage since it must be accessible from the target server during resource extend operations. (*Please disregard for MaxDB 7.8 as this directory no longer exists*.)
- The registry file `/etc/opt/sdb` must exist on each server in the cluster for MaxDB 7.5.x versions. This file should not be located on shared storage since it must be accessible from the target server during resource extend operations.
- The database instance data device spaces (`data devspaces`), log device spaces (`log devspaces`) and system device spaces (`sys devspaces`) must reside on a shared disk (either shared file system or shared raw device).

Scenario 2

The SAP DB *IndepDataPath* and *IndepProgPath* are installed locally on both servers. The SAP DB *DependPath* can be installed locally or on a shared file system on the primary server.

- The directory structure under `/usr/spool/sql` must exist on all servers with the same permissions as well as the same owner and group. (*Please disregard for MaxDB 7.8 as this directory no longer exists*.)
- The registry file `/etc/opt/sdb` must exist on each server in the cluster for MaxDB 7.5.x versions. This file should not be located on shared storage since it must be accessible from the target server during resource extend operations.

- The database instance data device spaces (`data devspaces`), log device spaces (`log devspaces`) and system device spaces (`sys devspaces`) must reside on a shared disk (either shared file system or shared raw device).

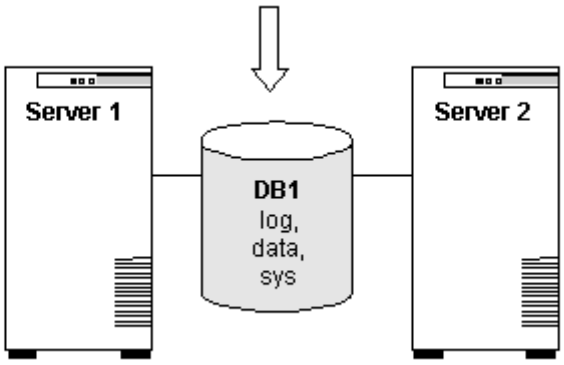
- The database instance run directory (`RUNDIRECTORY`)must be located on shared storage. The value of `RUNDIRECTORY` can be modified via the DBMCLI command `param_directput`. If the value of `RUNDIRECTORY` is modified after the database is created, the database instance must be stopped and restarted to complete the parameter update.

- The database instance config (`<IndepDataPath>/config`) directory structure must exist in the same location on all servers in the cluster where the database instance will be protected. In addition, the parameter files for the database instance must be copied from the template (or primary) server to all backup servers in the cluster. The parameter files must be redistributed to all servers in the cluster after any parameter has been updated. The required files are:

  ```
  config/<db instance>

  config/<db instance>.<01>…<N> (Note: There may be multiple .<number>
  files.)

  config/<db instance>.cfg

  config/<db instance>.pab

  config/<db instance>.upc
  ```

6.18.3.3.1. Active/Standby Configuration Example

IndepData = /shr1/data
IndepPrograms = /shr1/programs
DependPath = /shr1/depend



Configuration Notes:

- Both servers use the *IndepProgPath, DependPath* and *IndepDataPath* on the shared storage.
- The database instance `DB1` is located on the shared storage. This includes all log device spaces, data device spaces and system device spaces.
- The directory structure `/usr/spool/sql` has been replicated to *Server 2*. All entries for the `SAP_DBTech.ini` file have been added. (*Please disregard for MaxDB 7.8 as this directory and file no longer exist*.)
- *Server 2* cannot access files and directories on the shared disk while *Server 1* is active.

6.18.3.4. SAP DB / MaxDB Active/Active Considerations

In an Active/Active configuration, each server is actively running one SAP DB instance while acting as a backup for the other server in case of failure. The following scenario provides specific requirements that must be adhered to in sequential order when protecting an SAP DB resource instance in an Active/Active configuration.

Active/Active Scenario

The SAP DB *IndepDataPath, IndepProgPath* and *DependPath* are installed locally on both servers.

- The directory structure under `/usr/spool/sql` must exist on all servers with the same permissions as well as the same owner and group. (*Please disregard for MaxDB 7.8 as this directory no longer exists*.)

- The registry file `/etc/opt/sdb` must exist on each server in the cluster for MaxDB 7.5.x versions. This file should not be located on shared storage since it must be accessible from the target server during resource extend operations.

- The database instance data device spaces (`data devspaces`), log device spaces (`log devspaces`) and system device spaces (`sys devspaces`) must reside on a shared disk (either shared file system or shared raw device).

- The database instance run directory (`RUNDIRECTORY`) must be located on shared storage. The value of `RUNDIRECTORY` can be modified via the DBMCLI command `param_directput`. If the value of `RUNDIRECTORY` is modified after the database is created, the database instance must be stopped and restarted to complete the parameter update.

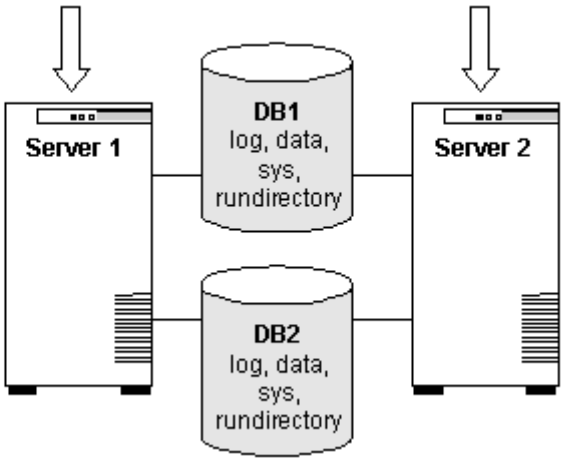
- The database instance config (`<IndepDataPath>/config`) directory structure must exist in the same location on all servers in the cluster where the database instance will be protected. In addition, the parameter files for the database instance must be copied from the template (or primary) server to all backup servers in the cluster. The parameter files must be redistributed to all servers in the cluster after any parameter has been updated. The required files are:

  ```
  config/<db instance>

  config/<db instance>.<01>…<N> (Note: There may be multiple .<number>
  files.)

  config/<db instance>.cfg

  config/<db instance>.pab
  ```

```
config/<db instance>.upc
```

6.18.3.4.1. Active/Active Configuration Example

IndepData = /usr/sapdb/data IndepPrograms = /usr/sapdb/prog DependPath = /usr/sapdb/dep	IndepData = /usr/sapdb/data IndepPrograms = /usr/sapdb/prog DependPath = /usr/sapdb/dep



Configuration Notes:

- The *IndepDataPath, IndepProgPath* and *DependPath* are locally installed on both servers.
- Each database is configured on separate shared disks. The database instance includes all log device spaces, system device spaces and data device spaces.
- The `RUNDIRECTORY` for each database instance is also on a shared disk.
- The database configuration files for DB1 have been copied to *Server 2* and the database configuration files for DB2 have been copied to Server 1. The configuration files are located at `<IndepDataPath>/config/<db instance>`.
- Initially, *Server 1* runs `DB1` and *Server 2* runs `DB2`. In a switchover situation, one server can run both databases.

6.18.4. Configuring SAP DB / MaxDB with SPS

The following sequence is recommended for installing and configuring the SAP DB / MaxDB database and SPS software. Each of these steps links to detailed tasks that follow.

Install the SAP DB / MaxDB Software

Create the SAP DB / MaxDB Database

Create the User_Key

Install the SPS Core and SAP DB / Max DB Recovery Kit

After you have performed these tasks, you will be ready to create the SPS resource hierarchy to protect your SAP DB / MaxDB database.

6.18.4.1. Install the SAP DB / MaxDB Software

Install the SAP DB /MaxDB software on all servers in the cluster using identical parameters/settings. Refer to the *SAP DB Installation Guide* for details. The following are additional recommendations to ensure that SPS will work with SAP DB / MaxDB:

- A non-root system user (OS User) must exist on all servers as follows:
- This OS User should be designated as the owner of the SAP DB / MaxDB software installation and subdirectories or have adequate permissions on the software installation path and subdirectories as required to manage a database instance.

- This OS User must have authority to use the DBMCLI and x_server utilities. The OS User must be able to start and stop the vserver using the x_server commands.

- The OS User name should contain alpha-numeric characters only.

- The User ID and Group ID of this OS User must be identical on all servers.

  - The SAP DB / MaxDB client software packages must be installed. These packages must include the SAP DB / MaxDB DBMCLI client utility, and the SAP DB / MaxDB x_server utility.

  - Each SPS server containing an SAP DB resource hierarchy must have identical service entries in the `/etc/services` file for the SAP DB instance.

6.18.4.2. Create the SAP DB / MaxDB Database

Follow the instructions in your *SAP DB / MaxDB User Manual* to create your database. In addition, please note the following recommendations:

- There must be a DBM operator with authority for starting, stopping, obtaining status and obtaining database parameters via client utilities.
- The database instance data device spaces (`data devspaces`), log device spaces (`log devspaces`) and system device spaces (`sys devspaces`) must reside on a shared disk (either shared file system, or shared raw device).
- The SAP DB / MaxDB database name should contain alphanumeric characters only.
- A User_Key is required for use by the SAP DB / Max DB Recovery Kit during operation with the DBMCLI utility. See Create the User_Key for required parameters.
- After creating your database, you should disable automatic startup of the SAP DB / MaxDB database instance. Once under SPS protection, SPS will handle the start and stop of the database.
- The *`SAP_DBTech.ini`* file must exist on all servers in */usr/spool/sql/ini*. If this file does not exist, several SAP DB / MaxDB utilities may return erroneous results. This will also affect the behavior of SPS during resource create and extension. In an Active/Standby configuration, you must manually copy this file to the backup server. (*Please disregard for MaxDB 7.8 as this directory and file no longer exist*.)
- For version 7.5.x, verify the `databases.ini`file exists on all servers in the *`IndepDataPath/config`* directory.

6.18.4.3. Create the User_Key

The SAP DB / MaxDB instance requires several options for a user to successfully access a database instance. These required pieces of information must be passed in to the SAP DB / MaxDB tool being used to access the database instance. The SAP DB / MaxDB software includes the **xuser** tool for simplifying the specification of many required options to SAP DB / MaxDB tools. The **xuser** tool allows you to predefine and save user data. Once this data has been saved, it can be used when you call the **DBMCLI** or other tools requiring user options. This predefined user data is stored in a user key (User_Key). An individual user can manage and maintain several user keys for the same or multiple databases. Each key includes a combination of options including username/password, database name as well as database server name.

The SAP DB / Max DB Recovery Kit requires a valid User_Key for each database instance under protection. This User_Key must be created and accessible by the OS User that owns the database instance. The user information specified for each User_Key must be for a DBM operator with the following permissions:

- DBStart
- DBStop
- DBInfoRead
- ParamRead

The User_Key can be generated using the command:

```
xuser -b <file name>
```

where `<file name>` is the name of a file containing the valid XUSER entries as follows:

Parameter	Parameter Definition
USERKEY	Unique name for the User_Key
USERID	User name of the dbm operator
PASSWORD	Password of the user
SERVERDB	Name of the database instance that this key will refer to
SERVERNODE	The name of the server where the database is running (this should be the DNS or host file entry for the SPS protected IP)
SQLMODE	This determines what SQL dialects are compatible
CACHELIMIT	This determines the cache limits for a given session
TIMEOUT	Time in seconds before terminating an inactive session (-1 is the default)
ISOLATION	Determines the isolation level used for locks that affect the user (-1 is the default)

DB_LOCAL	Specifies the database locale

Refer to the *SAP DB / MaxDB User Manual* for more information on parameters. Once proper entries have been specified, use the **xuser** tool to generate the *.XUSER.62* file in the OS User home directory. A sample XUSER file is included below containing two keys (an entry must exist for the DEFAULT User_Key).

```
DEFAULT
NULLDB
NULLDB
NULLDB
LKIP.example.com
INTERNAL
-1
-1
-1
my_locale
LK_USERKEY
LKDBMOPER
LKDBMPASSWD
DB1
LKIP.example.com
INTERNAL
-1
-1
-1
en_US
```

This example XUSER file specifies that two user keys be created, DEFAULT and LK_USERKEY. Once the **xuser** tool has been run to generate the User_Key(s), the file specified for use by the **xuser** tool should be deleted.

6.18.4.4. Install the SPS Software

Once you have installed the SAP DB / MaxDB software, created your database and created the User_Key, you are ready to install the SPS Core software and any required patches followed by the SAP DB / Max DB Recovery Kit. Also, if you plan to use SAP DB / MaxDB with raw devices, you must install the SPS Raw I/O Recovery Kit from the SPS Installation Image file. See the Appendix for requirements and instructions on setting up raw devices.

Refer to the SPS for Linux Installation Guide for details on installing the SPS packages.

6.18.5. SAP DB / MaxDB Resource Configuration Tasks

Once you have completed the setup tasks described in the previous section, you are ready to create and extend your SAP DB resource hierarchies.

The following tasks are available for configuring the SPS for Linux SAP DB / Max DB Recovery Kit:

- **Create Resource Hierarchy** – Creates an SAP DB resource hierarchy
- **Delete Resource Hierarchy** – Deletes an SAP DB resource hierarchy
- **Extend Resource Hierarchy** – Extends an SAP DB resource hierarchy from the primary server to the backup server
- **Unextend Resource Hierarchy** – Unextends (removes) an SAP DB resource hierarchy from a single server in the SPS cluster

Please refer to your SPS for Linux Technical Documentation located on the SIOS Technology website for instructions on configuring your LifeKeeper Core resource hierarchies.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View / Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You may also perform most of the tasks:

- from the toolbar
- by right-clicking on a global resource in the left pane of the status display
- by right-clicking on a resource in the right pane of the status display

Using the right-click method allows you to avoid entering information that is required using the **Edit** menu.

6.18.5.1. Creating an SAP DB Resource Hierarchy

Perform the following steps on the primary server:

1. On the **Edit** menu, select **Server**, then **Create Resource Hierarchy**.

   The ***Create Resource Wizard*** dialog will appear.

2. Select **SAP DB Database** from the drop-down list and click **Enter**.

3. You will be prompted for the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter any error requiring you to correct the previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

Field	Tips
Switchback Type	Choose either intelligent or automatic. This determines how the SAP DB resource will be switched back to the primary server after it comes in-service (active) on the backup server following a failover. Intelligent switchback requires administrative intervention to switch the resource back to the primary server, while automatic switchback occurs as soon as the primary server is back on line and reestablishes SPS communication paths. **Note**: The switchback strategy must match that of the dependent resources to be used by the SAP DB resource.
SAP DB Programs Directory	This field contains by default the SAP DB Program Path found in the **`SAP_DBTech.ini`** file on the corresponding server. You may type in another directory path. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . / (*Please disregard for MaxDB 7.8 as SAP_DBTech.ini no longer exists.*)
SAP DB Instance	This field contains by default the name of the first SAP DB instance found on the system for which no SPS hierarchy exists. The drop-down list shows other database instances that may be available on your system.
SAP DB System User	This is the System User that owns or has permission to execute SAP DB commands. This user must exist on the corresponding server. Enter a valid user name in the selection window.
User_Key	This field contains a default value for the XUSER User_Key. The User_Key is used to store database user data for use with SAP DB Tools. Enter a valid User_Key for the corresponding server, OS User and database instance combination.
SAP DB Database Tag	This is a unique tag name for the new SAP DB database resource on the primary server. The default tag name consists of the SAP DB instance. You may type in another unique tag name. The valid characters allowed for the tag are letters, digits and the following special characters: – _ . /

4. Click **Create**. The ***Create Resource Wizard*** will then create your SAP DB resource hierarchy.

SPS will validate the data entered. If SPS detects a problem, an error message will appear in the information box.

5. You should see a message indicating that you have successfully created an SAP DB resource hierarchy and you must extend that hierarchy to another server in your cluster to achieve failover protection. Click **Next**.

6. Click **Continue**. SPS will then launch the ***Pre-Extend Wizard***. Refer to Step 2 under Extending an SAP DB Resource Hierarchy for details on how to extend your resource hierarchy to another server.

6.18.5.2. Extending an SAP DB Resource Hierarchy

This operation can be started from the **Edit** menu or initiated automatically upon completing the **Create Resource Hierarchy** option, in which case you should refer to Step 2 below.

1. On the **Edit** menu, select **Resource**, then **Extend Resource Hierarchy**. The ***Pre-Extend Wizard*** appears. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the SPS Extend Resource Hierarchy defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults**.

2. The ***Pre-Extend Wizard*** will prompt you to enter the following information.

   Note: The first two fields appear only if you initiated the Extend from the **Edit** menu.

Field	Tips
Template Server	Select the server where your SAP DB resource is currently in service.
Tag to Extend	Select the SAP DB resource you wish to extend.
Target Server	Enter or select the server you are extending to.
Switchback Type	This determines how the SAP DB resource will be switched back to the primary server after it comes in-service (active) on the backup server following a failover. You can choose either intelligent or automatic. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. Note: Remember that the switchback strategy must match that of the dependent resources to be used by the SAP DB resource.
Template Priority	Select or enter a Template Priority. This is the priority for the SAP DB hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. Note: This selection will appear only for the initial extend of the hierarchy.
Target Priority	This is the priority for the new extended SAP DB hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. Note that SPS assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

3. After receiving the message that the pre-extend checks were successful, click **Next**.

4. Depending upon the hierarchy being extended, SPS will display a series of information boxes showing the Resource Tags to be extended, some of which cannot be edited.

5. The ***Extend Wizard*** will prompt you to enter the following information.

SAP DB Programs Directory	This field contains by default the SAP DB Program Path found in the **`SAP_DBTech.ini`** file on the corresponding server. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . / (*Please disregard for MaxDB 7.8 as SAP_DBTech.inino longer exists*.)
User_Key	This field contains a default value for the XUSER User_Key. The User_Key is used to store database user data for use with SAP DB Tools. Enter a valid User_Key for the corresponding server, OS User and database instance combination.
SAP DB Database Tag	This is a unique tag name for the new SAP DB database resource on the target server. The default tag name consists of the SAP DB instance. You may type in another unique tag name. The valid characters allowed for the tag are letters, digits and the following special characters: – _ . /

6. After receiving the message "`Hierarchy extend operations completed`", click **Next Server** to extend the hierarchy to another server, or click **Finish** if there are no other extend operations to perform.

7. After receiving the message "`Hierarchy Verification Finished`", click **Done**.

6.18.5.3. Unextending an SAP DB Resource Hierarchy

To remove a resource hierarchy from a single server in the SPS cluster, do the following:

1. On the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.

2. Select the **Target Server** where you want to unextend the SAP DB resource. It cannot be the server where the resource is currently in service. (*This dialog box will not appear if you selected the **Unextend** task by right-clicking on a resource instance in the right pane*.) Click **Next**.

3. Select the SAP DB hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the **Unextend** task by right-clicking on a resource instance in either pane).

4. An information box appears confirming the target server and the SAP DB resource hierarchy you have chosen to unextend. Click **Unextend**.

5. Another information box appears confirming that the SAP DB resource was unextended successfully. Click **Done** to exit the **Unextend Resource Hierarchy** menu selection.

6.18.5.4. Deleting an SAP DB Resource Hierarchy

To delete an SAP DB resource from all servers in your SPS configuration, complete the following steps:

1. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.

2. Select the name of the **Target Server** where you will be deleting your SAP DB resource hierarchy. (*This dialog will not appear if you selected the **Delete Resource** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance.*)

3. Select the **Hierarchy to Delete**. (*This dialog will not appear if you selected the **Delete Resource** task by right-clicking on a resource instance in the left or right pane.*) Click **Next**.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Next**.

5. Another information box appears confirming that the SAP DB resource was deleted successfully.

6. Click **Done** to exit.

6.18.5.5. Testing Your SAP DB Resource Hierarchy

You can test your SAP DB resource hierarchy by initiating a manual switchover that will simulate a failover of the resource instance from the primary server to a backup server.

Performing a Manual Switchover from the LifeKeeper GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, **Resource** and **InService**. For example, an in-service request executed on a backup server causes the SAP DB resource hierarchy to be placed in service on the backup server and taken out of service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the Out of Service request, the resource hierarchy is taken out of service without bringing it in service on the other server.

> **IMPORTANT**: After bringing your resource hierarchy in service on the backup server, you should attempt to connect to the databases, especially when using raw devices as device spaces. This is necessary to ensure that all disk partitions are visible on the backup servers and the raw bindings are being established correctly.
>
> If the raw bindings have not been established on the backup servers, it is most likely caused by the fact that new partitions were created on the primary server and added to the configuration, but the partition tables have not yet been updated on the backup servers.
>
> The solution is to reboot the backup servers so that the partition tables are updated correctly.

6.18.6. SAP DB Resource Hierarchy Administration

The following tasks may be required after your resource hierarchies have been created.

Modifying User_Keys

Modifying OS User

Updating Parameters

6.18.6.1. Modifying User_Keys

If the User_Key for an existing hierarchy needs to be changed, the hierarchy must be deleted and recreated.

6.18.6.2. Modifying OS User

If the OS User that owns the database instance needs to be changed, the hierarchy must be deleted and recreated.

6.18.6.3. Updating SAP DB Parameters

When database parameters are updated for an SAP DB instance, it is necessary to ensure that the updated parameter files are redistributed to all servers protecting the instance. If the `IndepDataPath` is on a shared disk, then all servers protecting the instance will be updated automatically.

6.18.7. SAP DB / MaxDB Troubleshooting

General Tips

The following error messages are not generated by the SAP DB / Max DB Recovery Kit but may be encountered while using the recovery kit

Error Message	Solution
Unable to create pipe */usr/spool/sql/fifo/<db instance>*	The directory */usr/spool/sql* must have proper permissions to allow access for system user that owns the database instance.
Open device space *<dev>* permission denied	The device spaces on the backup and primary must have the same owner as well as the same user and group permissions.
Unable to set uid on startup	The setuid bit on *<DependPath>/pgm/dbmsrv* must be set and the owner of the file must be the SAP DB system user.
runtime environment error	There are several possible causes with different solutions: • • The database instance parameter and configuration files do not exist. Create the database parameter files or copy the files from the template server. • • The database has encountered a library problem. The server and software installation combination may require the use of the library *libpthread-0.8.so.* Consult the SAP DB documentation for instructions. • • The database instance environment has been corrupted. The processes must be manually killed. Then attempt to restore the resource to the in-service state.
open Registry: Permission denied	The directory */usr/spool/sql/ini* should be owned by the system user and group that owns the SAP DB software. In addition, the user and group must also have read/write permissions on the directory.
ERR_USRREAD: could not read user data	The config files from *<IndepDataPath>/config/<db instance>* do not exist on the server or do not have the correct permissions. Verify that the files exist with the correct permissions for the system user that owns the database instance.

6.18.7.1. SAP DB / MaxDB Recovery Kit Error Messages

Error Number	Message*
111000	Usage: %s independent_program_path <validate:value_1:…:value_n:>
111001	Usage: %s %s %s
111002	No value specified to script %s for input argument %s.
111003	User %s with User_Key %s cannot access instance %s. Action: Specify a User_Key for the given user with database access rights.
111004	The user %s does not exist on the server %s.
111005	The SAP DB instance %s is not running on server %s.
111006	Database Manager Utilities were not found in the specified path %s.
111007	An SPS internal error occurred in utility %s. Action: Retry operation.
111008	Unable to obtain %s device space information for SAP DB instance %s for user %s and User_Key %s. Action: Verify that the user and User_Key are valid for the corresponding database instance.
111009	Unable to create raw resource hierarchy for %s. **Action: Verify that the underlying device is a shared device.**
111010	Unable to create filesystem resource hierarchy for %s. Action: Verify that the underlying device is a shared device.
111011	Unable to determine the type of the dev space or install path %s.

	Action: Valid dev space types include file system and/or raw devices.
111012	The path %s is not on a shared filesystem .
111013	The SAP DB instance %s is already under SPS protection on server %s.
111014	The SAP DB instance %s has been successfully started on server %s.
111015	The SAP DB instance %s has been successfully stopped on server %s.
111016	Unable to start SAP DB instance %s on server %s.
111017	Unable to stop SAP DB instance %s on server %s.
111018	Attempting db_warm for database instance %s after db_start failure.
111019	The SAP DB x_server has been successfully started on server %s.
111020	The SAP DB x_server has been successfully stopped on server %s.
111021	The SAP DB x_server is not running on server %s
111022	Unable to start SAP DB x_server on server %s. Action: A problem has occurred using the x_server utility, check the SAP DB logs and correct the problem.
111023	Unable to stop SAP DB x_server on server %s. Action: A problem has occurred using the x_server utility; check the SAP DB logs and correct the problem.
111024	The SAB DB file SAP_DBTech.ini was not found on server %s. Action: Verify that SAP DB is installed correctly on the specified server.
111025	The user id for user %s is not the same on server %s and %s.
111026	The group id for user %s is not the same on server %s and %s.
111027	The service file entries for are not the same on server %s and %s.
111028	One or more of the SAP DB service file entries do not exist on server %s.
111029	No dependents were found for resource %s on server %s.

6.18.8. Appendix – Creating Device Spaces Using Raw I/O with SAP DB

If you plan to use SAP DB / MaxDB with raw devices, you must install the SPS Raw I/O Recovery Kit from the SPS Installation Image file. You must also properly set up the raw I/O devices prior to use.

Requirements

In order to use the SAP DB / Max DB Recovery Kit with raw I/O, the following requirements must be met:

- The Linux OS must support raw I/O devices. For most distributions, this support was included in the 2.4 kernel, but there are some distributions that support raw I/O on a 2.2 kernel.
- All raw I/O devices must be bound to a shared disk partition. The number of device spaces *(devspaces)* that will be located on raw I/O devices determines the exact number of raw devices and shared disk partitions required. Refer to the *SAP DB Manual* for guidelines for creating *devspaces* on raw devices.
- The version of the SAP DB / MaxDB software must support the use of raw I/O devices.

6.18.8.1. Naming Conventions

The naming of raw devices and controller varies by Linux distribution.

- On Red Hat, the device name is */dev/raw/raw<number>* and the controller is */dev/rawctl*
- On SuSE SLES 11 versions, the device name is */dev/raw/raw<number>* and the controller is */dev/raw/rawctl*

Raw I/O Setup Steps

1. Select a shared disk partition of appropriate size for the SAP DB device space.
2. Bind an unused raw device node to this partition. Since this needs to be done every time the machine is rebooted and requires root access, you may want to add the raw bindings to a system initialization file (i.e. *rc.local* or *boot.local*). These bindings must be removed from the file once the hierarchy is under SPS protection. SPS will re-establish the raw bindings for raw I/O devices that are under SPS protection. Use the command *raw –qa* to see which raw device nodes are already in use. For example:

   ```
   # raw -qa

   # raw /dev/raw/raw1 /dev/sda1
   ```

3. Set global read permissions on both the raw device controller (`/dev/rawctl` or `/dev/raw`) and the disk partition on all servers that will protect the database instance.

   ```
   # chmod a+r /dev/rawctl (or chmod a+r /dev/raw )
   ```

4. Set group and user read/write permissions on the raw device on all servers that will protect the database instance.

   ```
   # chmod 664 /dev/raw/raw1
   ```

5. Change the owner of the raw device to the SAP DB OS User for the given database instance on all servers that will protect the database instance.

   ```
   # chown -R sapdb:sapdb /dev/raw/raw1
   ```

6. Add the device space to the database using param_adddevspace or db_adddevspace. Refer to the *SAP DB User Manual and/or the Database Manager CLI Manual.*

6.18.8.2. Adding a Device Space after Creating a Hierarchy

If a tablespace is added on a raw I/O device or shared file system after the SAP DB hierarchy has been created in SPS, you must manually create a resource hierarchy for the raw device or file system via the LifeKeeper GUI. The newly created resource hierarchy must then be made a dependent (child) of the SAP DB resource hierarchy. The updated parameter files must be redistributed if necessary to all servers that protect the database instance (*this is not required if the* `IndepDataPath` *is located on a shared disk*).

6.19. Sybase ASE Recovery Kit Administration Guide

Sybase Adaptive Server Enterprise is a powerful data management platform for high performance business applications. Sybase ASE is a versatile, enterprise-class RDBMS that is especially good at handling OLTP workloads. Sybase ASE is used widely in financial, E-commerce, and other technology arenas. The Sybase ASE platform includes many standard components, such as the Adaptive Server, Monitor Server, and Backup Server, as well as other plug-in components. The Adaptive Server component is the relational database server. The Monitor Server is a separate server from the database server that monitors the Adaptive Server. The Monitor Server can provide real time or historical data to client applications. The Backup Server is an Open Server-based application that manages all database backup (dump) and restore (load) operations for Adaptive Server.

The SIOS Protection Suite for Linux Sybase ASE Recovery Kit will provide SPS resource protection for the Sybase ASE components Adaptive Server, Monitor Server, and Backup Server.

SIOS Protection Suite Documentation

The following is a list of SIOS Protection Suite for Linux related information available from the SIOS Technology Corp. Documentation site:

- SPS for Linux Release Notes
- SPS for Linux Technical Documentation
- Optional Recovery Kit Documentation

Sybase ASE Documentation

You can find Sybase ASE documentation, including the *Installation Guide Adaptive Server for Linux, User Manual, Monitor Server User Manual, Troubleshooting Guide and Reference Manual(s)* at the following location on the web:

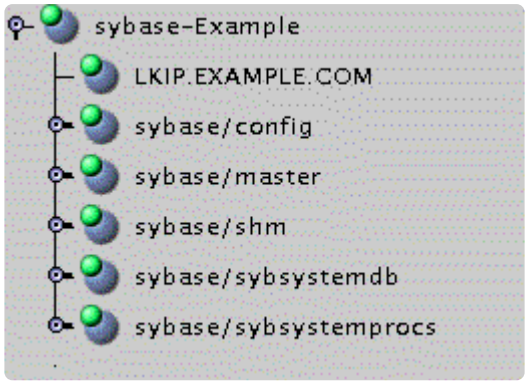
Sybase Product Documentation

6.19.1. Sybase ASE Recovery Kit Overview

The SIOS Protection Suite (SPS) for Linux Sybase ASE Recovery Kit provides a mechanism for protecting Sybase ASE Server instances within SPS. The Sybase ASE software, LifeKeeper Core and Sybase ASE Recovery Kit are installed on two or more servers in a cluster. Once the Sybase ASE Server instance is under SPS protection, clients connect to the database using an SPS protected IP address. The SPS protected IP address must be created separately prior to the creation of the Sybase ASE resource hierarchy. The Sybase ASE resource hierarchy creation will create the dependency between the parent Sybase ASE resource instance, and the child IP address resource. In the event that the Sybase ASE Server instance fails, SPS will first attempt to recover it on the local server. If the local recovery fails, then SPS will fail over to a backup server.

Sybase ASE Resource Hierarchy

The following example shows an example Sybase ASE resource hierarchy:



The dependencies in the above example correspond to the following protected resources:

Resource	Sybase ASE Software Component
LKIP.EXAMPLE.COM	Protects the switchable IP address used for client connections
sybase/config	Protects the file system containing the Sybase Adaptive Server, Monitor Server, and Backup Server configuration files
sybase/master	Protects the Sybase ASE master device
sybase/shm	Protects the Sybase Adaptive Server, and Monitor Server shared memory path
sybase/sybsystemdb	Protects the Sybase ASE sybsystemdb device
sybase/ sybsystemprocs	Protects the Sybase ASE sybsystemprocs device

In the event of failover, SPS will bring the file system, IP address and database resources (including all the resource dependencies) in service on a backup server. Clients will be disconnected, and will need to re-connect to the server. Any SQL statement that has not been committed will need to be re-entered.

6.19.2. Sybase ASE Recovery Kit Hardware and Software Requirements

Your LifeKeeper configuration must meet the following requirements prior to the installation of the LifeKeeper for Linux Sybase Recovery Kit. Please refer to the SPS for Linux Installation Guide for specific instructions regarding the installation and configuration of your LifeKeeper hardware and software.

Hardware Requirements

- **Servers** – Servers should be configured in accordance with the requirements described in the SPS for Linux Technical Documentation and the SPS for Linux Release Notes.

- **IP Network Interface Cards** – Each server requires at least one Ethernet TCP/IP-supported network interface card. Remember, however, that an SPS cluster requires at least two communication paths. Two separate LAN-based communication paths using dual independent sub-nets are recommended for heartbeats, and at least one of these should be configured as a private network. Using a combination of TCP and TTY heartbeats is also supported.

- **Storage** – Servers should be configured to use SPS supported shared storage or the DataKeeper for Linux storage.

Software Requirements

- **TCP/IP Software** – Each server in your SPS configuration requires TCP/IP Software.

- **Sybase ASE Software** – SPS supports version 15.5 and later of the Sybase ASE software. This version can be obtained from Sybase Inc. at http://www.sybase.com/products/databaseservers/ase. **Note**: The same version of the Sybase ASE software must be installed on all servers in the cluster. In addition, only one version of the Sybase ASE software may be installed on the SPS protected servers.

- **SPS Software** – It is imperative that you install the same version of the SPS software and apply the same versions of the SPS software patches to each server in your cluster.

- **SPS for Linux IP Recovery Kit** – The SPS for Linux IP Recovery Kit is required by the SPS for Linux Sybase ASE Recovery Kit. The SPS for Linux IP Recovery Kit is provided on the SPS for Linux image file (*sps.img*) via ftp download.

- **SPS for Linux Sybase ASE Recovery Kit** – The Sybase ASE Recovery Kit (*steeleye-lkSYBASE*) is provided on the SPS for Linux Installation image file (*sps.img*) via ftp download. It is installed and removed via this image file.

6.19.3. Sybase ASE Recovery Kit Configuration Considerations

Configuration Considerations

Contains information to consider before you install and configure the Sybase ASE Recovery Kit.

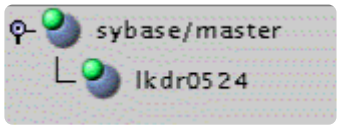
6.19.3.1. Using Raw I/O with Sybase

If you plan to use Sybase ASE with raw devices, you must install the SPS Raw I/O Recovery Kit from the SPS Core image file. You must also properly set up the raw I/O devices prior to use. See the topic Creating Database Devices Using Raw I/O for instructions.

6.19.3.2. Using Sybase ASE Mirrored File Systems with DataKeeper

The Sybase ASE Recovery Kit supports the use of SIOS DataKeeper as a device space. In addition, the Sybase ASE software can be installed on mirrored file systems.

For example, a dependent file system for a Sybase ASE resource would look similar to the following, which shows a file system for the system device space and its dependency, the DataKeeper resource mirror.



6.19.3.3. Sybase Interfaces File Considerations

The Sybase ASE Recovery Kit uses the Sybase ASE interfaces file for the detection of the client IP addresses and ports. This file is located under $SYBASE and is typically called interfaces. This file is updated whenever an Adaptive Server, Monitor Server or Backup Server instance is created using the srvbuild or similar configuration utility. The SPS for Linux Sybase ASE Recovery Kit requires this file to exist with entries for each Sybase ASE component to be protected. Comment lines are not allowed. All server names that appear in the interfaces file must be resolvable to a valid virtual IP address. All servers that will protect the Sybase ASE resource hierarchy must be able to resolve the server names that appear in the interfaces file. In addition, it is recommended that the virtual IP address be used instead of the server name.

Example

- master tcp ether example.com 4100
- query tcp ether example.com 4100

Example_back

- master tcp ether example.com 4200
- query tcp ether example.com 4200

Example_mon

- master tcp ether example.com 4200
- query tcp ether example.com 4200

Sample Interfaces File

6.19.3.4. Sybase ASE Software Asset Manager (SySAM)

The Sybase Software Asset Management (SySAM) is used to manage licensed Sybase products. At Sybase ASE server startup, each ASE server component checks the license file in its environment for permission to run specific features. In order for the ASE server to do this, a license manager and vendor module must be running. The SPS for Linux Sybase ASE Recovery Kit does not provide protection for the SySAM license manager. It is recommended that the license manager be configured in a redundant server system. In the redundant server system, the redundant license allows you to specify local servers as the first license server in the queue, and make remote servers available as backup license servers. The SySAM application attempts to check out a license from a license-file list, starting with the first server. If that server fails for any reason, the second server in the list is contacted, and so on. The `LM_LICENSE_FILE` variable must be set properly in the user profile for the redundant license server environment.

6.19.3.5. Sybase ASE Active/Standby Considerations

In an Active/Standby configuration, the backup server is not actively running the Sybase ASE but stands by in case the primary server experiences a failure. The following scenarios provide specific requirements that must be adhered to when protecting a Sybase ASE resource instance in active/standby configurations.

Scenario 1

The Sybase ASE product is installed **locally on all servers in the cluster**.

- All Sybase Adaptive Server, Monitor Server and Backup Server devices are configured on shared storage.
- The Sybase Adaptive Server, Monitor Server and Backup Server configuration files are stored on a shared file system.
- The Sybase Adaptive Server and Monitor Server shared memory directory is located on a shared file system.
- The interfaces file must be manually updated on all servers to contain common entries for each instance to be protected.
- All interfaces file entries must be resolvable by all servers where the resource will be protected.
- The RUN files for each Adaptive Server, Monitor Server and Backup Server must exist on all servers in the cluster.
- The RUN files for each Adaptive Server, Monitor Server and Backup Server must be executable on all servers in the cluster.
- The RUN files for each Adaptive Server, Monitor Server and Backup Server must contain the same options on all servers in the cluster.

Scenario 2

The Sybase ASE product is installed to **one or more shared file systems on the primary server**.

- All Sybase Adaptive Server, Monitor Server and Backup Server devices are configured on shared storage.
- The Sybase Adaptive Server, Monitor Server and Backup Server configuration files are stored on a shared file system.

- The Sybase Adaptive Server and Monitor Server shared memory directory is located on a shared file system.
- The interfaces file does not have to be updated on the target servers.
- All interfaces file entries must be resolvable by all servers where the resource will be protected.
- On the SPS backup server, */etc/ld.so.conf* must be updated to add entries for the Sybase product libraries.
    - º Add an entry for

`$SYBASE/ASE/lib`

    - º Add an entry for

`$SYBASE/OCS/lib`

    - º Mount the shared file system containing the Sybase

ASE installed products and run `ldconfig`

6.19.3.6. Sybase ASE Active/Active Considerations

In an Active/Active configuration, each server is actively running one or more Sybase ASE Servers, while acting as a backup for the other SPS server in case of failure. The following scenario provides specific requirements that must be adhered to in sequential order when protecting a Sybase ASE resource instance in an active/active configuration.

Scenario 1

The Sybase ASE product is installed locally on all servers in the cluster.

- All Sybase Adaptive Server, Monitor Server, and Backup Server devices are configured on shared storage.
- The Sybase Adaptive Server, Monitor Server, and Backup Server configuration files are stored on a shared file system.
- The Sybase Adaptive Server and Monitor Server shared memory directory is located on a shared file system.
- The interfaces file must be manually updated on all servers to contain common entries for each instance to be protected.
- All interfaces file entries must be resolvable by all servers where the resource will be protected.
- The RUN files for each Adaptive Server, Monitor Server and Backup Server must exist on all servers in the cluster.
- The RUN files for each Adaptive Server, Monitor Server and Backup Server must be executable on all servers in the cluster.
- The RUN files for each Adaptive Server, Monitor Server and Backup Server must contain the same options on all servers in the cluster.

Scenario 2

The Sybase ASE product is installed to one or more shared file systems on the primary server.

- All Sybase Adaptive Server, Monitor Server and Backup Server devices are configured on shared storage.
- The Sybase Adaptive Server, Monitor Server and Backup Server configuration files are stored on a shared file system.

- The Sybase Adaptive Server and Monitor Server shared memory directory is located on a shared file system.
- The interfaces file does not have to be updated on the target servers.
- All interfaces file entries must be resolvable by all servers where the resource will be protected.
- On the SPS backup server, */etc/ld.so.conf* must be updated to add entries for the Sybase product libraries.
    - º Add an entry for

`$SYBASE/ASE/lib`

    - º Add an entry for

`$SYBASE/OCS/lib`

    - º Mount the shared file system containing the Sybase

ASE installed products and run `ldconfig`

6.19.3.7. Sybase ASE Monitor Server and Backup Server

The SPS for Linux Sybase ASE Recovery Kit provides resource protection for the Adaptive Server, Backup Server, and Monitor Server components. However, the Backup Server and Monitor Server components are not required components of a resource hierarchy. The Sybase Backup Server, and the Sybase Monitor Server can be excluded from the resource protection. During the resource hierarchy creation users that do not wish to protect the Sybase Monitor Server, and/or the Sybase Backup Server can choose none for the respective component choices. Selecting none during the GUI resource creation will exclude the selected component from protection in the resource hierarchy. **Note**: ‘none’ is a reserved word in the Sybase ASE Recovery Kit, therefore neither the Sybase Backup Server nor the Sybase Monitor Server can be named ‘none’.

When choosing whether to protect these components it is important to note that the configuration files that share a common file system with the Adaptive Server configuration files, device paths, log paths, or shared memory directories will be protected by SPS. If one or more components will not be protected with SPS, considerations for file placement should be made to prevent sharing between the protected components and the non-protected components.

NOTE: The Sybase Monitor Server is no longer supported with the Sybase ASE ARK v9.0.2 and later.

6.19.3.8. Using Network Attached Storage with Sybase ASE

There are a couple of special considerations to take into account when configuring SPS to use an NFS file server (Network Attached Storage) as cluster storage.

Use the NAS Recovery Kit

The optional Network Attached Storage (NAS) Recovery Kit is required when using an NFS server as a shared storage array with SPS for Linux. Install the NAS Recovery Kit (and a license) on each cluster node. See the NAS Recovery Kit documentation for more details.

Possible Error Message

When using Network Attached Storage (NAS) with Sybase ASE, you may experience Sybase not restarting following a failover due to a system crash. The Sybase error log should indicate the cause of the error.

Sybase ASE 15.x

```
00:00:00000:00000:2011/05/09 16:08:51.66 kernel Adaptive Server
Enterprise(Developer Edition)
00:00:00000:00000:2011/05/09 16:08:51.66 kernel basis_dlock: file
'/s10/sybase-data155/data/master.dat' already in use by an ASE
00:00:00000:00000:2011/05/09 16:08:51.66 kernel kdconfig: unable to
read primary master device
00:00:00000:00000:2011/05/09 16:08:51.66 server kiconfig: read of
config block failed
```

This indicates that the Sybase dataserver has set an NFS lock on the file "*master.dat*" on the NFS file system that is being controlled by SPS. The lock was not cleared by the system crash, so SPS is unable to bring the dataserver back into service. Sybase thinks that some other process is using the *master.dat* file.

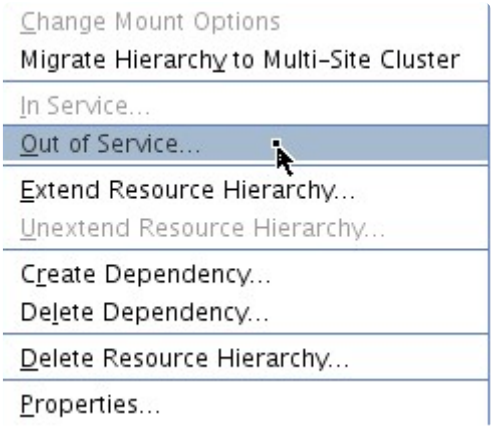
Solution

To fix this, mount the NFS file system that will hold *master.dat* with the "`nolock`" NFS option before the File System resource is created. By default, NFS allows file locks to be set. If the "`nolock`" option is used before resource creation, SPS will pick up this option and use it each time it brings the file system resource in service. Since SPS will be controlling access (from the cluster nodes) to the file system containing *master.dat*, the lock is not typically critical. The NFS mount options used during testing were "`rw,sync,tcp,nfsvers=3,noac,nolock`".

It is not necessary to use the "nolock" on other file systems used by the Sybase resource hierarchy such as the file system where the Sybase ASE binaries are located.

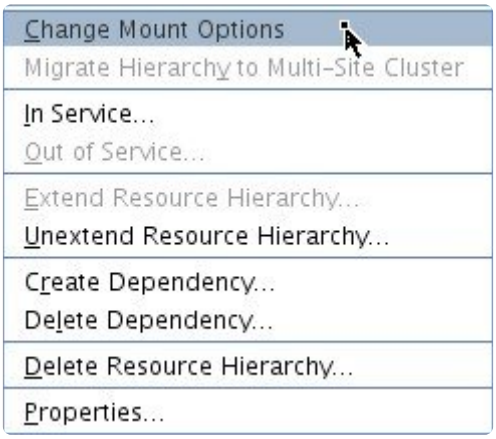
If the NAS File System resource has already been created without the "nolock" option set, use the following procedure to change the mount option:

1. Using the LifeKeeper GUI, take the file system resource that needs to be changed out of service. This can be done from the LifeKeeper GUI putting the pointer on the file system resource and doing a right mouse click, and select **"Out of Service"** from the drop-down menu. This action may take parent resources out of service as well.

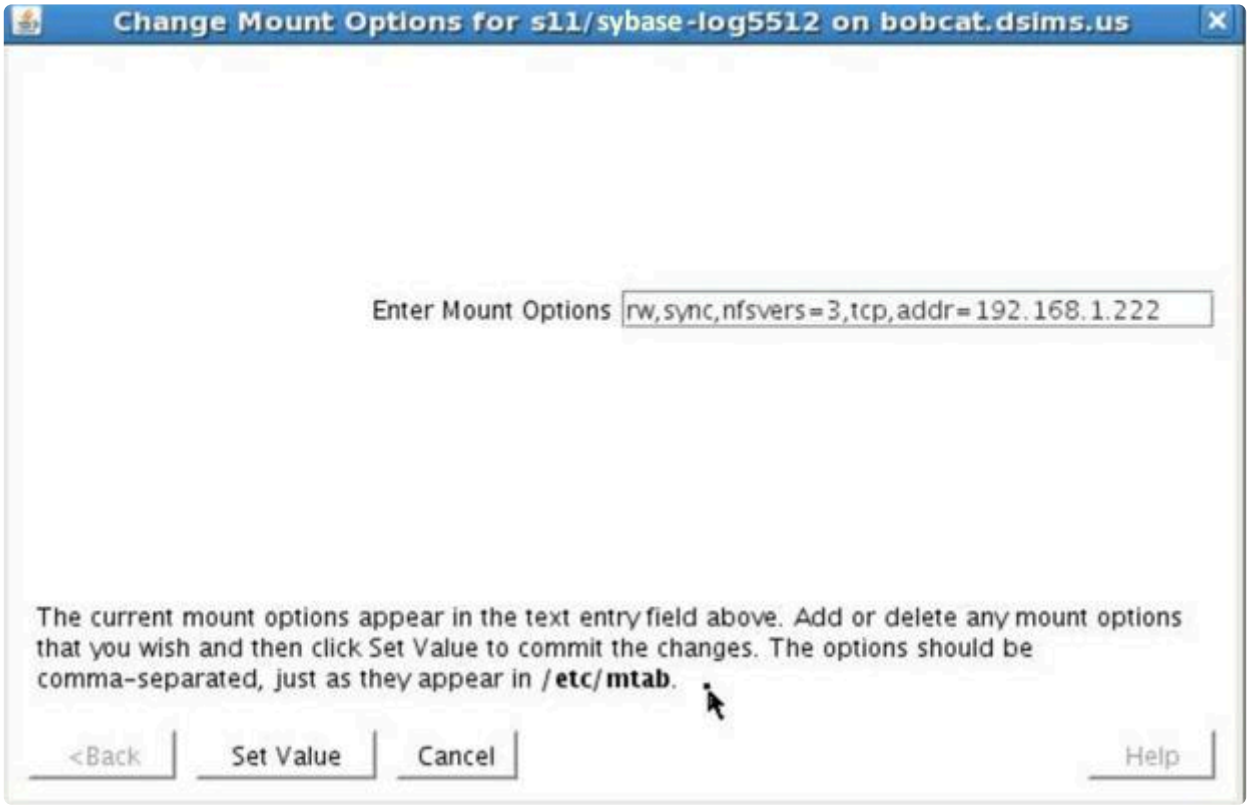


2. Confirm the **"Out of Service"** action and allow the process to complete.

3. Once the file system resource is out of service, you can put the pointer on the resource and do another right mouse click, and from the drop-down menu, select **"Change Mount Options"**.

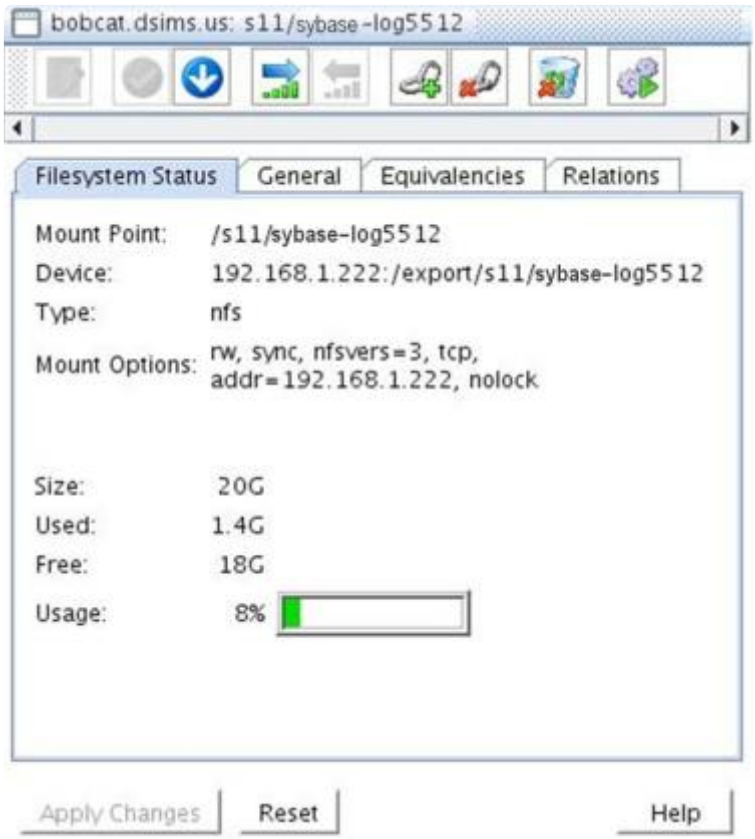


4. In the popup window, add "nolock" to the line of options, and click **"Set Value."** You will need to repeat steps 3 and 4 for each node in the cluster.



5. Bring the NAS File System resource back in service by doing a right mouse click, and selecting **"In Service"**.

6. The File System resource's property panel should now reflect that "nolock" is one of the current mount options.



6.19.4. Installing and Configuring Sybase ASE with SPS

The following sequence is recommended for installing and configuring the Sybase ASE product and SPS software. Each of these steps links to detailed tasks.

Install the Sybase ASE Software

Create the Sybase ASE Servers

Install the SPS Software

After you have performed these tasks, you will be ready to create the SPS resource hierarchy to protect your Sybase ASE Server(s).

Resource Configuration Tasks

Describes the various functions you may perform on your hierarchies using the LifeKeeper GUI: **create**, **extend**, **delete** and **unextend**.

Once you have completed the setup tasks described in the previous section, you are ready to create and extend your Sybase ASE resource hierarchies.

The following tasks are available for configuring the SPS for Linux Sybase ASE Recovery Kit:

- Create Resource Hierarchy – Creates a Sybase ASE resource hierarchy
- Delete Resource Hierarchy – Deletes a Sybase ASE resource hierarchy
- Extend Resource Hierarchy – Extends a Sybase ASE resource hierarchy from the primary server to the backup server
- Unextend Resource Hierarchy – Unextends (removes) a Sybase ASE resource hierarchy from a single server in the SPS cluster
- Testing Your Resource Hierarchy – Tests your Sybase ASE resource hierarchy

Refer to the GUI Administrative Tasks section of the SPS for Linux Technical Documentation for instructions on configuring SPS Core resource hierarchies, for instance, file system and IP resources.

The following tasks are described in the Administration section within the SPS for Linux Technical Documentation because they are common tasks with steps that are identical across all Recovery Kits.

- Create a Resource Dependency. Creates a parent/child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all

applicable servers in the cluster.

- Delete a Resource Dependency. Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service. Brings a resource hierarchy into service on a specific server.
- Out of Service. Takes a resource hierarchy out of service on a specific server.
- View /Edit Properties. View or edit the properties of a resource hierarchy on a specific server.

> **Note**: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You may also perform most of the tasks:

- from the toolbar
- by right-clicking on a global resource in the left pane of the status display
- by right-clicking on a resource in the right pane of the status display

Using the right-click method allows you to avoid entering information that is required using the **Edit** menu.

6.19.4.1. Install the Sybase ASE Software

Install the Sybase ASE software on all servers in the cluster using identical parameters/settings. Refer to the *Installation Guide Adaptive Server for Linux* for details. The following are additional recommendations to ensure that SPS will work with Sybase ASE:

- A non-root system user (Sybase OS User) must exist on all servers. The user must have the same user id, group id, and home directory on all servers where the resource(s) will be protected.
- The Sybase ASE common software packages must be installed. This package provides both the Sybase *srvbuild* and Sybase *isql* utilities.
- Each SPS server containing a Sybase ASE resource hierarchy must have identical service entries in the *$SYBASE/interfaces* file for the Sybase ASE Server(s).
- Verify that a link exists between *$SYBASE/ASE-<version>* and *$SYBASE/ASE*. If the link does not exist, it must be manually created. See the topic Creating Links for ASE and OCS for additional information.
- Verify that a link exists between *$SYBASE/OCS-<version>* and *$SYBASE/OCS*. If the link does not exist, it must be manually created. See the topic Creating Links for ASE and OCS for additional information.
- Refer to the *Installation Guide Adaptive Server for Linux* for details on configuring shared memory parameters for the Adaptive Server, Monitor Server and Backup Server.
- The database device must be protected by LifeKeeper as shared storage. In addition, configuration files and other files must be on the shared storage protected by LifeKeeper. See Creating the Sybase ASE Servers for details.
- The Sybase ASE common software package should be installed on the shared storage protected by LifeKeeper or on the same path in the local area on all servers in the cluster.

6.19.4.2. Create the Sybase ASE Servers

NOTE: The Sybase Monitor Server is no longer supported with the Sybase ASE ARK v9.0.2 and later.

Follow the instructions in your *Installation Guide Adaptive Server for Linux* for configuring the Sybase Adaptive Server, Monitor Server and Backup Server. The following considerations should be followed:

- Use the `srvbuild` utility or other Sybase ASE utility to create the Sybase Adaptive Server instance
  - Configure all system devices on shared storage
  - Configure the Adaptive Server configuration files on shared storage
  - Configure the Adaptive Server shared memory directory on shared storage
  - Configure the interface to use a SPS switchable IP address
  - Optionally configure the logs on shared storage

- If required, create the Sybase Monitor Server instance
  - Configure all system devices on shared storage
  - Configure the Monitor Server configuration files on shared storage
  - Configure the Monitor Server shared memory directory on shared storage
  - Configure the interface to use a SPS switchable IP address
  - Optionally configure the logs on shared storage

- If required, create the Sybase Backup Server instance
  - Configure all system devices on shared storage
  - Configure the Monitor Server configuration files on shared storage
  - Configure the Monitor Server shared memory directory on shared storage
  - Configure the interface to use a SPS switchable IP address
  - Optionally configure the logs on shared storage

6.19.4.3. Install the SPS Software with Sybase

Once you have installed the Sybase ASE software and created your database servers, you are ready to install the SPS Core software, SPS for Linux IP Recovery Kit and any required patches followed by the Sybase ASE Recovery Kit. Also, if you plan to use Sybase ASE with raw devices, you must install the SPS Raw I/O Recovery Kit from the SPS Core image file. See Creating Device Spaces Using Raw I/O for requirements and instructions on setting up *raw* devices.

Refer to the SPS for Linux Installation Guide for details on installing the SPS packages.

6.19.4.4. Creating a Sybase ASE Resource Hierarchy

Note: Make sure that the Sybase ASE is running on the primary server.

Perform the following steps on the primary server:

1. On the **Edit** menu, select **Server**, then **Create Resource Hierarchy**.

   The ***Create Resource Wizard*** dialog will appear.

2. Select **Sybase ASE Database** from the drop-down list and click **Next**.

3. You will be prompted for the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter any error requiring you to correct the previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

4. Click **Next**. The ***Create Resource Wizard*** will then create your Sybase ASE resource hierarchy. SPS will validate the data entered. If SPS detects a problem, an error message will appear in the information box.

Field	Tips
Server	Select the SPS server where the Sybase ASE resource is to be created.
Switchback Type	Choose either **intelligent** or **automatic**. This determines how the Sybase ASE resource will be switched back to the primary server after it comes in-service (active) on the backup server following a failover. Intelligent switchback requires administrative intervention to switch the resource back to the primary server, while automatic switchback occurs as soon as the primary server is back on line and reestablishes SPS communication paths. **Note**: The switchback strategy must match that of the dependent resources to be used by the Sybase ASE resource.

Sybase Install Directory	This field is used to specify the installation location of the Sybase ASE product. You may type in another directory path. The valid characters allowed for the pathname are letters, digits and the following special characters: – _ . /
Sybase Instance Directory	This field is used to specify the directory path that contains the Sybase data directory. The data directory will typically contain the ASE-<version>/RUN_* files for the instance.
Sybase Instance	This field contains by default the name of the first Sybase instance found on the system, for which no SPS hierarchy exists. The drop down list shows other Sybase instances that may be available on your SPS server. This field is used to specify the Sybase ASE Database instance that will be placed under LifeKeeper protection. The specified instance must exist and must be running.
Sybase Username	This field is used to enter the user name for the Sybase System Administrator. By default the user name is sa. This System Administrator must have login and full privileges on any database on the Sybase Adaptive Server being protected.
Sybase Login Password	This field is used to specify the password for the Sybase System Administrator.
Sybase Backup Server	This field is used to specify the Sybase Backup server for the specified Adaptive Server instance. This Sybase Backup will be placed under SPS protection. The user may select 'none' if the Sybase Backup Server does not need to be included under SPS protection.
Sybase ASE Database Tag	This is a unique tag name for the new Sybase ASE database resource on the primary server. The default tag name consists of the word sybase followed by the name of the Adaptive Server instance. You may type in another unique tag name. The valid characters allowed for the tag are letters, digits, and the following special characters: – _ . /

5. You should see a message indicating that you have successfully created a Sybase ASE resource hierarchy, and you must extend that hierarchy to another server in your cluster to achieve failover protection. Click **Next**.

6. Click **Continue**. SPS will then launch the ***Pre-extend Wizard***. Refer to **Step 2** under Extending a Sybase ASE Resource Hierarchy for details on how to extend your resource hierarchy to another server.

6.19.4.5. Extending a Sybase ASE Resource Hierarchy

This operation can be started from the Edit menu or initiated automatically upon completing the **Create Resource Hierarchy** option, in which case you should refer to Step 2 below.

1. On the Edit menu, select **Resource**, then **Extend Resource Hierarchy**. The ***Pre-Extend Wizard*** appears. If you are unfamiliar with the **Extend** operation, click **Next**. If you are familiar with the SPS **Extend Resource Hierarchy** defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults**.

2. The ***Pre-Extend Wizard*** will prompt you to enter the following information.

> **Note**: The first two fields appear only if you initiated the **Extend** from the **Edit** menu.

3. After receiving the message that the pre-extend checks were successful, click **Next**.

Field	Tips
Template Server	Select the server where your Sybase ASE resource is currently in service.
Tag to Extend	Select the Sybase ASE resource you wish to extend.
Target Server	Enter or select the server you are extending to.
Switchback Type	This determines how the Sybase ASE resource will be switched back tothe primary server after it comes in-service (active) on the backup server following a failover. You can choose either intelligent or automatic. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. **Note**: Remember that the switchback strategy must match that of the dependent resources to be used by the Sybase ASE resource.
Template Priority	Select or enter a Template Priority. This is the priority for the Sybase ASE hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note**: This selection will appear only for the initial extend of the hierarchy.

Target Priority	This is the priority for the new extended Sybase ASE hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from 1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. Note that SPS assigns the number "1" to the server on which the hierarchy is created by default. The priorities need not be consecutive, but no two servers can have the same priority for a given resource.

4. Depending upon the hierarchy being extended, SPS will display a series of information boxes showing the Resource Tags to be extended, some of which cannot be edited.

5. The ***Extend Wizard*** will prompt you to enter the following information.

6. After receiving the message "`Hierarchy extend operations completed`". click **Next Server** to extend the hierarchy to another server, or click **Finish** if there is no other extend operations to perform.

Sybase ASE Install Directory	This field contains by default the Sybase ASE install path of the Template Resource. The valid Sybase ASE installation path should be specified. The valid characters allowed for the pathname are letters, digits, and the following special characters: – _ . /
Sybase ASE Database Tag	This is a unique tag name for the new Sybase ASE database resource on the primary server. The default tag name consists of the word sybase followed by the name of the Adaptive Server instance. You may type in another unique tag name. The valid characters allowed for the tag are letters, digits, and the following special characters: – _ . /

7. After receiving the message "`Hierarchy Verification Finished`", click **Done**.

6.19.4.6. Unextending a Sybase ASE Resource Hierarchy

To remove a resource hierarchy from a single server in the SPS cluster, do the following:

1. On the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.

2. Select the Target Server where you want to unextend the Sybase ASE resource. It cannot be the server where the resource is currently in service. (This dialog box will not appear if you selected the **Unextend** task by right-clicking on a resource instance in the right pane.) Click **Next**.

3. Select the Sybase ASE hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the Unextend task by right-clicking on a resource instance in either pane.)

4. An information box appears confirming the target server and the Sybase ASE resource hierarchy you have chosen to unextend. Click **Unextend**.

5. Another information box appears confirming that the Sybase ASE resource was unextended successfully. Click **Done** to exit the **Unextend Resource Hierarchy** menu selection.

6.19.4.7. Deleting a Sybase ASE Resource Hierarchy

To delete a Sybase ASE resource from all servers in your SPS configuration, complete the following steps:

1. On the **Edit** menu, select **Resource**, then **Delete Resource Hierarchy**.
2. Select the name of the Target Server where you will be deleting your Sybase ASE resource hierarchy.

> **Note**: If you selected the **Delete Resource** task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

3. Select the Hierarchy to Delete. (This dialog will not appear if you selected the **Delete Resource** task by right-clicking on a resource instance in the left or right pane.) Click **Next**.
4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Next**.
5. Another information box appears confirming that the Sybase ASE resource was deleted successfully.
6. Click **Done** to exit.

6.19.4.8. Testing Your Sybase ASE Resource Hierarchy

You can test your Sybase ASE resource hierarchy by initiating a manual switchover that will simulate a failover of the resource instance from the primary server to a backup server.

Performing a Manual Switchover from the LifeKeeper GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit**, **Resource** and **In Service**. For example, an in-service request executed on a backup server causes the Sybase ASE resource hierarchy to be placed in service on the backup server and taken out-of-service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the **Out of Service** request, the resource hierarchy is taken out-of-service without bringing it in service on the other server.

IMPORTANT: After bringing your resource hierarchy in service on the backup server, you should attempt to connect to the databases, especially when using raw devices as device spaces. This is necessary to ensure that all disk partitions are visible on the backup servers and the raw bindings are being established correctly.
• If the raw bindings have not been established on the backup servers, it is most likely caused by the fact that new partitions were created on the primary server and added to the configuration, but the partition tables have not yet been updated on the backup servers.
• The solution is to reboot the backup servers so that the partition tables are updated correctly.

6.19.5. Sybase ASE Recovery Kit Administration

Resource Hierarchy Administration

Provides important recommendations for ongoing administration of the Sybase ASE hierarchy.

The following tasks may be required after your resource hierarchies have been created.

- Modifying Protection for the Sybase Backup Server
- Modifying Protection for the Sybase Monitor Server
- Updating Parameters

6.19.5.1. Modifying Protection for the Sybase Backup Server

The Sybase Backup Server is an Open Server-based application that manages all database backups (dump) and restores (load) operations for Adaptive Server. The Sybase Backup Server can be protected by the SPS Sybase ASE resource hierarchy during the resource creation, or added to the SPS protection after the resource hierarchy creation. In addition, the Sybase Backup Server can be removed from SPS protection after the hierarchy has been created.

Adding a Sybase Backup Server

To add a Sybase Backup Server to an existing Sybase ASE resource hierarchy, the Sybase `srvbuild` or other configuration utility must have created one.

1. On the **Edit** menu, select **Resource**, then select **Properties**. A ***Resource Properties Wizard*** will appear.

2. Select the resource tag from the **Select Resource** drop-down. This is the resource tag for the SPS protected Sybase ASE resource to modify.

3. Select the SPS Server from the **Select Server for Resource** drop-down. This will be the server to update the Sybase ASE resource instance on. If changes are required on more than one SPS server, then this process should be repeated for each server in the cluster.

4. Select the **Resource Configuration** button on the **Resource Properties** page. This will launch a ***Reconfiguration Wizard*** for the protected resource selected in Step 3. The first screen of the wizard will display the current configuration settings for the resource under SPS protection. Select Next.

5. If a valid Sybase Backup Server exists on the specified server, the next screen will display a drop-down for the Sybase Backup Server to add or remove. Select the Sybase Backup Server to add from the list. Select **Next**. **Note**: For Sybase ASE installations where the Sybase software is installed on shared storage, the file system containing the installation must be in service on the server where the reconfiguration will take place.

6. If a valid Sybase Monitor Server exists, the next screen will allow you to configure it now. Refer to Modifying Protection for the Sybase Monitor Server for considerations regarding modifying the Monitor Server protection.

7. Select **Reconfigure**. If any errors are displayed they must be corrected before proceeding. Otherwise, select **Done**.

8. Any Sybase Backup Server configuration file paths or associated database devices should be manually protected with an SPS file system resource and made a dependent of the parent

resource hierarchy.

9. The virtual IP address associated with the Sybase Backup Server must be made a dependent of the parent resource hierarchy. To find the associated IP address, look for the master and query lines following the Sybase Backup Server name in the interfaces file.

Removing a Sybase Backup Server

The following steps outline the process for removing a Sybase Backup Server from an existing Sybase ASE resource hierarchy.

1. On the **Edit** menu, select **Resource**, select **Properties**. A ***Resource Properties Wizard*** will appear.

2. Select the resource tag from the **Select Resource** drop-down. This is the resource tag for the SPS protected Sybase ASE resource to modify.

3. Select the SPS Server from the **Select Server for Resource** drop-down. This will be the server to update the Sybase ASE resource instance on. If changes are required on more than one SPS server, then this process should be repeated for each server in the cluster.

4. Select the **Resource Configuration** button on the **Resource Properties** page. This will launch a ***Reconfiguration Wizard*** for the protected resource selected in Step 3. The first screen of the wizard will display the current configuration settings for the resource under SPS protection. Select **Next**.

5. If a valid Sybase Backup Server exists on the specified server, the next screen will display a drop-down for the Sybase Backup Server to add or remove. Select ‘none’ from the list to remove protection for the Sybase Backup Server. Select **Next**.

6. If a valid Sybase Monitor Server exists, the next screen will allow you to configure it now. Refer to Modifying Protection for the Sybase Monitor Server for considerations regarding modifying the Monitor Server protection.

7. Select **Reconfigure**. If any errors are displayed, they must be corrected before proceeding. Otherwise, select **Done**.

8. Any Sybase Backup Server configuration file paths or associated database devices that are no longer in use should be removed from the Sybase ASE resource dependency and deleted from SPS.

9. Any Sybase Backup Server virtual IP resources that are no longer in use should be removed from the Sybase ASE resource dependency and deleted from SPS.

6.19.5.2. Modifying Protection for the Sybase Monitor Server

> **NOTE**: The Sybase Monitor Server is no longer supported with the Sybase ASE ARK v9.0.2 and later.

The Monitor Server is a separate server from the database server that monitors the Adaptive Server. The Monitor Server can provide real time or historical data to client applications. The Sybase Monitor Server can be protected by the SPS Sybase ASE resource hierarchy during the resource creation, or added to the SPS protection after the resource hierarchy creation. In addition, the Sybase Monitor Server can be removed from SPS protection after the hierarchy has been created.

Adding a Sybase Monitor Server

To add a Sybase Monitor Server to an existing Sybase ASE resource hierarchy, the Sybase srvbuild or other configuration utility must have created one.

1. On the **Edit** menu, select **Resource**, select **Properties**. A ***Resource Properties Wizard*** will appear.

2. Select the resource tag from the **Select Resource** drop-down. This is the resource tag for the SPS protected Sybase ASE resource to modify.

3. Select the SPS Server from the **Select Server for Resource** drop-down. This will be the server to update the Sybase ASE resource instance on. If changes are required on more than one SPS server, then this process should be repeated for each server in the cluster.

4. Select the **Resource Configuration** button on the **Resource Properties** page. This will launch a ***Reconfiguration Wizard*** for the protected resource selected in Step 3. The first screen of the wizard will display the current configuration settings for the resource under SPS protection. Select **Next**.

5. If a valid Sybase Backup Server exists, the next screen will allow you to configure it now. Refer to Modifying Protection for the Sybase Backup Server for considerations regarding modifying the Backup Server protection.

6. If a valid Sybase Monitor Server exists on the specified server, the next screen will display a drop-down for the Sybase Monitor Server to add or remove. Select the Sybase Monitor Server to add from the list. Select **Next**. **Note**: For Sybase ASE installations where the Sybase software is installed on shared storage, the file system containing the installation must be in-service on the server where the reconfiguration will take place.

7. Select **Reconfigure**. If any errors are displayed they must be corrected before proceeding.

Otherwise, select **Done**.

8. Any Sybase Monitor Server configuration file paths or associated database devices should be manually protected with an SPS file system resource and made a dependent of the parent Sybase ASE resource hierarchy.

9. The virtual IP address associated with the Sybase Monitor Server must be made a dependent of the parent Sybase ASE resource hierarchy. To find the associated IP address, look for the master and query lines following the Sybase Monitor Server name in the interfaces file.

Removing a Sybase Monitor Server

The following steps outline the process for removing a Sybase Monitor Server from an existing Sybase ASE resource hierarchy.

1. On the **Edit** menu, select **Resource**, select **Properties**. A ***Resource Properties Wizard*** will appear.

2. Select the resource tag from the **Select Resource** drop-down. This is the resource tag for the SPS protected Sybase ASE resource to modify.

3. Select the SPS Server from the Select Server for Resource pull down. This will be the server to update the Sybase ASE resource instance on. If changes are required on more than one SPS server, then this process should be repeated for each server in the cluster.

4. Select the **Resource Configuration** button on the **Resource Properties** page. This will launch a ***Reconfiguration Wizard*** for the protected resource selected in Step 3. The first screen of the wizard will display the current configuration settings for the resource under SPS protection. Select **Next**.

5. If a valid Sybase Backup Server exists, the next screen will allow you to configure it now. Refer to Modifying Protection for the Sybase Backup Server for considerations regarding modifying the Backup Server protection

6. If a valid Sybase Monitor Server exists on the specified server, the next screen will display a pull down for the Sybase Monitor Server to add or remove. Select ‘none’ from the list to remove protection for the Sybase Monitor Server. Select **Next**.

7. Select **Reconfigure**. If any errors are displayed they must be corrected before proceeding. Otherwise, select **Done**.

8. Any Sybase Monitor Server configuration file paths or associated database devices that are no longer in use should be removed from the Sybase ASE resource dependency and deleted from SPS.

9. Any Sybase Monitor Server virtual IP resources that are no longer in use should be removed from

the Sybase ASE resource dependency and deleted from SPS.

6.19.5.3. Updating Sybase ASE Parameters

When database parameters are updated for a Sybase ASE instance, it is necessary to check that all changes will allow the instance to function on all LifeKeeper servers in the cluster. If changes require the addition or deletion of LifeKeeper resources, such as file systems, raw devices or virtual IP addresses, these must be added manually and made a dependency of the parent Sybase ASE resource hierarchy.

6.19.6. Troubleshooting Sybase ASE Error During Resource Creation

Sybase ASE Error During Resource Creation

Symptom: Unable to create the resource instance during the resource creation
Cause: If the instance is running, it could be because the profile is not located in the default $Sybase directory
Solution: In /etc/default/LifeKeeper set the tunable SYBASE_PROFILE to the location of the correct SYBASE.sh profile
For example: Add to /etc/default/LifeKeeper

```
SYBASE_PROFILE=/opt/my-non-standard-path/SYBASE.sh
```

Symptom: SPS-L Sybase Resource fails to come in-service but the database instance is started.
Cause: The instance took longer than the default start up time to complete its startup and recovery process.
Solution: Increase the start wait tunable via the /etc/default/LifeKeeper file
For example: Add to /etc/default/LifeKeeper

```
SYBASE_STARTWAIT=120
```

Sybase ASE Recovery Kit Error Messages

Lists and describes the error messages associated with the Sybase ASE Recovery Kit.

114000	Usage: %s
114001	The Sybase Install Directory cannot be empty. ACTION: Please specify a value for this field.
114002	The path %s is not a valid directory
114003	The Sybase Product was not found in the directory %s on server %s. ACTION: Verify that a supported version of Sybase is installed in the specified location.
114004	The specified instance %s is not a valid Sybase ASE Server on %s.
114005	Unable to verify that the Sybase ASE Server %s is running.
114006	The Sybase Monitor Server %s will be protected.
114007	The Sybase Backup Server %s will be protected.
114008	The Sybase ASE Server %s is already under SPS protection on %s.

114009	An unknown error has occurred in utility %s on server %s. ACTION: View the SPS logs for details and retry the operation.
114010	Unable to get the version for the Sybase Server %s installed under %s on %s.
114011	The device %s for Sybase ASE Server %s is not a valid device.
114012	An error has occurred while trying to obtain the devices for Sybase ASE Server %s.
114013	Unable to create raw resource hierarchy for %s.
114014	Unable to create file system resource hierarchy for %s.
114015	The path %s is not on a shared file system.
114016	Unable to create resource dependency for parent %s and child %s.
114017	Information: SPS will not protect the path %s because it is not located on a shared file system.
114018	Unable to get the owner for the Sybase ASE Server%s installed under%s on %s.
114019	Unable to open file%s on server%s due to error %s.
114020	There are no hosts defined for the Sybase ASE Server %s in the file%s.
114021	There are no ports defined for the Sybase ASE Server %s in the file %s.
114022	The specified host name %s defined for the Sybase ASE Server %s in the file %s cannot be resolved.
114023	Unable to detect the host and ports for the Sybase ASE Server %s.
114024	A LifeKeeper resource hierarchy does not exist for the IP address %s on server %s. ACTION: Create a LifeKeeper resource hierarchy for the specified IP address
114025	The values specified for the target and the template servers are the same. ACTION: Please specify the correct values for the target and template servers.
114026	The system user %s does not exist on the server %s.
114027	The group id for user %s is not the same on template server %s and target server %s.
114028	The user id for user %s is not the same on template server %s and target server %s.
114029	There are no IP dependent resources defined for the Sybase resource %s on %s. ACTION: Create the required dependent IP resource hierarchy.
114030	The interfaces defined for Sybase ASE Server %s differ on template server %s and target server %s
114031	The ports defined for Sybase ASE Server %s differ on template server %s and target server %s

114032	The port %s used by the Sybase resource hierarchy %s on the server %s is in use by another application on server%s.
114033	The startup of the Sybase ASE Server(s) on %s failed for the following Sybase ASE Server(s): %s.
114034	Unable to stop the Sybase ASE Server(s) %s on %s.
114035	The Sybase ASE resource hierarchy %s does not contain any valid gen/filesys or scsi/raw resource dependents on server %s. ACTION: The hierarchy does not contain any valid dependents, you must delete and recreate the hierarchy.
114036	There are no Sybase ASE Servers available for protection with LifeKeeper.
114037	Unable to obtain the pid of the backupserver process corresponding to instance %s.
114038	The pid detected for Sybase Backup Server %s in the LifeKeeper pidfile %s.LK on server %s exists in another LifeKeeper pidfile on this server. ACTION: The duplicate pid entry in the pid files should be resolved. The pid file for the instance that is not running should be removed.
114039	Unable to update the resource instance %s on server %s.
114040	The update of the resource instance %s failed on server %s. All attempts to rollback the instance information field have failed. ACTION: Manual intervention is required.
114041	The interfaces file %s on %s contains an invalid comment line. ACTION: Please correct the interfaces file to remove any comment lines.
114042	One or more of the Sybase ASE Servers is missing from the file %s.
114043	The file %s does not exist on server %s.
114044	The reconfiguration of the Sybase ASE resource hierarchy %s on server %s was successful
114045	The update of the resource instance %s failed on server %s. The instance information field has not been modified. ACTION: Retry the reconfiguration operation
114046	The home directory for user %s is not the same on template server %s and target server %s
114047	The file %s on server %s is a link that does not resolve to a dependent shared resource on the template server %s.
114048	The link %s and its resolved path %s are not on a protected shared filesystem.

6.19.7. Appendix – Creating Device Spaces Using Raw I/O with Sybase ASE

Creating Device Spaces Using Raw I/O

6.19.7.1. Requirements for Using Sybase ASE with Raw I/O

In order to use the Sybase ASE Recovery Kit with raw I/O, the following requirements must be met:

- The Linux OS must support raw I/O devices. For most distributions this support was included in the 2.4 kernel, but there are some distributions that support raw I/O on a 2.2 kernel.
- All raw I/O devices must be bound to a shared disk partition. The number of database devices (devspaces) that will be located on raw I/O devices determines the exact number of raw devices and shared disk partitions required. Refer to the *Installation Guide Adaptive Server for Linux* for guidelines for creating database devices on raw devices.
- The version of the Sybase ASE software must support the use of raw I/O devices.

6.19.7.2. Naming Conventions

The naming of raw devices and controller varies by Linux distribution.

- On Red Hat, the device name is `/dev/raw/raw<number>` and the controller is `/dev/rawctl`
- On SuSE SLES 11 versions, the device name is `/dev/raw/raw<number>` and the controller is `/dev/raw/rawctl`

6.19.7.3. Using Raw I/O with Sybase Setup Steps

1. Select a shared disk partition of appropriate size for the Sybase ASE database device.

2. Bind an unused raw device node to this partition. Since this needs to be done every time the machine is rebooted, and requires root access, you may want to add the raw bindings to a system initialization file (i.e. rc.local or boot.local). These bindings must be removed from the file once the hierarchy is under SPS protection. SPS will re-establish the raw bindings for raw I/O devices that are under SPS protection. Use the command *raw –qa* to see which raw device nodes are already in use. For example:

   ```
   # raw -qa

   # raw /dev/raw/raw1 /dev/sda1
   ```

3. Set global read permissions on both the raw device controller (*/dev/rawctl* or */dev/raw/rawctl*), and the disk partition on all servers that will protect the database instance.

   ```
   # chmod a+r /dev/rawctl (or chmod a+r /dev/raw/rawctl)
   ```

4. Set group and user read/write permissions on the raw device on all servers that will protect the database instance.

   ```
   # chmod 664 /dev/raw/raw1
   ```

5. Change the owner of the raw device to the Sybase ASE owner for the given database instance on all servers that will protect the database instance.

   ```
   # chown -R sybase:sybase /dev/raw/raw1
   ```

6. Refer to the *Installation Guide Adaptive Server for Linux* for information on adding the raw device to the database server(s).

6.19.7.4. Adding a Device Space after Creating a Sybase Hierarchy

If a database device is added on a raw I/O device or shared file system after the Sybase ASE hierarchy has been created in SPS, you must manually create a resource hierarchy for the raw device or file system via the LifeKeeper GUI. The newly created resource hierarchy must then be made a dependent (child) of the Sybase ASE resource hierarchy.

6.19.7.5. Creating Links for ASE and OCS

The SPS for Linux Sybase ASE Recovery Kit requires that the path *$SYBASE/ASE-<version>* be symbolically linked to *$SYBASE/ASE*. In addition, the path *$SYBASE/OCS-<version>* must be symbolically linked to *$SYBASE/OCS*. The SPS for Linux Sybase ASE Recovery Kit uses these links to access various Sybase utilities and files. To create the links follow the steps below.

1. From the command line, change directories into the *$SYBASE* directory.

   Example:

   ```
   server1 # cd $SYBASE

   server1 # pwd

        /opt/sybase-15.5
   ```

2. Locate the *ASE-<version>* directory

Example:

```
server1 # ls -ld ASE*

drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 ASE-15_5

lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:35 ASEP -> ASEP-1_0

drwxrwxr-x 4 sybase sybase 4096 Nov 17 11:35 ASEP-1_0p
```

Note: If a link already exists between *ASE-15_5* and *ASE*, proceed to Step 5.

3. Verify that the *ASE-<version>* directory contains the bin/srvbuild utility.

Example:

```
server1 # ls ASE-15_5/bin/srvbuild

     srvbuild
```

Note: If a "no such file or directory" error occurs, then you have chosen the wrong path.

4. From the command line, create a link between the identified *ASE-<version>* directory and *ASE*.

Example:

```
server1 # pwd
     /opt/sybase-15.5
server1 # ln -s ASE-15_5 ASE
```

5. Verify the link was properly created.

Example:

```
server1 # ls -ld ASE*
     lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:20 ASE -> ASE-15_5
     drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 ASE-15_5
     lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:35 ASEP -> ASEP-1_0
     drwxrwxr-x 4 sybase sybase 4096 Nov 17 11:35 ASEP-1_0
server1 # ls ASE/bin/srvbuild
     srvbuild
```

6. From the command line, change directories into the *$SYBASE* directory.

Example:

```
server1 # cd $SYBASE
server1 # pwd
     /opt/sybase-15.5
```

7. Locate the *OCS-<version>* directory

Example:

```
server1 # ls -ld OCS*
     drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 OCS-15_5
```

Note: If a link already exists between *OCS-15_5* and *OCS*, proceed to Step 5.

8. Verify that the *OCS-<version>* directory contains the *bin/isql* utility.

Example:

```
server1 # ls OCS-15_5/bin/isql

    isql
```

Note: If a "no such file or directory" error occurs, then you have chosen the wrong path.

9. From the command line, create a link between the identified *OCS-<version>* directory and *OCS*.

Example:

```
server1 # pwd

    /opt/sybase-15.5

server1 # ln -s OCS-15_5 OCS
```

10. Verify the link was properly created.

Example:

```
server1 # ls -ld OCS*

    lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:20 OCS -> OCS-15_5

    drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 OCS-15_5

server1 # ls OCS/bin/isql

    isql
```

Version 15_5 is used only as an example.

6.20. VMDK Shared Storage Recovery Kit Administration Guide

VMDK Shared Storage Recovery Kit Technical Documentation

LifeKeeper for Linux VMDK Shared Storage Recovery Kit (hereafter referred to as the VMDK Recovery Kit) provides a VMware virtual hard disk as shared storage. The VMDK Recovery Kit allows LifeKeeper users to employ virtual hard disks as the storage basis for their LifeKeeper hierarchies.

This guide contains the following topics:

- Documentation and References. Provides a list of LifeKeeper for Linux documentation.
- Requirements. A description of the hardware and software necessary to properly setup, install, and operate the VMDK Recovery Kit. Refer to the SIOS Protection Suite Installation Guide for specific instructions on how to install or remove LifeKeeper for Linux software.
- Overview. A description of the VMDK Recovery Kit's features and functionality.
- Configuring LifeKeeper for Linux VMDK Recovery Kit. A description of the procedures required to properly configure the VMDK Recovery Kit.
- LifeKeeper Configuration Tasks. A description of the tasks for creating and managing your VMDK resource hierarchies using the LifeKeeper GUI.
- Troubleshooting. A list of LifeKeeper for Linux error messages including a description for each.

6.20.1. VMDK Documentation and References

The following SPS product documentation is available from SIOS Technology Corp.:

- SIOS Protection Suite for Linux Release Notes
- SIOS Protection Suite for Linux Technical Documentation
- SIOS Protection Suite for Linux Installation Guide
- Optional Application Recovery Kit Documentation

This documentation along with documentation associated with the optional SPS Application Recovery Kits is available at docs.us.sios.com.

6.20.2. VMDK Hardware and Software Requirements

Your SPS configuration must meet the following requirements prior to the installation of the LifeKeeper for Linux VMDK Recovery Kit. See the SIOS Protection Suite Installation Guide for specific instructions regarding the configuration of your LifeKeeper hardware and software.

Hardware Requirements

- **Servers** – LifeKeeper for Linux supported VMware guests are configured in accordance with the requirements described in the SPS for Linux Release Notes and SPS for Linux Installation Guide.

- **IP Network Interface Cards** – Each server requires at least one Ethernet TCP/IP supported network interface card. LifeKeeper clusters require two communications paths. Two separate LAN-based communication paths using dual independent sub-nets are recommended for heartbeats and at least one of these should be configured as a private network. Using a combination of TCP and TTY heartbeats is also supported.

Software Requirements

- **TCP/IP Software** – Each server in your LifeKeeper configuration requires TCP/IP software.

- **LifeKeeper Software** – It is imperative that you install the same version of the LifeKeeper for Linux software and apply the same versions of the LifeKeeper for Linux software patches to each server in your cluster.

- **LifeKeeper for Linux VMDK Recovery Kit** – The VMDK Recovery Kit is included in the LifeKeeper installation image. It will be installed when selected on the Recovery Kit Selection screen of the setup script.

- **Linux Software** – Additional software required to run the VMDK Recovery Kit is included in the LifeKeeper installation image. Run the LifeKeeper setup script to install the VMDK Recovery Kit.

See the SPS for Linux Installation Guide for specific instructions on the installation and removal of the LifeKeeper for Linux software.

6.20.3. VMDK Recovery Kit Overview

The primary focus of the LifeKeeper for Linux VMDK Recovery Kit is to offer LifeKeeper users an alternative storage method for shared storage and data replication. The VMDK Recovery Kit enables the creation of LifeKeeper resource hierarchies on LifeKeeper protected servers. A virtual hard disk provided by the VMware hypervisor is connected to this resource hierarchy and the file system created on that disk is mounted. When a failure is detected on a node in the cluster where the virtual hard disk is connected, the VMDK Recovery Kit initiates a failover to the predetermined backup node and connects the same virtual hard disk to the backup node.

Once the file system configured on the virtual disk is mounted on a LifeKeeper server, it can be fully utilized as additional storage for LifeKeeper hierarchies. Resource hierarchies for the VMDK Recovery Kit are created using the existing File System Recovery Kit available with the LifeKeeper Core product (**steeleye-lk** package).

VMDK Recovery Kit Restrictions

- This version of the VMDK Recovery Kit does not include support for a local recovery when access to the virtual hard disk fails. When a failure is detected the default action is to initiate a transfer of the hierarchy to a backup server. Depending on the makeup of the resource hierarchy, this action can result in hung processes. To avoid hung processes, the default action can be changed to halt the server and force a failover to a backup server. To change the default switchover behavior, alter the LKVMDKERROR setting in /etc/default/LifeKeeper. See Configuring the LifeKeeper for Linux VMDK Recovery Kit for more information on LKVMDKERROR.

- All guests participating in the cluster must have the same SCSI controller configuration. The VMDK Recovery Kit reconnects the virtual hard disk to the SCSI controller on the virtual hard disk that was connected when the resource was created.

- Snapshots cannot be created or restored on a LifeKeeper protected VMDK. When the virtual hard disk is switched between nodes, consistency of the snapshots cannot be guaranteed.

6.20.4. Configuring the VMDK Recovery Kit

This section describes the VMDK Recovery Kit configuration details. It also contains information you should consider before you start to configure and administer the VMDK Recovery Kit. Refer to the SPS for Linux Technical Documentation for instructions on configuring LifeKeeper Core resource hierarchies.

6.20.4.1. VMDK Configuration Considerations

1. Install the VMDK Recovery Kit on the servers in the cluster where you want to share your virtual hard disks. Create a virtual hard disk in VMDK format.

2. Virtual hard disks must be created on a datastore that is shared with the guests that make up the cluster.

3. Since exclusive control of the virtual hard disk depends on the hypervisor, the sharing setting of the connected SCSI controller must be set to “**None**”.

4. Because this kit operates the virtual hard disks using APIs provided by VMware, it is necessary to be able to access all VMware ESXi hosts running the guests that are participating in the cluster or managed vCenter Server via https.

5. The built-in file system recovery kit used to build the VMDK hierarchy detects and removes processes that are not under LifeKeeper protection using file systems mounted in a failover condition. **It is highly recommended that only processes that are under LifeKeeper protection be configured to use a file system under VMDK protection.**

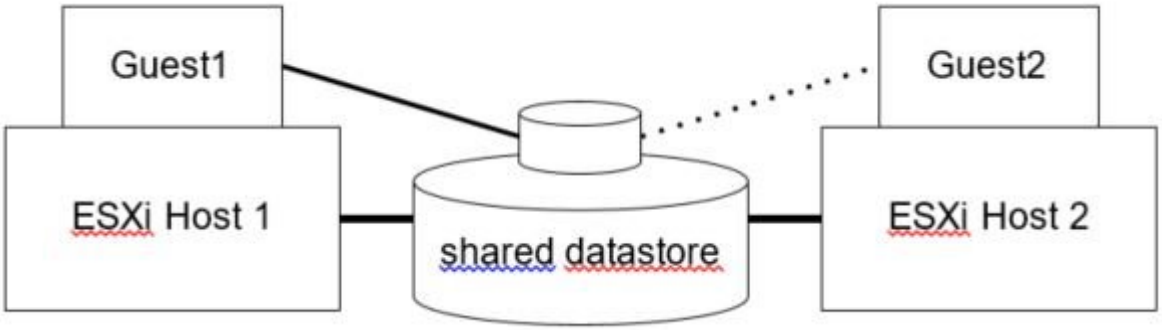
6. The LKVMDKERROR tunable controls the actions the VMDK Recovery Kit takes when access to the virtual hard disk fails. The tunable has two values, halt and event with halt being the default.

- ◦ If the value is set to **halt** and an access failure is detected, the VMDK Recovery Kit will immediately halt the system and force a failover to the backup server.

- ◦ If the value is set to **event**, the VMDK Recovery Kit notifies LifeKeeper with an abnormal status of the disk when access is lost. LifeKeeper will then attempt to initiate a switchover to a backup node. It is possible that the switchover process may hang, due to unkillable processes running on the shared VMDK.

6.20.4.2. VMDK Configuration Examples

Configuration 1: Active/Standby Configuration Example

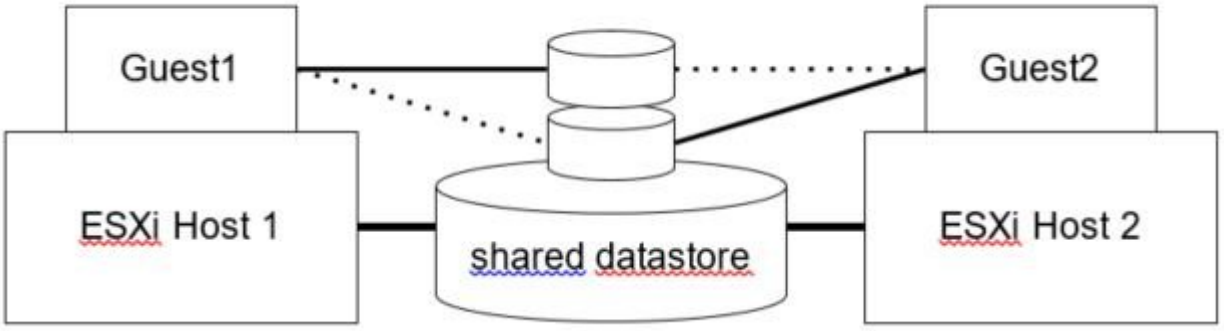


In this configuration, Guest 1 on ESXi Host 1 is considered active because it is able to access the virtual hard disk with the VMDK Recovery Kit software. If Guest 1 fails, Guest 2 gains access to the VMDK and the file system.

Configuration Notes

- The VMDK Recovery Kit must be installed on both servers.
- Create the file system on the shared VMDK virtual hard disk.
- Guest 2 should not access files and directories on the shared virtual disk while Guest 1 is active.

Configuration 2: Active/Active Configuration Example



An active/active configuration consists of two or more systems actively running the VMDK Recovery Kit software and connecting different virtual hard disks.

Configuration Notes:

- The VMDK Recovery Kit must be installed on both servers.
- Initially, Guest 1 imports a file system and Guest 2 imports a different file system. In a switchover situation, one system can import both file systems.

6.20.5. LifeKeeper VMDK Recovery Kit Configuration Tasks

You can perform all LifeKeeper for Linux VMDK Recovery Kit administrative tasks via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer, and monitor VMDK resources.

The following tasks are available for configuring the LifeKeeper for Linux VMDK Recovery Kit:

- Register an ESXi Host – Register the information on the ESXi host that manages the virtual hard disk.
- Change the VM Options – Set the options required for the VMDK Recovery Kit.
- Create a Resource Hierarchy – Creates a VMDK resource hierarchy.
- Delete a Resource Hierarchy – Deletes a VMDK resource hierarchy.
- Extend a Resource Hierarchy – Extends a VMDK resource hierarchy from the primary server to the backup server.
- Unextend a Resource Hierarchy – Unextends (removes) a VMDK resource hierarchy from a single server in the LifeKeeper cluster.
- Create Dependency – Creates a child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- Delete Dependency – Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- In Service – Activates a resource hierarchy.
- Out of Service – Deactivates a resource hierarchy.
- View / Edit Properties – View or edit the properties of a resource hierarchy.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You can also perform most of these tasks by:

1. From the toolbar, right–click on a global resource in the left pane of the status display.
2. Right–click on a resource instance in the right pane of the status display.

*Using the right-click method allows you to avoid entering information that is required when using the Edit menu.

6.20.5.1. Register ESXi Host

Before creating a VMDK resource, register the ESXi host information. Follow the steps below:

1. Execute the following command on the console screen.
   ```
   # /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -a <ESXi host name>
   ```

   - When you execute the command, you will be asked for the username and password. Enter the username and password used to log in to the

ESXi host. Failing to login will lead to an error and you will not be able to register.

2. Register each ESXi host in the cluster the same way.

3. Once all of the ESXi hosts have been registered, make sure that they have been registered correctly with the following command:
   ```
   # /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -l
   ```

   - A list of registered hosts will be provided.

4. Register the ESXi host in the same way on all nodes in the cluster.

For details of the esxi_register command, see VMDK Maintenance.

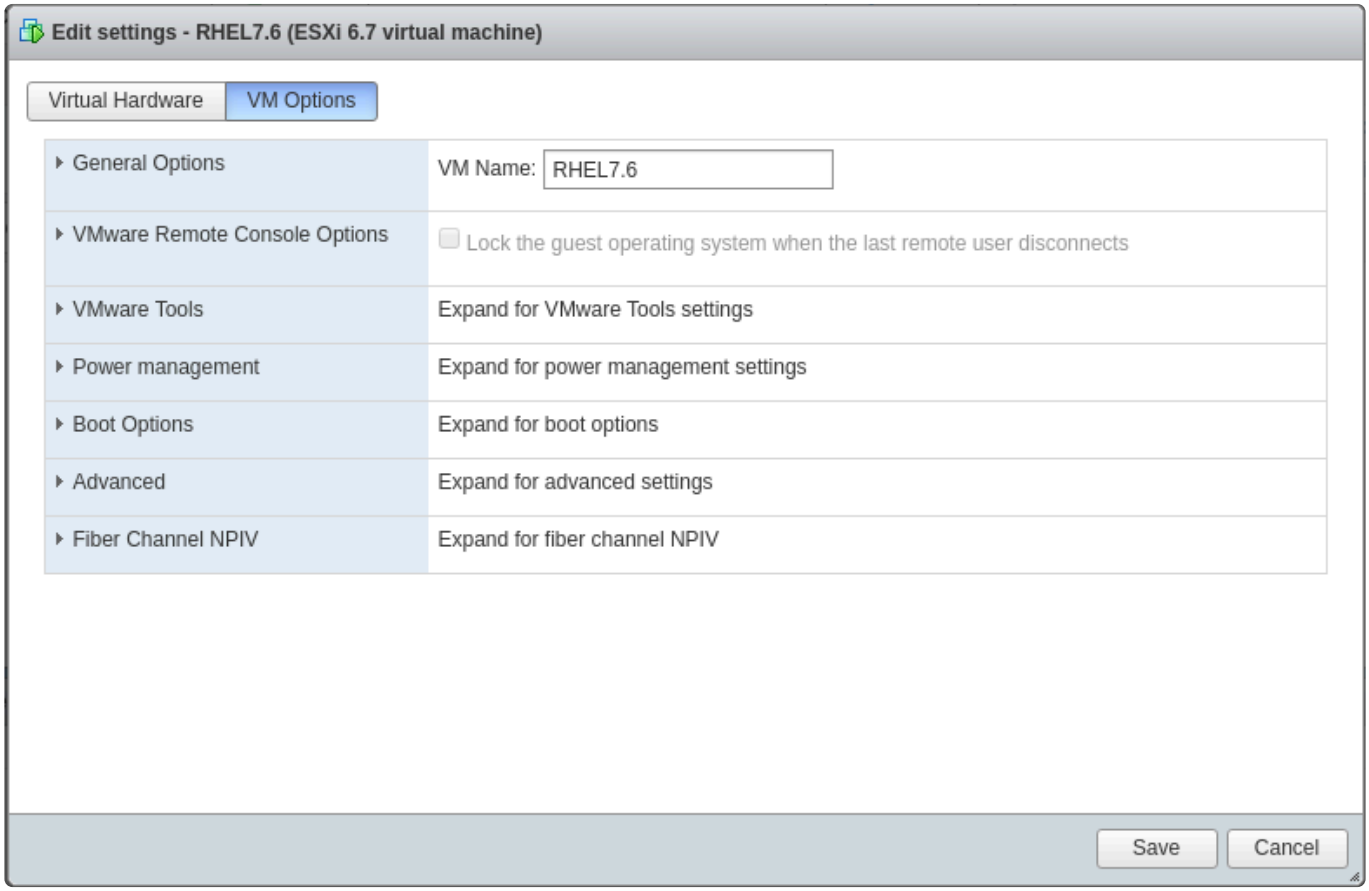
6.20.5.2. Changing the Virtual Machine Option Settings

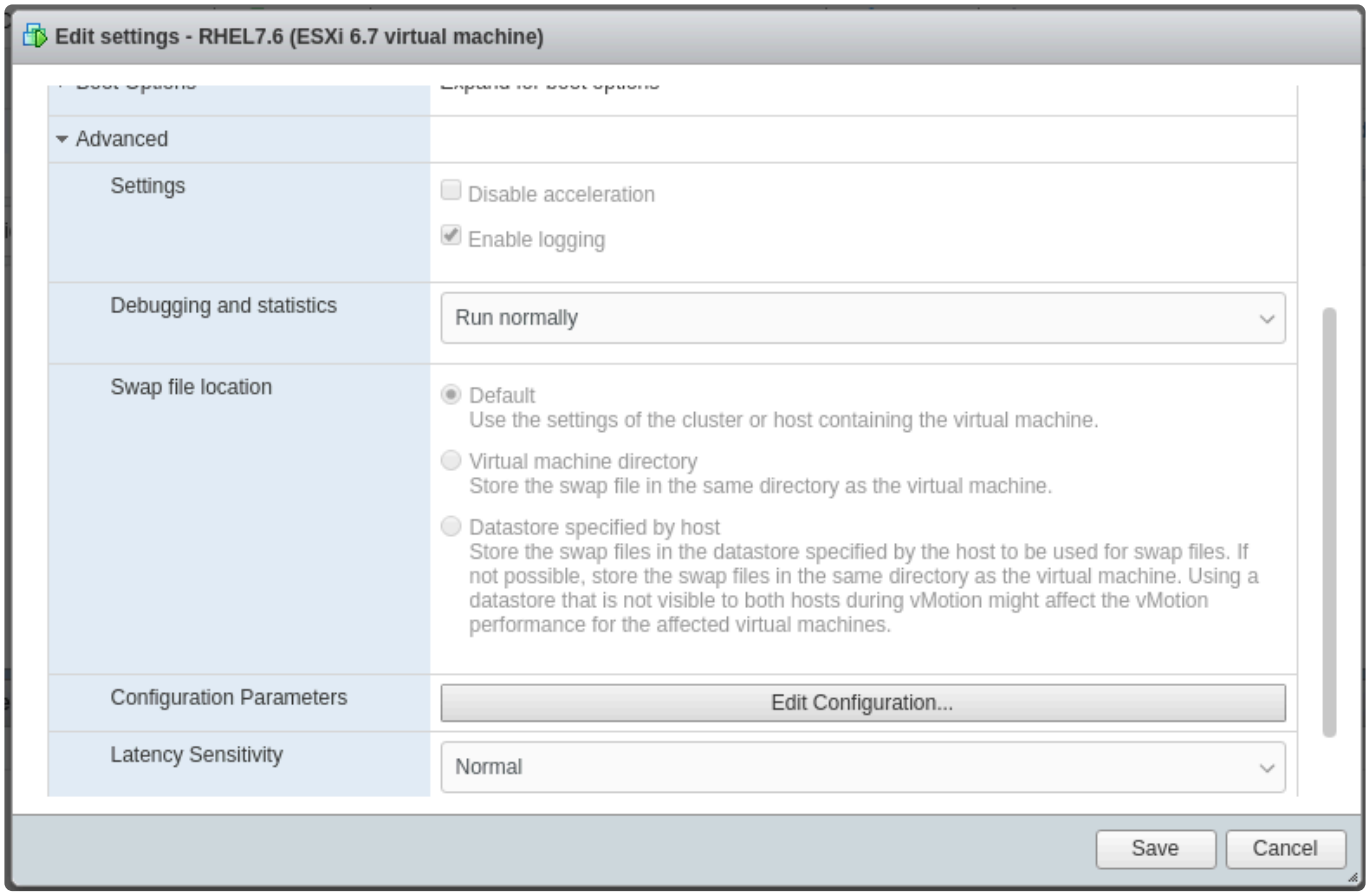
The VMDK Recovery Kit requires the following options to be set:

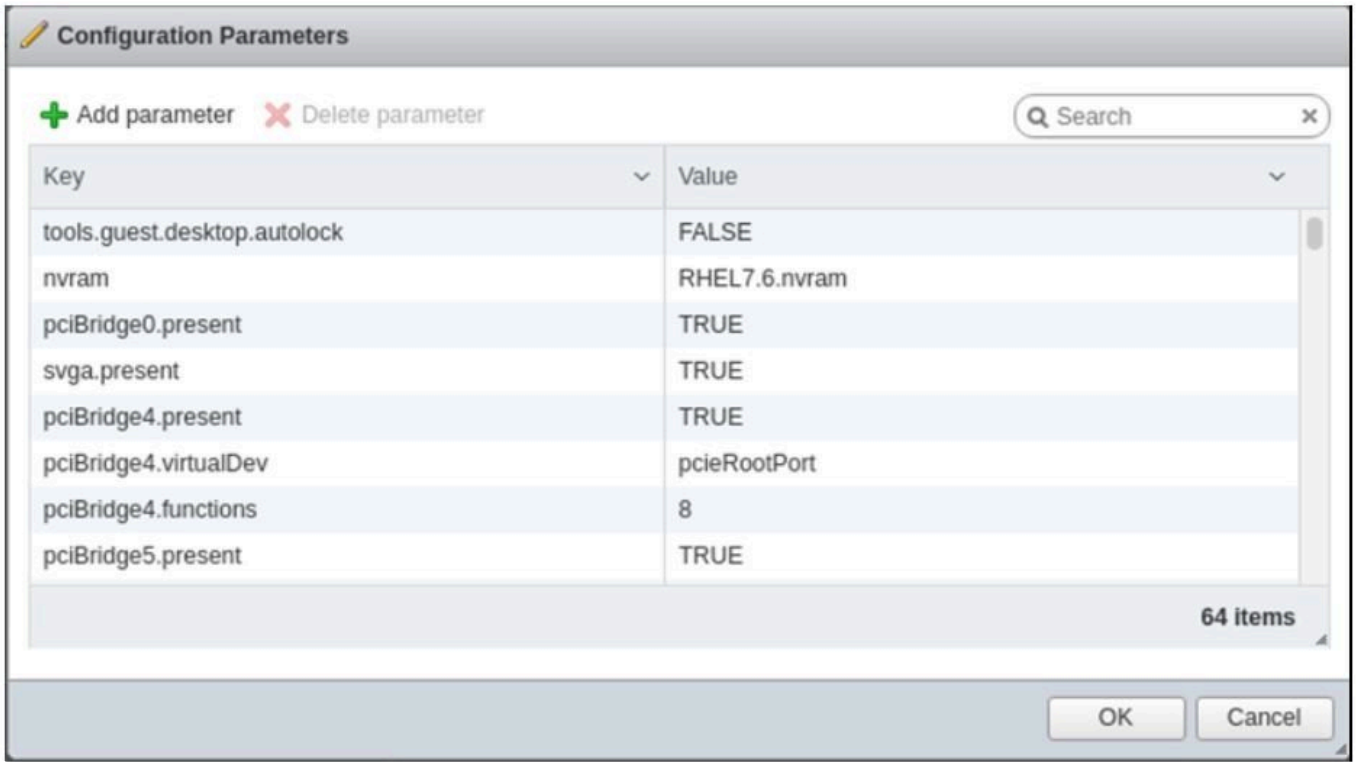
Key	Value
disk.enableUUID	TRUE

Open the edit dialog and add the above parameters for each VM:

- Edit settings
- VM Options
- Advanced
- Configuration Parameters



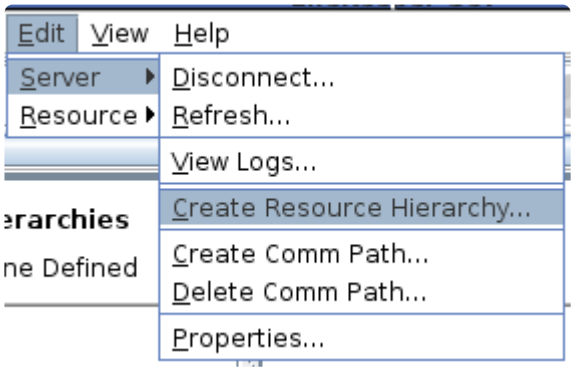

Edit settings - RHEL7.6 (ESXi 6.7 virtual machine)
Advanced
Settings
Disable acceleration
Enable logging
Debugging and statistics
Run normally
Swap file location
Default
Use the settings of the cluster or host containing the virtual machine.
Virtual machine directory
Store the swap file in the same directory as the virtual machine.
Datastore specified by host
Store the swap files in the datastore specified by the host to be used for swap files. If not possible, store the swap files in the same directory as the virtual machine. Using a datastore that is not visible to both hosts during vMotion might affect the vMotion performance for the affected virtual machines.
Configuration Parameters
Edit Configuration...
Latency Sensitivity
Normal
Save
Cancel


Configuration Parameters
Add parameter
Delete parameter
Search
Key
Value
tools.guest.desktop.autolock
FALSE
nvram
RHEL7.6.nvram
pciBridge0.present
TRUE
svga.present
TRUE
pciBridge4.present
TRUE
pciBridge4.virtualDev
pcieRootPort
pciBridge4.functions
8
pciBridge5.present
TRUE
64 items
OK
Cancel

6.20.5.3. Creating a VMDK Resource Hierarchy

Perform the following on your primary server and initiate the **Create Resource Wizard**.

1. Select **Edit** > **Server** > **Create Resource Hierarchy**



2. The **Select Recovery Kit** dialog appears. Select the **File System** option from the dropdown list. (**Note:** A VMDK Resource Hierarchy is a File System Hierarchy created on a shared virtual disk.)

Click **Next** to continue.

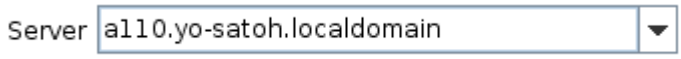
If you click the **Cancel** button at any time during the process of creating your hierarchy, LifeKeeper will cancel the entire creation process.

3. The **Switchback Type** dialog appears. The switchback type determines how the VMDK resource will be switched back to the primary server when it becomes in-service (active) on the backup server following a failover. Switchback types are either *intelligent* or *automatic*.

   Intelligent switchback requires administrative intervention to switch the resource back to the primary server while **automatic switchback** occurs as soon as the primary server is back on line and reestablishes LifeKeeper communication paths.

Click **Next** to continue.

4. The **Server** dialog appears. Select the name of the server where the VMDK resource will be created (typically this is your primary server). All servers in your cluster are included in the dropdown list.

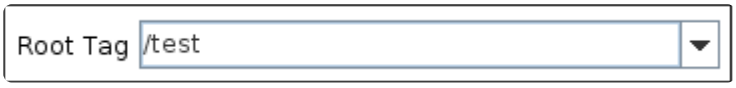


Click **Next** to continue.

5. Select the **Mount Point** path to be protected by the VMDK (File System) Resource Hierarchy. All “local” (i.e. file systems using shared storage) and mount points of the virtual hard disk that can be managed with the VMDK Recovery Kit are listed. Select the desired mount point from the dropdown list.

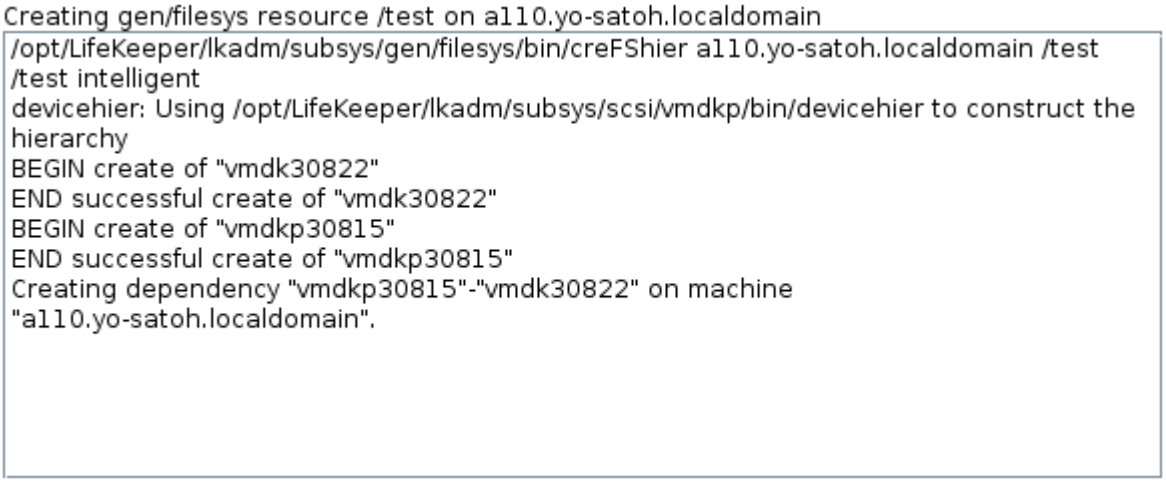
Click **Next** to continue.

6. The **Root Tag** dialog is automatically populated with a unique name for the resource instance on the target server (i.e. the server selected above). You may accept the default or enter a unique tag consisting of letters, numbers and the following special characters: **–**, **_**, **.**, or **/**.



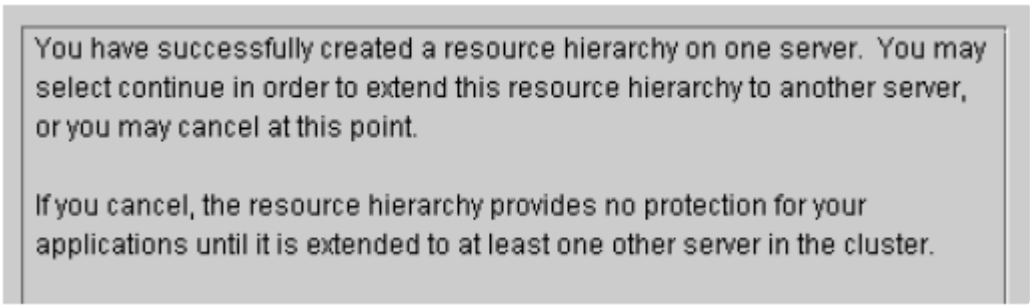
Click **Create Instance**.

7. An information box appears indicating the start of the hierarchy creation.



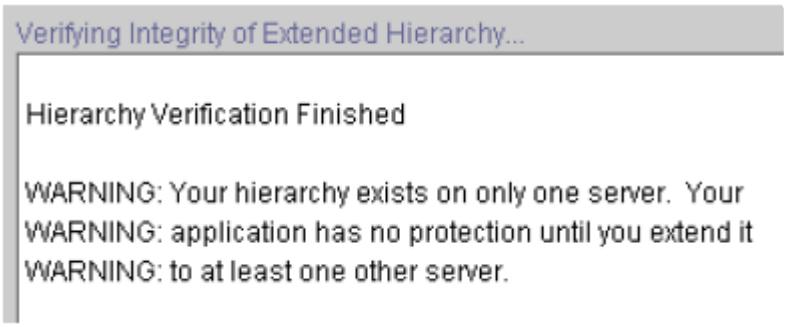
Click **Next** to continue.

8. An information box appears after the successful creation of your VMDK resource hierarchy. You must **Extend** the hierarchy to another server in your cluster in order to place it under LifeKeeper protection.



Click **Continue** to extend the resource.

Click **Cancel** if you want to extend your resource at a later time.



9. Click **Done** to exit the Create Resource Hierarchy menu.

6.20.5.4. Deleting a VMDK Resource Hierarchy

To delete a VMDK resource from all servers in your LifeKeeper configuration, complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Resource**. From the dropdown menu, select **Delete Resource Hierarchy**.

2. Select the name of the **Target Server** where you will be deleting your VMDK resource hierarchy.

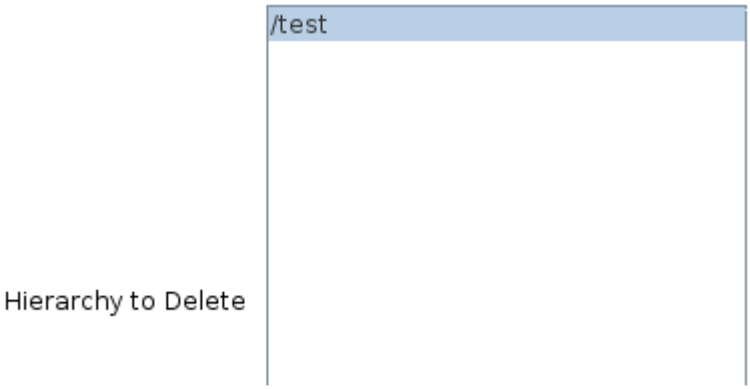
   Note: If you selected the **Delete Resource Hierarchy** by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next** to continue.

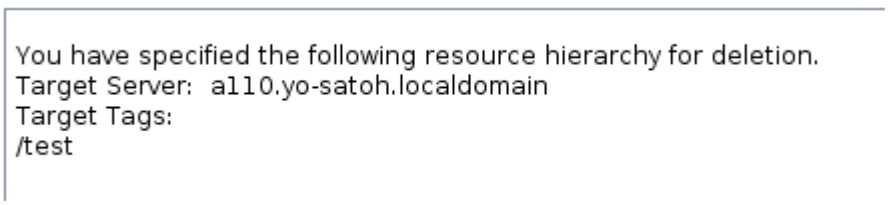
3. Select the **Hierarchy to Delete**. Identify the resource hierarchy you wish to delete, and highlight it.

   Note: If you selected the **Delete Resource Hierarchy** by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

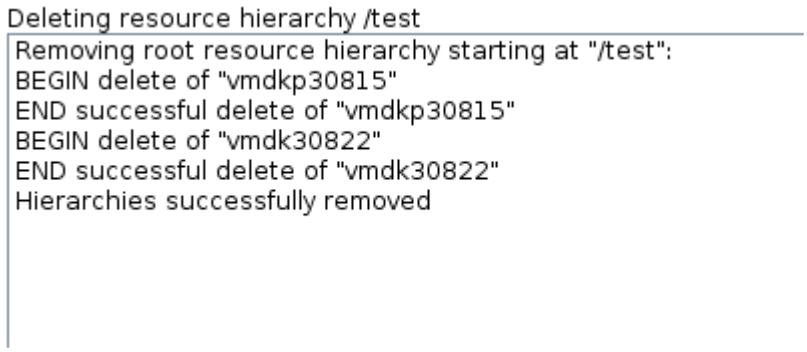
   Click **Next** to continue.

4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete.

Click **Delete** to continue.

5. An information box appears confirming that the VMDK resource instance was deleted successfully.



6. Click **Done** to exit the Delete Resource Hierarchy menu selection.

6.20.5.5. Extending Your VMDK Hierarchy

After you have created a hierarchy, you should extend that hierarchy to another server in the cluster. There are three possible ways to extend your resource instance:

1. When you successfully create your VMDK resource hierarchy you will have an opportunity to select **Continue** which will allow you to proceed with extending your resource hierarchy to your backup server.

2. Right-click on an unextended hierarchy in either the left or right pane on the LifeKeeper GUI.

3. Select the **Extend Resource Hierarchy** task from the LifeKeeper GUI by selecting **Edit**, **Resource**, **Extend Resource Hierarchy** from the dropdown menu. This sequence of selections will launch the Extend Resource Hierarchy wizard. The **Accept Defaults** button that is available for the **Extend Resource Hierarchy** option is intended for the user who is familiar with the LifeKeeper Extend Resource Hierarchy defaults and wants to quickly extend a LifeKeeper resource hierarchy without being prompted for input or confirmation. Users who prefer to extend a LifeKeeper resource hierarchy using the interactive, step-by- step interface of the GUI dialogs should use the **Next** button.

   a. The first dialog box to appear will ask you to select the **Template Server** where your VMDK resource hierarchy is currently in service. Remember that the **Template Server** you select now and the **Tag to Extend** that you select in the next dialog box represent an in-service (activated) resource hierarchy. An error message will appear if you select a resource tag that is not in service on the template server you have selected. The dropdown list in this dialog provides the names of all the servers in your cluster.

   Note: If you are entering the Extend Resource Hierarchy task by continuing from the creation of a VMDK resource hierarchy, this dialog box will not appear because the wizard has already identified the template server in the create stage. This is also the case when you right-click on either the VMDK resource icon in the left pane or right-click on the VMDK (File System) resource box in the right pane of the GUI window and choose **Extend Resource Hierarchy**.

> **CAUTION:** If you click the **Cancel** button at any time during the sequence of extending your hierarchy, LifeKeeper will cancel the extend hierarchy process. However, if you have already extended the resource to another server, that instance will continue to be in effect until you specifically unextend it.

   Click **Next** to continue.

   b. Select the **Tag to Extend**. This is the name of the VMDK instance you want to extend

from the template server to the target server. All of the resources that you have created on the template server will be listed in the dropdown.

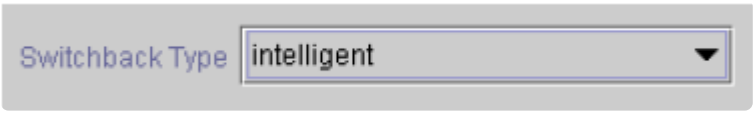
Note: If you are entering the Extend Resource Hierarchy task immediately following the creation of a VMDK hierarchy, this dialog box will not appear because the wizard has already identified the tag name of your resource in the create stage. This is also the case when you right-click on either the VMDK (File System) resource icon in the left pane or on the VMDK (File System) resource box in the right pane of the GUI window and choose **Extend Resource Hierarchy**.

Click **Next** to continue.

c. Select the **Target Server** where you will extend your VMDK resource hierarchy.

Click **Next** to continue.

d. The **Switchback Type** dialog appears. The switchback type determines how the VMDK resource will be switched back to the primary server when it becomes in service (active) on the backup server following a failover. Switchback types are either *intelligent* or *automatic*. **Intelligent switchback** requires administrative intervention to switch the resource back to the primary server while **automatic switchback** occurs as soon as the primary server is back on line and reestablishes LifeKeeper communication paths.



Click **Next** to continue.

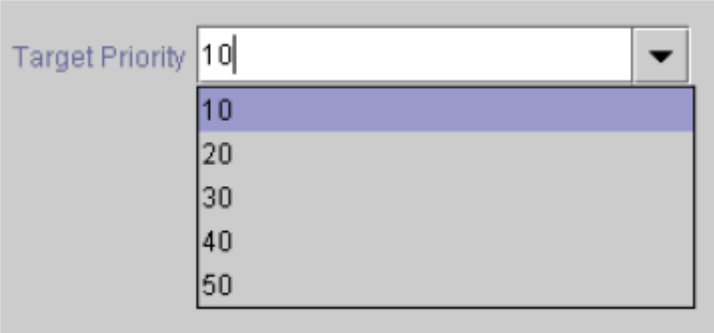
e. Select or enter a **Template Priority**. This is the priority for the VMDK hierarchy on the server where it is currently in service. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (1=highest). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended.

Note: This selection will appear only for the initial extending of the hierarchy.

Click **Next** to continue.

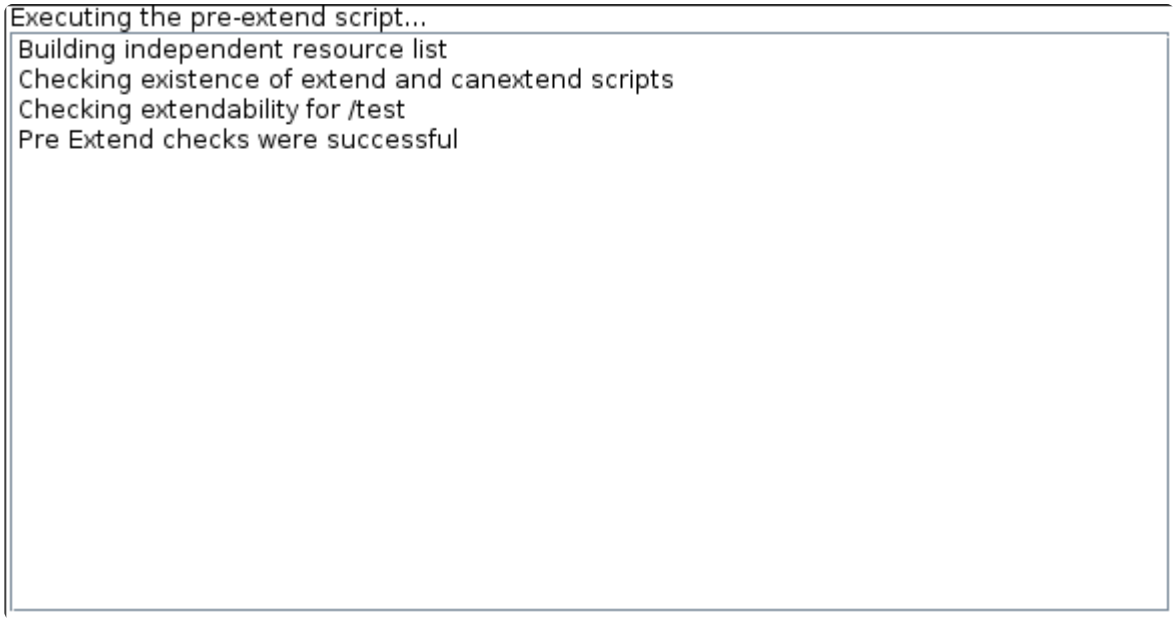
f. Select or enter the **Target Priority**. This is the priority for the new extended VMDK hierarchy relative to equivalent hierarchies on other servers. Any unused priority value from

1 to 999 is valid, indicating a server's priority in the cascading failover sequence for the resource. A lower number means a higher priority (1=highest). Note that LifeKeeper assigns the number "1" to the server on which the hierarchy is created by default. The priorities do not need to be consecutive, but no two servers can have the same priority for a given resource.



Click Next to continue.

g. An information box appears confirming that LifeKeeper has successfully checked your environment and that all requirements for extending this resource have been met. If there are requirements that have not been met, LifeKeeper will disable the **Next** button and enable the **Back** button.



Click **Back** to make changes to your resource extension.

Click **Cancel** to extend your resource another time.

Click **Next** to launch the Extend Resource Hierarchy configuration task.

Click **Finish** to confirm the successful extension of your VMDK resource instance.

4. Click **Done** to exit the Extend Resources Hierarchy menu selection.

   Note: Be sure to test the functionality of the new instance on both servers.

6.20.5.6. Unextending Your VMDK Hierarchy

1. From the LifeKeeper GUI menu, select **Edit** > **Resource** > **Unextend Resource Hierarchy**.
2. Select the **Target Server** where you want to unextend the VMDK resource. It cannot be the server where the resource is currently in-service (active).

   Note: If you selected the Unextend task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear

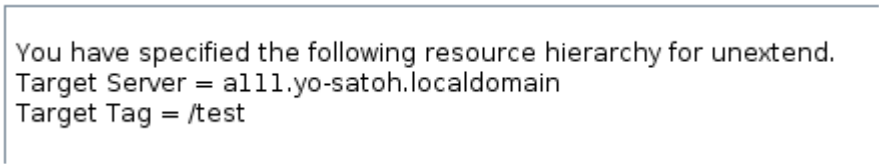
   

   Click **Next** to continue.
3. Select the **Hierarchy to Unextend**.

   Note: If you selected the Unextend task by right-clicking from either the left pane on a global resource or the right pane on an individual resource instance, this dialog will not appear.

   Click **Next** to continue.
4. An information box appears confirming the target server and the VMDK resource hierarchy you have chosen to unextend.

   Click **Unextend**.
5. Another information box appears confirming that the VMDK resource was unextended successfully.
6. Click **Done** to exit the Unextend Resource Hierarchy menu selection.

6.20.5.7. Testing Your VMDK Resource Hierarchy

You can test your VMDK resource hierarchy by initiating a manual switchover that will simulate a failover of the resource instance from the primary server to the backup server.

Performing a Manual Switchover from the LifeKeeper GUI

You can initiate a manual switchover from the LifeKeeper GUI by selecting **Edit** > **Resource** > **In Service**. For example, an in-service request executed on a backup server causes the VMDK resource hierarchy to be placed in-service on the backup server and taken out-of-service on the primary server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the Out of Service request, the resource hierarchy is taken out-of-service without bringing it in-service on the other server.

6.20.5.8. VMDK Maintenance

Recovery from a Failover Caused by a Node Failure

If a failover occurs due to an ESXi host failure, the virtual hard disk cannot be disconnected. Therefore, the virtual hard disk remains connected to the stopped guest. For this reason, multiple guests try to access the virtual hard disk when returning, but the operation is restricted by the hypervisor and therefore the guest cannot be started. In this case, use the vSphere client to manually disconnect the virtual hard disk from the guest.

Changing ESXi Login Information

To change the username and password of the ESXi host perform the following steps:

1. Stop LifeKeeper or all VMDK resources.
2. Execute the following command from the command line:
   `# /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -u <ESXi host name>`
   Enter a new username and password interactively. Once you log in successfully the information will be updated. If the VMDK resource is running or you cannot log in an error occurs and the information is not updated.
3. Start the stopped LifeKeeper or VMDK resources.
4. Repeat the same steps for all nodes and update the login information.

Deleting ESXi Host Information

To delete registered ESXi host information perform the following steps:

1. Stop LifeKeeper or all the VMDK resources.
2. Execute the following command from the command line:
   `# /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -d <ESXi host name>`
3. Start the stopped LifeKeeper or VMDK resources.
4. Repeat the same steps for all nodes and update the login information.

Esxi_register Details

Registering a host	esxi_register -a <ESXi host name>

Deleting a host	esxi_register -d <ESXi host name>
Updating login information	esxi_register -u <ESXi host name>
A list of registered hosts	esxi_register -l

6.20.6. VMDK Troubleshooting

Symptom	Possible Cause
Mount point is not included in the selection when creating resources.	Possible causes are as follows: • PowerShell/PowerCLI is not installed • An ESXi host is not registered • disk.enableUUID parameter is not set • The virtual hard disk is on a datastore that is not shared • SCSI controller sharing is configured as “virtual” or “physical” Error details are recorded in */var/log/lifekeeper.log*. Check the log and review the settings.

6.20.6.1. VMDK Error Messages

This section provides a list of messages that you may encounter while creating and extending a **LifeKeeper VMDK** resource hierarchy or removing and restoring a resource. Where appropriate, it provides an additional explanation of the cause of an error and necessary action to resolve the error condition.

Messages from other LifeKeeper components are also possible. In these cases, refer to the appropriate LifeKeeper component documentation.

The messages are grouped into the following topics:

- Common error message
- Creating a hierarchy
- Extending a hierarchy
- Deleting, restoring, recovering a hierarchy

Common Error Message

Error Number	Error Message
000002	Usage error
000010	Error getting resource information
000011	Both Tag and ID name not specified
000019	Resource not found on local server
000022	END failed hierarchy <tag name> in service on server <server name>
000026	END failed ACTION for <tag name> on server <server name> due to <signal> signal

Creating a Hierarchy

Error Number	Error Message
000012	Usage error
000013	Error getting resource information
000014	Resource with either matching tag <tag name> or ID exists
000015	ins_create failed on server <server name>
000018	Error creating resource <tag name> on server <server name>
000021	Removing resource instance <tag name> from server <server name> due to an error during creation
000023	Error bringing resource <tag name> in service on server <server name>
000024	Failed resource creation of resource <tag name> on server <server name>
000027	Removing file system dependency from <parent tag> to <child tag> on server <server name> due to an error during creation

000028	Removing file system hierarchy <filesys tag> created by <parent tag> on server <server name> due to an error during creation
000029	Switchback type mismatch between parent <parent tag> and child <child tag> on server <server name> **Action:** Switchback type mismatch can cause unexpected behavior. You can eliminate this by manually changing the switchback type using the ins_setas command.
000030	create: tag name not specified or extend: tag name not specified

Extending a Hierarchy

Error Number	Error Message
000003	Template resource <tag name> on server <server name> does not exist
000004	Template resource <tag name> cannot be extended to server <server name> because it already exists there
000005	Cannot access canextend script on server <server name>
000006	Cannot access extend script <path to extend> on server <server name>
000007	Cannot access depstoextend script <path to depstoexend> on server <server name>
000008	Cannot extend resource <tag name> to server <server name>
000009	Either <templatesys> or <templatetag> argument missing
000014	Resource with either matching tag <tag name> or ID exists
000015	ins_create failed on server <server name>
000018	Error creating resource <tag name> on server <server name>
000025	END failed resource extension of <tag name> on server <server name> due to a "<signal>" signal - backing out changes made to server
000030	create: tag name not specified or extend: tag name not specified

Restore

Error Number	Error Message
000023	Error bringing resource <tag name> in service on server <server name>

Resource Monitoring

Error Number	Error Message
000001	Calling sendevent for resource <tag name> on server <server name>

VMDK Recovery Kit

Error Number	Error Message
137000	PowerShell is not installed.
137001	PowerCLI is not installed.
137002	A valid network interface was not found.
137003	Attaching VMDK $vmdkfile.
137004	VMDK already attached.
137005	Failed to attach VMDK.
137006	Attach success.
137008	Detaching VMDK $vmdkfile.
137009	VMDK Already detached.
137010	Failed to detach VMDK.
137011	Detach success.
137012	Restarting VMDK status checker daemon.
137016	Stopping VMDK status checker daemon.
137020	Failed to execute VMDK status checker daemon.
137026	Flushing $dev.
137027	Skipping flush for $dev.
137030	Disk not specified.
137031	Cannot get disk uuid for $Disk. Please check your ESXi settings.
137032	PowerCLI failed. %s
137034	Cannot bring VMDK resource \”%s\” in service on server \”%s\”.
137035	Error detected conflict in expected tag name \”%s\” on target machine \”%s\”.
137036	Template resource \”%s\” on server \”%s\” does not exist.
137037	This system is not a VMware guest.
137038	Unable to find shared device on \”%s\” for \”%s\”.
137039	Unable to find SCSI controller on \”%s\” for \”%s\”.
137050	Failed to connect to ESXi server $addr.
137051	There is no ESXi server connected.
137055	Cannot determine ESXi VM ID because multiple network interfaces were found with the MAC address $MAC_ADDR.
137056	Failed to $action VMDK $VMDK_FILENAME. Retrying $action in $wait_sec milliseconds.
137057	Usable SCSI controller not found.

137058	Cannot find VMDK with ID $UUID.
137059	This guest has snapshots present.
137060	The VMDK with ID $UUID cannot be attached to this guest.
137061	The virtual storage controller has an incompatible sharing mode configured.
137062	VMDK_TIMER too short. Using default value.
137063	Calling sendevent for resource `"$Tag`" on server `"$me`""“
137064	skip quickcheck for `"$Tag`" on server `"$me`", sendevent pending.
137065	sendevent issued for tag `"$Tag`" has not finished, halt server `"$me`"
137066	skip quickcheck for `"$Tag`" on server `"$me`", sendevent pending.
137068	The VMDK detection failed. Retry count exceeded.
137070	Connect failed.
137071	Get-LocalVM failed.
137072	The VMDK is remote detached. This server has lost ownership.
137073	The VMDK quickCheck daemon has been stopped.
137074	VMDK_RETRY too small. Using default value.
137075	Cannot find virtual SCSI controller $CONTROLLER.
137076	The virtual storage controller has an incompatible sharing mode configured.
137077	Cannot find VM with MAC address $MAC_ADDR.
137078	Cannot find VM with MAC address $MAC_ADDR.
137100	Re-reading partition table on %s.
137101	Partition information not defined for %s on %s. Retry.
137102	Partition information not defined for %s on %s.
137103	Resource %s is OSF, Skip flushing buffers.
137104	Flushing buffers on %s.
137105	Device not specified.
137106	Cannot get device uuid for $Device. Please check your ESXi settings.
137107	%s is not shareable with any machine.
137109	Creating dependency \"%s\"-\"%s\" on machine \"%s\".
137110	Dependency \"%s\"-\"%s\" on machine \"%s\" exists.
137111	Failed to create dependency \"%s\"-\"%s\" on machine \"%s\".
137112	Cannot bring VMDKP resource \"%s\" in service on server \"%s\".
137113	detected conflict in expected tag name \"%s\" on target machine \"%s\".

7. Parameters List

Core Parameters List

The table below lists and explains names and meanings of the Core parameters. These values are tuned by editing the /etc/default/LifeKeeper configuration file.

Parameter Name	Meaning	Setting Value	Default Value	W
REMOTETIMEOUT	Number of seconds between when a process sends a request through the "lcdsendremote" function to another machine before it expects a response. If no response is received in this time interval, the function will try an alternate path if available.	Integers	900	Lifek start (**tak whe rest Life**
CONFIRMSODEF	The default action to take during machine failover processing when failover confirmation is configured. The default action is only taken when no manual response is received from the administrator within the timeout period (see CONFIRMSOTO).	0: proceed with failover 1: block the failover	0	As r (**tak imm**
CONFIRMSOTO	The time in seconds to wait for administrator action when failover confirmation is configured. When the timeout period expires the default action for CONFIMRSODEF is taken. Otherwise, the administrator action is taken.	Integers	600	As r (**tak imm**
FAILFASTTIMER	Number of seconds between verifying that a reserved device is still reserved by the local system. If the device is not reserved then the system will halt and reboot.	Integers	5	Lifek start (**tak whe rest Life**
SCSIERROR	Determines the action to take when a SCSI device cannot be opened, accessed, or another SCSI error occurs (e.g., timeout).	event:	event	Lifek start (**tak whe**

		LifeKeeper's core should be informed that a device needs to be switched over to a backup system halt: The system should immediately be halted and rebooted to avoid data corruption		**rest Life**
LKCHECKINTERVAL	Application health monitoring wait time (in seconds) between checks. Set to zero to disable health monitoring.	Integers (0, 1 and over)	120	Lifek start (**tak whe rest Life**
FILESYSFULLWARN	The file system full threshold at which time warning messages will start appearing in the LifeKeeper log. Setting to 0 will disable monitoring.	Integers	90	As r (**tak imm**
FILESYSFULLERROR	The file system full threshold at which time error messages will start appearing in the LifeKeeper log. Additionally the LKROOT/events/filesys/diskfull/notify script will be called when this threshold is reached. Setting to 0 will disable monitoring.	Integers	95	As r (**tak imm**
LK_TRAP_MGR	One or more network managers (separated by commas) to receive SNMP traps. No traps are sent if this variable is not set.	String	(not set)	As r (**tak imm**

LK_NOTIFY_ALIAS	Email address or address list used to receive notification messages when certain events occur in a LifeKeeper cluster. A null value indicates no notification will occur. The expected format is: LK_NOTIFY_ALIAS= – no notification is sent LK_NOTIFY_ALIAS=user1@domain1 – mail sent to user1 at domain1 LK_NOTIFY_ALIAS=user1@domain1,user2@domain1 – mail sent to user1 and user2 at domain1	String	(not set)	As r (**tak** **imm**
LKSYSLOGTAG	Tag for syslog.	String	LifeKeeper	Lifek start (**tak** **whe** **rest** **Life**
LKSYSLOGSELECTOR	Level for syslog.	user, daemon, local0, local1, …or local7	local6	Lifek start (**tak** **whe** **rest** **Life**
LCMHBEATTIME	The interval, in seconds, used to send heartbeats signal Failing to receive the LCM signal, which includes heartbeat signal, from another server within the interval time is determined as heartbeat stop.	Integers	5	Lifek start (**tak** **whe** **rest** **Life**
LCMNUMHBEATS	Number of consecutive missed heartbeats to mark a communication path down. In the real implemented system, it is not the number, but LCMHBEATTIME x	Integers	3	Lifek start (**tak**

	LCMNUMHBEATS seconds missing communication is determined as communication path disconnection.			**whe rest Life**
LC_MESSAGES	Changes the language environment.	String	C	Lifek start (**tak whe rest Life**
GUI_WEB_PORT	Specifies the port to use for LifeKeeper Management Web servers (lkGUI).	Integers	81	Res stee light
API_SSL_PORT	Specifies the port used for the LifeKeeper API.	Integers	778	Res stee light
LOGMGR_LOGLEVEL	Specifies the log level of Generic Applications.	LK_INFO or LK_ERROR	LK_ERROR	Lifek start resta lk_lo proc (**tak whe rest**

				Life

7.1. EC2 Parameters List

The table below lists the EC2 parameters. These values are set by adding them to the */etc/default/LifeKeeper* configuration file. Because none of the components of the Recovery Kit for EC2 are memory resident, changes to these particular values become effective immediately after they are changed in */etc/default/LifeKeeper* without requiring a LifeKeeper restart.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
EC2_RESTORE_TIMEOUT	Timeout for the resource restore, in seconds.	Integers	300	As required (**takes effect immediately**)	
EC2_REMOVE_TIMEOUT	Timeout for the resource remove, in seconds.	Integers	300	As required (**takes effect immediately**)	
EC2_RECOVER_TIMEOUT	Timeout for the local recovery, in seconds.	Integers	300	As required (**takes effect immediately**)	
EC2_QUICKCHECK_TIMEOUT	Timeout for the quickCheck, in seconds.	Integers	100	As required (**takes effect immediately**)	
EC2_AWS_REGION	Specifies the region where EC2 resources reside.	String	(not set)	As required (**takes effect immediately**)	
IP_NOLINKCHECK	Disables the link check for the protected network interface.	0: enabled 1: disabled	0	As required (**takes effect immediately**)	This value only applies when protecting an Elastic IP.
IP_WAIT_LINKDOWN	Number of seconds to wait in between taking the protected network interface down and back up. A delay between these two actions is necessary in some environments.	Integers	5	As required (**takes effect immediately**)	This value only applies when protecting an Elastic IP.
IP_MAX_LINKCHK	The maximum number of seconds to wait for the link to come back up after it has been repaired. In some environments, it may be necessary to increase this value.	Integers	5	As required (**takes effect immediately**)	This value only applies when protecting an Elastic IP.
AWSCLI_CONNECT_TIMEOUT	The connection timeout value in seconds used when running “AWS” commands. It	Integers	10	As required (**takes effect immediately**)	This is the same parameter

	is specified via --cli-connect-timeout argument.				as used in Route53.
AWSCLI_READ_TIMEOUT	The read timeout value in seconds used when running “AWS” commands. It is specified via --cli-read-timeout argument.	Integers	5	As required (**takes effect immediately**)	This is the same parameter as used in Route53.
HTTP_PROXY HTTPS_PROXY NO_PROXY	Set these parameters when using a HTTP proxy for accessing the service endpoint.The value set here is passed to AWS CLI. Please refer to AWS Documentation for details. https://docs.aws.amazon.com/cli/latest/userguide/cli-http-proxy.html	String	(not set)	As required (**takes effect immediately**)	This is the same parameter as used in Route53 and Quorum.

7.2. IP Parameters List

The table below lists and explains names and meanings of the IP parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
IP_PINGTRIES	Number of ping retries that will be performed during an IP health check.	Integers	3	As required (**takes effect immediately**)	
IP_PINGTIME	Time in seconds that LifeKeeper waits for one packet ping reply during IP health check.	Integers	1	As required (**takes effect immediately**)	When using a manually configured *Ping List* rather than the broadcast ping mechanism, any value greater than 3 for this tunable is ineffective, because the Linux TCP/IP implementation always returns a “Destination Host Unreachable” error after 3 seconds with no reply, regardless of the timeout value specified in the ping command.
NOIPUNIQUE	Disables the IP uniqueness checking done when an IP resource is brought in-service. By default LifeKeeper will ensure the IP address is not in use somewhere else on the network.	0: enabled 1: disabled	0	As required (**takes effect immediately**)	
NOBCASTPING	Disables the broadcast ping mechanism for checking the health of IP resources.	0: enabled 1: disabled	0	As required (**takes effect immediately**)	

IP_NOLINKCHECK	Disables the link status check portion of the IP health check.	0:enabled 1: disabled	0	As required (**takes effect immediately**)	This setting may need to be disabled on virtual environments, specifically after this message in the logs “Link check failed for virtual IP”.
IP_MAX_LINKCHK	The maximum number of seconds to wait for the link to come back up after it has been repaired. In some environments, it may be necessary to increase this value.	Integers	5	As required (**takes effect immediately**)	
IP_WAIT_LINKDOWN	Number of seconds to wait in between taking the protected network interface down and back up. A delay between these two actions is necessary in some environments.	Integers	5	As required (**takes effect immediately**)	

7.3. MD Parameters List

The table below lists and explains names and meanings of the MD parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
MD_ASSEMBLE_OPTIONS	User-defined options to use during mdadm --assemble.	String	(not set)	As required (**takes effect immediately**)	Refer to Software RAID Recovery Kit Notes and Restrictions of Software RAID (md) Recovery Kit documentation.

7.4. MQ Parameters List

The table below lists and explains names and meanings of the MQ parameters. These values are tuned by editing the /etc/default/LifeKeeper configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
MQS_QUICKCHECK_TIMEOUT_SC	Timeout in seconds for the server connect check.	Integers	10	As required (**takes effect immediately**)	
MQS_QUICKCHECK_TIMEOUT_CC	Timeout in seconds for the client connect check.	Integers	10	As required (**takes effect immediately**)	
MQS_QUICKCHECK_TIMEOUT_PUTGET	Timeout in seconds for the PUT/GET check.	Integers	10	As required (**takes effect immediately**)	
MQS_QUICKCHECK_TIMEOUT_PS	Timeout in seconds for checking whether publish/ subscribe is in use.	Integers	5	As required (**takes effect immediately**)	
MQS_QUICKCHECK_TIMEOUT_CLUSTER	Timeout in seconds for checking whether the queue manager is part of an WebSphere MQ cluster or not.	Integers	5	As required (**takes effect immediately**)	
MQS_QUICKCHECK_TIMEOUT	Timeout in seconds for the quickCheck script.	Integers	40	As required (**takes effect immediately**)	If the value is less than 10 seconds, it will be set to the default.
MQS_QMGR_START_TIMEOUT	Timeout in seconds for the queue manager start command to	Integers	60	As required (**takes effect immediately**)	

	complete.				
MQS_CMDS_START_TIMEOUT	Timeout in seconds for the command server start command to complete.	Integers	30	As required (**takes effect immediately**)	
MQS_LISTENER_START_TIMEOUT	Timeout in seconds for the listener start command to complete.	Integers	30	As required (**takes effect immediately**)	
MQS_LISTENER_LIST_TIMEOUT	Timeout in seconds for the listener list command to complete.	Integers	10	As required (**takes effect immediately**)	
MQS_CHECK_TIMEOUT_ACTION	The action in case a server connect check or client connect check times out.	ignore: a message about the timeout is logged, but no recovery is initiated sendevent: local recovery is initiated in case a server connect check timed out	ignore	As required (**takes effect immediately**)	
MQS_LISTENER_CHECK_DELAY	Time in seconds between the start of the listener and the check for the successful listener start. The default	Integers	2	As required (**takes effect immediately**)	If the value is less than 2 seconds, it will be set to the default.

	of 2 seconds should be sufficient to detect port in use conditions.				
NO_AUTO_STORAGE_DEPS	Determines if the shared storage checks and file system resource creation step are performed for the queue manager and log storage directories during MQ resource hierarchy creation. A value of 0 indicates these tasks will be perform. A value of 1 will bypass these tasks.	0 or 1	0	As required (**takes effect immediately**)	
MQS_DSPMQVER_TIMEOUT	Timeout in seconds for the dspmqver command (needed to find out the version of WebSphere MQ), must be at least 2 seconds.	Integers	5	As required (**takes effect immediately**)	
MQS_SKIP_CRT_MISSING_Q	Determines if missing test queue is automatically created. A value of 0 indicates missing test queues will automatically be created. A value of 1	0 or 1	0	As required (**takes effect immediately**)	

	indicates this process will be skipped.				
MQS_ALT_USER_NAME	The alternate user name to use for all WebSphere MQ commands. By default the user mqm is used. If set the alternate user must have its primary group set to the group mqm or must have secondary membership in that group.	Character string	mqm if not set or the user does not have membership in the "mqm" group.	As required (**takes effect immediately**)	Should only be set if a WebSphere MQ addon package requires a user other than "mqm"

7.5. NFS Parameters List

The table below lists and explains names and meanings of the NFS parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
RESTARTMOUNTD	Enables the stop and restart of rpc.mount on all NFS restores.	true: enabled false: disabled	true	As required (**takes effect immediately**)	

7.6. Oracle Parameters List

The table below lists and explains names and meanings of the Oracle parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
ORACLE_ORATABLOC	*/etc/oratab* is used by default. Specifies the directory where the alternative oratab file is located. */etc/oratab* is referenced, but if the file does not exist, oratab in the directory specified with this parameter is read.	String	/var/opt/oracle	As required (**takes effect immediately**)	
LK_ORA_NICE	Determines whether a recovery attempt will occur on a database connection failure caused when the maximum number of allowed connections has been reached. A recovery attempt when the maximum number has been reached can cause a failover to the standby node.	0: execute the recovery attempt 1: prevent the recovery attempt	0	As required (**takes effect immediately**)	
ORACLE_RESTORE_TIMEOUT	Specifies the timeout value in seconds when starting Oracle resources.	Integers	300	As required (**takes effect immediately**)	
ORACLE_REMOVE_TIMEOUT	Specifies the timeout value in seconds when stopping Oracle resources.	Integers	300	As required (**takes effect immediately**)	

7.7. PostgreSQL Parameters List

The table below lists and explains names and meanings of the PostgreSQL parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Val
LKPGSQL_KILLPID_TIME	Time in seconds to wait after a process id is killed before rechecking for this process.	Integers	3
LKPGSQL_CONN_RETRIES	Replaces LKPGSQLMAXCOUNT – number of times to try a client connection after an action (start or stop).	Integers	12
LKPGSQL_ACTION_RETRIES	Number of times to attempt start or stop action before failing the action command.	Integers	4
LKPGSQL_STATUS_TIME	Timeout in seconds for the status command.	Integers	17 + (3 * LKPGSQL_KILLPI
LKPGSQL_QCKHANG_MAX	Number of quickCheck script hangs allowed before a failover/sendevent is triggered for the database instance.	Integers	2
LKPGSQL_CUSTOM_DAEMON	Allows a user to specify additional aliases for the postgres daemons (postgres.bin,postmaster,postgres,edb-postgres).	String	(not set)
LKPGSQL_IDIRS	Replaces LKPGSQL_IPORTS – contains datadir entries for instances that will be shutdown using the immediate option only.	String	(not set)
LKPGSQL_SDIRS	Contains datadir entries for instances that will be shutdown using the smart option.	String	(not set)
LKPGSQL_DISCONNECT_CLIENT	Controls whether active clients will be disconnected in the event of a postmaster crash. When the value is set to 1, client processes will be sent a	0:	1

	SIGTERM signal to force them to disconnect from the database. This action will only be taken if the postmaster process is not running during local recovery.	enabled 1: disabled	
LKPGSQL_DISCONNECT_CLIENT_BYTAG	Similar to LKPGSQL_DISCONNECT_CLIENT, this setting limits the action to the comma separated list of tags specified by this tunable.	String	(not set)
LKPGSQL_RESUME_PROC	Determines if a process found in the stopped state (state = ~T) will be resumed when detected or ignored.	0:ignore 1:resume	1
LKPGSQLDEBUG	Turns on debug for PostgreSQL database kit as well as for the postgres database. Valid entry range: 0 – 5. This parameter will be passed on to the postmaster database using the option –d <LKPGSQLDEBUG>.	Integers (0 – 5)	0

7.8. Quorum Parameters List

The table below lists the Quorum parameters. These values are tuned by editing the /etc/default/LifeKeeper configuration file.

Parameter Name	Meaning	Setting Value	De Va
QUORUM_MODE	Specifies the quorum mode.	majority tcp_remote storage none or off	major
QUORUM_HOSTS	Specifies a host and port name combination to be used for determining quorum. The format for entries is “host:port”. When specifying more than one host:port combination, the entries must be comma separated (do not include a space). (Example) QUORUM_HOSTS=myhost:80,router1:443,router2:22	String	(not s
WITNESS_MODE	Specifies the witness mode.	remote_verify storage none or off	remot
QUORUM_TIMEOUT_SECS	The time allowed for tcp/ip witness connections to complete. Connections that don’t complete within this time are treated as failed/unavailable. This only applies when the QUORUM_MODE is tcp_remote.	Integers	20
QUORUM_LOSS_ACTION	Specifies the action when quorum is lost.	fastkill fastboot osu	fastkil

QWK_STORAGE_TYPE	Specifies the type of shared storage. Must be specified when QUORUM_MODE is storage.	block file aws_s3	(not s
QWK_STORAGE_HBEATTIME	Specifies the interval in seconds between reading and writing the QWK objects.	An integer between 5 and 10	6
QWK_STORAGE_NUMHBEATS	Specifies the number of consecutive heartbeat checks that when reached indicates the target node has failed. A missed heartbeat occurs when the QWK object has not been updated since the last check.	An integer of 3 or more	4
QWK_STORAGE_OBJECT_<Host name> **Note**: If the host name contains a "-" or ".", replace them with an underscore "_" (e.g. lksios-1 → lksios_1). **Note**: The host name used by LifeKeeper can be checked via the lcduname command	Specifies the path to the QWK objects. You must specify paths for all nodes in the cluster. **[When QWK_STORAGE_TYPE is block]** Specify the device file path. **Note:** Use WWID (/dev/disk/by-id/) to specify a permanent path. (Example) QWK_STORAGE_OBJECT_nodeA=/dev/disk/by-id/xxxxx QWK_STORAGE_OBJECT_nodeB=/dev/disk/by-id/yyyyy **[When QWK_STORAGE_TYPE is file]** Specify the regular file path. (Example) QWK_STORAGE_OBJECT_nodeA=/quorum/nodeA QWK_STORAGE_OBJECT_nodeB=/quorum/nodeB	String (Maximum length is 256 characters)	(not s

	[When QWK_STORAGE_TYPE is aws_s3] Specify the S3uri for the Amazon S3 object. Use an S3 object from a different region than the one where LifeKeeper is running. It is also recommended that 2 different S3 objects be used and that they reside in different regions. When specifying 2 regions, make sure to separate them with commas(do not include spaces). (Example 1) QWK_STORAGE_OBJECT_nodeA=s3://bucket1/nodeA,s3://bucket2/nodeA QWK_STORAGE_OBJECT_nodeB=s3://bucket1/nodeB,s3://bucket2/nodeB (Example 2) QWK_STORAGE_OBJECT_nodeA=s3://bucket/quorum/nodeA QWK_STORAGE_OBJECT_nodeB=s3://bucket/quorum/nodeB		
HTTP_PROXY HTTPS_PROXY NO_PROXY	Set this parameter when using HTTP proxy for accessing the service endpoint. The value set here will be passed to AWS CLI. See AWS Documentation for details.	String	(not s
QUORUM_DEBUG	Specifies the debug mode.	0: disabled 1: enabled	0

7.9. Route53 Parameters List

The table below lists and explains the tunable values that are available for modifying the behavior of the Route53 Recovery Kit. These values are set by adding the */etc/default/LifeKeeper* configuration file. Because none of the components of the Route53 Recovery Kit are memory resident, changes to these particular values become effective immediately after they are changed in */etc/default/LifeKeeperwithout* requiring a restart of LifeKeeper or the OS.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
ROUTE53_TTL	The default setting value for TTL (Time To Live) of the A record created for the Route53 resource, in seconds.	Integers	10	After switchover	
ROUTE53_CHANGEID_INTERVAL	The interval of Route 53 API communications when checking the status, in seconds.	Integers	20	As required (**takes effect immediately**)	
ROUTE53_CHANGEID_TRY_COUNT	The number of trials of Route 53 API communications when checking the status.	Integers	5	As required (**takes effect immediately**)	
AWSCLI_CONNECT_TIMEOUT	The connection timeout value in seconds used when running “AWS” commands. It is specified via —cli-connect-timeout argument.	Integers	10	As required (**takes effect immediately**)	This is the same parameter as used in EC2.
AWSCLI_READ_TIMEOUT	The read timeout value in seconds used when running “AWS” commands. It is specified via --cli-read-timeout argument.	Integers	5	As required (**takes effect immediately**)	This is the same parameter as used in EC2.
HTTP_PROXY HTTPS_PROXY NO_PROXY	Set these parameters when using HTTP proxy for accessing the service endpoint.The value set here is passed to AWS CLI. Please refer to AWS Documentation for details. https://docs.aws.amazon.com/cli/latest/userguide/cli-http-proxy.html	String	(not set)	As required (**takes effect immediately**)	This is the same parameter as used in EC2 and Quorum.

7.10. SAP Parameters List

The table below lists and explains names and meanings of the SAP parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
SAP_CONFIG_REFRESH	Refresh time in seconds of the Configuration properties page.	Integers	LKCHECKINTERVAL/2	As required (**takes effect immediately**)	If the value is less than 5 seconds, it will be set to the default.
SAP_CREATE_NAS	Automatically includes a NAS resource for NAS mounted file systems.	0: disabled 1: enabled	1	As required (**takes effect immediately**)	
SAP_NFS_CHECK_DIRS	Comma-separated list of NFS mount points to check	String	empty	As required when using NFS shared filesystems	Do not use for Amazon EFS mount points
SAP_QUICKCHECK_TIMEOUT	Timeout in seconds for the quickCheck process.	Integers	60 seconds	As required (**takes effect immediately**)	
SAP_RESTORE_TIMEOUT	Timeout in seconds for the restore process.	Integers	516 seconds	As required (**takes effect immediately**)	
SAP_REMOVE_TIMEOUT	Timeout in seconds for the remove process.	Integers	804 seconds	As required (**takes effect immediately**)	
SAP_RECOVER_TIMEOUT	Timeout in seconds for the recover process.	Integers	1320 seconds	As required (**takes effect immediately**)	If the value is less than the default, it will be set to the

					default.
SAP_DEBUG	Enables debugging.	0: disabled 1: enabled	0	As required (**takes effect immediately**)	

7.11. DataKeeper Parameters List

The table below lists and explains names and meanings of the DataKeeper parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	
LKDR_CHUNK_SIZE	Sets the chunk size of bitmap in kilobits.	Integers	256	Creating a resource	
LKDR_CONNECT_NBD_DURING_RESTORE	Specifies if the NBD connection should be established when the mirror is restored (brought in-service).	True or False	True	To improve switchover and failover performance on a cluster with multiple mirrors and multiple targets	The defa (“true”) p maximu integrity that whe restored immedia data to a available in-servic performa setting) connecti establish first quic interval LKCHEC 120 sec default). will decr required mirror in also cre where d replicati targets. performa improve noticeab multiple multiple
LKDR_SPEED_LIMIT	Specifies the maximum bandwidth that a resync will ever take – this should be set high enough to allow resyncs to go at the maximum speed possible.	Integers	50000	Resyncing a resource	

LKDR_SPEED_LIMIT_MIN	Specifies how fast the resync should be allowed to go when there is other I/O going on at the same time. As a rule of thumb, this should be set to half or less of the drive's maximum write throughput in order to avoid starving out normal I/O activity when a resync occurs.	Integers	20000	Resyncing a resource	
LKDR_ASYNC_LIMIT	Specifies the number of outstanding target writes that can be in flight at a given time. Increase this value for higher performance with asynchronous mirrors.	Integers	4096	Creating a resource	
LKDR_NO_FULL_SYNC	Suppresses a full resync of newly added targets.	0: not suppress 1: suppress	0	As required (**takes effect immediately**)	For deta informat "Avoidin Resynch
LKDR_WAIT_FOR_PREVIOUS_SOURCE_TIMEOUT	Specifies how long to wait for the previous source to join the cluster so that its bitmap can be merged. A full resync is required if	0: do not wait -1: wait indefinitely for the previous	-1	As required (**takes effect immediately**)	How SI DataKe

	replication is resumed to a target before the previous source's bitmap is merged. This setting applies to all netraid devices; individual devices can NOT be configured with a different value.	source to rejoin the cluster > 0 number of seconds to wait			
NBD_NR_REQUESTS	Specifies the number of outstanding I/O requests that can be in flight at a given time. Increase this value for higher mirroring performance.	Integers	2048	As required (**takes effect immediately**)	

7.12. Standby Node Health Check Parameters List

It is necessary to enable/disable each functionality and configure the value for the Standby Node Health Check. These settings can be customized in the */etc/default/LifeKeeper* configuration file.

Parameter	Description	Value to Set	Default Value	When to Apply	Notes
SNHC	Enables the overall functionality of the Standby Node Health Check. To enable, set this parameter to 1; it is disabled by default. SNHC_XX for individual functions must be enabled.	0: Disabled 1: Enabled	0	When *lkcheck* is started	After setting, restart *lkcheck*. See Standby Node Health Check for details.
SNHC_CPUCHECK	Enables CPU monitoring with Node Monitoring. To enable, set this parameter to 1; it is disabled by default.	0: Disabled 1: Enabled	0	Anytime	
SNHC_CPUCHECK_THRESHOLD	Specifies the threshold for CPU utilization that is considered to be abnormal in CPU monitoring. If not specified, 99 is used.	Integer value between 10 and 99	99	Anytime	If not configured users are prompted to configure with an ERROR message in the log.
SNHC_CPUCHECK_TIME	Specifies the number of consecutive times that CPU utilization must be over the threshold (SNHC_CPUCHECK_THRESHOLD) to be considered an error. If not specified, 1 is used.	Integer value between 1 and 100	1	Anytime	If not configured users are prompted to configure with an ERROR message in the log.

SNHC_MEMCHECK	Enables memory monitoring with Node Monitoring. To enable, set this parameter to 1; it is disabled by default.	0: Disabled 1: Enabled	0	Anytime	
SNHC_MEMCHECK_THRESHOLD	Specifies the threshold for memory utilization that is considered to be abnormal in memory monitoring. If not specified, 99 is used.	Integer value between 10 and 99	99	Anytime	If not configured users are prompted to configure with an ERROR message in the log.
SNHC_MEMCHECK_TIME	Specifies the number of consecutive times that memory utilization must be over the threshold (SNHC_MEMCHECK_THRESHOLD) to be considered an error. If not specified, 1 is used.	Integer value between 1 and 100	1	Anytime	If not configured users are prompted to configure with an ERROR message in the log.
SNHC_IPCHECK	Enables OSU resource monitoring for IP resources. It is disabled by default.	0: Disabled 1: Enabled	0	Anytime	
SNHC_IPCHECK_SLEEPTIME	Specifies the wait time to check the link after starting the NIC, when the relevant NIC is down during OSU monitoring for IP resources. If the NIC was down before monitoring, the NIC will be left down after monitoring.	Wait time (sec)	1	Anytime	
SNHC_DISKCHECK	Enables OSU resource monitoring for DMMP resources. It is disabled by default.	0: Disabled 1: Enabled	0	Anytime	

7.13. SAP HANA Parameters List

The table below lists and explains names and meanings of SAP HANA parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Not
HANA_STOP_WAIT	Specifies the amount of time (in seconds) to wait when stopping the SAP HANA database instance.	Positive integers	600	As required (**takes effect immediately**)	
HANA_STOP_COUNT	Specifies the number of attempts to issue the sapcontrol StopWait command when stopping the SAP HANA database instance.	Positive integers	3	As required (**takes effect immediately**)	
HANA_STOP_FINAL_WAIT	After HANA_STOP_COUNT failed attempts to stop the SAP HANA database, this specifies the amount of additional time (in seconds) to wait to allow the database to fully stop.	Positive integers	60	As required (**takes effect immediately**)	
HANA_START_WAIT	Specifies the amount of time (in seconds) to wait when stopping the SAP HANA database instance.	Positive integers	2700	As required (**takes effect immediately**)	
HANA_QUICKCHECK_TIMEOUT	Specifies the timeout value (in seconds) for the SAP HANA quickCheck script.	Positive integers, Minimum value 30	LKCHECKINTERVAL – 10	As required (**takes effect immediately**)	If set to than 30 will automa default
HANA_RECOVER_TIMEOUT	Specifies the timeout value (in seconds) for the SAP HANA recover script.	Positive integers	1800	As required (**takes effect immediately**)	
HANA_HSR_POLL_INTERVAL	Specifies how often (in seconds) LifeKeeper will poll the status of SAP HANA System Replication.	Positive integers	10	As required (**takes effect immediately**)	

7.14. SAP MaxDB Parameters List

The table below lists and explains names and meanings of the SAP MaxDB Recovery Kit parameters. These values are tuned by editing the */etc/default/LifeKeeper* configuration file.

Parameter Name	Meaning	Setting Value	Default Value	When to Apply	Notes
MAXDB_START_TIMEOUT	Number of seconds to wait before aborting an in-service action that is hung.	Integers	300	As required (**takes effect immediately**)	
MAXDB_PID_CLEANUP	Specifies whether child processes of a hung in-service action will be cleaned up (killed).	y: clean up n: no cleanup	n	As required (**takes effect immediately**)	
MAXDB_STOP_COUNT	Number of times LifeKeeper will attempt to offline the database.	Integers	5	As required (**takes effect immediately**)	
MAXDB_WAIT	Number of seconds to wait between database offline attempts.	Integers	5	As required (**takes effect immediately**)	
MAXDB_DEBUG	Enables or disables debug logging.	0: disabled 1: enabled	0	As required (**takes effect immediately**)	
MAXDB_OFFLINE_ENABLED	Specifies whether the database will be taken offline when the LifeKeeper MaxDB resource is taken out of service.	0: do not offline 1: do offline	1	Use with caution, if you need to temporarily stop LifeKeeper (e.g., to upgrade or perform maintenance) while not stopping the database.	Please ensure you re-enable database offlines after maintenance is complete.

8. Search for an Error Code

Please look at the Combined Message Catalog

8.1. Combined Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
000200	ERROR	pam_start() failed	
000201	ERROR	pam_authenticate failed (user %s, retval %d	
000202	ERROR	pam_end() failed?!?!	
000203	ERROR	Did not find expected group ‘lkguest’	
000204	ERROR	Did not find expected group ‘lkoper’	
000205	ERROR	Did not find expected group ‘lkadmin’	
000208	ERROR	pam_setcred establish credentials failed (user %s, retval %d	**Cause**: Unable to establish valid l for user {user}. The pam_setcred {retval}. **Action**: Check /var/log/security ar messages for more information.
000209	ERROR	pam_setcred delete credentials failed (user %s, retval %d	**Cause**: Unable to clear login cred {user}. The pam_setcred call retur **Action**: Check /var/log/security ar messages for more information.
000902	ERROR	Error removing system name from loopback address line in /etc/hosts file. You must do this manually before starting the GUI server.	**Cause**: System name did not get /etc/hosts file. **Action**: Remove system name ma restart the GUI server, then enter run <action name>
000918	ERROR	LifeKeeper GUI Server error during Startup	**Cause**: The GUI server terminate abnormal condition. **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
001052	FATAL	Template resource "%s" on server "%s" does not exist	**Cause**: LifeKeeper was unable to {tag} on {server}.
001053	ERROR	Cannot access canextend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to checks because it was unable to f CANEXTEND on {server}. **Action**: Check your LifeKeeper co
001054	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}.
001055	ERROR	Cannot access extend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to resource hierarchy because it was find the script EXTEND on {server **Action**: Check your LifeKeeper co
001057	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}.
001059	ERROR	Resource with tag "%s" already exists	**Cause**: The name provided for a r already in use. **Action**: Either choose a different r resource, or use the existing resou
001060	ERROR	Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s"	**Cause**: The name or id provided f already in use. **Action**: Either choose a different r resource or use the existing resou
001061	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected failure occ

Code	Severity	Message	Cause/Action
			creating a resource. **Action**: Check the logs for related resolve the reported problem.
001081	WARN	IP address \"$ip\" is neither v4 nor v6	**Cause**: The IP address provided i nor an IPv6 address. **Action**: Please check the name o provided and try again. If a name verify that name resolution returns address.
004024	ERROR		**Cause**: LCD failed to fetch resour resource id {id} during resource re **Action**: Verify the input resource i recovery operation.
004028	ERROR	%s occurred to resource \"%s\"	**Cause**: Local recovery failed for r {resource}. **Action**: Check the logs for related resolve the reported problem.
004055	ERROR	attempt to remote-remove resource \"%s\" that can’t be found	**Cause**: Remotely removing a resc failed while attempting to find the name {tag}. **Action**: Check input tag name and recovery operation.
004056	ERROR	attempt to remote-remove resource \"%s\" that is not a shared resource	**Cause**: Remotely removing a resc failed given that the tag name {tag resource. **Action**: Check input tag name and

Code	Severity	Message	Cause/Action
			recovery operation.
004060	ERROR	attempt to transfer-restore resource \"%s\" that can’t be found	**Cause**: Remote transfer of an in s failed given tag name {tag}. **Action**: Check input tag name and recovery operation.
004061	ERROR	attempt to transfer-restore resource \"%s\" that is not a shared resource with machine \"%s\"	**Cause**: LifeKeeper failed to find a by {tag} name during remote trans in service from a remote {machine **Action**: Check input tag name and recovery operation.
004089	ERROR	ERROR: Parallel recovery initialization failed.\n	**Cause**: Parallel recovery failed to resources in the hierarchy. **Action**: Check the logs for related resolve the reported problem.
004091	ERROR	ERROR: fork failed. continuing to next resource\n	**Cause**: Parallel recovery failed to process attempting to restore a si
004093	ERROR	ERROR: reserve failed. continuing to next resource\n	**Cause**: Parallel recovery failed to resource from the collective hierar **Action**: Check the logs for related resolve the reported problem.
004096	ERROR	ERROR: clone %d is hung, attempting to kill it\n	**Cause**: A single sub process of a is hung during parallel recovery of hierarchy. **Action**: A kill of the hanging sub p executed automatically.

Code	Severity	Message	Cause/Action
004097	ERROR	ERROR: Could not kill clone %d\n	**Cause**: Failed to kill the hung sub
004116	ERROR	%s	**Cause**: Writing an on-disk version data object failed attempting to cr intermediate folder. This is a syste **Action**: Check log for the error inf and determine why the intermedia created.
004117	ERROR	open(%s	**Cause**: Writing an on-disk version data object failed while attempting temporary file. This is a system er **Action**: Check log for the error inf and determine why the file is not s opened.
004118	ERROR	write(%s	**Cause**: Writing an on-disk version data object failed while attempting temporary file. This is a system er **Action**: Check log for the error inf and determine why the file writing
004119	ERROR	fsync(%s	**Cause**: Writing an on-disk version data object failed while attempting temporary file. This is a system er **Action**: Check log for the error inf and determine why the "fsync" is f
004120	ERROR	close(%s	**Cause**: Writing an on-disk version data object failed while attempting temporary file. This is a system er **Action**: Check log for the error inf

Code	Severity	Message	Cause/Action
			and determine why the file closing
004121	ERROR	rename(%s, %s	**Cause**: Writing an on-disk version data object failed while attempting temporary file {file} to original file system error. **Action**: Check log for the error inf and determine why the file renami
004122	ERROR	open(%s	**Cause**: Writing an on-disk version data object failed while attempting intermediate directory {directory}. error." **Action**: Check log for the error inf and determine why the directory c
004123	ERROR	fsync(%s	**Cause**: Writing an on-disk version data object failed while attempting intermediate directory {directory}. error. **Action**: Check log for the error inf and determine why the directory "
004124	ERROR	close(%s	**Cause**: Writing an on-disk version data object failed while attempting intermediate directory {directory}. error. **Action**: Check log for the error inf and determine why the directory c
004125	ERROR	wrote only %d bytes of requested %d\n	**Cause**: Writing an on-disk version data object failed as the final size written data is less than the reque

Code	Severity	Message	Cause/Action
			{number} of bytes. **Action**: Check log for the related in detail and determine why the da failed.
004126	ERROR	open(%s	**Cause**: Attempting to open a data the reading of an on-disk version into the buffer. This is a system er **Action**: Check log for the error inf and determine why the file open is
004127	ERROR	open(%s	**Cause**: Attempting to open a temp failed during the reading of an on- the data object into the buffer. Thi error. **Action**: Check log for the error inf and determine why the file open is
004128	ERROR	read(%s	**Cause**: Reading a data file failed of an on-disk version of the data o buffer. This is a system error. **Action**: Check log for the error inf and determine why the file reading
004129	ERROR	read buffer overflow (MAX=%d)\n	**Cause**: The read buffer limit {max while attempting to read an on-dis data object into the buffer. **Action**: Check the LifeKeeper cor restart the LifeKeeper.
004130	ERROR	close(%s	**Cause**: Failed to close a data file on-disk version of the data object

Code	Severity	Message	Cause/Action
			This is a system error. **Action**: Check log for the error inf and determine why the file close i
004131	ERROR	rename(%s, %s	**Cause**: Failed to rename a tempo during reading an on-disk version into the buffer. This is a system er **Action**: Check log for the error inf and determine why the file renami
004132	ERROR	Can’t open %s : %s	**Cause**: Failed to open a directory error {error} during reading an on- the application and resource type the buffer. This is a system error. **Action**: Check log for the error inf and determine why the open direc
004133	ERROR	path argument may not be NULL	**Cause**: The command "lcdrcp" fai copy because the input source pa **Action**: Check the input source pa "lcdrcp".
004134	ERROR	destination path argument may not be NULL	**Cause**: The "lcdrcp" command fai copy because the input destinatio **Action**: Check the input destinatio "lcdrcp".
004135	ERROR	destination path can’t be zero length string	**Cause**: Input destination path was copy when using "lcdrcp". **Action**: Check the input destinatio "lcdrcp".

Code	Severity	Message	Cause/Action
004136	ERROR	open(%s	**Cause**: Failed to open source file copy using "lcdrcp". This is a syst **Action**: Check the existence/avail source path and retry "lcdrcp". Als related log for error information in
004137	ERROR	fstat(%s	**Cause**: Failed to fetch file attribut during file copy by "lcdrcp". This is **Action**: Check the log for error inf
004138	ERROR	file \"%s\" is not an ordinary file (mode=0%o	**Cause**: Detected source file as a during file copy using "lcdrcp". **Action**: Check the input source fil "lcdrcp".
004151	FATAL	lcdMalloc failure	**Cause**: Failed to allocate memory size in shared memory. **Action**: A fatal error will be produ
004152	ERROR	having \"%s\" depend on \"%s\" would produce a loop	**Cause**: Adding the requested dep produce a loop of dependent relat **Action**: Correct the requested dep retry the dependency creation.
004164	ERROR	Priority mismatch between resources %s and %s. Dependency creation failed.	**Cause**: The priorities for {resource {resource2} do not match. **Action**: Resource priorities must one or both priorities to the same creating the dependency.

Code	Severity	Message	Cause/Action
004176	ERROR	%s	**Cause**: The command "doabort" f {directory} for writing the core file. error. **Action**: Check the logs for related resolve the reported problem.
004182	ERROR	received signal %d\n	**Cause**: Received signal {signum}.
004186	ERROR	%s: ::receive(%d) protocol error on incoming_mailbox %s	**Cause**: In function {function}, atte message within timeout {timeout} due to a non-idle status of incomi {mailbox}. **Action**: Check the status of the co the cluster and retry the process.
004190	ERROR	%s: ::receive(%d) did not receive message within %d seconds on incoming_mailbox %s	**Cause**: In function {function}, atte message within timeout {timeout} with incoming mailbox {mailbox}. **Action**: Check the status of the co the cluster and retry the process.
004204	ERROR	attempt to send illegal message	**Cause**: Sending message failed d message.
004205	ERROR	destination system \"%s\" is unknown	**Cause**: Sending message failed d destination system name {system} **Action**: Check the configuration a system and check the logs for rela the same process after the system
004206	ERROR	destination mailbox \"%s\" at system \"%s\" is unknown	**Cause**: Sending message failed d mailbox {mailbox} on destination s

Code	Severity	Message	Cause/Action
			{system}. This error may be cause message before the LCD is fully in **Action**: Check the configuration a system and check the logs for rela the same process after the system
004208	ERROR	destination system \"%s\" is alive but the \"%s\" mailbox process is not listening.	**Cause**: Sending message failed. T connection to destination system but the contact to the destination **Action**: Check the configuration a system and check the logs for rela the same process after the system
004209	ERROR	destination system \"%s\" is dead.	**Cause**: Sending message failed d connection with the destination sy **Action**: Check the configuration a system and check the logs for rela the same process after the system
004211	ERROR	can’t send to destination \"%s\" error=%d	**Cause**: Sending message to desti {system} failed due to internal erro **Action**: Check the logs for related resolve the reported problem.
004217	ERROR	destination system \"%s\" is out of service.	**Cause**: Sending message failed d connection with the destination sy **Action**: Check the configuration a system and check the logs for rela the same process after the system
004221	ERROR	destination system \"%s\" went out of service.	**Cause**: Sending message failed d connection with the destination sy

Code	Severity	Message	Cause/Action
			Action: Check the configuration a system and check the logs for rela the same process after the system
004228	ERROR	Can’t get host name from getaddrinfo(	**Cause**: Creating network object fa failure when getting host name us "getaddrinfo()". **Action**: Check the configuration a system. Do the same process aga
004234	ERROR	IP address pair %s already in use	**Cause**: Creating network object fa address pair {pair} is already used communication path. **Action**: Check the input IP addres network creation again.
004258	WARN	Communication to %s by %s FAILED	**Cause**: Communication to system communication path {path} failed. **Action**: Check system configurati connection.
004261	WARN	COMMUNICATIONS failover from system \"%s\" will be started.	**Cause**: A failover from system {sy started due to all the communicati down. **Action**: Check system configurati connection status. Confirm the sys failover is done.
004292	ERROR	resource \"%s\" %s	**Cause**: A resource could not be b because its current state is unkno **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
004293	ERROR	resource \"%s\" %s	**Cause**: A resource could not be b service because its current state c **Action**: Check the logs for related resolve the reported problem.
004294	ERROR	resource \"%s\" requires a license (for Kit %s/%s) but none is installed	**Cause**: The resource's related rec a license. **Action**: Install a license for the re server where the resource was to service.
004297	ERROR	secondary remote resource \"%s\" on machine \"%s\" is already in-service, so resource \"%s\" on machine \"%s\" can't be brought in-service.	**Cause**: A resource {resource} cou into service on machine {machine} secondary remote resource {resou service on machine {machine}. **Action**: Manually change the rem of service and do in service on loc
004298	ERROR	remote resource \"%s\" on machine \"%s\" is still in-service, restore of resource \"%s\" will not be attempted!\n	
004300	ERROR	restore of resource \"%s\" has failed	**Cause**: A resource could not be b service. **Action**: Check the logs for related resolve the reported problem.
004311	ERROR	can't perform \"remove\" action on resources in state \"%s\"	**Cause**: A resource could not be p due to the current state being {sta **Action**: Check the logs for related resolve the reported problem.
004313	ERROR	remove of resource \"%s\" has failed	**Cause**: A resource {resource} cou of service.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
004318	ERROR	%s,priv_globact(%d,%s): script %s FAILED returning %d	**Cause**: A global action script faile specified error code. **Action**: Check the logs for related resolve the reported problem.
004332	ERROR	action \"%s\" has failed on resource \"%s\"	**Cause**: A resource action failed. **Action**: Check the logs for related resolve the reported problem.
004351	ERROR	a \"%s\" equivalency must have one remote resource	**Cause**: Creating of a {eqvtype} eq due to the two input tag names ex system. **Action**: Correct the inputs resourc do the same process again.
004356	WARN	Use unsupported option %s for remove. This option may be removed in future upgrades.	
004376	FATAL	wait period of %u seconds for LCM to become available has been exceeded (lock file \"%s\" not removed	**Cause**: The LCM daemon did not within a reasonable time and the L operate without the LCM. **Action**: Check the logs for related resolve the reported problem.
004386	ERROR	initlcdMalloc;shmget	**Cause**: A shared memory segmer initialized. **Action**: Check the logs for related resolve the reported problem. Rev documentation and ensure that th the minimum requirements and th

Code	Severity	Message	Cause/Action
			system is configured properly.
004444	WARN	License key (for Kit %s/%s) has EXPIRED	**Cause**: Your license has expired. **Action**: Contact Support to obtain
004445	WARN	License key (for Kit %s/%s) will expire at midnight in %ld days	**Cause**: Your license is about to ex **Action**: Contact Support to obtain
004466	ERROR	system \"%s\" not defined on machine \"%s\".	**Cause**: The specified system nam **Action**: Verify the system name, a operation again.
004467	ERROR	system \"%s\" unknown on machine \"%s\"	**Cause**: The specified system nam recognized. **Action**: Verify the system name, a operation again.
004494	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004495	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004496	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
004497	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004498	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004499	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004500	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004501	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004502	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
004503	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004504	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004505	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004506	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004507	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004508	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
004509	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004510	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004511	ERROR	COMMAND OUTPUT: %s	**Cause**: An action or event script p unexpected output. **Action**: Check the logs for related resolve the reported problem.
004512	ERROR		**Cause**: An error occured on the re **Action**: Check the logs on the rem additional details.
004565	ERROR	can’t set resource state type to ILLSTATE	**Cause**: An attempt was made to p an illegal state. **Action**: Do not try to put a resourc state.
004567	ERROR	split brain detected while setting resource \"%s\" to \"%s\" state (SHARED equivalency to resource \"%s\" on machine \"%s\" which is in state \"%s\"). Setting local resource ISP but aborting the operation.	**Cause**: Changing resource {resou {state} failed since its SHARED eq {resource} on machine {machine}

Code	Severity	Message	Cause/Action
			Action: A split brain situation has failover operation has been aborte manually put the split brain resour multiple systems) in the proper sta
004575	ERROR	COMMAND OUTPUT: %s	
004607	ERROR	no resource instance has tag \"%s\"	**Cause**: No resource with the prov **Action**: Provide a valid tag, or che related errors and try to resolve th problem.
004608	ERROR	no resource instance has identifier \"%s\"	**Cause**: No resource exists with th identifier. **Action**: Provide a valid identifier, for related errors and try to resolv problem.
004619	ERROR	resource with tag \"%s\" already exists with identifier \"%s\"	**Cause**: The provided tag name al **Action**: Choose a different tag na
004620	ERROR	resource with identifier \"%s\" already exists with tag \"%s\"	**Cause**: The provided identifier alr **Action**: Choose a different identifi resource.
004643	ERROR	Instance tag name is too long. It must be shorter than %d characters.	**Cause**: Tag name is too long. **Action**: Provide a tag name that is characters.
004646	ERROR	Tag name contains illegal characters	**Cause**: Tag name contains an ille

Code	Severity	Message	Cause/Action
			Action: Specify a tag name that d one of these characters: _-./
004691	ERROR	can’t set both tag and identifier at same time	**Cause**: Both a tag and an identifie **Action**: Provide only one of tag or
004745	ERROR	failed to access lkexterrlog path=%s	**Cause**: The utility "lkexterrlog" ca for collecting system information. **Action**: Check the installation of p lk" and make sure the utility "lkext accessible.
004746	ERROR	lkexterrlog failed runret=%d cmdline=%s	**Cause**: The execution of utility "lk when collecting system informatio **Action**: Check the logs for related resolve the reported problem.
004782	ERROR	Resource \"%s\" was in state \"%s\" before event occurred – recovery will not be attempted	**Cause**: The resource is already in Recovery will not be attempted.
004783	ERROR	Resource \"%s\" was already in state \"%s\" before event occurred	**Cause**: A resource was not in an to allow recovery. **Action**: Put the resource in the IS is still needed.
004786	ERROR	%s on failing resource \"%s\"	**Cause**: An error occured why atte a resource. **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
004788	EMERG	failed to remove resource ‘%s’. SYSTEM HALTED.	**Cause**: A error occur that prevent from being taken out of service du The system has been restarted to resource is not active on two syste **Action**: Check the logs for related resolve the reported problem.
004793	ERROR	lcdsendremote transfer resource \"%s\" to \"%s\" on machine \"%s\" failed (rt=%d	**Cause**: A failure occured while tra resource and it’s dependencies to **Action**: Check the logs for related resolve the reported problem. Che other system for related errors.
004797	ERROR	Restore of SHARED resource \"%s\" has failed	**Cause**: There was an error while resource. **Action**: Check the logs for related resolve the reported problem.
004806	ERROR	Restore in parallel of resource \"%s\" has failed; will re-try serially	**Cause**: Parallel recovery failed. C related errors and try to resolve th problem. **Action**: No action is required. The continue to recover serially. If the check for error messages related which failed to recover to find out actions to take.
004819	ERROR	read_temporal_recovery_log(): failed to fopen file: %s. fopen() %s.	**Cause**: The opening of the tempo file {file} in preparation for loading memory failed with the error {erro **Action**: Check system log files an reported errors before retying the

Code	Severity	Message	Cause/Action
004820	ERROR	read_temporal_recovery_log(): failed to malloc initial buf for temporal_recovery_stamp.	**Cause**: Loading the temporal reco information into memory failed wh acquire memory to store the log in **Action**: Check system log files an reported errors before retying the
004821	ERROR	read_temporal_recovery_log(): failed to reallocate buffer for temporal_recovery_stamp.	**Cause**: Loading the temporal reco information into memory failed wh increase the amount of memory re the log information. **Action**: Check system log files an reported errors before retrying the
004822	ERROR	write_temporal_recovery_log(): failed to open file: %s.	**Cause**: The update of the tempora was terminated when the open of {temporary name} failed. **Action**: Check system log files an reported errors before retrying the
004823	ERROR	rename(%s, %s) failed.	**Cause**: The update of the tempora was terminated when the rename file {temporary name} to the real l failed. **Action**: Check system log files an reported errors before retrying the
004827	ERROR	%s: [%d,%s] ERROR IN LOCK FUNCTIONS	
004829	FATAL	err=%s line=%d Semid=%d numops=%zd perror=%s	**Cause**: The modification of semap {semaphore} failed with error {err} message description {perror}. **Action**: Check adjacent log messa details. Also, check the system log any reported errors before retrying

Code	Severity	Message	Cause/Action
004860	ERROR	restore ftok failed for resource %s with path %s	**Cause**: The attempt to generate a in semaphore operations for resou path {path} failed. This is a system **Action**: Check adjacent log messa details. Also check system log file reported errors before retrying the
004861	ERROR	semget failed with error %d	**Cause**: The attempt to retrieve the identification associated with the i failed. This is a system error. **Action**: Check adjacent log messa details. Also check system log file reported errors before retying the
004862	ERROR	semctl SEMSET failed with error %d	**Cause**: The attempt to create and semaphore used during the recov failed with the error {error number error. **Action**: Check adjacent log messa details. Also, check system log file reported errors before retying the
004863	ERROR	semop failed with error %d	**Cause**: The attempt to set a sema during the recovery process has fa {error number}. This is a system e **Action**: Check adjacent log messa details. Also, check system log file reported errors before retying the
004864	ERROR	semctl SEMSET failed with error %d	**Cause**: The attempt to release a s during the recovery process has fa {error number}. This is a system e **Action**: Check adjacent log messa

Code	Severity	Message	Cause/Action
			details. Also, check system log file reported errors before retying the
004865	ERROR	restore action failed for resource %s (exit: %d	**Cause**: The attempt to bring resou Service has failed. **Action**: Check adjacent log messa details. Correct any reported error operation.
004872	ERROR	Remote remove of resource \"%s\" on machine \"%s\" failed (rt=%d	**Cause**: The request to take resou Service on {server} for transfer to has failed. **Action**: Check adjacent log messa details on the local system. Also, messages on {server} for further d failure to remove the resource.
004875	ERROR	remote remove of resource \"%s\" on machine \"%s\" failed	**Cause**: The request to take resou Service on {server} for transfer to has failed. **Action**: Check adjacent log messa details on the local system. Also, messages on {server} for further d failure to remove the resource.
004876	ERROR	remote remove of resource \"%s\" on machine \"%s\" failed	**Cause**: The request to take resou Service on {server} for transfer to has failed. **Action**: Check adjacent log messa details on the local system. Also, messages on {server} for further d failure to remove the resource.

Code	Severity	Message	Cause/Action
005045	ERROR	tli_fdget_i::execute unable to establish a listener port	**Cause**: A network connection cou configured. **Action**: Verify that all network har are properly configured. If this me and resources cannot be put into Support.
005055	ERROR	tli_fdget_o::execute – async connect failure	**Cause**: A network connection cou configured. **Action**: Verify that all network har are properly configured. If this me and resources cannot be put into Support.
005061	ERROR	tli_fdget_o::execute – bind socket	**Cause**: A network connection cou configured. **Action**: Verify that all network har are properly configured. If this me and resources cannot be put into Support.
005090	WARN	system_driver::add_driver: cmd=%s\n	
005108	WARN	system_driver::rm_driver: cmd=%s\n	
005145	ERROR	opening the file	**Cause**: A pipe could not be opene **Action**: Check adjacent log messa details.
005164	ERROR	tli_handler::handle-error:sending/receiving data message	**Cause**: A message failed to be se **Action**: Check adjacent log messa details. This may be a temporary error continues and servers can't verify the network configuration or

Code	Severity	Message	Cause/Action
005165	WARN	errno %d\n	**Cause**: A message failed to be se **Action**: Check adjacent log messa details. This may be a temporary error continues and servers can't verify the network configuration or
005166	WARN	poll 0x%hx\n	**Cause**: A message failed to be se **Action**: Check adjacent log messa details. This may be a temporary error continues and servers can't verify the network configuration or
005167	WARN	handler for sys %s\n	**Cause**: A message failed to be se **Action**: Check adjacent log messa details. This may be a temporary error continues and servers can't verify the network configuration or
005225	WARN	so_driver::handle_error: sending/receiving data message errno %d: %s	**Cause**: A message failed to be se **Action**: Check adjacent log messa details. This may be a temporary error continues and servers canno verify the network configuration or
005235	WARN	found tcp connection to iwstp\n	
005236	WARN	didn't find tcp connection to iwstp\n	
005237	WARN	lcm_handler retry send from %s:%s to %s:%s (%d)\n	
005238	WARN	detected duplicate request from %s:%s to %s:%s\n	
005239	WARN	lcm_handler retry timer set to %d based on lcd remote timeout of %d s (%d s	
005240	WARN	lcm_handler retry count/time (%d/%zu) has exceeded the maximum. Giving up…\n	
005241	WARN	dup_list: %s %s %s %s %d %d (%d	
005242	WARN	clean up stale dup_list entry (%d s): %s:%s to %s:%s last: %s	

Code	Severity	Message	Cause/Action
005243	WARN	add new dup_list entry %s:%s to %s:%s %d %d (%d	
005244	WARN	closing fd %d\n	
005245	WARN	openpoll fd %d\n	
006012	ERROR	quickCheck script '%s' (%d) failed to exit after %lu seconds. Forcibly terminated. Please examine the script or adjust the LKCHECKINTERVAL parameter in %s.	**Cause**: A quickCheck script is pro long or hanging. **Action**: Perform the steps listed i text.
006014	ERROR	LKCHECKINTERVAL parameter is too short. It is currently set to %ld seconds. It should be at least %ld seconds. Please adjust this parameter in %s and execute 'kill %d' to restart the lkcheck daemon.	
006102	ERROR	COMMAND OUTPUT: $LKROOT/bin/sendevent	**Cause**: This is output from a "sen generator) command. **Action**: Check adjacent log messa details.
006103	ERROR	COMMAND OUTPUT: $LKROOT/bin/sendevent	**Cause**: This is output from a "sen generator) command. **Action**: Check adjacent log messa details.
006104	ERROR	COMMAND OUTPUT: $LKROOT/bin/sendevent	**Cause**: This is output from a "sen generator) command. **Action**: Check adjacent log messa details.
006502	ERROR	CPU usage has exceeded the threshold ($threshold%) for $count check cycles.	
006504	ERROR	Could not open /proc/meminfo	
006505	ERROR	Memory usage has exceeded the threshold ($threshold%) for $count check cycles.	
006508	ERROR	[$SUBJECT event] mail returned $err	
006509	ERROR	COMMAND OUTPUT: cat /tmp/err$$	

Code	Severity	Message	Cause/Action
006511	ERROR	snmptrap returned $err for Trap 190	
006512	ERROR	COMMAND OUTPUT: cat /tmp/err$$	
006514	ERROR	[$SUBJECT event] mail returned $err	
006515	ERROR	COMMAND OUTPUT: cat /tmp/err$$	
006517	ERROR	snmptrap returned $err for Trap 200	
006518	ERROR	COMMAND OUTPUT: cat /tmp/err$$	
006520	ERROR	Failed to update error count in $cpu_file: $!	
006521	ERROR	Failed to update error count in $mem_file: $!	
006523	ERROR	The SNHC_CPUCHECK_THRESHOLD setting is not valid. Please set SNHC_CPUCHECK_THRESHOLD to a value between 10 and 99 in /etc/default/LifeKeeper. If not set, the value will default to 99.	
006524	ERROR	The SNHC_CPUCHECK_TIME setting is not valid. Please set SNHC_CPUCHECK_TIME to a value between 1 and 100 in /etc/default/LifeKeeper. If not set, the value will default to 1.	
006525	ERROR	The SNHC_MEMCHECK_THRESHOLD setting is not valid. Please set SNHC_MEMCHECK_THRESHOLD to a value between 10 and 99 in /etc/default/LifeKeeper. If not set, the value will default to 99.	
006526	ERROR	The SNHC_MEMCHECK_TIME setting is not valid. Please set SNHC_MEMCHECK_TIME to a value between 1 and 100 in /etc/default/LifeKeeper. If not set, the value will default to 1.	
006528	ERROR	Could not open /proc/stat	
006529	ERROR	Could not open /proc/stat	
006530	ERROR	Could not use $tmp_path	
007053	ERROR	malloc failed. Assume that it is a monitoring target device.	
007058	ERROR	%s: %s failed on ‘%s’, result:%d, Sense Key = %d.	**Cause**: A SCSI device couldn’t be its status checked. This may be b storage is malfunctioning or becau been reserved by another server. **Action**: Check adjacent log messa details and verify that resources a properly.
007059	ERROR	%s: %s failed on ‘%s’, result:%d.	**Cause**: A SCSI device couldn’t be its status checked. This may be b storage is malfunctioning or becau

Code	Severity	Message	Cause/Action
			been reserved by another server. **Action**: Check adjacent log messa details and verify that resources a properly.
007060	EMERG	%s: failure on device '%s'. SYSTEM HALTED.	**Cause**: A SCSI device couldn't be its status checked. This may be b storage is malfunctioning or becau been reserved by another server. WILL BE REBOOTED/HALTED. **Action**: Verify that the storage is properly and, if so, that resources properly and have been put in ser server.
007072	ERROR	%s: failed to open SCSI device '%s', initiate recovery. errno=0x%x, retry count=%d.	**Cause**: The protected SCSI devic opened. The device may be failing been removed from the system. **Action**: The system will be halted the backup node will be initiated. in this case is a failover, but this c with the SCSIERROR tunable.
007073	ERROR	%s: failed to open SCSI device '%s', RETRY. errno=%d, retry count=%d.	**Cause**: The protected SCSI devic opened. The device may be failing been removed from the system. **Action**: This error is not critical. T be retried in 5 seconds. If the prob system will perform a halt or resou
007075	ERROR	%s: RESERVATION CONFLICT on SCSI device '%s'. ret=%d, errno=0x%x, retry count=%d.	**Cause**: A SCSI device couldn't be a conflict with another server. This the storage is malfunctioning or be has been reserved by another ser

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messa details and verify that resources a properly.
007077	ERROR	%s: DEVICE FAILURE on SCSI device '%s', initiate recovery. ret=%d, errno=0x%x, retry count=%d.	**Cause**: A SCSI device couldn't be its status checked. This may be b storage is malfunctioning or becau been reserved by another server. **Action**: Check adjacent log messa details and verify that resources a properly.
007078	ERROR	%s: DEVICE FAILURE on SCSI device '%s', RETRY. ret=%d, errno=0x%x, retry count=%d.	**Cause**: A SCSI device couldn't be its status checked. This may be b storage is malfunctioning or becau been reserved by another server. **Action**: Check adjacent log messa details and verify that resources a properly.
010002	WARN	flag $flag not present, send message again.	**Cause**: This message indicates a process that will be retried. **Action**: Check adjacent log messa errors.
010003	ERROR	COMMAND OUTPUT: $LKBIN/ins_remove	**Cause**: This message is part of th "ins_remove" command. **Action**: Check adjacent log messa details. This may not be a true err
010006	WARN	flg_list -d $i took more than $pswait seconds to complete…	**Cause**: A flag list operation on a s longer than expected. There may communicating with the other serv

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messa details.
010007	ERROR	flag $flag not present, switchovers may occur.	**Cause**: One of the servers in the c be told to disallow failover operati current server. **Action**: Check adjacent log messa details and monitor the cluster for behavior.
010008	WARN	flag $flag not present, send message again.	**Cause**: A process is incomplete b **Action**: Check adjacent log messa details and for repeated warnings/
010023	FATAL	LifeKeeper failed to initialize properly.	**Cause**: There was a fatal error wh start LifeKeeper. **Action**: Check adjacent log messa details.
010025	ERROR	`printf 'Unable to get a unique tag name on server "%s" for template resource "%s"' $MACH $DISK`	**Cause**: A suitable tag during the c a storage resource could not be a generated. **Action**: Check adjacent log messa details. Retry the operation if there errors.
010034	FATAL	Unable to start lcm.	**Cause**: A core component of the s be started. **Action**: Check adjacent log messa details and try to resolve the repo

Code	Severity	Message	Cause/Action
010038	WARN	Waiting for LifeKeeper core components to initialize has exceeded 10 seconds. Continuing anyway, check logs for further details.	**Cause**: Some parts of the softwar longer than expected to start up. **Action**: Perform the steps listed ir text.
010039	WARN	Waiting for LifeKeeper core components to initialize has exceeded 10 seconds. Continuing anyway, check logs for further details.	**Cause**: Some parts of the softwar longer than expected to start up. **Action**: Perform the steps listed ir text.
010046	ERROR	The dependency creation failed on server $SERVER:" `cat $TEMP_FILE`	**Cause**: A dependency relationship created on the given server. **Action**: Check the logs for related resolve the reported problem
010063	ERROR	$REMSH error	**Cause**: A command to request da another server failed. **Action**: Check the logs for related resolve the reported problem.
010085	ERROR	lkswitchback($MACH): Automatic switchback of \"$loctag\" failed	**Cause**: The resource was not swi as expected. **Action**: Check the logs for related resolve the reported problem.
010102	ERROR	admin machine not specified	**Cause**: Invalid parameters were s "getlocks" operation. **Action**: Verify the parameters and operation. If this error happens du operation, contact Support.

Code	Severity	Message	Cause/Action
010107	WARN	Lock for $m is ignored because system is OOS	**Cause**: A lock was ignored becau which the lock was created is not **Action**: Check the logs for related be a harmless error.
010108	ERROR	lock acquisition timeout	**Cause**: Acquiring a lock tool longe allowed. **Action**: Check the logs for related resolve the reported problem.
010109	ERROR	could not get admin locks." `cat /tmp/ER$$`	**Cause**: The software failed to acq required to manage resources. **Action**: Check the logs for related resolve the reported problem.
010112	ERROR	lcdrcp failed with error no: $LCDRCPRES	**Cause**: A file could not be copied **Action**: Check the logs for related resolve the reported problem.
010116	ERROR	unable to set !lkstop flag	**Cause**: A flag could not be set to server is being stopped by user re **Action**: Check the logs for related resolve the reported problem.
010121	ERROR	Extended logs aborted due to a failure in opening $destination. ($syserrmsg	**Cause**: The execution of utility "lk when opening the extended log fil **Action**: Check the logs for related resolve the reported problem.
010132	ERROR	Unable to retrieve reservation id from "%s". Error: "%s". Attempting to regenerate.	**Cause**: Unable to open the file /op

Code	Severity	Message	Cause/Action
			config/.reservation_id to retrieve t for SCSI 3 persistent reservations **Action**: None. An attempt will be regenerate the ID and update the
010135	ERROR	The current reservation ID of "%s" is not unique within the cluster. A new ID must be generated by running "%s/bin/genresid -g" on "%s".	**Cause**: The reservation id defined not unique within the cluster and **Action**: Take all resources out of node and then run "/opt/LifeKeepe to generate a unique reservation i
010136	ERROR	Unable to store reservation id in "%s". Error: "%s"	**Cause**: Unable to open the file /op config.reservation_id to store the SCSI 3 persistent reservations. **Action**: Correct the error listed as open failed and then take all resou service on this node and then run bin/genresid -g" to generate a new reservation id.
010137	ERROR	Failed to generate a reservation ID that is unique within the cluster.	**Cause**: The generated reservation defined on another node in the clu unique within the cluster. **Action**: Take all resources out of node and then run "/opt/LifeKeepe to generate a new unique reserva
010138	ERROR	$message	
010139	WARN	$message	
010140	ERROR	$COMMAND_SNMPTRAP returned $exitcode for Trap $oid:$result	
010141	ERROR	LK_TRAP_MGR is specified in /etc/default/LifeKeeper, but $COMMAND_SNMPTRAP command is not in PATH.	
010142	ERROR	$COMMAND_EMAIL returned $exitcode:$result	
010143	ERROR	LK_NOTIFY_ALIAS is specified in /etc/default/LifeKeeper,	

Code	Severity	Message	Cause/Action
		but $COMMAND_EMAIL command is not in PATH.	
010144	ERROR	can't opendir $LICENSE_DIR: $!	
010145	ERROR	lktest failed	
010146	ERROR	lkcheck failed	
010147	ERROR	ins_list failed: exit code = $exit_code	
010159	ERROR	Maintenance mode disable currently in progress, can't enable maintenance mode. If this problem persists, consider using the —force option.	
010160	ERROR	Maintenance mode enable currently in progress, can't disable maintenance mode. If this problem persists, consider using the —force option.	
010161	ERROR	$tag is not a valid resource tag on the local machine, aborting. Please check the spelling and try again.	
010163	ERROR	$cmd script not found or not executable on system $sys.	
010165	ERROR	An error occurred while running \'$LKROOT/lkadm/subsys/appsuite/sap/bin/$cmd -m=${opt_mode}${tag_cmd}${force_cmd}\' on system $sys (exit code: $remexec_ret). Please inspect the logs on that system for more information.	
010168	WARN	Maintenance mode was not fully ${opt_mode}d for at least one resource on system $sys.	
010172	ERROR	Maintenance mode was not fully ${opt_mode}d for the requested resources on at least one system in the cluster.	
010173	ERROR	LifeKeeper is not running on system $me. Unable to ${opt_mode} maintenance mode. Aborting.	
010179	ERROR	Maintenance mode action \'${opt_mode}\' did not complete successfully for resource $tag on system $me.	
010181	ERROR	An error occurred while attempting maintenance mode action \'${opt_mode}\' on system $me for resources: @{local_hier_tags}.	
010187	ERROR	Resource $tag has not been extended to system $sys.	
010222	ERROR	scsifree(%s): LKSCSI_Release(%s) unsuccessful	**Cause**: A SCSI device that appea reserved was not released as exp **Action**: Check the logs for related resolve any reported problems. Th benign if the system is functioning
010231	ERROR	scsiplock(%s): reserve failed.	**Cause**: A reservation on a SCSI d be acquired.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
010250	ERROR	Failed to exec command ‘%s’	**Cause**: The "lklogmsg" tool failed command {command}. **Action**: Check the logs for related resolve any reported problems. Ve command exists and is a valid cor program. If this message happens operation, contact Support.
010402	EMERG	local recovery failure on resource $opts{‘N’}, trigger VMware HA…	**Cause**: When in LifeKeeper Single Protection operation, a resource c recovered and VMware-HA is abo to handle the failure (if VMware-H **Action**: No action is required. VM handle the failure.
010413	ERROR	COMMAND OUTPUT: cat /tmp/err$$	**Cause**: This is the output from an command that may have failed. **Action**: Check the logs for related resolve the reported problem.
010420	EMERG	local recovery failure on resource $opts{‘N’}, trigger reboot…	**Cause**: When in LifeKeeper Single Protection operation, a resource c recovered and a reboot is about to handle the failure. **Action**: No action is required.
010440	ERROR	[$SUBJECT event] mail returned $err	**Cause**: This indicates a notificatio be sent via the "mail" command. **Action**: Check the logs for related

Code	Severity	Message	Cause/Action
			resolve the reported problem.
010443	ERROR	COMMAND OUTPUT: cat /tmp/err$$	**Cause**: This is the output from a " that may have failed. **Action**: Check the logs for related resolve the reported problem.
010445	ERROR	COMMAND OUTPUT: cat /tmp/err$$	**Cause**: This is the output from a " that may have failed. **Action**: Check the logs for related resolve the reported problem.
010463	ERROR	LifeKeeper: name of machine is not specified, ARGS=$ARGS	**Cause**: Invalid arguments were sp "comm_down" event. **Action**: Check your LifeKeeper co retry the operation.
010471	ERROR	COMM_DOWN: Attempt to obtain local comm_down lock flag failed	**Cause**: During the handling of a c failure with another node, a local l acquired. This will likely stop a fai proceeding. **Action**: Check the logs for related resolve the reported problem. If fa taking place properly, contact Sup
010482	ERROR	LifeKeeper: name of machine is not specified, ARGS=$ARGS	**Cause**: Invalid arguments were sp "comm_up" event. **Action**: Check your LifeKeeper co retry the operation.
010484	WARN	flg_list -d $MACH check timed-out ($delay seconds).	**Cause**: "flg_list" command reache

Code	Severity	Message	Cause/Action
			value {delay} seconds.
010487	WARN	flg_list -d $MACH check timed-out, unintended switchovers may occur.	**Cause**: "flg_list" command reache value. **Action**: Switch back the resource switchover occurs.
010492	WARN	$m	**Cause**: One of other servers looke server {server}, but witness server **Action**: Ensure other server is de over the resource manually.
010494	ERROR	LifeKeeper: COMM_UP to machine $MACH completed with errors.	**Cause**: An unexpected failure occ "COMM_UP" event. **Action**: Check adjacent log messa details.
010503	ERROR	lcdrecover hung or returned error, attempting kill of process $FPID	**Cause**: "lcdrecover" took too long
010506	ERROR	Intelligent Switchback Check Failed	**Cause**: Failed 5 times to perform **Action**: Switch over the resource
010535	ERROR	LifeKeeper: name of machine is not specified, ARGS=$ARGS	
010600	ERROR	removing hierarchy remnants	
010627	WARN	Equivalency Trim: does not have a full complement of equivalencies. Hierarchy will be unextended from	
010629	WARN	Your hierarchy exists on only one server. Your application has no protection until you extend it to at least one other server.	
010712	ERROR	Unextend hierarchy failed	**Cause**: A resource hierarchy faile

Code	Severity	Message	Cause/Action
			unextended from a server. **Action**: Check the logs for related resolve the reported problem.
010746	ERROR	$ERRMSG Target machine \"$TARGET_MACH\" does not have an active LifeKeeper communication path to machine \"$aMach\" in the hierarchy." >&2	**Cause**: A hierarchy cannot be une the target server does not have ac communication with the other serv **Action**: Check the logs for related resolve the reported problem. Ens servers have communication path
010763	ERROR	lock failed	
011000	ERROR	appremote: unknown command type %d('%c')\n	**Cause**: Internal error. **Action**: Try restarting the product
011001	ERROR	depremote: unknown command type %d('%c')\n	**Cause**: Internal error. **Action**: Try restarting the product
011002	ERROR	eqvremote: unknown command type %d('%c')\n	**Cause**: Internal error. **Action**: Try restarting the product
011003	ERROR	flgremote: unknown command type %d('%c')\n	**Cause**: Internal error. **Action**: Try restarting the product
011004	WARN	Illegal creation of resource	**Cause**: This will not occur under r circumstances.
011009	ERROR	insremote: unknown change field command type %d('%c')\n	**Cause**: Internal error.

Code	Severity	Message	Cause/Action
			Action: Try restarting the product
011010	ERROR	insremote: unknown command type %d(‘%c’)\n	**Cause**: Internal error. **Action**: Try restarting the product
011011	FATAL	%s	**Cause**: LifeKeeper could not get l **Action**: Check adjacent log messa details.
011012	FATAL	semget(%s,%c	**Cause**: LifeKeeper could not get s **Action**: Check adjacent log messa details.
011013	FATAL	shmget(%s,%c	**Cause**: System could not allocate segment. **Action**: Check adjacent log messa details.
011014	FATAL	prefix_lkroot("out"	**Cause**: A system error has occurr accessing /opt/LifeKeeper/out. **Action**: Determine why /opt/LifeK accessible.
011015	ERROR	DEMO_UPGRADE_MSG	**Cause**: You are running a demo li **Action**: Contact Support to obtain
011016	ERROR	lic_single_node_msg	**Cause**: You have a license for Life Server Protection but you do not h Single Server Protection installed.

Code	Severity	Message	Cause/Action
			Action: Either install LifeKeeper S Protection or obtain a license that product you are running.
011018	ERROR	lic_init_fail_msg, lc_errstring(lm_job	**Cause**: License manager initializa **Action**: Check adjacent log messa details.
011020	EMERG	lic_init_fail_msg, lc_errstring(lm_job	**Cause**: License manager initializa **Action**: Check adjacent log messa details.
011021	EMERG	lic_error_msg, lc_errstring(lm_job	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011022	EMERG	lic_error_msg, lc_errstring(lm_job	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011023	EMERG	lic_no_rest_suite, ""	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011024	EMERG	lic_error_msg, lc_errstring(lm_job	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011025	EMERG	lic_no_license, ""	**Cause**: LifeKeeper could not find **Action**: Ensure license keys are v and retry the operation.

Code	Severity	Message	Cause/Action
011026	EMERG	lic_error_msg, lc_errstring(lm_job	**Cause**: There is an unknown prob license. **Action**: Contact Support to obtain
011027	EMERG	lic_no_license, ""	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011028	ERROR	lang_error_msg	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011029	FATAL	can’t set reply system	**Cause**: A message failed to be se **Action**: Check adjacent log messa details. This may be a temporary
011030	FATAL	can’t set reply mailbox	**Cause**: A message failed to be se **Action**: Check adjacent log messa details. This may be a temporary
011031	ERROR	Failure reading output of ‘%s’ on behalf of %s	**Cause**: A system error has occurr accessing temporary file /tmp/OU **Action**: Determine why /tmp/OUT accessible.
011032	ERROR	Failure reading output of ‘%s’	**Cause**: A system error has occurr accessing temporary file /tmp/ERR **Action**: Determine why /tmp/ERR accessible.

Code	Severity	Message	Cause/Action
011033	ERROR	event \"%s,%s\" already posted for resource with id \"%s\"	**Cause**: This message is for inform
011034	ERROR	no resource has id of \"%s\"	**Cause**: LifeKeeper could not find **Action**: Verify the parameters and "sendevent" operation.
011044	ERROR	flagcleanup:fopen(%s	**Cause**: A system error has occurr /opt/LifeKeeper/config/flg. **Action**: Determine why /opt/LifeK not readable.
011045	ERROR	flagcleanup:fopen(%s	**Cause**: A system error has occurr /opt/LifeKeeper/config/flg. **Action**: Determine why /opt/LifeK not writable.
011046	ERROR	flagcleanup:fputs(%s	**Cause**: A system error has occurr /opt/LifeKeeper/config/flg. **Action**: Determine why /opt/LifeK not writable.
011047	ERROR	flagcleanup:rename(%s,%s	**Cause**: A system error has occurr renaming /opt/LifeKeeper/config/.f LifeKeeper/config/flg. **Action**: Determine why /opt/LifeK was not able to be renamed.
011048	ERROR	flagcleanup:chmod(%s	**Cause**: A system error has occurr permissions in /opt/LifeKeeper/co **Action**: Determine why LifeKeepe

Code	Severity	Message	Cause/Action
			change permissions in /opt/LifeKe
011049	ERROR	License check failed with error code %d	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011051	ERROR	lcdinit: clearing Disk Reserve file failed	**Cause**: A system error has occurr /opt/LifeKeeper/subsys/scsi/resou disk.reserve. **Action**: Determine why /opt/LifeK resources/disk/disk.reserve is not
011052	FATAL	malloc() failed	**Cause**: The system could not allo LifeKeeper. **Action**: Increase the process limit segment.
011053	FATAL	lcm_is_unavail	**Cause**: A system error has occurr /tmp/LK_IS_UNAVAIL. **Action**: Determine why /tmp/LK_I writable.
011054	FATAL	lk_is_unavail	**Cause**: A system error has occurr /opt/LifeKeeper/config/LK_IS_ON. **Action**: Determine why /opt/LifeK LK_IS_ON is not writable.
011055	FATAL	usr_alarm_config_LK_IS_ON	**Cause**: A system error has occurr /tmp/LCM_IS_UNAVAI. **Action**: Determine why /tmp/LCM not writable.

Code	Severity	Message	Cause/Action
011056	ERROR	License check failed with error code %d	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011057	ERROR	lcdremote: unknown command type %d(‘%c’)\n	**Cause**: A message failed to be re **Action**: Check adjacent log messa details. This may be a temporary error continues and servers canno verify the network configuration or
011059	FATAL	Could not write to: %s	**Cause**: A system error has occurr accessing /opt/LifeKeeper/config/L **Action**: Determine why /opt/LifeK LK_START_TIME is not accessibl
011060	FATAL	received NULL message	**Cause**: A message failed to be re **Action**: Check adjacent log messa details. This may be a temporary error continues and servers canno verify the network configuration or
011061	ERROR	unknown data type %d(‘%c’) on machine \"%s\"\n	**Cause**: A message failed to be re **Action**: Check adjacent log messa details. This may be a temporary error continues and servers canno verify the network configuration or
011062	WARN	LifeKeeper shutdown in progress. Unable to perform failover recovery processing for %s\n	**Cause**: LifeKeeper was unable to resource during shutdown. **Action**: Switch over the resource server manually.

Code	Severity	Message	Cause/Action
011063	WARN	LifeKeeper resource initialization in progress. Unable to perform failover recovery processing for %s\n	**Cause**: LifeKeeper was unable to resource during start up. **Action**: Switch over the resource after LifeKeeper starts up.
011068	ERROR	ERROR on command %s	**Cause**: An error occurred while ru "rlslocks" command. **Action**: Check adjacent messages
011070	ERROR	ERROR on command %s	**Cause**: An error occurred while ru "getlocks" command. **Action**: Check adjacent log messa details.
011080	FATAL	out of memory	**Cause**: Internal error. **Action**: Increase the process limit segment.
011081	FATAL	Failed to ask ksh to run: %s	**Cause**: A system error has occurr ksh. **Action**: Make sure the pdksh (v8. the steeleye-pdksh (v81 and later) installed.
011082	ERROR	Failed to remove: %s	**Cause**: A system error has occurr /tmp/LCM_IS_UNAVAIL. **Action**: Determine why /tmp/LCM not removable.
011083	ERROR	Failed to remove: %s	**Cause**: A system error has occurr

Code	Severity	Message	Cause/Action
			unlink /tmp/LK_IS_UNAVAIL. **Action**: Determine why /tmp/LK_I removable.
011084	FATAL	Failed to generate an IPC key based on: %s	**Cause**: A system error has occurr accessing /opt/LifeKeeper. **Action**: Determine why /opt/LifeK accessible.
011085	ERROR	semget(%s,%c) failed	**Cause**: A system error has occurr a semaphore. **Action**: Try to remove the semaph
011086	ERROR	shmget(%s,%c) failed	**Cause**: A system error has occurr a shared memory segment. **Action**: Try to remove the shared manually.
011087	ERROR	semctl(IPC_RMID) failed	**Cause**: A system error has occurr a semaphore. **Action**: Try to remove the semaph
011088	ERROR	shmctl(IPC_RMID) failed	**Cause**: A system error has occurr a shared memory segment. **Action**: Try to remove the shared manually.
011089	FATAL	Execution of lcdstatus on remote system <%s> failed\n	**Cause**: The remote {node} is dow the network or some other system occurred on the remote node.

Code	Severity	Message	Cause/Action
			Action: Bring the remote node ba check adjacent messages for addi or check the logs on the remote n information.
011090	FATAL		**Cause**: Internal error. **Action**: Perform the steps listed ir text.
011091	WARN		**Cause**: This will not occur under r circumstances.
011092	FATAL		**Cause**: There is a problem with yo **Action**: Perform the steps listed ir text.
011093	FATAL		**Cause**: There is a problem with yo **Action**: Perform the steps listed ir text.
011094	FATAL		**Cause**: There is a problem with yo **Action**: Perform the steps listed ir text.
011095	FATAL		**Cause**: There is a problem with yo **Action**: Perform the steps listed ir text.
011096	FATAL		**Cause**: There is a problem with yo

Code	Severity	Message	Cause/Action
			Action: Perform the steps listed in text.
011097	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011098	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011099	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011100	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011101	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011102	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011103	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011104	FATAL		**Cause**: There is a problem with yo **Action**: Contact Support to obtain

Code	Severity	Message	Cause/Action
011105	FATAL		**Cause**: There is a problem with yo **Action**: Perform the steps listed in text.
011111	ERROR	action \"%s\" on resource with tag \"%s\" has failed	**Cause**: The {action} for resource { **Action**: See adjacent error messa details.
011112	ERROR		**Cause**: LifeKeeper could not find device. **Action**: Check your LifeKeeper co
011113	ERROR	netremote: unknown subcommand type %d(‘%c’)\n	**Cause**: Internal error. **Action**: Try restarting the product
011114	ERROR	netremote: unknown command type %d(‘%c’)\n	**Cause**: Internal error. **Action**: Try restarting the product
011117	ERROR	sysremote: system \"%s\" not found on \"%s\"	**Cause**: An invalid system name w **Action**: Recheck the system nam command.
011118	ERROR	sysremote: unknown subcommand type %d(‘%c’)\n	**Cause**: Internal error. **Action**: Try restarting the product
011119	ERROR	sysremote: unknown command type %d(‘%c’)\n	**Cause**: Internal error. **Action**: Try restarting the product

Code	Severity	Message	Cause/Action
011120	ERROR	typremote: unknown command type %d(‘%c’)\n	**Cause**: Internal error. **Action**: Try restarting the product
011129	ERROR	Failure during run of ‘%s’ on behalf of %s	**Cause**: Command execution failed **Action**: See message details to d problem.
011130	ERROR	%s	**Cause**: The command {command} unexpected output. **Action**: Action should be determin of adjacent error messages.
011131	EMERG	demo_update_msg	**Cause**: There is a problem with yo **Action**: Contact Support to obtain
011132	EMERG	demo_tamper_msg	**Cause**: You have a demo license tampering has been detected. **Action**: Contact Support to obtain
011133	EMERG	demo_tamper_msg	**Cause**: You have a demo license tampering has been detected. **Action**: Contact Support to obtain
011134	EMERG	demo_expire_msg	**Cause**: The demo license for this expired. **Action**: Contact Support to obtain
011135	EMERG	demo_tamper_msg	**Cause**: You have a demo license

Code	Severity	Message	Cause/Action
			tampering has been detected. **Action**: Contact Support to obtain
011136	EMERG	buf	**Cause**: You are running a demo li **Action**: Contact Support to obtain
011138	EMERG	buf	**Cause**: You are running a demo li **Action**: Contact Support to obtain
011142	WARN	LifeKeeper Recovery Kit %s license key NOT FOUND	**Cause**: An Application Recovery K was not found. **Action**: Contact Support to obtain
011150	ERROR	COMMAND OUTPUT: %s	**Cause**: The command "eventslcm unexpected output. **Action**: Check the logs for related resolve the reported problem.
011151	EMERG	&localebuf[3]	**Cause**: This version of the LifeKe package is restricted to being use territories of the People's Republi Japan.
011152	EMERG	Localized license failure	**Cause**: There was a mis-match be and the locale for which the produ created. **Action**: Contact Support to obtain which matches your locale.

Code	Severity	Message	Cause/Action
011154	EMERG	Single Node flag check failed.	**Cause**: You have a license for Life Server Protection but you do not h Single Server Protection installed. **Action**: Either install LifeKeeper S Protection or obtain a license that product you are running.
011155	EMERG	lic_master_exp_msg, ""	**Cause**: Your license key for this p expired. **Action**: Contact Support to obtain
011162	EMERG	lic_restricted_exp_msg, ""	**Cause**: Your license key for this p expired. **Action**: Contact Support to obtain
011163	EMERG	Single Node license check failed	**Cause**: You have a license for Life Server Protection but you do not h Single Server Protection installed. **Action**: Either install LifeKeeper S Protection or obtain a license that product you are running.
011164	EMERG	demo_expire_msg, DEMO_UPGRADE_MSG	**Cause**: Your license key for this p expired. **Action**: Please contact Support to permanent license key for your pr
011165	ERROR	LifeKeeper initialize timed out in tag \"%s\"	
015000	ERROR	COMMAND OUTPUT: /opt/LifeKeeper/sbin/chpst	**Cause**: An error has ocurred with lighttpd" process. Specific details included in the actual log message

Code	Severity	Message	Cause/Action
			Action: Correct the configration a lighttpd" will automatically be resta
103001	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The db2nodes.cfg does no server names. **Action**: Ensure the db2nodes.cfg
103002	ERROR	LifeKeeper was unable to get the version for the requested instance "%s"	**Cause**: "db2level" command did n version. **Action**: Check your DB2 configura
103003	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Reco unable to find any nodes for the D **Action**: Check your DB2 configura
103004	ERROR	Unable to get the information for resource "%s"	**Cause**: Failed to get resource info **Action**: Check your LifeKeeper co
103005	ERROR	Unable to get the information for resource "%s"	**Cause**: Failed to get resource info **Action**: Check your LifeKeeper co
103006	ERROR	Unable to get the instance information for resource "%s"	**Cause**: Failed to get the instance **Action**: Check your LifeKeeper co
103007	ERROR	Unable to get the instance home directory information for resource "%s"	**Cause**: Failed to get the instance path. **Action**: Check your LifeKeeper co

Code	Severity	Message	Cause/Action
103008	ERROR	Unable to get the instance type information for resource "%s"	**Cause**: The DB2 Application Reco invalid instance type. **Action**: Check your LifeKeeper co
103009	ERROR	LifeKeeper has encountered an error while trying to get the database configuration parameters for database \"$DB\"	**Cause**: There was an unexpected "db2 get db cfg for $DB" comman **Action**: Check the logs for related resolve the reported problem.
103012	ERROR	LifeKeeper was unable to start the database server for instance "%s"	**Cause**: The requested startup of t failed. **Action**: Check the logs for related resolve the reported problem. Cor errors before retrying the "restore"
103013	ERROR	LifeKeeper was unable to start the database server for instance "%s"	**Cause**: The requested startup of t failed. **Action**: Check the logs for related resolve the reported problem. Cor errors before retrying the "restore"
103015	ERROR	An entry for the home directory "%s" of instance "%s" does not exist in "/etc/fstab"	**Cause**: The home directory of inst Partition database should exist in **Action**: Ensure the home director fstab".
103016	ERROR	LifeKeeper was unable to mount the home directory for the DB2 instance "%s"	**Cause**: Failed to mount the home instance of Multiple Partition datal **Action**: Ensure the home director retry the operation.

Code	Severity	Message	Cause/Action
103017	ERROR	Unable to get the instance nodes information for resource "%s"	**Cause**: Failed to get the instance **Action**: Check your LifeKeeper co
103018	ERROR	LifeKeeper was unable to start database partition server "%s" for instance "%s"	**Cause**: The requested startup of t failed. **Action**: Check the logs for related resolve the reported problem. Cor errors before retrying the "restore"
103020	ERROR	LifeKeeper was unable to stop the database server for instance "%s"	**Cause**: The requested shutdown instance failed. **Action**: Check the logs for related resolve the reported problem. Cor errors before retrying the "remove
103021	ERROR	LifeKeeper was unable to stop the database server for instance "%s"	**Cause**: The requested shutdown instance failed. **Action**: Check the logs for related resolve the reported problem. Cor errors before retrying the "remove
103023	ERROR	Unable to get the instance nodes information for resource "%s"	**Cause**: Failed to get the instance **Action**: Check your LifeKeeper co
103024	ERROR	LifeKeeper was unable to stop database partition server "%s" for instance "%s"	**Cause**: The requested shutdown instance failed. **Action**: Check the logs for related resolve the reported problem. Cor errors before retrying the "remove

Code	Severity	Message	Cause/Action
103026	ERROR	Unable to get the instance nodes information for resource "%s"	**Cause**: Failed to get the instance **Action**: Check your LifeKeeper co
103027	FATAL	The argument for the DB2 instance is empty	**Cause**: Invalid parameters were s create operation. **Action**: Verify the parameters and operation.
103028	FATAL	Unable to determine the DB2 instance home directory	**Cause**: The DB2 Application Reco unable to determine the DB2 insta directory. **Action**: Ensure the instance owne as the instance name and retry th
103029	FATAL	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Reco unable to determine the DB2 insta **Action**: Check your DB2 configura
103030	FATAL	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Reco unable to find any nodes for the D **Action**: Check your DB2 configura
103031	ERROR	The path "%s" is not on a shared filesystem	**Cause**: The instance home directo shared filesystem. **Action**: Ensure the path is on sha and retry the create operation.
103032	ERROR	LifeKeeper was unable to get the DB tablespace containers for instance "%s" or the log path for one of its databases	**Cause**: LifeKeeper could not dete of the database table space conta they are located in a path which is

Code	Severity	Message	Cause/Action
			filesystem. **Action**: Check the logs for related resolve the reported problem. Cor errors before retrying the "create"
103033	ERROR	The path "%s" is not on a shared filesystem	**Cause**: The path of database tabl should be on a shared filesystem. **Action**: Ensure database table sp on a shared filesystem and retry t
103034	ERROR	A DB2 Hierarchy already exists for instance "%s"	**Cause**: An attempt was made to p instance that is already under Life protection. **Action**: You must select a differer for LifeKeeper protection.
103035	ERROR	The file system resource "%s" is not in-service	**Cause**: The file system which the depends on should be in service. **Action**: Ensure the file system res service and retry the "create" oper
103036	ERROR	Unable to create the hierarchy for raw device "%s"	**Cause**: LifeKeeper was unable to resource {raw device} . **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the operation.
103037	ERROR	A RAW hierarchy does not exist for the tag "%s"	**Cause**: LifeKeeper was unable to resource {tag} . **Action**: Check your LifeKeeper co

Code	Severity	Message	Cause/Action
103038	ERROR	LifeKeeper was unable to create a dependency between the DB2 hierarchy "%s" and the Raw hierarchy "%s"	**Cause**: The requested dependenc between the parent DB2 resource Raw resource failed. **Action**: Check adjacent log messa details and related messages. Co errors before retrying the "create"
103039	ERROR	LifeKeeper could not disable the automatic startup feature of DB2 instance "%s"	**Cause**: An unexpected error occu attempting to update the DB2 sett **Action**: The DB2AUTOSTART wil updated manually to turn off the a of the instance at system boot.
103040	ERROR	DB2 version "%s" is not installed on server "%s"	**Cause**: LifeKeeper could not find location. **Action**: Check your DB2 configura
103041	ERROR	The instance owner "%s" does not exist on target server "%s"	**Cause**: An attempt to retrieve the owner from template server during "extend" operation failed. **Action**: Verify the DB2 instance o specified server. If the user does be created with the same uid and in the cluster.
103042	ERROR	The instance owner "%s" uids are different on target server "%s" and template server "%s"	**Cause**: The user id on the target s server} for the DB2 instance owne match the value of the user {user} server {template server}. **Action**: The user ids for the DB2 i {user} must match on all servers i user id mismatch should be correc all servers before retrying the "cr

Code	Severity	Message	Cause/Action
			operation.
103043	ERROR	The instance owner "%s" gids are different on target server "%s" and template server "%s"	**Cause**: The group id on the target server} for the DB2 instance owne match the value of the user {user} server {template server}. **Action**: The group ids for the DB2 {user} must match on all servers i group id mismatch should be corre all servers before retrying the "can operation.
103044	ERROR	The instance owner "%s" home directories are different on target server "%s" and template server "%s"	**Cause**: The home directory locatio {user} on the target server {target match the DB2 instance owner's h the template server {template serv **Action**: The home directory locati instance owner {user} must match the cluster. The location mismatch corrected manually on all servers the "canextend" operation.
103045	ERROR	LifeKeeper was unable to get the DB2 "SVCENAME" parameter for the DB2 instance	**Cause**: There was an unexpected "db2 get dbm cfg" command. **Action**: Check your DB2 configura
103046	ERROR	Unable to get the value of the DB2 "SVCENAME" parameter for the DB2 instance %s.	**Cause**: The DB2 "SVCENAME" pa null. **Action**: Check your DB2 configura
103047	ERROR	LifeKeeper was unable to get the contents of the "/etc/services" file on the server "%s"	**Cause**: "/etc/services" on the temp not contain the service names for

Code	Severity	Message	Cause/Action
			Action: The service names in "/et DB2 instance must match on all s cluster. The service names misma corrected manually on all servers the "canextend" operation.
103048	ERROR	LifeKeeper was unable to get the contents of the "/etc/ services" file on the server "%s"	**Cause**: "/etc/services" on the targ contain the service names for the **Action**: The service names in "/et DB2 instance must match on all s cluster. The service names misma corrected manually on all servers the "canextend" operation.
103049	ERROR	The "/etc/services" entries for the instance "%s" are different on target server "%s" and template server "%s"	**Cause**: The "/etc/services" entries are mismatched. **Action**: The service names in "/et DB2 instance must match on all s cluster. The service names misma corrected manually on all servers the "canextend" operation.
103050	ERROR	The home directory "%s" for instance "%s" is not mounted on server "%s"	**Cause**: LifeKeeper could not find Multiple Partition instance. **Action**: Ensure the home director retry the operation.
103051	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: Failed to get resource info template server. **Action**: Check your LifeKeeper co
103052	ERROR	LifeKeeper was unable to add instance "%s" and/or its variables to the DB2 registry	**Cause**: There was an unexpected "db2iset" command.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
103053	ERROR	Usage: %s instance	
103054	ERROR	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Reco unable to determine the DB2 insta **Action**: Check your DB2 configura
103055	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Reco unable to find any nodes for the D **Action**: Check your DB2 configura
103056	ERROR	Usage: %s instance	
103058	ERROR	Usage: %s instance	
103059	ERROR	Usage: %s instance	
103060	ERROR	Unable to determine the DB2 instance home directory	**Cause**: The DB2 Application Reco unable to determine the DB2 insta directory. **Action**: Ensure the instance owne as the instance name and retry th
103061	ERROR	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Reco unable to determine the DB2 insta **Action**: Check your DB2 configura
103062	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Reco unable to find the node for the DB **Action**: Check your DB2 configura
103063	ERROR	Unable to determine the DB2 install path	**Cause**: The DB2 Application Reco

Code	Severity	Message	Cause/Action
			unable to find DB2 for the instanc **Action**: Check your DB2 configura
103064	ERROR	Usage: nodes -t tag -a add_nodenum | nodes -t tag -d delete_nodenum | nodes -t tag -p	
103065	ERROR	Invalid input provided for "%s" utility operation, characters are not allowed.	**Cause**: Invalid parameters were s "nodes" command. **Action**: Verify the parameters and operation.
103066	ERROR	Unable to get the information for resource "%s"	**Cause**: LifeKeeper was unable to {tag}. **Action**: Verify the parameters and operation.
103067	ERROR	The DB2 instance "%s" is not a EEE or Multiple Partition instance	**Cause**: The resource {tag} is sing instance. **Action**: Verify the parameters and operation.
103069	ERROR	Node "%s" is already protected by this hierarchy	**Cause**: Invalid parameters were s "nodes" command. **Action**: Verify the parameters and operation.
103070	ERROR	Node number "%s" is the last remaining node protected by resource "%s". Deleting all nodes is not allowed.	**Cause**: Invalid parameters were s "nodes" command. **Action**: Verify the parameters and operation.

Code	Severity	Message	Cause/Action
103071	ERROR	LifeKeeper is unable to get the equivalent instance for resource "%s"	**Cause**: There was an unexpected "nodes" command. **Action**: Check the logs for related resolve the reported problem.
103072	ERROR	Unable to set NodesInfo for resource "%s" on "%s"	**Cause**: There was an unexpected "nodes" command. **Action**: Check the logs for related resolve the reported problem.
103073	ERROR	Unable to set NodesInfo for resource "%s" on "%s"	**Cause**: There was an unexpected "nodes" command. **Action**: Check the logs for related resolve the reported problem.
103074	ERROR	Usage: %s instance	
103075	ERROR	Usage: %s instance	
103076	ERROR	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Reco unable to determine the DB2 insta **Action**: Check your DB2 configura
103077	ERROR	Unable to determine the DB2 instance home directory	**Cause**: The DB2 Application Reco unable to determine the DB2 insta directory. **Action**: Ensure the instance owne as the instance name and retry th
103078	ERROR	The database server is not running for instance "%s"	**Cause**: A process check for the D not find any processes running. **Action**: The DB2 instance must b

Code	Severity	Message	Cause/Action
103079	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Reco unable to find any nodes for the D **Action**: Check your DB2 configura
103080	ERROR	One or more of the database partition servers for instance "%s" is down	**Cause**: All database partition serv running. **Action**: Ensure all database partit running and retry the operation.
103081	ERROR	DB2 local recovery detected another recovery process in progress for "%s" and will exit.	
103082	ERROR	Failed to create flag "%s"	**Cause**: An unexpected error occu create a flag for controlling DB2 lo processing. **Action**: Check the adjacent log m further details and related messag reported errors.
103083	ERROR	Failed to remove flag "%s"	**Cause**: An unexpected error occu remove a flag for controlling DB2 processing. **Action**: Check the adjacent log m further details and related messag reported errors.
103084	ERROR	Unable to determine the DB2 instance \"$Instance\" home directory	**Cause**: The DB2 Application Reco unable to determine the DB2 insta directory. **Action**: Ensure the instance owne as the instance name and retry th
104002	FATAL	$msg	**Cause**: This message indicates ar

Code	Severity	Message	Cause/Action
			error. **Action**: The stack trace indicates error.
104003	FATAL	$self->Val('Tag') . " is not an SDR resource"	**Cause**: A data replication action w a non data replication resource. **Action**: Check the logs for related resolve the reported problem.
104010	ERROR	$self->{'md'}: bitmap merge failed, $action	**Cause**: The bitmap merge operati **Action**: The target server may hav or protected filesystem mounted, may be missing on the target. Che server.
104022	ERROR	$argv[1]: mdadm failed ($ret	**Cause**: The "mdadm" command h device into the mirror. **Action**: This is usually a temporar
104023	ERROR	$_	**Cause**: The message contains the "mdadm" command.
104025	ERROR	failed to spawn monitor	**Cause**: The system failed to start monitor process. This should not h normal circumstances. **Action**: Reboot the system to ens potential conflicts are resolved.
104026	ERROR	cannot create $md	**Cause**: The mirror device could n **Action**: Ensure the device is not a

Code	Severity	Message	Cause/Action
			that all other parameters for the m correct.
104027	ERROR	$_	**Cause**: This message contains th command output.
104035	ERROR	Too many failures. Aborting resync of $md	**Cause**: The device was busy for a period of time. **Action**: Reboot the system to be s device is no longer busy.
104036	ERROR	Failed to start nbd-server on $target (error $port	**Cause**: The nbd-server process c started on the target server. **Action**: Ensure that the target dis available and that its Device ID ha
104037	ERROR	Failed to start compression (error $port	**Cause**: The system was unable to ‘balance’ tunnel process or there w problem. **Action**: Ensure that the network is properly and that TCP ports in the 10000-10512 are opened and unu the software is installed properly c
104038	ERROR	Failed to start nbd-client on $source (error $ret	**Cause**: The nbd-client process ha the source server. **Action**: Look up the reported errn resolve the problem reported. For errno value of 110 means "Conne which may indicate a network or fi
104039	ERROR	Failed to add $nbd to $md on $source	**Cause**: This is usually a temporar

Code	Severity	Message	Cause/Action
			Action: If this error persists, rebo resolve any potential conflicts.
104045	ERROR	failed to stop $self->{‘md’}	**Cause**: The mirror device could n **Action**: Ensure that the device is mounted. Try running "mdadm —s stop the device.
104048	WARN	failed to kill $proc, pid $pid	**Cause**: The process could not be may indicate that the process has **Action**: Ensure that the process i longer running. If it is, then reboot clear up the unkillable process.
104050	ERROR	Setting $name on $dest failed: $ret. Please try again.	**Cause**: The system failed to set a setting. **Action**: Check the network and sy the mirror setting operation.
104052	FATAL	Specified existing mount point "%s" is not mounted	**Cause**: The mount point became **Action**: Ensure that the mount po retry the operation.
104055	ERROR	Failed to set up temporary $type access to data for $self->{‘tag’}. Error: $ret	**Cause**: The filesystem or device v on the target server. The mirrored available on the target server until paused and resumed again. **Action**: Reboot the target server t potential conflicts.
104057	ERROR	Failed to undo temporary access for $self->{‘tag’} on $self->{‘sys’}. Error: $ret. Please verify that $fsid is not mounted on server $self->{‘sys’}.	**Cause**: The filesystem could not b

Code	Severity	Message	Cause/Action
			the target server. **Action**: Ensure that the filesystem not busy on the target server. Reb server to resolve any potential cor
104062	FATAL	Cannot find a device with unique ID "%s"	**Cause**: The target disk could not **Action**: Ensure that the appropria recovery kits are installed on the t Ensure that the Device ID of the ta changed.
104066	FATAL	Cannot get the hardware ID of device "%s"	**Cause**: A unique ID could not be f target disk device. **Action**: Ensure that the appropria recovery kits are installed on the t Ensure that the Device ID of the ta changed.
104067	FATAL	Asynchronous writes cannot be enabled without a bitmap file	**Cause**: An attempt was made to c with invalid parameters. **Action**: A bitmap file parameter m or synchronous writes must be sp
104068	FATAL	Failed to extend dependent resource %s to system %s. Error %s	**Cause**: The hierarchy extend oper **Action**: Check the logs for related resolve the reported problem.
104070	FATAL	Unable to extend the mirror "%s" to system "%s"	**Cause**: The hierarchy extend oper **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
104071	ERROR	Failed to restore target device resources on $target->{'sys'} : $err	**Cause**: The in-service operation h target server. **Action**: Check the logs for related resolve the reported problem.
104074	FATAL	Cannot get the hardware ID of device "%s"	**Cause**: There is no storage recov recognizes the underlying disk de attempting to use for the mirror. **Action**: Make sure the appropriate kits are installed. If necessary, pla name in the /opt/LifeKeeper/subsy DEVNAME/device_pattern file.
104081	FATAL	Cannot make the %s filesystem on "%s" (%d)	**Cause**: The "mkfs" command faile **Action**: Ensure that the disk devic free of errors and that the filesyste selected filesystem are installed.
104082	FATAL	%s	**Cause**: This message contains th "mkfs" command.
104083	FATAL	Cannot create filesys hierarchy "%s"	**Cause**: The resource creation fail **Action**: Check the logs for related resolve the reported problem.
104086	ERROR	The "%s_data_corrupt" flag is set in "%s/subsys/scsi/ resources/netraid/" on system "%s". To avoid data corruption, LifeKeeper will not restore the resource.	**Cause**: The data corrupt flag file h precaution to prevent accidental d The mirror cannot be restored on the file is removed. **Action**: If you are sure that the da server in question, you can either: file and restore the mirror, or 2) fo

Code	Severity	Message	Cause/Action
			online using the LifeKeeper GUI o force' command.
104092	ERROR	Mirror target resource movement to system %s : status %s	**Cause**: The hierarchy switchover failed. **Action**: Check the logs for related resolve the reported problem.
104099	ERROR	Unable to unextend the mirror for resource "%s" from system "%s"	**Cause**: The hierarchy unextend o **Action**: Reboot the target server t potential conflicts and retry the op
104106	ERROR	remote 'bitmap -m' command failed on $target->{'sys'}: $ranges	**Cause**: The bitmap merge comma target server. This may be caused things: 1) The bitmap file may be corrupted, or 2) the mirror (md) de active on the target. **Action**: Make sure that the mirror filesystem are not active on the ta bitmap file is missing, pause and to recreate the bitmap file.
104107	ERROR	Asynchronous writes cannot be enabled without a bitmap file	**Cause**: Invalid parameters were s mirror create operation.
104108	ERROR	Local Partition not available	**Cause**: Invalid parameters were s mirror create operation.
104109	ERROR	Cannot get the hardware ID of device "%s"	**Cause**: A unique ID could not be disk device. **Action**: Ensure that the appropria recovery kits are installed on the s

Code	Severity	Message	Cause/Action
			that the Device ID of the disk has
104111	FATAL	Insufficient input parameters for "%s" creation	**Cause**: Invalid parameters were s mirror create operation.
104112	FATAL	Insufficient input parameters for "%s" creation	**Cause**: Invalid parameters were s mirror create operation.
104113	FATAL	Insufficient input parameters for "%s" creation	**Cause**: Invalid parameters were s mirror create operation.
104114	FATAL	Insufficient input parameters for "%s" creation	**Cause**: Invalid parameters were s mirror create operation.
104115	FATAL	Insufficient input parameters for "%s" creation	**Cause**: Invalid parameters were s mirror create operation.
104117	FATAL	Insufficient input parameters for "%s" creation	**Cause**: Invalid parameters were s mirror create operation.
104118	FATAL	Cannot unmount existing Mount Point "%s"	**Cause**: The mount point is busy. **Action**: Make sure the filesystem any processes or applications tha accessing the filesystem.
104119	FATAL	Invalid data replication resource type requested ("%s"	**Cause**: An invalid parameter was mirror create operation.
104124	EMERG	WARNING: A temporary communication failure has occurred between systems %s and %s. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should take one of the following resources out of service: %s on %s or %s on %s. The	**Cause**: A temporary communicati brain scenario) has occurred betw and target servers.

Code	Severity	Message	Cause/Action
		resource that is taken out of service will become the mirror target.	**Action**: Perform the steps listed in text.
104125	ERROR	failed to start '$cmd $_[2] $user_args' on '$_[3]	**Cause**: The specified command fa **Action**: Check the logs for related resolve the reported problem.
104126	ERROR	$_	**Cause**: This message contains th command that was reported as fai 104125.
104128	FATAL	comm path/server not specified	**Cause**: The netraid.down script w specifying the communication path name. This script is called interna always have the proper parameter **Action**: Report this error to SIOS
104129	WARN		**Cause**: The replication connection down. **Action**: Check the network.
104130	ERROR	Mirror resize failed on %s (%s). You must successfully complete this operation before using the mirror. Please try again.	**Cause**: The mirror resize operatio update the mirror metadata on the **Action**: You must successfully co before using the mirror. Re-run mi (possibly using -f to force the oper necessary).
104132	ERROR	The partition "%s" has an odd number of sectors and system "%s&qquot; is running kernel >= 4.12. Mirrors with this configuration will not work correctly with DataKeeper. Please see the SIOS product documentation for information on how to resize the mirror.	**Cause**: The partition or disk chose creation has an odd number of dis have to be resized to be used with

Code	Severity	Message	Cause/Action
			Action: Resize the partition using command or resize the disk (is po platform (VMware, AWS) tools. Ca be lost if this is not done carefully.
104136	ERROR	Extend failed.	**Cause**: The hierarchy extend oper **Action**: Check the logs for related resolve the reported problem.
104143	ERROR	Mirror resume was unsuccessful ($ret	**Cause**: The mirror could not be es **Action**: Check the logs for related resolve the reported problems.
104144	ERROR	Unable to stop the mirror access for $self->{'md'} on system $self->{'sys'}. Error: $ret. Use the \"mdadm —stop $self->{'md'}\" command to manually stop the mirror.	**Cause**: The mirror device created node when the mirror was paused stopped. **Action**: Ensure that the device is mounted. Try running "mdadm —s stop the device.
104145	WARN	Unable to dirty full bitmap. Setting fullsync flag.	**Cause**: A full resync could not be the full bitmap. The fullsync flag w instead. This is a non-fatal error a synchronization will still be done. **Action**: None
104146	EMERG	WARNING: The target system %s for mirror %s has the target mirror %s currently active. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should reboot system %s.	**Cause**: The mirror is configured o system. **Action**: The target system should DataKeeper should then be able t mirror.

Code	Severity	Message	Cause/Action
104147	EMERG	WARNING: The target system %s for mirror %s has the target disk %s currently mounted. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should unmount %s on %s. A full resync will occur.	**Cause**: The mirror disk is mounte system. **Action**: The mirror disk should be the target system, in order to initia resync. A full resync is required b changes have occurred on the dis
104148	EMERG	The storage configuration for mirror "%s (%s)" does not have a unique identifier and may have potential for data corruption in some environments in certain circumstances. Please refer to the SIOS Product Documentation for details on DataKeeper storage configuration options.	**Cause**: The disk chosen for mirror provide a UUID to the operating s DataKeeper cannot mirror a disk w **Action**: You may be able to create table on the disk to provide a UUI partitions.
104156	WARN	Resynchronization of "%s" is in PENDING state. Current sync_action is: "%s"	**Cause**: The resynchronization of t detected in PENDING state. **Action**: LifeKeeper will try to fix th a resynchronization. Check the log errors. When successful assure th state has been cleared in /proc/md resynchronization is in progress o completed for the datarep resourc
104157	WARN	/etc/sysconfig/raid-check update failed. Please %s \"md%d\" to SKIP_DEVS.	**Cause**: Unable to make changes i raid-check to add or remove an en MD devices to skip (SKIP_DEVS). **Action**: Check system logs for an raid-check or SKIP_DEVS. Manua md listed.
104158	EMERG	WARNING: The local disk partition $self->{'part'} for data replication device\n$self->{'md'} has failed. MANUAL INTERVENTION IS REQUIRED.	**Cause**: The local device for a mirr recovery action has been set to no LKDR_FAILURE requiring manual recover.

Code	Severity	Message	Cause/Action
			Action: Check system logs and Li errors related to the local disk.
104163	WARN	The "%s_data_corrupt" flag is set in "%s/subsys/scsi/resources/netraid/" on system "%s". The mirror is being forced online.	**Cause**: The mirror is being forced the data_corrupt flag. The data on system will be treated as the lates not correct then this can lead to da data loss. **Action**: None
104164	ERROR	The "%s_data_corrupt" flag for related mirror resource "%s" is set in "%s/subsys/scsi/resources/netraid/" on system "%s". To avoid data corruption, LifeKeeper will not restore this mirror or any related mirrors in the hierarchy.	**Cause**: The data_corrupt flag exis mirrors in the hierarchy. To avoid additional data none of the mirrors service until all of the data_corrup resolved. **Action**: Check the LifeKeeper log where each mirror was last in-serv the latest data for each mirror resi should be brought in-service on th where the full hierarchy was in-se the mirrors to synchronize with all
104165	ERROR	The "%s_data_corrupt" flag for related mirror resource "%s" is set in "%s/subsys/scsi/resources/netraid/" on system "%s". The mirror resource "%s" is being forced online.	**Cause**: The mirror is being forced the data_corrupt flag. The data on system will be treated as the corre synchronized with all targets. This corruption or data loss if this is no **Action**: None
104170	ERROR	Failed to create \"source\" flag file on shared source %s to track mirror source. This may result in a full resync.	**Cause**: The 'source' flag file was specified system to track the mirro **Action**: Check the LKROOT (/opt/ system for errors or that it is full.

Code	Severity	Message	Cause/Action
104171	ERROR	Failed to create \"source\" flag file on %s to track mirror source. Target %s will not be added to mirror.	**Cause**: The 'source' flag file was specified system to track the mirro **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104172	ERROR	The \"source\" flag file on %s does not contain a valid target (%s). Full resync to remaining targets is required.	**Cause**: The 'source' flag file shou system name of a previous source listed was not found in the list of s configured. **Action**: Report this problem to SI
104173	ERROR	Failed to create \"source\" flag file on %s to track mirror source.	**Cause**: The 'source' flag file was specified system to track the mirro **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104174	ERROR	Failed to create \"previous_source\" flag file to track time waiting on source. Will not be able to timeout.	**Cause**: The 'previous_source' flag created on the mirror source to tra previous source. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104175	ERROR	Failed to create "data_corrupt" flag file on target "%s".	**Cause**: The 'data_corrupt' flag file on the target listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104176	ERROR	The \"source\" flag file on %s to track mirror source does not exist. Full resync is required.	**Cause**: The source flag file should system and without it the consiste can not be verified. A full resync is assure data reliability.

Code	Severity	Message	Cause/Action
			Action: Check the LKROOT (/opt/ system for error or that it is full.
104177	ERROR	Failed to determine amount of time waiting on %s.	**Cause**: The amount of time waitin source could not be determined. T added with a full resync if the prev found. **Action**: none
104178	ERROR	Failed to update "source" flag file on target "%s", previous source must be merged first.	**Cause**: The source flag file on the when it is in-sync and stopped so service does not require the previ **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104180	ERROR	Internal Error: \"previous_source\" has the local system name (%s).	**Cause**: The local system name sh previous_source flag file. **Action**: Report this error to SIOS
104181	ERROR	Internal Error: There are no targets waiting on %s to be merged.	**Cause**: There are no targets waiti source to merge. **Action**: Report this error to SIOS
104182	ERROR	Failed to create \"source\" flag file on %s to track mirror source. This may result in a full resync.	**Cause**: The ‘source’ flag file was target listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104186	ERROR	Failed to create \"last_owner\" flag file on %s to track mirror source. This may allow in-service of mirror on old data.	**Cause**: The ‘last_owner’ flag file v the source.

Code	Severity	Message	Cause/Action
			Action: Check the LKROOT (/opt/ system for errors or that it is full.
104187	WARN	$REM_MACH has ${REM_TAG}_last_owner file, create flag ${FLAGTAG}_data_corrupt.	**Cause**: The system listed had the last. **Action**: The system listed has the that indicates it has the most rece most likely the best system to in-s to avoid losing data.
104188	WARN	$REM_MACH is not alive, create flag ${FLAGTAG}_data_corrupt.	**Cause**: The system listed is not a **Action**: Since the system listed is cannot be determined whether tha more recent mirror source than th Therefore the local system should be allowed to bring the mirror in-s
104200	EMERG	Continue to wait for %s to merge bitmap and do partial resyncs to all targets, no timeout set.	**Cause**: In a multi-target configura not be configured until the previou available to merge its bitmap so th be able to partially resynchronize. LKDR_WAIT_ON_PREVIOUS_SO entry in /etc/defaults/LifeKeeper is indefinitely. **Action**: Check on the status of the listed in the message and resolve are preventing it from rejoining the
104201	EMERG	To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: \"%s/bin/mirror_action %s fullresync %s %s\" on %s.	**Cause**: In a multi-target configura not being configured, waiting on th source to rejoin the cluster. **Action**: Run the command listed i force an immediate full resynchron target and any remaining targets v

Code	Severity	Message	Cause/Action
			resynchronized.
104202	EMERG	Continue to wait for %s to merge bitmap and do partial resyncs to all targets. Continue to wait %d more seconds.	**Cause**: In a multi-target configura not be configured until the previou available to merge its bitmap so th be able to partially resynchronize. LKDR_WAIT_ON_PREVIOUS_SC entry in /etc/defaults/LifeKeeper is number of seconds to wait. If the p does not join the cluster in that tim be added with a full resynchroniza **Action**: Check on the status of the listed in the message and resolve are preventing it from rejoining the
104203	EMERG	To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: \"%s/bin/mirror_action %s fullresync %s %s\" on %s.	**Cause**: In a multi-target configura not being configured, waiting on th source to rejoin the cluster. **Action**: Run the command listed i force an immediate full resynchron target and any remaining targets w resynchronized. Note: Run this command to stop w target listed is deleted and never r
104207	ERROR	Failed to create "data_corrupt" flag file on "%s".	**Cause**: The ‘data_corrupt’ flag file on the source listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104208	ERROR	Failed to create "data_corrupt" flag file on target "%s".	**Cause**: The ‘data_corrupt’ flag file on the target listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.

Code	Severity	Message	Cause/Action
104209	ERROR	Failed to create "data_corrupt" flag file on "%s".	**Cause**: The ‘data_corrupt’ flag file on the source listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104210	ERROR	Failed to create "data_corrupt" flag file on target "%s".	**Cause**: The ‘data_corrupt’ flag file on the target listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104211	ERROR	Failed to create "data_corrupt" flag file on target "%s".	**Cause**: The ‘data_corrupt’ flag file on the target listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104212	ERROR	The \"source\" flag file on %s to track mirror source does not exist. Full resync to remaining targets is required.	**Cause**: The source flag file should system and without it the consiste can not be verified. A full resync is assure data reliability. All targets mirrored will require a full resync. **Action**: Check the LKROOT (/opt/ system for error or that it is full.
104214	ERROR	Failed to create \"source\" flag file on %s to track mirror source.	**Cause**: The ‘source’ flag file was specified system to track the mirro **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104216	ERROR	Failed to create "data_corrupt" flag file on target "%s".	**Cause**: The ‘data_corrupt’ flag file on the target listed. **Action**: Check the LKROOT (/opt/

Code	Severity	Message	Cause/Action
			system for errors or that it is full.
104217	ERROR	Failed to create \"source\" flag file on shared source %s to track mirror source. This may result in a full resync.	**Cause**: The ‘source’ flag file was specified system to track the mirro **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104218	ERROR	Failed to create "data_corrupt" flag file on target "%s".	**Cause**: The ‘data_corrupt’ flag file on the target listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full
104221	ERROR	Failed to create \"last_owner\" flag file on %s to track mirror source. This may allow in-service of mirror on old data.	**Cause**: The ‘last_owner’ flag file w the target listed. **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104222	ERROR	Failed to create "last_owner" flag file to track mirror source". This may allow in-service of mirror on old data.	**Cause**: The ‘last_ownerâ€™ flag know where the mirror was last in- **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104223	ERROR	Failed to create "last_owner" flag file to track mirror source". This may allow in-service of mirror on old data.	**Cause**: The ‘last_ownerâ€™ flag know where the mirror was last in- **Action**: Check the LKROOT (/opt/ system for errors or that it is full.
104224	ERROR	Failed to create \"previous_source\" flag file.	**Cause**: The ‘previous_source’ flag created. This is needed to merge source bitmap to avoid a full resyn

Code	Severity	Message	Cause/Action
			Action: Check the LKROOT (/opt/ system for errors or that it is full.
104227	ERROR	Failed to set %s to %s.	**Cause**: This message indicates a sysfs parameter for the nbd driver nbdX). **Action**: It may be necessary to ad of: NBD_NR_REQUESTS NBD_SCHEDULER LKDR_ASYNC_LIMIT in /etc/default/LifeKeeper to avoid
104232	ERROR	Mirror resize failed on %s (%s). Could not set size to %d.	**Cause**: The mirror resize operatio
104233	ERROR	Mirror resize failed on %s (%s). Could not set bitmap to %s and bitmap-chunk to %d.	**Cause**: The mirror resize operatio
104234	ERROR	The mirror %s failed to resize. You must successfully complete this operation before using the mirror. Please try again.	**Cause**: The mirror resize operatio
104235	ERROR	mirror_resize of mirror %s failed due to signal "%s".	**Cause**: The mirror resize operatio
104251	ERROR	There is no LifeKeeper protected resource with tag $tag on system $me.	**Cause**: The given tag does not co LifeKeeper protected resource on **Action**: Verify that the resource ta name are correct.
104252	ERROR	Resource $tag is not a $app/$typ resource. Please use the $ins_app/$ins_typ resource-specific canfailover script instead.	**Cause**: The scsi/netraid-specific c was called for a non-scsi/netraid r **Action**: Use the canfailover script corresponding to the appropriate a

Code	Severity	Message	Cause/Action
			the given resource.
112976	ERROR	There is no LifeKeeper protected resource with tag $tag on system $me.	
112977	ERROR	Resource $tag is not a $app/$typ resource. Please use the $ins_app/$ins_typ resource-specific canfailover script instead.	
122005	ERROR	Unable to "%s" on "%s"	**Cause**: There was an unexpected "getlocks". **Action**: Check the logs for related resolve the reported problem.
122007	ERROR	Unable to "%s" on "%s"	**Cause**: There was an unexpected "rlslocks". **Action**: Check the logs for related resolve the reported problem.
122009	ERROR	The path %s is not a valid file.	**Cause**: There is no listener.ora fil **Action**: Ensure the file exists and operation.
122010	ERROR	The listener user does not exist on the server %s.	**Cause**: "Stat" command could not **Action**: Retry the operation.
122011	ERROR	The listener user does not exist on the server %s.	**Cause**: UID is not in passwd file. **Action**: Ensure the UID exists in p retry the operation.
122012	ERROR	The listener user does not exist on the server %s.	**Cause**: User name is not in passw **Action**: Ensure the user name exi

Code	Severity	Message	Cause/Action
			retry the operation.
122023	ERROR	The %s command failed (%d	**Cause**: This message contains th the "lsnrctl" command. **Action**: Check the logs for related resolve the reported problem.
122024	ERROR	$line	**Cause**: The message contains the "lsnrctl" command. **Action**: Check the logs for related resolve the reported problem.
122039	ERROR	Usage error	**Cause**: Invalid parameters were s restore operation. **Action**: Verify the parameters and operation.
122040	ERROR	Script $cmd has hung on the restore of \"$opt_t\". Forcibly terminating.	**Cause**: The listener restore script timeout value. **Action**: Ensure listener.ora is vali LSNR_START_TIME (default 35 s default/LifeKeeper is set to a valu equal to the time needed to start t
122041	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the logs for related resolve the reported problem.
122045	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: Failed to get resource info

Code	Severity	Message	Cause/Action
			Action: Check your LifeKeeper co
122046	ERROR	Usage error	**Cause**: Invalid parameters were s restore operation. **Action**: Verify the parameters and operation.
122049	ERROR	The script $cmd has hung on remove of \"$opt_t\". Forcibly terminating.	**Cause**: The listener remove script timeout value. **Action**: Ensure listener.ora is vali LSNR_STOP_TIME (default 35 se default/LifeKeeper is set to a valu equal to the time needed to stop t
122051	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122055	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the logs for related resolve the reported problem.
122057	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122064	WARN	The %s level is set to %s a %s will not occur.	**Cause**: The minimal Listener prot Start and Monitor. **Action**: Start the listener manually

Code	Severity	Message	Cause/Action
122066	ERROR	Script has hung checking \"$tag\". Forcibly terminating.	**Cause**: The listener quickCheck s timeout value. **Action**: Ensure listener.ora is vali LSNR_STATUS_TIME (default 15 default/LifeKeeper is set to a valu equal to the time needed to check
122067	ERROR	Usage error	**Cause**: Invalid parameters were s quickCheck operation. **Action**: Verify the parameters and operation.
122069	ERROR	Usage error	**Cause**: Invalid parameters were s delete operation. **Action**: Verify the parameters and operation.
122072	ERROR	%s: resource "%s" not found on local server	**Cause**: Invalid parameters were s recover operation. **Action**: Verify the parameters and operation.
122074	WARN	The local recovery attempt has failed but %s level is set to %s preventing a failover to another node in the cluster. With %s recovery set all local recovery failures will exit successfully to prevent resource failovers.	**Cause**: The optional listener recov local recovery only. **Action**: Switch over the resource
122078	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
122082	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122083	ERROR	$cmd has hung checking \"$tag\". Forcibly terminating	**Cause**: The recover script was sto **Action**: Ensure listener.ora is vali
122084	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Verify the parameters and operation.
122085	ERROR	Usage: %s %s	**Cause**: Invalid parameters were s canextend operation. **Action**: Verify the parameters and operation.
122086	ERROR	The values specified for the target and the template servers are the same. Please specify the correct values for the target and template servers.	**Cause**: The values specified for th template servers are the same. **Action**: Perform the steps listed in text.
122087	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122088	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: Failed to get listener user resource information. **Action**: Ensure the resource info

Code	Severity	Message	Cause/Action
			retry the operation.
122089	ERROR	The listener user %s does not exist on the server %s.	**Cause**: User name is not in passw **Action**: Ensure the user name exi and retry the operation.
122090	ERROR	The id for user %s is not the same on template server %s and target server %s.	**Cause**: User ID should be same o **Action**: Trim user ID to the same.
122091	ERROR	The group id for user %s is not the same on template server %s and target server %s.	**Cause**: Group ID should be same **Action**: Trim group ID to the same
122092	ERROR	Cannot access canextend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to checks because it was unable to f "canextend" script on {server}. **Action**: Check your LifeKeeper co
122097	ERROR	Usage: %s %s	**Cause**: Invalid arguments were sp "configActions" operation. **Action**: Verify the arguments and operation.
122098	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122099	ERROR	Unable to update the resource %s to change the %s to %s on %s.	**Cause**: LifeKeeper failed to put in info field.

Code	Severity	Message	Cause/Action
			Action: Restart LifeKeeper and re
122100	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122101	ERROR	Unable to update the resource %s to change the %s to %s on %s.	**Cause**: LifeKeeper failed to put in field on {server}. **Action**: Restart LifeKeeper on {se operation.
122103	ERROR	Usage: %s %s	**Cause**: Invalid parameters were s create operation. **Action**: Verify the parameters and operation.
122124	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the logs for related resolve the reported problem.
122126	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected "rlslocks". **Action**: Check the logs for related resolve the reported problem.
122127	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
122129	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected "getlocks." **Action**: Check the logs for related resolve the reported problem.
122131	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check your LifeKeeper co
122133	ERROR	Unable to create a file system resource hierarchy for the file system %s.	**Cause**: There was an unexpected "filesyshier." **Action**: Check adjacent log messa details.
122135	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected "dep_create." **Action**: Check the logs for related resolve the reported problem.
122140	ERROR	Resource "%s" is not ISP on server "%s" Manually bring the resource in service and retry the operation	**Cause**: IP resource {tag} which th resource depends on should be IS **Action**: Perform the steps listed in text.
122141	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected "dep_create." **Action**: Check the logs for related resolve the reported problem.
122144	ERROR	Usage: %s %s	**Cause**: Invalid parameters were s

Code	Severity	Message	Cause/Action
			"create_ins" operation. **Action**: Verify the parameters and operation.
122145	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected "app_create." **Action**: Check the logs for related resolve the reported problem.
122146	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected "typ_create." **Action**: Check the logs for related resolve the reported problem.
122147	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected "newtag." **Action**: Check the logs for related resolve the reported problem.
122148	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check your LifeKeeper co
122149	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected "ins_setstate." **Action**: Check the logs for related resolve the reported problem.
122150	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}.

Code	Severity	Message	Cause/Action
			Action: Check your LifeKeeper co
122151	ERROR	Usage: %s %s	**Cause**: Invalid arguments were sp "depstoextend" operation. **Action**: Verify the arguments and operation.
122152	ERROR	Usage: %s %s	**Cause**: Invalid parameters were s "extend" operation. **Action**: Verify the parameters and operation.
122153	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122154	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}.
122155	ERROR	Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s"	**Cause**: During the Listener resour resource instance was found using and/or {id} but with a different reso and type. **Action**: Resource IDs must be un resource instance with the ID mat Listener resource instance must b
122156	ERROR	Cannot access extend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to resource hierarchy because it was find the script EXTEND on {server **Action**: Check your LifeKeeper co

Code	Severity	Message	Cause/Action
122157	ERROR	Usage: %s %s	**Cause**: Invalid arguments were sp "getConfigIps" operation. **Action**: Verify the arguments and operation.
122158	ERROR	The file %s is not a valid listener file. The file does not contain any listener definitions.	**Cause**: Failed to find any listener **Action**: Ensure listener definition listener.ora and retry the operatio
122159	ERROR	Usage: %s %s	**Cause**: Invalid arguments were sp "getSidListeners" operation. **Action**: Verify the arguments and operation.
122160	ERROR	The file %s is not a valid listener file. The file does not contain any listener definitions.	**Cause**: Failed to find any listener **Action**: Ensure listener definition listener.ora and retry the operatio
122161	ERROR	Usage: %s %s	**Cause**: Invalid arguments were sp "lsn-display" operation. **Action**: Verify the arguments and operation.
122162	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to {tag} on {server}. **Action**: Check your LifeKeeper co
122163	ERROR	Usage: %s %s	**Cause**: Invalid parameters were s updateHelper operation.

Code	Severity	Message	Cause/Action
			Action: Verify the parameters and operation.
122164	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the logs for related resolve the reported problem.
122166	ERROR	Usage: %s %s	**Cause**: Invalid parameters were s "updateHelper" operation. **Action**: Verify the parameters and operation.
122170	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected "dep_create." **Action**: Check the logs for related resolve the reported problem.
122171	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected "dep_create." **Action**: Check the logs for related resolve the reported problem.
122172	ERROR	Usage: %s %s	**Cause**: Invalid arguments were sp "updIPDeps" operation. **Action**: Verify the arguments and operation.
122173	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: LifeKeeper was unable to resource {resource} on {server}.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
122175	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected "rlslocks." **Action**: Check the logs for related resolve the reported problem.
122177	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected "getlocks." **Action**: Check the logs for related resolve the reported problem.
122180	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected "dep_create." **Action**: Check the logs for related resolve the reported problem.
122181	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected "dep_create." **Action**: Check the logs for related resolve the reported problem.
122183	ERROR	The path %s is not a valid file.	**Cause**: There is no listener.ora fil **Action**: Ensure the file exists and operation.
122185	ERROR	The file %s is not a valid listener file. The file does not contain any listener definitions.	**Cause**: LifeKeeper failed to find a definitions. **Action**: Ensure there are valid list

Code	Severity	Message	Cause/Action
			the listener.ora and retry the opera
122186	ERROR	The value specified for %s cannot be empty. Please specify a value for this field.	**Cause**: The config and/or executa empty. **Action**: Input a non-empty value f the operation.
122187	ERROR	The path %s is not a valid file or directory.	**Cause**: The defined {path} is inval **Action**: Ensure the {path} exists a operation.
122188	ERROR	The path %s is not a valid file or directory.	**Cause**: There is no {path}. **Action**: Ensure the {path} exists a operation.
122189	ERROR	The value specified for %s cannot be empty. Please specify a value for this field.	**Cause**: The config and/or executa empty. **Action**: Input path for the field.
122190	ERROR	Usage: %s %s	**Cause**: Invalid arguments were sp "valid_rpath" operation. **Action**: Verify the arguments and operation.
122191	ERROR	The values specified for the target and the template servers are the same.	**Cause**: Invalid argument of valid_ **Action**: Ensure arguments and re
122192	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in / {path}.

Code	Severity	Message	Cause/Action
			Action: Ensure oratab file exists i ORACLE_ORATABLOC in /etc/de set to a valid path.
122193	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in / {path}. **Action**: Ensure oratab file exists i ORACLE_ORATABLOC in /etc/de set to a valid path.
122194	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in / {path}. **Action**: Ensure oratab file exists i ORACLE_ORATABLOC in /etc/de set to a valid path.
122195	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in / {path}. **Action**: Ensure oratab file exists i ORACLE_ORATABLOC in /etc/de set to a valid path.
122196	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the logs for related resolve the reported problem.
122197	ERROR	Unable to find the configuration file \"oratab\" in its default locations, /etc/oratab or $listener::oraTab on \"$me\"	**Cause**: There is no oratab file in / {path}. **Action**: Ensure oratab file exists i ORACLE_ORATABLOC in /etc/de set to a valid path.

Code	Severity	Message	Cause/Action
122198	ERROR	remove for $okListener failed.	
122251	ERROR	Update of pluggable database info field for "%s" on "%s" failed (%s).	
122252	ERROR	Initial connect with query buffer to database "%s" on "%s" failed, testing output.	
122253	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	
122261	ERROR	The Oracle resource (%s) and dependency are not set on %s.	
122262	ERROR	Usage: %s %s	
122263	ERROR	The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper.	
122264	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122268	ERROR	Failed to create object instance for Oracle on "%s".	
122269	ERROR	no dependency for Oracle on "%s".	
122270	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122271	ERROR	Usage: %s %s	
122272	ERROR	The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper.	
122273	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122277	ERROR	Usage: %s %s	
122278	ERROR	Failed to create object instance for Oracle on "%s".	
122279	ERROR	no dependency for Oracle on "%s".	
122280	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122281	ERROR	The quickCheck of %s has timed out on server %s. The default TIMEOUT is 45 seconds. To increase the TIMEOUT, set ORACLE_QUICKCHECK_TIMEOUT in /etc/default/LifeKeeper.	
122282	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122284	ERROR	Failed to create object instance for Oracle on "%s".	
122285	ERROR	no dependency for Oracle on "%s".	
122287	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122288	ERROR	Usage: %s %s	
122291	ERROR	Cannot extend resource "%s" to server "%s"	
122292	ERROR	The values specified for the target and the template servers are the same: "%s".	
122294	ERROR	Cannot access canextend script "%s" on server "%s"	
122295	ERROR	Usage: %s %s	

Code	Severity	Message	Cause/Action
122296	ERROR	DB instance "%s" is not protected on "%s".	
122297	ERROR	Failed to create object instance for OraclePDB on "%s".	
122298	ERROR	Unable to locate the oratab file "%s" on "%s".	
122299	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	
122301	ERROR	Unable to "%s" on "%s" during resource create.	
122302	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122304	ERROR	Unable to "%s" on "%s" during resource create.	
122305	ERROR	Unable to determine Oracle user for "%s" on "%s".	
122306	ERROR	Error creating resource "%s" on server "%s"	
122308	ERROR	Dependency creation between Oracle pluggable database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	
122309	ERROR	%s	
122311	ERROR	In-service attempted failed for tag "%s" on "%s".	
122312	ERROR	Usage: %s %s	
122313	ERROR	Create of app "%s" on "%s" failed with return code of "%d".	
122314	ERROR	Create of typ "%s" for app "%s" on "%s" failed with return code of "%d".	
122316	ERROR	Create of resource tag via "newtag" on "%s" failed.	
122318	ERROR	Error creating resource "%s" on server "%s"	
122320	ERROR	Setting "resstate" for resource "%s" on "%s" failed with return code of "%d".	
122321	ERROR	Error creating resource "%s" on server "%s"	
122322	ERROR	Usage: %s %s	
122323	ERROR	Usage: %s %s	
122324	ERROR	Usage: %s %s	
122325	ERROR	Cannot extend resource "%s" to server "%s"	
122326	ERROR	Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s"	
122327	ERROR	Error creating resource "%s" on server "%s"	
122328	ERROR	Cannot access extend script "%s" on server "%s"	
122329	ERROR	Cannot extend resource "%s" to server "%s"	
122330	ERROR	Usage: %s %s	
122331	ERROR	Failed to create object instance for OraclePDB on "%s".	
122332	ERROR	Usage: %s %s	
122334	ERROR	Backup node %s is unreachable; abort protection PDB changes.	

Code	Severity	Message	Cause/Action
122336	ERROR	Update of protection PDB failed for "%s" on "%s".	
122339	ERROR	Usage: %s %s	
122340	ERROR	Usage: %s %s	
122341	ERROR	Usage: %s %s	
122342	ERROR	Failed to create object instance for OraclePDB on "%s".	
122343	ERROR	Usage: %s %s	
122344	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122348	ERROR	Failed to create flag "%s" on "%s".	
122350	ERROR	Failed to create object instance for Oracle on "%s".	
122351	ERROR	no dependency for Oracle on "%s".	
122353	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122356	ERROR	Usage: %s %s	
122357	ERROR	Failed to create object instance for OraclePDB on "%s".	
122358	ERROR	The selected oracle SID "%s" is not a CDB.	
122359	ERROR	No protectable PDB found for the selected SID "%s".	
122360	ERROR	No protected Oracle database found on "%s".	
122500	ERROR	Usage: %s %s	**Cause**: Invalid parameters were s create operation. **Action**: Verify the parameters are the operation.
122501	ERROR	DB instance "%s" is already protected on "%s".	**Cause**: An attempt was made to p database instance {sid} that is alre LifeKeeper protection on {server}. **Action**: You must select a differen instance {sid} for LifeKeeper prote
122502	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the

Code	Severity	Message	Cause/Action
122503	ERROR	Unable to locate the oratab file "%s" on "%s".	**Cause**: The oratab file was not fo or alternate locations on {server}. **Action**: Verify the oratab file exist permissions for the Oracle user. A is required to complete the "create
122504	ERROR	Unable to determine Oracle user for "%s" on "%s".	**Cause**: The Oracle Application Re unable to determine the ownershi database installation binaries. **Action**: The owner of the Oracle b valid non-root user on {server}. Co permissions and ownership of the installation and retry the operation
122505	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	**Cause**: The database instance {si running or connections to the data available via the credentials provi **Action**: The database instance {s started on {server} and the proper be provided for the completion of operation.
122506	ERROR	Unable to determine Oracle dbspaces and logfiles for "%s" on "%s".	**Cause**: A query to determine the l required tablespaces, logfiles and files failed. This may have been ca internal database error. **Action**: Check the adjacent log m further details and related errors. log (alert.log) and related trace log additional information and correct problem(s).
122507	ERROR	Unknown chunk type found for "%s" on "%s".	**Cause**: The specified tablespace, required database file is not one o

Code	Severity	Message	Cause/Action
			supported file or character device **Action**: The specified file {databa reference an existing character de Consult the Oracle installation doc recreate the specified file {databa supported file or character device
122508	ERROR	DB Chunk "%s" for "%s" on "%s" does not reside on a shared file system.	**Cause**: The specified tablespace, required database file {database_ reside on a file system that is shar systems in the cluster. **Action**: Use the LifeKeeper UI or to verify that communication paths properly created. Use "rpm" to ver necessary Application Recovery K protection have been installed. Ve in fact, not on shared storage, and a shared storage device.
122510	ERROR	File system create failed for "%s" on "%s". Reason	**Cause**: LifeKeeper was unable to resource {filesystem} on the speci {server}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the operation.
122511	ERROR	%s	**Cause**: The message contains the "filesyshier" command. **Action**: Check the adjacent log m further details and related messag reported errors.
122513	ERROR	Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	**Cause**: LifeKeeper was unable to

Code	Severity	Message	Cause/Action
			dependency between the databas and the necessary child resource **Action**: Check adjacent log messa details and related messages. On have been corrected, it may be po the dependency between {tag} an manually.
122514	ERROR	Unable to "%s" on "%s" during resource create.	**Cause**: The Oracle Application Re unable to release the administrativ "rlslocks" command. **Action**: Check adjacent log messa details and related messages.
122516	ERROR	Raw device resource created failed for "%s" on "%s". Reason	**Cause**: LifeKeeper was unable to resource {raw device} on the spec {server}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the operation.
122519	ERROR	In-service attempted failed for tag "%s" on "%s".	**Cause**: The "perform_action" com {server} failed to start the databas service operation has failed. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the operation.
122521	ERROR	Create of app "%s" on "%s" failed with return code of "%d".	**Cause**: There was an error runnin "app_create" to create the interna **Action**: Check adjacent log messa

Code	Severity	Message	Cause/Action
			details and related messages. You reported errors before retrying the
122522	ERROR	Create of typ "%s" for app "%s" on "%s" failed with return code of "%d".	**Cause**: There was an error runnin "typ_create" to create the internal **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
122524	ERROR	Setting "resstate" for resource "%s" on "%s" failed with return code of "%d".	**Cause**: There was an error runnin "ins_setstate" to set the resource **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
122525	ERROR	The values specified for the target and the template servers are the same: "%s".	**Cause**: The value specified for the template servers for the "extend" the same. **Action**: You must specify the corr the {target server} and {template s {target server} is the server where extended.
122526	ERROR	Unable to locate the oratab file in "/etc" or in "%s" on "%s".	**Cause**: The oratab file was not fou or alternate locations on {server}. **Action**: Verify the oratab file exist permissions for the Oracle user. A is required to complete the "exten
122527	ERROR	Unable to retrieve the Oracle user on "%s".	**Cause**: An attempt to retrieve the {template server} during a "canext operation failed.

Code	Severity	Message	Cause/Action
			Action: The owner of the Oracle b valid user on {target server} and {t Correct the permissions and owne Oracle database installation and r
122528	ERROR	The Oracle user and/or group information for user "%s" does not exist on the server "%s".	**Cause**: LifeKeeper is unable to fir and/or group information for the O on the server {server}. **Action**: Verify the Oracle user {us specified {server}. If the user {use it should be created with the same servers in the cluster.
122529	ERROR	The id for user "%s" is not the same on template server "%s" and target server "%s".	**Cause**: The user id on the target s server} for the Oracle user {user} the value of the user {user} on the {template server}. **Action**: The user ids for the Oracl must match on all servers in the c mismatch should be corrected ma servers before retrying the "extend
122530	ERROR	The group id for user "%s" is not the same on template server "%s" and target server "%s".	**Cause**: The group id on the target server} for the Oracle user {user} the value of the user {user} on the {template server}. **Action**: The group ids for the Orac must match on all servers in the c id mismatch should be corrected r servers before retrying the "extend
122532	ERROR	No file system or raw devices found to extend for "%s" on "%s".	**Cause**: There were no dependent device resources found for the Ora {tag} on server {template server}.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messa details and related messages.
122533	WARN	A RAMDISK (%s) was detected in the ORACLE Database configuration for "%s" on "%s". LifeKeeper cannot protect RAMDISK. This RAMDISK resource will not be protected by LifeKeeper! ORACLE hierarchy creation will continue.	**Cause**: The specified tablespace, database file {database_file} was ramdisk. No protection is available resource in the current LifeKeeper **Action**: The ramdisk will not be pr must manually ensure that the rec file {database_file} will be availabl Oracle database operations.
122534	ERROR	Failed to initialize object instance for Oracle sid "%s" on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
122537	ERROR	Update of instance info field for "%s" on "%s" failed (%s).	**Cause**: There was an error while command "ins_setinfo" to update resource information field. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
122538	ERROR	Initial connect with query buffer to database "%s" on "%s" failed, testing output.	**Cause**: A connection attempt to th database {sid} to determine the da failed. **Action**: The connection attempt fa specified credentials. Check the a messages for further details and r Check the Oracle log (alert.log) ar logs (*.trc) for additional informatio

Code	Severity	Message	Cause/Action
			reported problem(s).
122542	ERROR	The "%s [%s]" attempt of the database "%s" appears to have failed on "%s".	**Cause**: The attempted Oracle acti method {action_method} for the da {sid} failed on the server {server}. **Action**: Check the adjacent log m further details and related errors. log (alert.log) and related trace log additional information and correct problem(s).
122543	ERROR	All attempts to "%s" database "%s" on "%s" failed	**Cause**: All efforts to perform the a the Oracle database {sid} on serve failed. **Action**: Check the adjacent log m further details and related errors. log (alert.log) and related trace log additional information and correct problem(s).
122544	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occu attempting to update the oratab en database {sid}. The error occurred to open the oratab file. **Action**: The oratab file entry for {s updated manually to turn off the a of the database at system boot.
122545	ERROR	Unable to locate the oratab file in "/etc" or in "%s" on "%s".	**Cause**: The oratab file was not fou or alternate locations on {server}. **Action**: Verify the oratab file exist permissions for the oracle user. A required to complete the "extend"

Code	Severity	Message	Cause/Action
122546	ERROR	Unable to open file "%s" on "%s" (%s).	**Cause**: The specified file {file} cou or accessed on the server {server} {error}. **Action**: Verify the existence and p specified file {file}. Check adjacen for further details and related erro correct any reported errors before operation.
122547	ERROR	(cleanUpPids):Forcefully killing hung pid(s):pid(s)="%s"	**Cause**: The process {pid} failed to request to terminate gracefully. Th will be forcefully terminated. **Action**: Use the command line to process {pid} has been terminated adjacent log messages for further related messages.
122548	ERROR	Unable to locate the DB utility (%s/%s) on this host.	**Cause**: The Oracle binaries and r utility {utility} located at {path/utilit on this server {server}. **Action**: Verify that the Oracle bina software utilities are installed and configured on the server {server}. binaries must be installed locally o located on shared storage availab the cluster.
122549	ERROR	Oracle internal error or non-standard Oracle configuration detected. Oracle User and/or Group set to "root".	**Cause**: The detected ownership o database installation resolves to th or root group. Ownership of the O by root is a non-standard configur **Action**: The owner of the Oracle b valid non-root user on {server}. Co permissions and ownership of the installation and retry the operation

Code	Severity	Message	Cause/Action
122550	ERROR	Initial inspection of "%s" failed, verifying failure or success of received output.	**Cause**: The previous Oracle query command {cmd} failed to return su **Action**: Check the adjacent log m further details and related errors. log (alert.log) and related trace log additional information and correct problem(s).
122551	ERROR	Logon failed with "%s" for "%s" on "%s". Please check username/password and privileges.	**Cause**: The logon with the creden for the database instance {sid} on failed. An invalid user {user} or pa specified. **Action**: Verify that the Oracle data and password {password} are inde addition, the Oracle database use sufficient privileges for the attemp
122552	ERROR	%s	**Cause**: The message contains the "sqlplus" command. **Action**: Check adjacent log messa details and related messages.
122553	ERROR	Unable to open file "%s" on "%s" (%s).	**Cause**: The specified file {file} cou or accessed on the server {server {error}. **Action**: Verify the existence and p specified file {file}. Check adjacen for further details and related erro correct any reported errors before operation.
122554	ERROR	The tag "%s" on "%s" is not an Oracle instance or it does not exist.	**Cause**: The specified tag {tag} on does not refer to an existing and v resource instance.

Code	Severity	Message	Cause/Action
			Action: Use the UI or "lcdstatus (existence of the resource tag {tag tag {tag} must be an Oracle resou use the command "ora-display."
122555	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected while attempting to upda user, password and database role resource instance. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
122557	ERROR	Update of user and password failed for "%s" on "%s".	**Cause**: A request to update the us for the resource tag {tag} failed. T credentials failed the initial validat attempt on server {server}. **Action**: Verify the correct credenti password} were specified for the a operation. Check adjacent log mes details and related messages. You reported errors before retrying the
122559	ERROR	Update of user and password failed for "%s" on "%s".	**Cause**: The update of the user an information for the resource tag {ta {server} failed. **Action**: Verify the correct credenti password} were specified for the a operation. Check adjacent log mes details and related messages. You reported errors before retrying the
122562	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The required Oracle execu not found on this server {server}.

Code	Severity	Message	Cause/Action
			Action: Verify that the Oracle bina software utilities are installed and configured on the server {server}. binaries must be installed locally o located on shared storage availab the cluster.
122566	ERROR	Unable to find Oracle home for "%s" on "%s".	**Cause**: The Oracle home directory does not appear to contain files ne proper operation of the Oracle ins **Action**: Verify using the command Oracle home directory {Oracle hor Oracle binaries, a valid spfile{sid}. file.
122567	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: There was an unexpected internal representation of the Orac protected. The specified internal I match the expected SID {sid}. **Action**: Verify the parameters are adjacent log messages for further related messages. You must corre errors before retrying the operatio
122568	ERROR	DB Processes are not running on "%s".	**Cause**: A process check for the O not find any processes running on **Action**: If local recovery is enable instance will be restarted locally. C log messages for further details ar messages.
122572	ERROR	Failed to create flag "%s" on "%s".	**Cause**: An unexpected error occu create a flag for controlling Oracle processing causing a failover to th

Code	Severity	Message	Cause/Action
			Action: Check the adjacent log m further details and related messag reported errors.
122574	ERROR	all attempts to shutdown the database %s failed on "%s".	**Cause**: The shutdown of the Orac during a local recovery process m because the maximum number of connections has been reached. **Action**: Check the Oracle logs for failures caused by the maximum r available connections being reach consider increasing the value. Add tunable LK_ORA_NICE to 1 to pre failures from causing a quickChec by a local recovery attempt.
122597	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected during pre-extend check **Action**: Check the adjacent log m further details and related messag reported errors before retrying the pre-extend.
122598	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac created while attempting to determ the Oracle home directory. **Action**: Check the adjacent log m further details and related messag reported errors before retrying the "create."
122599	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac

Code	Severity	Message	Cause/Action
			protected while attempting to look user on the template system. **Action**: Check the adjacent log m further details and related messag reported errors before retrying the extend.
122600	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected while attempting to disp properties. **Action**: Check the adjacent log m further details and related messag reported errors before retrying the resource properties.
122601	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected while attempting to chec database authorization. **Action**: Check the adjacent log m further details and related messag reported errors before retrying the
122603	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected while attempting to perfo on the Oracle resource instance. **Action**: Check that correct argume to the quickCheck command and a adjacent log messages for further related messages. Correct any rep before retrying the restore.

Code	Severity	Message	Cause/Action
122604	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected internal representation of the Orac protected while attempting to perf recovery on the Oracle resource i **Action**: Check that correct argum to the "recover" command, and als adjacent log messages for further related messages. Correct any rep before retrying the recover.
122606	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	**Cause**: The database instance {si running or connections to the data available via the credentials provi **Action**: The database instance {s started on {server} and the proper be provided for the completion of operation.
122607	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	**Cause**: The database instance {si running or connections to the data available via the credentials provi **Action**: The database instance {s started on {server} and the proper be provided for the completion of operation.
122608	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: The "remove" operation fa resource object instance required resource Out of Service. **Action**: Check that correct argum to the "remove" command and als adjacent log messages for further related messages. Correct any rep before retrying the restore.

Code	Severity	Message	Cause/Action
122609	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: The "restore" operation fa resource object instance required resource In Service. **Action**: Check that correct argum to the "restore" command and also adjacent log messages for further related messages. Correct any rep before retrying the "restore."
122610	ERROR	Unable to "%s" on "%s" during resource create.	**Cause**: The Oracle Application Re unable to create the administrative "getlocks" command during resou **Action**: Check adjacent log messa details and related messages. Co errors before retrying the create.
122611	ERROR	%s	**Cause**: The requested dependenc between the parent Oracle resour File System resource failed. **Action**: Check adjacent log messa details and related messages. Co errors before retrying the create o
122612	ERROR	%s	**Cause**: The requested dependenc between the parent Oracle resour Raw resource failed. **Action**: Check adjacent log messa details and related messages. Co errors before retrying the create o
122613	ERROR	%s	**Cause**: The requested dependenc between the parent Oracle resour Raw resource failed.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messa details and related messages. Co errors before retrying the create o
122614	ERROR	%s	**Cause**: The requested dependenc between the parent Oracle resour Listener resource failed. **Action**: Check adjacent log messa details and related messages. Co errors before retrying the create o
122616	ERROR	%s	**Cause**: The requested start up or Oracle database failed. **Action**: Check adjacent log messa details and related messages. Co errors before retrying the "restore" operation.
122618	ERROR	Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	**Cause**: LifeKeeper was unable to dependency between the databas and the necessary child resource **Action**: Check adjacent log messa details and related messages. On have been corrected, it may be po the dependency between {tag} an manually.
122619	ERROR	Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	**Cause**: LifeKeeper was unable to dependency between the databas and the necessary child resource **Action**: Check adjacent log messa details and related messages. On have been corrected, it may be po the dependency between {tag} an

Code	Severity	Message	Cause/Action
			manually.
122625	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The quickCheck process v the Oracle executable "sqlplus." **Action**: Check the Oracle configu check adjacent log messages for t related messages. Correct any rep
122626	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The remove process was Oracle executable "sqlplus." **Action**: Check the Oracle configu check adjacent log messages for t related messages. Correct any rep
122627	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The restore process was u Oracle executable "sqlplus." **Action**: Check the Oracle configu check adjacent log messages for t related messages. Correct any rep
122628	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The recover process was Oracle executable "sqlplus." **Action**: Check the Oracle configu check adjacent log messages for t related messages. Correct any rep
122632	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: During a remove, the resc {sid} passed to the remove proces internal resource instance informa **Action**: Check adjacent log messa details and related messages. Co errors.

Code	Severity	Message	Cause/Action
122633	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: During a restore, the reso {sid} passed to restore does not m resource instance information for **Action**: Check adjacent log messa details and related messages. Co errors.
122634	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: During resource recovery, instance {sid} passed to recovery internal resource instance informa **Action**: Check adjacent log messa details and related messages. Co errors.
122636	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: The create of the Oracle r {tag} failed on {server}. **Action**: Check adjacent log messa details and related messages. Co errors.
122638	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The create action for the C resource {tag} on server {server} f {sig} was received by the create p **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
122640	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occu create the Oracle resource instanc {server}. **Action**: Check adjacent log messa details and related messages. Co errors.

Code	Severity	Message	Cause/Action
122641	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occu create the Oracle resource instanc {server}. **Action**: Check adjacent log messa details and related messages. Co errors.
122642	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occu create the Oracle resource instanc {server}. **Action**: Check adjacent log messa details and related messages. Co errors.
122643	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check the adjacent log m further details and related messag reported errors.
122644	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: An unexpected error occu retrieve resource instance informa {server}. **Action**: Check adjacent log messa details and related messages. Co errors and retry the extend.
122645	ERROR	Cannot access canextend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to checks because it was unable to f "canextend" script on {server} for resource. **Action**: Check your LifeKeeper co

Code	Severity	Message	Cause/Action
122646	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: An unexpected error occu retrieve resource instance informa {server}. **Action**: Check adjacent log messa details and related messages. Co errors and retry the extend.
122647	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to resource {resource} on {server}. **Action**: Check adjacent log messa details and related messages. Co errors.
122648	ERROR	Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s"	**Cause**: During the database resou resource instance was found usin and/or {id} but with a different reso and type. **Action**: Resource IDs must be un resource instance with the ID mat resource instance must be remove
122649	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occu create the Oracle resource instan {server}. **Action**: Check adjacent log messa details and related messages. Co errors.
122650	ERROR	Cannot access extend script "%s" on server "%s"	**Cause**: The request to extend the resource {resource} to {server} fai was unable to the find the script {e {server} for a dependent child reso **Action**: Check your LifeKeeper co

Code	Severity	Message	Cause/Action
122651	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: The request to extend the resouce {resource} to {server} fail error attempting to extend a depe resource. **Action**: Check the adjacent log m further details and related messag reported errors.
122654	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The health check for the q terminated because the quickChe received a signal. This is most like quickCheck process requiring mor complete than was allotted. **Action**: The health check time for resource is controlled by the tunab ORACLE_QUICKCHECK_TIMEO value greater than 45 seconds to for the health check process to co
122655	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The request to take datab Service" was terminated because process received a signal. This is caused by the remove process rec to complete than was allotted. **Action**: The remove time for an O controlled by the tunable value ORACLE_REMOVE_TIMEOUT. S a value greater than 240 seconds time for the remove process to co
122659	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The request to place data Service" was terminated because process received a signal. This is caused by the restore process rec to complete than was allotted.

Code	Severity	Message	Cause/Action
			Action: The restore time for an O controlled by the tunable value ORACLE_RESTORE_TIMEOUT. a value greater than 240 seconds time for the restore process to co
122663	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The recovery of the failed terminated because the recovery a signal. This is most likely cause process requiring more time to co allotted. **Action**: The recovery time for an controlled by the tunable values ORACLE_RESTORE_TIMEOUT a ORACLE_REMOVE_TIMEOUT. S these to a value greater than 240 more time for a recovery to compl
122670	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occu attempting to update the oratab e database {sid}. The error occurre to open the temporary file used in process. **Action**: The oratab file entry for {s updated manually to turn off the a of the database at system boot.
122671	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occu attempting to update the oratab e database {sid}. The error occurre to close the temporary file used in process. **Action**: The oratab file entry for {s updated manually to turn off the a of the database at system boot.

Code	Severity	Message	Cause/Action
122672	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occu attempting to update the oratab en database {sid}. The error occurred to rename the temporary file back **Action**: The oratab file entry for {s updated manually to turn off the a of the database at system boot.
122673	ERROR	Unable to log messages queued while running as oracle user %s on %s. Reason: $!	**Cause**: An unexpected error {reas while attempting to add messages These messages were generated the Oracle user. **Action**: Review the reason for the corrective action.
122674	ERROR	Unable to open %s Reason: %s.	**Cause**: An unexpected error occu attempting to open a connection to database and run the database {c **Action**: Check adjacent log messa details and related messages. Add the Oracle log (alert.log) and relat (*.trc) for additional information ar reported problems.
122680	ERROR	Unable to find Oracle home for "%s" on "%s".	
122681	ERROR	Failed to create object instance for Oracle on "%s".	
122682	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122683	ERROR	Backup node %s is unreachable; abort username/password changes.	
122684	ERROR	The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper.	
122685	ERROR	The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper.	
122686	ERROR	The quickCheck of %s has timed out on server %s. The default TIMEOUT is 45 seconds. To increase the TIMEOUT, set ORACLE_QUICKCHECK_TIMEOUT in /etc/default/	

Code	Severity	Message	Cause/Action
		LifeKeeper.	
122687	ERROR	Usage: %s %s	
123006	FATAL	Unknown version %s of IP address	**Cause**: The IP address does not a for either IPv4 or IPv6. **Action**: Provide a valid IP address
123008	ERROR	No pinglist found for %s.	**Cause**: Problem while opening the IP address. **Action**: Make sure you have provi this IP address.
123009	ERROR	List ping test failed for virtual IP %s	**Cause**: No response was received addresses in the ping list. **Action**: Check network connectivi and the systems on which the IPs reside.
123013	ERROR	Link check failed for virtual IP %s on interface %s.	**Cause**: The requested interface is 'NO-CARRIER' indicating that no the phycial layer connection. **Action**: Check the physical conne interface and bring the physical la
123015	ERROR	Link check failed for virtual IP %s on interface %s.	**Cause**: The requested interface is interface, and one of the slaves is 'NO-CARRIER' indicating that no the phycial layer connection. **Action**: Check the physical conne slave interface and bring the phys
123024	ERROR	IP address seems to still exist somewhere else.	**Cause**: The IP address appears to

Code	Severity	Message	Cause/Action
			elsewhere on the network. **Action**: Either select a different IP or locate and disable the current u address.
123037	ERROR	must specify machine name containing primary hierarchy	**Cause**: Not enough arguments we crelPhier. **Action**: Supply all of the needed a crelPhier.
123038	ERROR	must specify IP resource name	**Cause**: Not enough arguments we crelPhier. **Action**: Supply all of the needed a crelPhier.
123039	ERROR	must specify primary IP Resource tag	**Cause**: The argument specifying t Resource tag was missing from th command. **Action**: Supply all of the needed a
123042	ERROR	An unknown error has occurred in utility validmask on machine %s.	**Cause**: There was an unexpected "validmask" utility. **Action**: Check adjacent log messa details.
123045	ERROR	An unknown error has occurred in utility getlocks.	**Cause**: There was an unexpected "getlocks" utility. **Action**: Check adjacent log messa details.

Code	Severity	Message	Cause/Action
123053	ERROR	Cannot resolve hostname %s	**Cause**: A hostname was provided address, but the system was unab name to an IP address. **Action**: Check the correctness of verify that name resolution (DNS working correctly and returns the hostname.
123055	ERROR	An unknown error has occurred in utility %s on machine %s.	**Cause**: There was a failure while resource. **Action**: Check the logs for related resolve the reported problem.
123056	ERROR	create ip hierarchy failure: perform_action failed	**Cause**: Unexpected error trying to address during creation. **Action**: Check adjacent log messa details.
123059	ERROR	Resource already exists on machine %s	**Cause**: Attempted to create an IP already exists. **Action**: Reuse the existing resour remove the IP address if it exists IP address.
123060	ERROR	ins_create failed on machine %s	**Cause**: An unexpected failure occ creating an IP resource. **Action**: Check adjacent log messa details.
123064	ERROR	An unknown error has occurred in utility %s on machine %s.	**Cause**: There was a failure while dependency for the IP resource.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
123066	ERROR	An error occurred during creation of LifeKeeper application=comm on %s.	**Cause**: A failure occured while ca **Action**: Check the logs for related resolve the reported problem.
123068	ERROR	An error occurred during creation of LifeKeeper resource type=ip on %s.	**Cause**: A failure occured while ca **Action**: Check the logs for related resolve the reported problem.
123089	ERROR	Link check failed for virtual IP %s on interface %s.	
123091	ERROR	the link for interface %s is down	**Cause**: The requested interface is ‘NO-CARRIER’ indicating that no the phycial layer connection. **Action**: Check the physical conne interface and bring the physical la
123093	ERROR	the ping list check failed	**Cause**: No response was received addresses in the ping list. **Action**: Check network connectivi and the systems on which the IPs reside.
123095	ERROR	broadcast ping failed	**Cause**: No replies were received f ping. **Action**: Verify that at least one ho will respond to broadcast pings. V is on the correct network interface a pinglist instead of a broadcast p

Code	Severity	Message	Cause/Action
123096	ERROR	$msg	**Cause**: The broadcast ping used t viability of the virtual IP failed. **Action**: Please ensure that the pi resource is properly configured in panel or that broadcast ping chec adding NOBCASTPING=1 to the / LifeKeeper configuration file.
123097	ERROR	exec_list_ping(): broadcast ping failed.	**Cause**: The broadcast ping used t viability of the virtual IP failed. **Action**: Ensure that the ping list fo properly configured in the properti broadcast ping checking is disable NOBCASTPING=1 to the /etc/defa configuration file.
123299	ERROR	Unable to open %s. Reason %s	
123410	ERROR	Usage error OSUquickCheck	
123411	ERROR	OSUquickCheck: both tag and id name not specified	
123412	ERROR	resource $Tag not found on local server	
123414	ERROR	The link for network interface $IPObj->{‘device’} is down	
123415	ERROR	No pinglist found for $IPObj->{‘ipaddr’}	
123416	ERROR	List ping test failed for virtual IP $IPObj->{‘ipaddr’}	
124004	FATAL	resource tag name not specified	**Cause**: Invalid arguments were sp "quickCheck" operation. **Action**: Ensure that the correct ar passed.
124005	FATAL	resource id not specified	**Cause**: Invalid arguments were sp "quickCheck" operation. **Action**: Ensure that the correct ar passed.

Code	Severity	Message	Cause/Action
124007	FATAL	Failed to get resource information	**Cause**: The filesystem resource's contain the correct information. **Action**: Put the correct informatio info field or restore the system fro "lkbackup" to restore the original i
124008	ERROR	getId failed	**Cause**: The filesystem resource c underlying disk device. **Action**: Check adjacent log messa details. Verify that the resource hi and that all required storage kits a
124009	ERROR	LifeKeeper protected filesystem is in service but quickCheck detects the following error	**Cause**: The filesystem kit has fou wrong with the resource. **Action**: Check the messages imm this one for more details.
124010	ERROR	\"$id\" is not mounted	**Cause**: The filesystem resource is mounted. **Action**: No action is required. Allo to remount the resource.
124011	ERROR	\"$id\" is mounted but with the incorrect mount options (current mount option list: $mntopts, expected mount option list: $infoopts	**Cause**: The filesystem resource is incorrectly. **Action**: No action is required. Allo to remount the resource.
124012	ERROR	\"$id\" is mounted but on the wrong device (current mount device: $tmpdev, expected mount device: $dev	**Cause**: The filesystem resource h device mounted. **Action**: No action is required. Allo to remount the resource.

Code	Severity	Message	Cause/Action
124015	ERROR	LifeKeeper protected filesystem \"$tag\" ($id) is $percent% full ($blocksfree free blocks).	**Cause**: The filesystem is getting f **Action**: Remove or migrate data f filesystem.
124016	WARN	LifeKeeper protected filesystem \"$tag\" ($id) is $percent% full ($blocksfree free blocks).	**Cause**: The filesystem is getting f **Action**: Remove or migrate data f filesystem.
124020	FATAL	cannot find device information for filesystem $id	**Cause**: The filesystem resource c underlying disk device. **Action**: Check adjacent log messa details. Verify that the resource hi and that all required storage kits a
124029	ERROR	Failed to find child resource.	**Cause**: The filesystem resource c determine its underlying disk reso **Action**: Ensure that the resource correct.
124032	FATAL	Script has hung. Exiting.	**Cause**: Processes had files open filesystem that needed to be unmo those processes has taken too lor **Action**: If this error continues, try stop all software that may be usin to allow it to be unmounted. If the cannot be unmounted, contact Su
124042	ERROR	file system $fs failed unmount; will try again	**Cause**: Processes had files open filesystem that needed to be unmo multiple attempts to clear those pr **Action**: No action is required. Allo

Code	Severity	Message	Cause/Action
			continue.
124046	ERROR	file system $fsname failed unmount	**Cause**: A filesystem could not be **Action**: If this error continues, try stop all software that may be usin to allow it to be unmounted. If the cannot be unmounted, contact Su
124049	ERROR	Local recovery of resource has failed (err=$err	**Cause**: A filesystem resource has cannot be repaired locally. **Action**: No action is required. Allo be failed over to another system.
124051	WARN	getId failed, try count : $cnt/$try	
124052	ERROR	\"$id\" is mounted but filesystem is shutdown state.	
124103	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: This message should not normal circumstances. **Action**: Check adjacent log messa details.
124104	ERROR	$ERRMSG Required template machine name is null	**Cause**: Invalid arguments were sp canextend operation. **Action**: Ensure that the argument this error happens during normal contact Support.
124105	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Invalid arguments were sp canextend operation. **Action**: Ensure that the argument this error happens during normal contact Support.

Code	Severity	Message	Cause/Action
124106	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The resource's underlying cannot be determined. **Action**: Ensure the hierarchy is co template system before extending
124107	ERROR	$ERRMSG Resource \"$TemplateTagName\" must have one and only one device resource dependency	**Cause**: The resource has too mar devices in the hierarchy. **Action**: Ensure the hierarchy is co template system before extending
124108	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The resource cannot be fo template system. **Action**: Ensure the hierarchy is co template system before extending
124109	ERROR	$ERRMSG Can not access canextend for scsi/$DeviceResType resources on machine \"$TargetSysName\	**Cause**: The target system is missi components. **Action**: Ensure that the target sys correct kits installed and licensed.
124110	ERROR	$ERRMSG Either filesystem \"$TemplateLKId\" is not mounted on \"$TemplateSysName\" or filesystem is not shareable with \"$TargetSysName\	**Cause**: The filesystem isn't in serv template system or doesn't meet t for extending to the target system **Action**: Make sure the resource is template system and review the p documentation regarding the requ extending filesystems.
124111	ERROR	$ERRMSG File system type \"${FSType}\" is not supported by the kernel currently running on \"${TargetSysName}\	**Cause**: The filesystem's type cann on the target system due to lack o **Action**: Ensure that the target sys

Code	Severity	Message	Cause/Action
			kernel modules configured correct extending the resource.
124112	ERROR	must specify machine name containing primary hierarchy	**Cause**: Invalid arguments were sp creFShier operation. **Action**: If this error happens durin operation, please contact Support
124113	ERROR	must specify primary ROOT tag	**Cause**: Invalid arguments were sp creFShier operation. **Action**: If this error happens durin operation, please contact Support
124114	ERROR	must specify primary mount point	**Cause**: Invalid arguments were sp creFShier operation. **Action**: If this error happens durin operation, please contact Support
124115	ERROR	must specify primary switchback type	**Cause**: Invalid arguments were sp creFShier operation. **Action**: If this error happens durin operation, please contact Support
124118	ERROR	dep_remove failure on machine \""$PRIMACH"\" for parent \"$PRITAG\" and child \"$DEVTAG.\	**Cause**: Cleanup after a dependen **Action**: Check adjacent log messa details.
124119	ERROR	ins_remove failure on machine \""$PRIMACH"\" for \"$PRITAG.\	**Cause**: Cleanup after an instance **Action**: Check adjacent log messa details.

Code	Severity	Message	Cause/Action
124121	ERROR	ins_remove failure on machine \""$PRIMACH"\	**Cause**: Cleanup after a resource **Action**: Check adjacent log messa details.
124122	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: This message should not normal circumstances. **Action**: Check adjacent log messa details.
124123	ERROR	$ERRMSG Required template machine name is null	**Cause**: Invalid arguments were sp depstoextend operation. **Action**: Ensure the script is called error happens during normal oper contact Support.
124124	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Invalid arguments were sp depstoextend operation. **Action**: Ensure the script is called error happens during normal oper contact Support.
124125	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The resource was unable underlying disk resource. **Action**: Ensure the hierarchy and are correct before extending.
124126	ERROR	unextmgr failure on machine \""$PRIMACH"\	**Cause**: The cleanup, after a failed operation, failed. **Action**: Manually clean up any re and check adjacent log messages details.

Code	Severity	Message	Cause/Action
124128	ERROR	unextmgr failure on machine \""$PRIMACH"\" for \"$PRITAG.\	**Cause**: The cleanup, after a failed operation, failed. **Action**: Manually clean up any rer and check adjacent log messages details.
124129	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: This message should not normal circumstances. **Action**: Look for additional log me details.
124130	ERROR	$ERRMSG Required template machine name is null	**Cause**: Invalid arguments were sp extend operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124131	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Invalid arguments were sp extend operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124132	ERROR	$ERRMSG Required target mount point is null	**Cause**: Invalid arguments were sp extend operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124133	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The tag being extended d template system.

Code	Severity	Message	Cause/Action
			Action: Ensure that the hierarchy template system before extending
124134	ERROR	$ERRMSG Detected conflict in expected tag name \"$TargetTagName\" on target machine.	**Cause**: A resource already exists system with the same tag as the r extended. **Action**: Recreate one of the confli with a different tag.
124135	ERROR	$ERRMSG Resource \"$TemplateTagName\" does not have required device resource dependency or unable to access this resource on template machine.	**Cause**: The resource or its underl cannot be found on the template s **Action**: Ensure that the hierarchy template system before extending
124136	ERROR	$ERRMSG Resource \"$TemplateTagName\" must have one and only one device resource dependency	**Cause**: The resource has multiple devices in the hierarchy on the ter **Action**: Ensure the hierarchy is co extending and that the filesystem depends on a single disk resource
124137	ERROR	$ERRMSG Can not access extend for scsi/$DeviceResType resources on machine \"$TargetSysName\	**Cause**: The files required to suppo storage type aren't available on th **Action**: Ensure that the required k on the target system and licensed
124138	ERROR	$ERRMSG Unable to access target device resource \"$DeviceTagName\" on machine \"$TargetSysName\	**Cause**: The required underlying d doesn't exist on the target system **Action**: Check adjacent log messa details and ensure that the target configured for hosting the resourc extended.

Code	Severity	Message	Cause/Action
124139	ERROR	$ERRMSG Unable to access template \"/etc/mtab\" file	**Cause**: The target system cannot system's /etc/mtab file. **Action**: Check adjacent log messa details. Ensure that the /etc/mtab template system.
124140	ERROR	$ERRMSG Unable to find mount point entry \"$TemplateLKId\" in template \"/etc/mtab\" file. Is template resource in-service?	**Cause**: The resource doesn't app on the template system. **Action**: Make sure the resource is extending.
124141	ERROR	$ERRMSG Unable to find mount point \"$TemplateLKId\" mode on template machine	**Cause**: The details of the mount p template system cannot be determ **Action**: Ensure that the resource accessible on the template system extending.
124142	ERROR	$ERRMSG Unable to create or access mount point \"$TargetLKId\" on target machine	**Cause**: The mount point could not the target system. **Action**: Ensure that the mount po directory exists and is accessible system.
124143	ERROR	$ERRMSG Two or more conflicting entries found in /etc/ fstab on \"$TargetSysName\	**Cause**: The device or mount point mounted more than once on the ta **Action**: Ensure that the mount po on the target system before exten
124144	ERROR	$ERRMSG Failed to create resource instance on \"$TargetSysName\	**Cause**: The resource creation on failed. **Action**: Check adjacent log messa

Code	Severity	Message	Cause/Action
			details. Make sure to check the lo server.
124145	ERROR	$ERRMSG Failed to set resource instance state for \"$TargetTagName\" on \"$TargetSysName\	**Cause**: The source state could no OSU on the target system. **Action**: Check adjacent log messa details.
124146	ERROR	must specify machine name containing primary hierarchy	**Cause**: Invalid arguments were sp filesyshier operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124147	ERROR	must specify primary mount point	**Cause**: Invalid arguments were sp filesyshier operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124149	ERROR	create file system hierarchy failure	**Cause**: The process of finding the instance failed. **Action**: Check adjacent log messa details.
124150	ERROR	create file system hierarchy failure	**Cause**: The system failed to read **Action**: Check adjacent log messa details.
124151	ERROR	create file system hierarchy failure	**Cause**: The mount point could not /etc/mtab file.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messa details.
124152	ERROR	create file system hierarchy failure	**Cause**: The underlying disk resou found. **Action**: Check adjacent log messa details.
124153	ERROR	create file system hierarchy failure	**Cause**: Creating the filesystem re **Action**: Check adjacent log messa details.
124154	ERROR	create file system hierarchy failure	**Cause**: The info field for the resou updated. **Action**: Check adjacent log messa details.
124155	ERROR	create file system hierarchy failure	**Cause**: The switchback strategy c the resource. **Action**: Check adjacent log messa details.
124157	ERROR	create file system hierarchy failure \(conflicting entries in /etc/fstab\	**Cause**: The mount point could not the /etc/fstab file. **Action**: Check adjacent log messa details.
124160	ERROR	Unknown error in script filesysins, err=$err	**Cause**: This message should not normal circumstances. **Action**: Check adjacent log messa

Code	Severity	Message	Cause/Action
			details.
124161	ERROR	create filesys instance – existid – failure	**Cause**: This message should not normal circumstances. **Action**: Check adjacent log messa details.
124163	ERROR	create filesys instance – ins_list – failure	**Cause**: Checking for an existing r **Action**: Check adjacent log messa details.
124164	ERROR	create filesys instance – newtag – failure	**Cause**: The system failed to gene tag for the resource. **Action**: If this error happens durin operation, contact Support.
124168	ERROR	create filesys instance – ins_create – failure	**Cause**: The filesystem resource c created. **Action**: Check adjacent log messa details.
124169	ERROR	filesys instance – ins_setstate – failure	**Cause**: The new filesystem resou not be initialized. **Action**: Check adjacent log messa details.
124173	ERROR	create filesys instance – dep_create – failure	**Cause**: The resource's dependen with its underlying disk could not b **Action**: Check adjacent log messa details.

Code	Severity	Message	Cause/Action
124174	ERROR	machine not specified	**Cause**: Invalid arguments were sp rmenu_mp operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124175	ERROR	mount point not specified	**Cause**: Invalid arguments were sp rmenu_mp operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124177	ERROR	unexpected multiple matches found	**Cause**: One or more systems sho mount point used more than once **Action**: Verify filesystem devices and ensure that filesystems are or Look for additional log messages
124178	ERROR	machine name not specified	**Cause**: Invalid arguments were sp rmenump operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124180	ERROR	must specify filesystem type	**Cause**: Invalid arguments were sp validfstype operation. **Action**: Ensure the script is called error happens during normal opera contact Support.
124181	ERROR	mount point not specified	**Cause**: Invalid arguments were sp validmp operation.

Code	Severity	Message	Cause/Action
			Action: Ensure the script is called error happens during normal oper contact Support.
124182	ERROR	The mount point $MP is not an absolute path	**Cause**: A mount point was specifi absolute path (doesn't start with a **Action**: Specify a mount point as starting with a '/'.
124183	ERROR	$MP is already mounted on $MACH	**Cause**: The requested mount poir on the system. **Action**: Specify a mount point tha unmount it before retrying the ope
124184	ERROR	The mount point $MP is already protected by LifeKeeper on $MACH	**Cause**: The system is already pro specified mount point. **Action**: Choose a different mount already being protected.
124185	ERROR	The mount point $MP is not a directory on $MACH	**Cause**: The mount point refers to such as a regular file. **Action**: Choose a mount point tha directory.
124186	ERROR	The mount point directory $MP is not empty on $MACH	**Cause**: The specified mount point directory that isn't empty. **Action**: Choose a mount point tha remove the contents of the specifi before retrying the operation.
124187	ERROR	server name not specified	**Cause**: Invalid arguments were sp

Code	Severity	Message	Cause/Action
			valuepmp operation. **Action**: Ensure the script is called error happens during normal oper contact Support.
124188	ERROR	There are no mount points on server $MACH	**Cause**: There are no possible mo filesystem resource on the server. **Action**: Check adjacent log messa details.
124194	WARN	Please correct conflicting \"/etc/fstab\" entries on server $UNAME for: $FSDEV, $FSNAME	**Cause**: After deleting a filesystem entries in /etc/fstab need to me ma up. **Action**: Manually clean up the /et
124195	ERROR	getchildinfo found no $OKAPP child for $PTAG	**Cause**: The system could not find in the hierarchy. **Action**: Check adjacent log messa details and ensure that the hierarc before retrying the operation.
124196	ERROR	enablequotas – quotacheck may have failed for $FS_NAME	**Cause**: The quota operation failed **Action**: Check adjacent log messa details in both the lifekeeper log a messages.
124198	ERROR	enablequotas – quotaon failed to turn on quotas for $FS_NAME, reason	**Cause**: The quota operation failed **Action**: Check adjacent log messa details in both the lifekeeper log a messages.

Code	Severity	Message	Cause/Action
124200	ERROR	The device node $dev was not found or did not appear in the udev create time limit of $delay seconds	**Cause**: A device node (/dev/…) w udev. This may indicate an issue v the server's connection to the stor **Action**: Check adjacent log messa details in both the lifekeeper log a messages.
124201	WARN	Device $device not found. Will retry wait to see if it appears.	**Cause**: This can happen under nc while udev creates device node en This message should not happen **Action**: Check adjacent log messa details in both the lifekeeper log a messages.
124202	ERROR	Command \"$commandwithargs\" failed. Retrying ….	**Cause**: The given command failed failed temporarily. This failure may normal operations but should not **Action**: Check adjacent log messa details if this message continues.
124204	WARN	cannot make file system $FSNAME mount point	**Cause**: The mount point directory created. **Action**: Ensure that the mount po created. This may be due to filesy mount options, etc.
124207	ERROR	\"fsck\"ing file system $FSNAME failed, trying alternative superblock	**Cause**: This message indicates th filesystem check failed. This mess for ext2 filesystems or other filesy alternative superblock location is u **Action**: Check adjacent log messa details.

Code	Severity	Message	Cause/Action
124209	ERROR	\"fsck\"ing file system $FSNAME with alternative superblock failed	**Cause**: This indicates that an ext2 other filesystem where an alternat location is used) check failed with superblock location. **Action**: Check adjacent log messa details and instructions on how to
124210	WARN	POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME	**Cause**: A filesystem was put in se over when it was out of sync with **Action**: Check adjacent log messa details and review the product doc information on how to bring the file safely.
124211	ERROR	Reason for fsck failure ($retval): $ret	**Cause**: This log message is part o messages and gives the actual ex fsck process. **Action**: Check adjacent log messa details and instructions on how to
124212	ERROR	\"fsck\" of file system $FSNAME failed	**Cause**: The check of the filesyste usually due to the filesystem havi **Action**: Check adjacent log messa details. Review the product docum instructions on how to handle poss corruption.
124213	WARN	POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME	**Cause**: The system or user tried t service a filesystem that may be c can happen if a filesystem is switc when it was out of sync with its mi **Action**: Check adjacent log messa details and review the product doc

Code	Severity	Message	Cause/Action
			instructions on how to bring the re service safely.
124214	ERROR	Reason for fsck failure ($retval)	**Cause**: This message should follo message about a filesystem check the process exit code of the fsck p **Action**: Check adjacent log messa details.
124218	ERROR	File system $FSNAME was found to be already	**Cause**: This message is part of a messages. **Action**: Check adjacent log messa details.
124219	ERROR	mounted after initial mount attempt failed.	**Cause**: This message is part of a messages. This should not happe circumstances but may not be fata can be put in service. **Action**: Check adjacent log messa details in both the lifekeeper log a messages.
124220	ERROR	File system $FSNAME failed to mount.	**Cause**: The filesystem could not b **Action**: Check adjacent log messa details.
124221	WARN	Protected Filesystem $ID is full	**Cause**: The filesystem is full. **Action**: Remove unused data from migrate to a larger filesystem.
124222	WARN	Dependent Applications may be affected <>	**Cause**: This indicates that an ope

Code	Severity	Message	Cause/Action
			resource is likely to cause operati resources based on the resource **Action**: Make sure it's acceptable resources to be affected before co
124223	ERROR	Put \"$t\" Out-Of-Service Failed By Signal	**Cause**: This message should not normal circumstances. **Action**: Check adjacent log messa details.
124227	ERROR	Put \"$i\" Out-Of-Service Failed	**Cause**: The operation failed. **Action**: Check adjacent log messa details.
124230	ERROR	Put \"$t\" In-Service Failed By Signal	**Cause**: This message should not normal circumstances. **Action**: Check adjacent log messa details.
124231	ERROR	Put \"$t\" In-Service Failed	**Cause**: The operation failed. **Action**: Check adjacent log messa details.
124234	ERROR	Put \"$t\" In-Service Failed	**Cause**: The operation failed. **Action**: Check adjacent log messa details.
125102	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to {tag} on {server}.

Code	Severity	Message	Cause/Action
125103	ERROR	`printf ‘%s is not shareable with any machine.’ $DEV`	**Cause**: The device does not appe with any other systems. **Action**: Verify that the device is a servers in the cluster. Ensure that storage drivers and software are i configured properly.
125104	ERROR	`printf ‘Failed to create disk hierarchy for "%s" on "%s"’ $PRIMACH $DEV`	**Cause**: The creation of a resource physical disk failed. **Action**: Check adjacent log messa details and try to resolve the repo
125107	ERROR	`printf ‘Template resource "%s" on server "%s" does not exist’ $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to {tag} on {server}.
125114	ERROR	`printf ‘Template resource "%s" on server "%s" does not exist’ $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to {tag} on {server}.
125120	ERROR	`printf ‘Template resource "%s" on server "%s" does not exist’ $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to {tag} on {server}.
125123	ERROR	`printf ‘Cannot access depstoextend script "%s" on server "%s"’ $depstoextend $TargetSysName`	**Cause**: LifeKeeper was unable to checks on the resource hierarchy unable to find the script "DEPSTO {server}. **Action**: Check your LifeKeeper co
125126	ERROR	`printf ‘Template resource "%s" on server "%s" does not exist’ $ChildTag $TemplateSysName`	**Cause**: LifeKeeper was unable to {tag} on {server}.
125129	ERROR	`printf ‘Template resource "%s" on server "%s" does not exist’ $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to

Code	Severity	Message	Cause/Action
			{tag} on {server}.
125155	ERROR	SCSI $DEV failed to lock.	**Cause**: There was a problem lock device. **Action**: Check adjacent log messa details and try to resolve the repo
125164	ERROR	SCSI $INFO failed to unlock.	**Cause**: There was a problem unlo device. **Action**: Check adjacent log messa details and try to resolve the repo
125181	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTag $TemplateSysName`	**Cause**: LifeKeeper was unable to {tag} on {server}.
126105	ERROR	script not specified – $PTH is a directory	**Cause**: The specified script path i **Action**: Correct the path of the sc
126110	ERROR	script $PTH does not exist	**Cause**: The specified script path d **Action**: Correct the path of the sc
126115	ERROR	script $PTH is a zero length file	**Cause**: The specified script is an **Action**: Correct the script's file pa contents inside the script.
126117	ERROR	script $PTH is not executable	**Cause**: The specified script is not **Action**: Correct the script's file pa contents inside the script file and the proper execute permissions.

Code	Severity	Message	Cause/Action
126125	ERROR	required template machine name is null	**Cause**: The input template machi **Action**: Correct the input templat
126130	ERROR	required template resource tag name is null	**Cause**: The input template resou **Action**: Correct the input templat name.
126135	ERROR	Unable to generate a new tag	**Cause**: Failed to generate a new as the template tag name on the t the "newtag" script during the exte name is already existing. **Action**: Avoid using duplicate tag same node and check the log for
126140	ERROR	Unable to generate a new tag	**Cause**: Failed to generate a new tag name on the target node usin script during the extension. The ta existing. **Action**: Avoid using duplicate tag same node and check log for deta
126150	ERROR	unable to remote copy template \"$_lscript\" script file	**Cause**: Failed to remote copy tem The cause may be due to the non availability of template script file o or any transaction failure during "l **Action**: Check the existence/avail script and the connection to templ check the logs for related errors a the reported problem.
126155	ERROR	failed to create resource instance on \"$TargetSysName\	**Cause**: Failed to create resource "ins_create".

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
126160	ERROR	failed to set resource instance state for \"$TargetTagName\" on \"$TargetSysName\	**Cause**: Failed to set resource inst "ins_setstate" during GenApp reso **Action**: Check the logs for related resolve the reported problem.
126170	ERROR	getlocks failure	**Cause**: Failed to get the administr creating a resource hierarchy. **Action**: Check the logs for related resolve the reported problem.
126175	ERROR	instance create failed	**Cause**: Failed to create a GenApp "appins". **Action**: Check the logs for related resolve the reported problem.
126180	ERROR	unable to set state to OSU	**Cause**: Failed to set resource inst "ins_setstate" during GenApp reso **Action**: Check the logs for related resolve the reported problem.
126190	ERROR	resource restore has failed	**Cause**: Failed to restore GenApp **Action**: Check the logs for related resolve the reported problem.
126200	ERROR	create application hierarchy rlslocks failure	**Cause**: Failed to release lock afte resource created. **Action**: Check the logs for related

Code	Severity	Message	Cause/Action
			resolve the reported problem.
126210	ERROR	copy $ltype script $lscript failure	**Cause**: Failed to copy user provid appropriate GenApp directory duri creation. **Action**: Check the existence/avail provided script and the GenApp d Also check the logs for related err resolve the reported problem.
126215	ERROR	no $ltype script specified	**Cause**: Missing user defined scrip resource creation. **Action**: Check the input action scr resource creation again.
126220	ERROR	no machine name specified	**Cause**: Missing specified machine GenApp resource creation. Failed user script due to missing the inpu name. **Action**: Check the input for machi resource creation again.
126225	ERROR	no resource tag specified	**Cause**: Missing specified tag nam creation. **Action**: Check the input for source run resource creation again.
126230	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: Failed to extend GenApp unknown reason. **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
126235	ERROR	$ERRMSG Required template machine name is null	**Cause**: Missing the input for temp name during GenApp resource ex **Action**: Check the input for templa and do the resource extension aga
126240	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Missing the input for temp name during GenApp resource ex **Action**: Check the input for templa name and do the resource extensi
126245	ERROR	$ERRMSG Can not access extend for $AppType/$ResType resources on machine \"$TargetSysName\	**Cause**: Failed to locate "extend" s GenApp resource extension on ta **Action**: Check the existence/avail script and do GenApp resource ex
126250	ERROR	create application failure – ins_list failed	**Cause**: Failed when calling "ins_li GenApp resource creation. **Action**: Check the logs for related resolve the reported problem.
126255	ERROR	create application failure – unable to generate a new tag	**Cause**: Failed to generate a new t GenApp resource creation. **Action**: Avoid using duplicate tag same node. Also check the logs fo and try to resolve the reported pro
126270	ERROR	create application failure – ins_create failed	**Cause**: Failed using "ins_create" t instance. **Action**: Check the logs for related resolve the reported problem.

Code	Severity	Message	Cause/Action
126275	ERROR	create application failure – copy_actions failed	**Cause**: Failed using "copy_action specified template script file. **Action**: Check the existence/avail script. Also check the logs for rela to resolve the reported problem.
126290	ERROR	Unable to obtain tag for resource with id $ID	**Cause**: Failed to fetch GenApp re by input ID during recovery. **Action**: Check the correctness of existance/availability of GenApp r Also check the logs for related err resolve the reported problem.
126300	ERROR	generic application recover script for $TAG was not found or was not executable	**Cause**: Failed to locate the user d GenApp resource during recovery **Action**: Check the existence/avail defined script and do the GenApp again.
126310	ERROR	-t flag not specified	**Cause**: Missing the input for resou during GenApp resource restore. **Action**: Check the input for resou do resource restore again.
126315	ERROR	-i flag not specified	**Cause**: Missing the input for resou during GenApp resource restore. **Action**: Check the input for resou do resource restore again.
126327	ERROR	END timeout restore of \"$TAG\" (forcibly terminating	
126335	ERROR	restore script \"$LCDAS/$APP_RESTOREDIR/$TAG\" was not found or is not executable	**Cause**: Failed to locate the user d GenApp resource during restore.

Code	Severity	Message	Cause/Action
			Action: Check the existence/avail defined script and do the GenApp again.
126340	ERROR	-t flag not specified	**Cause**: Missing the input for resou during GenApp resource remove. **Action**: Check the input for resou do resource remove again.
126345	ERROR	-i flag not specified	**Cause**: Missing the input for resou during GenApp resource remove. **Action**: Check the input for resou do resource remove again.
126357	ERROR	END timeout remove of \"$TAG\" (forcibly terminating	
126365	ERROR	remove script \"$LCDAS/$APP_REMOVEDIR/$TAG\" was not found or was not executable	**Cause**: Failed to locate the user d GenApp resource during remove. **Action**: Check the existence/avail defined script and do the GenApp again.
126375	ERROR	Script has hung checking \"$tag\". Forcibly terminating.	**Cause**: The "quickCheck" Script w terminated for GenApp resource w {tag} due to a waiting time over th timeout. **Action**: Check the GenApp resou and restart quickChecking. Also ch related errors and try to resolve th problem.
126380	ERROR	Usage error: no tag specified	**Cause**: Missing the input for resou during GenApp resource quickChe

Code	Severity	Message	Cause/Action
			Action: Check the input for resou retry resource quickCheck.
126385	ERROR	Internal error: ins_list failed on $tag.	**Cause**: Failed using "ins_list" to fe resource information by input tag quickCheck process. **Action**: Correct the input tag nam quickCheck process again. Also c related errors and try to resolve th problem.
126390	FATAL	Failed to fork process to execute $userscript: $!	**Cause**: Failed to fork process to e defined "quickCheck" script during resource "quickCheck" process. **Action**: Check the existency/avail defined "quickCheck" script and d "quickCheck" process again.
126391	ERROR	quickCheck has failed for \"$tag\". Starting recovery.	**Cause**: The GenApp resource wit is determined to be failed by using health monitoring script – "quickC recovery process will be initiated. **Action**: Check the performance o resource when local recovery finis the logs for related errors and try reported problem.
126392	WARN	${convtag}_TIMEOUT: This parameter is old. This parameter will not be supported soon.	
126400	ERROR	-t flag not specified	**Cause**: Missing the input for resou during GenApp resource deletion **Action**: Check the input for resou do resource deletion process agai

Code	Severity	Message	Cause/Action
126405	ERROR	-i flag not specified	**Cause**: Missing the input for resou during GenApp resource deletion **Action**: Check the input for resou do resource deletion process agai
126478	ERROR	Failed to create tag \'$new_leaf\'.	**Cause**: The 'creapphier' utility fail specified tag. **Action**: Check /var/log/lifekeeper. messages from 'creapphier'.
126479	ERROR	Failed to extend tag \'$new_leaf\'.	**Action**: Check /var/log/lifekeeper. errors from extend manager.
126481	ERROR	Failed to create dependency on \'$sys\' for \'$new_leaf\' to \'$hier{$leaf}{$sys}{'Tag'}\'.	**Action**: Check /var/log/lifekeeper. the 'dep_create' function
126484	ERROR	Tag \'$root_tag\' is not in-service.	**Cause**: The specified tag is not in node in the cluster. **Action**: Bring the specified tag in- node in the cluster and re-run 'create_terminal_leaf'.
126485	ERROR	Tag \'$root_tag_1\' was not found, select the root tag for a hierarchy to add a terminal leaf resource.	**Cause**: The first tag passed to 'create_terminal_leaf' was not fou where the utility was run. **Action**: Verify the resource is in-s in the cluster and fully extended to
126486	ERROR	Unable to create leaf tag from \'$tag\'.	**Cause**: A unique terminal leaf tag created. A unique terminal leaf tag created after 100 tries.

Code	Severity	Message	Cause/Action
			Action: Check for multiple leaf tag in /var/log/lifekeeper.log that may problem.
126487	ERROR	Tag \'$root_tag_2\' was not found, select the root tag for a hierarchy to add a terminal leaf resource.	**Cause**: The second tag passed to 'create_terminal_leaf' was not fou where the utility was run. **Action**: Verify the resource is in-s in the cluster and fully extended to
126488	ERROR	Tag \'$root_tag_1' is not extended to 3 or more systems.	**Cause**: The specified tag is not ex more nodes. **Action**: Extend the specified tag t and retry 'create_terminal_leaf'.
126489	ERROR	$cmd does not support SDRS resources.	**Cause**: A multi-site configuration **Action**: none
126492	ERROR	Remove resource $tag failed.	
126494	ERROR	Delete dependency failed on \'$sys\' for \'$tag\' to \'$parent\'.	
126495	ERROR	New tag \'$new_tag\' was modified during create, expected \'$new_leaf\'.	
126496	ERROR	Tag \'$root_tag_1\' is not in-service.	
126497	ERROR	Tag \'$root_tag_2\' is not in-service.	
126498	ERROR	Tag \'$root_tag_1\' and \'$root_tag_2' are not extended to same systems.	
128005	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The quickCheck of {resou failed due to an operating system **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the

Code	Severity	Message	Cause/Action
128008	ERROR	Usage: quickCheck -t <tag name> -i <id>	**Cause**: Incorrect arguments have the dmmp device quickCheck com it from running. **Action**: Make sure all software co properly installed and at the corre the command and supply the corr -t <Resource Tag> and -i <Resou identifies the dmmp device resour quickChecked.
128010	ERROR	quickCheck for "%s" failed checks of underlying paths, initiate recovery. retry count=%s.	**Cause**: The dmmp kit failed to qui after {count} times of retries. A rec protected dmmp resource will be e **Action**: Check adjacent log messa details and related messages.
128021	ERROR	unable to find device for uuid "%s".	**Cause**: The device could not be fo during a restore operation. **Action**: Verify that the resource is properly. Rerun the command and correct device id that identifies the resource to be restored.
128025	ERROR	Device "%s" failed to unlock.	**Cause**: A non working {device} wa could not be unlocked during the r **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128026	ERROR	Device "%s" failed to lock.	**Cause**: The {device} could not be resotre. **Action**: Check adjacent log messa details and related messages. You

Code	Severity	Message	Cause/Action
			reported errors before retrying the
128031	ERROR	unable to find device for uuid "%s".	**Cause**: The device could not be fo during the remove operation. **Action**: Verify that the resource is properly. Rerun the command and correct device id that identifies the resource to be removed.
128034	ERROR	Device "%s" failed to unlock.	**Cause**: The {device} could not be the remove. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128036	ERROR	unable to load existing information for device with uuid "%s".	**Cause**: The device information co by unique id. **Action**: Make sure the resource is properly. Rerun the command and correct device id that identifies the resource.
128037	ERROR	unable to load existing information for device "%s".	**Cause**: The device information co by name. **Action**: Make sure the resource is properly. Rerun the command and correct device name that identifies resource.
128038	ERROR	unable to load existing information for device, no dev or uuid defined.	**Cause**: The device information co since neither a unique device id n device were defined.

Code	Severity	Message	Cause/Action
			Action: Make sure the resource is properly. Rerun the command and correct device id or name that iden device resource.
128041	ERROR	unable to load existing information for device with uuid "%s".	**Cause**: The device information co by unique id. **Action**: Make sure the resource is properly. Rerun the command and correct device id that identifies the resource.
128057	ERROR	All paths are failed on "%s".	**Cause**: LifeKeeper detected all pa protected dmmp device are in the **Action**: Check adjacent log messa details and related messages.
128058	ERROR	could not determine registrations for "%s"! All paths failed.	**Cause**: LifeKeeper could not dete registrations for protected dmmp { to the dmmp {device} are in the fa **Action**: Check adjacent log messa details and related messages.
128059	WARN	path "%s" no longer configured for "%s", remove from path list.	**Cause**: LifeKeeper detected listed protected {device} is not valid any remove it from the path list. **Action**: Check adjacent log messa details and related messages.
128060	WARN	registration failed on path "%s" for "%s".	**Cause**: LifeKeeper failed the regis for protected dmmp {device}. **Action**: Check adjacent log messa

Code	Severity	Message	Cause/Action
			details and related messages.
128062	ERROR	all paths failed for "%s".	**Cause**: LifeKeeper failed to verify protected dmmp {device}. **Action**: Check adjacent log messa details and related messages.
128072	ERROR	The daemon "%s" does not appear to be running and could not be restarted. Path failures may not be correctly handled without this daemon.	**Cause**: LifeKeeper failed to verify running and could not restart it. **Action**: Check adjacent log messa details and related messages.
128078	ERROR	"%s" resource type is not installed on "%s".	**Cause**: The Device Mapper Multip for dmmp device support is not ins system. **Action**: Install the steeleye-lkDMM Multipath Recovery Kit rpm on the
128083	ERROR	This script must be executed on "%s".	**Cause**: An incorrect system name supplied as an argument to the de used to create the dmmp device r **Action**: Make sure the cluster noc paths are properly configured. Sup system name to the devicehier scr must match the name of the syste command is run.
128084	ERROR	The device %s is not active.	**Cause**: LifeKeeper failed to find th {device} as a valid device on the s resource creation. **Action**: Check adjacent log messa details and related messages. Re

Code	Severity	Message	Cause/Action
			and supply the correct argument li Tag> and -i <Resource ID> that id device resource to be created.
128086	ERROR	Failed to create "%s" hierarchy.	**Cause**: LifeKeeper failed to create hierarchy for {device}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128088	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper failed to create with {tagname} on {server}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128090	ERROR	Failed to create dependency "%s"-"%s" on system "%s".	**Cause**: LifeKeeper failed to create {resource tag name} – {resource t {system} during creation. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128091	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper failed to create {system}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128101	ERROR	"%s" constructor requires a valid argument.	**Cause**: LifeKeeper failed to create dmmp resouce during construction **Action**: Rerun the command and

Code	Severity	Message	Cause/Action
			argument list: -t <Resource Tag> ID> that identifies the dmmp devic
128102	ERROR	Invalid tag "%s".	**Cause**: A resource instance could the given tag name. **Action**: Make sure the resource is properly. Rerun the command and correct argument list: -t <Resource <Resource ID> that identifies the resource.
128111	ERROR	Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations.	**Cause**: LifeKeeper failed to get th {device} with the message, "bad fi reservation in cdb". **Action**: Verify if the storage suppo reservations. Check adjacent log m further details and related messag correct any reported errors before operation.
128112	ERROR	Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations.	**Cause**: LifeKeeper failed to get th {device} with the message, "illega **Action**: Verify if the storage suppo reservations. Check adjacent log m further details and related messag correct any reported errors before operation.
128136	ERROR	A previous quickCheck with PID "%s" running for device "%s" has been terminated.	**Cause**: LifeKeeper detected that a quickCheck operation is still runni dmmp resource restore operation. terminated by LifeKeeper. **Action**: Check adjacent log messa details and related messages.

Code	Severity	Message	Cause/Action
128137	ERROR	SCSI reservation conflict on %s during LifeKeeper resource initialization. Manual intervention required.	**Cause**: LifeKeeper detected a SC conflict on {device} during dmmp r **Action**: Check adjacent log messa details and related messages. Ma and fix of the reservation conflict c required.
128138	ERROR	unable to clear registrations on %s.	**Cause**: LifeKeeper failed to clear registrations on {device}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128140	WARN	registration failed on path %s for %s.	**Cause**: LifeKeeper failed to make on {path} for {device}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128143	ERROR	reserve failed (%d) on %s.	**Cause**: LifeKeeper failed to make {resource} on {device}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128145	ERROR	The server ID "%s" returned by "%s" is not valid.	**Cause**: LifeKeeper failed to gener {ID}. **Action**: The ID used to register a of 1 to 12 Hex digits that uniquely server in the cluster. Check adjace for further details and related mes correct any reported errors before operation.

Code	Severity	Message	Cause/Action
128146	ERROR	device failure on %s. SYSTEM HALTED.	**Cause**: LifeKeeper detected failur will reboot the server. **Action**: Check adjacent log messa details and related messages.
128148	ERROR	device failure on %s. SYSTEM HALTED DISABLED.	**Cause**: LifeKeeper detected a fail The reboot was skipped due to Lif configuration. **Action**: Check adjacent log messa details and related messages. Tur configuration "SCSIHALT" to mak available for any detected device
128149	ERROR	device failure or SCSI Error on %s. SENDEVENT DISABLED.	**Cause**: LifeKeeper detected a fail The event generation was skipped LifeKeeper configuration. **Action**: Check adjacent log messa details and related messages. Tur configuration "SCSIEVENT" to ma available for any detected device
128150	ERROR	%s does not have EXCLUSIVE access to %s, halt system.	**Cause**: LifeKeeper detected a res for {device} on {server} and will re **Action**: Check adjacent log messa details and related messages.
128151	ERROR	%s does not have EXCLUSIVE access to %s, halt system DISABLED.	**Cause**: LifeKeeper detected a res for {device} on {server}. The reboo due to LifeKeeper configuration. **Action**: Check adjacent log messa details and related messages. Tur configuration "RESERVATIONCO the reboot available for any detect

Code	Severity	Message	Cause/Action
			conflicts.
128154	WARN	unable to flush buffers on %s.	**Cause**: LifeKeeper failed to flush {device} during dmmp resource re **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128157	WARN	%s utility not found, limited healthcheck for %s.	**Cause**: LifeKeeper failed to find " health check of {device}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128160	ERROR	%s failed to read %s.	**Cause**: LifeKeeper failed a disk I/ when using {utility}. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128163	ERROR	Registration ID "%s" for "%s" is not valid.	**Cause**: LifeKeeper failed to gener registration {ID} for {device}. **Action**: The ID used to register a of 4 Hex digits derived from the pa Check adjacent log messages for related messages. You must corre errors before retrying the operatio
128170	ERROR	Usage: canextend <Template system name> <Template tag name>	
128173	ERROR	Usage error OSUquickCheck	
128174	ERROR	both tag and id name not specified	
128175	ERROR	Failed multipath check.	

Code	Severity	Message	Cause/Action
128500	ERROR	Usage error	**Cause**: Incorrect arguments have the dmmp device restore comman from running. **Action**: Rerun the command and argument list: -t <Resource Tag> ID> that identifies the dmmp devic restored.
128504	ERROR	"%s" resource type is not installed on "%s".	**Cause**: The Device Mapper Multip for dmmp device support is not ins system. **Action**: Install the steeleye-lkDMM Multipath Recovery Kit rpm on the
128506	ERROR	Usage error	**Cause**: Incorrect arguments have the dmmp device devShared comm from running. **Action**: Rerun the command and argument list: <Template Resourc and <Template Resource Tag> th dmmp device resource to be creat
128507	FATAL	This script must be executed on "%s".	**Cause**: An incorrect system name supplied as an argument to the de used to create the dmmp device r **Action**: Supply the correct system devicehier script. The name must of the system on which the comma
128511	ERROR	Failed to get the ID for the device "%s". Hierarchy create failed.	**Cause**: The devicehier script used dmmp device resource was unable SCSI ID for the supplied device. **Action**: Check that the supplied d

Code	Severity	Message	Cause/Action
			and that is for a supported SCSI s
128512	ERROR	Failed to get the disk ID for the device "%s". Hierarchy create failed.	**Cause**: The devicehier script used dmmp disk resource was unable t SCSI ID for the supplied disk. **Action**: Check that the supplied d and that is for a supported SCSI s
128513	ERROR	Failed to create the underlying resource for device "%s". Hierarchy create failed.	**Cause**: The creation of the underl resource failed. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128515	ERROR	Error creating resource "%s" on server "%s"	**Cause**: The creation of the dmmp failed. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128517	ERROR	Failed to create dependency "%s"-"%s" on system "%s".	**Cause**: The parent child depende between the dmmp device and dm resources failed. **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the
128519	ERROR	Error creating resource "%s" on server "%s"	**Cause**: The attempt to bring the n dmmp device resource in service **Action**: Check adjacent log messa details and related messages. You reported errors before retrying the

Code	Severity	Message	Cause/Action
128521	ERROR	Either TEMPLATESYS or TEMPLATETAG argument missing	**Cause**: Incorrect arguments have the extend command for the dmm **Action**: Rerun the dmmp device r and supply the correct template sy names.
128540	ERROR	Usage error	**Cause**: Incorrect arguments have the dmmp device getId command the SCSI ID. **Action**: Rerun the command and argument list: -i <device path> or
128541	ERROR	Usage error	**Cause**: Incorrect arguments have the command used to delete the d resource. **Action**: Rerun the command and argument list: -t <dmmp device re
128543	ERROR	device node \"$dev\" does not exist.	**Cause**: The device node required dmmp device resource does not e allocated wait time in restore for u creation has been exceed. **Action**: Rerun the dmmp device r once udev has created the device
128544	ERROR	Usage error	**Cause**: Incorrect arguments have the remove command used to take device resource out of service. **Action**: Rerun the command and argument list: -t <dmmp device re
129100	FATAL	Failed to load instance from LifeKeeper.	**Cause**: An invalid resource tag or

Code	Severity	Message	Cause/Action
			Action: Check that the tag or ID is the command.
129103	FATAL	No resource matches tag \"$self->{'tag'}\".	**Cause**: An invalid resource tag wa **Action**: Check the tag and re-run
129104	FATAL	An error occurred setting LifeKeeper resource information	**Cause**: An internal error has occu LifeKeeper.
129110	ERROR	Could not get the Elastic Network Interface ID for $dev	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129111	ERROR	Failed to get Allocation ID of Elastic IP \"$elasticIp\".	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129113	ERROR	Failed to get my instance ID.	**Cause**: The EC2 instance metada **Action**: Check the Amazon conso operation.
129114	ERROR	Failed to get ENI ID.	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129116	ERROR	Failed to associate Elastic IP \"$self->{'EIP'}\" on \"$self->{'DEV'}\".	**Cause**: The EC2 API call failed, p network issue.

Code	Severity	Message	Cause/Action
			Action: Check the network and th console and retry the operation.
129118	WARN	$self->{‘EIP’} is not associated with any instance.	**Cause**: The Elastic IP is not asso instance. **Action**: LifeKeeper will try to fix th the EC2 API to associate the Elas adjacent log messages for more d
129119	WARN	$self->{‘EIP’} is associated with another instance.	**Cause**: The Elastic IP is associate instance. **Action**: LifeKeeper will try to fix th the EC2 API to associate the Elas adjacent log messages for more d
129120	ERROR	Failed to recover Elastic IP.	**Cause**: The EC2 API call failed to Elastic IP. **Action**: Check the network and th console and retry the operation.
129121	ERROR	Recovery process ended but Elastic IP is not associated with this instance. Please check AWS console.	**Cause**: The EC2 API call failed to Elastic IP. **Action**: Check the network and th console and retry the operation.
129122	ERROR	Error creating resource \"$target_tag\" with return code of \"$err\".	**Cause**: LifeKeeper was unable to resource instance on the server. **Action**: Check adjacent log messa details and related messages. Co errors.

Code	Severity	Message	Cause/Action
129123	ERROR	Failed to get ENI ID.	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129124	WARN	$self->{‘EIP’} is associated with another network interface.	**Cause**: The Elastic IP is associate instance, but the wrong ENI. **Action**: LifeKeeper will try to fix th the EC2 API to associate the Elas adjacent log messages for more d
129125	ERROR	Link check failed for interface \’$dev\’.	**Cause**: The requested interface is ‘NO-CARRIER’ indicating that no **Action**: Check the network interfa link up.
129126	ERROR	Link check failed for interface \’$dev\’. Reason: down slave.	**Cause**: The requested interface is ‘NO-CARRIER’ indicating that no **Action**: Check the network interfa link up.
129129	WARN	The link for network interface \’$self->{‘DEV’}\’ is down. Attempting to bring the link up.	**Cause**: The requested interface is ‘NO-CARRIER’ indicating that no **Action**: LifeKeeper will try to fix th bringing the link up and associatin with the interface. Check adjacent more details.
129130	ERROR	Failed to modify \"$opt_t\" to end pint URL \"$endpoint\".	
129137	ERROR	The link for network interface \’$self->{‘DEV’}\’ is still down.	**Cause**: LifeKeeper could not bring **Action**: Ensure the interface is en

Code	Severity	Message	Cause/Action
			Check adjacent log messages for
129139	WARN	The link for network interface \'$self->{'DEV'}\' is down.	**Cause**: The requested interface is 'NO-CARRIER' indicating that no **Action**: Check the network interfa link up.
129140	ERROR	Could not get ENI ID for $self->{IP}.	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129142	ERROR	Failed to update route table	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129143	ERROR	You must have exactly one IP address resource as the parent of the RouteTable EC2 resource. Please reconfigure your resource hierarchy.	**Cause**: The Route Table EC2 res one and only one IP resource as a **Action**: Repair the resource hiera necessary.
129144	ERROR	$func called with invalid timeout: $timeout	**Cause**: An invalid timeout value w the /etc/default/LifeKeeper file. **Action**: Verify all EC2_*_TIMEOU valid in /etc/default/LifeKeeper.
129145	ERROR	$func action timed out after $timeout seconds	**Cause**: The action did not comple timeout period. **Action**: Consider increasing the E

Code	Severity	Message	Cause/Action
			value for the given action (in /etc/ LifeKeeper).
129146	ERROR	failed to run $func with timeout: $@	**Cause**: This is an internal error.
129148	ERROR	Amazon describe-route-tables call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129150	ERROR	Elastic IP \"$elasticIp\" is associated with another instance.	**Cause**: The Elastic IP is not asso proper instance. **Action**: LifeKeeper will try to fix th the EC2 API to associate the Elas adjacent log messages for more d
129151	ERROR	Could not get the Association ID for Elastic IP \"$elasticIp\".	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129152	ERROR	Failed to disassociate Elastic IP \"$self->{'EIP'}\" on \"$self->{'DEV'}\".	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129153	ERROR	Failed to disassociate Elastic IP \"$elasticIp\", (err=%s)(output=%s	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.

Code	Severity	Message	Cause/Action
129154	ERROR	Amazon describe-addresses call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129155	ERROR	Amazon describe-address call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129157	ERROR	curl call failed (err=%s)(output=%s	**Cause**: The EC2 instance metada **Action**: Check the Amazon conso operation.
129159	ERROR	Amazon associate-address call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129160	ERROR	Amazon describe-addresses call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, p network issue. **Action**: Check the network and th console and retry the operation.
129161	ERROR	Error deleting resource \"$otherTag\" on \"$otherSys\" with return code of \"$err\".	**Cause**: LifeKeeper was unable to resource instance on the server. **Action**: Check adjacent log messa details and related messages. Co errors.
129162	ERROR	Could not getRouteTablesByIP	

Code	Severity	Message	Cause/Action
129163	ERROR	Could not getRouteTablesByIP	
129164	ERROR	[$SUBJECT event] mail returned $err	
129165	ERROR	COMMAND OUTPUT: cat /tmp/err$$	
129167	ERROR	snmptrap returned $err for Trap 180	
129168	ERROR	COMMAND OUTPUT: cat /tmp/err$$	
129170	ERROR	This resource is in the old format. Please update.	
129403	ERROR	END failed create of $TAG due to a $sig signal	**Cause**: The create process was ir signal.
129409	ERROR	The IP resource $IP_RES is not \"ISP\".	**Cause**: The IP resource is not in s **Action**: Bring the resource in serv operation.
129410	ERROR	Could not find IP resource $IP_RES	**Cause**: Ensure that the IP resourc the operation.
129412	ERROR	EC2 resource $ID is already protected	**Cause**: A resource with the specif exists. **Action**: Make sure to clean up an old resource before re-creating a r
129416	ERROR	Error creating resource \"$TAG\" with return code of \"$lcderror\".	**Cause**: LifeKeeper was unable to resource instance. **Action**: Check adjacent log messa details and related messages. Co errors.
129418	ERROR	Dependency creation between \"$IP_RES\" and \"$TAG\" failed with return code of \"$lcderror\".	**Cause**: LifeKeeper was unable to resource dependency. **Action**: Check adjacent log messa details and related messages. Co

Code	Severity	Message	Cause/Action
			errors.
129420	ERROR	In-service failed for tag \"$TAG\".	**Cause**: LifeKeeper could not bring instance into service. **Action**: Check adjacent log messa details and related messages. Co errors.
129423	ERROR	Could not get ENI ID for $dev.	
129425	ERROR	Failed to update route table	
129426	ERROR	In-service (dummy) failed for tag \"$TAG\".	
129800	ERROR	canextend checks failed for \"$self->{'tag'}\" (err=$ret	**Cause**: The pre-extend checks fai server. **Action**: Check adjacent log messa details and related messages. Co errors.
129801	ERROR	canextend checks failed for \"$self->{'tag'}\". EC2_HOME \"$self->{EC2_HOME}\" does not exist on $me.	
133106	ERROR	You must have exactly one IP address resource as the child of the route53 resource. Please reconfigure your resource hierarchy.	
133111	ERROR	Failed to init the object.	
133114	ERROR	Failed start Route53 resource $self->{Tag}	
133115	ERROR	Failed start Route53 resource $self->{Tag}	
133116	WARN	Could not Get A Record Value Address : $aAWSresult	
133117	WARN	aws call failed : $aAWSresult	
133118	WARN	aws call failed : $upsertResp	
133119	ERROR	Could not Update and Create A Record Set : $upsertResp	
133120	WARN	Could not get Route53 API batch request ID form UPSERT response XML data : $upsertResp	
133121	WARN	Could not Get A Record Value Address : $aAWSresult	
133122	WARN	aws call failed : $statusResult	
133123	ERROR	Faild check change batch request status : $statusResult	
133126	WARN	Failed to Route53 API response : $rc	

Code	Severity	Message	Cause/Action
133128	WARN	Failed to Route53 API response	
133129	ERROR	Route53 resource $self->{Tag} is stopped	
133130	ERROR	$self->{Tag} is in the old format. Please update.	
133601	ERROR	$usage	
133602	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" does not exist	
133608	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" does not exist	
133614	ERROR	Error creating resource \"$Tag\" on server \"$me\"	
133617	ERROR	END failed hierarchy extend of resource $Tag with return value of $ecode.	
133619	ERROR	$usage	
133620	ERROR	END failed extend of \"$Tag\" due to a \"$sig\" signal	
133700	ERROR	$usage	
133701	ERROR	END failed create of \"$Route53Tag\" due to a \"$sig\" signal	
133703	ERROR	Unable to getlocks on $me during resource create.	
133704	ERROR	The route53 on $Route53HostName is already protected by LifeKeeper.	
133706	ERROR	Error creating resource $Route53Tag. Error ($rc	
133708	ERROR	Dependency create failed between $Route53Tag and $Route53IPResTag. Error ($rc).	
133710	ERROR	In-service attempted failed for tag $Route53Tag.	
133813	ERROR	Usage: $usage	
133815	ERROR	The host name \"$hostname\" is too long.	
133816	ERROR	The host name \"$hostname\" is too short.	
133817	ERROR	The host name \"$hostname\" contains invalid character.	
133818	ERROR	The first character must be an alpha character.	
133819	ERROR	The last character must not be a minus sign or period.	
133826	ERROR	Host Name cannot be blank.	
134003	ERROR	catch a \"$sig\" signal	**Cause**: The “create” process was signal. **Action**: Check adjacent log messa
134004	ERROR	Unable to getlocks on $server during resource create. Error ($rc	**Cause**: Failed to get the administr creating a resource hierarchy.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related resolve the reported problem.
134005	ERROR	The service \"$serviceName\" is not supported on $server. Error ($rc	**Cause**: The service does not exis protected. **Action**: Input the appropriate serv
134006	ERROR	The service \"$serviceName\" is already protected on $server.	**Cause**: This service is already pro **Action**: Can not create a resource of this service.
134007	ERROR	Error creating resource $tag. Error ($rc	**Cause**: LifeKeeper was unable to resource instance. **Action**: Check adjacent log messa cause of the error.
134011	ERROR	In-service attempted failed for tag $tag.	**Cause**: Failed to restore QSP res **Action**: Check the log related to t you want to protect. Resolve the p
134015	ERROR	Unable to rlslocks on $server during resource create. Error ($rc	**Cause**: Failed to release lock afte created. **Action**: Check the logs for related resolve the reported problem.
134103	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" does not exist	**Cause**: The resource cannot be fo template server. **Action**: Ensure the hierarchy is co template server before extending.

Code	Severity	Message	Cause/Action
134104	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" is not QSP resource (app=$ins[1], res=$ins[2]	**Cause**: The template resource is **Action**: Expand to the same type template resource.
134105	ERROR	The service \"$service\" is not supported on $me. Error ($check	**Cause**: The service does not exis **Action**: Install the service on targ extending.
134106	ERROR	The service \"$service\" is already protected on $me.	**Cause**: There is already a resourc on target server. **Action**: Cannot create the resourc service.
134203	ERROR	catch a \"$sig\" signal	**Cause**: The “extend” process resc interrupted by a signal. **Action**: Check adjacent log messa
134204	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" does not exist	**Cause**: The resource cannot be fo template server. **Action**: Ensure the hierarchy is co template server before extending.
134208	ERROR	Error creating resource \"$tag\" on server \"$me\"	**Cause**: LifeKeeper was unable to resource instance on target serve **Action**: Check adjacent log messa cause of the error.
134401	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The “restore” process of th not terminate within the specified

Code	Severity	Message	Cause/Action
			Action: Check about the protecte the “restore” operation. Also chec related errors and try to resolve th problem.
134405	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a sy **Action**: Determine why fork fails.
134407	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service **Action**: It is an error to manually command with “start” option. Corr the error by reference to error mes
134501	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The “remove” process of t not terminate within the specified **Action**: Check about the protecte the “remove” operation. Also chec related errors and try to resolve th problem.
134505	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a sy **Action**: Determine why fork fails.
134507	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service **Action**: It is an error to manually command with “stop” option. Corre the error by reference to error mes
134601	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The "quickCheck" process terminated due to a waiting time o defined timeout.

Code	Severity	Message	Cause/Action
			Action: Check about the protecte check the logs for related errors a the reported problem.
134605	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a sy **Action**: Determine why fork fails.
134607	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service **Action**: It is an error to manually command with “status” option. Co the error by reference to error mes
134701	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The “recover” process of t not terminate within the specified **Action**: Check about the protecte check the logs for related errors a the reported problem.
134706	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a sy **Action**: Determine why fork fails.
134708	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service **Action**: It is an error to manually command with “start” option. Corr the error by reference to error mes
134803	ERROR	tag \"$tag\" does not exist on server \"$me\"	**Cause**: The specified tag does no internal error.
134804	ERROR	app type \"$ins[1]\" is not $app	**Cause**: The specified tag is not Q

Code	Severity	Message	Cause/Action
			is an internal error.
134805	ERROR	res type \"$ins[2]\" is not $res	**Cause**: The specified tag is not Q is an internal error.
134823	ERROR	tag \"$tag\" does not exist on server \"$me\"	**Cause**: The specified tag does no internal error.
134824	ERROR	app type \"$ins[1]\" is not $app	**Cause**: The specified tag is not Q is an internal error.
134825	ERROR	res type \"$ins[2]\" is not $res	**Cause**: The specified tag is not Q is an internal error.
134843	ERROR	tag \"$tag\" does not exist	**Cause**: The specified tag does no internal error.
134844	ERROR	app type \"$ins[1]\" is not $app	**Cause**: The specified tag is not Q is an internal error.
134845	ERROR	res type \"$ins[2]\" is not $res	**Cause**: The specified tag is not Q is an internal error.
135802	ERROR	object[%d] not iterated for %lld ms.	
135806	ERROR	qwk_config() failed.	
135807	ERROR	thread_initialize() failed.	
135808	ERROR	state_monitor_initialize() failed.	
135809	ERROR	state_monitor() failed.	
135810	ERROR	start_server() failed.	
135812	ERROR	‘fopen’ for %s failed: %m	
135813	ERROR	‘qwk_object_path=’ line cannot be found for %s.	
135814	ERROR	‘%s’ is unknown config.	

Code	Severity	Message	Cause/Action
135815	ERROR	configuration of hbeattime ‘%d’ is incorrect.	
135816	ERROR	configuration of numhbeats ‘%d’ is incorrect.	
135817	ERROR	configuration of timeout_multiplier ‘%d’ is incorrect.	
135818	ERROR	configuration of lcmhbeattime ‘%d’ is incorrect.	
135819	ERROR	configuration of lcmnumhbeats ‘%d’ is incorrect.	
135820	ERROR	configuration of qwk_object_type is incorrect.	
135821	ERROR	configuration of my_node is incorrect.	
135822	ERROR	configuration of number_of_node ‘%d’ is incorrect.	
135823	ERROR	configuration of number_of_object ‘%d’ is incorrect.	
135824	ERROR	configuration of object node is incorrect.	
135825	ERROR	configuration of object path is incorrect.	
135826	ERROR	my_node is not include in qwk_objects.	
135827	ERROR	‘open’ for %s failed: %m	
135828	ERROR	‘read’ for %s failed: %m	
135829	ERROR	‘popen’ for %s failed: %m	
135830	ERROR	‘open’ for %s failed: %m	
135831	ERROR	‘write’ for %s failed: %m	
135832	ERROR	‘popen’ for %s failed: %m	
135833	ERROR	(bug) buffer overflow	
135834	ERROR	(bug) data is corrupted	
135835	ERROR	‘signature=’ line cannot be found.	
135836	ERROR	signature ‘%s’ does not match.	
135837	ERROR	‘local_node=’ line cannot be found.	
135838	ERROR	local_node ‘%s’ does not match.	
135839	ERROR	‘time=’ line cannot be found.	
135840	ERROR	‘sequence=’ line cannot be found.	
135841	ERROR	sequence ‘%s’ scan failed.	
135842	WARN	‘node=’ line cannot be found. index=%d	
135843	ERROR	‘commstat=’ line cannot be found. index=%d	
135844	ERROR	‘checksum=’ line cannot be found.	
135845	ERROR	checksum ‘%s’ scan failed.	
135846	ERROR	checksum does not match.	
135849	ERROR	qwk object was not found.	
135851	ERROR	failed to read qwk object.	
135852	ERROR	failed to decode node_info.	

Code	Severity	Message	Cause/Action
135854	WARN	sequence backed down from %llu to %llu.	
135855	ERROR	‘malloc’ for %zu failed: %m	
135856	ERROR	thread_create() failed. index=%d	
135870	ERROR	(bug) data is corrupted	
135871	ERROR	format error in request.	
135872	ERROR	format error in quorum_verify request. lkevent cannot be found.	
135873	ERROR	format error in witness_verify request. lkevent cannot be found.	
135874	ERROR	format error in witness_verify request. target_node cannot be found.	
135875	ERROR	‘%s’ is unknown command.	
135877	ERROR	qwk_receive() did not receive full header. Close socket.	
135878	ERROR	Request is too long. Close socket.	
135879	ERROR	qwk_receive() did not receive full request. Close socket.	
135880	ERROR	do_request() failed.	
135881	ERROR	qwk_send() did not send full header. Close socket.	
135882	ERROR	qwk_send() did not send full response. Close socket.	
135884	ERROR	qwk_accept() failed.	
135885	ERROR	thread_create() failed.	
135886	ERROR	cannot create socket.	
135887	ERROR	‘bind’ failed: %m	
135888	ERROR	‘listen’ failed: %m	
135889	ERROR	create_sockets() failed.	
135890	ERROR	thread_create() failed.	
135891	ERROR	‘pthread_attr_init’ failed: %m	
135892	ERROR	‘pthread_attr_setstacksize’ failed: %m	
135893	ERROR	‘pthread_create’ failed: %m	
135894	ERROR	requst is too long. header.size=%zu	
135895	ERROR	qwk_send() failed for header.	
135896	ERROR	qwk_send() failed for request.	
135897	ERROR	qwk_send() failed for termination.	
135898	ERROR	qwk_receive() did not receive full header.	
135899	ERROR	Response buffer is not enough large. Server sent %zu bytes.	
135900	ERROR	qwk_receive() did not receive full response.	
135901	ERROR	‘%s’ is unknown command.	

Code	Severity	Message	Cause/Action
135903	ERROR	Cannot create socket.	
135904	ERROR	‘connect’ failed: %m	
135905	ERROR	request_send() failed.	
135906	ERROR	request_receive() failed.	
135907	ERROR	‘%s’ is unknown lkevent.	
135908	ERROR	‘%s’ is unknown qwktype.	
135909	ERROR	‘%s’ is unknown node state.	
135910	ERROR	‘%s’ is unknown quorum state.	
135911	ERROR	‘accept’ failed: %m	
135912	ERROR	You must install the LifeKeeper license key for Storage Quorum Witness Kit.	
135999	ERROR	-c $cmd	
136002	ERROR	Usage: $usage	**Cause**: Invalid arguments provide HANA create script. **Action**: Please provide appropriat the form: <Resource Tag> <SAP S Instance> [Switchback Type] [Virtu Tag]
136003	ERROR	END failed create of resource $tag on server $me with return value of $errcode.	**Cause**: Failure during SAP HANA creation. **Action**: Verify that SAP HANA Sy is fully configured and enabled on and secondary replication sites ar resource creation operation.
136005	ERROR	An unknown error has occurred in utility rlslocks on server $me. View the LifeKeeper logs for details and retry the operation.	**Cause**: Failure of the LifeKeeper r during SAP HANA resource creati **Action**: Resolve any issues found log file and reattempt the resource operation.
136006	ERROR	END failed create of resource $tag on server $me with signal $sig.	**Cause**: Failure during SAP HANA

Code	Severity	Message	Cause/Action
			due to a signal. **Action**: Review the LifeKeeper log
136008	ERROR	An unknown error has occurred in utility getlocks on server $me. View the LifeKeeper logs for details and retry the operation.	**Cause**: Failure of the LifeKeeper during SAP HANA resource creati **Action**: Resolve any issues found log file and reattempt the resource operation.
136009	ERROR	The SAP HANA product was not found in the directory $obj->{'UTIL_PATH'} on server $me.	**Cause**: Required SAP HANA bina located during resource creation. **Action**: Verify that SAP HANA is and configured on all servers in th
136010	ERROR	Failed to create resource as id $id already exists on system $me.	**Cause**: A LifeKeeper resource wit already exists on the system. **Action**: Check whether the SAP H already protected by LifeKeeper.
136011	ERROR	Failed to create new tag $tag for SAP HANA resource on $me.	**Cause**: The provided SAP HANA already in use by another LifeKee the LifeKeeper newtag utility failed tag. **Action**: Choose a different tag na HANA resource.
136012	ERROR	Failed creation of resource with $tag on system $me.	**Cause**: Failed to create the given resource in LifeKeeper. **Action**: Resolve any issues found log file and reattempt the resource operation.

Code	Severity	Message	Cause/Action
136014	ERROR	Failed to create resource dependency for parent $tag and child $virtual_ip_tag.	**Cause**: Failed to create a depend SAP HANA resource and its depe resource. **Action**: Resolve any issues found log file and reattempt the resource operation.
136015	ERROR	The info field for resource $tag could not be successfully generated using values [SID: $info_sid, Instance: $info_instance, Replication Mode: $info_repl_mode, Site Name: $info_site_name, Operation Mode: $info_oper_mode]. Please verify that SAP HANA System Replication is fully configured and enabled on both the primary and secondary systems before creating the SAP HANA resource.	**Cause**: An invalid value was foun the SAP HANA resource info field **Action**: Verify that SAP HANA is and configured and that SAP HAN Replication is fully configured and servers in the cluster.
136016	ERROR	The selected server $me is not the primary/source system for SAP HANA System Replication for the selected SID $sid and HDB instance $instance. Please select 'Cancel' and start this action on the primary/source HANA System Replication system.	**Cause**: Resource creation is initia secondary system in HANA Syste **Action**: Initiate the create action system in HANA System Replicati
136031	ERROR	END failed extend of resource $target_tag on server $me with return value of $err_code.	**Cause**: Failure during SAP HANA extension. **Action**: Resolve any issues found log file and reattempt the resource operation.
136033	ERROR	Usage: $usage	**Cause**: Invalid arguments provide HANA extend script. **Action**: Please provide appropriat the form: <Template System> <Te <Switchback Type> <Target Tag>
136035	ERROR	Template resource $template_tag on server $template_sys does not exist.	**Cause**: The template SAP HANA

Code	Severity	Message	Cause/Action
			extended does not exist on the ter **Action**: Verify that the SAP HANA being extended exists on the temp
136036	ERROR	Resource with matching id $target_id already exists on server $me for App $app_type and Type $res_type.	**Cause**: An SAP HANA resource w LifeKeeper ID already exists on th **Action**: Check whether the SAP H already protected by LifeKeeper o server.
136037	ERROR	Resource with matching tag $target_tag already exists on server $me for App $app_type and Type $res_type	**Cause**: An SAP HANA resource w LifeKeeper resource tag already e server. **Action**: Check whether the SAP H already protected by LifeKeeper o server.
136039	ERROR	Error creating resource $target_tag on system $me.	**Cause**: Failed to create an equiva resource on the target server. **Action**: Resolve any issues found log file and reattempt the resource operation.
136040	ERROR	The target tag ($target_tag) and template tag ($template_tag) must be the same.	**Cause**: Resource tag name used system is different than resource t secondary system. Both must be s **Action**: While resource creation u primary system tag and secondary
136045	ERROR	Cannot extend resource $template_tag to server $me.	**Cause**: The SAP HANA canexten that the resource cannot be exten target system.

Code	Severity	Message	Cause/Action
			Action: Resolve any issues found log file and reattempt the resource operation.
136047	ERROR	Usage: $usage	**Cause**: Invalid arguments provide HANA canextend script. **Action**: Please provide appropriat the form: <Template System> <Te
136048	ERROR	Resource $template_tag does not exist on server $template_sys.	**Cause**: The template SAP HANA extended does not exist on the te **Action**: Verify that the SAP HANA being extended exists on the temp
136049	ERROR	The system user $hana_user does not exist on server $me.	**Cause**: The SAP Administrative U HANA database does not exist on **Action**: Verify that SAP HANA is and configured.
136050	ERROR	The user id for user $hana_user ($template_uid) on template server $template_sys is not the same as user id ($uid) on target server $me.	**Cause**: The user ID for the SAP A User differs between the template servers. **Action**: Verify that SAP HANA is and configured on all servers in th
136051	ERROR	The group id for user $hana_user ($template_gid) on template server $template_sys is not the same as group id ($gid) on target server $me.	**Cause**: The group ID for the SAP User differs between the template servers. **Action**: Verify that SAP HANA is and configured on all servers in th

Code	Severity	Message	Cause/Action
136052	ERROR	The home directory for user $hana_user ($template_home) on template server $template_sys is not the same as home directory ($user_home) on target server $me.	**Cause**: The home directory for the Administrative User differs betwee and target servers. **Action**: Verify that SAP HANA is and configured on all servers in th
136053	ERROR	The SAP HANA instance $instance does not exist for $sid on server $me.	**Cause**: Installation directories for HANA database instance could no **Action**: Verify that SAP HANA is and configured on all servers in th
136055	ERROR	The SAP HANA site name $target_obj->{'site_name'} on server $me must be different from site name $template_obj->{'site_name'} on $template_obj->{'sys'}.	**Cause**: The SAP HANA System R name is the same on both the prim secondary servers. **Action**: Stop the SAP HANA data secondary server and use the hdb register the secondary replication different site name.
136056	ERROR	Unable to obtain SAP HANA System Replication parameters for database $instance on server $me. Please verify that SAP HANA System Replication is enabled and properly configured and that the database instance is running on all servers in the cluster.	**Cause**: SAP HANA System Replic could not be determined for the da given system. **Action**: Verify that SAP HANA Sy is enabled and properly configure database is running on all servers
136083	ERROR	Usage: $usage	**Cause**: Invalid arguments in the S script. **Action**: Provide both a valid HANA resource tag name and resource I -t <tag> -i <id> [-U]

Code	Severity	Message	Cause/Action
136160	ERROR	Unable to create SAP HANA object. The SID and instance for the SAP HANA database must be provided.	**Cause**: Either the SAP SID or the instance name were missing while an SAP HANA object. **Action**: Inspect the LifeKeeper log details.
136161	ERROR	Unable to create SAP HANA object. Resource system name and tag name must be provided.	**Cause**: Either the system name or name were missing while trying to HANA object. **Action**: Inspect the LifeKeeper log details.
136162	ERROR	Could not find any information regarding resource $tag on $sys.	**Cause**: Failed to obtain informatio HANA resource with the given tag **Action**: Verify that a SAP HANA r given tag exists on the given serv
136168	ERROR	Unable to check status of SAP Host Agent on server $self->{'sys'}. Command \"$curr_cmd\" returned exit code $ret.	**Cause**: Failed to determine the st Host Agent processes on the give **Action**: Inspect the SAP Host Age (e.g., dev_saphostexec) for more
136174	ERROR	Unable to create SAP HANA object. The SID, instance, or one of the SAP HANA system replication values is missing.	**Cause**: Either the SAP SID, the S instance name, or one of the SAP Replication values were missing w create an SAP HANA object. **Action**: Inspect the LifeKeeper log details.
136178	ERROR	Attempt to register server $node as the secondary SAP HANA System Replication site for database $self->{'instance'} failed with exit code $ret.	**Cause**: Failed to register the given secondary SAP HANA System Re the given database.

Code	Severity	Message	Cause/Action
			Action: Inspect the SAP HANA tra nameserver_<hostname>.xxxxx.x details.
136182	ERROR	Failed to $flg_action flag \"${HANA_FLAG_DATA_OUT_OF_SYNC}_$eqv_tag{$sys}\" on server $sys.	**Cause**: Failed to create or remove !HANA_DATA_OUT_OF_SYNC_< given server. **Action**: Inspect the LifeKeeper log details.
136190	ERROR	Failed to start SAP Host Agent processes on server $self->{'sys'}. Command \"$hostagent_cmd\" returned $ret.	**Cause**: Failed to start the SAP Ho processes on the given server. **Action**: Inspect the SAP Host Age (e.g., dev_saphostexec) for more
136191	ERROR	Failed to start SAP OS Collector process on server $self->{'sys'}. Command \"$oscol_cmd\" returned $ret.	**Cause**: Failed to start the SAP OS process on the given server. **Action**: Inspect the SAP OS Colle (e.g., dev_coll) for more details.
136193	ERROR	Takeover of SAP HANA System Replication for SAP HANA database $self->{'instance'} failed on server $node with exit code $ret.	**Cause**: Failed to register the given primary master for the given datab HANA System Replication. **Action**: Inspect the SAP HANA tra nameserver_<hostname>.xxxxx.x details.
136197	ERROR	Update of resource info field for $eqv_tag{$sys} on $sys failed with exit code $setinfo_ret. Current info: [$info]. Attempted new info: [$new_info].	**Cause**: Failed to update the info f resource on the given server. **Action**: Inspect the LifeKeeper log details.

Code	Severity	Message	Cause/Action
136202	EMERG	Failed to disable Autostart for SAP HANA instance $self->{'instance'} on server $sys with exit code $remexec_ret. Please manually set \"Autostart = 0\" in the instance profile $profile on $sys.	**Cause**: The value of the Autostart not be modified in the given HDB the given server. **Action**: Edit the HDB instance pro set "Autostart = 0".
136205	ERROR	Failed start of SAP Start Service for SAP HANA database $instance on server $sys.	**Cause**: Failed to start SAP Start S given SAP HANA database. **Action**: Inspect the SAP Start Ser (e.g., sapstartsrv.log) for more det
136208	ERROR	LifeKeeper was unable to determine the SAP HANA System Replication mode for database $rem_obj->{'instance'} on server $rem_obj->{'sys'} while attempting to identify the previous primary replication site. Please resolve the issue and bring the SAP HANA resource in-service on the system where the database should be registered as primary master.	**Cause**: Failed to determine the SA Replication mode on the given ser the previous primary replication si identified. **Action**: Inspect the LifeKeeper log details.
136210	ERROR	Failed start of SAP Start Service for SAP HANA database $rem_obj->{'instance'} on server $rem_obj->{'sys'}. Unable to stop the database on the server where it is currently registered as primary master.	**Cause**: Failed to start SAP Start S given SAP HANA database on the a result, the database could not be current primary SAP HANA Syster **Action**: Inspect the SAP Start Ser (e.g., sapstartsrv.log) for more det
136212	ERROR	Failed stop of SAP HANA database $rem_obj->{'instance'} on server $rem_obj->{'sys'} where it is currently registered as primary master.	**Cause**: Failed to stop the given SA database on the given server. **Action**: Inspect the SAP HANA tra LifeKeeper log file for more details
136217	ERROR	Failed start of SAP HANA database $instance on server $sys.	**Cause**: Failed to start the given S database on the given server.

Code	Severity	Message	Cause/Action
			Action: Inspect the SAP HANA tra LifeKeeper log file for more details
136220	ERROR	Unable to register $sys as a secondary SAP HANA System Replication site for database $instance. The host name of the current primary replication site was not provided.	**Cause**: The host name of the curr HANA System Replication site wa when attempting to register a seco site. **Action**: Inspect the LifeKeeper log details.
136233	EMERG	WARNING: A temporary communication failure has occurred between servers $self->{'sys'} and $rem_obj->{'sys'}. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please take one of the following resource hierarchies out of service: $self->{'tag'} on $self->{'sys'} or $rem_obj->{'tag'} on $rem_obj->{'sys'}. The server that the resource hierarchy is taken out of service on will become the secondary SAP HANA System Replication site.	**Cause**: A temporary communicati caused the given equivalent HANA both be brought in-service at the s respective host servers. **Action**: Take the entire HANA res out of service on the server which the secondary replication site. On has been stopped on that server, automatically register it as a seco site during the next quickCheck cy
136234	EMERG	WARNING: SAP HANA database $self->{'instance'} is running and registered as primary master on both $self->{'sys'} and $rem_obj->{'sys'}. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please leave resource $self->{'tag'} in-service on $self->{'sys'} and stop database $self->{'instance'} on $rem_obj->{'sys'} by running the command \'su – $rem_obj->{'sid_admin'} -c \"sapcontrol -nr $rem_obj->{'instance_number'} -function Stop\"\' on that server. Once stopped, it will become the secondary SAP HANA System Replication site.	**Cause**: The given SAP HANA data and registered as primary master servers concurrently. **Action**: Use the command provide to stop the database on the standb the database is stopped, LifeKeep automatically register the standby secondary replication site.
136238	ERROR	Unable to create SAP HANA object. Resource system name and tag name must be provided.	**Cause**: Either the server name or name were missing while trying to HANA object. **Action**: Inspect the LifeKeeper log

Code	Severity	Message	Cause/Action
			details.
136239	ERROR	Failed start of SAP Start Service for SAP HANA database $obj->{'instance'} on server $obj->{'sys'}. Unable to determine status of the database on $obj->{'sys'}.	**Cause**: Failed to start SAP Start S given SAP HANA database on the a result, the status of the databas determined. **Action**: Inspect the SAP Start Ser (e.g., sapstartsrv.log) for more det
136263	ERROR	Usage: $usage	**Cause**: Invalid arguments in the S resource restore script. **Action**: Please provide appropriat the form: -t <Resource Tag> -i <R
136265	ERROR	Error getting resource information for $tag on server $me.	**Cause**: Failed to obtain informatio SAP HANA resource on the given **Action**: Verify that a SAP HANA r given tag exists on the given serv
136266	ERROR	The resource $tag protecting SAP HANA database $instance is not in sync. To protect the data LifeKeeper will not restore the resource on $me. Please restore the resource on the previous source server to allow the resync to complete.	**Cause**: SAP HANA System Replic sync before attempting to bring th resource in-service on the backup **Action**: Bring the SAP HANA resc on the previous primary server an resynchronization to complete.
136275	ERROR	Failed to determine SAP HANA System Replication mode for database $instance on server $me.	**Cause**: Failed to determine the SA Replication mode for the given da given server. **Action**: Inspect the SAP HANA tra nameserver_<hostname>.xxxxx.x LifeKeeper log file for more details

Code	Severity	Message	Cause/Action
136351	ERROR	Usage: $usage	**Cause**: Invalid arguments in the S resource quickCheck script. **Action**: Please provide appropriat the form: -t <Resource Tag> -i <R
136353	ERROR	Error getting resource information for $tag.	**Cause**: Failed to obtain informatio SAP HANA resource. **Action**: Verify that a SAP HANA r given tag exists on the given serv
136354	EMERG	The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKeeper and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: The SAP HANA System R for the given database was modifi LifeKeeper. **Action**: Bring the SAP HANA reso on the server where it should be r primary master.
136363	EMERG	LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: Failed to determine the SA Replication mode on the given ser **Action**: Inspect the LifeKeeper log details.
136450	ERROR	Usage: $usage	**Cause**: Invalid arguments provide HANA remove script. **Action**: Please provide appropriat the form: <Template Tag> <Temp
136454	ERROR	Error getting resource information for $tag.	**Cause**: Failed to obtain informatio SAP HANA resource. **Action**: Verify that a SAP HANA r given tag exists on the given serv

Code	Severity	Message	Cause/Action
136456	ERROR	Failed start of SAP Start Service for SAP HANA database $instance on server $me.	**Cause**: Failed to start SAP Start S given SAP HANA database. **Action**: Inspect the SAP Start Ser (e.g., sapstartsrv.log) for more det
136550	ERROR	Usage: $usage	**Cause**: Invalid arguments in the H recover script. **Action**: Provide both a valid HANA resource tag name and resource I -d <tag> -n <id>
136555	ERROR	Error getting resource information for $tag on server $me.	**Cause**: Failed to obtain informatio HANA resource on the given serv **Action**: Verify that the server is or is running, and the HANA resourc
136556	EMERG	The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKeeper and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: The SAP HANA System R for the given database was modifi LifeKeeper. **Action**: Bring the SAP HANA resc on the server where it should be r primary master.
136558	EMERG	LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: The SAP HANA System R could not be determined for the gi the given server. **Action**: Inspect the SAP HANA tra LifeKeeper log file for more details
136559	ERROR	Resource $tag is no longer ISP on server $me. Exiting $cmd for $tag.	**Cause**: The given SAP HANA res ISP on the given server.

Code	Severity	Message	Cause/Action
			Action: Inspect the LifeKeeper log details.
136650	ERROR	Usage: $usage	**Cause**: Invalid arguments provide HANA hana_stop_all_dbs script. **Action**: Please provide appropriat the form: hana_stop_all_dbs -t <ta
136654	ERROR	Error getting resource information for $tag.	**Cause**: Failed to obtain informatio SAP HANA resource. **Action**: Verify that a SAP HANA r given tag exists on the given serv
136658	ERROR	Failed start of SAP Start Service for SAP HANA database $x->{'instance'} on server $x->{'sys'}. Could not determine status of SAP HANA DB on $x->{'sys'}.	**Cause**: Failed to start SAP Start S given SAP HANA database. **Action**: Inspect the SAP Start Ser (e.g., sapstartsrv.log) for more det
136661	ERROR	Failed stop of SAP HANA database $x->{'instance'} on server $x->{'sys'}.	**Cause**: Failed to stop the given SA database on the given server. **Action**: Inspect the SAP HANA tra LifeKeeper log file for more details
137000	ERROR	PowerShell is not installed.	
137001	ERROR	PowerCLI is not installed.	
137002	ERROR	A valid network interface was not found.	
137005	ERROR	Failed to attach VMDK.	
137010	ERROR	Failed to detach VMDK.	
137020	ERROR	Failed to execute VMDK status checker daemon.	
137030	ERROR	Disk not specified.	
137031	ERROR	Cannot get disk uuid for $Disk. Please check your ESXi settings.	
137032	ERROR	PowerCLI failed. %s	

Code	Severity	Message	Cause/Action
137034	ERROR	Cannot bring VMDK resource \"%s\" in service on server \"%s\".	
137037	ERROR	This system is not a VMware guest.	
137050	ERROR	Failed to connect to ESXi server $addr.	
137051	ERROR	There is no ESXi server connected.	
137055	ERROR	Cannot determine ESXi VM ID because multiple network interfaces were found with the MAC address $MAC_ADDR.	
137057	ERROR	Usable SCSI controller not found.	
137058	ERROR	Cannot find VMDK with ID $UUID.	
137059	ERROR	This guest has snapshots present.	
137060	ERROR	The VMDK with ID $UUID cannot be attached to this guest.	
137061	ERROR	The virtual storage controller has an incompatible sharing mode configured.	
137068	ERROR	The VMDK detection failed. Retry count exceeded.	
137070	ERROR	Connect failed.	
137071	ERROR	Get-LocalVM failed.	
137072	ERROR	The VMDK has been detached remotely. This server has lost ownership.	
137075	ERROR	Cannot find virtual SCSI controller $CONTROLLER.	
137076	ERROR	The virtual storage controller has an incompatible sharing mode configured.	
137077	ERROR	Cannot find VM with MAC address $MAC_ADDR.	
137078	ERROR	Cannot find VM with MAC address $MAC_ADDR.	
137101	WARN	Partition information not defined for %s on %s. Retry.	
137102	ERROR	Partition information not defined for %s on %s.	
137105	ERROR	Device not specified.	
137106	ERROR	Cannot get device uuid for $Device. Please check your ESXi settings.	
137107	ERROR	%s is not shareable with any machine.	
137111	ERROR	Failed to create dependency \"%s\"-\"%s\" on machine \"%s\".	
137112	ERROR	Cannot bring VMDKP resource \"%s\" in service on server \"%s\".	

8.1.1. DataKeeper Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
104002	FATAL	$msg	**Cause**: This message indicates an internal software error. **Action**: The stack trace indicates the source of the error.
104003	FATAL	$self->Val('Tag') . " is not an SDR resource"	**Cause**: A data replication action was attempted on a non data replication resource. **Action**: Check the logs for related errors and try to resolve the reported problem.
104010	ERROR	$self->{'md'}: bitmap merge failed, $action	**Cause**: The bitmap merge operation has failed. **Action**: The target server may have the mirror and/or protected filesystem mounted, or the bitmap file may be missing on the target. Check the target server.
104022	ERROR	$argv[1]: mdadm failed ($ret	**Cause**: The "mdadm" command has failed to add a device into the mirror. **Action**: This is usually a temporary condition.
104023	ERROR	$_	**Cause**: The message contains the output of the "mdadm" command.
104025	ERROR	failed to spawn monitor	**Cause**: The system failed to start the 'mdadm -F' monitor process. This should not happen under normal circumstances.

Code	Severity	Message	Cause/Action
			Action: Reboot the system to ensure that any potential conflicts are resolved.
104026	ERROR	cannot create $md	**Cause**: The mirror device could not be created. **Action**: Ensure the device is not already in use and that all other parameters for the mirror creation are correct.
104027	ERROR	$_	**Cause**: This message contains the “mdadm” command output.
104035	ERROR	Too many failures. Aborting resync of $md	**Cause**: The device was busy for an abnormally long period of time. **Action**: Reboot the system to be sure that the device is no longer busy.
104036	ERROR	Failed to start nbd-server on $target (error $port	**Cause**: The nbd-server process could not be started on the target server. **Action**: Ensure that the target disk device is available and that its Device ID has not changed.
104037	ERROR	Failed to start compression (error $port	**Cause**: The system was unable to start the ‘balance’ tunnel process or there was a network problem. **Action**: Ensure that the network is operating properly and that TCP ports in the range 10000-10512 are opened and unused. Ensure that the software is installed properly on all systems.
104038	ERROR	Failed to start nbd-client on $source (error $ret	**Cause**: The nbd-client process has failed to start on the source server.

Code	Severity	Message	Cause/Action
			Action: Look up the reported errno value and try to resolve the problem reported. For example, an errno value of 110 means “Connection timed out”, which may indicate a network or firewall problem.
104039	ERROR	Failed to add $nbd to $md on $source	**Cause**: This is usually a temporary condition. **Action**: If this error persists, reboot the system to resolve any potential conflicts.
104045	ERROR	failed to stop $self->{‘md’}	**Cause**: The mirror device could not be stopped. **Action**: Ensure that the device is not busy or mounted. Try running “mdadm —stop” manually to stop the device.
104048	WARN	failed to kill $proc, pid $pid	**Cause**: The process could not be signalled. This may indicate that the process has already died. **Action**: Ensure that the process in question is no longer running. If it is, then reboot the system to clear up the unkillable process.
104050	ERROR	Setting $name on $dest failed: $ret. Please try again.	**Cause**: The system failed to set a ‘mirrorinfo’ file setting. **Action**: Check the network and systems and retry the mirror setting operation.
104052	FATAL	Specified existing mount point “%s” is not mounted	**Cause**: The mount point became unmounted. **Action**: Ensure that the mount point is mounted and retry the operation.
104055	ERROR	Failed to set up temporary $type access to data for $self->{‘tag’}. Error: $ret	**Cause**: The filesystem or device was not available on the target server. The mirrored data will not be

Code	Severity	Message	Cause/Action
			available on the target server until the mirror is paused and resumed again. **Action**: Reboot the target server to resolve any potential conflicts.
104057	ERROR	Failed to undo temporary access for $self->{'tag'} on $self->{'sys'}. Error: $ret. Please verify that $fsid is not mounted on server $self->{'sys'}.	**Cause**: The filesystem could not be unmounted on the target server. **Action**: Ensure that the filesystem and device are not busy on the target server. Reboot the target server to resolve any potential conflicts.
104062	FATAL	Cannot find a device with unique ID "%s"	**Cause**: The target disk could not be identified. **Action**: Ensure that the appropriate storage recovery kits are installed on the target server. Ensure that the Device ID of the target disk has not changed.
104066	FATAL	Cannot get the hardware ID of device "%s"	**Cause**: A unique ID could not be found for the target disk device. **Action**: Ensure that the appropriate storage recovery kits are installed on the target server. Ensure that the Device ID of the target disk has not changed.
104067	FATAL	Asynchronous writes cannot be enabled without a bitmap file	**Cause**: An attempt was made to create a mirror with invalid parameters. **Action**: A bitmap file parameter must be specified or synchronous writes must be specified.
104068	FATAL	Failed to extend dependent resource %s to system %s. Error %s	**Cause**: The hierarchy extend operation failed. **Action**: Check the logs for related errors and try to

Code	Severity	Message	Cause/Action
			resolve the reported problem.
104070	FATAL	Unable to extend the mirror “%s” to system “%s”	**Cause**: The hierarchy extend operation failed. **Action**: Check the logs for related errors and try to resolve the reported problem.
104071	ERROR	Failed to restore target device resources on $target->{‘sys’} : $err	**Cause**: The in-service operation has failed on the target server. **Action**: Check the logs for related errors and try to resolve the reported problem.
104074	FATAL	Cannot get the hardware ID of device “%s”	**Cause**: There is no storage recovery kit that recognizes the underlying disk device that you are attempting to use for the mirror. **Action**: Make sure the appropriate storage recovery kits are installed. If necessary, place your device name in the /opt/LifeKeeper/subsys/scsi/resources/DEVNAME/device_pattern file.
104081	FATAL	Cannot make the %s filesystem on “%s” (%d)	**Cause**: The “mkfs” command failed. **Action**: Ensure that the disk device is writable and free of errors and that the filesystem tools for the selected filesystem are installed.
104082	FATAL	%s	**Cause**: This message contains the output of the “mkfs” command.
104083	FATAL	Cannot create filesys hierarchy “%s”	**Cause**: The resource creation failed. **Action**: Check the logs for related errors and try to resolve the reported problem.

Code	Severity	Message	Cause/Action
104086	ERROR	The “%s_data_corrupt” flag is set in “%s/subsys/scsi/resources/netraid/” on system “%s”. To avoid data corruption, LifeKeeper will not restore the resource.	**Cause**: The data corrupt flag file has been set as a precaution to prevent accidental data corruption. The mirror cannot be restored on this server until the file is removed. **Action**: If you are sure that the data is valid on the server in question, you can either: 1) remove the file and restore the mirror, or 2) force the mirror online using the LifeKeeper GUI or ‘mirror_action force’ command.
104092	ERROR	Mirror target resource movement to system %s : status %s	**Cause**: The hierarchy switchover operation has failed. **Action**: Check the logs for related errors and try to resolve the reported problem.
104099	ERROR	Unable to unextend the mirror for resource “%s” from system “%s”	**Cause**: The hierarchy unextend operation failed. **Action**: Reboot the target server to resolve any potential conflicts and retry the operation.
104106	ERROR	remote ‘bitmap -m’ command failed on $target->{‘sys’}: $ranges	**Cause**: The bitmap merge command failed on the target server. This may be caused by one of two things: 1) The bitmap file may be missing or corrupted, or 2) the mirror (md) device may be active on the target. **Action**: Make sure that the mirror and protected filesystem are not active on the target. If the target’s bitmap file is missing, pause and resume the mirror to recreate the bitmap file.
104107	ERROR	Asynchronous writes cannot be enabled without a bitmap file	**Cause**: Invalid parameters were specified for the mirror create operation.
104108	ERROR	Local Partition not available	**Cause**: Invalid parameters were specified for the

Code	Severity	Message	Cause/Action
			mirror create operation.
104109	ERROR	Cannot get the hardware ID of device “%s”	**Cause**: A unique ID could not be determined for the disk device. **Action**: Ensure that the appropriate storage recovery kits are installed on the server. Ensure that the Device ID of the disk has not changed.
104111	FATAL	Insufficient input parameters for “%s” creation	**Cause**: Invalid parameters were specified for the mirror create operation.
104112	FATAL	Insufficient input parameters for “%s” creation	**Cause**: Invalid parameters were specified for the mirror create operation.
104113	FATAL	Insufficient input parameters for “%s” creation	**Cause**: Invalid parameters were specified for the mirror create operation.
104114	FATAL	Insufficient input parameters for “%s” creation	**Cause**: Invalid parameters were specified for the mirror create operation.
104115	FATAL	Insufficient input parameters for “%s” creation	**Cause**: Invalid parameters were specified for the mirror create operation.
104117	FATAL	Insufficient input parameters for “%s” creation	**Cause**: Invalid parameters were specified for the mirror create operation.
104118	FATAL	Cannot unmount existing Mount Point “%s”	**Cause**: The mount point is busy. **Action**: Make sure the filesystem is not busy. Stop any processes or applications that may be accessing the filesystem.

Code	Severity	Message	Cause/Action
104119	FATAL	Invalid data replication resource type requested (“%s”	**Cause**: An invalid parameter was specified for the mirror create operation.
104124	EMERG	WARNING: A temporary communication failure has occurred between systems %s and %s. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should take one of the following resources out of service: %s on %s or %s on %s. The resource that is taken out of service will become the mirror target.	**Cause**: A temporary communication failure (split-brain scenario) has occurred between the source and target servers. **Action**: Perform the steps listed in the message text.
104125	ERROR	failed to start ‘$cmd $_[2] $user_args’ on ‘$_[3]‘	**Cause**: The specified command failed. **Action**: Check the logs for related errors and try to resolve the reported problem.
104126	ERROR	$_	**Cause**: This message contains the output of the command that was reported as failing in message 104125.
104128	FATAL	comm path/server not specified	**Cause**: The netraid.down script was called without specifying the communication path or the server name. This script is called internally so should always have the proper parameters. **Action**: Report this error to SIOS support.
104129	WARN		**Cause**: The replication connection for the mirror is down. **Action**: Check the network.

Code	Severity	Message	Cause/Action
104130	ERROR	Mirror resize failed on %s (%s). You must successfully complete this operation before using the mirror. Please try again.	**Cause**: The mirror resize operation has failed to update the mirror metadata on the listed system. **Action**: You must successfully complete the resize before using the mirror. Re-run mirror_resize (possibly using -f to force the operation if necessary).
104132	ERROR	The partition “%s” has an odd number of sectors and system “%s&qquot; is running kernel >= 4.12. Mirrors with this configuration will not work correctly with DataKeeper. Please see the SIOS product documentation for information on how to resize the mirror.	**Cause**: The partition or disk chosen for mirror creation has an odd number of disk sectors and will have to be resized to be used with DataKeeper. **Action**: Resize the partition using the ‘parted’ command or resize the disk (is possible) using platform (VMware, AWS) tools. Caution: data may be lost if this is not done carefully.
104136	ERROR	Extend failed.	**Cause**: The hierarchy extend operation failed. **Action**: Check the logs for related errors and try to resolve the reported problem.
104143	ERROR	Mirror resume was unsuccessful ($ret	**Cause**: The mirror could not be established. **Action**: Check the logs for related errors and try to resolve the reported problems.
104144	ERROR	Unable to stop the mirror access for $self->{‘md’} on system $self->{‘sys’}. Error: $ret. Use the \“mdadm —stop $self->{‘md’}\” command to manually stop the mirror.	**Cause**: The mirror device created on the target node when the mirror was paused could not be stopped. **Action**: Ensure that the device is not busy or mounted. Try running “mdadm —stop” manually to stop the device.
104145	WARN	Unable to dirty full bitmap. Setting fullsync flag.	**Cause**: A full resync could not be done by dirtying the full bitmap. The fullsync flag will be used

Code	Severity	Message	Cause/Action
			instead. This is a non-fatal error as a full synchronization will still be done. **Action**: None
104146	EMERG	WARNING: The target system %s for mirror %s has the target mirror %s currently active. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should reboot system %s.	**Cause**: The mirror is configured on the target system. **Action**: The target system should be rebooted. DataKeeper should then be able to resync the mirror.
104147	EMERG	WARNING: The target system %s for mirror %s has the target disk %s currently mounted. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should unmount %s on %s. A full resync will occur.	**Cause**: The mirror disk is mounted on the target system. **Action**: The mirror disk should be unmounted on the target system, in order to initiate a full mirror resync. A full resync is required because untracked changes have occurred on the disk.
104148	EMERG	The storage configuration for mirror “%s (%s)” does not have a unique identifier and may have potential for data corruption in some environments in certain circumstances. Please refer to the SIOS Product Documentation for details on DataKeeper storage configuration options.	**Cause**: The disk chosen for mirroring does not provide a UUID to the operating system. DataKeeper cannot mirror a disk without a UUID. **Action**: You may be able to create a GPT partition table on the disk to provide a UUID for the disk partitions. See DataKeeper for Linux Troubleshooting
104156	WARN	Resynchronization of “%s” is in PENDING state. Current sync_action is: “%s”	**Cause**: The resynchronization of the md device is detected in PENDING state. **Action**: LifeKeeper will try to fix the issue by forcing a resynchronization. Check the logs for related errors. When successful assure that the PENDING state has been cleared in /proc/mdstat and the resynchronization is in progress or has been completed for the datarep resource.

Code	Severity	Message	Cause/Action
104157	WARN	/etc/sysconfig/raid-check update failed. Please %s \“md%d\” to SKIP_DEVS.	**Cause**: Unable to make changes in /etc/sysconfig/raid-check to add or remove an entry to the list of MD devices to skip (SKIP_DEVS). **Action**: Check system logs for any errors related to raid-check or SKIP_DEVS. Manually add or remove md listed.
104158	EMERG	WARNING: The local disk partition $self->{‘part’} for data replication device\n$self->{‘md’} has failed. MANUAL INTERVENTION IS REQUIRED.	**Cause**: The local device for a mirror failed. The recovery action has been set to nothing in LKDR_FAILURE requiring manual intervention to recover. **Action**: Check system logs and LifeKeeper logs for errors related to the local disk.
104163	WARN	The “%s_data_corrupt” flag is set in “%s/subsys/scsi/resources/netraid/” on system “%s”. The mirror is being forced online.	**Cause**: The mirror is being forced online, overriding the data_corrupt flag. The data on the specified system will be treated as the latest data. If this is not correct then this can lead to data corruption or data loss. **Action**: None
104164	ERROR	The “%s_data_corrupt” flag for related mirror resource “%s” is set in “%s/subsys/scsi/resources/netraid/” on system “%s”. To avoid data corruption, LifeKeeper will not restore this mirror or any related mirrors in the hierarchy.	**Cause**: The data_corrupt flag exists for one or more mirrors in the hierarchy. To avoid corrupting additional data none of the mirrors are brought in-service until all of the data_corrupt flags are resolved. **Action**: Check the LifeKeeper logs to determine where each mirror was last in-service, aka where the latest data for each mirror resides. The mirrors should be brought in-service on the previous source where the full hierarchy was in-service and allow the mirrors to synchronize with all targets.
104165	ERROR	The “%s_data_corrupt” flag for related mirror resource “%s” is set in “%s/subsys/scsi/	**Cause**: The mirror is being forced online, overriding

Code	Severity	Message	Cause/Action
		resources/netraid/" on system "%s". The mirror resource "%s" is being forced online.	the data_corrupt flag. The data on the specified system will be treated as the correct data to be synchronized with all targets. This can lead to data corruption or data loss if this is not the latest data. **Action**: None
104170	ERROR	Failed to create \"source\" flag file on shared source %s to track mirror source. This may result in a full resync.	**Cause**: The 'source' flag file was not created on the specified system to track the mirror source. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104171	ERROR	Failed to create \"source\" flag file on %s to track mirror source. Target %s will not be added to mirror.	**Cause**: The 'source' flag file was not created on the specified system to track the mirror source. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104172	ERROR	The \"source\" flag file on %s does not contain a valid target (%s). Full resync to remaining targets is required.	**Cause**: The 'source' flag file should contain the system name of a previous source but the name listed was not found in the list of systems configured. **Action**: Report this problem to SIOS support.
104173	ERROR	Failed to create \"source\" flag file on %s to track mirror source.	**Cause**: The 'source' flag file was not created on the specified system to track the mirror source **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104174	ERROR	Failed to create \"previous_source\" flag file to track time waiting on source. Will not be able to timeout.	**Cause**: The 'previous_source' flag file was not created on the mirror source to track the mirror's previous source. **Action**: Check the LKROOT (/opt/LifeKeeper) file

Code	Severity	Message	Cause/Action
			system for errors or that it is full.
104175	ERROR	Failed to create “data_corrupt” flag file on target “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104176	ERROR	The \“source\” flag file on %s to track mirror source does not exist. Full resync is required.	**Cause**: The source flag file should exist on the system and without it the consistency of the mirror can not be verified. A full resync is required to assure data reliability. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for error or that it is full.
104177	ERROR	Failed to determine amount of time waiting on %s.	**Cause**: The amount of time waiting for the previous source could not be determined. Targets will be added with a full resync if the previous source is not found. **Action**: none
104178	ERROR	Failed to update “source” flag file on target “%s”, previous source must be merged first.	**Cause**: The source flag file on the target is updated when it is in-sync and stopped so the the next in-service does not require the previous source. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104180	ERROR	Internal Error: \“previous_source\” has the local system name (%s).	**Cause**: The local system name should not be in the previous_source flag file. **Action**: Report this error to SIOS support.
104181	ERROR	Internal Error: There are no targets waiting on %s to be merged.	**Cause**: There are no targets waiting for a previous

Code	Severity	Message	Cause/Action
			source to merge. **Action**: Report this error to SIOS support.
104182	ERROR	Failed to create \“source\” flag file on %s to track mirror source. This may result in a full resync.	**Cause**: The ‘source’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104186	ERROR	Failed to create \“last_owner\” flag file on %s to track mirror source. This may allow in-service of mirror on old data.	**Cause**: The ‘last_owner’ flag file was not created on the source. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104187	WARN	$REM_MACH has ${REM_TAG}_last_owner file, create flag $FLAGTAG_data_corrupt.	**Cause**: The system listed had the mirror in-service last. **Action**: The system listed has the last_owner file that indicates it has the most recent data and is most likely the best system to in-service the mirror to avoid losing data.
104188	WARN	$REM_MACH is not alive, create flag $FLAGTAG_data_corrupt.	**Cause**: The system listed is not alive. **Action**: Since the system listed is not alive, it cannot be determined whether that system was a more recent mirror source than the local system. Therefore the local system should not automatically be allowed to bring the mirror in-service.
104200	EMERG	Continue to wait for %s to merge bitmap and do partial resyncs to all targets, no timeout set.	**Cause**: In a multi-target configuration targets will not be configured until the previous source is available to merge its bitmap so that all targets will be able to partially resynchronize. The LKDR_WAIT_ON_PREVIOUS_SOURCE_TIMEOUT

Code	Severity	Message	Cause/Action
			entry in /etc/defaults/LifeKeeper is set to “-1” to wait indefinitely. **Action**: Check on the status of the previous source listed in the message and resolve any issues that are preventing it from rejoining the cluster.
104201	EMERG	To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: \”%s/bin/mirror_action %s fullresync %s %s\” on %s.	**Cause**: In a multi-target configuration targets are not being configured, waiting on the previous source to rejoin the cluster. **Action**: Run the command listed in the message to force an immediate full resynchronization to this target and any remaining targets waiting to be resynchronized.
104202	EMERG	Continue to wait for %s to merge bitmap and do partial resyncs to all targets. Continue to wait %d more seconds.	**Cause**: In a multi-target configuration targets will not be configured until the previous source is available to merge its bitmap so that all targets will be able to partially resynchronize. The LKDR_WAIT_ON_PREVIOUS_SOURCE_TIMEOUT entry in /etc/defaults/LifeKeeper is set to the number of seconds to wait. If the previous source does not join the cluster in that time then targets will be added with a full resynchronization. **Action**: Check on the status of the previous source listed in the message and resolve any issues that are preventing it from rejoining the cluster.
104203	EMERG	To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: \”%s/bin/mirror_action %s fullresync %s %s\” on %s.	**Cause**: In a multi-target configuration targets are not being configured, waiting on the previous source to rejoin the cluster. **Action**: Run the command listed in the message to force an immediate full resynchronization to this target and any remaining targets waiting to be resynchronized. Note: Run this command to stop waiting even if the

Code	Severity	Message	Cause/Action
			target listed is deleted and never returning.
104207	ERROR	Failed to create “data_corrupt” flag file on “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the source listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104208	ERROR	Failed to create “data_corrupt” flag file on target “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104209	ERROR	Failed to create “data_corrupt” flag file on “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the source listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104210	ERROR	Failed to create “data_corrupt” flag file on target “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104211	ERROR	Failed to create “data_corrupt” flag file on target “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104212	ERROR	The \“source\” flag file on %s to track mirror source does not exist. Full resync to remaining targets is required.	**Cause**: The source flag file should exist on the system and without it the consistency of the mirror can not be verified. A full resync is required to assure data reliability. All targets not already being

Code	Severity	Message	Cause/Action
			mirrored will require a full resync. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for error or that it is full.
104214	ERROR	Failed to create \“source\” flag file on %s to track mirror source.	**Cause**: The ‘source’ flag file was not created on the specified system to track the mirror source. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104216	ERROR	Failed to create “data_corrupt” flag file on target “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104217	ERROR	Failed to create \“source\” flag file on shared source %s to track mirror source. This may result in a full resync.	**Cause**: The ‘source’ flag file was not created on the specified system to track the mirror source. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104218	ERROR	Failed to create “data_corrupt” flag file on target “%s”.	**Cause**: The ‘data_corrupt’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full
104221	ERROR	Failed to create \“last_owner\” flag file on %s to track mirror source. This may allow in-service of mirror on old data.	**Cause**: The ‘last_owner’ flag file was not created on the target listed. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104222	ERROR	Failed to create “last_owner” flag file to track mirror source”. This may allow in-service of	**Cause**: The ‘last_ownerâ€™ flag file is used to

Code	Severity	Message	Cause/Action
		mirror on old data.	know where the mirror was last in-service. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104223	ERROR	Failed to create “last_owner” flag file to track mirror source”. This may allow in-service of mirror on old data.	**Cause**: The ‘last_ownerâ€™’ flag file is used to know where the mirror was last in-service. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104224	ERROR	Failed to create \“previous_source\” flag file.	**Cause**: The ‘previous_source’ flag file was not created. This is needed to merge the previous source bitmap to avoid a full resync. **Action**: Check the LKROOT (/opt/LifeKeeper) file system for errors or that it is full.
104227	ERROR	Failed to set %s to %s.	**Cause**: This message indicates a failure to set a sysfs parameter for the nbd driver (/sys/block/nbdX). **Action**: It may be necessary to adjust one or more of: NBD_NR_REQUESTS NBD_SCHEDULER LKDR_ASYNC_LIMIT in /etc/default/LifeKeeper to avoid this error.
104232	ERROR	Mirror resize failed on %s (%s). Could not set size to %d.	**Cause**: The mirror resize operation failed.
104233	ERROR	Mirror resize failed on %s (%s). Could not set bitmap to %s and bitmap-chunk to %d.	**Cause**: The mirror resize operation failed.
104234	ERROR	The mirror %s failed to resize. You must successfully complete this operation before	**Cause**: The mirror resize operation failed.

Code	Severity	Message	Cause/Action
		using the mirror. Please try again.	
104235	ERROR	mirror_resize of mirror %s failed due to signal “%s”.	**Cause**: The mirror resize operation failed.
104251	ERROR	There is no LifeKeeper protected resource with tag $tag on system $me.	**Cause**: The given tag does not correspond to a LifeKeeper protected resource on the given system. **Action**: Verify that the resource tag and system name are correct.
104252	ERROR	Resource $tag is not a $app/$typ resource. Please use the $ins_app/$ins_typ resource-specific canfailover script instead.	**Cause**: The scsi/netraid-specific canfailover script was called for a non-scsi/netraid resource. **Action**: Use the canfailover script, if it exists, corresponding to the appropriate app and type of the given resource.

8.1.2. DB2 Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
103001	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The db2nodes.cfg does not contain any server names. **Action**: Ensure the db2nodes.cfg is valid.
103002	ERROR	LifeKeeper was unable to get the version for the requested instance "%s"	**Cause**: "db2level" command did not return DB2 version. **Action**: Check your DB2 configuration.
103003	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Recovery Kit was unable to find any nodes for the DB2 instance. **Action**: Check your DB2 configuration.
103004	ERROR	Unable to get the information for resource "%s"	**Cause**: Failed to get resource information. **Action**: Check your LifeKeeper configuration.
103005	ERROR	Unable to get the information for resource "%s"	**Cause**: Failed to get resource information. **Action**: Check your LifeKeeper configuration.
103006	ERROR	Unable to get the instance information for resource "%s"	**Cause**: Failed to get the instance

Code	Severity	Message	Cause/Action
			information. **Action**: Check your LifeKeeper configuration.
103007	ERROR	Unable to get the instance home directory information for resource "%s"	**Cause**: Failed to get the instance home directory path. **Action**: Check your LifeKeeper configuration.
103008	ERROR	Unable to get the instance type information for resource "%s"	**Cause**: The DB2 Application Recovery Kit found invalid instance type. **Action**: Check your LifeKeeper configuration.
103009	ERROR	LifeKeeper has encountered an error while trying to get the database configuration parameters for database \"$DB\"	**Cause**: There was an unexpected error running "db2 get db cfg for $DB" command. **Action**: Check the logs for related errors and try to resolve the reported problem.
103012	ERROR	LifeKeeper was unable to start the database server for instance "%s"	**Cause**: The requested startup of the DB2 instance failed. **Action**: Check the logs for related errors and try to resolve the reported problem. Correct any reported errors before retrying the "restore" operation.
103013	ERROR	LifeKeeper was unable to start the database server for instance "%s"	**Cause**: The requested startup of the DB2 instance failed. **Action**: Check the logs for related

Code	Severity	Message	Cause/Action
			errors and try to resolve the reported problem. Correct any reported errors before retrying the "restore" operation.
103015	ERROR	An entry for the home directory "%s" of instance "%s" does not exist in "/etc/fstab"	**Cause**: The home directory of instance of Multiple Partition database should exist in "/etc/fstab". **Action**: Ensure the home directory exists in "/etc/fstab".
103016	ERROR	LifeKeeper was unable to mount the home directory for the DB2 instance "%s"	**Cause**: Failed to mount the home directory of instance of Multiple Partition database. **Action**: Ensure the home directory is mounted and retry the operation.
103017	ERROR	Unable to get the instance nodes information for resource "%s"	**Cause**: Failed to get the instance nodes. **Action**: Check your LifeKeeper configuration.
103018	ERROR	LifeKeeper was unable to start database partition server "%s" for instance "%s"	**Cause**: The requested startup of the DB2 instance failed. **Action**: Check the logs for related errors and try to resolve the reported problem. Correct any reported errors before retrying the "restore" operation.
103020	ERROR	LifeKeeper was unable to stop the database server for instance "%s"	**Cause**: The requested shutdown of the DB2 instance failed. **Action**: Check the logs for related errors and try to resolve the reported

Code	Severity	Message	Cause/Action
			problem. Correct any reported errors before retrying the "remove" operation.
103021	ERROR	LifeKeeper was unable to stop the database server for instance "%s"	**Cause**: The requested shutdown of the DB2 instance failed. **Action**: Check the logs for related errors and try to resolve the reported problem. Correct any reported errors before retrying the "remove" operation.
103023	ERROR	Unable to get the instance nodes information for resource "%s"	**Cause**: Failed to get the instance nodes. **Action**: Check your LifeKeeper configuration.
103024	ERROR	LifeKeeper was unable to stop database partition server "%s" for instance "%s"	**Cause**: The requested shutdown of the DB2 instance failed. **Action**: Check the logs for related errors and try to resolve the reported problem. Correct any reported errors before retrying the "remove" operation.
103026	ERROR	Unable to get the instance nodes information for resource "%s"	**Cause**: Failed to get the instance nodes. **Action**: Check your LifeKeeper configuration.
103027	FATAL	The argument for the DB2 instance is empty	**Cause**: Invalid parameters were specified for the create operation. **Action**: Verify the parameters and retry the operation.

Code	Severity	Message	Cause/Action
103028	FATAL	Unable to determine the DB2 instance home directory	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance home directory. **Action**: Ensure the instance owner name is same as the instance name and retry the operation.
103029	FATAL	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance type. **Action**: Check your DB2 configuration.
103030	FATAL	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Recovery Kit was unable to find any nodes for the DB2 instance. **Action**: Check your DB2 configuration.
103031	ERROR	The path "%s" is not on a shared filesystem	**Cause**: The instance home directory should be on a shared filesystem. **Action**: Ensure the path is on shared filesystem and retry the create operation.
103032	ERROR	LifeKeeper was unable to get the DB tablespace containers for instance "%s" or the log path for one of its databases	**Cause**: LifeKeeper could not determine the location of the database table space containers or verify that they are located in a path which is on a mounted filesystem. **Action**: Check the logs for related errors and try to resolve the reported problem. Correct any reported errors before retrying the "create" operation.

Code	Severity	Message	Cause/Action
103033	ERROR	The path "%s" is not on a shared filesystem	**Cause**: The path of database table space container should be on a shared filesystem. **Action**: Ensure database table space container is on a shared filesystem and retry the operation.
103034	ERROR	A DB2 Hierarchy already exists for instance "%s"	**Cause**: An attempt was made to protect the DB2 instance that is already under LifeKeeper protection. **Action**: You must select a different DB2 instance for LifeKeeper protection.
103035	ERROR	The file system resource "%s" is not in-service	**Cause**: The file system which the DB2 resource depends on should be in service. **Action**: Ensure the file system resource is in service and retry the "create" operation.
103036	ERROR	Unable to create the hierarchy for raw device "%s"	**Cause**: LifeKeeper was unable to create the resource {raw device} . **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the "create" operation.
103037	ERROR	A RAW hierarchy does not exist for the tag "%s"	**Cause**: LifeKeeper was unable to find the raw resource {tag} . **Action**: Check your LifeKeeper configuration.

Code	Severity	Message	Cause/Action
103038	ERROR	LifeKeeper was unable to create a dependency between the DB2 hierarchy "%s" and the Raw hierarchy "%s"	**Cause**: The requested dependency creation between the parent DB2 resource and the child Raw resource failed. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors before retrying the "create" operation.
103039	ERROR	LifeKeeper could not disable the automatic startup feature of DB2 instance "%s"	**Cause**: An unexpected error occurred while attempting to update the DB2 setting. **Action**: The DB2AUTOSTART will need to be updated manually to turn off the automatic startup of the instance at system boot.
103040	ERROR	DB2 version "%s" is not installed on server "%s"	**Cause**: LifeKeeper could not find DB2 installed location. **Action**: Check your DB2 configuration.
103041	ERROR	The instance owner "%s" does not exist on target server "%s"	**Cause**: An attempt to retrieve the DB2 instance owner from template server during a "canextend" or "extend" operation failed. **Action**: Verify the DB2 instance owner exists on the specified server. If the user does not exist, it should be created with the same uid and gid on all servers in the cluster.
103042	ERROR	The instance owner "%s" uids are different on target server "%s" and template server "%s"	**Cause**: The user id on the target server {target server} for the DB2 instance owner {user} does not match the value

Code	Severity	Message	Cause/Action
			of the user {user} on the template server {template server}. **Action**: The user ids for the DB2 instance owner {user} must match on all servers in the cluster. The user id mismatch should be corrected manually on all servers before retrying the "canextend" operation.
103043	ERROR	The instance owner "%s" gids are different on target server "%s" and template server "%s"	**Cause**: The group id on the target server {target server} for the DB2 instance owner {user} does not match the value of the user {user} on the template server {template server}. **Action**: The group ids for the DB2 instance owner {user} must match on all servers in the cluster. The group id mismatch should be corrected manually on all servers before retrying the "canextend" operation.
103044	ERROR	The instance owner "%s" home directories are different on target server "%s" and template server "%s"	**Cause**: The home directory location of the user {user} on the target server {target server} does not match the DB2 instance owner's home directory on the template server {template server}. **Action**: The home directory location of the DB2 instance owner {user} must match on all servers in the cluster. The location mismatch should be corrected manually on all servers before retrying the "canextend" operation.
103045	ERROR	LifeKeeper was unable to get the DB2 "SVCENAME" parameter for the DB2 instance	**Cause**: There was an unexpected error running "db2 get dbm cfg" command.

Code	Severity	Message	Cause/Action
			Action: Check your DB2 configuration.
103046	ERROR	Unable to get the value of the DB2 "SVCENAME" parameter for the DB2 instance %s.	**Cause**: The DB2 "SVCENAME" parameter is set to null. **Action**: Check your DB2 configuration.
103047	ERROR	LifeKeeper was unable to get the contents of the "/etc/services" file on the server "%s"	**Cause**: "/etc/services" on the template server does not contain the service names for the DB2 instance. **Action**: The service names in "/etc/services" for the DB2 instance must match on all servers in the cluster. The service names mismatch should be corrected manually on all servers before retrying the "canextend" operation.
103048	ERROR	LifeKeeper was unable to get the contents of the "/etc/services" file on the server "%s"	**Cause**: "/etc/services" on the target server does not contain the service names for the DB2 instance. **Action**: The service names in "/etc/services" for the DB2 instance must match on all servers in the cluster. The service names mismatch should be corrected manually on all servers before retrying the "canextend" operation.
103049	ERROR	The "/etc/services" entries for the instance "%s" are different on target server "%s" and template server "%s"	**Cause**: The "/etc/services" entries for the instance are mismatched. **Action**: The service names in "/etc/services" for the DB2 instance must match on all servers in the cluster. The service names mismatch should be

Code	Severity	Message	Cause/Action
			corrected manually on all servers before retrying the "canextend" operation.
103050	ERROR	The home directory "%s" for instance "%s" is not mounted on server "%s"	**Cause**: LifeKeeper could not find db2nodes.cfg for Multiple Partition instance. **Action**: Ensure the home directory is mounted and retry the operation.
103051	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: Failed to get resource information from the template server. **Action**: Check your LifeKeeper configuration.
103052	ERROR	LifeKeeper was unable to add instance "%s" and/or its variables to the DB2 registry	**Cause**: There was an unexpected error running "db2iset" command. **Action**: Check the logs for related errors and try to resolve the reported problem.
103053	ERROR	Usage: %s instance	
103054	ERROR	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance type. **Action**: Check your DB2 configuration.
103055	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Recovery Kit was unable to find any nodes for the DB2 instance. **Action**: Check your DB2 configuration.

Code	Severity	Message	Cause/Action
103056	ERROR	Usage: %s instance	
103058	ERROR	Usage: %s instance	
103059	ERROR	Usage: %s instance	
103060	ERROR	Unable to determine the DB2 instance home directory	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance home directory. **Action**: Ensure the instance owner name is same as the instance name and retry the operation.
103061	ERROR	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance type. **Action**: Check your DB2 configuration.
103062	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Recovery Kit was unable to find the node for the DB2 instance. **Action**: Check your DB2 configuration.
103063	ERROR	Unable to determine the DB2 install path	**Cause**: The DB2 Application Recovery Kit was unable to find DB2 for the instance. **Action**: Check your DB2 configuration.
103064	ERROR	Usage: nodes -t tag -a add_nodenum | nodes -t tag -d delete_nodenum | nodes -t tag -p	
103065	ERROR	Invalid input provided for "%s" utility operation, characters are not allowed.	**Cause**: Invalid parameters were specified for the "nodes" command. **Action**: Verify the parameters and retry the operation.

Code	Severity	Message	Cause/Action
103066	ERROR	Unable to get the information for resource "%s"	**Cause**: LifeKeeper was unable to find the resource {tag}. **Action**: Verify the parameters and retry the operation.
103067	ERROR	The DB2 instance "%s" is not a EEE or Multiple Partition instance	**Cause**: The resource {tag} is single partition instance. **Action**: Verify the parameters and retry the operation.
103069	ERROR	Node "%s" is already protected by this hierarchy	**Cause**: Invalid parameters were specified for the "nodes" command. **Action**: Verify the parameters and retry the operation.
103070	ERROR	Node number "%s" is the last remaining node protected by resource "%s". Deleting all nodes is not allowed.	**Cause**: Invalid parameters were specified for the "nodes" command. **Action**: Verify the parameters and retry the operation.
103071	ERROR	LifeKeeper is unable to get the equivalent instance for resource "%s"	**Cause**: There was an unexpected error running "nodes" command. **Action**: Check the logs for related errors and try to resolve the reported problem.
103072	ERROR	Unable to set NodesInfo for resource "%s" on "%s"	**Cause**: There was an unexpected error running "nodes" command. **Action**: Check the logs for related errors and try to resolve the reported problem.

Code	Severity	Message	Cause/Action
103073	ERROR	Unable to set NodesInfo for resource "%s" on "%s"	**Cause**: There was an unexpected error running "nodes" command. **Action**: Check the logs for related errors and try to resolve the reported problem.
103074	ERROR	Usage: %s instance	
103075	ERROR	Usage: %s instance	
103076	ERROR	Unable to determine the DB2 instance type	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance type. **Action**: Check your DB2 configuration.
103077	ERROR	Unable to determine the DB2 instance home directory	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance home directory. **Action**: Ensure the instance owner name is same as the instance name and retry the operation.
103078	ERROR	The database server is not running for instance "%s"	**Cause**: A process check for the DB2 instance did not find any processes running. **Action**: The DB2 instance must be started.
103079	ERROR	LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance	**Cause**: The DB2 Application Recovery Kit was unable to find any nodes for the DB2 instance. **Action**: Check your DB2 configuration.

Code	Severity	Message	Cause/Action
103080	ERROR	One or more of the database partition servers for instance "%s" is down	**Cause**: All database partition servers should be running. **Action**: Ensure all database partition servers are running and retry the operation.
103081	ERROR	DB2 local recovery detected another recovery process in progress for "%s" and will exit.	
103082	ERROR	Failed to create flag "%s"	**Cause**: An unexpected error occurred attempting to create a flag for controlling DB2 local recovery processing. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors.
103083	ERROR	Failed to remove flag "%s"	**Cause**: An unexpected error occurred attempting to remove a flag for controlling DB2 local recovery processing. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors.
103084	ERROR	Unable to determine the DB2 instance \"$Instance\" home directory	**Cause**: The DB2 Application Recovery Kit was unable to determine the DB2 instance home directory. **Action**: Ensure the instance owner name is same as the instance name and retry the operation.

8.1.3. DMMP Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
128005	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The quickCheck of {resource} on {server} failed due to an operating system signal {signal}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128008	ERROR	Usage: quickCheck -t <tag name> -i <id>	**Cause**: Incorrect arguments have been supplied to the dmmp device quickCheck command preventing it from running. **Action**: Make sure all software components are properly installed and at the correct version. Rerun the command and supply the correct argument list: -t <Resource Tag> and -i <Resource ID> that identifies the dmmp device resource to be quickChecked.
128010	ERROR	quickCheck for "%s" failed checks of underlying paths, initiate recovery. retry count=%s.	**Cause**: The dmmp kit failed to quickCheck a device after {count} times of retries. A recovery of the protected dmmp resource will be executed. **Action**: Check adjacent log messages for further details and related messages.

Code	Severity	Message	Cause/Action
128021	ERROR	unable to find device for uuid "%s".	**Cause**: The device could not be found by unique id during a restore operation. **Action**: Verify that the resource is configured properly. Rerun the command and supply the correct device id that identifies the dmmp device resource to be restored.
128025	ERROR	Device "%s" failed to unlock.	**Cause**: A non working {device} was detected and could not be unlocked during the restore operation. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128026	ERROR	Device "%s" failed to lock.	**Cause**: The {device} could not be locked during the resotre. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128031	ERROR	unable to find device for uuid "%s".	**Cause**: The device could not be found by unique id during the remove operation. **Action**: Verify that the resource is configured properly. Rerun the command and supply the correct device id that identifies the dmmp device resource to be removed.

Code	Severity	Message	Cause/Action
128034	ERROR	Device "%s" failed to unlock.	**Cause**: The {device} could not be unlocked during the remove. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128036	ERROR	unable to load existing information for device with uuid "%s".	**Cause**: The device information could not be loaded by unique id. **Action**: Make sure the resource is configured properly. Rerun the command and supply the correct device id that identifies the dmmp device resource.
128037	ERROR	unable to load existing information for device "%s".	**Cause**: The device information could not be loaded by name. **Action**: Make sure the resource is configured properly. Rerun the command and supply the correct device name that identifies the dmmp device resource.
128038	ERROR	unable to load existing information for device, no dev or uuid defined.	**Cause**: The device information could not be loaded since neither a unique device id nor name of the device were defined. **Action**: Make sure the resource is configured properly. Rerun the command and supply the correct device id or name that identifies the dmmp device resource.

Code	Severity	Message	Cause/Action
128041	ERROR	unable to load existing information for device with uuid "%s".	**Cause**: The device information could not be loaded by unique id. **Action**: Make sure the resource is configured properly. Rerun the command and supply the correct device id that identifies the dmmp device resource.
128057	ERROR	All paths are failed on "%s".	**Cause**: LifeKeeper detected all paths listed to the protected dmmp device are in the failed state. **Action**: Check adjacent log messages for further details and related messages.
128058	ERROR	could not determine registrations for "%s"! All paths failed.	**Cause**: LifeKeeper could not determine registrations for protected dmmp {device}. All paths to the dmmp {device} are in the failed state. **Action**: Check adjacent log messages for further details and related messages.
128059	WARN	path "%s" no longer configured for "%s", remove from path list.	**Cause**: LifeKeeper detected listed {path} to protected {device} is not valid anymore and will remove it from the path list. **Action**: Check adjacent log messages for further details and related messages.
128060	WARN	registration failed on path "%s" for "%s".	**Cause**: LifeKeeper failed the registration on {path} for protected dmmp {device}.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and related messages.
128062	ERROR	all paths failed for "%s".	**Cause**: LifeKeeper failed to verify a valid path to protected dmmp {device}. **Action**: Check adjacent log messages for further details and related messages.
128072	ERROR	The daemon "%s" does not appear to be running and could not be restarted. Path failures may not be correctly handled without this daemon.	**Cause**: LifeKeeper failed to verify dmmp daemon is running and could not restart it. **Action**: Check adjacent log messages for further details and related messages.
128078	ERROR	"%s" resource type is not installed on "%s".	**Cause**: The Device Mapper Multipath Recovery Kit for dmmp device support is not installed on the system. **Action**: Install the steeleye-lkDMMP Device Mapper Multipath Recovery Kit rpm on the system.
128083	ERROR	This script must be executed on "%s".	**Cause**: An incorrect system name has been supplied as an argument to the devicehier script used to create the dmmp device resource. **Action**: Make sure the cluster nodes and comm-paths are properly configured. Supply the correct system name to the devicehier script. The name must match the name of the system on which the command is run.

Code	Severity	Message	Cause/Action
128084	ERROR	The device %s is not active.	**Cause**: LifeKeeper failed to find the specified {device} as a valid device on the system during resource creation. **Action**: Check adjacent log messages for further details and related messages. Rerun the command and supply the correct argument list: -t <Resource Tag> and -i <Resource ID> that identifies the dmmp device resource to be created.
128086	ERROR	Failed to create "%s" hierarchy.	**Cause**: LifeKeeper failed to create resource hierarchy for {device}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128088	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper failed to create the resource with {tagname} on {server}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128090	ERROR	Failed to create dependency "%s"-"%s" on system "%s".	**Cause**: LifeKeeper failed to create dependency {resource tag name} – {resource tag name} on {system} during creation. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the

Code	Severity	Message	Cause/Action
			operation.
128091	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper failed to create {resource} on {system}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128101	ERROR	"%s" constructor requires a valid argument.	**Cause**: LifeKeeper failed to create an object for the dmmp resouce during construction. **Action**: Rerun the command and supply the correct argument list: -t <Resource Tag> and -i <Resource ID> that identifies the dmmp device resource.
128102	ERROR	Invalid tag "%s".	**Cause**: A resource instance could not be found for the given tag name. **Action**: Make sure the resource is configured properly. Rerun the command and supply the correct argument list: -t <Resource Tag> and -i <Resource ID> that identifies the dmmp device resource.
128111	ERROR	Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations.	**Cause**: LifeKeeper failed to get the registrations of {device} with the message, "bad field in Persistent reservation in cdb". **Action**: Verify if the storage supports persistent reservations. Check adjacent log messages for further details and

Code	Severity	Message	Cause/Action
			related messages. You must correct any reported errors before retrying the operation.
128112	ERROR	Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations.	**Cause**: LifeKeeper failed to get the registrations of {device} with the message, "illegal request". **Action**: Verify if the storage supports persistent reservations. Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128136	ERROR	A previous quickCheck with PID "%s" running for device "%s" has been terminated.	**Cause**: LifeKeeper detected that a previous quickCheck operation is still running during the dmmp resource restore operation. It has been terminated by LifeKeeper. **Action**: Check adjacent log messages for further details and related messages.
128137	ERROR	SCSI reservation conflict on %s during LifeKeeper resource initialization. Manual intervention required.	**Cause**: LifeKeeper detected a SCSI reservation conflict on {device} during dmmp resource restore. **Action**: Check adjacent log messages for further details and related messages. Manual intervention and fix of the reservation conflict on {device} is required.
128138	ERROR	unable to clear registrations on %s.	**Cause**: LifeKeeper failed to clear all the registrations on {device}.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128140	WARN	registration failed on path %s for %s.	**Cause**: LifeKeeper failed to make the registration on {path} for {device}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128143	ERROR	reserve failed (%d) on %s.	**Cause**: LifeKeeper failed to make reservation for {resource} on {device}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128145	ERROR	The server ID "%s" returned by "%s" is not valid.	**Cause**: LifeKeeper failed to generate a valid host {ID}. **Action**: The ID used to register a device is made up of 1 to 12 Hex digits that uniquely identifies the server in the cluster. Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128146	ERROR	device failure on %s. SYSTEM HALTED.	**Cause**: LifeKeeper detected failure on {device} and will reboot the server.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and related messages.
128148	ERROR	device failure on %s. SYSTEM HALTED DISABLED.	**Cause**: LifeKeeper detected a failure on {device}. The reboot was skipped due to LifeKeeper configuration. **Action**: Check adjacent log messages for further details and related messages. Turn on the configuration "SCSIHALT" to make the reboot available for any detected device failure.
128149	ERROR	device failure or SCSI Error on %s. SENDEVENT DISABLED.	**Cause**: LifeKeeper detected a failure on {device}. The event generation was skipped due to LifeKeeper configuration. **Action**: Check adjacent log messages for further details and related messages. Turn on the configuration "SCSIEVENT" to make the sendevent available for any detected device failure.
128150	ERROR	%s does not have EXCLUSIVE access to %s, halt system.	**Cause**: LifeKeeper detected a reservation conflict for {device} on {server} and will reboot the server. **Action**: Check adjacent log messages for further details and related messages.
128151	ERROR	%s does not have EXCLUSIVE access to %s, halt system DISABLED.	**Cause**: LifeKeeper detected a reservation conflict for {device} on {server}. The reboot was skipped due to

Code	Severity	Message	Cause/Action
			LifeKeeper configuration. **Action**: Check adjacent log messages for further details and related messages. Turn on the configuration "RESERVATIONCONFLICT" to make the reboot available for any detected reservation conflicts.
128154	WARN	unable to flush buffers on %s.	**Cause**: LifeKeeper failed to flush the buffers for {device} during dmmp resource remove. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128157	WARN	%s utility not found, limited healthcheck for %s.	**Cause**: LifeKeeper failed to find "dd" utility for the health check of {device}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128160	ERROR	%s failed to read %s.	**Cause**: LifeKeeper failed a disk I/O test for {device} when using {utility}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128163	ERROR	Registration ID "%s" for "%s" is not valid.	**Cause**: LifeKeeper failed to generate a

Code	Severity	Message	Cause/Action
			valid registration {ID} for {device}. **Action**: The ID used to register a device is made up of 4 Hex digits derived from the path to the device. Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128170	ERROR	Usage: canextend <Template system name> <Template tag name>	
128173	ERROR	Usage error OSUquickCheck	
128174	ERROR	both tag and id name not specified	
128175	ERROR	Failed multipath check.	
128500	ERROR	Usage error	**Cause**: Incorrect arguments have been supplied to the dmmp device restore command preventing it from running. **Action**: Rerun the command and supply the correct argument list: -t <Resource Tag> and -i <Resource ID> that identifies the dmmp device resource to be restored.
128504	ERROR	"%s" resource type is not installed on "%s".	**Cause**: The Device Mapper Multipath Recovery Kit for dmmp device support is not installed on the system. **Action**: Install the steeleye-lkDMMP Device Mapper Multipath Recovery Kit rpm on the system.
128506	ERROR	Usage error	**Cause**: Incorrect arguments have been supplied to the dmmp device devShared command preventing it from running.

Code	Severity	Message	Cause/Action
			Action: Rerun the command and supply the correct argument list: <Template Resource System Name> and <Template Resource Tag> that identifies the dmmp device resource to be created.
128507	FATAL	This script must be executed on "%s".	**Cause**: An incorrect system name has been supplied as an argument to the devicehier script used to create the dmmp device resource. **Action**: Supply the correct system name to the devicehier script. The name must match the name of the system on which the command is run.
128511	ERROR	Failed to get the ID for the device "%s". Hierarchy create failed.	**Cause**: The devicehier script used to create the dmmp device resource was unable to determine the SCSI ID for the supplied device. **Action**: Check that the supplied device path exists and that is for a supported SCSI storage array.
128512	ERROR	Failed to get the disk ID for the device "%s". Hierarchy create failed.	**Cause**: The devicehier script used to create the dmmp disk resource was unable to determine the SCSI ID for the supplied disk. **Action**: Check that the supplied device path exists and that is for a supported SCSI storage array.
128513	ERROR	Failed to create the underlying resource for device "%s". Hierarchy create failed.	**Cause**: The creation of the underlying dmmp disk resource failed.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128515	ERROR	Error creating resource "%s" on server "%s"	**Cause**: The creation of the dmmp device resource failed. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128517	ERROR	Failed to create dependency "%s"-"%s" on system "%s".	**Cause**: The parent child dependency creation between the dmmp device and dmmp disk resources failed. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128519	ERROR	Error creating resource "%s" on server "%s"	**Cause**: The attempt to bring the newly created dmmp device resource in service has failed. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
128521	ERROR	Either TEMPLATESYS or TEMPLATETAG argument missing	**Cause**: Incorrect arguments have been supplied to the extend command for the dmmp device resource.

Code	Severity	Message	Cause/Action
			Action: Rerun the dmmp device resource extend and supply the correct template system and tag names.
128540	ERROR	Usage error	**Cause**: Incorrect arguments have been supplied to the dmmp device getId command used to retrieve the SCSI ID. **Action**: Rerun the command and supply the correct argument list: -i <device path> or -b <device ID>.
128541	ERROR	Usage error	**Cause**: Incorrect arguments have been supplied to the command used to delete the dmmp device resource. **Action**: Rerun the command and supply the correct argument list: -t <dmmp device resource tag>.
128543	ERROR	device node \"$dev\" does not exist.	**Cause**: The device node required for restoring the dmmp device resource does not exist. The allocated wait time in restore for udev device creation has been exceed. **Action**: Rerun the dmmp device resource restore once udev has created the device.
128544	ERROR	Usage error	**Cause**: Incorrect arguments have been supplied to the remove command used to take the dmmp device resource out of service. **Action**: Rerun the command and supply the correct argument list: -t <dmmp device resource tag>.

8.1.4. Recovery Kit for EC2 Message Catalog

The Recovery Kit for EC2 Message Catalog below contains listings of all messages that may be encountered while utilizing the Recovery Kit for EC2.

The Combined Message Catalog provides a listing of all messages that may be encountered while using SIOS Protection Suite for Linux and, where appropriate, provides additional explanation of the cause of the errors and necessary action to resolve the error condition. This full listing may be searched for any error code received,

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
129100	FATAL	Failed to load instance from LifeKeeper.	**Cause**: An invalid resource tag or ID was specified. **Action**: Check that the tag or ID is valid and re-run the command.
129103	FATAL	No resource matches tag \"$self->{'tag'}\".	**Cause**: An invalid resource tag was specified. **Action**: Check the tag and re-run the command.
129104	FATAL	An error occurred setting LifeKeeper resource information	**Cause**: An internal error has occurred in LifeKeeper.
129110	ERROR	Could not get the Elastic Network Interface ID for $dev	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129111	ERROR	Failed to get Allocation ID of Elastic IP \"$elasticIp\".	**Cause**: The EC2 API call failed, possibly due to a network issue.

Code	Severity	Message	Cause/Action
			Action: Check the network and the Amazon console and retry the operation.
129113	ERROR	Failed to get my instance ID.	**Cause**: The EC2 instance metadata access failed. **Action**: Check the Amazon console and retry the operation.
129114	ERROR	Failed to get ENI ID.	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129116	ERROR	Failed to associate Elastic IP \"$self->{'EIP'}\" on \"$self->{'DEV'}\".	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129118	WARN	$self->{'EIP'} is not associated with any instance.	**Cause**: The Elastic IP is not associated with any instance. **Action**: LifeKeeper will try to fix the issue by calling the EC2 API to associate the Elastic IP. Check adjacent log messages for more details.
129119	WARN	$self->{'EIP'} is associated with another instance.	**Cause**: The Elastic IP is associated with another instance. **Action**: LifeKeeper will try to fix the issue by calling the EC2 API to

Code	Severity	Message	Cause/Action
			associate the Elastic IP. Check adjacent log messages for more details.
129120	ERROR	Failed to recover Elastic IP.	**Cause**: The EC2 API call failed to associate the Elastic IP. **Action**: Check the network and the Amazon console and retry the operation.
129121	ERROR	Recovery process ended but Elastic IP is not associated with this instance. Please check AWS console.	**Cause**: The EC2 API call failed to associate the Elastic IP. **Action**: Check the network and the Amazon console and retry the operation.
129122	ERROR	Error creating resource \"$target_tag\" with return code of \"$err\".	**Cause**: LifeKeeper was unable to create the resource instance on the server. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
129123	ERROR	Failed to get ENI ID.	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129124	WARN	$self->{‘EIP’} is associated with another network interface.	**Cause**: The Elastic IP is associated with the proper instance, but the wrong ENI. **Action**: LifeKeeper will try to fix the

Code	Severity	Message	Cause/Action
			issue by calling the EC2 API to associate the Elastic IP. Check adjacent log messages for more details.
129125	ERROR	Link check failed for interface \'$dev\'.	**Cause**: The requested interface is showing 'NO-CARRIER' indicating that no link is present. **Action**: Check the network interface and bring the link up.
129126	ERROR	Link check failed for interface \'$dev\'. Reason: down slave.	**Cause**: The requested interface is showing 'NO-CARRIER' indicating that no link is present. **Action**: Check the network interface and bring the link up.
129129	WARN	The link for network interface \'$self->{'DEV'}\' is down. Attempting to bring the link up.	**Cause**: The requested interface is showing 'NO-CARRIER' indicating that no link is present. **Action**: LifeKeeper will try to fix the issue by bringing the link up and associating the Elastic IP with the interface. Check adjacent log messages for more details.
129130	ERROR	Failed to modify \"$opt_t\" to end pint URL \"$endpoint\".	
129137	ERROR	The link for network interface \'$self->{'DEV'}\' is still down.	**Cause**: LifeKeeper could not bring the link up. **Action**: Ensure the interface is enabled and up. Check adjacent log messages for more details.

Code	Severity	Message	Cause/Action
129139	WARN	The link for network interface \'$self->{'DEV'}\' is down.	**Cause**: The requested interface is showing 'NO-CARRIER' indicating that no link is present. **Action**: Check the network interface and bring the link up.
129140	ERROR	Could not get ENI ID for $self->{IP}.	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129142	ERROR	Failed to update route table	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129143	ERROR	You must have exactly one IP address resource as the parent of the RouteTable EC2 resource. Please reconfigure your resource hierarchy.	**Cause**: The Route Table EC2 resource must have one and only one IP resource as a parent. **Action**: Repair the resource hierarchy as necessary.
129144	ERROR	$func called with invalid timeout: $timeout	**Cause**: An invalid timeout value was specified in the /etc/default/LifeKeeper file. **Action**: Verify all EC2_*_TIMEOUT settings are valid in /etc/default/LifeKeeper.
129145	ERROR	$func action timed out after $timeout seconds	**Cause**: The action did not complete

Code	Severity	Message	Cause/Action
			within the timeout period. **Action**: Consider increasing the EC2_*_TIMEOUT value for the given action (in /etc/default/LifeKeeper).
129146	ERROR	failed to run $func with timeout: $@	**Cause**: This is an internal error.
129148	ERROR	Amazon describe-route-tables call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129150	ERROR	Elastic IP \"$elasticIp\" is associated with another instance.	**Cause**: The Elastic IP is not associated with the proper instance. **Action**: LifeKeeper will try to fix the issue by calling the EC2 API to associate the Elastic IP. Check adjacent log messages for more details.
129151	ERROR	Could not get the Association ID for Elastic IP \"$elasticIp\".	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129152	ERROR	Failed to disassociate Elastic IP \"$self->{‘EIP’}\" on \"$self->{‘DEV’}\".	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.

Code	Severity	Message	Cause/Action
129153	ERROR	Failed to disassociate Elastic IP \"$elasticIp\", (err=%s)(output=%s	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129154	ERROR	Amazon describe-addresses call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129155	ERROR	Amazon describe-address call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129157	ERROR	curl call failed (err=%s)(output=%s	**Cause**: The EC2 instance metadata access failed. **Action**: Check the Amazon console and retry the operation.
129159	ERROR	Amazon associate-address call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, possibly due to a network issue. **Action**: Check the network and the Amazon console and retry the operation.
129160	ERROR	Amazon describe-addresses call failed (err=%s)(output=%s	**Cause**: The EC2 API call failed, possibly due to a network issue.

Code	Severity	Message	Cause/Action
			Action: Check the network and the Amazon console and retry the operation.
129161	ERROR	Error deleting resource \"$otherTag\" on \"$otherSys\" with return code of \"$err\".	**Cause**: LifeKeeper was unable to delete the resource instance on the server. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
129162	ERROR	Could not getRouteTablesByIP	
129163	ERROR	Could not getRouteTablesByIP	
129164	ERROR	[$SUBJECT event] mail returned $err	
129165	ERROR	COMMAND OUTPUT: cat /tmp/err$$	
129167	ERROR	snmptrap returned $err for Trap 180	
129168	ERROR	COMMAND OUTPUT: cat /tmp/err$$	
129170	ERROR	This resource is in the old format. Please update.	
129403	ERROR	END failed create of $TAG due to a $sig signal	**Cause**: The create process was interrupted by a signal.
129409	ERROR	The IP resource $IP_RES is not \"ISP\".	**Cause**: The IP resource is not in service. **Action**: Bring the resource in service and retry the operation.
129410	ERROR	Could not find IP resource $IP_RES	**Cause**: Ensure that the IP resource exists and retry the operation.
129412	ERROR	EC2 resource $ID is already protected	**Cause**: A resource with the specified ID already exists.

Code	Severity	Message	Cause/Action
			Action: Make sure to clean up any remnants of an old resource before re-creating a new resource.
129416	ERROR	Error creating resource \"$TAG\" with return code of \"$lcderror\".	**Cause**: LifeKeeper was unable to create the resource instance. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
129418	ERROR	Dependency creation between \"$IP_RES\" and \"$TAG\" failed with return code of \"$lcderror\".	**Cause**: LifeKeeper was unable to create the resource dependency. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
129420	ERROR	In-service failed for tag \"$TAG\".	**Cause**: LifeKeeper could not bring the resource instance into service. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
129423	ERROR	Could not get ENI ID for $dev.	
129425	ERROR	Failed to update route table	
129426	ERROR	In-service (dummy) failed for tag \"$TAG\".	
129800	ERROR	canextend checks failed for \"$self->{‘tag’}\" (err=$ret	**Cause**: The pre-extend checks failed for the target server. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
129801	ERROR	canextend checks failed for \"$self->{‘tag’}\". EC2_HOME \"$self-	

Code	Severity	Message	Cause/Action
		>{EC2_HOME}\" does not exist on $me.	

8.1.5. File System Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
124004	FATAL	resource tag name not specified	**Cause**: Invalid arguments were specified for the "quickCheck" operation. **Action**: Ensure that the correct arguments are passed.
124005	FATAL	resource id not specified	**Cause**: Invalid arguments were specified for the "quickCheck" operation. **Action**: Ensure that the correct arguments are passed.
124007	FATAL	Failed to get resource information	**Cause**: The filesystem resource's info field does not contain the correct information. **Action**: Put the correct information in the resource's info field or restore the system from a recent "lkbackup" to restore the original info field.
124008	ERROR	getId failed	**Cause**: The filesystem resource could not find the underlying disk device. **Action**: Check adjacent log messages for further details. Verify that the resource hierarchy is valid and that all required storage kits are installed.

Code	Severity	Message	Cause/Action
124009	ERROR	LifeKeeper protected filesystem is in service but quickCheck detects the following error	**Cause**: The filesystem kit has found something wrong with the resource. **Action**: Check the messages immediately following this one for more details.
124010	ERROR	\"$id\" is not mounted	**Cause**: The filesystem resource is no longer mounted. **Action**: No action is required. Allow local recovery to remount the resource.
124011	ERROR	\"$id\" is mounted but with the incorrect mount options (current mount option list: $mntopts, expected mount option list: $infoopts	**Cause**: The filesystem resource is mounted incorrectly. **Action**: No action is required. Allow local recovery to remount the resource.
124012	ERROR	\"$id\" is mounted but on the wrong device (current mount device: $tmpdev, expected mount device: $dev	**Cause**: The filesystem resource has the wrong device mounted. **Action**: No action is required. Allow local recovery to remount the resource.
124015	ERROR	LifeKeeper protected filesystem \"$tag\" ($id) is $percent% full ($blocksfree free blocks).	**Cause**: The filesystem is getting full. **Action**: Remove or migrate data from the filesystem.
124016	WARN	LifeKeeper protected filesystem \"$tag\" ($id) is $percent% full ($blocksfree free blocks).	**Cause**: The filesystem is getting full. **Action**: Remove or migrate data from the filesystem.
124020	FATAL	cannot find device information for filesystem $id	**Cause**: The filesystem resource could

Code	Severity	Message	Cause/Action
			not find the underlying disk device. **Action**: Check adjacent log messages for further details. Verify that the resource hierarchy is valid and that all required storage kits are installed.
124029	ERROR	Failed to find child resource.	**Cause**: The filesystem resource could not determine its underlying disk resource. **Action**: Ensure that the resource hierarchy is correct.
124032	FATAL	Script has hung. Exiting.	**Cause**: Processes had files open on a mounted filesystem that needed to be unmounted. Killing those processes has taken too long. **Action**: If this error continues, try to temporarily stop all software that may be using the mount point to allow it to be unmounted. If the filesystem still cannot be unmounted, contact Support.
124042	ERROR	file system $fs failed unmount; will try again	**Cause**: Processes had files open on a mounted filesystem that needed to be unmounted. It can take multiple attempts to clear those processes. **Action**: No action is required. Allow the process to continue.
124046	ERROR	file system $fsname failed unmount	**Cause**: A filesystem could not be unmounted. **Action**: If this error continues, try to temporarily stop all software that may

Code	Severity	Message	Cause/Action
			be using the mount point to allow it to be unmounted. If the filesystem still cannot be unmounted, contact Support.
124049	ERROR	Local recovery of resource has failed (err=$err	**Cause**: A filesystem resource has a problem that cannot be repaired locally. **Action**: No action is required. Allow the resource to be failed over to another system.
124051	WARN	getId failed, try count : $cnt/$try	
124052	ERROR	\"$id\" is mounted but filesystem is shutdown state.	
124103	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: This message should not occur under normal circumstances. **Action**: Check adjacent log messages for further details.
124104	ERROR	$ERRMSG Required template machine name is null	**Cause**: Invalid arguments were specified for the canextend operation. **Action**: Ensure that the arguments are correct. If this error happens during normal operation, please contact Support.
124105	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Invalid arguments were specified for the canextend operation. **Action**: Ensure that the arguments are correct. If this error happens during normal operation, please contact Support.

Code	Severity	Message	Cause/Action
124106	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The resource's underlying disk information cannot be determined. **Action**: Ensure the hierarchy is correct on the template system before extending.
124107	ERROR	$ERRMSG Resource \"$TemplateTagName\" must have one and only one device resource dependency	**Cause**: The resource has too many underlying devices in the hierarchy. **Action**: Ensure the hierarchy is correct on the template system before extending.
124108	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The resource cannot be found on the template system. **Action**: Ensure the hierarchy is correct on the template system before extending.
124109	ERROR	$ERRMSG Can not access canextend for scsi/$DeviceResType resources on machine \"$TargetSysName\	**Cause**: The target system is missing some required components. **Action**: Ensure that the target system has all the correct kits installed and licensed.
124110	ERROR	$ERRMSG Either filesystem \"$TemplateLKId\" is not mounted on \"$TemplateSysName\" or filesystem is not shareable with \"$TargetSysName\	**Cause**: The filesystem isn't in service on the template system or doesn't meet the requirements for extending to the target system. **Action**: Make sure the resource is in service on the template system and review the product documentation regarding the requirements for extending filesystems.

Code	Severity	Message	Cause/Action
124111	ERROR	$ERRMSG File system type \"${FSType}\" is not supported by the kernel currently running on \"${TargetSysName}\	**Cause**: The filesystem's type cannot be mounted on the target system due to lack of kernel support. **Action**: Ensure that the target system has all its kernel modules configured correctly before extending the resource.
124112	ERROR	must specify machine name containing primary hierarchy	**Cause**: Invalid arguments were specified for the creFShier operation. **Action**: If this error happens during normal operation, please contact Support.
124113	ERROR	must specify primary ROOT tag	**Cause**: Invalid arguments were specified for the creFShier operation. **Action**: If this error happens during normal operation, please contact Support.
124114	ERROR	must specify primary mount point	**Cause**: Invalid arguments were specified for the creFShier operation. **Action**: If this error happens during normal operation, please contact Support.
124115	ERROR	must specify primary switchback type	**Cause**: Invalid arguments were specified for the creFShier operation. **Action**: If this error happens during normal operation, please contact Support.
124118	ERROR	dep_remove failure on machine \""$PRIMACH"\" for parent \"$PRITAG\" and child \"$DEVTAG.\	**Cause**: Cleanup after a dependency

Code	Severity	Message	Cause/Action
			creation failed. **Action**: Check adjacent log messages for further details.
124119	ERROR	ins_remove failure on machine \""$PRIMACH"\" for \"$PRITAG.\	**Cause**: Cleanup after an instance creation failed. **Action**: Check adjacent log messages for further details.
124121	ERROR	ins_remove failure on machine \""$PRIMACH"\	**Cause**: Cleanup after a resource creation failed. **Action**: Check adjacent log messages for further details.
124122	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: This message should not occur under normal circumstances. **Action**: Check adjacent log messages for further details.
124123	ERROR	$ERRMSG Required template machine name is null	**Cause**: Invalid arguments were specified for the depstoextend operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124124	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Invalid arguments were specified for the depstoextend operation. **Action**: Ensure the script is called

Code	Severity	Message	Cause/Action
			correctly. If this error happens during normal operation, please contact Support.
124125	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The resource was unable to locate its underlying disk resource. **Action**: Ensure the hierarchy and all dependencies are correct before extending.
124126	ERROR	unextmgr failure on machine \""$PRIMACH"\	**Cause**: The cleanup, after a failed resource extend operation, failed. **Action**: Manually clean up any remaining resources and check adjacent log messages for further details.
124128	ERROR	unextmgr failure on machine \""$PRIMACH"\" for \"$PRITAG.\	**Cause**: The cleanup, after a failed resource extend operation, failed. **Action**: Manually clean up any remaining resources and check adjacent log messages for further details.
124129	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: This message should not occur under normal circumstances. **Action**: Look for additional log messages for more details.
124130	ERROR	$ERRMSG Required template machine name is null	**Cause**: Invalid arguments were specified for the extend operation. **Action**: Ensure the script is called

Code	Severity	Message	Cause/Action
			correctly. If this error happens during normal operation, please contact Support.
124131	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Invalid arguments were specified for the extend operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124132	ERROR	$ERRMSG Required target mount point is null	**Cause**: Invalid arguments were specified for the extend operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124133	ERROR	$ERRMSG Unable to access template resource \"$TemplateTagName\	**Cause**: The tag being extended doesn't exist on the template system. **Action**: Ensure that the hierarchy is correct on the template system before extending.
124134	ERROR	$ERRMSG Detected conflict in expected tag name \"$TargetTagName\" on target machine.	**Cause**: A resource already exists on the target system with the same tag as the resource being extended. **Action**: Recreate one of the conflicting resources with a different tag.
124135	ERROR	$ERRMSG Resource \"$TemplateTagName\" does not have required device resource dependency or unable to access this resource on template machine.	**Cause**: The resource or its underlying disk resource cannot be found on the template system.

Code	Severity	Message	Cause/Action
			Action: Ensure that the hierarchy is correct on the template system before extending.
124136	ERROR	$ERRMSG Resource \"$TemplateTagName\" must have one and only one device resource dependency	**Cause**: The resource has multiple underlying devices in the hierarchy on the template system. **Action**: Ensure the hierarchy is correct before extending and that the filesystem resource only depends on a single disk resource.
124137	ERROR	$ERRMSG Can not access extend for scsi/$DeviceResType resources on machine \"$TargetSysName\	**Cause**: The files required to support the given storage type aren't available on the target system. **Action**: Ensure that the required kits are installed on the target system and licensed.
124138	ERROR	$ERRMSG Unable to access target device resource \"$DeviceTagName\" on machine \"$TargetSysName\	**Cause**: The required underlying disk resource doesn't exist on the target system. **Action**: Check adjacent log messages for further details and ensure that the target system is properly configured for hosting the resources being extended.
124139	ERROR	$ERRMSG Unable to access template \"/etc/mtab\" file	**Cause**: The target system cannot read the template system's /etc/mtab file. **Action**: Check adjacent log messages for further details. Ensure that the /etc/mtab file exists on the template system.

Code	Severity	Message	Cause/Action
124140	ERROR	$ERRMSG Unable to find mount point entry \"$TemplateLKId\" in template \"/etc/mtab\" file. Is template resource in-service?	**Cause**: The resource doesn't appear to be mounted on the template system. **Action**: Make sure the resource is in service before extending.
124141	ERROR	$ERRMSG Unable to find mount point \"$TemplateLKId\" mode on template machine	**Cause**: The details of the mount point on the template system cannot be determined. **Action**: Ensure that the resource is in service and accessible on the template system before extending.
124142	ERROR	$ERRMSG Unable to create or access mount point \"$TargetLKId\" on target machine	**Cause**: The mount point could not be created on the target system. **Action**: Ensure that the mount point's parent directory exists and is accessible on the target system.
124143	ERROR	$ERRMSG Two or more conflicting entries found in /etc/fstab on \"$TargetSysName\	**Cause**: The device or mount point appears to be mounted more than once on the target system. **Action**: Ensure that the mount point is not mounted on the target system before extending.
124144	ERROR	$ERRMSG Failed to create resource instance on \"$TargetSysName\	**Cause**: The resource creation on the target system failed. **Action**: Check adjacent log messages for further details. Make sure to check the logs on the target server.
124145	ERROR	$ERRMSG Failed to set resource instance state for \"$TargetTagName\" on \"$TargetSysName\	**Cause**: The source state could not be

Code	Severity	Message	Cause/Action
			changed to OSU on the target system. **Action**: Check adjacent log messages for further details.
124146	ERROR	must specify machine name containing primary hierarchy	**Cause**: Invalid arguments were specified for the filesyshier operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124147	ERROR	must specify primary mount point	**Cause**: Invalid arguments were specified for the filesyshier operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124149	ERROR	create file system hierarchy failure	**Cause**: The process of finding the resource instance failed. **Action**: Check adjacent log messages for further details.
124150	ERROR	create file system hierarchy failure	**Cause**: The system failed to read the /etc/mtab file. **Action**: Check adjacent log messages for further details.
124151	ERROR	create file system hierarchy failure	**Cause**: The mount point could not be found in the /etc/mtab file. **Action**: Check adjacent log messages

Code	Severity	Message	Cause/Action
			for further details.
124152	ERROR	create file system hierarchy failure	**Cause**: The underlying disk resource could not be found. **Action**: Check adjacent log messages for further details.
124153	ERROR	create file system hierarchy failure	**Cause**: Creating the filesystem resource failed. **Action**: Check adjacent log messages for further details.
124154	ERROR	create file system hierarchy failure	**Cause**: The info field for the resource could not be updated. **Action**: Check adjacent log messages for further details.
124155	ERROR	create file system hierarchy failure	**Cause**: The switchback strategy could not be set on the resource. **Action**: Check adjacent log messages for further details.
124157	ERROR	create file system hierarchy failure \(conflicting entries in /etc/fstab\	**Cause**: The mount point could not be removed from the /etc/fstab file. **Action**: Check adjacent log messages for further details.
124160	ERROR	Unknown error in script filesysins, err=$err	**Cause**: This message should not occur under normal circumstances. **Action**: Check adjacent log messages

Code	Severity	Message	Cause/Action
			for further details.
124161	ERROR	create filesys instance – existid – failure	**Cause**: This message should not occur under normal circumstances. **Action**: Check adjacent log messages for further details.
124163	ERROR	create filesys instance – ins_list – failure	**Cause**: Checking for an existing resource failed. **Action**: Check adjacent log messages for further details.
124164	ERROR	create filesys instance – newtag – failure	**Cause**: The system failed to generate a suggested tag for the resource. **Action**: If this error happens during normal operation, contact Support.
124168	ERROR	create filesys instance – ins_create – failure	**Cause**: The filesystem resource could not be created. **Action**: Check adjacent log messages for further details.
124169	ERROR	filesys instance – ins_setstate – failure	**Cause**: The new filesystem resource's state could not be initialized. **Action**: Check adjacent log messages for further details.
124173	ERROR	create filesys instance – dep_create – failure	**Cause**: The resource's dependency relationship with its underlying disk could not be created.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details.
124174	ERROR	machine not specified	**Cause**: Invalid arguments were specified for the rmenu_mp operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124175	ERROR	mount point not specified	**Cause**: Invalid arguments were specified for the rmenu_mp operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124177	ERROR	unexpected multiple matches found	**Cause**: One or more systems show a filesystem or mount point used more than once. **Action**: Verify filesystem devices and mount points and ensure that filesystems are only mounted once. Look for additional log messages for more details.
124178	ERROR	machine name not specified	**Cause**: Invalid arguments were specified for the rmenump operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.

Code	Severity	Message	Cause/Action
124180	ERROR	must specify filesystem type	**Cause**: Invalid arguments were specified for the validfstype operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124181	ERROR	mount point not specified	**Cause**: Invalid arguments were specified for the validmp operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124182	ERROR	The mount point $MP is not an absolute path	**Cause**: A mount point was specified that isn't an absolute path (doesn't start with a '/'). **Action**: Specify a mount point as an absolute path starting with a '/'.
124183	ERROR	$MP is already mounted on $MACH	**Cause**: The requested mount point is already in use on the system. **Action**: Specify a mount point that isn't in use or unmount it before retrying the operation.
124184	ERROR	The mount point $MP is already protected by LifeKeeper on $MACH	**Cause**: The system is already protecting the specified mount point. **Action**: Choose a different mount point that isn't already being protected.
124185	ERROR	The mount point $MP is not a directory on $MACH	**Cause**: The mount point refers to a

Code	Severity	Message	Cause/Action
			non-directory such as a regular file. **Action**: Choose a mount point that refers to a directory.
124186	ERROR	The mount point directory $MP is not empty on $MACH	**Cause**: The specified mount point refers to a directory that isn't empty. **Action**: Choose a mount point that is empty or remove the contents of the specified directory before retrying the operation.
124187	ERROR	server name not specified	**Cause**: Invalid arguments were specified for the valuepmp operation. **Action**: Ensure the script is called correctly. If this error happens during normal operation, please contact Support.
124188	ERROR	There are no mount points on server $MACH	**Cause**: There are no possible mount points for filesystem resource on the server. **Action**: Check adjacent log messages for further details.
124194	WARN	Please correct conflicting \"/etc/fstab\" entries on server $UNAME for: $FSDEV, $FSNAME	**Cause**: After deleting a filesystem resource, some entries in /etc/fstab need to me manually cleaned up. **Action**: Manually clean up the /etc/fstab file.
124195	ERROR	getchildinfo found no $OKAPP child for $PTAG	**Cause**: The system could not find a child resource in the hierarchy.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and ensure that the hierarchy is correct before retrying the operation.
124196	ERROR	enablequotas – quotacheck may have failed for $FS_NAME	**Cause**: The quota operation failed. **Action**: Check adjacent log messages for further details in both the lifekeeper log and in /var/log/messages.
124198	ERROR	enablequotas – quotaon failed to turn on quotas for $FS_NAME, reason	**Cause**: The quota operation failed. **Action**: Check adjacent log messages for further details in both the lifekeeper log and /var/log/messages.
124200	ERROR	The device node $dev was not found or did not appear in the udev create time limit of $delay seconds	**Cause**: A device node (/dev/…) was not created by udev. This may indicate an issue with the storage or the server's connection to the storage. **Action**: Check adjacent log messages for further details in both the lifekeeper log and in /var/log/messages.
124201	WARN	Device $device not found. Will retry wait to see if it appears.	**Cause**: This can happen under normal conditions while udev creates device node entries for storage. This message should not happen repeatedly. **Action**: Check adjacent log messages for further details in both the lifekeeper log and in /var/log/messages.
124202	ERROR	Command \"$commandwithargs\" failed. Retrying ….	**Cause**: The given command failed but may have failed temporarily. This failure

Code	Severity	Message	Cause/Action
			may happen during normal operations but should not keep failing. **Action**: Check adjacent log messages for further details if this message continues.
124204	WARN	cannot make file system $FSNAME mount point	**Cause**: The mount point directory could not be created. **Action**: Ensure that the mount point can be created. This may be due to filesystem permissions, mount options, etc.
124207	ERROR	\"fsck\"ing file system $FSNAME failed, trying alternative superblock	**Cause**: This message indicates that the typical filesystem check failed. This message may be ok for ext2 filesystems or other filesystems where an alternative superblock location is used. **Action**: Check adjacent log messages for further details.
124209	ERROR	\"fsck\"ing file system $FSNAME with alternative superblock failed	**Cause**: This indicates that an ext2 filesystem (or other filesystem where an alternative superblock location is used) check failed with the alternative superblock location. **Action**: Check adjacent log messages for further details and instructions on how to proceed.
124210	WARN	POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME	**Cause**: A filesystem was put in service or failed over when it was out of sync with its mirror source.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and review the product documentation for information on how to bring the filesystem in service safely.
124211	ERROR	Reason for fsck failure ($retval): $ret	**Cause**: This log message is part of a series of messages and gives the actual exit code from the fsck process. **Action**: Check adjacent log messages for further details and instructions on how to proceed.
124212	ERROR	\"fsck\" of file system $FSNAME failed	**Cause**: The check of the filesystem failed. This is usually due to the filesystem having corruption. **Action**: Check adjacent log messages for further details. Review the product documentation for instructions on how to handle possible filesystem corruption.
124213	WARN	POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME	**Cause**: The system or user tried to bring into service a filesystem that may be corrupted. This can happen if a filesystem is switched or failed over when it was out of sync with its mirror source. **Action**: Check adjacent log messages for further details and review the product documentation for instructions on how to bring the resource into service safely.
124214	ERROR	Reason for fsck failure ($retval)	**Cause**: This message should follow a

Code	Severity	Message	Cause/Action
			previous log message about a filesystem check failure and gives the process exit code of the fsck process. **Action**: Check adjacent log messages for further details.
124218	ERROR	File system $FSNAME was found to be already	**Cause**: This message is part of a series of messages. **Action**: Check adjacent log messages for further details.
124219	ERROR	mounted after initial mount attempt failed.	**Cause**: This message is part of a series of messages. This should not happen under normal circumstances but may not be fatal if the resource can be put in service. **Action**: Check adjacent log messages for further details in both the lifekeeper log and in /var/log/messages.
124220	ERROR	File system $FSNAME failed to mount.	**Cause**: The filesystem could not be mounted. **Action**: Check adjacent log messages for further details.
124221	WARN	Protected Filesystem $ID is full	**Cause**: The filesystem is full. **Action**: Remove unused data from the filesystem or migrate to a larger filesystem.
124222	WARN	Dependent Applications may be affected <>	**Cause**: This indicates that an operation on a resource is likely to cause

Code	Severity	Message	Cause/Action
			operation on other resources based on the resource hierarchy. **Action**: Make sure it's acceptable for the indicated resources to be affected before continuing.
124223	ERROR	Put \"$t\" Out-Of-Service Failed By Signal	**Cause**: This message should not occur under normal circumstances. **Action**: Check adjacent log messages for further details.
124227	ERROR	Put \"$i\" Out-Of-Service Failed	**Cause**: The operation failed. **Action**: Check adjacent log messages for further details.
124230	ERROR	Put \"$t\" In-Service Failed By Signal	**Cause**: This message should not occur under normal circumstances. **Action**: Check adjacent log messages for further details.
124231	ERROR	Put \"$t\" In-Service Failed	**Cause**: The operation failed. **Action**: Check adjacent log messages for further details.
124234	ERROR	Put \"$t\" In-Service Failed	**Cause**: The operation failed. **Action**: Check adjacent log messages for further details.

8.1.6. Gen/App Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
126105	ERROR	script not specified – $PTH is a directory	**Cause**: The specified script path is a directory. **Action**: Correct the path of the script.
126110	ERROR	script $PTH does not exist	**Cause**: The specified script path does not exist. **Action**: Correct the path of the script.
126115	ERROR	script $PTH is a zero length file	**Cause**: The specified script is an empty file. **Action**: Correct the script's file path and check the contents inside the script.
126117	ERROR	script $PTH is not executable	**Cause**: The specified script is not executable. **Action**: Correct the script's file path, check the contents inside the script file and make sure it has the proper execute permissions.
126125	ERROR	required template machine name is null	**Cause**: The input template machine name is null. **Action**: Correct the input template machine name.

Code	Severity	Message	Cause/Action
126130	ERROR	required template resource tag name is null	**Cause**: The input template resource {tag} is null. **Action**: Correct the input template resource tag name.
126135	ERROR	Unable to generate a new tag	**Cause**: Failed to generate a new tag as the same as the template tag name on the target node using the "newtag" script during the extension. The tag name is already existing. **Action**: Avoid using duplicate tag name on the same node and check the log for detail.
126140	ERROR	Unable to generate a new tag	**Cause**: Failed to generate a new tag as input target tag name on the target node using the "newtag" script during the extension. The tag name is already existing. **Action**: Avoid using duplicate tag name on the same node and check log for detail.
126150	ERROR	unable to remote copy template \"$_lscript\" script file	**Cause**: Failed to remote copy template script file. The cause may be due to the non-existence/availability of template script file on template node or any transaction failure during "lcdrcp" process. **Action**: Check the existence/availability of template script and the connection to template node. Also check the logs for related errors and try to resolve the reported problem.

Code	Severity	Message	Cause/Action
126155	ERROR	failed to create resource instance on \"$TargetSysName\	**Cause**: Failed to create resource instance using "ins_create". **Action**: Check the logs for related errors and try to resolve the reported problem.
126160	ERROR	failed to set resource instance state for \"$TargetTagName\" on \"$TargetSysName\	**Cause**: Failed to set resource instance state using "ins_setstate" during GenApp resource extension. **Action**: Check the logs for related errors and try to resolve the reported problem.
126170	ERROR	getlocks failure	**Cause**: Failed to get the administrative lock when creating a resource hierarchy. **Action**: Check the logs for related errors and try to resolve the reported problem.
126175	ERROR	instance create failed	**Cause**: Failed to create a GenApp instance using "appins". **Action**: Check the logs for related errors and try to resolve the reported problem.
126180	ERROR	unable to set state to OSU	**Cause**: Failed to set resource instance state using "ins_setstate" during GenApp resource creation. **Action**: Check the logs for related errors and try to resolve the reported problem.

Code	Severity	Message	Cause/Action
126190	ERROR	resource restore has failed	**Cause**: Failed to restore GenApp resource. **Action**: Check the logs for related errors and try to resolve the reported problem.
126200	ERROR	create application hierarchy rlslocks failure	**Cause**: Failed to release lock after GenApp resource created. **Action**: Check the logs for related errors and try to resolve the reported problem.
126210	ERROR	copy $ltype script $lscript failure	**Cause**: Failed to copy user provided script to appropriate GenApp directory during resource creation. **Action**: Check the existence/availability of user provided script and the GenApp directory as well. Also check the logs for related errors and try to resolve the reported problem.
126215	ERROR	no $ltype script specified	**Cause**: Missing user defined script during GenApp resource creation. **Action**: Check the input action script and run resource creation again.
126220	ERROR	no machine name specified	**Cause**: Missing specified machine name during GenApp resource creation. Failed to copy specified user script due to missing the input for machine name. **Action**: Check the input for machine name and run resource creation again.

Code	Severity	Message	Cause/Action
126225	ERROR	no resource tag specified	**Cause**: Missing specified tag name during resource creation. **Action**: Check the input for source tag name and run resource creation again.
126230	ERROR	$ERRMSG Script was terminated for unknown reason	**Cause**: Failed to extend GenApp resource due to unknown reason. **Action**: Check the logs for related errors and try to resolve the reported problem.
126235	ERROR	$ERRMSG Required template machine name is null	**Cause**: Missing the input for template machine name during GenApp resource extension. **Action**: Check the input for template machine name and do the resource extension again.
126240	ERROR	$ERRMSG Required template resource tag name is null	**Cause**: Missing the input for template resource tag name during GenApp resource extension. **Action**: Check the input for template resource tag name and do the resource extension again.
126245	ERROR	$ERRMSG Can not access extend for $AppType/$ResType resources on machine \"$TargetSysName\	**Cause**: Failed to locate "extend" script during GenApp resource extension on target node. **Action**: Check the existence/availability of "extend" script and do GenApp resource extension again.

Code	Severity	Message	Cause/Action
126250	ERROR	create application failure – ins_list failed	**Cause**: Failed when calling "ins_list" during GenApp resource creation. **Action**: Check the logs for related errors and try to resolve the reported problem.
126255	ERROR	create application failure – unable to generate a new tag	**Cause**: Failed to generate a new tag during the GenApp resource creation. **Action**: Avoid using duplicate tag name on the same node. Also check the logs for related errors and try to resolve the reported problem.
126270	ERROR	create application failure – ins_create failed	**Cause**: Failed using "ins_create" to create GenApp instance. **Action**: Check the logs for related errors and try to resolve the reported problem.
126275	ERROR	create application failure – copy_actions failed	**Cause**: Failed using "copy_actions" to copy user specified template script file. **Action**: Check the existence/availability of template script. Also check the logs for related errors and try to resolve the reported problem.
126290	ERROR	Unable to obtain tag for resource with id $ID	**Cause**: Failed to fetch GenApp resource tag name by input ID during recovery. **Action**: Check the correctness of input ID and existance/availability of GenApp resource in LCD. Also check the logs for related errors and try to resolve the

Code	Severity	Message	Cause/Action
			reported problem.
126300	ERROR	generic application recover script for $TAG was not found or was not executable	**Cause**: Failed to locate the user defined script for GenApp resource during recovery. **Action**: Check the existence/availability of the user defined script and do the GenApp recovery process again.
126310	ERROR	-t flag not specified	**Cause**: Missing the input for resource tag name during GenApp resource restore. **Action**: Check the input for resource tag name and do resource restore again.
126315	ERROR	-i flag not specified	**Cause**: Missing the input for resource internal id during GenApp resource restore. **Action**: Check the input for resource internal id and do resource restore again.
126327	ERROR	END timeout restore of \"$TAG\" (forcibly terminating	
126335	ERROR	restore script \"$LCDAS/$APP_RESTOREDIR/$TAG\" was not found or is not executable	**Cause**: Failed to locate the user defined script for GenApp resource during restore. **Action**: Check the existence/availability of the user defined script and do the GenApp restore process again.
126340	ERROR	-t flag not specified	**Cause**: Missing the input for resource

Code	Severity	Message	Cause/Action
			tag name during GenApp resource remove. **Action**: Check the input for resource tag name and do resource remove again.
126345	ERROR	-i flag not specified	**Cause**: Missing the input for resource internal id during GenApp resource remove. **Action**: Check the input for resource internal id and do resource remove again.
126357	ERROR	END timeout remove of \"$TAG\" (forcibly terminating	
126365	ERROR	remove script \"$LCDAS/$APP_REMOVEDIR/$TAG\" was not found or was not executable	**Cause**: Failed to locate the user defined script for GenApp resource during remove. **Action**: Check the existence/availability of the user defined script and do the GenApp remove process again.
126375	ERROR	Script has hung checking \"$tag\". Forcibly terminating.	**Cause**: The "quickCheck" Script will be forcibly terminated for GenApp resource with tag name {tag} due to a waiting time over the user defined timeout. **Action**: Check the GenApp resource performance and restart quickChecking. Also check the logs for related errors and try to resolve the reported problem.
126380	ERROR	Usage error: no tag specified	**Cause**: Missing the input for resource tag name during GenApp resource quickCheck.

Code	Severity	Message	Cause/Action
			Action: Check the input for resource tag name and retry resource quickCheck.
126385	ERROR	Internal error: ins_list failed on $tag.	**Cause**: Failed using "ins_list" to fetch the GenApp resource information by input tag name during the quickCheck process. **Action**: Correct the input tag name and do the quickCheck process again. Also check the logs for related errors and try to resolve the reported problem.
126390	FATAL	Failed to fork process to execute $userscript: $!	**Cause**: Failed to fork process to execute user defined "quickCheck" script during the GenApp resource "quickCheck" process. **Action**: Check the existency/availability of the user defined "quickCheck" script and do the "quickCheck" process again.
126391	ERROR	quickCheck has failed for \"$tag\". Starting recovery.	**Cause**: The GenApp resource with tag name {tag} is determined to be failed by using the user defined health monitoring script – "quickCheck" and the recovery process will be initiated. **Action**: Check the performance of the GenApp resource when local recovery finished. Also check the logs for related errors and try to resolve the reported problem.
126392	WARN	${convtag}_TIMEOUT: This parameter is old. This parameter will not be supported soon.	

Code	Severity	Message	Cause/Action
126400	ERROR	-t flag not specified	**Cause**: Missing the input for resource tag name during GenApp resource deletion process. **Action**: Check the input for resource tag name and do resource deletion process again.
126405	ERROR	-i flag not specified	**Cause**: Missing the input for resource internal id during GenApp resource deletion process. **Action**: Check the input for resource internal id and do resource deletion process again.
126478	ERROR	Failed to create tag \'$new_leaf\'.	**Cause**: The ‘creapphier’ utility failed to create the specified tag. **Action**: Check /var/log/lifekeeper.log for additional messages from ‘creapphier’.
126479	ERROR	Failed to extend tag \'$new_leaf\'.	**Action**: Check /var/log/lifekeeper.log for additional errors from extend manager.
126481	ERROR	Failed to create dependency on \'$sys\' for \'$new_leaf\' to \'$hier{$leaf}{$sys}{‘Tag’}\'.	**Action**: Check /var/log/lifekeeper.log for errors from the ‘dep_create’ function
126484	ERROR	Tag \'$root_tag\' is not in-service.	**Cause**: The specified tag is not in-service on any node in the cluster. **Action**: Bring the specified tag in-service on any node in the cluster and re-run ‘create_terminal_leaf’.
126485	ERROR	Tag \'$root_tag_1\' was not found, select the root tag for a hierarchy to add a terminal leaf resource.	**Cause**: The first tag passed to

Code	Severity	Message	Cause/Action
			‘create_terminal_leaf’ was not found on the system where the utility was run. **Action**: Verify the resource is in-service on a node in the cluster and fully extended to all nodes.
126486	ERROR	Unable to create leaf tag from \’$tag\’.	**Cause**: A unique terminal leaf tag could not be created. A unique terminal leaf tag could not be created after 100 tries. **Action**: Check for multiple leaf tags and for errors in /var/log/lifekeeper.log that may indicate the problem.
126487	ERROR	Tag \’$root_tag_2\’ was not found, select the root tag for a hierarchy to add a terminal leaf resource.	**Cause**: The second tag passed to ‘create_terminal_leaf’ was not found on the system where the utility was run. **Action**: Verify the resource is in-service on a node in the cluster and fully extended to all nodes.
126488	ERROR	Tag \’$root_tag_1’ is not extended to 3 or more systems.	**Cause**: The specified tag is not extended to 3 or more nodes. **Action**: Extend the specified tag to at least 3 nodes and retry ‘create_terminal_leaf’.
126489	ERROR	$cmd does not support SDRS resources.	**Cause**: A multi-site configuration was detected. **Action**: none
126492	ERROR	Remove resource $tag failed.	
126494	ERROR	Delete dependency failed on \’$sys\’ for \’$tag\’ to \’$parent\’.	

Code	Severity	Message	Cause/Action
126495	ERROR	New tag \'$new_tag\' was modified during create, expected \'$new_leaf\'.	
126496	ERROR	Tag \'$root_tag_1\' is not in-service.	
126497	ERROR	Tag \'$root_tag_2\' is not in-service.	
126498	ERROR	Tag \'$root_tag_1\' and \'$root_tag_2' are not extended to same systems.	

8.1.7. IP Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
123006	FATAL	Unknown version %s of IP address	**Cause**: The IP address does not appear to be valid for either IPv4 or IPv6. **Action**: Provide a valid IP address.
123008	ERROR	No pinglist found for %s.	**Cause**: Problem while opening the pinglist for this IP address. **Action**: Make sure you have provided a pinglist for this IP address.
123009	ERROR	List ping test failed for virtual IP %s	**Cause**: No response was received from any of the addresses in the ping list. **Action**: Check network connectivity of this node and the systems on which the IPs in the ping list reside.
123013	ERROR	Link check failed for virtual IP %s on interface %s.	**Cause**: The requested interface is showing ‘NO-CARRIER’ indicating that no link is present on the phycial layer connection. **Action**: Check the physical connections for the interface and bring the physical layer link up.
123015	ERROR	Link check failed for virtual IP %s on interface %s.	**Cause**: The requested interface is a bonded interface, and one of the slaves is showing ‘NO-CARRIER’ indicating

Code	Severity	Message	Cause/Action
			that no link is present on the phycial layer connection. **Action**: Check the physical connections for the slave interface and bring the physical layer link up.
123024	ERROR	IP address seems to still exist somewhere else.	**Cause**: The IP address appears to be in use elsewhere on the network. **Action**: Either select a different IP address to use or locate and disable the current use of this IP address.
123037	ERROR	must specify machine name containing primary hierarchy	**Cause**: Not enough arguments were provided to creIPhier. **Action**: Supply all of the needed arguments to creIPhier.
123038	ERROR	must specify IP resource name	**Cause**: Not enough arguments were passed to creIPhier. **Action**: Supply all of the needed arguments to creIPhier.
123039	ERROR	must specify primary IP Resource tag	**Cause**: The argument specifying the primary IP Resource tag was missing from the "creIPhier" command. **Action**: Supply all of the needed arguments.
123042	ERROR	An unknown error has occurred in utility validmask on machine %s.	**Cause**: There was an unexpected error running the "validmask" utility. **Action**: Check adjacent log messages

Code	Severity	Message	Cause/Action
			for additional details.
123045	ERROR	An unknown error has occurred in utility getlocks.	**Cause**: There was an unexpected error running the "getlocks" utility. **Action**: Check adjacent log messages for additional details.
123053	ERROR	Cannot resolve hostname %s	**Cause**: A hostname was provided for the IP address, but the system was unable to resolve the name to an IP address. **Action**: Check the correctness of the hostname and verify that name resolution (DNS or /etc/hosts) is working correctly and returns the IP for the hostname.
123055	ERROR	An unknown error has occurred in utility %s on machine %s.	**Cause**: There was a failure while creating the IP resource. **Action**: Check the logs for related errors and try to resolve the reported problem.
123056	ERROR	create ip hierarchy failure: perform_action failed	**Cause**: Unexpected error trying to restore the IP address during creation. **Action**: Check adjacent log messages for additional details.
123059	ERROR	Resource already exists on machine %s	**Cause**: Attempted to create an IP address that already exists. **Action**: Reuse the existing resource or manually remove the IP address if it

Code	Severity	Message	Cause/Action
			exists or use a different IP address.
123060	ERROR	ins_create failed on machine %s	**Cause**: An unexpected failure occurred while creating an IP resource. **Action**: Check adjacent log messages for further details.
123064	ERROR	An unknown error has occurred in utility %s on machine %s.	**Cause**: There was a failure while creating a dependency for the IP resource. **Action**: Check the logs for related errors and try to resolve the reported problem.
123066	ERROR	An error occurred during creation of LifeKeeper application=comm on %s.	**Cause**: A failure occured while calling "app_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
123068	ERROR	An error occurred during creation of LifeKeeper resource type=ip on %s.	**Cause**: A failure occured while calling "typ_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
123089	ERROR	Link check failed for virtual IP %s on interface %s.	
123091	ERROR	the link for interface %s is down	**Cause**: The requested interface is showing ‘NO-CARRIER’ indicating that no link is present on the phycial layer connection.

Code	Severity	Message	Cause/Action
			Action: Check the physical connections for the interface and bring the physical layer link up.
123093	ERROR	the ping list check failed	**Cause**: No response was received from any of the addresses in the ping list. **Action**: Check network connectivity of this node and the systems on which the IPs in the ping list reside.
123095	ERROR	broadcast ping failed	**Cause**: No replies were received from a broadcast ping. **Action**: Verify that at least one host on the subnet will respond to broadcast pings. Verify that virtual IP is on the correct network interface. Consider using a pinglist instead of a broadcast ping.
123096	ERROR	$msg	**Cause**: The broadcast ping used to determine the viability of the virtual IP failed. **Action**: Please ensure that the ping list for this resource is properly configured in the properties panel or that broadcast ping checking is disabled by adding NOBCASTPING=1 to the /etc/default/LifeKeeper configuration file.
123097	ERROR	exec_list_ping(): broadcast ping failed.	**Cause**: The broadcast ping used to determine the viability of the virtual IP failed. **Action**: Ensure that the ping list for this resource is properly configured in the

Code	Severity	Message	Cause/Action
			properties panel or that broadcast ping checking is disabled by adding NOBCASTPING=1 to the /etc/default/LifeKeeper configuration file.
123299	ERROR	Unable to open %s. Reason %s	
123410	ERROR	Usage error OSUquickCheck	
123411	ERROR	OSUquickCheck: both tag and id name not specified	
123412	ERROR	resource $Tag not found on local server	
123414	ERROR	The link for network interface $IPObj->{‘device’} is down	
123415	ERROR	No pinglist found for $IPObj->{‘ipaddr’}	
123416	ERROR	List ping test failed for virtual IP $IPObj->{‘ipaddr’}	

8.1.8. Oracle Listener Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
122005	ERROR	Unable to "%s" on "%s"	**Cause**: There was an unexpected error running "getlocks". **Action**: Check the logs for related errors and try to resolve the reported problem.
122007	ERROR	Unable to "%s" on "%s"	**Cause**: There was an unexpected error running "rlslocks". **Action**: Check the logs for related errors and try to resolve the reported problem.
122009	ERROR	The path %s is not a valid file.	**Cause**: There is no listener.ora file. **Action**: Ensure the file exists and retry the operation.
122010	ERROR	The listener user does not exist on the server %s.	**Cause**: "Stat" command could not get user id. **Action**: Retry the operation.
122011	ERROR	The listener user does not exist on the server %s.	**Cause**: UID is not in passwd file. **Action**: Ensure the UID exists in passwd file and retry the operation.
122012	ERROR	The listener user does not exist on the server %s.	**Cause**: User name is not in passwd file.

Code	Severity	Message	Cause/Action
			Action: Ensure the user name exists in passwd file; retry the operation.
122023	ERROR	The %s command failed (%d	**Cause**: This message contains the return code of the "lsnrctl" command. **Action**: Check the logs for related errors and try to resolve the reported problem.
122024	ERROR	$line	**Cause**: The message contains the output of the "lsnrctl" command. **Action**: Check the logs for related errors and try to resolve the reported problem.
122039	ERROR	Usage error	**Cause**: Invalid parameters were specified for the restore operation. **Action**: Verify the parameters and retry the operation.
122040	ERROR	Script $cmd has hung on the restore of \"$opt_t\". Forcibly terminating.	**Cause**: The listener restore script reached its timeout value. **Action**: Ensure listener.ora is valid and that LSNR_START_TIME (default 35 seconds) in /etc/default/LifeKeeper is set to a value greater than or equal to the time needed to start the listener.
122041	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to restore the resource {resource} on {server}. **Action**: Check the logs for related

Code	Severity	Message	Cause/Action
			errors and try to resolve the reported problem.
122045	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: Failed to get resource information. **Action**: Check your LifeKeeper configuration.
122046	ERROR	Usage error	**Cause**: Invalid parameters were specified for the restore operation. **Action**: Verify the parameters and retry the operation.
122049	ERROR	The script $cmd has hung on remove of \"$opt_t\". Forcibly terminating.	**Cause**: The listener remove script reached its timeout value. **Action**: Ensure listener.ora is valid and that LSNR_STOP_TIME (default 35 seconds) in /etc/default/LifeKeeper is set to a value greater than or equal to the time needed to stop the listener.
122051	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122055	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to quickCheck the resource {resource} on {server}. **Action**: Check the logs for related errors and try to resolve the reported problem.

Code	Severity	Message	Cause/Action
122057	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122064	WARN	The %s level is set to %s a %s will not occur.	**Cause**: The minimal Listener protection level is Start and Monitor. **Action**: Start the listener manually.
122066	ERROR	Script has hung checking \"$tag\". Forcibly terminating.	**Cause**: The listener quickCheck script reached its timeout value. **Action**: Ensure listener.ora is valid and that LSNR_STATUS_TIME (default 15 seconds) in /etc/default/LifeKeeper is set to a value greater than or equal to the time needed to check the listener.
122067	ERROR	Usage error	**Cause**: Invalid parameters were specified for the quickCheck operation. **Action**: Verify the parameters and retry the operation.
122069	ERROR	Usage error	**Cause**: Invalid parameters were specified for the delete operation. **Action**: Verify the parameters and retry the operation.
122072	ERROR	%s: resource "%s" not found on local server	**Cause**: Invalid parameters were specified for the recover operation. **Action**: Verify the parameters and retry the operation.

Code	Severity	Message	Cause/Action
122074	WARN	The local recovery attempt has failed but %s level is set to %s preventing a failover to another node in the cluster. With %s recovery set all local recovery failures will exit successfully to prevent resource failovers.	**Cause**: The optional listener recovery level is set to local recovery only. **Action**: Switch over the resource tree manually.
122078	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to recover the resource {resource} on {server}. **Action**: Check the logs for related errors and try to resolve the reported problem.
122082	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122083	ERROR	$cmd has hung checking \"$tag\". Forcibly terminating	**Cause**: The recover script was stopped by signal. **Action**: Ensure listener.ora is valid.
122084	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to extend the resource {resource} on {server}. **Action**: Verify the parameters and retry the operation.
122085	ERROR	Usage: %s %s	**Cause**: Invalid parameters were specified for the canextend operation. **Action**: Verify the parameters and retry the operation.

Code	Severity	Message	Cause/Action
122086	ERROR	The values specified for the target and the template servers are the same. Please specify the correct values for the target and template servers.	**Cause**: The values specified for the target and the template servers are the same. **Action**: Perform the steps listed in the message text.
122087	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122088	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: Failed to get listener user name from resource information. **Action**: Ensure the resource info field is valid then retry the operation.
122089	ERROR	The listener user %s does not exist on the server %s.	**Cause**: User name is not in passwd file. **Action**: Ensure the user name exists in passwd file and retry the operation.
122090	ERROR	The id for user %s is not the same on template server %s and target server %s.	**Cause**: User ID should be same on both servers. **Action**: Trim user ID to the same.
122091	ERROR	The group id for user %s is not the same on template server %s and target server %s.	**Cause**: Group ID should be same on both servers. **Action**: Trim group ID to the same.
122092	ERROR	Cannot access canextend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to run pre-extend checks because it was

Code	Severity	Message	Cause/Action
			unable to find the "canextend" script on {server}. **Action**: Check your LifeKeeper configuration.
122097	ERROR	Usage: %s %s	**Cause**: Invalid arguments were specified for the "configActions" operation. **Action**: Verify the arguments and retry the operation.
122098	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122099	ERROR	Unable to update the resource %s to change the %s to %s on %s.	**Cause**: LifeKeeper failed to put information into the info field. **Action**: Restart LifeKeeper and retry the operation.
122100	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122101	ERROR	Unable to update the resource %s to change the %s to %s on %s.	**Cause**: LifeKeeper failed to put information to info field on {server}. **Action**: Restart LifeKeeper on {server} and retry the operation.

Code	Severity	Message	Cause/Action
122103	ERROR	Usage: %s %s	**Cause**: Invalid parameters were specified for the create operation. **Action**: Verify the parameters and retry the operation.
122124	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: LifeKeeper was unable to create the resource {resource} on {server}. **Action**: Check the logs for related errors and try to resolve the reported problem.
122126	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected error running "rlslocks". **Action**: Check the logs for related errors and try to resolve the reported problem.
122127	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to create the resource {resource} on {server}. **Action**: Check the logs for related errors and try to resolve the reported problem.
122129	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected error running "getlocks." **Action**: Check the logs for related errors and try to resolve the reported problem.
122131	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to

Code	Severity	Message	Cause/Action
			create the resource {resource} on {server}. **Action**: Check your LifeKeeper configuration.
122133	ERROR	Unable to create a file system resource hierarchy for the file system %s.	**Cause**: There was an unexpected error running "filesyshier." **Action**: Check adjacent log messages for further details.
122135	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected error running "dep_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
122140	ERROR	Resource "%s" is not ISP on server "%s" Manually bring the resource in service and retry the operation	**Cause**: IP resource {tag} which the listener resource depends on should be ISP. **Action**: Perform the steps listed in the message text.
122141	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected error running "dep_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
122144	ERROR	Usage: %s %s	**Cause**: Invalid parameters were specified for the "create_ins" operation. **Action**: Verify the parameters and retry

Code	Severity	Message	Cause/Action
			the operation.
122145	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected error running "app_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
122146	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected error running "typ_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
122147	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected error running "newtag." **Action**: Check the logs for related errors and try to resolve the reported problem.
122148	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to create the resource {resource} on {server}. **Action**: Check your LifeKeeper configuration.
122149	ERROR	An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation.	**Cause**: There was an unexpected error running "ins_setstate." **Action**: Check the logs for related errors and try to resolve the reported problem.

Code	Severity	Message	Cause/Action
122150	ERROR	Error creating resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to create the resource {resource} on {server}. **Action**: Check your LifeKeeper configuration.
122151	ERROR	Usage: %s %s	**Cause**: Invalid arguments were specified for the "depstoextend" operation. **Action**: Verify the arguments and retry the operation.
122152	ERROR	Usage: %s %s	**Cause**: Invalid parameters were specified for the "extend" operation. **Action**: Verify the parameters and retry the operation.
122153	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122154	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to extend the resource {resource} on {server}.
122155	ERROR	Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s"	**Cause**: During the Listener resource extension, a resource instance was found using the same {tag} and/or {id} but with a different resource application and type. **Action**: Resource IDs must be unique.

Code	Severity	Message	Cause/Action
			The resource instance with the ID matching the Oracle Listener resource instance must be removed.
122156	ERROR	Cannot access extend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to extend the resource hierarchy because it was unable to the find the script EXTEND on {server}. **Action**: Check your LifeKeeper configuration.
122157	ERROR	Usage: %s %s	**Cause**: Invalid arguments were specified for the "getConfigIps" operation. **Action**: Verify the arguments and retry the operation.
122158	ERROR	The file %s is not a valid listener file. The file does not contain any listener definitions.	**Cause**: Failed to find any listener definitions. **Action**: Ensure listener definition is in the listener.ora and retry the operation.
122159	ERROR	Usage: %s %s	**Cause**: Invalid arguments were specified for the "getSidListeners" operation. **Action**: Verify the arguments and retry the operation.
122160	ERROR	The file %s is not a valid listener file. The file does not contain any listener definitions.	**Cause**: Failed to find any listener definitions. **Action**: Ensure listener definition is in the listener.ora and retry the operation.

Code	Severity	Message	Cause/Action
122161	ERROR	Usage: %s %s	**Cause**: Invalid arguments were specified for the "lsn-display" operation. **Action**: Verify the arguments and retry the operation.
122162	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}. **Action**: Check your LifeKeeper configuration.
122163	ERROR	Usage: %s %s	**Cause**: Invalid parameters were specified for the updateHelper operation. **Action**: Verify the parameters and retry the operation.
122164	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: LifeKeeper was unable to update the resource {resource} on {server}. **Action**: Check the logs for related errors and try to resolve the reported problem.
122166	ERROR	Usage: %s %s	**Cause**: Invalid parameters were specified for the "updateHelper" operation. **Action**: Verify the parameters and retry the operation.
122170	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected error running "dep_create."

Code	Severity	Message	Cause/Action
			Action: Check the logs for related errors and try to resolve the reported problem.
122171	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected error running "dep_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
122172	ERROR	Usage: %s %s	**Cause**: Invalid arguments were specified for the "updIPDeps" operation. **Action**: Verify the arguments and retry the operation.
122173	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: LifeKeeper was unable to update the resource {resource} on {server}. **Action**: Check the logs for related errors and try to resolve the reported problem.
122175	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected error running "rlslocks." **Action**: Check the logs for related errors and try to resolve the reported problem.
122177	ERROR	Unable to "%s" on "%s	**Cause**: There was an unexpected error running "getlocks." **Action**: Check the logs for related errors and try to resolve the reported

Code	Severity	Message	Cause/Action
			problem.
122180	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected error running "dep_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
122181	ERROR	Unable to create a dependency between parent tag %s and child tag %s.	**Cause**: There was an unexpected error running "dep_create." **Action**: Check the logs for related errors and try to resolve the reported problem.
122183	ERROR	The path %s is not a valid file.	**Cause**: There is no listener.ora file. **Action**: Ensure the file exists and retry the operation.
122185	ERROR	The file %s is not a valid listener file. The file does not contain any listener definitions.	**Cause**: LifeKeeper failed to find any valid listener definitions. **Action**: Ensure there are valid listener definitions in the listener.ora and retry the operation.
122186	ERROR	The value specified for %s cannot be empty. Please specify a value for this field.	**Cause**: The config and/or executable {path} field is empty. **Action**: Input a non-empty value for {path} and retry the operation.
122187	ERROR	The path %s is not a valid file or directory.	**Cause**: The defined {path} is invalid.

Code	Severity	Message	Cause/Action
			Action: Ensure the {path} exists and retry the operation.
122188	ERROR	The path %s is not a valid file or directory.	**Cause**: There is no {path}. **Action**: Ensure the {path} exists and retry the operation.
122189	ERROR	The value specified for %s cannot be empty. Please specify a value for this field.	**Cause**: The config and/or executable Path field is empty. **Action**: Input path for the field.
122190	ERROR	Usage: %s %s	**Cause**: Invalid arguments were specified for the "valid_rpath" operation. **Action**: Verify the arguments and retry the operation.
122191	ERROR	The values specified for the target and the template servers are the same.	**Cause**: Invalid argument of valid_rpath. **Action**: Ensure arguments and retry the operation.
122192	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in /etc/oratab or {path}. **Action**: Ensure oratab file exists in {path} or ORACLE_ORATABLOC in /etc/default/Lifekeeper is set to a valid path.
122193	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in /etc/oratab or {path}. **Action**: Ensure oratab file exists in

Code	Severity	Message	Cause/Action
			{path} or ORACLE_ORATABLOC in /etc/default/Lifekeeper is set to a valid path.
122194	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in /etc/oratab or {path}. **Action**: Ensure oratab file exists in {path} or ORACLE_ORATABLOC in /etc/default/Lifekeeper is set to a valid path.
122195	ERROR	Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s"	**Cause**: There is no oratab file in /etc/oratab or {path}. **Action**: Ensure oratab file exists in {path} or ORACLE_ORATABLOC in /etc/default/Lifekeeper is set to a valid path.
122196	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: LifeKeeper was unable to remove the resource {resource} on {server}. **Action**: Check the logs for related errors and try to resolve the reported problem.
122197	ERROR	Unable to find the configuration file \"oratab\" in its default locations, /etc/oratab or $listener::oraTab on \"$me\"	**Cause**: There is no oratab file in /etc/oratab or {path}. **Action**: Ensure oratab file exists in {path} or ORACLE_ORATABLOC in /etc/default/Lifekeeper is set to a valid path.
122198	ERROR	remove for $okListener failed.	

8.1.9. Oracle Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
122500	ERROR	Usage: %s %s	**Cause**: Invalid parameters were specified for the create operation. **Action**: Verify the parameters are correct and retry the operation.
122501	ERROR	DB instance "%s" is already protected on "%s".	**Cause**: An attempt was made to protect an Oracle database instance {sid} that is already under LifeKeeper protection on {server}. **Action**: You must select a different database instance {sid} for LifeKeeper protection.
122502	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122503	ERROR	Unable to locate the oratab file "%s" on "%s".	**Cause**: The oratab file was not found at the default or alternate locations on {server}. **Action**: Verify the oratab file exists and has proper permissions for the Oracle user. A valid oratab file is required to complete the "create" operation.

Code	Severity	Message	Cause/Action
122504	ERROR	Unable to determine Oracle user for "%s" on "%s".	**Cause**: The Oracle Application Recovery Kit was unable to determine the ownership of the Oracle database installation binaries. **Action**: The owner of the Oracle binaries must be a valid non-root user on {server}. Correct the permissions and ownership of the Oracle database installation and retry the operation.
122505	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	**Cause**: The database instance {sid} was not running or connections to the database were not available via the credentials provided. **Action**: The database instance {sid} must be started on {server} and the proper credentials must be provided for the completion of the "create" operation.
122506	ERROR	Unable to determine Oracle dbspaces and logfiles for "%s" on "%s".	**Cause**: A query to determine the location of required tablespaces, logfiles and related database files failed. This may have been caused by an internal database error. **Action**: Check the adjacent log messages for further details and related errors. Check the Oracle log (alert.log) and related trace logs (*.trc) for additional information and correct the reported problem(s).
122507	ERROR	Unknown chunk type found for "%s" on "%s".	**Cause**: The specified tablespace, logfile or other required database file is not one of the LifeKeeper supported file or character device types.

Code	Severity	Message	Cause/Action
			Action: The specified file {database_file} must reference an existing character device or file. Consult the Oracle installation documentation to recreate the specified file {database_file} as a supported file or character device type.
122508	ERROR	DB Chunk "%s" for "%s" on "%s" does not reside on a shared file system.	**Cause**: The specified tablespace, logfile or other required database file {database_file} does not reside on a file system that is shared with other systems in the cluster. **Action**: Use the LifeKeeper UI or "lcdstatus (1M)" to verify that communication paths have been properly created. Use "rpm" to verify that the necessary Application Recovery Kits for storage protection have been installed. Verify that the file is, in fact, not on shared storage, and if not, move it to a shared storage device.
122510	ERROR	File system create failed for "%s" on "%s". Reason	**Cause**: LifeKeeper was unable to create the resource {filesystem} on the specified server {server}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the "create" operation.
122511	ERROR	%s	**Cause**: The message contains the output of the "filesyshier" command. **Action**: Check the adjacent log messages for further details and related

Code	Severity	Message	Cause/Action
			messages. Correct any reported errors.
122513	ERROR	Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	**Cause**: LifeKeeper was unable to create a dependency between the database resource {tag} and the necessary child resource {childtag}. **Action**: Check adjacent log messages for further details and related messages. Once any problems have been corrected, it may be possible to create the dependency between {tag} and {childtag} manually.
122514	ERROR	Unable to "%s" on "%s" during resource create.	**Cause**: The Oracle Application Recovery Kit was unable to release the administrative lock using the "rlslocks" command. **Action**: Check adjacent log messages for further details and related messages.
122516	ERROR	Raw device resource created failed for "%s" on "%s". Reason	**Cause**: LifeKeeper was unable to create the resource {raw device} on the specified server {server}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the "create" operation.
122519	ERROR	In-service attempted failed for tag "%s" on "%s".	**Cause**: The "perform_action" command for {tag} on {server} failed to start the database {sid}. The in-service operation has failed.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the "create" operation.
122521	ERROR	Create of app "%s" on "%s" failed with return code of "%d".	**Cause**: There was an error running the command "app_create" to create the internal application type. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122522	ERROR	Create of typ "%s" for app "%s" on "%s" failed with return code of "%d".	**Cause**: There was an error running the command "typ_create" to create the internal resource type. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122524	ERROR	Setting "resstate" for resource "%s" on "%s" failed with return code of "%d".	**Cause**: There was an error running the command "ins_setstate" to set the resource state to {state}. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122525	ERROR	The values specified for the target and the template servers are the same: "%s".	**Cause**: The value specified for the target and template servers for the

Code	Severity	Message	Cause/Action
			"extend" operation were the same. **Action**: You must specify the correct parameter for the {target server} and {template server}. The {target server} is the server where the {tag} will be extended.
122526	ERROR	Unable to locate the oratab file in "/etc" or in "%s" on "%s".	**Cause**: The oratab file was not found at the default or alternate locations on {server}. **Action**: Verify the oratab file exists and has proper permissions for the Oracle user. A valid oratab file is required to complete the "extend" operation.
122527	ERROR	Unable to retrieve the Oracle user on "%s".	**Cause**: An attempt to retrieve the Oracle user from {template server} during a "canextend" or "extend" operation failed. **Action**: The owner of the Oracle binaries must be a valid user on {target server} and {template server}. Correct the permissions and ownership of the Oracle database installation and retry the operation.
122528	ERROR	The Oracle user and/or group information for user "%s" does not exist on the server "%s".	**Cause**: LifeKeeper is unable to find the Oracle user and/or group information for the Oracle user {user} on the server {server}. **Action**: Verify the Oracle user {user} exists on the specified {server}. If the user {user} does not exist, it should be created with the same uid and gid on all servers in the cluster.

Code	Severity	Message	Cause/Action
122529	ERROR	The id for user "%s" is not the same on template server "%s" and target server "%s".	**Cause**: The user id on the target server {target server} for the Oracle user {user} does not match the value of the user {user} on the template server {template server}. **Action**: The user ids for the Oracle user {user} must match on all servers in the cluster. The user id mismatch should be corrected manually on all servers before retrying the "extend" operation.
122530	ERROR	The group id for user "%s" is not the same on template server "%s" and target server "%s".	**Cause**: The group id on the target server {target server} for the Oracle user {user} does not match the value of the user {user} on the template server {template server}. **Action**: The group ids for the Oracle user {user} must match on all servers in the cluster. The group id mismatch should be corrected manually on all servers before retrying the "extend" operation.
122532	ERROR	No file system or raw devices found to extend for "%s" on "%s".	**Cause**: There were no dependent file system or raw device resources found for the Oracle resource {tag} on server {template server}. **Action**: Check adjacent log messages for further details and related messages.
122533	WARN	A RAMDISK (%s) was detected in the ORACLE Database configuration for "%s" on "%s". LifeKeeper cannot protect RAMDISK. This RAMDISK resource will not be protected by LifeKeeper! ORACLE hierarchy creation will continue.	**Cause**: The specified tablespace, logfile or other database file {database_file} was detected as a ramdisk. No protection is available for this type of resource in the current

Code	Severity	Message	Cause/Action
			LifeKeeper product. **Action**: The ramdisk will not be protected. You must manually ensure that the required database file {database_file} will be available during all Oracle database operations.
122534	ERROR	Failed to initialize object instance for Oracle sid "%s" on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122537	ERROR	Update of instance info field for "%s" on "%s" failed (%s).	**Cause**: There was an error while running the command "ins_setinfo" to update the internal resource information field. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122538	ERROR	Initial connect with query buffer to database "%s" on "%s" failed, testing output.	**Cause**: A connection attempt to the Oracle database {sid} to determine the database status has failed. **Action**: The connection attempt failed with the specified credentials. Check the adjacent log messages for further details and related errors. Check the Oracle log (alert.log) and related trace logs (*.trc) for additional information

Code	Severity	Message	Cause/Action
			and correct the reported problem(s).
122542	ERROR	The "%s [%s]" attempt of the database "%s" appears to have failed on "%s".	**Cause**: The attempted Oracle action {action} using method {action_method} for the database instance {sid} failed on the server {server}. **Action**: Check the adjacent log messages for further details and related errors. Check the Oracle log (alert.log) and related trace logs (*.trc) for additional information and correct the reported problem(s).
122543	ERROR	All attempts to "%s" database "%s" on "%s" failed	**Cause**: All efforts to perform the action {action} on the Oracle database {sid} on server {server} have failed. **Action**: Check the adjacent log messages for further details and related errors. Check the Oracle log (alert.log) and related trace logs (*.trc) for additional information and correct the reported problem(s).
122544	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occurred while attempting to update the oratab entry for the database {sid}. The error occurred while attempting to open the oratab file. **Action**: The oratab file entry for {sid} will need to be updated manually to turn off the automatic start up of the database at system boot.
122545	ERROR	Unable to locate the oratab file in "/etc" or in "%s" on "%s".	**Cause**: The oratab file was not found at the default or alternate locations on

Code	Severity	Message	Cause/Action
			{server}. **Action**: Verify the oratab file exists and has proper permissions for the oracle user. A valid oratab file is required to complete the "extend" operation.
122546	ERROR	Unable to open file "%s" on "%s" (%s).	**Cause**: The specified file {file} could not be opened or accessed on the server {server} due to the error {error}. **Action**: Verify the existence and permissions on the specified file {file}. Check adjacent log messages for further details and related errors. You must correct any reported errors before retrying the operation.
122547	ERROR	(cleanUpPids):Forcefully killing hung pid(s):pid(s)="%s"	**Cause**: The process {pid} failed to respond to the request to terminate gracefully. The process {pid} will be forcefully terminated. **Action**: Use the command line to verify that the process {pid} has been terminated. Check the adjacent log messages for further details and related messages.
122548	ERROR	Unable to locate the DB utility (%s/%s) on this host.	**Cause**: The Oracle binaries and required database utility {utility} located at {path/utility} were not found on this server {server}. **Action**: Verify that the Oracle binaries and required software utilities are installed and properly configured on the server {server}. The Oracle binaries must be installed locally on each node

Code	Severity	Message	Cause/Action
			or located on shared storage available to all nodes in the cluster.
122549	ERROR	Oracle internal error or non-standard Oracle configuration detected. Oracle User and/or Group set to "root".	**Cause**: The detected ownership of the Oracle database installation resolves to the root user and/or root group. Ownership of the Oracle installation by root is a non-standard configuration. **Action**: The owner of the Oracle binaries must be a valid non-root user on {server}. Correct the permissions and ownership of the Oracle database installation and retry the operation.
122550	ERROR	Initial inspection of "%s" failed, verifying failure or success of received output.	**Cause**: The previous Oracle query {query} or command {cmd} failed to return success. **Action**: Check the adjacent log messages for further details and related errors. Check the Oracle log (alert.log) and related trace logs (*.trc) for additional information and correct the reported problem(s).
122551	ERROR	Logon failed with "%s" for "%s" on "%s". Please check username/password and privileges.	**Cause**: The logon with the credentials {credentials} for the database instance {sid} on server {server} failed. An invalid user {user} or password was specified. **Action**: Verify that the Oracle database user {user} and password {password} are indeed valid. In addition, the Oracle database user {user} must have sufficient privileges for the attempted action.

Code	Severity	Message	Cause/Action
122552	ERROR	%s	**Cause**: The message contains the output of the "sqlplus" command. **Action**: Check adjacent log messages for further details and related messages.
122553	ERROR	Unable to open file "%s" on "%s" (%s).	**Cause**: The specified file {file} could not be opened or accessed on the server {server} due to the error {error}. **Action**: Verify the existence and permissions on the specified file {file}. Check adjacent log messages for further details and related errors. You must correct any reported errors before retrying the operation.
122554	ERROR	The tag "%s" on "%s" is not an Oracle instance or it does not exist.	**Cause**: The specified tag {tag} on server {server} does not refer to an existing and valid Oracle resource instance. **Action**: Use the UI or "lcdstatus (1M)" to verify the existence of the resource tag {tag}. The resource tag {tag} must be an Oracle resource instance to use the command "ora-display."
122555	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected while attempting to update the authorized user, password and database role for the Oracle resource instance. **Action**: Check adjacent log messages for further details and related

Code	Severity	Message	Cause/Action
			messages. You must correct any reported errors before retrying the operation.
122557	ERROR	Update of user and password failed for "%s" on "%s".	**Cause**: A request to update the user and password for the resource tag {tag} failed. The specified credentials failed the initial validation/connection attempt on server {server}. **Action**: Verify the correct credentials {user/password} were specified for the attempted operation. Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122559	ERROR	Update of user and password failed for "%s" on "%s".	**Cause**: The update of the user and password information for the resource tag {tag} on server {server} failed. **Action**: Verify the correct credentials {user/password} were specified for the attempted operation. Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122562	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The required Oracle executable {exe} was not found on this server {server}. **Action**: Verify that the Oracle binaries and required software utilities are installed and properly configured on the server {server}. The Oracle binaries must be installed locally on each node

Code	Severity	Message	Cause/Action
			or located on shared storage available to all nodes in the cluster.
122566	ERROR	Unable to find Oracle home for "%s" on "%s".	**Cause**: The Oracle home directory {Oracle home} does not appear to contain files necessary for the proper operation of the Oracle instance {sid}. **Action**: Verify using the command line that the Oracle home directory {Oracle home} contains the Oracle binaries, a valid spfile{sid}.ora or init{sid}.ora file.
122567	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected. The specified internal ID {id} does not match the expected SID {sid}. **Action**: Verify the parameters are correct. Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122568	ERROR	DB Processes are not running on "%s".	**Cause**: A process check for the Oracle instance did not find any processes running on server {server}. **Action**: If local recovery is enabled, the Oracle instance will be restarted locally. Check adjacent log messages for further details and related messages.
122572	ERROR	Failed to create flag "%s" on "%s".	**Cause**: An unexpected error occurred attempting to create a flag for controlling Oracle local recovery

Code	Severity	Message	Cause/Action
			processing causing a failover to the standby node. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors.
122574	ERROR	all attempts to shutdown the database %s failed on "%s".	**Cause**: The shutdown of the Oracle database failed during a local recovery process most likely caused because the maximum number of database connections has been reached. **Action**: Check the Oracle logs for connection failures caused by the maximum number of available connections being reached, and if found, consider increasing the value. Additionally, set the tunable LK_ORA_NICE to 1 to prevent connection failures from causing a quickCheck failure followed by a local recovery attempt.
122597	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected during pre-extend checking. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the pre-extend.
122598	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being created while attempting to determine the validity of the Oracle home directory.

Code	Severity	Message	Cause/Action
			Action: Check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the "create."
122599	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected while attempting to look up the Oracle user on the template system. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the extend.
122600	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected while attempting to display the resource properties. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the display of the resource properties.
122601	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected while attempting to check for valid database authorization. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the command.

Code	Severity	Message	Cause/Action
122603	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected while attempting to perform health checks on the Oracle resource instance. **Action**: Check that correct arguments were passed to the quickCheck command and also check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the restore.
122604	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: There was an unexpected error creating an internal representation of the Oracle instance being protected while attempting to perform a local recovery on the Oracle resource instance. **Action**: Check that correct arguments were passed to the "recover" command, and also check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the recover.
122606	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	**Cause**: The database instance {sid} was not running or connections to the database are not available via the credentials provided. **Action**: The database instance {sid} must be started on {server} and the proper credentials must be provided for the completion of the selected operation.

Code	Severity	Message	Cause/Action
122607	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	**Cause**: The database instance {sid} was not running or connections to the database are not available via the credentials provided. **Action**: The database instance {sid} must be started on {server} and the proper credentials must be provided for the completion of the selected operation.
122608	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: The "remove" operation failed to create the resource object instance required to take the Oracle resource Out of Service. **Action**: Check that correct arguments were passed to the "remove" command and also check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the restore.
122609	ERROR	Failed to create object instance for Oracle on "%s".	**Cause**: The "restore" operation failed to create the resource object instance required to put the Oracle resource In Service. **Action**: Check that correct arguments were passed to the "restore" command and also check the adjacent log messages for further details and related messages. Correct any reported errors before retrying the "restore."
122610	ERROR	Unable to "%s" on "%s" during resource create.	**Cause**: The Oracle Application Recovery Kit was unable to create the administrative lock using the "getlocks" command during resource creation.

Code	Severity	Message	Cause/Action
			Action: Check adjacent log messages for further details and related messages. Correct any reported errors before retrying the create.
122611	ERROR	%s	**Cause**: The requested dependency creation between the parent Oracle resource and the child File System resource failed. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors before retrying the create operation.
122612	ERROR	%s	**Cause**: The requested dependency creation between the parent Oracle resource and the child Raw resource failed. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors before retrying the create operation.
122613	ERROR	%s	**Cause**: The requested dependency creation between the parent Oracle resource and the child Raw resource failed. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors before retrying the create operation.
122614	ERROR	%s	**Cause**: The requested dependency creation between the parent Oracle resource and the child Listener

Code	Severity	Message	Cause/Action
			resource failed. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors before retrying the create operation.
122616	ERROR	%s	**Cause**: The requested start up or shutdown of the Oracle database failed. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors before retrying the "restore" or "remove" operation.
122618	ERROR	Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	**Cause**: LifeKeeper was unable to create a dependency between the database resource {tag} and the necessary child resource {childtag}. **Action**: Check adjacent log messages for further details and related messages. Once any problems have been corrected, it may be possible to create the dependency between {tag} and {childtag} manually.
122619	ERROR	Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	**Cause**: LifeKeeper was unable to create a dependency between the database resource {tag} and the necessary child resource {childtag}. **Action**: Check adjacent log messages for further details and related messages. Once any problems have been corrected, it may be possible to create the dependency between {tag} and {childtag} manually.

Code	Severity	Message	Cause/Action
122625	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The quickCheck process was unable to find the Oracle executable "sqlplus." **Action**: Check the Oracle configuration and also check adjacent log messages for further details and related messages. Correct any reported problems.
122626	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The remove process was unable to find the Oracle executable "sqlplus." **Action**: Check the Oracle configuration and also check adjacent log messages for further details and related messages. Correct any reported problems.
122627	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The restore process was unable to find the Oracle executable "sqlplus." **Action**: Check the Oracle configuration and also check adjacent log messages for further details and related messages. Correct any reported problems.
122628	ERROR	Unable to find the Oracle executable "%s" on "%s".	**Cause**: The recover process was unable to find the Oracle executable "sqlplus." **Action**: Check the Oracle configuration and also check adjacent log messages for further details and related messages. Correct any reported problems.

Code	Severity	Message	Cause/Action
122632	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: During a remove, the resource instance {sid} passed to the remove process does not match internal resource instance information for the {sid}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122633	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: During a restore, the resource instance {sid} passed to restore does not match internal resource instance information for the {sid}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122634	ERROR	Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command.	**Cause**: During resource recovery, the resource instance {sid} passed to recovery does not match internal resource instance information for the {sid}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122636	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	**Cause**: The create of the Oracle resource hierarchy {tag} failed on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122638	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The create action for the Oracle

Code	Severity	Message	Cause/Action
			database resource {tag} on server {server} failed. The signal {sig} was received by the create process. **Action**: Check adjacent log messages for further details and related messages. You must correct any reported errors before retrying the operation.
122640	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occurred attempting to create the Oracle resource instance {tag} on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122641	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occurred attempting to create the Oracle resource instance {tag} on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122642	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occurred attempting to create the Oracle resource instance {tag} on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122643	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to extend the resource {resource} on {server}.

Code	Severity	Message	Cause/Action
			Action: Check the adjacent log messages for further details and related messages. Correct any reported errors.
122644	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: An unexpected error occurred attempting to retrieve resource instance information for {tag} on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors and retry the extend.
122645	ERROR	Cannot access canextend script "%s" on server "%s"	**Cause**: LifeKeeper was unable to run pre-extend checks because it was unable to find the "canextend" script on {server} for a dependent child resource. **Action**: Check your LifeKeeper configuration.
122646	ERROR	Error getting resource information for resource "%s" on server "%s"	**Cause**: An unexpected error occurred attempting to retrieve resource instance information for {tag} on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors and retry the extend.
122647	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: LifeKeeper was unable to extend the resource {resource} on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.

Code	Severity	Message	Cause/Action
122648	ERROR	Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s"	**Cause**: During the database resource extension, a resource instance was found using the same {tag} and/or {id} but with a different resource application and type. **Action**: Resource IDs must be unique. The resource instance with the ID matching the Oracle resource instance must be removed.
122649	ERROR	Error creating resource "%s" on server "%s"	**Cause**: An unexpected error occurred attempting to create the Oracle resource instance {tag} on {server}. **Action**: Check adjacent log messages for further details and related messages. Correct any reported errors.
122650	ERROR	Cannot access extend script "%s" on server "%s"	**Cause**: The request to extend the database resource {resource} to {server} failed because it was unable to the find the script {extend} on {server} for a dependent child resource. **Action**: Check your LifeKeeper configuration.
122651	ERROR	Cannot extend resource "%s" to server "%s"	**Cause**: The request to extend the database resouce {resource} to {server} failed because of an error attempting to extend a dependent child resource. **Action**: Check the adjacent log messages for further details and related messages. Correct any reported errors.
122654	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The health check for the

Code	Severity	Message	Cause/Action
			database {sid} was terminated because the quickCheck process received a signal. This is most likely caused by the quickCheck process requiring more time to complete than was allotted. **Action**: The health check time for an Oracle resource is controlled by the tunable value ORACLE_QUICKCHECK_TIMEOUT. Set it to a value greater than 45 seconds to allow more time for the health check process to complete.
122655	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The request to take database {sid} "Out of Service" was terminated because the remove process received a signal. This is most likely caused by the remove process requiring more time to complete than was allotted. **Action**: The remove time for an Oracle resource is controlled by the tunable value ORACLE_REMOVE_TIMEOUT. Set the tunable to a value greater than 240 seconds to allow more time for the remove process to complete.
122659	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The request to place database {sid} "In Service" was terminated because the restore process received a signal. This is most likely caused by the restore process requiring more time to complete than was allotted. **Action**: The restore time for an Oracle resource is controlled by the tunable value ORACLE_RESTORE_TIMEOUT. Set the tunable to a value greater than 240 seconds to allow more time for the

Code	Severity	Message	Cause/Action
			restore process to complete.
122663	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	**Cause**: The recovery of the failed database was terminated because the recovery process received a signal. This is most likely caused by the recovery process requiring more time to complete than was allotted. **Action**: The recovery time for an Oracle resource is controlled by the tunable values ORACLE_RESTORE_TIMEOUT and ORACLE_REMOVE_TIMEOUT. Set one or both of these to a value greater than 240 seconds to allow more time for a recovery to complete.
122670	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occurred while attempting to update the oratab entry for the database {sid}. The error occurred while attempting to open the temporary file used in the update process. **Action**: The oratab file entry for {sid} will need to be updated manually to turn off the automatic start up of the database at system boot.
122671	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occurred while attempting to update the oratab entry for the database {sid}. The error occurred while attempting to close the temporary file used in the update process. **Action**: The oratab file entry for {sid} will need to be updated manually to turn off the automatic start up of the

Code	Severity	Message	Cause/Action
			database at system boot.
122672	ERROR	Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s".	**Cause**: An unexpected error occurred while attempting to update the oratab entry for the database {sid}. The error occurred while attempting to rename the temporary file back to oratab. **Action**: The oratab file entry for {sid} will need to be updated manually to turn off the automatic start up of the database at system boot.
122673	ERROR	Unable to log messages queued while running as oracle user %s on %s. Reason: $!	**Cause**: An unexpected error {reason} occurred while attempting to add messages to the log file. These messages were generated while running as the Oracle user. **Action**: Review the reason for the failure and take corrective action.
122674	ERROR	Unable to open %s Reason: %s.	**Cause**: An unexpected error occurred while attempting to open a connection to the Oracle database and run the database {cmd}. **Action**: Check adjacent log messages for further details and related messages. Additionally, check the Oracle log (alert.log) and related trace logs (*.trc) for additional information and correct any reported problems.
122680	ERROR	Unable to find Oracle home for "%s" on "%s".	
122681	ERROR	Failed to create object instance for Oracle on "%s".	
122682	ERROR	Unable to find the Oracle executable "%s" on "%s".	

Code	Severity	Message	Cause/Action
122683	ERROR	Backup node %s is unreachable; abort username/password changes.	
122684	ERROR	The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper.	
122685	ERROR	The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper.	
122686	ERROR	The quickCheck of %s has timed out on server %s. The default TIMEOUT is 45 seconds. To increase the TIMEOUT, set ORACLE_QUICKCHECK_TIMEOUT in /etc/default/LifeKeeper.	
122687	ERROR	Usage: %s %s	

8.1.10. Oracle PDB Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
122251	ERROR	Update of pluggable database info field for "%s" on "%s" failed (%s).	
122252	ERROR	Initial connect with query buffer to database "%s" on "%s" failed, testing output.	
122253	ERROR	The Oracle database "%s" is not running or no open connections are available on "%s".	
122261	ERROR	The Oracle resource (%s) and dependency are not set on %s.	
122262	ERROR	Usage: %s %s	
122263	ERROR	The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper.	
122264	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122268	ERROR	Failed to create object instance for Oracle on "%s".	
122269	ERROR	no dependency for Oracle on "%s".	
122270	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122271	ERROR	Usage: %s %s	
122272	ERROR	The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper.	
122273	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122277	ERROR	Usage: %s %s	
122278	ERROR	Failed to create object instance for Oracle on "%s".	
122279	ERROR	no dependency for Oracle on "%s".	
122280	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122281	ERROR	The quickCheck of %s has timed out on server %s. The default TIMEOUT is 45	

Code	Severity	Message	Cause/Action
		seconds. To increase the TIMEOUT, set ORACLE_QUICKCHECK_TIMEOUT in /etc/default/LifeKeeper.	
122282	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122284	ERROR	Failed to create object instance for Oracle on "%s".	
122285	ERROR	no dependency for Oracle on "%s".	
122287	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122288	ERROR	Usage: %s %s	
122291	ERROR	Cannot extend resource "%s" to server "%s"	
122292	ERROR	The values specified for the target and the template servers are the same: "%s".	
122294	ERROR	Cannot access canextend script "%s" on server "%s"	
122295	ERROR	Usage: %s %s	
122296	ERROR	DB instance "%s" is not protected on "%s".	
122297	ERROR	Failed to create object instance for OraclePDB on "%s".	
122298	ERROR	Unable to locate the oratab file "%s" on "%s".	
122299	ERROR	END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d	
122301	ERROR	Unable to "%s" on "%s" during resource create.	
122302	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122304	ERROR	Unable to "%s" on "%s" during resource create.	
122305	ERROR	Unable to determine Oracle user for "%s" on "%s".	
122306	ERROR	Error creating resource "%s" on server "%s"	
122308	ERROR	Dependency creation between Oracle pluggable database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason	
122309	ERROR	%s	
122311	ERROR	In-service attempted failed for tag "%s" on	

Code	Severity	Message	Cause/Action
		"%s".	
122312	ERROR	Usage: %s %s	
122313	ERROR	Create of app "%s" on "%s" failed with return code of "%d".	
122314	ERROR	Create of typ "%s" for app "%s" on "%s" failed with return code of "%d".	
122316	ERROR	Create of resource tag via "newtag" on "%s" failed.	
122318	ERROR	Error creating resource "%s" on server "%s"	
122320	ERROR	Setting "resstate" for resource "%s" on "%s" failed with return code of "%d".	
122321	ERROR	Error creating resource "%s" on server "%s"	
122322	ERROR	Usage: %s %s	
122323	ERROR	Usage: %s %s	
122324	ERROR	Usage: %s %s	
122325	ERROR	Cannot extend resource "%s" to server "%s"	
122326	ERROR	Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s"	
122327	ERROR	Error creating resource "%s" on server "%s"	
122328	ERROR	Cannot access extend script "%s" on server "%s"	
122329	ERROR	Cannot extend resource "%s" to server "%s"	
122330	ERROR	Usage: %s %s	
122331	ERROR	Failed to create object instance for OraclePDB on "%s".	
122332	ERROR	Usage: %s %s	
122334	ERROR	Backup node %s is unreachable; abort protection PDB changes.	
122336	ERROR	Update of protection PDB failed for "%s" on "%s".	
122339	ERROR	Usage: %s %s	
122340	ERROR	Usage: %s %s	
122341	ERROR	Usage: %s %s	
122342	ERROR	Failed to create object instance for OraclePDB on "%s".	

Code	Severity	Message	Cause/Action
122343	ERROR	Usage: %s %s	
122344	ERROR	END failed %s of "%s" on server "%s" due to a "%s" signal	
122348	ERROR	Failed to create flag "%s" on "%s".	
122350	ERROR	Failed to create object instance for Oracle on "%s".	
122351	ERROR	no dependency for Oracle on "%s".	
122353	ERROR	Unable to find the Oracle executable "%s" on "%s".	
122356	ERROR	Usage: %s %s	
122357	ERROR	Failed to create object instance for OraclePDB on "%s".	
122358	ERROR	The selected oracle SID "%s" is not a CDB.	
122359	ERROR	No protectable PDB found for the selected SID "%s".	
122360	ERROR	No protected Oracle database found on "%s".	

8.1.11. SCSI Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
125102	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}.
125103	ERROR	`printf '%s is not shareable with any machine.' $DEV`	**Cause**: The device does not appear to be shared with any other systems. **Action**: Verify that the device is accessible from all servers in the cluster. Ensure that all relevant storage drivers and software are installed and configured properly.
125104	ERROR	`printf 'Failed to create disk hierarchy for "%s" on "%s"' $PRIMACH $DEV`	**Cause**: The creation of a resource to protect a physical disk failed. **Action**: Check adjacent log messages for more details and try to resolve the reported problem.
125107	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}.
125114	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}.
125120	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}.
125123	ERROR	`printf 'Cannot access depstoextend script "%s" on server "%s"' $depstoextend $TargetSysName`	**Cause**: LifeKeeper was unable to run

Code	Severity	Message	Cause/Action
			pre-extend checks on the resource hierarchy because it was unable to find the script "DEPSTOEXTEND" on {server}. **Action**: Check your LifeKeeper configuration.
125126	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $ChildTag $TemplateSysName`	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}.
125129	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName`	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}.
125155	ERROR	SCSI $DEV failed to lock.	**Cause**: There was a problem locking a SCSI device. **Action**: Check adjacent log messages for more details and try to resolve the reported problem.
125164	ERROR	SCSI $INFO failed to unlock.	**Cause**: There was a problem unlocking a SCSI device. **Action**: Check adjacent log messages for more details and try to resolve the reported problem.
125181	ERROR	`printf 'Template resource "%s" on server "%s" does not exist' $TemplateTag $TemplateSysName`	**Cause**: LifeKeeper was unable to find the resource {tag} on {server}.

8.1.12. Quick Service Protection Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
134003	ERROR	catch a \"$sig\" signal	**Cause**: The “create” process was interrupted by a signal. **Action**: Check adjacent log messages.
134004	ERROR	Unable to getlocks on $server during resource create. Error ($rc	**Cause**: Failed to get the administrative lock when creating a resource hierarchy. **Action**: Check the logs for related errors and try to resolve the reported problem.
134005	ERROR	The service \"$serviceName\" is not supported on $server. Error ($rc	**Cause**: The service does not exist or cannot be protected. **Action**: Input the appropriate service name.
134006	ERROR	The service \"$serviceName\" is already protected on $server.	**Cause**: This service is already protected. **Action**: Can not create a resource for the protection of this service.
134007	ERROR	Error creating resource $tag. Error ($rc	**Cause**: LifeKeeper was unable to create the resource instance. **Action**: Check adjacent log messages. Correct the cause of the error.

Code	Severity	Message	Cause/Action
134011	ERROR	In-service attempted failed for tag $tag.	**Cause**: Failed to restore QSP resource. **Action**: Check the log related to the service that you want to protect. Resolve the problem.
134015	ERROR	Unable to rlslocks on $server during resource create. Error ($rc	**Cause**: Failed to release lock after QSP resource created. **Action**: Check the logs for related errors and try to resolve the reported problem.
134103	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" does not exist	**Cause**: The resource cannot be found on the template server. **Action**: Ensure the hierarchy is correct on the template server before extending.
134104	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" is not QSP resource (app=$ins[1], res=$ins[2]	**Cause**: The template resource is not QSP resource. **Action**: Expand to the same type of resources as a template resource.
134105	ERROR	The service \"$service\" is not supported on $me. Error ($check	**Cause**: The service does not exist on target server. **Action**: Install the service on target server before extending.
134106	ERROR	The service \"$service\" is already protected on $me.	**Cause**: There is already a resource of the same ID on target server. **Action**: Cannot create the resource of the same service.

Code	Severity	Message	Cause/Action
134203	ERROR	catch a \"$sig\" signal	**Cause**: The “extend” process resource was interrupted by a signal. **Action**: Check adjacent log messages.
134204	ERROR	Template resource \"$template_tag\" on server \"$template_sys\" does not exist	**Cause**: The resource cannot be found on the template server. **Action**: Ensure the hierarchy is correct on the template server before extending.
134208	ERROR	Error creating resource \"$tag\" on server \"$me\"	**Cause**: LifeKeeper was unable to create the resource instance on target server. **Action**: Check adjacent log messages. Correct the cause of the error.
134401	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The “restore” process of the service does not terminate within the specified time. **Action**: Check about the protected service and retry the “restore” operation. Also check the logs for related errors and try to resolve the reported problem.
134405	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a system error. **Action**: Determine why fork fails.
134407	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service command.

Code	Severity	Message	Cause/Action
			Action: It is an error to manually run the service command with “start” option. Correct the cause of the error by reference to error message.
134501	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The “remove” process of the service does not terminate within the specified time. **Action**: Check about the protected service and retry the “remove” operation. Also check the logs for related errors and try to resolve the reported problem.
134505	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a system error. **Action**: Determine why fork fails.
134507	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service command. **Action**: It is an error to manually run the service command with “stop” option. Correct the cause of the error by reference to error message.
134601	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The "quickCheck" process will be forcibly terminated due to a waiting time over the user defined timeout. **Action**: Check about the protected service. Also check the logs for related errors and try to resolve the reported problem.

Code	Severity	Message	Cause/Action
134605	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a system error. **Action**: Determine why fork fails.
134607	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service command. **Action**: It is an error to manually run the service command with “status” option. Correct the cause of the error by reference to error message.
134701	ERROR	timeout $cmd for \"$tag\". Forcibly terminating.	**Cause**: The “recover” process of the service does not terminate within the specified time. **Action**: Check about the protected service. Also check the logs for related errors and try to resolve the reported problem.
134706	FATAL	Failed to fork process to execute service command: $!	**Cause**: Failed to fork. This is a system error. **Action**: Determine why fork fails.
134708	ERROR	service command has failed for \"$tag\"	**Cause**: Failed to execute service command. **Action**: It is an error to manually run the service command with “start” option. Correct the cause of the error by reference to error message.
134803	ERROR	tag \"$tag\" does not exist on server \"$me\"	**Cause**: The specified tag does not exist. This is an internal error.

Code	Severity	Message	Cause/Action
134804	ERROR	app type \"$ins[1]\" is not $app	**Cause**: The specified tag is not QSP resource. This is an internal error.
134805	ERROR	res type \"$ins[2]\" is not $res	**Cause**: The specified tag is not QSP resource. This is an internal error.
134823	ERROR	tag \"$tag\" does not exist on server \"$me\"	**Cause**: The specified tag does not exist. This is an internal error.
134824	ERROR	app type \"$ins[1]\" is not $app	**Cause**: The specified tag is not QSP resource. This is an internal error.
134825	ERROR	res type \"$ins[2]\" is not $res	**Cause**: The specified tag is not QSP resource. This is an internal error.
134843	ERROR	tag \"$tag\" does not exist	**Cause**: The specified tag does not exist. This is an internal error.
134844	ERROR	app type \"$ins[1]\" is not $app	**Cause**: The specified tag is not QSP resource. This is an internal error.
134845	ERROR	res type \"$ins[2]\" is not $res	**Cause**: The specified tag is not QSP resource. This is an internal error.

8.1.13. GUI Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
000200	ERROR	pam_start() failed	
000201	ERROR	pam_authenticate failed (user %s, retval %d	
000202	ERROR	pam_end() failed?!?!	
000203	ERROR	Did not find expected group ‘lkguest’	
000204	ERROR	Did not find expected group ‘lkoper’	
000205	ERROR	Did not find expected group ‘lkadmin’	
000208	ERROR	pam_setcred establish credentials failed (user %s, retval %d	**Cause**: Unable to establish valid login credentials for user {user}. The pam_setcred call returned: {retval}. **Action**: Check /var/log/security and /var/log/messages for more information.
000209	ERROR	pam_setcred delete credentials failed (user %s, retval %d	**Cause**: Unable to clear login credentials for user {user}. The pam_setcred call returned: {retval}. **Action**: Check /var/log/security and /var/log/messages for more information.
000902	ERROR	Error removing system name from loopback address line in /etc/hosts file. You must do this manually before starting the GUI server.	**Cause**: System name did not get removed from /etc/hosts file. **Action**: Remove system name manually then restart the GUI server, then enter the following: run <action name>
000918	ERROR	LifeKeeper GUI Server error during Startup	**Cause**: The GUI server terminated due to an abnormal condition.

Code	Severity	Message	Cause/Action
			Action: Check the logs for related errors and try to resolve the reported problem.

8.1.14. SAP HANA Recovery Kit Message Catalog

Use **Control F** to search for a specific error code in each catalog. To search for any error code, select the **Search** button at the top right of the screen.

Code	Severity	Message	Cause/Action
136002	ERROR	Usage: $usage	**Cause**: Invalid arguments provide the SAP HANA create script. **Action**: Please provide appropriat arguments in the form: <Resource Tag> <SAP SID> <HDB Instance> [Switchback Type] [Virtual IP Reso Tag]
136003	ERROR	END failed create of resource $tag on server $me with return value of $errcode.	**Cause**: Failure during SAP HANA resource creation. **Action**: Verify that SAP HANA Sy Replication is fully configured and enabled on both the primary and secondary replication sites and reattempt the resource creation operation.
136005	ERROR	An unknown error has occurred in utility rlslocks on server $me. View the LifeKeeper logs for details and retry the operation.	**Cause**: Failure of the LifeKeeper rlslocks utility during SAP HANA resource creation. **Action**: Resolve any issues found the LifeKeeper log file and reatten the resource creation operation.
136006	ERROR	END failed create of resource $tag on server $me with signal $sig.	**Cause**: Failure during SAP HANA resource creation due to a signal. **Action**: Review the LifeKeeper log

Code	Severity	Message	Cause/Action
			for details.
136008	ERROR	An unknown error has occurred in utility getlocks on server $me. View the LifeKeeper logs for details and retry the operation.	**Cause**: Failure of the LifeKeeper getlocks utility during SAP HANA resource creation. **Action**: Resolve any issues found the LifeKeeper log file and reatten the resource creation operation.
136009	ERROR	The SAP HANA product was not found in the directory $obj->{'UTIL_PATH'} on server $me.	**Cause**: Required SAP HANA bina could not be located during resou creation. **Action**: Verify that SAP HANA is correctly installed and configured servers in the cluster.
136010	ERROR	Failed to create resource as id $id already exists on system $me.	**Cause**: A LifeKeeper resource wit given ID already exists on the sys **Action**: Check whether the SAP H database is already protected by LifeKeeper.
136011	ERROR	Failed to create new tag $tag for SAP HANA resource on $me.	**Cause**: The provided SAP HANA resource tag is already in use by another LifeKeeper resource and t LifeKeeper newtag utility failed to create a new tag. **Action**: Choose a different tag na for the SAP HANA resource.
136012	ERROR	Failed creation of resource with $tag on system $me.	**Cause**: Failed to create the given HANA resource in LifeKeeper.

Code	Severity	Message	Cause/Action
			Action: Resolve any issues found the LifeKeeper log file and reatten the resource creation operation.
136014	ERROR	Failed to create resource dependency for parent $tag and child $virtual_ip_tag.	**Cause**: Failed to create a depend between the SAP HANA resource its dependent virtual IP resource. **Action**: Resolve any issues found the LifeKeeper log file and reatten the resource creation operation.
136015	ERROR	The info field for resource $tag could not be successfully generated using values [SID: $info_sid, Instance: $info_instance, Replication Mode: $info_repl_mode, Site Name: $info_site_name, Operation Mode: $info_oper_mode]. Please verify that SAP HANA System Replication is fully configured and enabled on both the primary and secondary systems before creating the SAP HANA resource.	**Cause**: An invalid value was found when creating the SAP HANA reso info field. **Action**: Verify that SAP HANA is properly installed and configured a that SAP HANA System Replicatio fully configured and enabled on al servers in the cluster.
136016	ERROR	The selected server $me is not the primary/source system for SAP HANA System Replication for the selected SID $sid and HDB instance $instance. Please select ‘Cancel’ and start this action on the primary/source HANA System Replication system.	**Cause**: Resource creation is initia on a secondary system in HANA System Replication. **Action**: Initiate the create action o primary system in HANA System Replication.
136031	ERROR	END failed extend of resource $target_tag on server $me with return value of $err_code.	**Cause**: Failure during SAP HANA resource extension. **Action**: Resolve any issues found the LifeKeeper log file and reatten the resource extension operation.

Code	Severity	Message	Cause/Action
136033	ERROR	Usage: $usage	**Cause**: Invalid arguments provide the SAP HANA extend script. **Action**: Please provide appropriat arguments in the form: <Template System> <Template Tag> <Switch Type> <Target Tag>
136035	ERROR	Template resource $template_tag on server $template_sys does not exist.	**Cause**: The template SAP HANA resource to be extended does not on the template server. **Action**: Verify that the SAP HANA resource that is being extended ex on the template server.
136036	ERROR	Resource with matching id $target_id already exists on server $me for App $app_type and Type $res_type.	**Cause**: An SAP HANA resource w the same LifeKeeper ID already ex on the target system. **Action**: Check whether the SAP H database is already protected by LifeKeeper on the target server.
136037	ERROR	Resource with matching tag $target_tag already exists on server $me for App $app_type and Type $res_type	**Cause**: An SAP HANA resource w the same LifeKeeper resource tag already exists on the target server **Action**: Check whether the SAP H database is already protected by LifeKeeper on the target server.
136039	ERROR	Error creating resource $target_tag on system $me.	**Cause**: Failed to create an equiva SAP HANA resource on the target server. **Action**: Resolve any issues found the LifeKeeper log file and reattem

Code	Severity	Message	Cause/Action
			the resource extension operation.
136040	ERROR	The target tag ($target_tag) and template tag ($template_tag) must be the same.	**Cause**: Resource tag name used primary system is different than resource tag name used on secon system. Both must be same. **Action**: While resource creation u same name as primary system tag secondary system tag.
136045	ERROR	Cannot extend resource $template_tag to server $me.	**Cause**: The SAP HANA canexten script indicates that the resource cannot be extended to the given ta system. **Action**: Resolve any issues found the LifeKeeper log file and reatten the resource extension operation.
136047	ERROR	Usage: $usage	**Cause**: Invalid arguments provide the SAP HANA canextend script. **Action**: Please provide appropriat arguments in the form: <Template System> <Template Tag>
136048	ERROR	Resource $template_tag does not exist on server $template_sys.	**Cause**: The template SAP HANA resource to be extended does not on the template server. **Action**: Verify that the SAP HANA resource that is being extended ex on the template server.
136049	ERROR	The system user $hana_user does not exist on server $me.	**Cause**: The SAP Administrative U for the SAP HANA database does

Code	Severity	Message	Cause/Action
			exist on the given server. **Action**: Verify that SAP HANA is properly installed and configured.
136050	ERROR	The user id for user $hana_user ($template_uid) on template server $template_sys is not the same as user id ($uid) on target server $me.	**Cause**: The user ID for the SAP Administrative User differs betwee template and target servers. **Action**: Verify that SAP HANA is properly installed and configured o servers in the cluster.
136051	ERROR	The group id for user $hana_user ($template_gid) on template server $template_sys is not the same as group id ($gid) on target server $me.	**Cause**: The group ID for the SAP Administrative User differs betwee template and target servers. **Action**: Verify that SAP HANA is properly installed and configured o servers in the cluster.
136052	ERROR	The home directory for user $hana_user ($template_home) on template server $template_sys is not the same as home directory ($user_home) on target server $me.	**Cause**: The home directory for the Administrative User differs betwee template and target servers. **Action**: Verify that SAP HANA is properly installed and configured o servers in the cluster.
136053	ERROR	The SAP HANA instance $instance does not exist for $sid on server $me.	**Cause**: Installation directories for given SAP HANA database instan could not be located. **Action**: Verify that SAP HANA is properly installed and configured o servers in the cluster.

Code	Severity	Message	Cause/Action
136055	ERROR	The SAP HANA site name $target_obj->{'site_name'} on server $me must be different from site name $template_obj->{'site_name'} on $template_obj->{'sys'}.	**Cause**: The SAP HANA System Replication site name is the same both the primary and secondary servers. **Action**: Stop the SAP HANA datal on the secondary server and use t hdbnsutil utility to re-register the secondary replication site using a different site name.
136056	ERROR	Unable to obtain SAP HANA System Replication parameters for database $instance on server $me. Please verify that SAP HANA System Replication is enabled and properly configured and that the database instance is running on all servers in the cluster.	**Cause**: SAP HANA System Replic parameters could not be determin the database on the given system **Action**: Verify that SAP HANA Sy Replication is enabled and properl configured and that the database i running on all servers in the cluste
136083	ERROR	Usage: $usage	**Cause**: Invalid arguments in the S HANA delete script. **Action**: Provide both a valid HANA LifeKeeper resource tag name and resource ID. Usage: delete -t <tag <id> [-U]
136160	ERROR	Unable to create SAP HANA object. The SID and instance for the SAP HANA database must be provided.	**Cause**: Either the SAP SID or the HANA instance name were missin while trying to create an SAP HAN object. **Action**: Inspect the LifeKeeper log for more details.
136161	ERROR	Unable to create SAP HANA object. Resource system name and tag name must be provided.	**Cause**: Either the system name or resource tag name were missing w

Code	Severity	Message	Cause/Action
			trying to create an SAP HANA obj **Action**: Inspect the LifeKeeper log for more details.
136162	ERROR	Could not find any information regarding resource $tag on $sys.	**Cause**: Failed to obtain informatio about the SAP HANA resource wit given tag. **Action**: Verify that a SAP HANA resource with the given tag exists the given server.
136168	ERROR	Unable to check status of SAP Host Agent on server $self->{'sys'}. Command \"$curr_cmd\" returned exit code $ret.	**Cause**: Failed to determine the sta of the SAP Host Agent processes the given server. **Action**: Inspect the SAP Host Age trace files (e.g., dev_saphostexec more details.
136174	ERROR	Unable to create SAP HANA object. The SID, instance, or one of the SAP HANA system replication values is missing.	**Cause**: Either the SAP SID, the SA HANA instance name, or one of th SAP HANA System Replication va were missing while trying to create SAP HANA object. **Action**: Inspect the LifeKeeper log for more details.
136178	ERROR	Attempt to register server $node as the secondary SAP HANA System Replication site for database $self->{'instance'} failed with exit code $ret.	**Cause**: Failed to register the given server as the secondary SAP HAN System Replication site for the giv database. **Action**: Inspect the SAP HANA tra files (e.g., nameserver_<hostname>.xxxxx.x

Code	Severity	Message	Cause/Action
			for more details.
136182	ERROR	Failed to $flg_action flag \"${HANA_FLAG_DATA_OUT_OF_SYNC}_$eqv_tag{$sys}\" on server $sys.	**Cause**: Failed to create or remove !HANA_DATA_OUT_OF_SYNC_< flag on the given server. **Action**: Inspect the LifeKeeper log for more details.
136190	ERROR	Failed to start SAP Host Agent processes on server $self->{'sys'}. Command \"$hostagent_cmd\" returned $ret.	**Cause**: Failed to start the SAP Ho Agent processes on the given serv **Action**: Inspect the SAP Host Age trace files (e.g., dev_saphostexec more details.
136191	ERROR	Failed to start SAP OS Collector process on server $self->{'sys'}. Command \"$oscol_cmd\" returned $ret.	**Cause**: Failed to start the SAP OS Collector process on the given ser **Action**: Inspect the SAP OS Colle trace files (e.g., dev_coll) for more details.
136193	ERROR	Takeover of SAP HANA System Replication for SAP HANA database $self->{'instance'} failed on server $node with exit code $ret.	**Cause**: Failed to register the given server as primary master for the g database in SAP HANA System Replication. **Action**: Inspect the SAP HANA tra files (e.g., nameserver_<hostname>.xxxxx.xx for more details.
136197	ERROR	Update of resource info field for $eqv_tag{$sys} on $sys failed with exit code $setinfo_ret. Current info: [$info]. Attempted new info: [$new_info].	**Cause**: Failed to update the info fi for the given resource on the given server.

Code	Severity	Message	Cause/Action
			Action: Inspect the LifeKeeper log for more details.
136202	EMERG	Failed to disable Autostart for SAP HANA instance $self->{'instance'} on server $sys with exit code $remexec_ret. Please manually set \"Autostart = 0\" in the instance profile $profile on $sys.	**Cause**: The value of the Autostart parameter could not be modified in given HDB instance profile on the server. **Action**: Edit the HDB instance pro manually and set “Autostart = 0”.
136205	ERROR	Failed start of SAP Start Service for SAP HANA database $instance on server $sys.	**Cause**: Failed to start SAP Start Service for the given SAP HANA database. **Action**: Inspect the SAP Start Ser trace files (e.g., sapstartsrv.log) fo more details.
136208	ERROR	LifeKeeper was unable to determine the SAP HANA System Replication mode for database $rem_obj->{'instance'} on server $rem_obj->{'sys'} while attempting to identify the previous primary replication site. Please resolve the issue and bring the SAP HANA resource in-service on the system where the database should be registered as primary master.	**Cause**: Failed to determine the SA HANA System Replication mode o given server. As a result, the prev primary replication site could not b identified. **Action**: Inspect the LifeKeeper log for more details.
136210	ERROR	Failed start of SAP Start Service for SAP HANA database $rem_obj->{'instance'} on server $rem_obj->{'sys'}. Unable to stop the database on the server where it is currently registered as primary master.	**Cause**: Failed to start SAP Start Service for the given SAP HANA database on the given server. As result, the database could not be stopped on the current primary SA HANA System Replication site. **Action**: Inspect the SAP Start Ser trace files (e.g., sapstartsrv.log) fo more details.

Code	Severity	Message	Cause/Action
136212	ERROR	Failed stop of SAP HANA database $rem_obj->{'instance'} on server $rem_obj->{'sys'} where it is currently registered as primary master.	**Cause**: Failed to stop the given SA HANA database on the given serv **Action**: Inspect the SAP HANA tra files and LifeKeeper log file for mo details.
136217	ERROR	Failed start of SAP HANA database $instance on server $sys.	**Cause**: Failed to start the given SA HANA database on the given serv **Action**: Inspect the SAP HANA tra files and LifeKeeper log file for mo details.
136220	ERROR	Unable to register $sys as a secondary SAP HANA System Replication site for database $instance. The host name of the current primary replication site was not provided.	**Cause**: The host name of the curr primary SAP HANA System Replic site was not provided when attemp to register a secondary replication **Action**: Inspect the LifeKeeper log for more details.
136233	EMERG	WARNING: A temporary communication failure has occurred between servers $self->{'sys'} and $rem_obj->{'sys'}. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please take one of the following resource hierarchies out of service: $self->{'tag'} on $self->{'sys'} or $rem_obj->{'tag'} on $rem_obj->{'sys'}. The server that the resource hierarchy is taken out of service on will become the secondary SAP HANA System Replication site.	**Cause**: A temporary communicatio failure has caused the given equiv HANA resources to both be broug service at the same time on their respective host servers. **Action**: Take the entire HANA res hierarchy out of service on the ser which should become the seconda replication site. Once the databas been stopped on that server, LifeKeeper will automatically regis as a secondary replication site du the next quickCheck cycle.
136234	EMERG	WARNING: SAP HANA database $self->{'instance'} is running and registered as primary master on both $self->{'sys'} and $rem_obj->{'sys'}. Manual intervention is	**Cause**: The given SAP HANA data

Code	Severity	Message	Cause/Action
		required in order to minimize the risk of data loss. To resolve this situation, please leave resource $self->{'tag'} in-service on $self->{'sys'} and stop database $self->{'instance'} on $rem_obj->{'sys'} by running the command \'su – $rem_obj->{'sid_admin'} -c \"sapcontrol -nr $rem_obj->{'instance_number'} -function Stop\"\' on that server. Once stopped, it will become the secondary SAP HANA System Replication site.	is running and registered as prima master on two cluster servers concurrently. **Action**: Use the command provide the message to stop the database the standby server. Once the data is stopped, LifeKeeper will automatically register the standby server as a secondary replication
136238	ERROR	Unable to create SAP HANA object. Resource system name and tag name must be provided.	**Cause**: Either the server name or resource tag name were missing w trying to create an SAP HANA obj **Action**: Inspect the LifeKeeper log for more details.
136239	ERROR	Failed start of SAP Start Service for SAP HANA database $obj->{'instance'} on server $obj->{'sys'}. Unable to determine status of the database on $obj->{'sys'}.	**Cause**: Failed to start SAP Start Service for the given SAP HANA database on the given server. As result, the status of the database not be determined. **Action**: Inspect the SAP Start Ser trace files (e.g., sapstartsrv.log) fo more details.
136263	ERROR	Usage: $usage	**Cause**: Invalid arguments in the S HANA resource restore script. **Action**: Please provide appropriat arguments in the form: -t <Resour Tag> -i <Resource ID>
136265	ERROR	Error getting resource information for $tag on server $me.	**Cause**: Failed to obtain informatio about the given SAP HANA resou on the given server.

Code	Severity	Message	Cause/Action
			Action: Verify that a SAP HANA resource with the given tag exists the given server.
136266	ERROR	The resource $tag protecting SAP HANA database $instance is not in sync. To protect the data LifeKeeper will not restore the resource on $me. Please restore the resource on the previous source server to allow the resync to complete.	**Cause**: SAP HANA System Replic was not in sync before attempting bring the database resource in-se on the backup server. **Action**: Bring the SAP HANA reso in-service on the previous primary server and allow the resynchroniz to complete.
136275	ERROR	Failed to determine SAP HANA System Replication mode for database $instance on server $me.	**Cause**: Failed to determine the SA HANA System Replication mode f given database on the given serve **Action**: Inspect the SAP HANA tra files (e.g., nameserver_<hostname>.xxxxx.x and the LifeKeeper log file for mor details.
136351	ERROR	Usage: $usage	**Cause**: Invalid arguments in the S HANA resource quickCheck script **Action**: Please provide appropriat arguments in the form: -t <Resour Tag> -i <Resource ID>
136353	ERROR	Error getting resource information for $tag.	**Cause**: Failed to obtain informatio about the given SAP HANA resou **Action**: Verify that a SAP HANA resource with the given tag exists the given server.

Code	Severity	Message	Cause/Action
136354	EMERG	The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKeeper and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: The SAP HANA System Replication mode for the given database was modified outside of LifeKeeper. **Action**: Bring the SAP HANA resc in-service on the server where it s be registered as primary master.
136363	EMERG	LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: Failed to determine the SA HANA System Replication mode c given server. **Action**: Inspect the LifeKeeper log for more details.
136450	ERROR	Usage: $usage	**Cause**: Invalid arguments provide the SAP HANA remove script. **Action**: Please provide appropriat arguments in the form: <Template <Template Id>
136454	ERROR	Error getting resource information for $tag.	**Cause**: Failed to obtain informatio about the given SAP HANA resou **Action**: Verify that a SAP HANA resource with the given tag exists the given server.
136456	ERROR	Failed start of SAP Start Service for SAP HANA database $instance on server $me.	**Cause**: Failed to start SAP Start Service for the given SAP HANA database. **Action**: Inspect the SAP Start Ser trace files (e.g., sapstartsrv.log) fo more details.

Code	Severity	Message	Cause/Action
136550	ERROR	Usage: $usage	**Cause**: Invalid arguments in the H resource recover script. **Action**: Provide both a valid HANA LifeKeeper resource tag name and resource ID. Usage: recover -d <ta -n <id>
136555	ERROR	Error getting resource information for $tag on server $me.	**Cause**: Failed to obtain informatio about the given HANA resource or given server. **Action**: Verify that the server is on LifeKeeper is running, and the HA resource exists.
136556	EMERG	The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKeeper and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: The SAP HANA System Replication mode for the given database was modified outside of LifeKeeper. **Action**: Bring the SAP HANA reso in-service on the server where it s be registered as primary master.
136558	EMERG	LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved.	**Cause**: The SAP HANA System Replication mode could not be determined for the given database the given server. **Action**: Inspect the SAP HANA tra files and LifeKeeper log file for mo details.
136559	ERROR	Resource $tag is no longer ISP on server $me. Exiting $cmd for $tag.	**Cause**: The given SAP HANA reso is no longer ISP on the given serv **Action**: Inspect the LifeKeeper log

Code	Severity	Message	Cause/Action
			for more details.
136650	ERROR	Usage: $usage	**Cause**: Invalid arguments provide the SAP HANA hana_stop_all_dbs script. **Action**: Please provide appropriat arguments in the form: hana_stop_all_dbs -t <tag>
136654	ERROR	Error getting resource information for $tag.	**Cause**: Failed to obtain informatio about the given SAP HANA resou **Action**: Verify that a SAP HANA resource with the given tag exists the given server.
136658	ERROR	Failed start of SAP Start Service for SAP HANA database $x->{'instance'} on server $x->{'sys'}. Could not determine status of SAP HANA DB on $x->{'sys'}.	**Cause**: Failed to start SAP Start Service for the given SAP HANA database. **Action**: Inspect the SAP Start Ser trace files (e.g., sapstartsrv.log) fo more details.
136661	ERROR	Failed stop of SAP HANA database $x->{'instance'} on server $x->{'sys'}.	**Cause**: Failed to stop the given SA HANA database on the given serv **Action**: Inspect the SAP HANA tra files and LifeKeeper log file for mo details.

9. SIOS Protection for Linux Support Matrix

Supported Operating Systems

Product	Supported Operating System	SIOS Prote				
		v9.1	v9.1.1	v9.1.2	v9.2	v9.2.1
SIOS Protection Suite for Linux	Red Hat Enterprise Linux 5	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit
	Red Hat Enterprise Linux 6	6.0 to 6.8 64-Bit (6.0 NOT Recommended)	6.0 to 6.8 64-Bit (6.0 NOT Recommended)	6.0 to 6.9 64-Bit (6.0 NOT Recommended)	6.0 to 6.9 64-Bit (6.0 NOT Recommended)	6.0 to 6.9 64-Bit (6.0 NOT Recommended)
	Red Hat Enterprise Linux 7	7.0 to 7.2 64-Bit	7.0 to 7.3 64-Bit	7.0 to 7.3 64-Bit	7.0 to 7.4 64-Bit (Some kernel versions do not support asynchronous mode.)	7.0 to 7.4 64-Bit (Some kernel versions do not support asynchronous mode.)
	Red Hat Enterprise Linux 8					

	SUSE Linux Enterprise Server (SLES) 11(*1)	11.0 to SP4 64-Bit	11.0 to SP4 64-Bit	11.0 to SP4 64-Bit	11.0 to SP4 64-Bit	11.0 to SP4 64-Bit
	SUSE Linux Enterprise Server (SLES) 12(*2)		12SP1 64-Bit	12SP1 to SP2 64-Bit	12SP1 to SP2 64-Bit	12SP1 to SP3 64-Bit
	SUSE Linux Enterprise Server (SLES) 15					

	Oracle Enterprise Linux 5	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit
	Oracle Linux 6	6.3 to 6.8 (including UEK R3) 64-Bit	6.3 to 6.8 (including UEK R3) 64-Bit	6.3 to 6.9 (including UEK R3) 64-Bit	6.3 to 6.9 (including UEK R3, R4) 64-Bit	6.3 to 6.9 (including UEK R3, R4) 64-Bit
	Oracle Linux 7	7.0 to 7.2 64-Bit	7.0 to 7.3 64-Bit	7.0 to 7.3 64-Bit	7.0 to 7.3 (including UEK R3, R4) 64-Bit	7.0 to 7.4 (including UEK R3, R4) 64-Bit (Some kernel versions do not support asynchronous mode.)
	Oracle Linux 8					
	CentOS 5	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit	5.0 to 5.11 64-Bit
	CentOS 6(*3)	6.0 to 6.8 64-Bit	6.0 to 6.8 64-Bit	6.0 to 6.9 64-Bit	6.0 to 6.9 64-Bit	6.0 to 6.9 64-Bit

		(6.0 DataKeeper Configuration NOT Supported)	(6.0 DataKeeper Configuration NOT Supported)	(6.0 DataKeeper Configuration NOT Supported)	(6.0 DataKeeper Configuration NOT Supported)	(6.0 DataKeeper Configuration NOT Supported)	
	CentOS 7	7.0 to 7.2 64-Bit	7.0 to 7.2 64-Bit	7.0 to 7.3 64-Bit	7.0 to 7.3 64-Bit	7.0 to 7.4 64-Bit (Some kernel versions do not support asynchronous mode.)	
	CentOS 8						

*1 *The kernel should be updated to 3.0.42-0.7.3 for SLES11SP2.*

*2 *The kernel should be updated to 4.4.82-6.9.1 for SLES12SP3.*

*3 *CentOS 6.0 – DataKeeper configuration is NOT Supported. This limitation was missing in the past release notes of each version.*

Supported Applications

Product	Supported Application	SIOS Protection Suite					
		v9.1	v9.1.1	v9.1.2	v9.2	v9.2.1	v9.2.2
Apache ARK	Apache Web Server	2.0, 2.2, 2.4	2.0, 2.2, 2.4	2.0, 2.2, 2.4	2.0, 2.2, 2.4	2.0, 2.2, 2.4	2.0, 2.2,
SAP ARK (See the SIOS Protection Suite for SAP Solution Page for more details)	SAP NetWeaver	7.0 Enhancement Package 1 – 2, 7.1, 7.3, 7.4	7.0 Enhancement Package 1 – 2, 7.1, 7.3, 7.4, 7.5	7.0 Enhancement Package 1 – 2, 7.1, 7.3, 7.4, 7.5	7.0 including Enhancement Package 1-3, 7.3 including Enhancement Package 1, 7.4, 7.5	7.0 including Enhancement Package 1-3, 7.3 including Enhancement Package 1, 7.4, 7.5	7.0 includ Enhancer Package 7.3 includ Enhancer Package 7.4, 7.
	SAP NetWeaver				7.51 innovation	7.51 innovation	7.51 innovati

	AS for ABAP				package	package	packag
	SAP S/ 4HANA Platform						
SAP DB ARK	SAP MaxDB	7.5, 7.6, 7.7, 7.8, 7.9	7.5, 7.6, 7.7, 7.8, 7.9	7.5, 7.6, 7.7, 7.8, 7.9	7.5, 7.6, 7.7, 7.8, 7.9	7.5, 7.6, 7.7, 7.8, 7.9	7.5, 7.6, 7.8, 7.
SAP HANA ARK	SAP HANA						
Postfix ARK	Postfix	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided the suppo Linux distributi
DB2 ARK	IBM Db2 Enterprise Server Edition (ESE) and Workgroup Server Edition (WSE)	9.0, 9.5, 9.7, 10.1, 10.5	9.0, 9.5, 9.7, 10.1, 10.5	9.0, 9.5, 9.7, 10.1, 10.5, 11.1	9.0, 9.5, 9.7, 10.1, 10.5, 11.1	9.0, 9.5, 9.7, 10.1, 10.5, 11.1	9.0, 9.5, 10.1, 10 11.1
	IBM Db2 Express Edition	9.0, 9.5, 9.7, 10.1, 10.5	9.0, 9.5, 9.7, 10.1, 10.5	9.0, 9.5, 9.7, 10.1, 10.5, 11.1	9.0, 9.5, 9.7, 10.1, 10.5, 11.1	9.0, 9.5, 9.7, 10.1, 10.5, 11.1	9.0, 9.5, 10.1, 10 11.1
	IBM Db2 Advanced, Standard and Community Editions						
PostgreSQL ARK	PostgreSQL	8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4	8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5	8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6	8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6	8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6	8.3, 8.4, 9.1, 9.2, 9.4, 9.5, 10
	EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server	8.3, 8.4, 9.1, 9.2, 9.3, 9.4, 9.5	8.3, 8.4, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6	8.3, 8.4, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6, 10	9.3, 9.4, 9.5, 9.6,10	9.3, 9.4, 9.5, 9.6,10	9.3, 9.4, 9.6,10,11
	EnterpriseDB Postgres Plus Solution Pack	9.1, 9.2, 9.3	9.1, 9.2, 9.3	9.1, 9.2, 9.3			
	EnterpriseDB Postgres Plus	8.4, 9.0	8.4, 9.0	8.4, 9.0			

	Standard Server						
	Symfoware Server Enterprise Edition	12.2, 12.3	12.2, 12.3	12.2, 12.3	12.2, 12.3	12.2, 12.3	12.2, 12.3
	Symfoware Server Standard Edition	12.2, 12.3	12.2, 12.3	12.2, 12.3	12.2, 12.3	12.2, 12.3	12.2, 12.3
	Symfoware Server Lite Edition	12.3	12.3	12.3	12.3	12.3	12.3
	FUJITSU Software Enterprise Postgres Advanced Edition	9.5	9.5	9.5	9.5	9.5	9.5, 10
	FUJITSU Software Enterprise Postgres Standard Edition	9.5, 9.6	9.5, 9.6	9.5, 9.6	9.5, 9.6	9.5, 9.6	9.5, 9.6, 1
	FUJITSU Software Enterprise Postgres Community Edition						10
	PowerGres Plus						
	PowerGres on Linux						
Oracle ARK	Oracle Database Enterprise Edition	10g R2, 11g, 11g R2, 12c, 12c R2	10g R2, 11g, 11g R2, 12c, 12c R2	10g R2, 11g, 11g R2, 12c, 12c R2	11g R2, 12c, 12c R2	11g R2, 12c, 12c R2, 18c	11g R2, 1 12c R2, 1 19c
	Oracle Database Standard Edition	10g R2, 11g, 11g R2, 12c	10g R2, 11g, 11g R2, 12c	10g R2, 11g, 11g R2, 12c	11g R2	11g R2	11g R2
	Oracle Database Standard Edition One	10g R2, 11g, 11g R2, 12c	10g R2, 11g, 11g R2, 12c	10g R2, 11g, 11g R2, 12c	11g R2	11g R2	11g R2
	Oracle Database Standard Edition 2	12c, 12c R2	12c, 12c R2	12c, 12c R2	12c, 12c R2	12c, 12c R2c 18c	12c, 12c 18c 19c

Sybase ASE ARK	SAP Adaptive Server Enterprise	15.5, 15.7, 16.0	15.5, 15.7, 16.0	15.5, 15.7, 16.0	15.7, 16.0	15.7, 16.0	15.7, 16.0
Samba ARK	Standard Samba file shares	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided the suppo Linux distributio
NFS Server ARK	Linux kernel version	Kernel 2.6 or later (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.)	Kernel 2.6 or later (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.)	Kernel 2.6 or later (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.)	Kernel 2.6 or later (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.)	Kernel 2.6 or later (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.)	Kernel 2.6 later (NFSv2 is supported RHEL 7/ CentOS 7 7 or later.
NAS ARK	NFS version of Mounted NFS file systems from an NFS server or Network Attached Storage (NAS) device	NFSv2 (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.) NFSv3 NFSv4	NFSv2 (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.) NFSv3 NFSv4	NFSv2 (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.) NFSv3 NFSv4	NFSv2 (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.) NFSv3 NFSv4	NFSv2 (NFSv2 is not supported on RHEL 7/ CentOS 7/OL 7 or later.) NFSv3 NFSv4	NFSv2 (NFSv2 is supported RHEL 7/ CentOS 7 7 or later. NFSv3 NFSv4
MySQL ARK	MySQL Community Edition, MySQL Enterprise Edition	5.1, 5.5, 5.6, 5.7	5.1, 5.5, 5.6, 5.7	5.1, 5.5, 5.6, 5.7	5.5, 5.6, 5.7	5.5, 5.6, 5.7, 8.0	5.5, 5.6, 8.0
	MariaDB	5.5, 10.0	5.5, 10.0	5.5, 10.0	5.5, 10.0	5.5, 10.0	5.5, 10.0
WebSphere MQ ARK	WebSphere MQ/IBM MQ	7.1, 7.5, 8.0, 9.0 (9.0.5 is not supported.)	7.1, 7.5, 8.0, 9.0 (9.0.5 is not supported.)	7.1, 7.5, 8.0, 9.0 (9.0.5 is not supported.)	7.5, 8.0, 9.0	7.5, 8.0, 9.0	7.5, 8.0, 9.1
Software RAID (md) ARK	RHEL5, OL5, CentOS5	5.0 to 5.11	5.0 to 5.11	5.0 to 5.11			
	RHEL6, OL6,	6.0 to 6.7	6.0 to 6.7	6.0 to 6.7	6.0 to 6.7	6.0 to 6.7	6.0 to 6.7

	CentOS6						
	SLES11	11.0 to SP1	11.0 to SP1	11.0 to SP1	11.0 to SP1	11.0 to SP1	11.0 to SI
VMDK as Shared Storage ARK	VMware vSphere						

Supported Virtualization

Product	Supported Virtualization	SIOS Protection Suite for						
		v9.1	v9.1.1	v9.1.2	v9.2	v9.2.1	v9.2.2	
SIOS Protection Suite for Linux	VMware vSphere	4.0, 4.1, 5.0, 5.1, 5.5, 6.0	4.0, 4.1, 5.0, 5.1, 5.5, 6.0	4.0, 4.1, 5.0, 5.1, 5.5, 6.0, 6.5	4.0, 4.1, 5.0, 5.1, 5.5, 6.0, 6.5	4.0, 4.1, 5.0, 5.1, 5.5, 6.0, 6.5	4.0, 4.1, 5.0, 5.1, 5.5, 6.0, 6.5	4 5 6
	KVM	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	provided with the supported Linux distributions	p v s L
	Oracle VM Server for x86	3(*4)	3(*4)	3(*4)	3(*4)	3(*4)	3(*4)	3
	Citrix XenServer	5.0 or later(*4)	5.0 or later(*4)	5.0 or later(*4)	5.0 or later(*4)	5.0 or later(*4)	5.0 or later(*4)	5 l
	Nutanix Acropolis Hypervisor（AHV）					20160925.57 20160925.90	20160925.57 20160925.90	2 2

*4 *It does not support shared storage.*

For a list of the disk array storage models and adapters currently supported by SPS in shared storage configurations as well as their type of certification, see the Supported Storage List.

For version requirements for SAP, see the SIOS Protection Suite for SAP Solution Page.

10. Supported Storage

Supported Storage List for LifeKeeper for Linux v9

Last updated: August 12, 2020

The table below is a list of LifeKeeper for Linux v9 supported storage and should be considered when configuring your environment.

About Supported Storage

Some types of storage used as shared storage in LifeKeeper require certification.

The following storage must be certified. Make sure the storage you plan to use is listed in the table below. Supported storage should be used for shared storage such as SCSI/FC/iSCSI/SAS that refer to the same data from multiple nodes for which LifeKeeper's IO fencing with SCSI-2/3 Reservation is required.

Configurations that do not require certification

- NAS storage (requires the NAS Recovery Kit)
- All disk devices that make up data replication by DataKeeper (both built-in and external)
- Virtual disks on shared storage protected by the VMDK Recovery Kit on vSphere
- Storage which is used in the environment where all of the requirements below are satisfied:
  - OS, hardware and platform has supported the storage
  - The LifeKeeper SCSI reservation feature is disabled
  - The fencing mechanism of LifeKeeper Quorum/Witness is configured

Note: If your shared storage meets the requirements above, it does not require SIOS certification.

Unsupported Hardware

Consumer storage that is connected via USB or IEEE1394 is not supported.

DELL

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)	Notes
DELL	Dell EqualLogic: PS4100 PS4110 PS4210 PS6100	ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	With a large number of luns (over 20), change the REMOTETIMEOUT setting in */etc/default/LifeKeeper* to REMOTETIMEOUT=600.
			Single Path	YES	DMMP ARK		

	PS6110 PS6210 PS6610		(DMMP)				
	MD3800f MD3820f MD3860f	FC	vSphere (RDM)	No	–	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
	MD3400 MD3420 MD3460	SAS	vSphere (RDM)	No	–	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
	SCv3000 Series: SCv3000 SCv3020	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	
			Single Path	No	–		
		SAS	vSphere	No	–	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
	SCv2000 Series: SCv2000 SCv2020 SCv2080	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	

			Single Path	No	–		
		SAS	vSphere	No	–	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
	Dell Storage: SC7020 SC5020 SC4020	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	
			Single Path	No	–		
	Dell Compellent: SC9000 SC8000	FC	vSphere (RDM)	No	–	–	
			Multi Path (DMMP)	No	–		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	
			Single Path	No	–		

Fujitsu

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)	Notes
Fujitsu	ETERNUS: DX60 S2 DX80 S2 DX90 S2 DX410 S2 DX440 S2 DX8100	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	If ETERNUS is used as a boot device, a file system error may occur when the boot device is not accessible from a primary node and switching to a
			Multi Path (DMMP)	YES	DMMP ARK		

	S2 DX8700 S2 DX60 S3 DX100 S3 DX200 S3 DX500 S3 DX600 S3 DX8700 S3 DX8900 S3 DX200F DX60 S4 DX100 S4 DX200 S4 DX500 S4 DX600 S4 DX60 S5 DX100 S5 DX200 S5 DX500 S5 DX600 S5 DX900 S5 AF250 AF650 AF250 S2 AF650 S2 AF250 S3 AF650 S3 DX8900 S4		Multi Path (EMPD)	YES	No Multipath ARK		
			Single Path	YES	No Multipath ARK		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	secondary node may fail. In order to avoid this, set a panic when a file system error occurs as shown below. Example: for xfs, configure the following and reboot the server. `echo 'fs.xfs.panic_mask=127' > /etc/sysctl.d/01-xfs.conf`
			Multi Path (EMPD)	YES	No Multipath ARK		
			Single Path	YES	No Multipath ARK		

Hitachi / Hitachi Vantara

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)
Hitachi / Hitachi Vantara	Hitachi Virtual Storage Platform G1000 Hitachi Virtual Storage Platform G1500 Hitachi Virtual Storage Platform F1500	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (HDLM)	YES	HDLM ARK	
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	YES	–	
		ISCSI	Multi Path (HDLM)	No	–	–
			Single Path	No	–	

	Hitachi Virtual Storage Platform	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (HDLM)	YES	HDLM ARK	
			Single Path	YES	–	
		ISCSI	Multi Path (HDLM)	No	–	–
			Single Path	No	–	
	Hitachi Virtual Storage Platform Mid Range Family: Hitachi Virtual Storage Platform F900 (VSP F900) Hitachi Virtual Storage Platform F700 (VSP F700) Hitachi Virtual Storage Platform F370 (VSP F370) Hitachi Virtual Storage Platform F350 (VSP F350) Hitachi Virtual Storage Platform G900 (VSP G900) Hitachi Virtual Storage Platform G700 (VSP G700) Hitachi Virtual Storage Platform G370 (VSP G370) Hitachi Virtual Storage Platform G350 (VSP G350) Hitachi Virtual Storage Platform G150 (VSP G150) Hitachi Virtual Storage Platform G130 (VSP G130)	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (HDLM)	YES	HDLM ARK	
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	YES	No Multipath ARK	
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0
			Single Path	YES	No Multipath ARK	
	Hitachi Virtual Storage Platform Mid Range Family: Hitachi Virtual Storage Platform F800 (VSP F800) Hitachi Virtual Storage Platform F600 (VSP F600) Hitachi Virtual Storage Platform F400 (VSP F400) Hitachi Virtual Storage Platform G800 (VSP G800) Hitachi Virtual Storage Platform G600 (VSP G600) Hitachi Virtual Storage Platform G400 (VSP G400) Hitachi Virtual Storage Platform G200 (VSP G200) Hitachi Virtual Storage Platform G100 (VSP G100)	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (HDLM)	YES	HDLM ARK	
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	YES	No Multipath ARK	
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0
			Single Path	YES	No Multipath ARK	
	Hitachi Unified Storage VM (HUS VM)	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0

			Multi Path (HDLM)	YES	HDLM ARK	
			Single Path	YES	No Multipath ARK	
		ISCSI	Multi Path (HDLM)	No	–	–
			Single Path	No	–	
	Hitachi Unified Storage 100 Series: Hitachi Unified Storage 150 (HUS 150) Hitachi Unified Storage 130 (HUS 130) Hitachi Unified Storage 110 (HUS 110)	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (HDLM)	YES	HDLM ARK	
			Single Path	YES	No Multipath ARK	
		ISCSI	Multi Path (HDLM)	No	–	–
			Single Path	No	–	

Hitachi

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)	Notes
Hitachi	BR1200	ISCSI	Multi Path (HDLM)	No	–	–	
			Multi Path (DMMP)	No	–		
			Single Path	No	–		
		SAS	Multi Path (RDAC)	YES	No Multipath ARK	v9.0 – v9.5.0	Both the single path and multipath configurations require the RDAC driver. Only the BR1200 configuration using the RDAC driver is supported. The BR1200 configuration using the HDLM (HDLM ARK) is not supported.
			Single Path (RDAC)	YES	No Multipath ARK		
	BR1250	ISCSI	Multi Path (HDLM)	No	–	–	

			Multi Path (DMMP)	No	–		
			Single Path	No	–		
		SAS	Multi Path (NECSPS)	YES	NEC SPS ARK	v9.0 – v9.5.0	Both the single path and multipath configurations require the NEC SPS ARK. The BR1250 configuration using HDLM (HDLM ARK) is not supported. LifeKeeper for Linux v9.1 or later can be used for NEC iStorage StoragePathSavior (NECSPS) Recovery Kit on RHEL7 environment
			Single Path (NECSPS)	YES	NEC SPS ARK		
	BR1650E BR 1650S	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (HDLM)	YES	HDLM ARK		
			Single Path	YES	–		
		ISCSI	Multi Path (HDLM)	No	–	–	
			Single Path	No	–		

HPE

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)	Notes
HPE	Nimble Storage: AF20 AF20Q AF40 AF60 AF80 AF1000 AF3000 AF5000 AF7000 AF9000 CS1000H CS1000 CS3000 CS5000	ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.2.2 – v9.5.0	
			Single Path	No	–		

	CS7000 HF20C HF20H HF20 HF40 HF60						
	3PAR: 3PAR StoreServ 8200 3PAR StoreServ 8400 3PAR StoreServ 8440 3PAR StoreServ 8450	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
	3PAR: 3PAR StoreServ 7400 3PAR StoreServ 7400c 3PAR StoreServ 7440 3PAR StoreServ 7440c 3PAR StoreServ 7450 3PAR StoreServ 7450c	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	**Note:** 3PAR StoreServ 7400 returns a reservation conflict with the default path checker setting. To avoid this conflict, set the following parameter in “/etc/default/LifeKeeper”: DMMP_REGISTRATION_TYPE=hba User friendly device mapping is not supported. Set the following parameter in “multipath.conf” “user_friendly_names no” * When using the vSphere(RDM) for multipath configurations, no configuration parameters need to be set.
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	**Note:** 3PAR StoreServ 7400 iSCSI returns a reservation conflict. To avoid this conflict, set the following parameter in “/etc/default/LifeKeeper”: DMMP_REGISTER_IGNORE=TRUE
			Single Path	No	–		
	3PAR: 3PAR StoreServ 7200 3PAR StoreServ 7200c	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	**Note:** 3PAR StoreServ 7200 returns a reservation conflict with the default path checker. To avoid this conflict, set the following parameter in “/etc/default/LifeKeeper”: DMMP_REGISTRATION_TYPE=hba
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	No	–	–	
			Single	No	–		

			Path				
	3PAR: 3PAR StoreServ 9450 3PAR StoreServ 20800 R2 3PAR StoreServ 20840 R2 3PAR StoreServ 20850 R2	FC	vSphere (RDM)	YES	No Multipath ARK	v9.2.1 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	No	–	–	
			Single Path	No	–		
	3PAR: 3PAR StoreServ 20450 3PAR StoreServ 20800 3PAR StoreServ 20840 3PAR StoreServ 20850	FC	vSphere (RDM)	YES	No Multipath ARK	v9.1 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	No	–	–	
			Single Path	No	–		
	3PAR StoreServ 10400	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	No	–	–	
			Single Path	No	–		
	3PAR StoreServ 10800	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	**Note:** 3PAR StoreServ 10800 FC returns a reservation conflict with the default path checker setting. To avoid this conflict, set the following parameter in “/etc/default/LifeKeeper”: DMMP_REGISTRATION_TYPE=hba
			Multi Path (DMMP)	YES	DMMP ARK		
			Single	No	–		

			Path				User friendly device mapping is not supported. Set the following parameter in “multipath.conf” “user_friendly_names no” * When using the vSphere(RDM) for multipath configurations, no configuration parameters need to be set.
		ISCSI	Multi Path (DMMP)	No	–	–	
			Single Path	No	–		
	MSA: MSA2050 MSA2052	FC	vSphere (RDM)	YES	No Multipath ARK	v9.2 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.2 – v9.5.0	
			Single Path	No	–		
		SAS	vSphere (RDM)	No	–	–	
			Multi Path (DMMP)	No	–		
			Single Path	No	–		
	MSA: MSA2042	FC	vSphere (RDM)	YES	No Multipath ARK	v9.1 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.1 – v9.5.0	
			Single Path	No	–		

		SAS	vSphere (RDM)	YES	No Multipath ARK	v9.1 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
	MSA: MSA1040 MSA2040	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.0 – v9.5.0	
			Single Path	No	–		
		SAS	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
	StoreVirtual (LeftHand) Series LeftHand OS version 12.6	FC	vSphere (RDM)	No	–	–	LeftHand OS (SAN/iQ) version 12.6 is supported in HPE StoreVirtual (LeftHand) storage.All StoreVirtual series are supported, including HC250 / HC380 and StoreVirtual VSA as the virtual storage appliance.See the release note of each version of LeftHand OS (SAN/iQ) to know the virtual storage applicance in detail.
			Multi Path (DMMP)	No	–		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK	v9.1 – v9.5.0	
			Single Path	No	–		
	HP P9500	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		

			Single Path	YES	No Multipath ARK		
	XP7 Storage System	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (HDLM)	YES	HDLM ARK		
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	YES	No Multipath ARK		

IBM

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)	Notes
IBM	IBM SAN Volume Controller * • IBM TotalStorage Proven	FC	vSphere (RDM)	No	–	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	YES	No Multipath ARK		
		ISCSI	Multi Path (DMMP)	No	–	–	
			Single Path	No	–		
	IBM Storwize V7000	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	YES	DMMP ARK Quorum/ Witness	v9.0 – v9.5.0	The IBM Storwize V7000 (Firmware Version 6.3.0.1) has been certified by partner testing using iSCSI (iscsi-

					Server Kit		initiator-utils6.2.0.872-34.el6.x86_64) with DMMP (device-mapper-1.02.66-6.el6, device-mapper-multipath-0.4.9-46.el6). The test was performed with LifeKeeper for Linux v7.5 using RHEL 6.2. **Restriction:** IBM Storwize V7000 must be used in combination with the Quorum/Witness Server Kit and STONITH. See the online documentation for the LifeKeeper version that you are using for more information. http://docs.us.sios.com/
			Single Path	No	–		
	IBM Storwize V3700	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path	No	–		
		ISCSI	Multi Path (DMMP)	No	–	–	
			Single Path	No	–		
		SAS	vSphere (RDM)	No	–	–	
			Multi Path (DMMP)	No	–		
			Single Path	No	–		
	IBM XIV Storage System	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	If you are creating more than 32 LUNs on an IBM XIV Storage System using LifeKeeper, please contact IBM for further details.
			Multi Path (DMMP)	YES	DMMP ARK		
			Single Path (DMMP)	YES	DMMP ARK		
		ISCSI	Multi Path	No	–	–	

			(DMMP)				
			Single Path	No	–		

Lenovo

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)
Lenovo	Lenovo Storage V3700 V2 Lenovo Storage V3700 V2 XP	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	No	–	
		ISCSI	Multi Path (DMMP)	No	–	–
			Single Path	No	–	
		SAS	vSphere (RDM)	No	–	–
			Multi Path (DMMP)	No	–	
			Single Path	No	–	
	IBM Storwize V3700	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	No	–	
		ISCSI	Multi Path (DMMP)	No	–	–
			Single Path	No	–	
		SAS	vSphere (RDM)	No	–	–
			Multi Path (DMMP)	No	–	
			Single Path	No	–	

NEC

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)	Notes

NEC	NEC iStorage: M10e M11e M100 M110 M300 M12e M120 M320	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	LifeKeeper for Linux v9.1 or later can be used for NEC iStorage StoragePathSavior (NECSPS) Recovery Kit on RHEL7 environment.
			Multi Path (NECSPS)	YES	NECSPS ARK		
			Single Path (NECSPS)	YES	NECSPS ARK		
		ISCSI	Multi Path (NECSPS)	YES	NECSPS ARK	v9.0 – v9.5.0	
			Single Path (NECSPS)	No	–		
		SAS	vSphere (RDM)	No	–	v9.0 – v9.5.0	
			Multi Path (NECSPS)	YES	NECSPS ARK		
			Single Path (NECSPS)	YES	NECSPS ARK		
	NEC iStorage: M310 M310F M320F M500 M510 M700 M710 M710F M520 M720 M720F	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0	
			Multi Path (NECSPS)	YES	NECSPS ARK		
			Single Path (NECSPS)	YES	NECSPS ARK		
		ISCSI	Multi Path (NECSPS)	YES	NECSPS ARK	v9.0 – v9.5.0	
			Single Path (NECSPS)	No	–		

Pure Storage

Vendor (*1)	Storage Model	Bus Type	Connect Type (*2)	Support Status	Required ARK (i.e. Multipath ARK)	Supported LK Version (*4)
Pure Storage	FA-400 Series	FC	vSphere (RDM)	YES	No Multipath ARK	v9.0 – v9.5.0
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	No	–	
		ISCSI	Multi Path (DMMP)	No	–	–

			Single Path	No	–	
	FlashArray//m Series //m10 //m20 //m50 //m70	FC	vSphere (RDM)	No	–	v9.0 – v9.5.0
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	No	–	
	FlashArray//x Series //x10 //x20 //x50 //x70	FC	vSphere (RDM)	No	–	v9.0 – v9.5.0
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	No	–	
	FlashArray//x Series //X10R2 //X20R2 //X50R2 //X70R2 //X90R2	FC	vSphere (RDM)	No	–	v9.3.2 – v9.5.0
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	No	–	
		iSCSI	vSphere (RDM)	No	–	v9.0 – v9.5.0
			Multi Path (DMMP)	YES	DMMP ARK	
			Single Path	No	–	

(*1) EMC storage for use

LifeKeeper acquired the E-Lab certification. See the matrix below for the EMC storage support.

http://www.emc.com/interoperability

(*2) Connecting Configurations

vSphere (RDM)	Connected a shared disk using Raw Device Mapping (RDM) on the VMWare ESX server
Multi Path (DMMP)	Multipath configuration of the shared disk using the Device Mapper Multipath driver
Multi Path (EMPD)	Multipath configuration of the shared disk using the ETERNUS Multipath Driver
Multi Path (HDLM)	Multipath configuration of the shared disk using the Hitachi Dynamic Link Manager driver
Multi Path (RDAC)	Multipath configuration of the shared disk using the IBM Redundant Disk Array Controller driver
Multi Path (NECSPS)	Multipath configuration of the shared disk using the iStorage StoragePathSavior driver
Single Path (DMMP)	Single path configuration of the shared disk using the Device Mapper Multipath driver
Single Path	Single path configuration of the shared disk using the IBM Redundant Disk Array

(RDAC)	Controller
Single Path	Single path connection not using a specific multipath driver

(*3) Multipath Software

The support information by maker must be referred about the connectivity information of each Multipath software.
Please refer to the release notes for the details about Multipath software support by LifeKeeper.

(*4) Supported versions of LifeKeeper

This document is written for LifeKeeper for Linux v9.
If you are using a previous version of LifeKeeper, you must refer to the Release Notes for that version for the
supported storage list.

11. Quick Start Guides

AWS Direct Connect Quick Start Guide Protection Suite for Linux in the AWS Cloud (SAP)

SAP HANA Recovery Kit

Chef Support Documentation

AWS VPC Peering Connections Quick Start Guide

DataKeeper for Linux Evaluation Guide

MySQL Cluster with Data Replication (i.e.

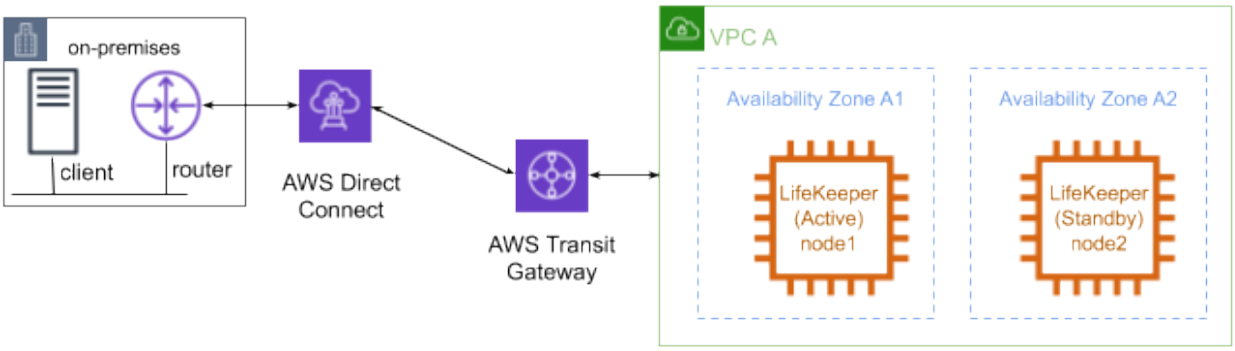
PostgreSQL Cluster with Shared Storage

Apache/MySQL Cluster Using Both Shared and Replicated Storage

11.1. AWS Direct Connect Quick Start Guide

Objective

With the release of AWS Transit Gateway, a route table scenario for the Recovery Kit for EC2 is now available with the configuration where the on-premises environment (on-premises in the figure below) using AWS Direct Connect is connected to the HA cluster nodes located in VPC (VPC A) via AWS Transit Gateway.



This document descries the requirements and basic operations for building connections from outside VPC with LifeKeeper for Linux v9.4.1.

This document does not cover the basic settings, operations, and technical details of LifeKeeper and Amazon Web Service (AWS). For terms, operations and technical information related to LifeKeeper and AWS, that are the prerequisites of this configuration, review the related documents and user websites.

> **Note: Amazon Web Services**, **Powered by Amazon Web Services** logo, **AWS Amazon EC2**, **EC2**, **Amazon Elastic Compute Cloud**, **Amazon Virtual Private Cloud**, **Amazon Route 53** and **Amazon VPC** are trademarks of Amazon.com, Inc. or its affiliates in the United States and other countries.

11.1.1. AWS Direct Connect Requirements

The following is a summary of requirements that should be met for an AWS environment and instances created on it.

Requirements for AWS Environment

Create a base environment on AWS to provide services. The requirements for using this configuration are as follows.

Amazon Virtual Private Cloud (VPC)

- VPC needs to be configured in AWS.
- The subnet where the primary instance is located and the subnet where the standby instance is located must be created in different Availability Zones (AZ).

Amazon Elastic Compute Cloud (EC2)

- At least 2 instances are required.
- A primary instance and a standby instance need to be configured to start with different AZ for each.
- Instances are connected to Elastic Network Interface (ENI).
- Instances are required to satisfy LifeKeeper's installation requirements.
- The AWS Command Line Interface (AWS CLI) needs to be installed in each of EC2 the instances. For the details, please refer to AWS Command Line Interface installation.
- You need to be able to access Amazon EC2 Web Services endpoint URL (EC2 URL) using https and Amazon EC2 metadata URL (http://169.254.169.254/) using http.

AWS Identity and Access Management (IAM)

In order for LifeKeeper to operate AWS, an IAM user or IAM role with the following access privilege is required. Cconfigure an EC2 IAM role or configure AWSCLI appropriately so that it can be accessed from root user of the EC2 instance.

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress

- ec2:DescribeRouteTables
- ec2:ReplaceRoute

AWS Transit Gateway

- The VPC with the cluster nodes and the on-premises environment where the clients are located must be connected via AWS Transit Gateway; not via Virtual Private Gateway.
- Enable the Default route table association and the Default route table propagation when creating AWS Transit Gateway.
- Connect VPC by creating Transit Gateway Attachment.
- Connect to AWS Direct Connect by selecting the created AWS Transit Gateway in the Gateway association configuration of Direct Connect Gateway. At this time, configure both the network address of the VPC where the cluster nodes are located and the virtual IP address in Allowed prefixes.

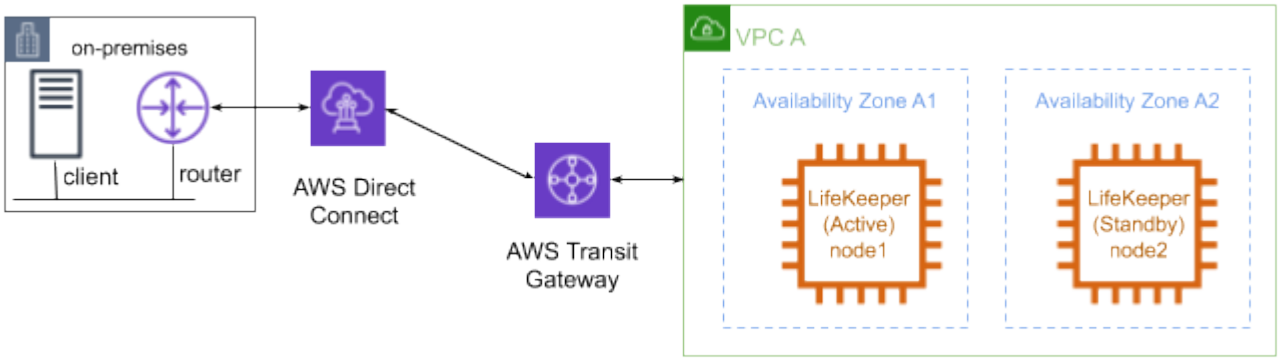
LifeKeeper Software Requirements

You need to install the same version of LifeKeeper software and patches on each server. The Application Recovery Kit (ARK) required for this configuration is shown below. For the specific LifeKeeper requirements, please refer to: SPS for Linux Technical Documentation and SPS for Linux Release Notes

- LifeKeeper IP Recovery Kit
- LifeKeeper Recovery Kit for EC2

11.1.2. AWS Direct Connect Setup Procedure

This section describes the general procedure to setup the environment shown below.



11.1.2.1. AWS Direct Connect Preparations

Create an environment that meets the AWS Direct Connect requirements. Install LifeKeeper on each instance and create a communication path between Node1 and Node2.

11.1.2.2. Creating Direct Connect Resources

Creating an IP Resource

- Create a virtual IP resource. The IP resource address must be outside the CIDR block managed by the VPC.

Creating an EC2 Resource

- Create EC2 resources. For the IP resource requested when creating resources, specify the resource created in “Create IP Resource” above. Specify the Route Table (Backend Cluster) as the EC2 resource type required when creating resources.

Creating Resources for Protected Services

- Create resources for the services you want to protect. If an IP resource is required for resource creation, specify the resource created in “IP Resource Creation” above. Configure resource dependencies so that the resources of the protected service are the parent resources and the EC2 resources are the child resources.

11.1.2.3. Configuring a Route Table

Configure a route table as shown below.

- Add the route information to the on-premises environment network to the route table of the VPC or subnet where the cluster nodes are located.

Destination Address	Target
On-premises network address	Created Transit Gateway

- Add the route information to the Virtual IP address in the route table of the Transit Gateway.

Destination Address	Target
Virtual IP address	VPC where the cluster nodes are located

- Configure the routing information of clients and routers in the on-premises environment so that the destination of packets to the network address and virtual IP address of the VPC where the cluster nodes are located are the Direct Connect.

Once configured, make sure that the client can access the private address and virtual IP address of the cluster server.

11.1.3. Considerations for Settings and Operations in AWS Direct Connect

Considering the Use of LifeKeeper I-O Fencing

Since the shared disk environment cannot be used in an AWS environment, you cannot use SCSI reservations to prevent a split-brain. IP resources may cause a split-brain as it uses the real IP resource with different IP addresses for each node.

For this reason, please consider the use of Quorum/Witness server or STONITH, an I/O fencing function of LifeKeeper to use this configuration safely.

Since you can implement I/O fencing separately without the Quorum server, if you use the TCP_REMOTE setting in Quorum mode, it is easy to implement in the cloud environment. For more details, please refer to the following:

- Quorom/Witness
- STONITH

AWS Direct Connect Known Issues and Troubleshooting

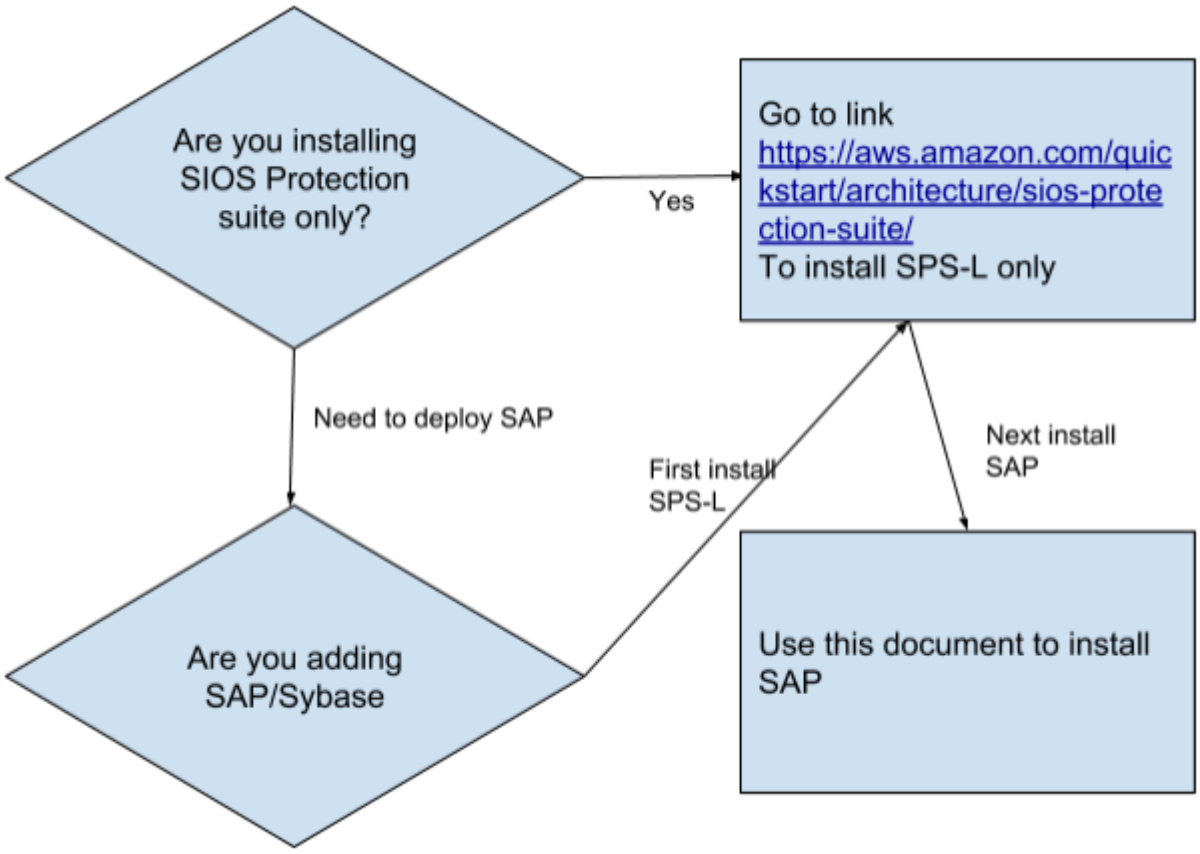
There are currently no Known Issues.

11.2. SIOS Protection Suite for Linux in the AWS Cloud (SAP)

Overview

This document will guide the user during SIOS Protection Suite for Linux installation (SPS).

Follow the quick decision matrix to understand how to install SPS for SAP environment.



Note: The link above is provided here so it can be copied.

11.2.1. Additional Steps to Configure SAP on SPS

Follow the steps below to configure SAP on SPS.

Step 1. Test the Deployment

To connect to the SPS-L nodes, you need to connect to Windows jumpbox. To connect to a Windows machine, you need to connect to remote desktop terminal.

In AWS console, select the windows jumpbox node that was created, click on **Actions** and click on **Connect**. You can now download remote desktop program to connect. You will also need to decrypt the password that will need to be used to login to the machine.

Once you are connected to Windows machine, we suggest you download Putty and VNC Viewer. Download them from these sites.

- **Putty** – www.putty.org
- **VNC Viewer** – https://www.realvnc.com/en/connect/download/viewer/

You can now use Putty to connect to the private IP address of each node, as well as VNC Viewer to connect to the node using the same private IP address. Note that the nodes are not accessible outside the windows jumpbox, but the nodes should be able to access the internet using the NAT gateway. (**Note:** If there are issues with the NAT gateway, make sure to check the security group rules/main route).

Once you have connected to one of the nodes, you can su to root using the password you created in the template earlier, and run the program vncserver. This will allow you to connect using VNC Viewer to that node in a graphical interface.

Code snippet for installing VNC Server
Run as root the command vncserver with the following options:

Enter password and repeat for confirmation
Set Read-Only password to **No**

Optionally edit /root/.vnc/config and add
securitytypes=none
vncserver -kill:1

Access to VNC is ipv4:5901 where 5901 is the port number specified.

Right click on the desktop and click on **Open Terminal**, and enter the command **/opt/LifeKeeper/bin/lkGUIapp**, that will connect to the LifeKeeper GUI. Login using root and password setup previously. You

will see the 2 nodes connected.

Now that you have reached this point, basic LifeKeeper 2 node is setup. Proceed with SAP installation and protection of SAP services using LifeKeeper.

Step 2. Configure Virtual IP

Now that SAP has been setup on the node, you can continue to setup LifeKeeper protecting SAP services and filesystems.

Amazon AWS Elastic Compute Cloud (EC2) setup

The AWS command line interface (cli) needs to be installed on each node. For details, please refer to AWS Command Line Interface Installation. All the EC2 instances must be able to access Amazon EC2 services endpoints using the protocols HTTP and HTTPS. In order to obtain metadata of Amazon EC2 instance, it is necessary to have an access to IP address 169.254.169.254 using the HTTP protocol.

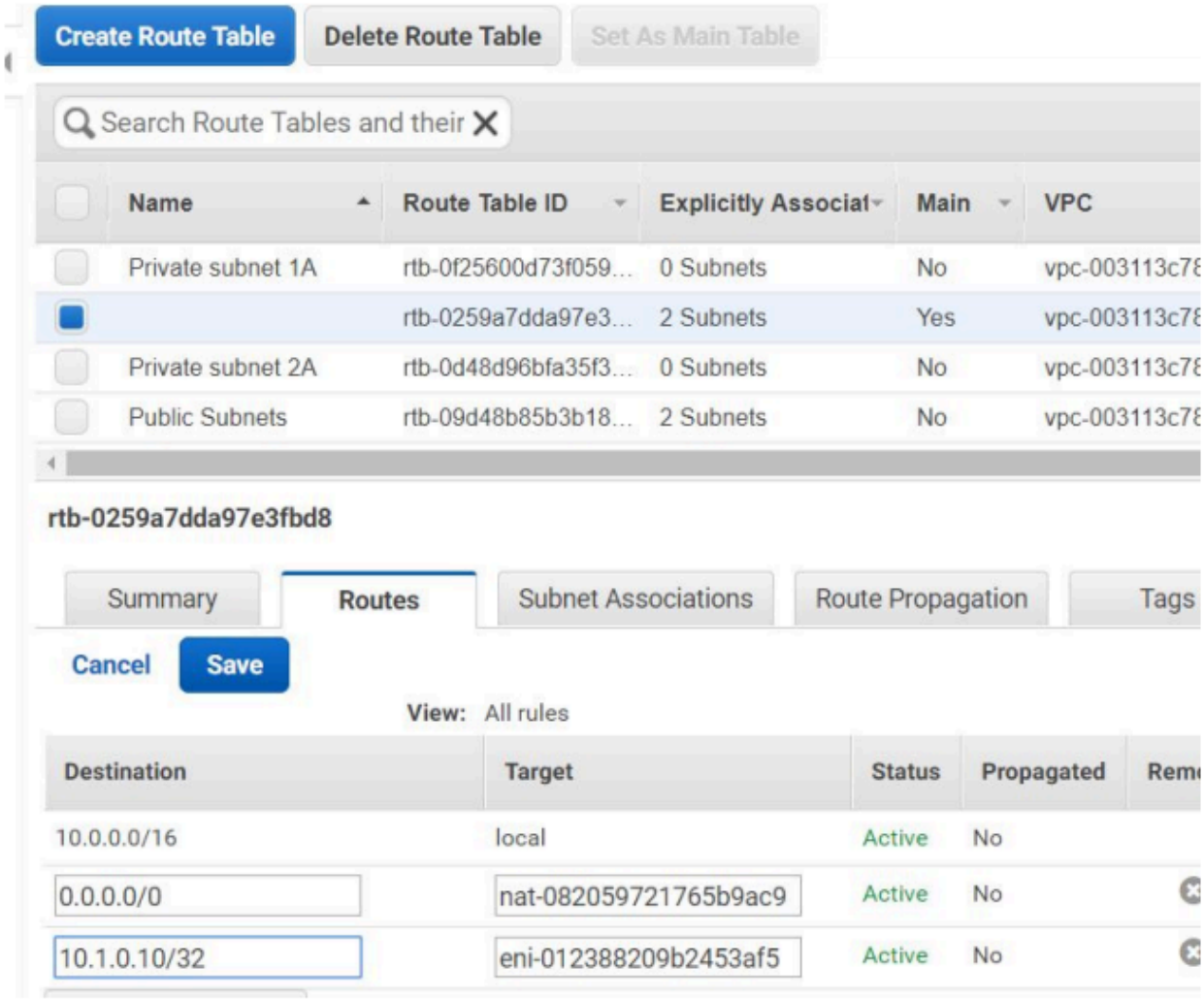
In order for LifeKeeper to operate AWS, an IAM user or IAM role with the following access privilege is required. Please configure an EC2 IAM role or configure AWS CLI appropriately so that it can be accessed from root user of the EC2 instance.

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress
- ec2:DescribeRouteTables
- ec2:ReplaceRoute

Create the virtual IP resource

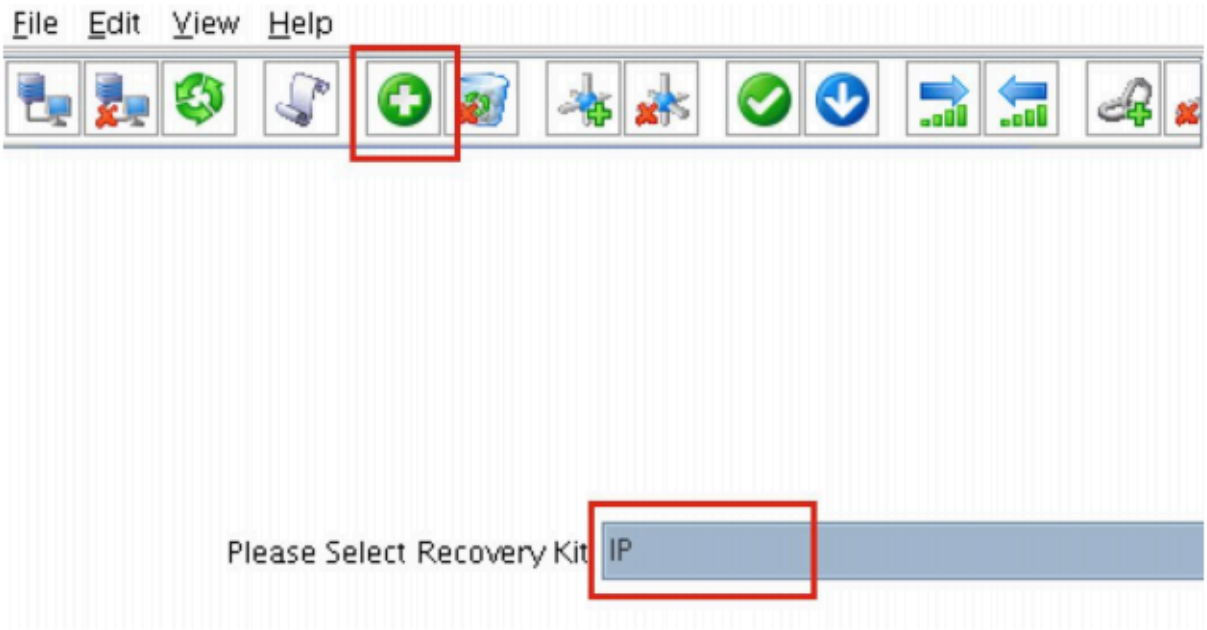
Determine the IP address. The IP address should be an IP address outside the CIDR block range of the current IP of the nodes. The IP address should be placed in the VPC route table for the node.

Note in the following diagram we placed the ip address of 10.1.0.10/32 and associated it to one of the nodes, using the eni- network adapter.



Edit /etc/default/LifeKeeper and set NOBCASTPING=1 to disable broadcast ping before continuing.

Click the **green plus** icon to create a new resource:



Follow the wizard to create the IP resource with these selections:

Select Recovery Kit: IP

Switchback Type: Intelligent

IP Resource: 10.1.0.10

Netmask: 255.255.255.0

Network Interface: eth0

IP Resource Tag: ip-10.1.0.10

Extend the IP resource with these selections:

Switchback Type: Intelligent

Template Priority: 1

Target Priority: 10

IP Resource: 10.1.0.10

Netmask: 255.255.255.0

Network Interface: eth0

IP Resource Tag: ip-10.1.0.10

The cluster will now look like this, with both Mirror and IP resources created:

Step 3. Setup SAP

Download the SAP software and setup on the node. You can access the SAP marketplace to download SAP software on each node.

There are a number of choices to setup SAP. The decision to implement one would depend on various factors, such as cost, experience and RAS (Reliability, Availability and Serviceability) factors.

- ASCS without NFS
- ASCS + ERS with NFS

Each configuration has advantages and disadvantages. We recommend that you work with SAP experts at your site, or you engage with SIOS Professional Services to determine the best fit for your environment.

Note: Future documentation will detail installation for HANA, all-in-one, using EFS (AWS Elastic File System) and Cloudwatch. There are also planned automated installation quick start scripts and using SAP Landscape Manager (LaMa) to manage the installation.

11.2.2. ASCS without NFS

Here are the steps to setup ASCS without NFS:

- General Setup Steps for ASCS without NFS
- Installing SAP
- Creating the SAP Resource Hierarchy

11.2.2.1. General Setup Steps for ASCS without NFS

1. Create Virtual IP, done in earlier steps
2. Create an EC2 resource and create as dependency for virtual IP, done in earlier steps
3. Install SAP on node 1 on “virtual hostname” based on “virtual IP”
4. Stopsap on node1
5. Use the LifeKeeper GUI to “In-service” the virtual IP to node 2, and Install SAP on node 2 on “virtual hostname” based on “virtual IP”
6. Stopsap on node 2 and modify profile files on both nodes (see below)
7. Use the LifeKeeper GUI to “In-service” the virtual IP back to node 1
8. Create replication resource for the mount points needed for SAP, done in earlier steps, as advised by SAP consultants
9. Startsap on node 1 and ensure SAP is working properly
10. In /etc/default/LifeKeeper add the follow entries to the end on both nodes:

    SAP_EXPERTMODE=1

    SAP_NFS_CHECK_IGNORE=1

    SAP_DB_CHECK_IGNORE=1

11. Re-run the SPS setup program to add the SAP Recovery Kit

    ./setup -k

    Select the recovery kit for SAP from the menu of available recovery kits using the arrow keys and pressing the <spacebar> to select, press <enter> to continue and complete the installation.

12. Create SAP resources following the SAP Recovery Kit guide

11.2.2.2. Installing SAP

1. ASCS should be installed based on “virtual hostname” based on “virtual IP”, which should have been added to host files during earlier installation steps. Please be sure to do so if they have not already been done prior to installing SAP.

- ◦ When installing SAP (specifically ASCS or SCS) you need to specify the SAPINST_USE_HOSTNAME=vip option. This is not required for ERS. (**Note:** Document the SAPINST_USE_HOSTNAME virtual IP address as it will be used later during creation of the SAP resources in LifeKeeper.)

  Note: Specify `sapinst SAPINST_USE_HOSTNAME=vip` where **vip** is the virtual IP that will float between the nodes.

  Run ./sapinst SAPINST_USE_HOSTNAME= *{hostname}*

- ◦ In seven phases, the **Core Services** should be created and started. If permission errors occur on *jdbcconnect.jar*, go to */sapmnt/STC/exe/uc/linuxx86_64* and make that directory as well as file *jdbcconnect.jar* writable (chmod 777 —-).

ASCS profiles should be pointing to local mount point containing /usr/sap, sapmnt or any other necessary for SAP files in your environment.

The ASCS and ERS instance profiles must be modified in order to prevent the enqueue server and enqueue replicator processes from being automatically restarted by the sapstart utility. After updating the instance profiles, SAP Start Service for each of these instances must be restarted in order for the changes to take effect. Follow the steps provided in Modify ASCS and ERS Instance Profile Settings, then return to this page to continue the setup process.

2. Sapstop SAP on node 1

3. Using the LifeKeeper GUI, right click on the IP address resource created for ASCS, select “In-Service” and select node 2 to switch the IP onto node 2.

4. Repeat step 1 to install SAP onto node 2 and ensure that it’s able to run correctly

5. Sapstop SAP on node 2

6. Using the LifeKeeper GUI, right click on the IP address resource created for ASCS, select “In-Service” and select node1 to switch the IP back onto node 1.

7. Sapstart SAP on node 1 and ensure that it’s able to run correctly

11.2.2.3. Creating the SAP Resource Hierarchy

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy**.

   A dialog box will appear with a drop-down list box with all recognized recovery kits installed within the cluster. Select **SAP** from the drop-down listing.

   Click **Next**

   When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information.

   If you click **Cancel** at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

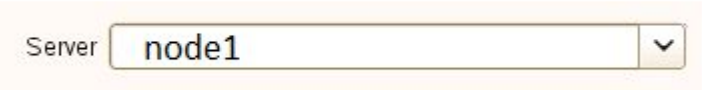
2. Select the **Switchback Type**. This dictates how the SAP instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either **intelligent** or **automatic**. **Intelligent switchback** requires administrative intervention to switch the instance back to the primary/original server. **Automatic switchback** means the switchback will occur as soon as the primary server comes back on line and re-establishes LifeKeeper communication paths.

   The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box.

   Click **Next**.

3. Select the Server where you want to place the SAP PAS, ASCS or SCS (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop-down list box.



4. Select the **SAP SID**. This is the system identifier of the SAP PAS, ASCS or SCS system being protected.

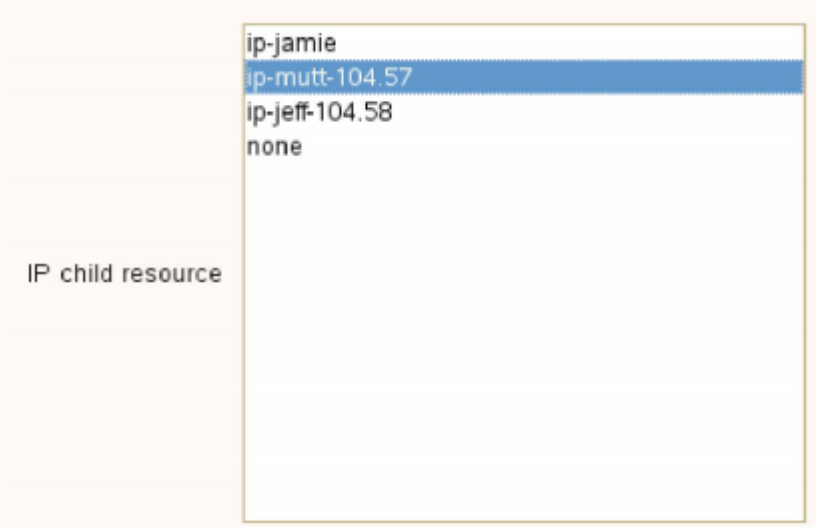
Click **Next**

5. Select the SAP Instance Name (ex. ASCS<*No*.>) (Core Instance first) for the SID being protected.

Click **Next**

Note: Additional screens may appear related to customization of Protection and Recovery Levels.

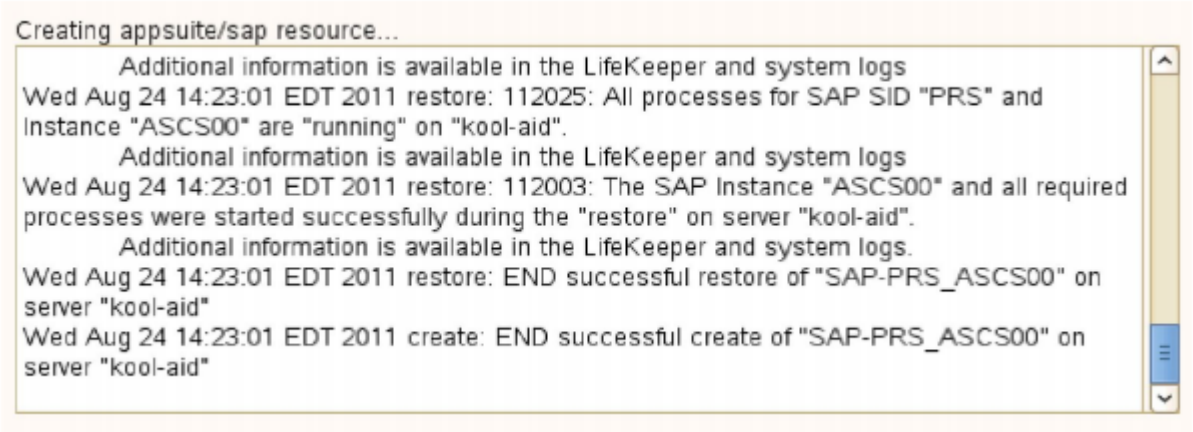
6. Select the **IP Child Resource**. This is typically either the Virtual Host IP address noted during SAP installation (SAPINST_USE_HOSTNAME) or the IP address needed for failover.



7. Select or enter the **SAP Tag**. This is a tag name that LifeKeeper gives to the SAP hierarchy. You can select the default or enter your own tag name. The default tag is SAP-<SID>_<ID>.

When you click **Create**, the ***Create SAP Resource Wizard*** will create your SAP resource.

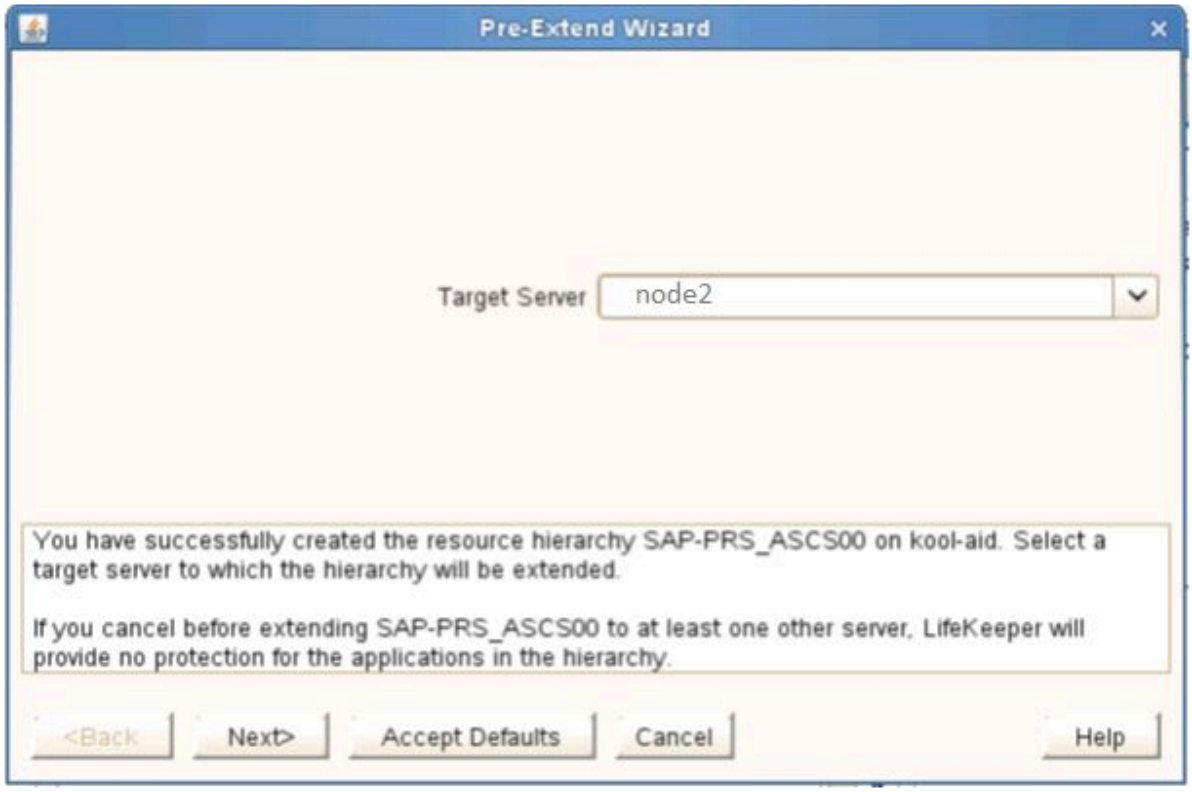
8. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your SAP resource hierarchy. If LifeKeeper detects a problem, an ERROR will appear in the information box. If the validation is successful, your resource will be created. There may also be errors or messages output from the SAP startup scripts that are displayed in the information box.



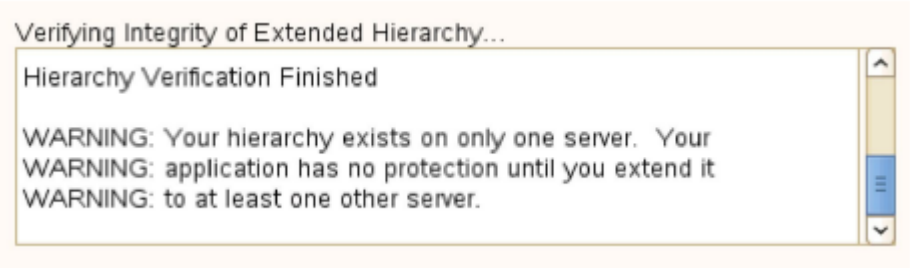
Click **Next**

9. Another information box will appear explaining that you have successfully created an SAP resource hierarchy, and you must **Extend** that hierarchy to another server in your cluster in order to place it under LifeKeeper protection.

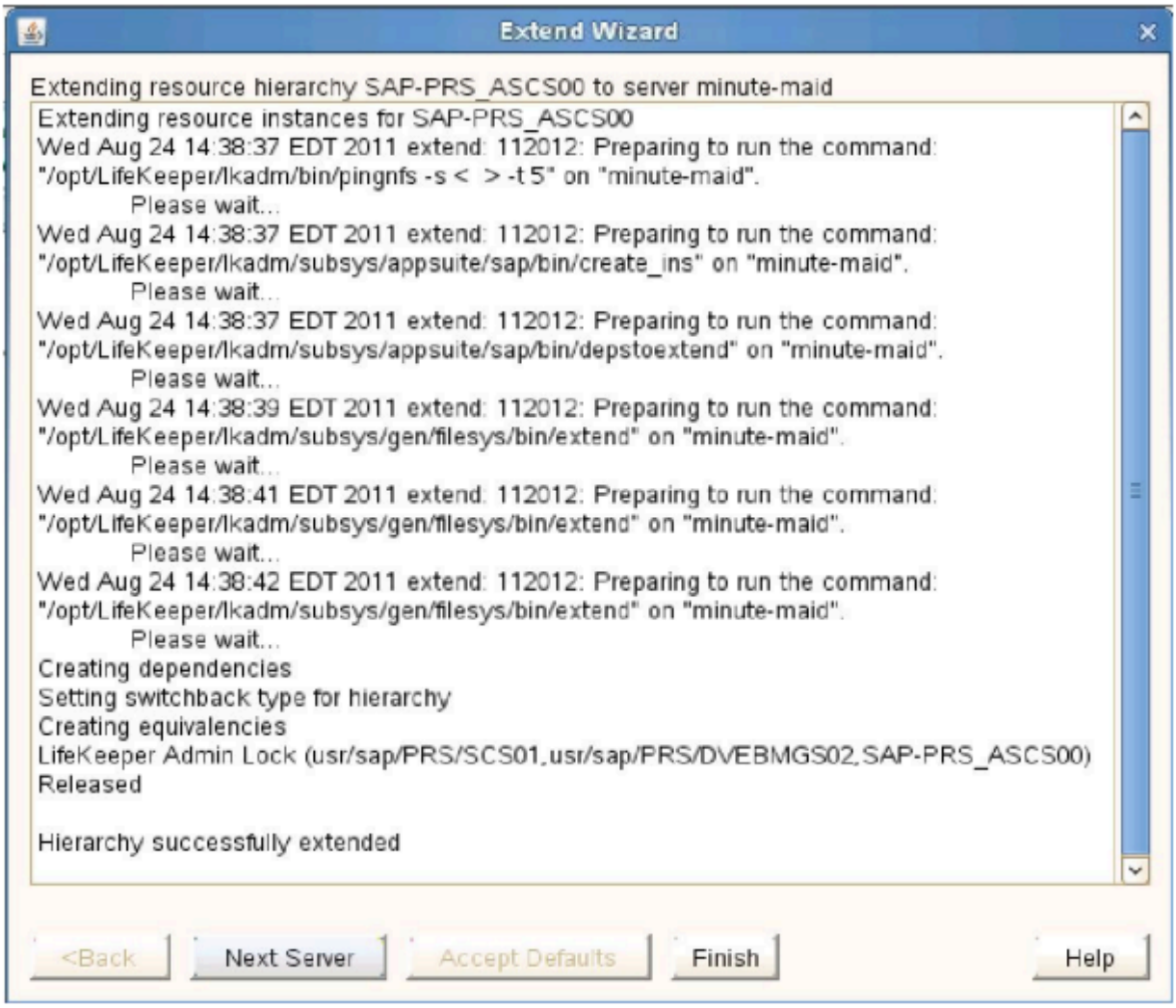
When you click **Next**, LifeKeeper will launch the ***Pre-Extend Wizard*** that is explained later in this section.



If you click **Cancel** now, a dialog box will appear warning you that you will need to come back and extend your SAP resource hierarchy to another server at some other time to put it under LifeKeeper protection.



10. The ***Extend Wizard*** dialog will appear stating **Hierarchy successfully extended**. Click **Finish**.



The **Hierarchy Integrity Verification** dialog appears. Once Hierarchy Verification finishes, click **Done** to exit the **Create Resource Hierarchy** menu selection.

Hierarchy with the Core as the Top Level

While SIOS Protection Suite can be used to protect the PAS and AAS servers, most customers would simply use them as independent standby servers with no additional HA on them. This guide does not cover their protection steps but you can refer to our SAP Recovery Kit documentation for details and steps.

11.2.3. ASCS + ERS with NFS

Here are the steps to setup ASCS + ERS with NFS:

- General Setup Steps
- Installing SAP
- Setting up NFS
- Creating an NFS Resource Hierarchy
- Creating the SAP Resource Hierarchy
- Create the ERS Resource

11.2.3.1. General Setup Steps

1. Create Virtual IP, done in earlier steps on node1, extend, done in earlier steps
2. Create EC2 resource and create as dependency for virtual IP, done in earlier steps
3. Install SAP on node1 on “virtual hostname” based on “virtual IP”
4. Stopsap on node 1
5. Use the LifeKeeper GUI to “In-service” the virtual IP to node 2, and Install SAP on node 2 on “virtual hostname” based on “virtual IP”
6. Stopsap on node 2 and modify profile files on both nodes (see below)
7. Use the LifeKeeper GUI to “In-service” the virtual IP back to node 1
8. Create replication resource for the mount points needed for SAP, done in earlier steps, as advised by SAP consultants
9. Startsap on node1 and ensure SAP is working properly
10. In /etc/default/LifeKeeper on both nodes add the follow entries to the end:

    SAP_EXPERTMODE=1

    SAP_NFS_CHECK_IGNORE=1

    SAP_DB_CHECK_IGNORE=1

11. Re-run the SPS setup program to add the SAP recovery kit

    Mount the sps.img file (downloaded as per earlier steps) using the following command:

    mount <PATH/IMAGE_NAME> <MOUNT_POINT> -t iso9660 -o loop

    Where PATH is the path to the image

    IMAGE_NAME is the name of the image

    MOUNT_POINT is the path to mount location

    Change to the sps.img mounted directory and type the following: ./setup -k

    You will now be shown a menu of recovery kits available. Select the recovery kit for SAP by using the arrow keys and pressing the <spacebar> to select, press <enter> to continue and complete

the installation.

12. Setup NFS servers

13. Copy file systems onto the SAP server and create replication resources on the file systems for redundancy and failover

14. Create NFS resources following the NFS Recovery Kit guide

    Simplified steps are given here

15. Create SAP resources following the SAP Recovery Kit guide

    Simplified steps are given here

11.2.3.2. Installing SAP

1. ASCS and ERS should be installed based on “virtual hostname” based on “virtual IP”, which should have been added to hosts files during earlier installation steps. Be sure to do so if they have not already been done prior to installing SAP.

- ◦ When installing SAP (specifically ASCS or SCS) you need to specify the SAPINST_USE_HOSTNAME=vip option. This is not required for ERS. (**Note:** Document the SAPINST_USE_HOSTNAME virtual IP address as it will be used later during creation of the SAP resources in LifeKeeper.)

  Note: Specify `sapinst SAPINST_USE_HOSTNAME=vip` where **vip** is the virtual IP that will float between the nodes.

  Run ./sapinst SAPINST_USE_HOSTNAME= *{hostname}*

- ◦ In seven phases, the **Core Services** should be created and started. If permission errors occur on *jdbcconnect.jar,* go to */sapmnt/STC/exe/uc/linuxx86_64* and make that directory as well as file *jdbcconnect.jar* writable (chmod 777 ---).

Enqueue replication should be configured and checked working based on SAP documentation and best practices.

2. ASCS and ERS profiles should be pointing to local mount point containing /usr/sap, sapmnt or any other necessary for SAP files in your environment, the actual files will be moved onto NFS mount points after it is installed and configured.

   The ASCS and ERS instance profiles must be modified in order to prevent the enqueue server and enqueue replicator processes from being automatically restarted by the sapstart utility. After updating the instance profiles, SAP Start Service for each of these instances must be restarted in order for the changes to take effect. Follow the steps provided in Modify ASCS and ERS Instance Profile Settings, then return to this page to continue the setup process.

3. Sapstop SAP on node 1

4. Using the LifeKeeper GUI, right click on the IP address resource created for ASCS, select “In-Service” and select node2 to switch the IP onto node 2.

5. Repeat step 1 to install SAP onto node 2 and ensure that it’s able to run correctly

6. Sapstop SAP on node 2

7. Using the LifeKeeper GUI, right click on the IP address resource created for ASCS, select “In-Service” and select node1 to switch the IP back onto node 1.

8. Sapstart SAP on node 1 and ensure that it’s able to run correctly

11.2.3.3. Setting up NFS

NFS server should have been installed on both cluster nodes prior to installation of SIOS as a prerequisite.

Create the NFS exports based on the SAP's requirements in your SAP design. Below are examples that may be use as a guide but not a representation of your SAP environment.

LifeKeeper maintains NFS share information using inodes; therefore, every NFS share is required to have a unique inode. Since every file system root directory has the same inode, NFS shares must be at least one directory level down from root in order to be protected by LifeKeeper. For example, referring to the information above, if the */usr/sap/trans* directory is NFS shared on the SAP server, the */trans* directory is created on the shared storage device which would require mounting the shared storage device as */usr/sap*. It is not necessarily desirable, however, to place all files under */usr/sap* on shared storage which would be required with this arrangement. To circumvent this problem, it is recommended that you create an */exports* directory tree for mounting all shared file systems containing directories that are NFS shared and then create a soft link between the SAP directories and the */exports* directories, or alternately, locally NFS mount the NFS shared directory. (**Note:** The name of the directory that we refer to as */exports* can vary according to user preference; however, for simplicity, we will refer to this directory as */exports* throughout this documentation.) For example, the following directories and links/ mounts for our example on the SAP Primary Server would be:

For the */usr/sap/trans* share	
Directory	**Notes**
/trans	created on share file system and shared through NFS
/exports/usr/sap	mounted to / (on shared file system)
/user/sap/trans	soft linked to */exports/usr/sap/trans*

The following directories and links for the <sapmnt>/<SAPSID> share would be:

For the <sapmnt>/<SAPSID> share	
Directory	**Notes**
/<SAPSID>	created on shared file systems and shared through NFS
/exports/sapmnt	mounted to / (on shared file system)
<sapmnt>/<SAPSID>	NFS mounted to *<virtual SAP server>:/exports/sapmnt/<SAPSID>*

Local NFS Mounts

The recommended directory structure for SAP in a LifeKeeper environment requires a locally mounted NFS share for one or more SAP system directories. If the NFS export point for any of the locally mounted NFS shares becomes unavailable, the system may hang while waiting for the export point to become available again. Many system operations will not work correctly, including a system reboot. You should be aware that the NFS server for the SAP cluster should be protected by LifeKeeper and should

not be manually taken out of service while local mount points exist.

To avoid accidentally causing your cluster to hang by inadvertently stopping the NFS server, please follow the recommendations listed in the NFS Considerations topic. It is also helpful to mount all NFS shares using the 'intr' mount option so that hung processes resulting from inaccessible NFS shares can be killed.

Location of <INST> Directories

Since the /usr/sap/<SAPSID> path is not NFS shared, it can be mounted to the root directory of the file system. The /usr/sap/<SAPSID> path contains the SYS subdirectory and an <INST> subdirectory for each SAP instance that can run on the server. For certain configurations, there may only be one <INST> directory, so it is acceptable for it to be located under /usr/sap/<SAPSID> on the shared file system. For other configurations, however, the backup server may also contain a local AS instance whose <INST> directory should not be on a shared file system since it will not always be available. To solve this problem, it is recommended that for certain configurations, the PAS's, ASCS's or SCS's /usr/sap/<SAPSID>/<INST>, /usr/sap/<SAPSID>/<ASCS-INST> or /usr/sap/<SAPSID>/<SCS-INST> directories should be mounted to the shared file system instead of /usr/sap/<SAPSID> and the /usr/sap/<SAPSID>/SYS and /usr/sap/<SAPSID>/<AS-INST> for the AS should be located on the local server.

For example, the following directories and mount points should be created for the ABAP+Java Configuration

Directory	Notes
*usr/sap/<SAPSID>/DVEBMS<**No.**>*	mounted to / (Replicated non-NFS file system)
*usr/sap/<SAPSID>/SCS<**No.**>*	mounted to / (Replicated non-NFS file system)
*usr/sap/<SAPSID>/ERS<**No.**>* *(for SCS instance)*	should be locally mounted on all cluster nodes or mounted from a NAS share (*should not be mounted on shared storage*)
*usr/sap/<SAPSID>/ASCS<**Instance No.** >*	mounted to / (Replicated from non-NFS file system)
*usr/sap/<SAPSID>/ERS<**No.**>* *(for ASCS instance)*	should be locally mounted on all cluster nodes or mounted from a NAS share (*should not be mounted on shared storage*)
*usr/sap/<SAPSID>AS<**Instance No.** >*	created for AS backup server

Mount NFS and Move File Systems

After mount points has been created for the main SAP file systems, mount them accordingly (required). At this point, stop all SAP services before proceeding with these steps.

mount /dev/sap/sapmnt /exports/sapmnt

mount /dev/sap/saptrans /exports/saptrans

Move Data to NFS

1. Edit */etc/exports* and insert the mount points for SAP's main directories.

```
/exports/sapmnt * (rw,sync,no_root_squash)

/exports/saptrans * (rw,sync,no_root_squash)
```

Example NFS Export

> Replace each occurrence of <nfsvip> with the appropriate virtual IP that will be used for that NFS share. Depending on the design of your SAP system, different virtual IP's may be used to share different filesystems.

more /etc/exports

/exports/sapmnt 10.2.0.69(rw,sync,all_squash,anonuid=0,anongid=1001)

/exports/sapmnt 10.2.0.11(rw,sync,all_squash,anonuid=0,anongid=1001)

/exports/usr/sap/<instance name>/ASCS01 10.2.0.69(rw,sync,all_squash,anonuid=0,anongid=1001)

/exports/sap/<instance name>/ASCS01
10.2.0.11(rw,sync,all_squash,anonuid=0,anongid=1001)

more /etc/fstab

#

/etc/fstab

Created by anaconda on Mon Nov 9 20:20:10 2015

#

Accessible filesystems, by reference, are maintained under '/dev/disk'

See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info

#

UUID=367df610-4210-4a5a-8c8d-51ddf499fc17 / xfs defaults 0 0

/dev/xvdb swap swap defaults 0 0

/dev/xvdc /tmp xfs nodev,nosuid,noexec,relatime 0 0

/dev/xvdp1 /var xfs defaults 0 0

/dev/xvdp2 /var/log xfs defaults 0 0

/dev/xvdp3 /var/log/audit xfs defaults 0 0

/dev/xvdp4 /home xfs defaults,nodev 0 0

/tmp /var/tmp none bind,nodev,nosuid 0 0

/dev/xvdj /usr/sap xfs defaults 0 0

/dev/xvdg /exports/usr/sap/P4G/ASCS01 xfs defaults 0 0

/dev/xvdh /usr/sap/P4G/D00 xfs defaults 0 0

/dev/xvdi /sapcd xfs defaults 0 0

/dev/xvdk /exports/sapmnt xfs defaults 0 0

<nfsvip>:/exports/usr/sap/P4G/ASCS01 /usr/sap/**<instance name>**/ASCS01 nfs nfsvers=3,proto=udp,rw,sync,intr,bg 0 0

<nfsvip>:/exports/sapmnt /sapmnt nfs nfsvers=3,proto=udp,rw,sync,intr,bg 0 0

<nfsvip>:/exports/usr/sap/P4G/ASCS01 /usr/sap/PG4/ASCS01 nfs nfsvers=3,proto=udp,rw,sync,intr,bg 0 0 (**Note:** This ERS entry will only be present if using an ERSv2 configuration with a shared ERS filesystem.)

2. Start the NFS server using the `rcnfsserver start` command (this is for SLES; for Red Hat perform `service nfs start`). If the NFS server is already active, you may need to do an "`exportfs -va`" to export those mount points.

3. On both node1 & 2, execute the following mount commands (**note the usage of udp; this is important for failover and recovery**), ensuring you are able to mount the NFS shares.

mount {virtual ip}:/exports/sapmnt/<PG4> /sapmnt/<PG4> -o rw,sync,bg,intr,udp

mount {virtual ip}:/exports/saptrans /usr/sap/trans -o rw,sync,bg,intr,udp

4. From node 1, copy the necessary file systems from the /usr/sap and /sapmnt or any other required files into the NFS mount points, mounted from the NFS servers onto node 1.

5. Log in to SAP and start SAP (after su to stcadm).

```
startsap sap{No.}
```

6. Make sure all processes have started.

```
ps –ef | grep en.sap (2 processes)

ps –ef | grep ms.sap (2 processes)

ps –ef | grep dw.sap (17 processes)
```

“SAP Logon” or “SAP GUI for Windows” is an SAP supplied Windows client the Windows client. The program can be downloaded from the SAP download site. The virtual IP address may be used as the “Application Server” on the **Properties** page. This ensures that a connection to the primary machine where the virtual ip resides is active.

7. If not already done, create the Data Replication Cluster resource on the NFS shares mount points to replicate the data from node1 to node2.

11.2.3.4. Creating an NFS Resource Hierarchy

To create a resource instance from the primary server, you should complete the following steps:

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy**.

   To change a selection already entered or if an error message is encountered during any step in the creation of your NFS resource hierarchy, use the **Back** button to change your selection or make corrections (assuming the **Back** button is enabled).

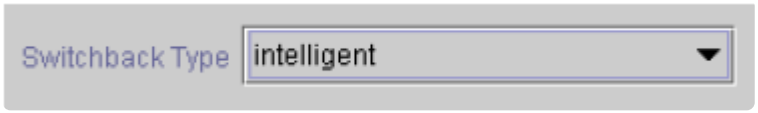
   A dialog box will appear with a drop-down menu listing all recognized Recovery Kits installed within the cluster. Select **NFS** from the drop-down menu

   Click **Next** to proceed to the next dialog box.

   Note: If you click the **Cancel** button at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

2. Select the **Switchback Type**. This dictates how the NFS instance will be switched back to the primary server when it comes back into service after a failover to the backup server. Choose either **Intelligent** or **Automatic**. **Intelligent switchback** requires administrative intervention to switch the instance back to the primary/original server. **Automatic switchback** means the switchback will occur as soon as the primary server comes back on line and is under LifeKeeper protection

   The switchback type can be changed later, if desired, from the **General** tab of the **Resource Properties** dialog box.

3. Select the **Server** where you want to create the NFS resource (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop-down men

   Click **Next** to proceed to the next dialog box.

4. The **Export Point** dialog displays a drop-down list of export points for NFS file systems that meet the following criteria:

   - The export point has been exported by NFS.
   - The export point is on a shared drive.
   - If the underlying file system is LifeKeeper-protected, it must be in service and have the highest priority on the server selected on the **Server** dialog.

   - NFSv4 criteria:
     - º If an

   NFS v4 root export is already being protected, no choices will be provided (there should only be one v4 and a mixture of V2/v3 with v4 cannot be protected).

   -

   º If an

NFS v2/v3 is already being protected, no NFS v4 will be listed in the choices.

   -

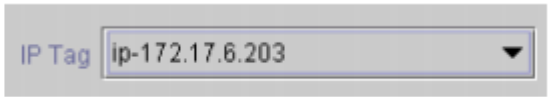
   º If nothing is protected, then the list could contain both v2/v3 and v4.

   Select the NFS export point to be protected from the drop-down list.

   Click **Next** to proceed to the next dialog box.

5. The **IP Tag** dialog displays a drop-down list of tags corresponding to virtual IP addresses currently under LifeKeeper protection and in service on the server where the NFS resource is being created. Select the **tag** for the virtual IP address used by clients to access the protected NFS file system.

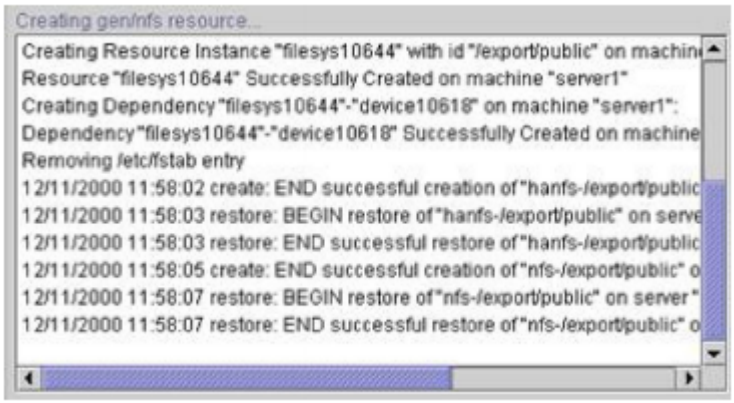
   Note: At this point, LifeKeeper will check to ensure that there is a protected IP resource available. It will also validate that you have provided valid data to create your NFS resource hierarchy. If LifeKeeper detects a problem with either of these validations, an ERROR box will appear on the screen. If the directory paths are valid but there are errors with the NFS configuration itself, you may pause to correct these errors and continue with the hierarchy creation. You may even pause to create any LifeKeeper IP resources that are required.

Note: If you are using other LifeKeeper Recovery Kits that have virtual IP address dependencies, you should create a different virtual IP address for the NFS resource. Otherwise, if the virtual IP resource fails over to a backup server, all of the resources that depend on that IP resource will fail over at the same time.

Click **Next** to proceed to the next dialog box

6. Select or enter the **NFS Tag**. This is a tag name given to the NFS hierarchy. You can select the default or enter your own tag name.

When you click the **Create** button, the ***Create Resource Wizard*** will create your NFS resource.



When you click **Next**, LifeKeeper will launch the ***Pre-Extend Wizard*** that is discussed in Extending Your Hierarchy. **Note:** The NFS resource hierarchy should be created successfully at this point. However, error messages may be encountered indicating that the new NFS instance has failed to start correctly. Note that the new NFS hierarchy must be started (In Service) before it can be extended to another system.

A failure to start may remove the hierarchy, but if not, you may pause at this point and correct the problem based on the error message displayed. If the errors are not correctable, you will only be given the choice to cancel which cancels the resource create.

Bring the new hierarchy In Service before proceeding with extending your hierarchy.

*** Repeat the steps above to create additional resource hierarchy for each NFS share.

Notes: Disable automatic startup of nfs-server.service after creating NFS resources on RHEL 7.1 or later and SLES12 SP1 or later. Since it is necessary for rpcbind.service to be running at the startup of NFS resources, configure rpcbind.service to start automatically.

11.2.3.5. Creating the SAP Resource Hierarchy

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the drop-down menu, select **Create Resource Hierarchy**.

   A dialog box will appear with a drop-down list box with all recognized recovery kits installed within the cluster. Select **SAP** from the drop-down listing.

   Click **Next**.

   When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is especially helpful should you encounter an error that might require you to correct previously entered information.

   If you click **Cancel** at any time during the sequence of creating your hierarchy, LifeKeeper will cancel the entire creation process.

2. Select the **Switchback Type**. This dictates how the SAP instance will be switched back to this server when it comes back into service after a failover to the backup server. You can choose either **intelligent** or **automatic**. **Intelligent switchback** requires administrative intervention to switch the instance back to the primary/original server. **Automatic switchback** means the switchback will occur as soon as the primary server comes back on line and re-establishes LifeKeeper communication paths.

   The switchback type can be changed later from the **General** tab of the **Resource Properties** dialog box.

   Click **Next**

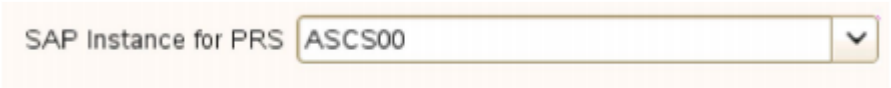
3. Select the Server where you want to place the SAP PAS, ASCS or SCS (typically this is referred to as the primary or template server). All the servers in your cluster are included in the drop-down list box.

4. Select the **SAP SID**. This is the system identifier of the SAP PAS, ASCS or SCS system being protected.

Click **Next**.

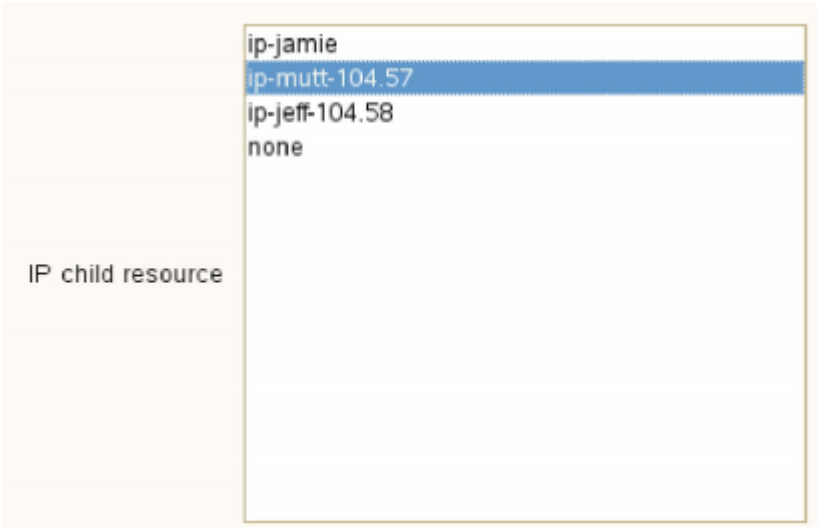
5. Select the SAP Instance Name (ex. ASCS<*No*.>) (Core Instance first) for the SID being protected.



Click **Next**.

Note: Additional screens may appear related to customization of Protection and Recovery Levels.

6. Select the **IP Child Resource**. This is typically either the Virtual Host IP address noted during SAP installation (SAPINST_USE_HOSTNAME) or the IP address needed for failover.

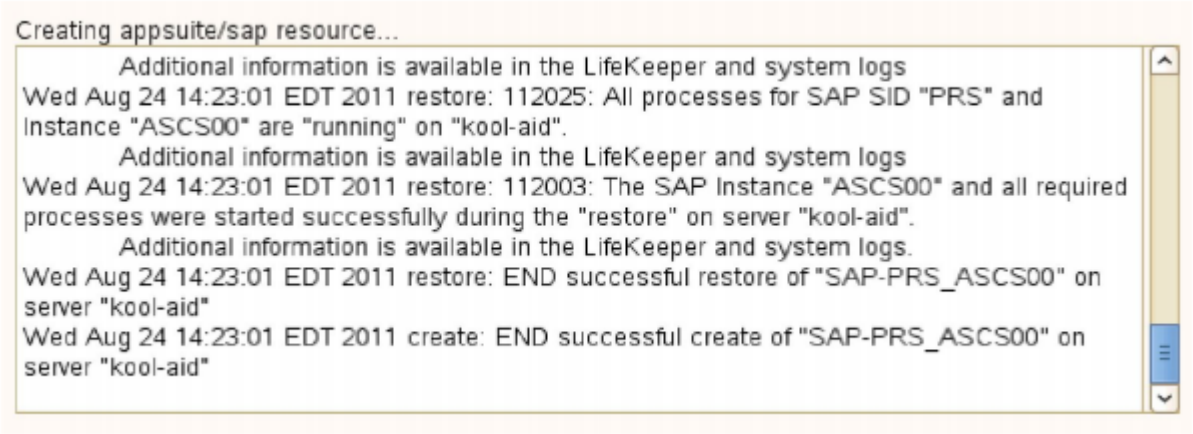


7. Select or enter the **SAP Tag**. This is a tag name that LifeKeeper gives to the SAP hierarchy. You can select the default or enter your own tag name. The default tag is *SAP-<SID>_<ID>*.

When you click **Create**, the ***Create SAP Resource Wizard*** will create your SAP resource.

8. At this point, an information box appears and LifeKeeper will validate that you have provided valid data to create your SAP resource hierarchy. If LifeKeeper detects a problem, an ERROR will

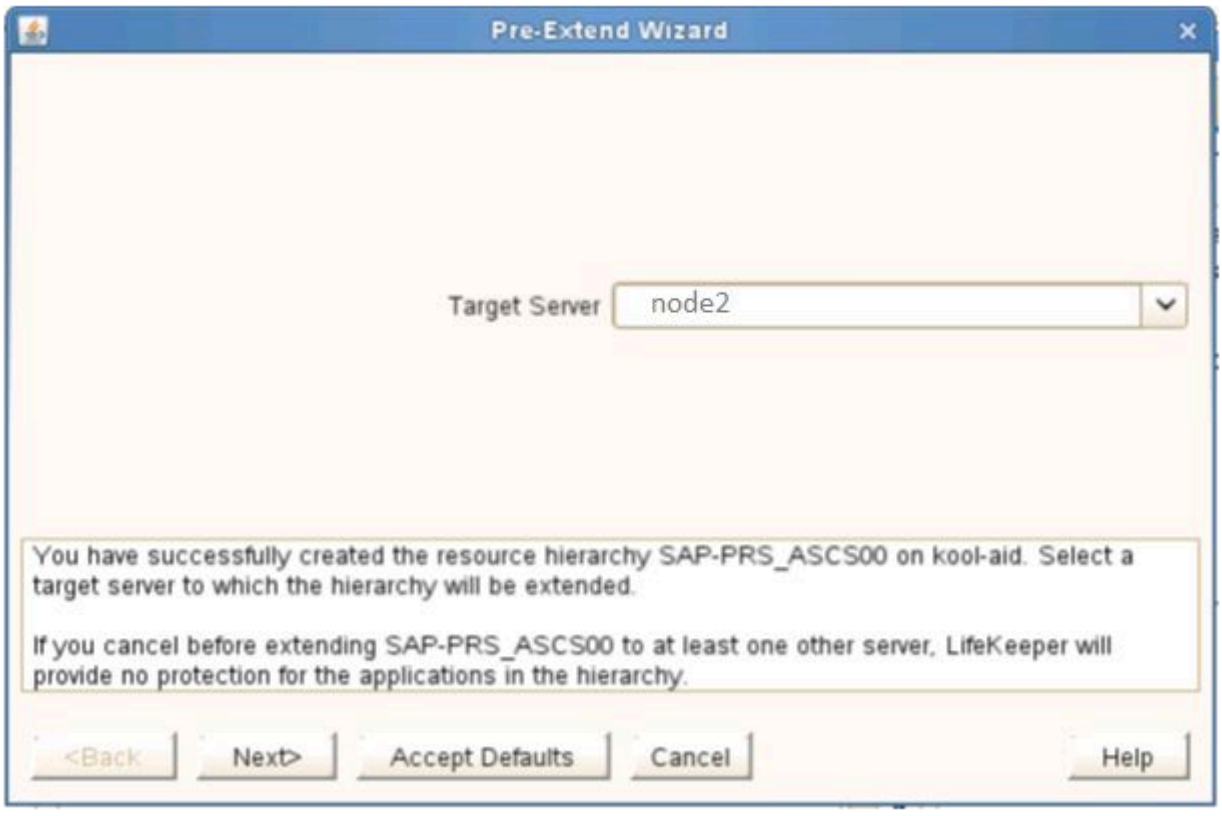
appear in the information box. If the validation is successful, your resource will be created. There may also be errors or messages output from the SAP startup scripts that are displayed in the information box.



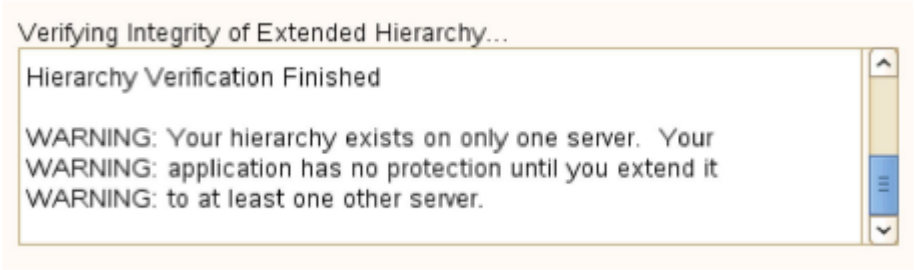
Click **Next**.

9. Another information box will appear explaining that you have successfully created an SAP resource hierarchy, and you must **Extend** that hierarchy to another server in your cluster in order to place it under LifeKeeper protection.

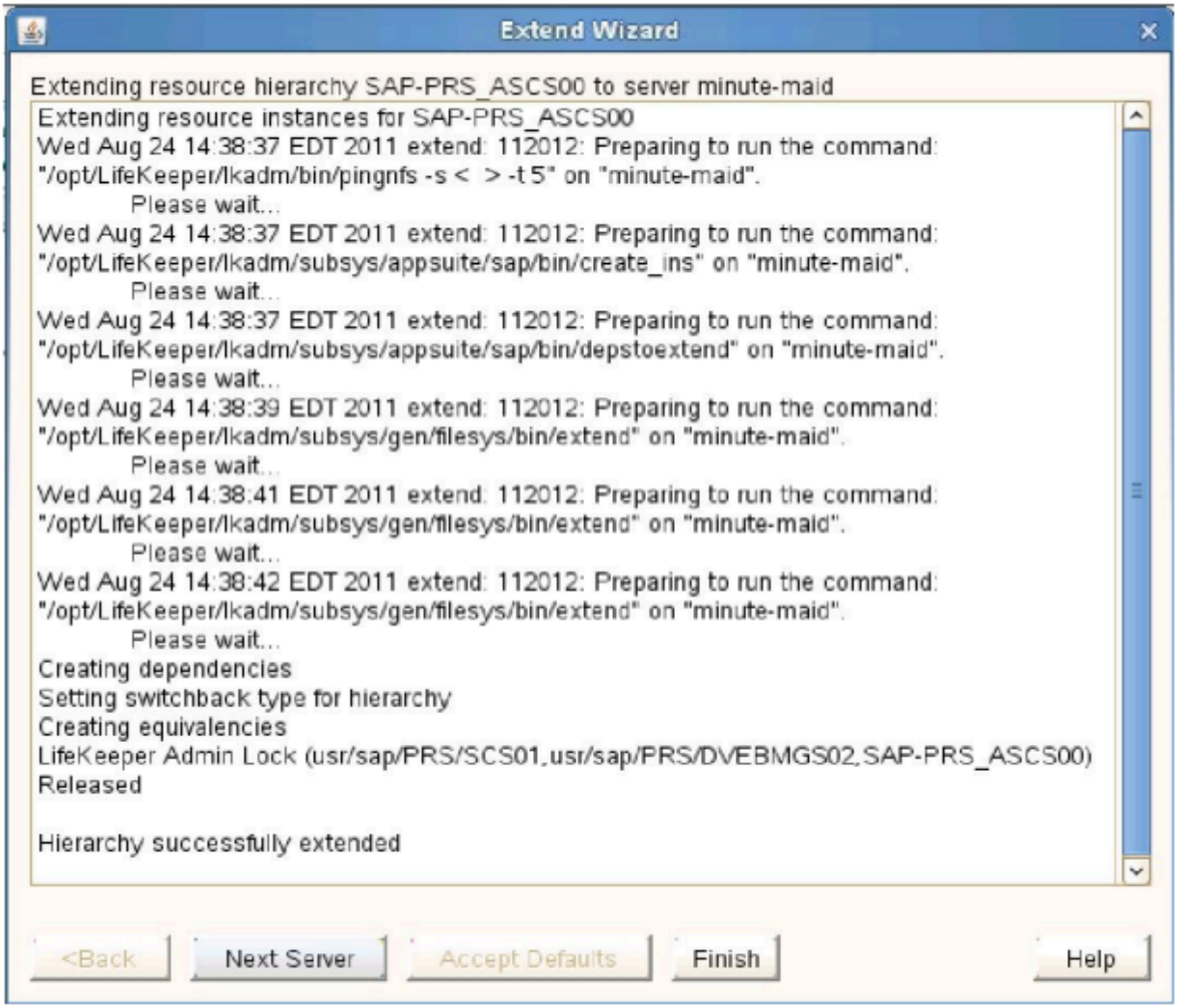
   When you click **Next**, LifeKeeper will launch the ***Pre-Extend Wizard*** that is explained later in this section



If you click **Cancel** now, a dialog box will appear warning you that you will need to come back and extend your SAP resource hierarchy to another server at some other time to put it under LifeKeeper protection.



10. The ***Extend Wizard*** dialog will appear stating **Hierarchy successfully extended**. Click **Finish**.



The **Hierarchy Integrity Verification** dialog appears. Once Hierarchy Verification finishes, click **Done** to exit the **Create Resource Hierarchy** menu selection.

Hierarchy with the Core as the Top Level

11.2.3.6. Create the ERS Resources

The ERS resource provides additional protection against a single point of failure of a Core Instance (Central Services Instance) or enqueue server process. When a Core Instance (Central Services Instance) fails and is restarted, it will retrieve the current status of the lock table and transactions. The result is that, in the event of the enqueue server failure, no transactions or updates are lost and the service for the SAP system continues.

Perform the following steps to create this ERS Resource.

1. For this same SAP SID, repeat the above steps to create the ERS Resource selecting your **ERS instance** when prompted.

2. You will then be prompted to select **Dependent Instances**. Select the **Core Resource** that was created above, and then click **Next**.

3. Follow prompts to **extend resource hierarchy**.

4. Once **Hierarchy Successfully Extended** displays, select **Finish**.

5. Select **Done**

   Note: The Enqueue Replication Server (ERS) resource will be in-service (ISP) on the primary node in your cluster. However, the architecture and function of the ERS requires that the actual processes for the instance run on the backup node. This allows the standby server to hold a complete copy of the lock table information for the primary server and primary enqueue server instance. When the primary server running the enqueuer server fails, it will be restarted by SIOS Protection Suite on the backup server on which the ERS process is currently running. The lock table (replication table) stored on the ERS is transferred to the enqueue server process being recovered and the new lock table is created from it. Once this process is complete, the active replication server is then deactivated (it closes the connection to the enqueue server and deletes the replication table). SIOS Protection Suite will then restart the ERS processes on the new current backup node (formerly the primary) which has been inactive until now. Once the ERS process becomes active, it connects to the enqueue server and creates a replication table. For more information on the ERS process and SAP architecture features, visit http://help.sap.com and search for **Enqueue Replication Service**.

Hierarchy with ERS as Top Level

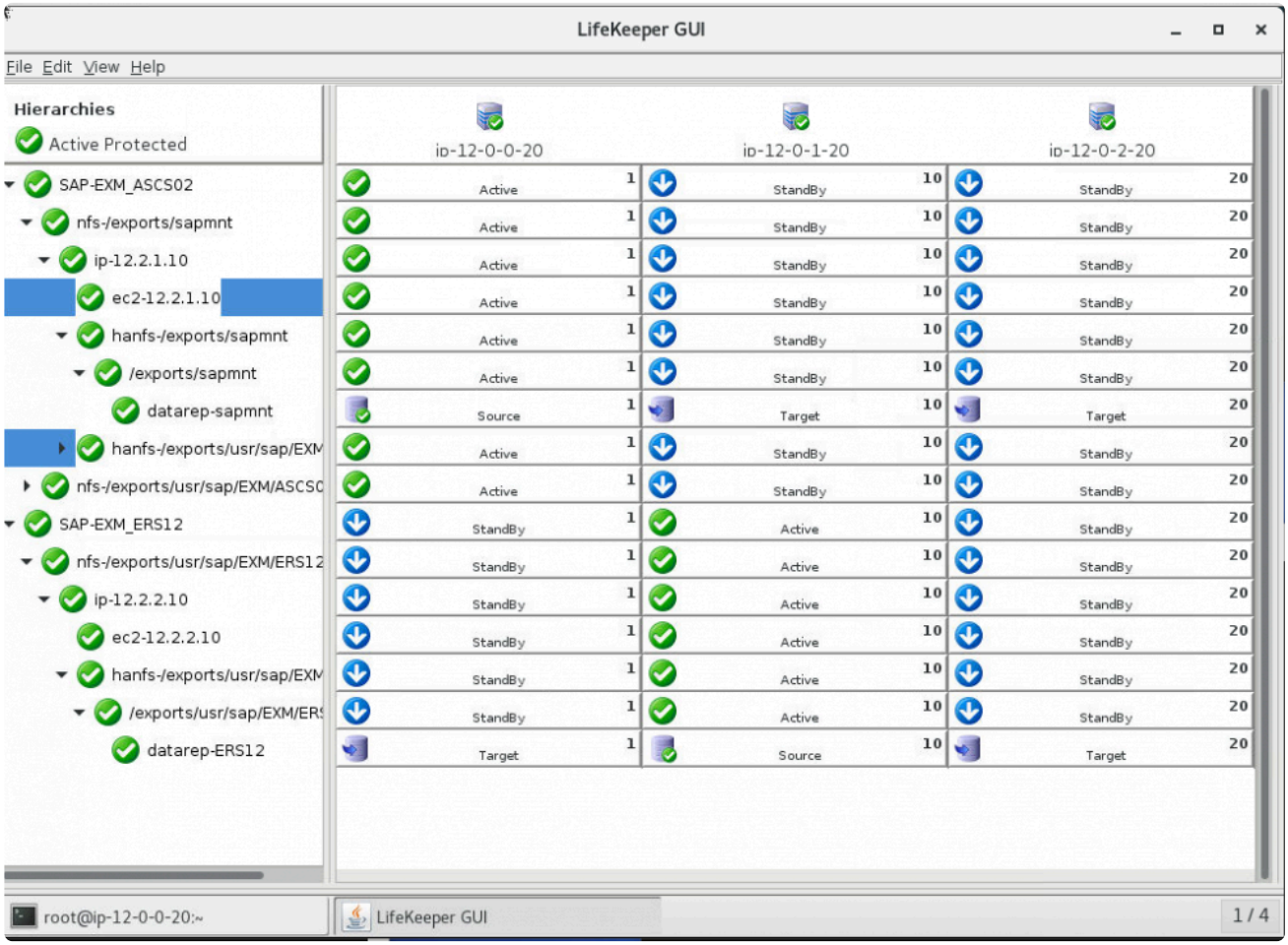
While SIOS Protection Suite can be used to protect the PAS and AAS servers, most customers would simply use them as independent standby servers with no additional HA on them. This guide does not cover their protection steps but you can refer to our SAP Recovery Kit documentation for details and steps.

11.2.3.7. Enforcing ASCS/ERS Avoidance Behavior When Using ENSA2/ERSv2

ERSv2 is intended to be active (in-service) on a node in the cluster where the ASCS resource is not active. The ERS quickCheck will automatically transfer the ERS hierarchy if ERS and ASCS are active on the same node and another node is available. To avoid getting into the situation where ASCS and ERS are both active (in-service) on the same node after a switchover or failover, a gen/app terminal leaf resource can be created to automatically route the in-service to a node where the corresponding resource hierarchy is not active (in-service). To facilitate creating this terminal leaf node a new utility is provided, /opt/LifeKeeper/bin/create_terminal_leaf (1M).

To create the avoidance terminal leaf the utility takes two parameters, the ASCS and ERS hierarchy root resource tags. The two hierarchies should be fully extended to all nodes in the cluster and in-service on a node in the cluster. It does not require that the hierarchies be in-service on the node where the utility is run as long as the utility is run on a node in the cluster. The terminal leaf node will be named “avoid_<tag>” where tag is the appropriate root node to be avoided. The terminal leaf node will be attached as a child dependency on each branch of the hierarchy.

For example, in a configuration with SAP-EXM_ASCS02 and SAP-EXM_ERS12 as the root nodes:

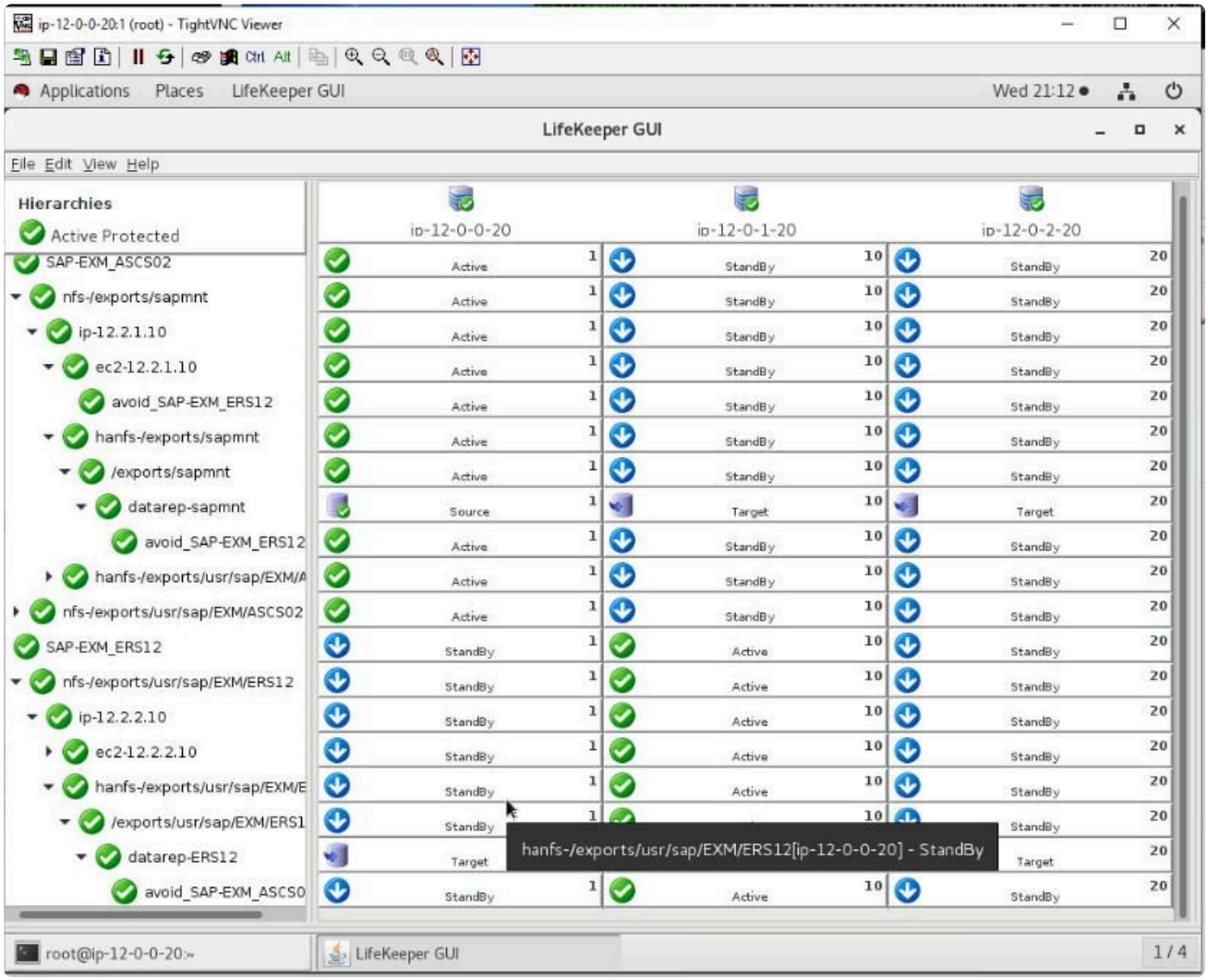


The appropriate terminal leaf nodes are created by running /opt/LifeKeeper/bin/create_terminal_leaf

SAP-EXM_ASCS02 SAP-EXM_ERS12

```
[root@ip-12-0-2-20 ~]# /opt/LifeKeeper/bin/create_terminal_leaf SAP-EXM_ASCS02 SAP-EXM_ERS12
Create avoidance terminal leaf resource for root hierarchy 'SAP-EXM_ASCS02' and 'SAP-EXM_ERS12'.
creapphier: WARNING No quickCheck script specified
creapphier: WARNING No local recovery script specified
BEGIN create of "avoid_SAP-EXM_ERS12"
creating resource "avoid_SAP-EXM_ERS12"
resource "avoid_SAP-EXM_ERS12" successfully created
restoring resource "avoid_SAP-EXM_ERS12"
BEGIN restore of "avoid_SAP-EXM_ERS12"
Attempting to avoid resource avoid_SAP-EXM_ASCS02. Since the resource was not found on ip-12-0-0-20, res
tore of avoid_SAP-EXM_ERS12 is successful on ip-12-0-0-20.
END successful restore of "avoid_SAP-EXM_ERS12"
resource "avoid_SAP-EXM_ERS12" restored
END successful create of "avoid_SAP-EXM_ERS12"
Extending resource instances for avoid_SAP-EXM_ERS12
Creating dependencies
Setting switchback type for hierarchy
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ERS12) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ERS12"
END successful extend of "avoid_SAP-EXM_ERS12"
Extending resource instances for avoid_SAP-EXM_ERS12
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ERS12) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ERS12"
END successful extend of "avoid_SAP-EXM_ERS12"
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCS02'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCS02'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCS02'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
creapphier: WARNING No quickCheck script specified
creapphier: WARNING No local recovery script specified
BEGIN create of "avoid_SAP-EXM_ASCS02"
creating resource "avoid_SAP-EXM_ASCS02"
resource "avoid_SAP-EXM_ASCS02" successfully created
restoring resource "avoid_SAP-EXM_ASCS02"
BEGIN restore of "avoid_SAP-EXM_ASCS02"
Attempting to avoid resource avoid_SAP-EXM_ERS12. Since the resource is not ISP on ip-12-0-1-20, restore
 of avoid_SAP-EXM_ASCS02 is successful on ip-12-0-1-20.END successful restore of "avoid_SAP-EXM_ASCS02"
resource "avoid_SAP-EXM_ASCS02" restored
END successful create of "avoid_SAP-EXM_ASCS02"
Extending resource instances for avoid_SAP-EXM_ASCS02
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ASCS02) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ASCS02"
END successful extend of "avoid_SAP-EXM_ASCS02"
Extending resource instances for avoid_SAP-EXM_ASCS02
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ASCS02) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ASCS02"
END successful extend of "avoid_SAP-EXM_ASCS02"
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
[root@ip-12-0-2-20 ~]#
```

The terminal leaf node has now been attached:



The avoid_SAP-EXM_ERS12 resource will not allow the SAP-EXM_ASCS02 hierarchy to come in-service on a node if the avoid_SAP-EXM_ASCS02 resource (in the SAP-EXM_ERS12 hierarchy) is in-service on that node and there is another viable node available in the cluster. A node is **NOT** a viable option when:

1. The node is not responding.
2. LifeKeeper is not running on the node.
3. A local recovery has failed on the node. This is determined by checking the output of /opt/LifeKeeper/bin/flg_list for the flag '!volatile!recover_fail_<tag>'.

The avoidance leaf can be disabled on a particular system by creating the flag:

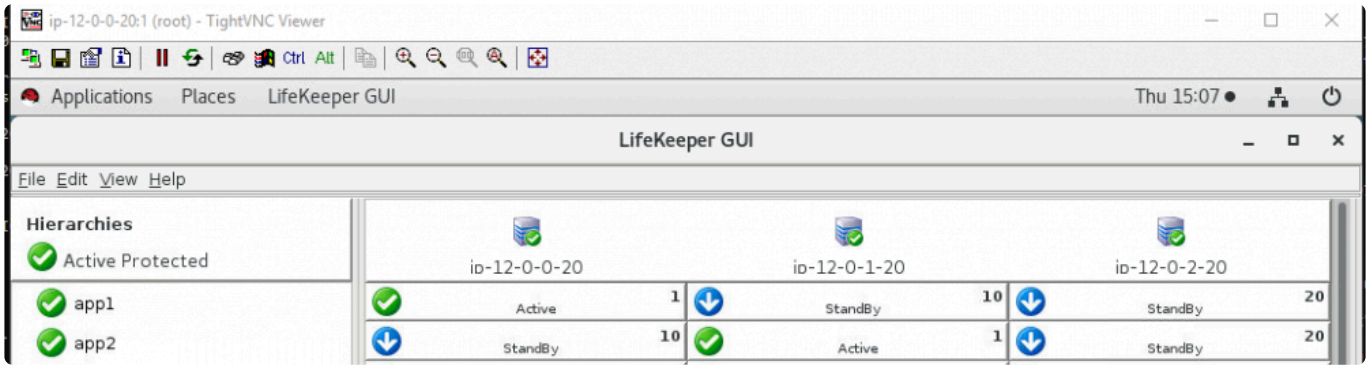
1. "ignore_avoidance_leaf" – will disable the avoidance leaf checking for any resource, aka the avoidance leaf will come in-service at all times.
2. "ignore_<tag>" – will disable the particular so that it will always come in-service but other avoidance leafs will still avoid in-service.

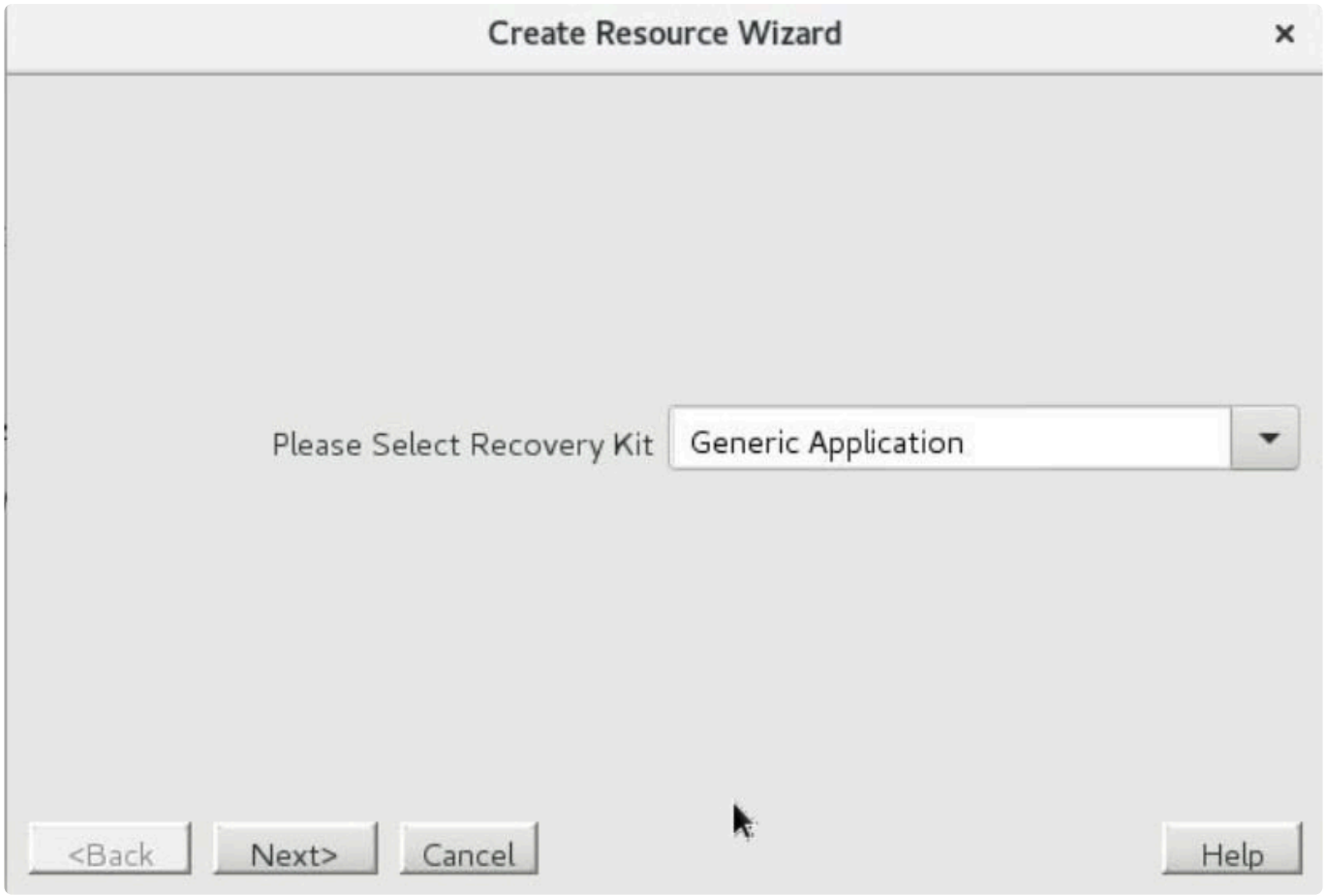
NOTE: The flags will not affect the SAP quickCheck from migrating ERS when it detects it is running on ASCS. The flag "sap_no_ers_relocation_<tag>" will disable quickCheck from relocating ERS where <tag> is the ERS resource tag.

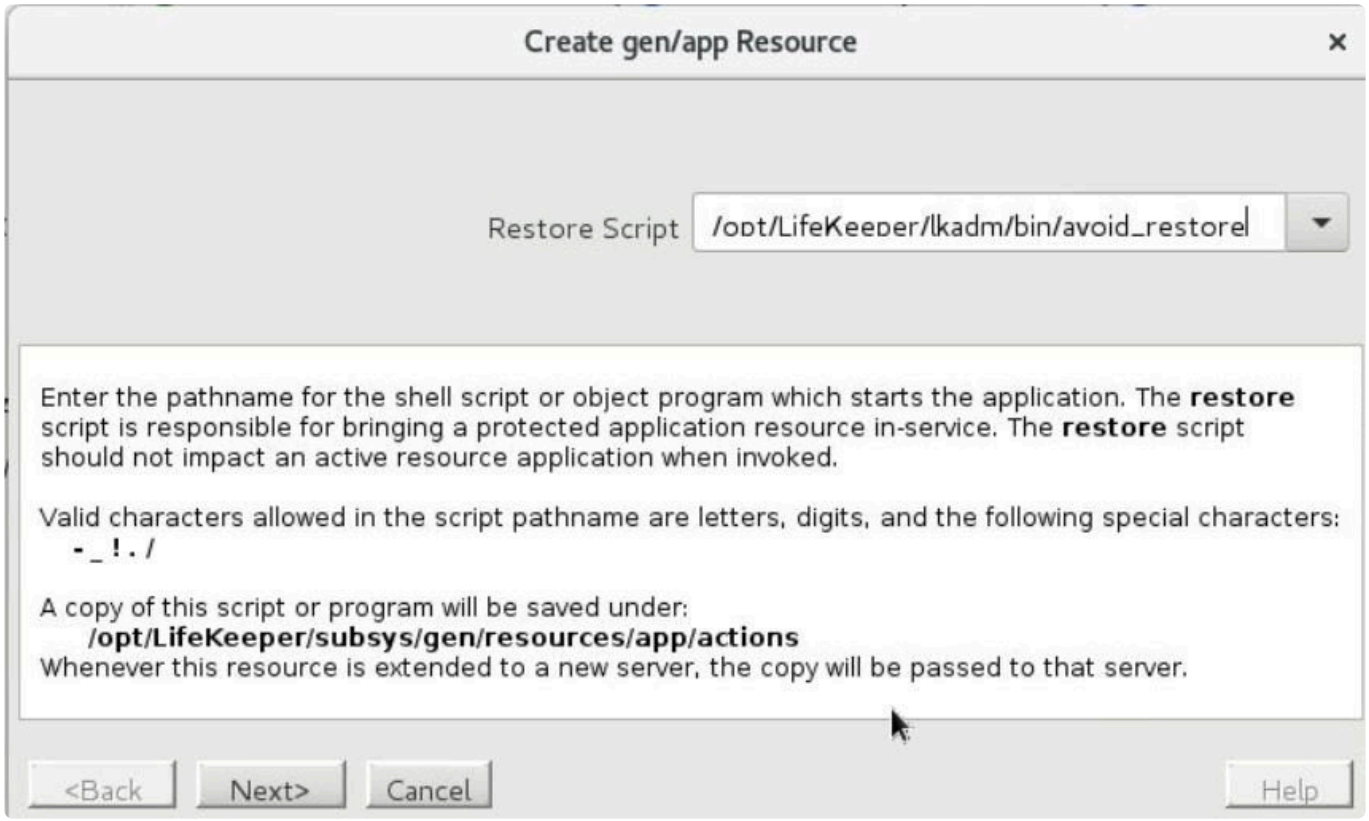
Creating Avoidance Terminal Leaf Node Using the GUI

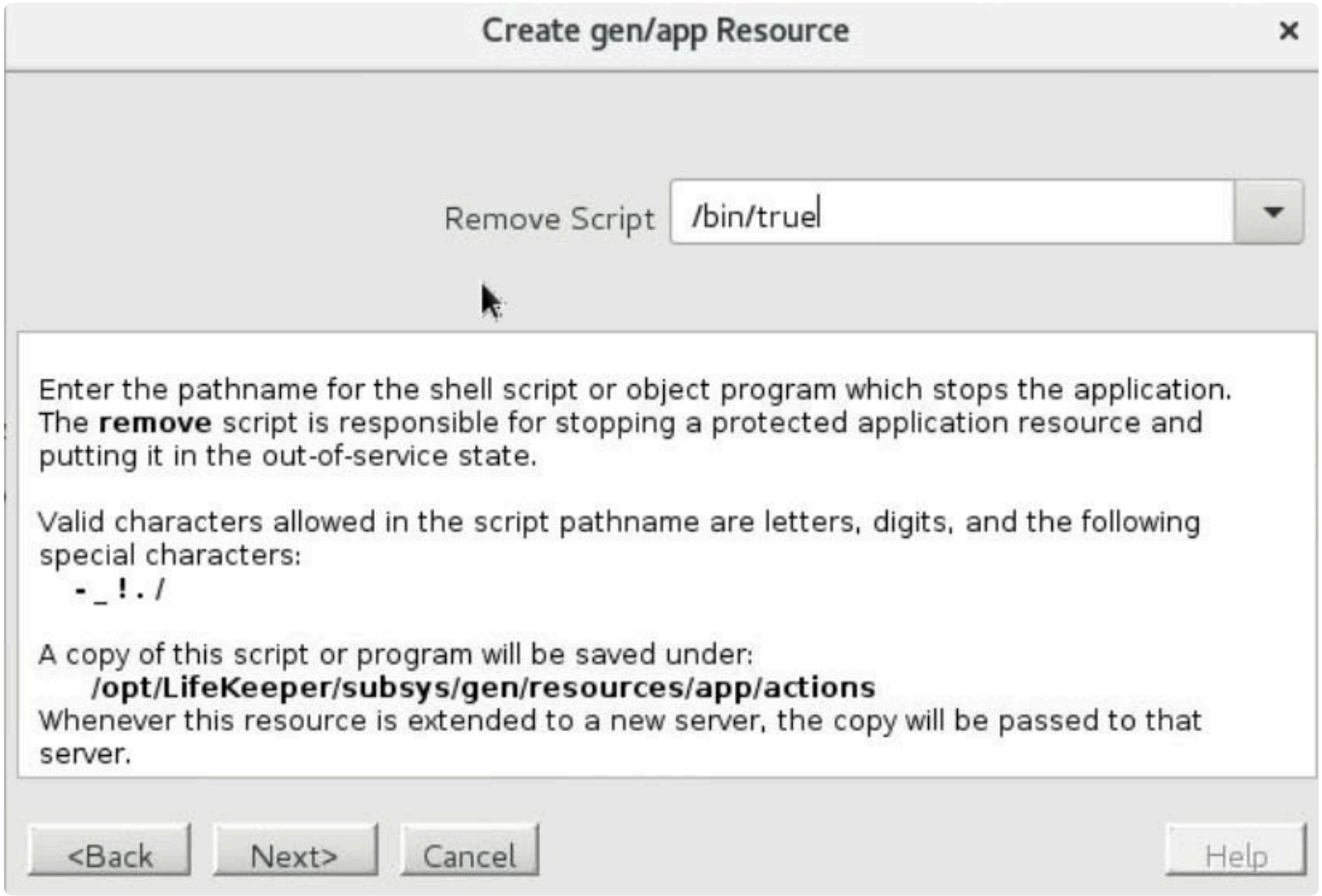
The avoidance terminal leaf node is a gen/app resource that can be created using the GUI. The restore script for the avoidance terminal leaf is '/opt/LifeKeeper/lkadm/bin/avoid_restore'. The remove script should be '/bin/true'. There is no quickCheck script. The info field should be the name of the tag to avoid.

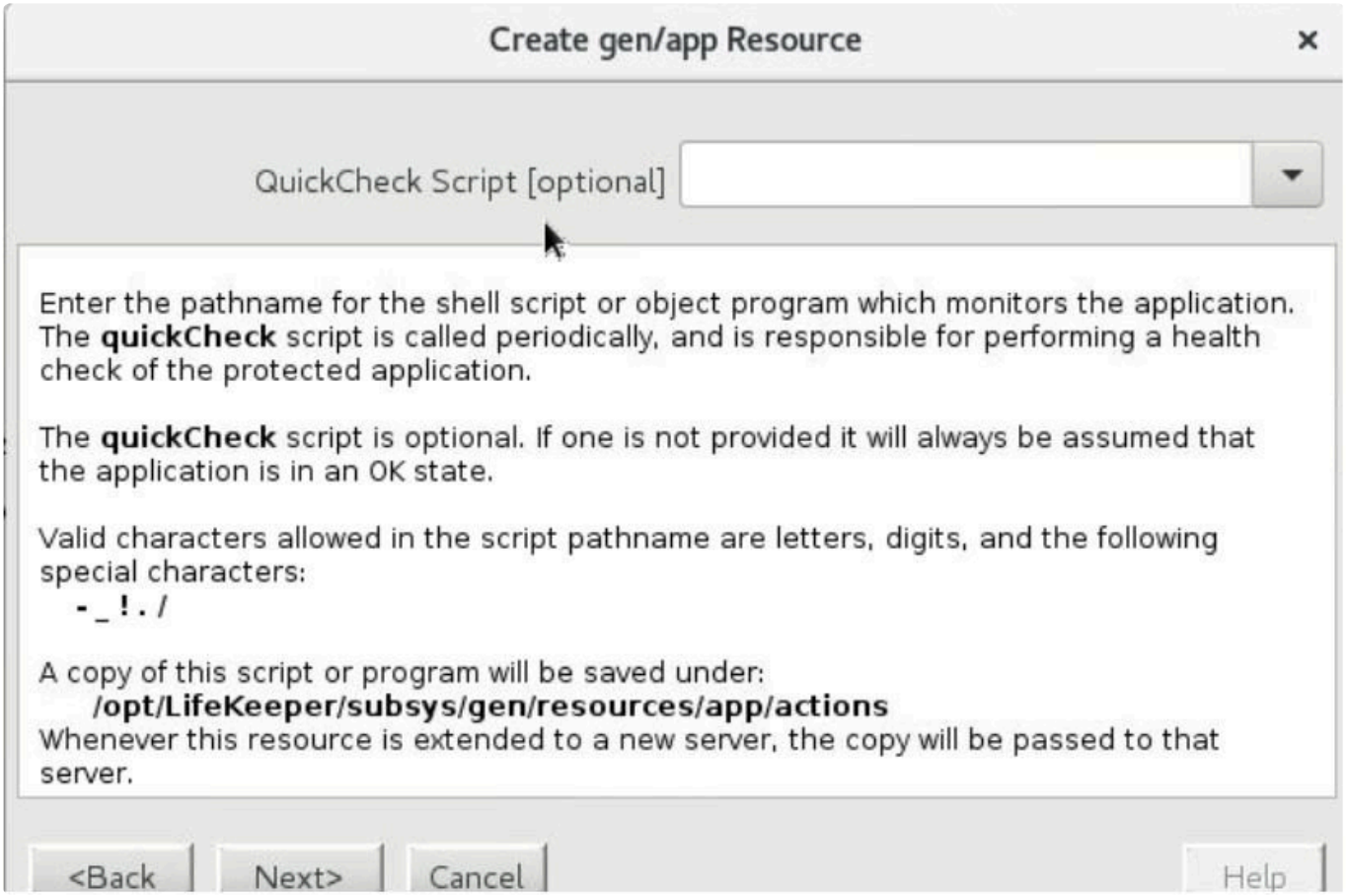
For example, there are two resources, app1 and app2, that you want to be on different nodes when possible. You can create two gen/app resources, "avoid_app1" and "avoid_app2". The 'info' field for avoid_app1 would have 'avoid_app2'. The 'info' field for avoid_app2 would have 'avoid_app1'. The 'avoid_app2' is a dependent child resource to 'app1' and 'avoid_app1' is a child resource to 'app2'.
Note: The tag name is not required to be 'avoid_<tag>' but this makes it clear what the resource is doing.

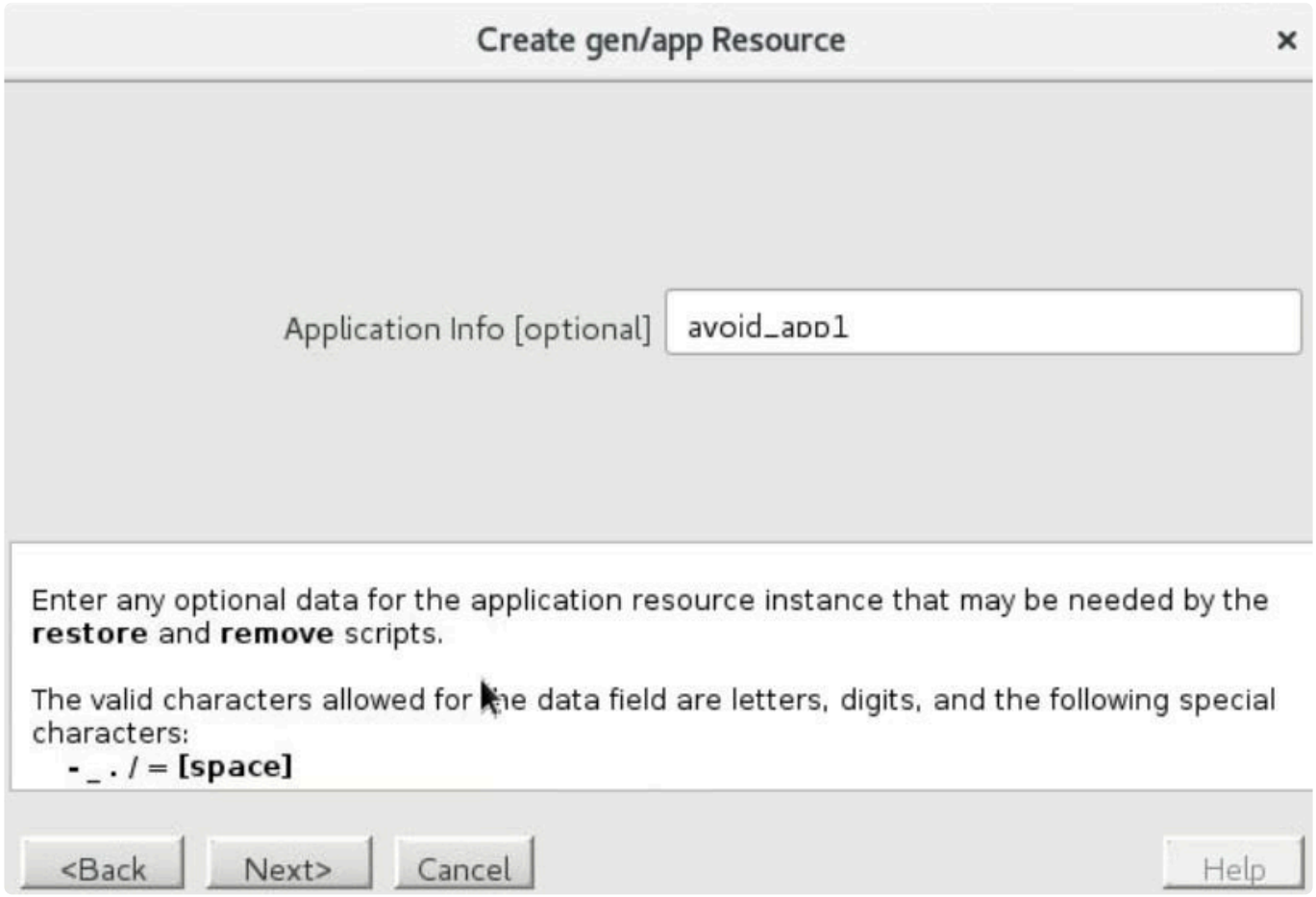





Create gen/app Resource
Restore Script
/opt/LifeKeeper/lkadm/bin/avoid_restore
Enter the pathname for the shell script or object program which starts the application. The restore script is responsible for bringing a protected application resource in-service. The restore script should not impact an active resource application when invoked.
Valid characters allowed in the script pathname are letters, digits, and the following special characters:
- _ ! . /
A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
Whenever this resource is extended to a new server, the copy will be passed to that server.
<Back
Next>
Cancel
Help

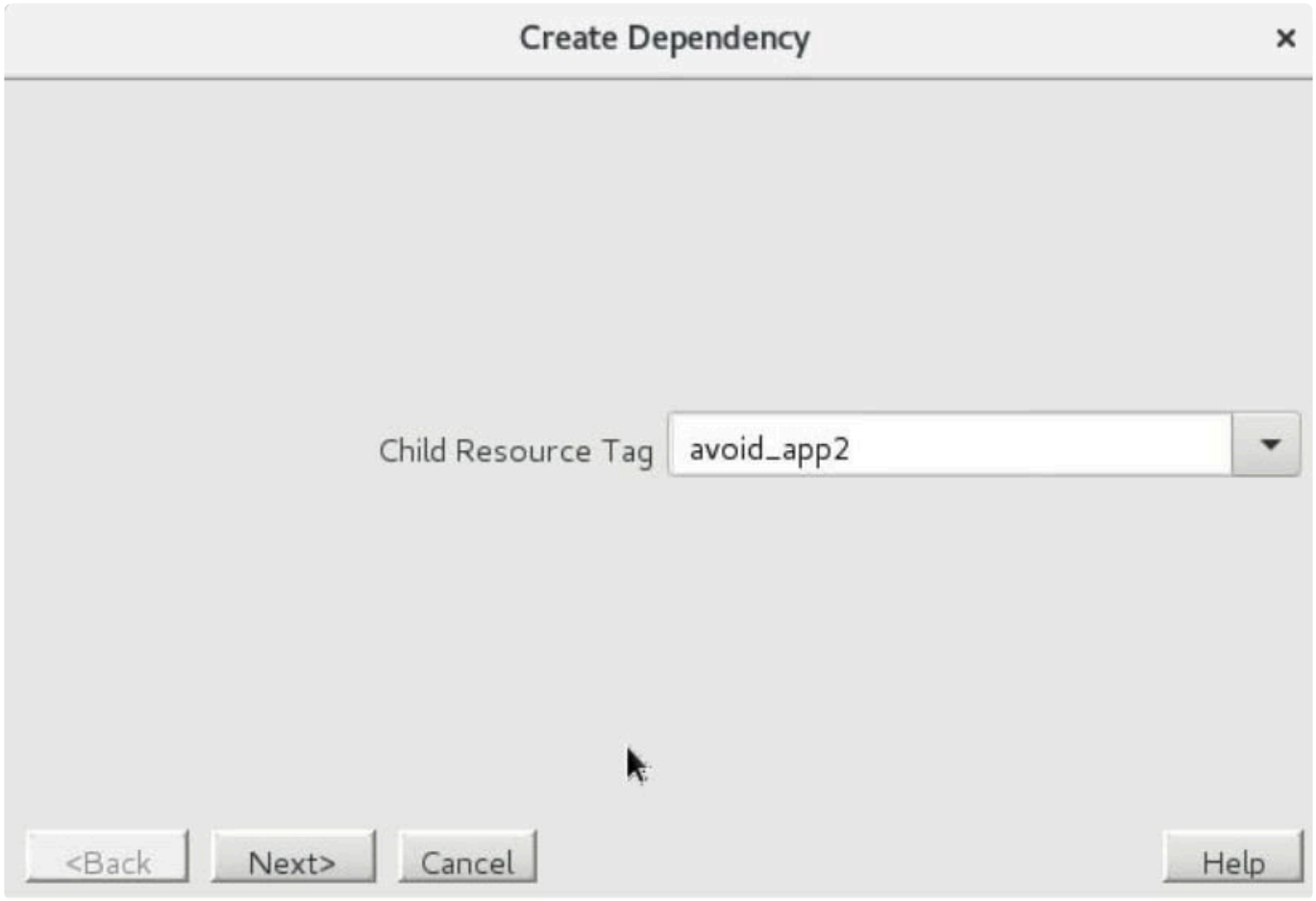

Create gen/app Resource
Remove Script
/bin/truel
Enter the pathname for the shell script or object program which stops the application. The remove script is responsible for stopping a protected application resource and putting it in the out-of-service state.
Valid characters allowed in the script pathname are letters, digits, and the following special characters:
- _ ! . /
A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
Whenever this resource is extended to a new server, the copy will be passed to that server.
<Back
Next>
Cancel
Help


Create gen/app Resource
QuickCheck Script [optional]
Enter the pathname for the shell script or object program which monitors the application. The quickCheck script is called periodically, and is responsible for performing a health check of the protected application.
The quickCheck script is optional. If one is not provided it will always be assumed that the application is in an OK state.
Valid characters allowed in the script pathname are letters, digits, and the following special characters:
- _ ! . /
A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
Whenever this resource is extended to a new server, the copy will be passed to that server.
<Back
Next>
Cancel
Help

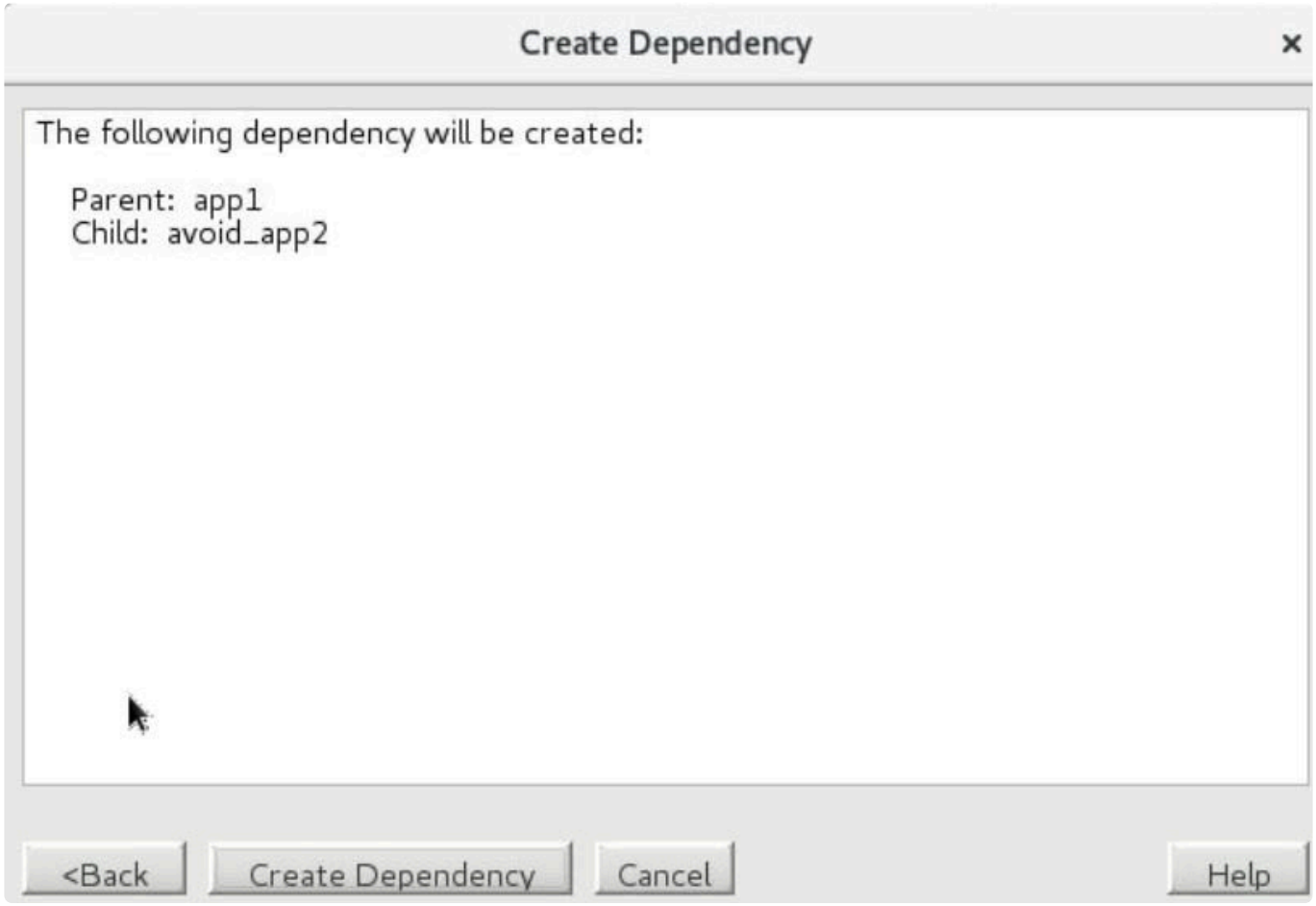

Create gen/app Resource
Application Info [optional]
avoid_app1
Enter any optional data for the application resource instance that may be needed by the restore and remove scripts.
The valid characters allowed for the data field are letters, digits, and the following special characters:
- _ . / = [space]
<Back
Next>
Cancel
Help

After creating ‘avoid_app2’ extend it to all nodes with the same priorities that app1 has.

Then select the ‘app1’ resource and create a child dependency with ‘avoid_app2’.



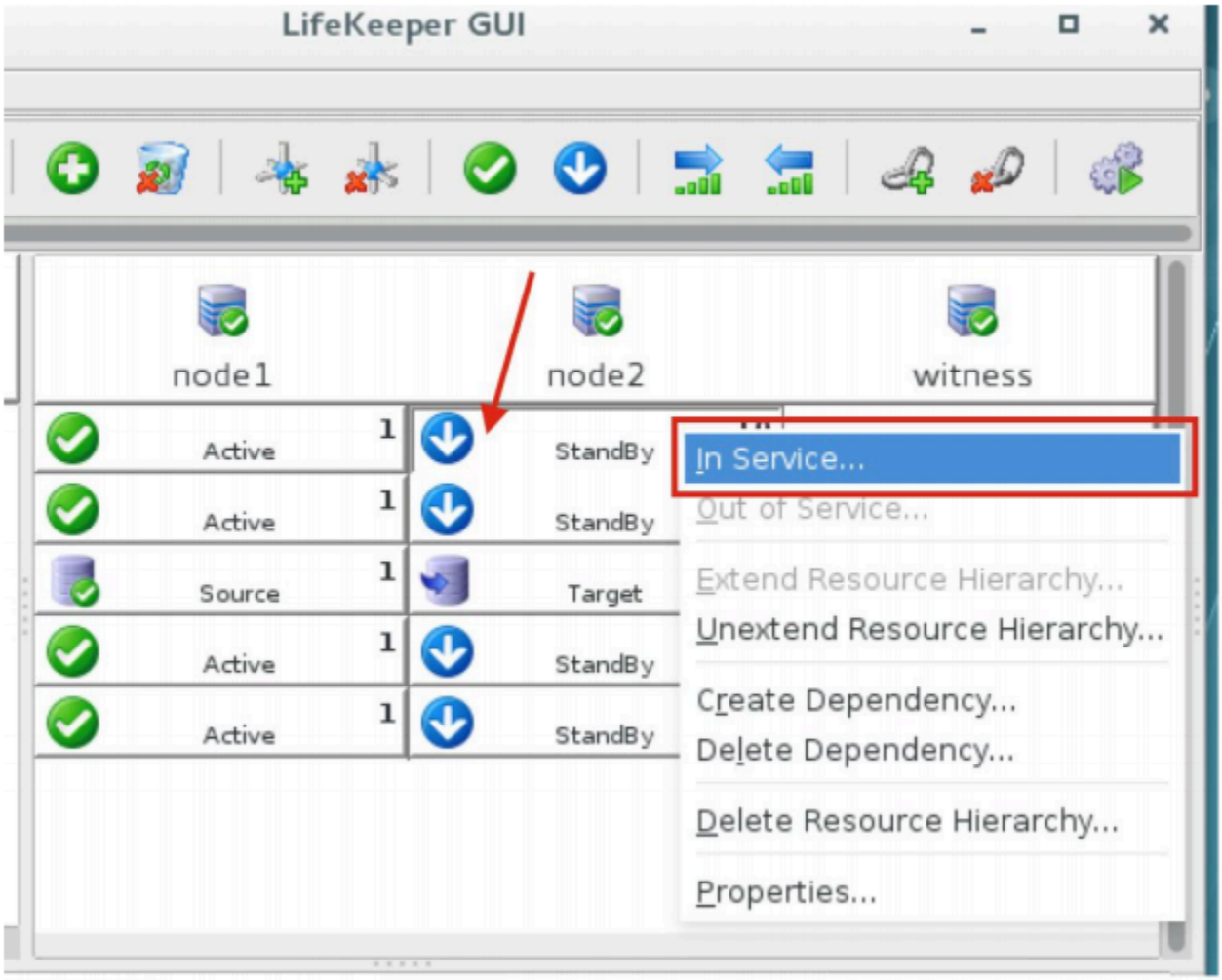
After creating the avoid_app1 resource similarly the hierarchy will look like:


Create Dependency
The following dependency will be created:
Parent: app1
Child: avoid_app2
<Back
Create Dependency
Cancel
Help

11.2.4. Switchover and Failover Testing

Steps below are for testing the switchover and failover of a SIOS cluster for SAP. Open “SAP Logon” or “SAP GUI for Windows”, which is an SAP supplied Windows client the Windows client. The program can be downloaded from the SAP download site. The virtual IP address may be used as the “Application Server” on the **Properties** page. This ensures that a connection to the primary machine where the virtual ip resides is active.

1. Using the LifeKeeper GUI, failover from Node1 -> Node2. Right click on the top resource in the cluster underneath node2, and select “In Service…”. This demonstrates that node 2 is able to take over from node1 during a failure



After switchover has completed, check on the SAP GUI or reconnect where necessary and examine that SAP is still running normally.

You may also check if SAP processes are running in the OS.

2. Using the LifeKeeper GUI, failover from Node2 -> Node1. Right click on the top resource in the cluster underneath node2, and select “In Service…”. This demonstrates that node 1 is able to take over from node2 during a failure

After switchover has completed, check on the SAP GUI or reconnect where necessary and examine that SAP is still running normally.

You may also check if SAP processes are running in the OS.

3. On the command line interface on node 1 (active node), execute the following command to perform a “hard crash” of the OS

   # halt -fni

   After failover has completed, use the LifeKeeper GUI on node 2 to check visually that the services are failed over normally.

   Check on the SAP GUI or reconnect where necessary and examine that SAP is still running normally.

   You may also check if SAP processes are running in the OS.

   Turn on node 1 again and use the LifeKeeper GUI on node 2 to check visually that the services are on node 1 becomes standby, and replication is started.

   Note: Before attempting to do any more switchover or failover testing, ensure that the data replication resources have already completed their synchronization and are in sync.

4. Repeat step 2 or step 1 as necessary to switchover back to node 1, or to perform another crash testing on node 2.

11.2.4.1. Additional Resources

AWS services

- Amazon EC2
  https://aws.amazon.com/documentation/ec2/

- AWS CloudFormation
  https://aws.amazon.com/documentation/cloudformation/

- Amazon VPC
  https://aws.amazon.com/documentation/vpc/

SIOS Protection Suite for Linux

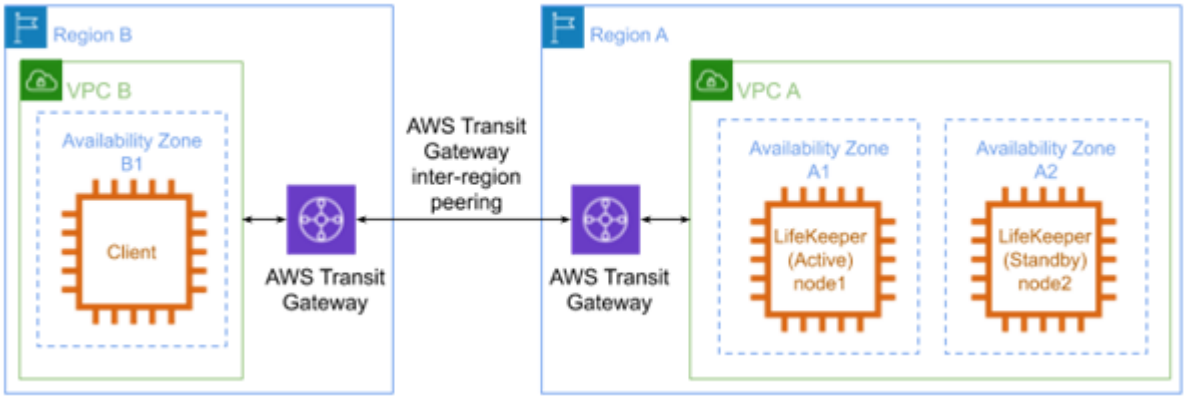
- Step-By-Step: How to configure a Linux failover cluster in Amazon EC2 without shared storage
  http://www.linuxclustering.net/2016/03/21/step-by-step-how-to-configure-a-linux-failover-cluster-in-amazon-ec2-without-shared-storage-amazon-aws-sanless-cluster/

Quick Start reference deployments

- AWS Quick Start home page
  https://aws.amazon.com/quickstart/

11.3. Connecting to a LifeKeeper Cluster using AWS Transit Gateway Quick Start Guide

With the release of AWS Transit Gateway and AWS Transit Gateway inter-region peering, the Recovery Kit for EC2 route table scenario is now available for configurations where a client in a VPC (VPC B in the figure below) connects to an HA cluster located in a different region and VPC (VPC A in the figure below).



This document describes the requirements and basic operations for building a configuration where a client connects to a LifeKeeper for Linux HA cluster in another region.

This document does not explain the basic settings, operations or technical details of LifeKeeper or Amazon Web Service (AWS). For terms, operations and technical information related to LifeKeeper and AWS required for this configuration, review the related documents and user websites.

Note: AWS Transit Gateway inter-region peering is available only in the Eastern U.S. (N. Virginia, Ohio), Western U.S. (Oregon) and Europe (Ireland, Frankfurt) as of February 2020. If you deploy the server or client in another region, the configuration described in this document cannot be used. If you place your server or client in a region where AWS Transit Gateway inter-region peering is not available, consider using the Route53 Recovery Kit to update DNS A records (corresponding IP address to host names) registered in “Route53” of the AWS DNS service.

Note: This document is for configurations where cluster nodes are located within a single VPC. Route table scenarios cannot be used with configurations where cluster nodes are located across multiple regions or multiple VPCs.

Note: Amazon Web Services, Powered by Amazon Web Services logo, AWS, Amazon EC2, EC2, Amazon Elastic Compute Cloud, Amazon Virtual Private Cloud, AWS Direct Connect, AWS Identity and Access Management, AWS Transit Gateway, AWS Transit

11.3.1. AWS VPC Peering Connections Requirements

The following requirements should be met when using this configuration. Below is a summary of requirements for the AWS environment and instances created on it.

Requirements for AWS environment

Create a base environment on AWS to provide services. The requirements for using this configuration are as follows.

Amazon Virtual Private Cloud (VPC)

- A VPC needs to be configured in AWS.
- The VPC where the client is located must be configured in a different region from the VPC where the cluster nodes are located.
- Create a subnet for the primary instance and a subnet for the standby instance in the VPC where the cluster nodes reside. The subnets must be created in different Availability Zones (AZ).
- The security groups for the subnets in the VPC containing the cluster nodes must be configured to allow incoming traffic from the subnet in the VPC containing the client, and vice-versa.

Amazon Elastic Compute Cloud (EC2)

- At least 2 instances are required.
- A primary instance and a standby instance need to be configured in different AZs from each other.
- Cluster node instances are connected to an Elastic Network Interface (ENI).
- Cluster node instances must satisfy LifeKeeper installation requirements.
- The AWS Command Line Interface (AWS CLI) must be installed on all of the cluster node instances. Refer to Installing the AWS CLI for more details. The path to the AWS CLI executable files must be appended to the PATH parameter in the LifeKeeper defaults file `/etc/default/LifeKeeper` if it is not already present there.
- The cluster nodes need to be able to access the Amazon EC2 web service endpoint URL (see https://docs.aws.amazon.com/general/latest/gr/rande.html#ec2_region) using https and the Amazon EC2 metadata URL (http://169.254.169.254/) using http.

AWS Identity and Access Management (IAM)

In order for LifeKeeper to operate in AWS, an IAM user or IAM role with the following access privilege is

required. Please configure EC2 IAM role or configure AWS CLI appropriately so that it can be accessed by the root user of the EC2 instance.

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress
- ec2:DescribeRouteTables
- ec2:ReplaceRoute

AWS Transit Gateway

- The VPC with cluster nodes and VPCs with clients should not be directly connected to each other with AWS Inter-Region VPC Peering. Instead, create an AWS Transit Gateway in each region and connect the AWS Transit Gateways with AWS Transit Gateway inter-region peering.

- Enable the default route table association and the default route table propagation when creating each AWS Transit Gateway.

- Create a Transit Gateway Attachment in each region to connect each AWS Transit Gateway to its corresponding VPC.

- An AWS Transit Gateway inter-region peering connection between AWS Transit Gateways should be enabled by creating a Transit Gateway Attachment. Note that this step requires manual confirmation in the target region before the Transit Gateway Attachment will actually be created by AWS.

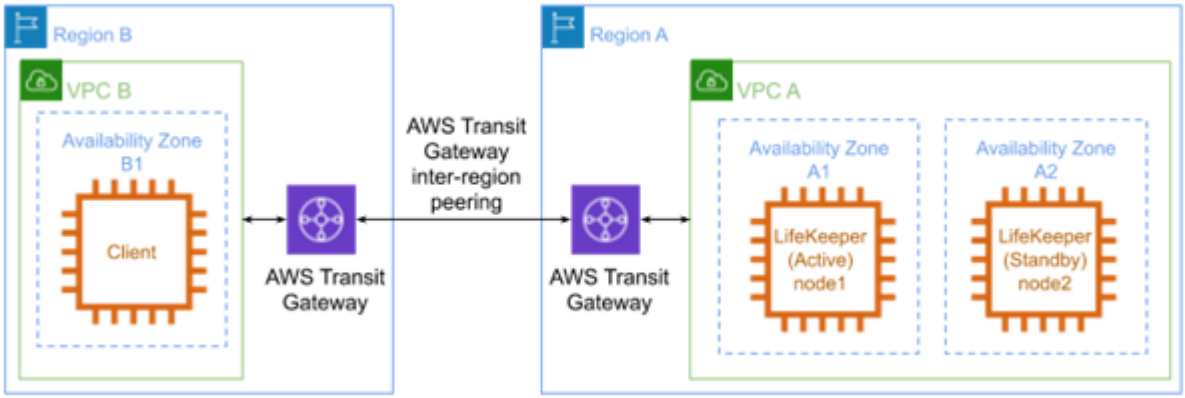
11.3.1.1. LifeKeeper Software Requirements for AWS Environment

Install the same version of LifeKeeper software and patches on each server in the high-availability cluster. The Application Recovery Kits (ARK) required for this configuration are shown below. For the specific LifeKeeper requirements, please refer to: SPS for Linux Technical Documentation and SPS for Linux Release Notes.

- LifeKeeper IP Recovery Kit
- LifeKeeper EC2 Recovery Kit

11.3.2. AWS VPC Peering Setup Procedure

This section describes the general procedure to set up the environment shown in the figure below.



Preparations

- Create an environment that satisfies the AWS VPC Peering Connections Requirements.
- Install LifeKeeper on each instance and create a communication path between node 1 and node 2.

Creating an IP Resource

- Create a Virtual IP resource on node 1 and extend it to node 2. The IP resource address must be outside of the CIDR block managed by VPC A.

Creating an EC2 Resource

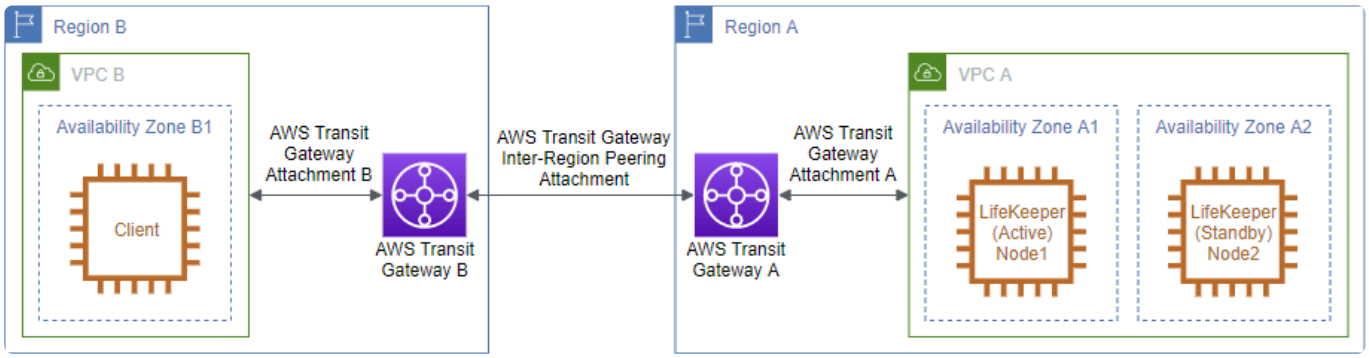
Create an EC2 resource on node 1 and extend it to node 2. For the IP resource requested when creating a resource, specify the resource created in “Creating an IP Resource”. Specify “Route Table (Backend Cluster)” for the EC2 resource type when creating the resource.

Creating Resources for Protected Services

Create a resource for the service or application you want to protect. If an IP resource is required when creating a resource, specify the resource created in “Creating an IP Resource”. If necessary, configure resource dependencies so that the service/application is the parent resource and the EC2 resource is the child resource.

11.3.3. Configuring the Route Table

The AWS environment should be configured as in the following diagram:



Add the following routes to the route table for VPC B or the subnet that contains the client instance:

Destination Address	Target
VPC A CIDR Block	AWS Transit Gateway B
Virtual IP Address	AWS Transit Gateway B

Add the following routes to the route table for AWS Transit Gateway B:

CIDR	Choose Attachment
VPC A CIDR Block	AWS Transit Gateway Inter-Region Peering Attachment
Virtual IP address	AWS Transit Gateway Inter-Region Peering Attachment

Add the following routes to the route table for AWS Transit Gateway A:

CIDR	Choose Attachment
VPC B CIDR Block	AWS Transit Gateway Inter-Region Peering Attachment
Virtual IP address	AWS Transit Gateway Attachment A

Add the following route to the route table for VPC A or the subnets that contain the LifeKeeper instances:

Destination Address	Target
VPC B CIDR Block	AWS Transit Gateway A

Once configured, make sure that the client can access the private address of each server in the high-availability cluster as well as the virtual IP address.

11.3.4. Considerations for Settings and Operations in AWS VPC Peering

Considering the Use of LifeKeeper I-O Fencing – AWS VPC Peering

11.3.4.1. Considering the Use of LifeKeeper I-O Fencing – AWS VPC Peering

Since an AWS environment does not support shared disk configurations, SCSI reservations cannot be used to prevent split brain scenarios. For this reason, consider using the Quorum/Witness Server or STONITH, LifeKeeper's I/O fencing functionality, to operate more safely with this configuration.

Quorum fencing functionality can be easily configured in cloud environments by using the TCP_REMOTE Quorum mode, instead of setting up a separate Quorum server. For details, please see the URL below

Quorum/Witness

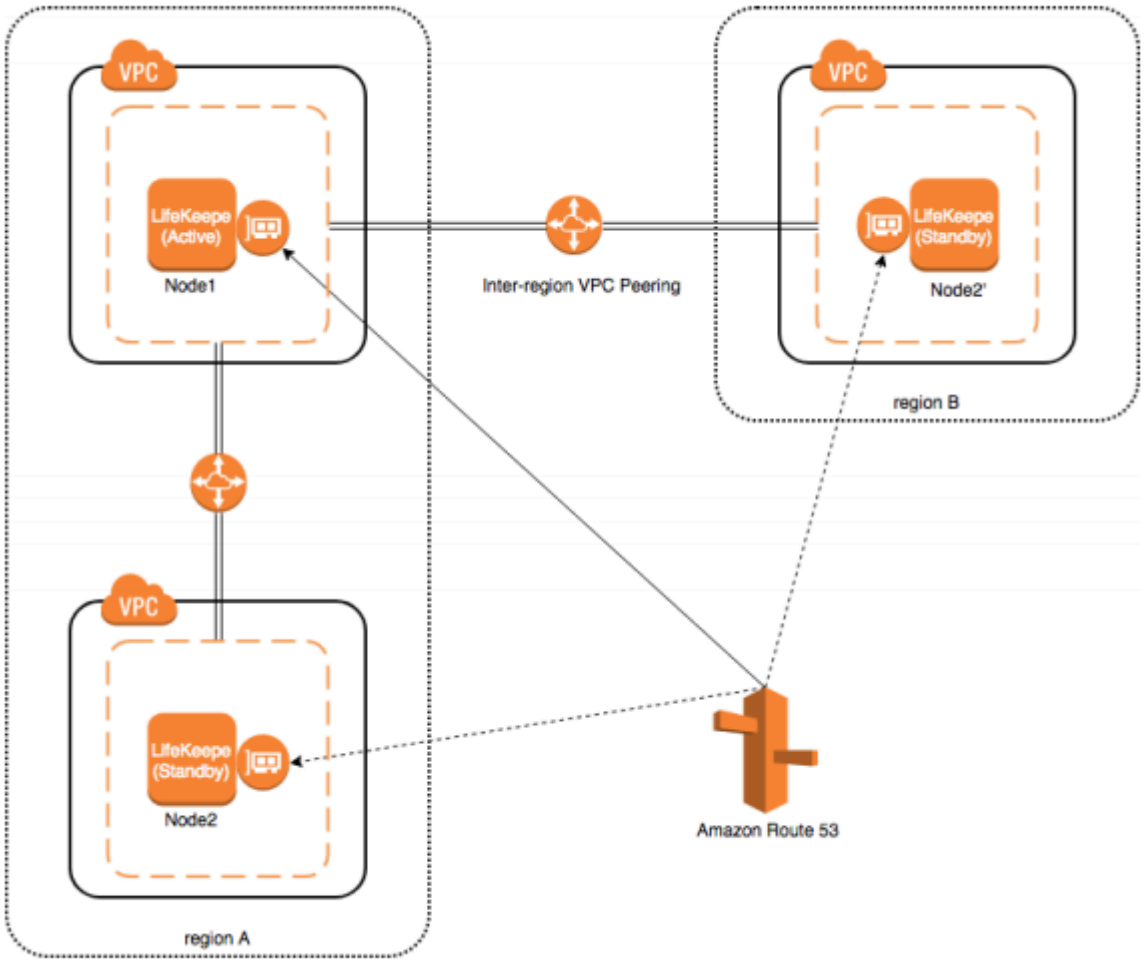
STONITH

11.3.5. AWS Direct Connect Known Issues and Troubleshooting

There are currently no Known Issues.

11.4. Connecting to a LifeKeeper Cluster using AWS VPC Peering Quick Start Guide

Objective



This document descries the requirements and basic operations for building connections among VPCs with LifeKeeper for Linux for v9.5.0.

You can also build HA clusters in the AWS environment using the existing Recovery Kit for EC 2; however, you cannot connect from your on-premises environment with AWS Direct Connect due to the problems described below.

Recovery Kit for EC2 provides two functions: “Route Tables Scenario” and “Elastic IP Scenario.”

“**Route Tables Scenario**” manages VPC route tables are configured to be routed to an active IP resources. An address of IP resource should be outside CIDR block which is managed within the VPC. However, the address should be the one within the VPC CIDR block in order to connect from other VPC via VPC Peering Connection. With this route table scenario, you cannot connect to the VPC from the on-premises environment.

“**Elastic IP Scenario**” can be used where the access from the Internet is available since the elastic IP address is a public address. An access from the on-premises environment is enabled through the Internet. In this case, you can access to HA cluster nodes on VPC without VPC Peering Connection.

For above reasons, Recovery kit for EC2 does not support an access to VPC from other VPC using VPC Peering Connection. If you need to access to HA cluster nodes on the VPC via VPC Peering Connection, please use the configuration provided in this document.

> It is also now possible to use AWS Transit Gateways for inter-region peering. Refer to Connecting to a LifeKeeper Cluster using AWS Transit Gateway Quick Start Guide for information.

Please note that this document does not describe the basic settings, operations, and technical details of LifeKeeper and Amazon Web Service (AWS). For terms, operations and technical information related to LifeKeeper and AWS, that are the prerequisites of this configuration, please read related documents and user websites beforehand.

> **Note:** “Amazon Web Services,” “Powered by Amazon Web Services” logo, “AWS,” “Amazon EC2,” “EC2,” “Amazon Elastic Compute Cloud,” “Amazon Virtual Private Cloud,” “Amazon Route 53” and “Amazon VPC” is trademarks of Amazon.com, Inc. or its affiliates in the United States and other countries.

11.4.1. Connecting to a LifeKeeper Cluster using AWS Requirements

Some requirements should be met when using this configuration. Below is a summary of requirements for the AWS environment and instances created on it.

Requirements for AWS environment

Create a base environment on AWS to provide services. The requirements for using this configuration are as follows.

Amazon Virtual Private Cloud (VPC)

- VPC needs to be configured in AWS.
- Need to create more than two subnets in different Availability Zones (AZ) or in different VPCs.

Amazon Elastic Compute Cloud (EC2)

- At least 2 instances are required.
- A primary instance and a standby instance need to be configured to start with different AZ or different VPC for each.
- Instances are connected to Elastic Network Interface (ENI).
- Instances are required to satisfy LifeKeeper's installation requirements.
- AWS Command Line Interface (AWS CLI) needs to be installed in each of EC2 the instances. For the details, please refer to AWS Command Line Interface Installation.
- Instances need to have an access to route53.amazonaws.com with HTTPS protocol. Please configure EC2 and the OS properly

AWS Identity and Access Management (IAM)

In order for LifeKeeper to operate AWS, IAM user or IAM role with the following access privilege is required. Please configure EC2 IAM role or configure AWS CLI appropriately so that it can be accessed from root user of the EC2 instance.

- route53:GetChange
- route53:ListHostedZones
- route53:ChangeResourceRecordSets

- route53:ListResourceRecordSets

Amazon Route 53

- You need to register your domain name on Amazon Route 53 to use the service. This is required to create a Route53 resource.

11.4.1.1. Peering Requirements for Connecting to a LifeKeeper Cluster using AWS

You need to install the same version of LifeKeeper software and patches on each server. The Application Recovery Kit (ARK) required for this configuration is shown below. For the specific LifeKeeper requirements, please refer to: SPS for Linux Technical Documentation and SPS for Linux Release Notes.

- LifeKeeper IP Recovery Kit
- LifeKeeper Route53 Recovery Kit

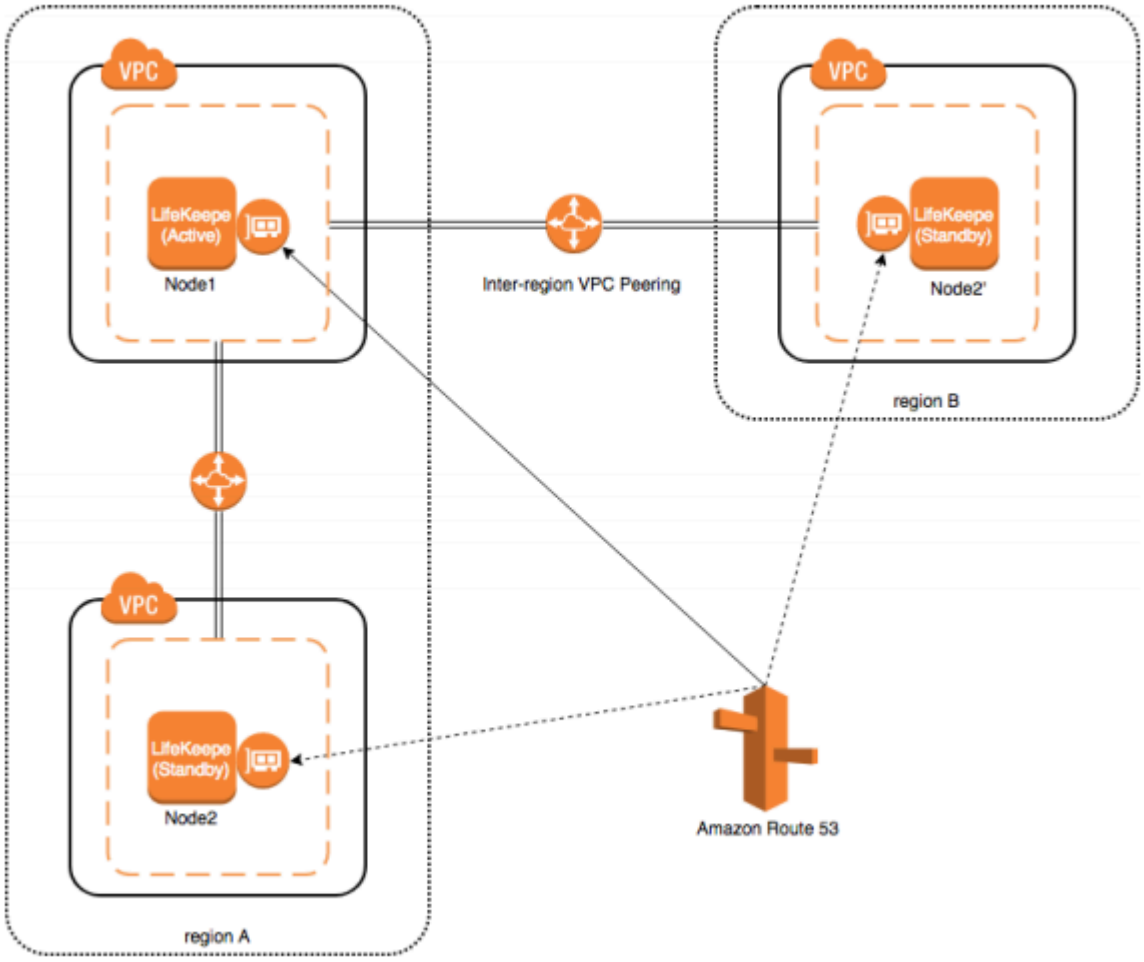
11.4.1.2. Other AWS VPC Requirements

Requirements for using this service from other VPCs are as follows:

- Clients using the service should be able to resolve names of the hosts that are protected by Route53 resources.
- Clients using the service should access with the host name protected by Route53 resource.

11.4.2. Setup Procedure for Connecting to a LifeKeeper Cluster using AWS

In this section, a general procedure to setup the environment shown as the figure below



Preparations

Create an environment that satisfies Requirements. Please install LifeKeeper on each instance and create a communication path between Node1 and Node2 (or Node2'). Please confirm that you can access from other VPC environment to ENI's private address connected to Node1/Node2 (or Node2').

Creating IP Resource

Create an IP resource: not a virtual IP resources but a real IP resource (**Note:** resource for a primary IP address configured for NIC). Please specify ENI private IP address when creating a resource. Also, specify ENI private IP address for an extension target node when extending.

Creating Route53 Resource

Create Route53 resource. Please specify the IP resource created in Creating IP Resources if required

when creating Route53 resource.

Creating Resources for Protected Services

Create resources for protected services. Please specify the IP resource created in Creating IP Resources if required when creating resources. Also, please create a resource dependency to enable the resources of the services protected by the parent resource and the child resource to become Route53 resources.

11.4.3. Related LifeKeeper Resources for AWS VPC Peering

Route53 Resource

Summary

When switchover occurs, it is necessary to update Amazon Route 53 DNS information in order to continue to secure the connection to the service. This feature is provided in Route53 resources. When the status of Route53 resource becomes “In Service,” the IP address of the IP resource with a dependency is registered in the corresponding DNS A record using API.

IP Resource

Summary

IP resource is a resource generated with using IP Recovery Kit included in the LifeKeeper Core product. In order to support this configuration, it is now possible to generate IP resource (real IP resource) with a real IP address. This allows you to use real IP addresses as a LifeKeeper resource.

Please do not use the real IP resource except for this configuration.

For more information, please refer to: IP Recovery Kit Technical Documentation

11.4.4. Connecting to a LifeKeeper Cluster using AWS Settings and Operations Considerations

11.4.4.1. Considering the Use of LifeKeeper I-O Fencing when Connecting to a LifeKeeper Cluster using AWS

Since the shared disk environment cannot be used in AWS environment, you cannot use SCSI reservations to prevent a split-brain. Also, IP resource may cause the split-brain as it uses the real IP resource with different IP addresses for each node.

For this reason, please consider the use of Quorum/Witness server or STONITH, an I/O fencing function of LifeKeeper to use this configuration safely.

Especially, because you can implement I/O fencing function separately without the Quorum server if you use the TCP_REMOTE setting in Quorum mode, it is easy to be implemented in the cloud environment. For more details, please refer to the following URLs:

Quorum/Witness

STONITH

11.5. DataKeeper for Linux Evaluation Guide

Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product with DataKeeper to enable real time, host based, block-level data replication

There are five phases in this process:

- Phase 1 – Prepare to Install
- Phase 2 – Configure Storage
- Phase 3 – Install SIOS Protection Suite for Linux
- Phase 4 – Configure your LifeKeeper Cluster
- Phase 5 – Test Your Environment

11.5.1. DK for Linux Terms to Know

The following terms are used throughout this document and, while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

Network Communication Terms

Crossover cable – A cable used to directly connect computing devices together, instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

Types of LifeKeeper Servers

Server – A computer system dedicated to running software application programs.

Active Server – This is the server where the resource hierarchy is currently running (IN SERVICE).

Standby Server – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

Primary Server – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

Secondary Server – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

Source Server – In a LifeKeeper cluster, using data replication, this is the Active Server. It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

Target Server – In a LifeKeeper cluster, using data replication, this is the Standby Server. The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

SIOS DataKeeper Terms

Replication – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

Asynchronous – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there and can be written on them.

Rate of Change – A measure of the amount of data which is changing over a set period of time.

Compression – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

Throttling – An optionally implemented mechanism to limit the bandwidth used for replication.

LifeKeeper Product Terms

Communications Path – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

Heartbeat – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

Split Brain – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

Failover – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

Switchover – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

Switchback – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

Resource – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

Extend a Resource – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously

Resource Hierarchy – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

Shared Storage – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally called I/O fencing.

Data Replication (Disk Mirroring) – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

Source – The partition on the source server used for replication. The “gold” copy of the data.

Target – The partition on the target server used for replication.

Switchable IP Address – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

11.5.2. The Evaluation Process

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux within a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

Important: Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

11.5.3. Prepare to Install DK for Linux

Hardware Requirements

Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional disk partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system (/) and boot (/boot) partitions are not eligible for replication.

Software Requirements

Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
    - º RedHat Enterprise Linux 6.x, 7.x, or 8.x
    - º CentOS Linux 6.x, 7.x, or 8.x
    - º Oracle Enterprise Linux 6.x, 7.x, or 8.x
    - º SuSE Linux Enterprise Server 11, 12, or 15
    - º See Linux Release Notes for a full list of supported Operating Systems

Current patches / security updates are recommended. Satisfied dependencies; especially if the Linux installation package selection was base/minimal you will need to refer to the dependencies documentation at Linux Dependencies It is recommended the firewall be disabled

- º # service iptables stop (systemctl stop firewalld)

- º # chkconfig iptables off (systemctl disable firewalld)

- º See

here for information regarding the ports SIOS Protection Suite for Linux uses. Disable SELinux :

- º Edit /etc/selinux/config

- º Set

SELINUX=disabled (note: permissive mode is also acceptable) Check the configuration of your /etc/hosts file

- º localhost.localdomain and localhost are the only entries that can be on 127.0.0.1

- º Create a separate entry for your hostname with a static address

GUI Authentication with PAM

- º

SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).

- º Users are identified and authenticated against the system's

PAM configuration. Privilege levels are determined from group membership as provided through PAM.

- º In order to access the

GUI, a user must be a member in one of the three LifeKeeper groups: lkadmin, lkoper or lkguest.

- º See the following

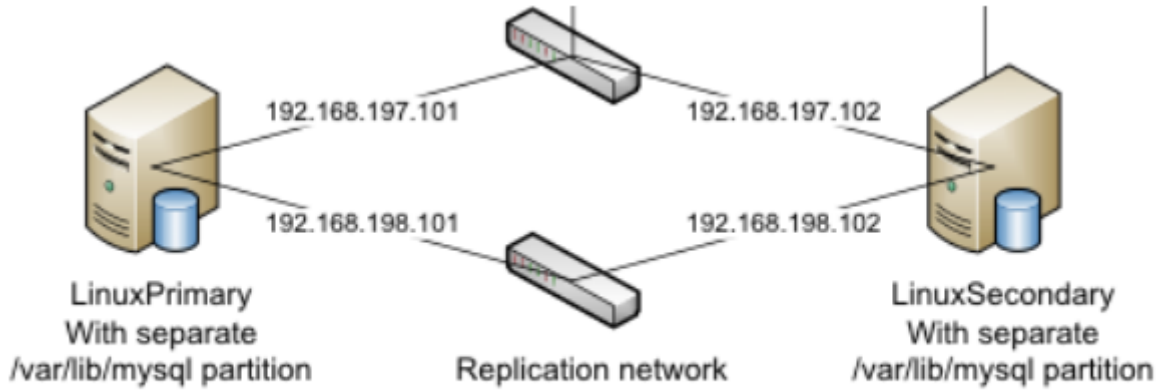
URL for more information on this topic:
Configuring GUI Users

Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi-homed, between two LAN segments (the second NIC in each server could even be connected via a cross-over cable if a second physical network is not available). The second NIC is optional in this configuration, but highly recommended in production environments to avoid a single point of failure.

In this example evaluation scenario we will be replicating direct attached storage



Network Configuration Example

IMPORTANT: Rate of Change Analysis

When replicating data in real time, its critical to ensure that you have sufficient bandwidth to keep the replication in a mirroring state at all times. To perform a Rate of Change analysis on your server, which will collect and analyze Write activity over time vs. bandwidth, refer to Measuring Rate of Change on a Linux System.

Primary Server and Secondary Servers

- Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

  Example:

  192.168.197.101 LinuxPrimary

  192.168.197.102 LinuxSecondary

- See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.

- Public Network connection(s) configured with:
  - º Static IP address
  - º Correct subnet mask

- ◦
  - º Correct gateway address

- Private Network connection(s) configured with:
  - ◦
    - º Static IP address (on a different subnet from the public network)
  - ◦
    - º Correct network mask
  - ◦
    - º No gateway IP address
  - ◦
    - º No

  DNS server addresses

11.5.4. Configure Storage for DK for Linux

Before You Begin

Ensure the following:

- You have an extra disk/partition on both servers that can be used for data replication. A target volume's size must equal to or larger than the size of its source disk/partition.

Partition local storage for use with SIOS DataKeeper for Linux

Primary Server

On your Primary server, perform the following actions:

1. Identify an existing free, unused disk partition to use as the source of the mirror. Alternatively, create a new partition. Use the "gdisk" utility to partition your disk appropriately. In this example /dev/sdb is an unused disk where we will create a single partition

   a. gdisk /dev/sdb

   b. Press "n" to create a new partition

   c.This example uses a new disk, so we will use all default values (Partition 1, entire disk and Linux filesystem partition type) Hit Enter four times to confirm these parameters.

   d. Press "w" to write the partition table

   e. Press "Y" to confirm to overwrite existing partitions

Example

[root@LinuxPrimary ~]# **gdisk /dev/sdb**

GPT fdisk (gdisk) version 0.8.10

Partition table scan:
MBR: protective
BSD: not present
APM: not present
GPT: present

Found valid GPT with protective MBR; using GPT.

Command (? for help): **n**
Partition number (1-128, default 1): **<enter>**
First sector (34-2047, default = 34) or {+-}size{KMGTP}: **<enter>**
Last sector (34-2047, default = 2047) or {+-}size{KMGTP}: **<enter>**
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): **<enter>**
Changed type of partition to 'Linux filesystem'

Command (? for help): **w**

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING PARTITIONS!!

Do you want to proceed? (Y/N): **Y**
OK; writing new GUID partition table (GPT) to /dev/sdb.
Warning: The kernel is still using the old partition table.
The new table will be used at the next reboot.
The operation has completed successfully.

[root@LinuxPrimary ~]#

2. Format the newly created disk partition

   # mkfs.ext3 /dev/sdb1

3. Mount the partition at the desired location, for example /var/lib/mysql

   # mount /dev/sdb1 /var/lib/mysql

4. Note: there is no need to add an entry to /etc/fstab. Lifekeeper will take care of mounting this automatically.

Result
[root@LinuxPrimary ~]# df /var/lib/mysql
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sdb1 253855 11083 229666 5% /var/lib/mysql

Secondary Server

5. On your Secondary server, only perform Step #1 above, where you partition the disk. The size of the Target disk/partition needs to be the same size, or greater, than our Source disk/partition. There is no need to format or mount the filesystem.

11.5.5. Install SIOS Protection Suite for Linux

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

SPS for Linux Component	Install Location
LifeKeeper Software	/opt/LifeKeeper
LifeKeeper Config File	/etc/default/LifeKeeper

Perform the following actions on **both** Primary and Secondary server.

Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.

2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After successful download you will have downloaded contents similar to the follow directory listing:

      [root@LinuxPrimary ~]# ls -l <directory> total 63680

      -rw-r—r— 1 root root 23163 May 30 14:03 EULA.pdf

      -rw-r—r— 1 root root 536 May 30 14:03 readme.txt

      -rw-r—r— 1 root root 65179648 May 30 14:03 sps.img

3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

      # mount -o loop sps.img /mnt

```
# cd /mnt

# ./setup
```

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

```
# cd /root

# umount /mnt
```

Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

```
# /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic
```

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

```
# /opt/LifeKeeper/bin/lklicmgr

License File: 20101230.lic
```

Product	Type	Expiry
LifeKeeper for Linux	Eval	27 Mar 2013 (87 days)

SIOS Data Replication ARK	Eval	27 Mar 2013 (87 days)

Start the SIOS Protection Suite for Linux

1. Start:

```
# /opt/LifeKeeper/bin/lkstart
```

11.5.6. Configure the Cluster – DK for Linux

Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

Important Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously

Access the LifeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application, or as an applet within your Java-Enabled Web Browser.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 errors.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

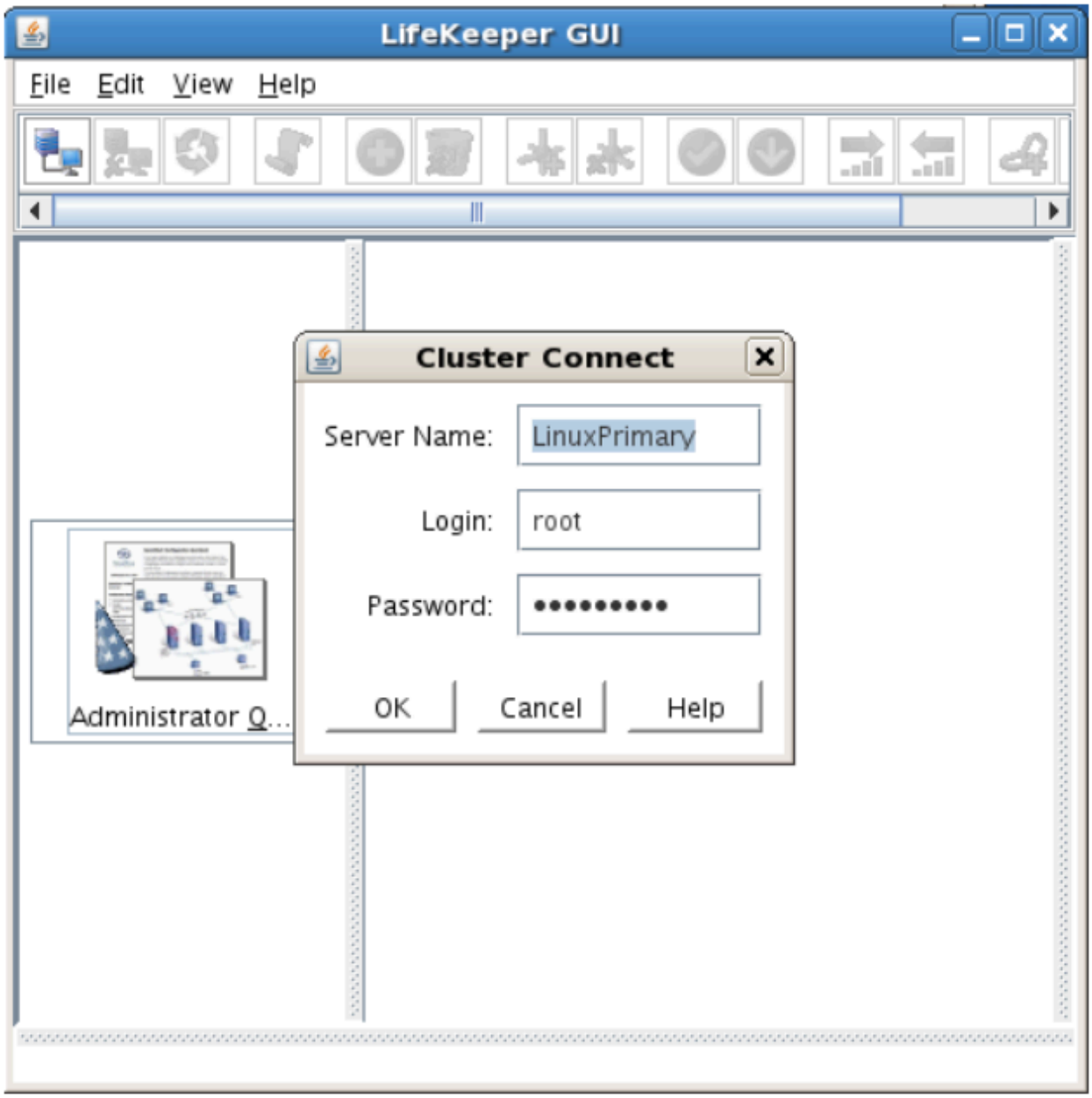
   a. /opt/LifeKeeper/bin/lkGUIapp &

3. To Connect to the LifeKeeper GUI Applet from a Web Browser, go to:

   a. http://<hostname>:81

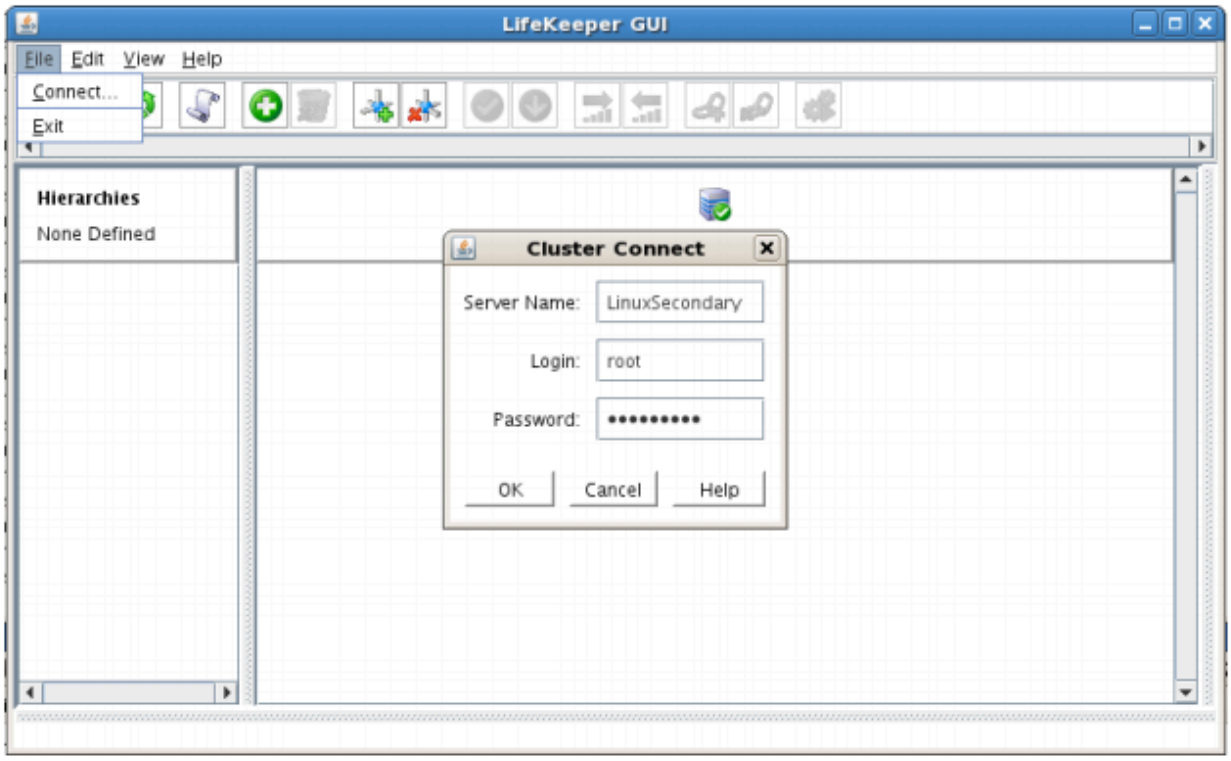
4. Enter the name of the server you wish to connect to (this field will be populated with the name of

the server you are on, if you are running the GUI from a server with LifeKeeper installed) along with your root credentials and click OK.



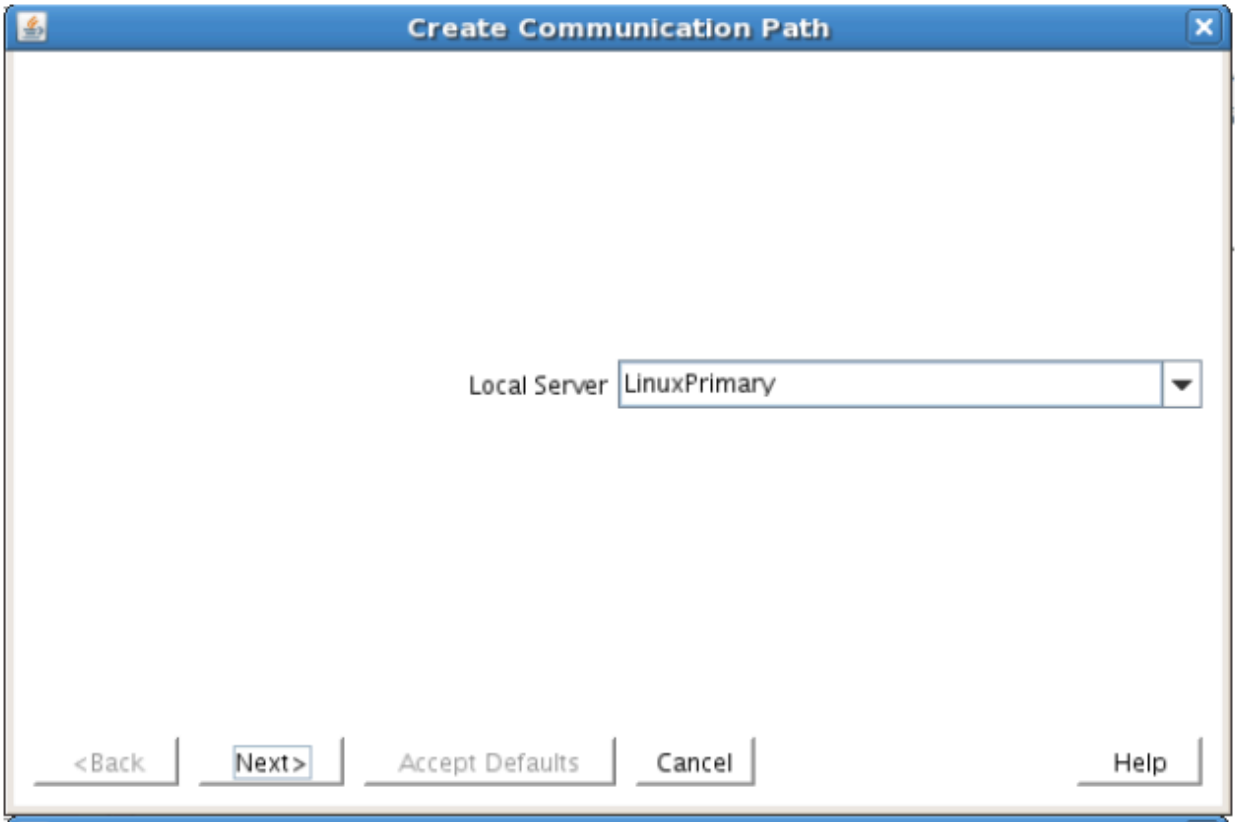
Create Communication (Comm) Paths

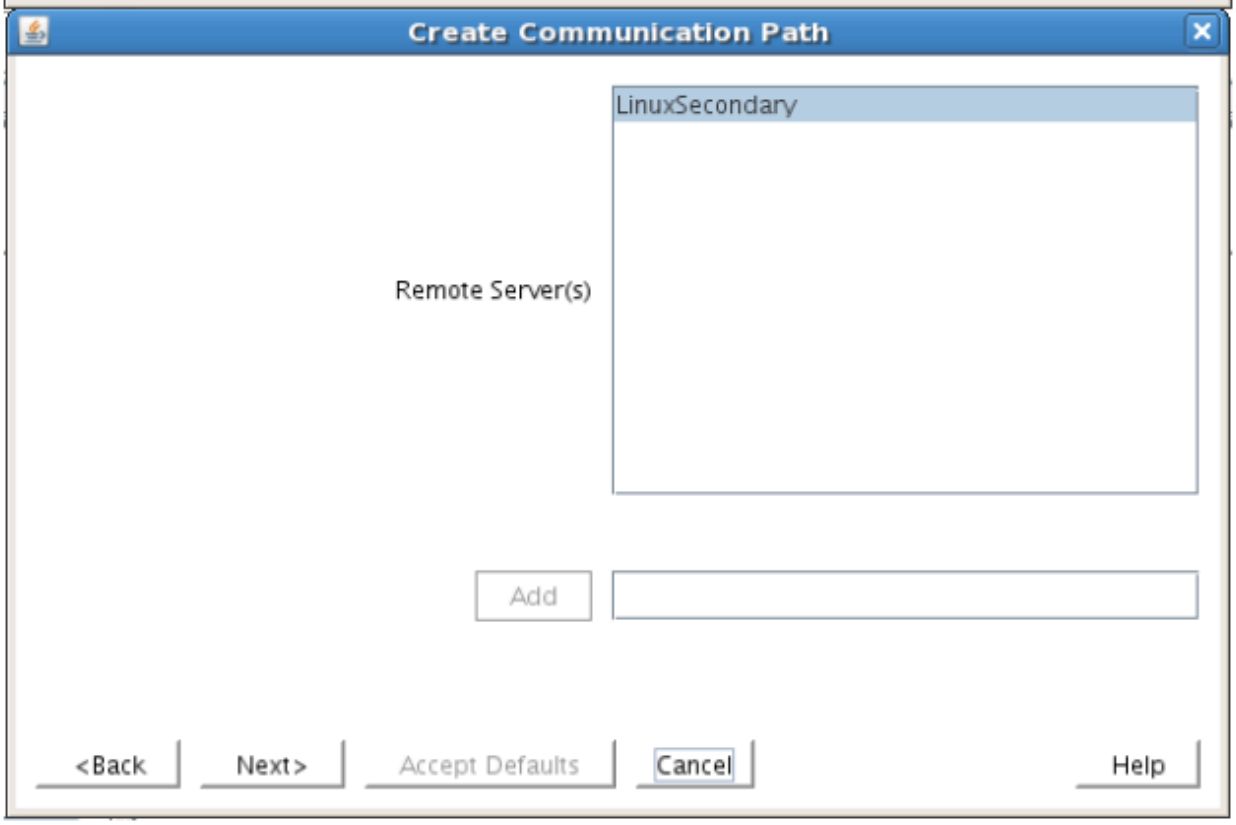
5. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



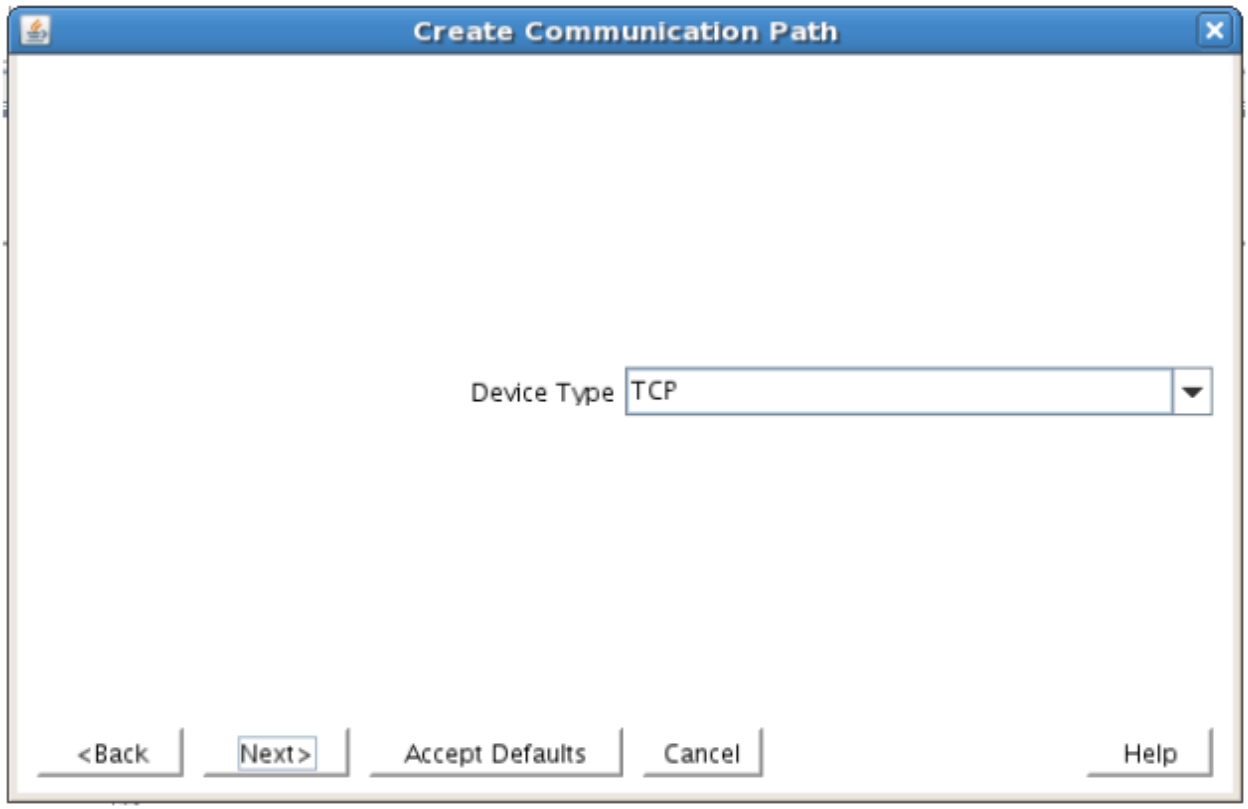
6. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

7. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

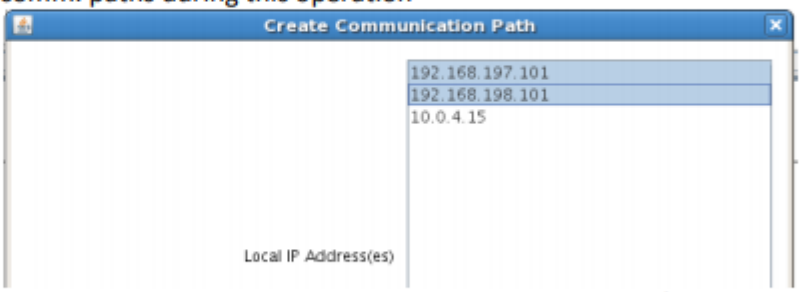


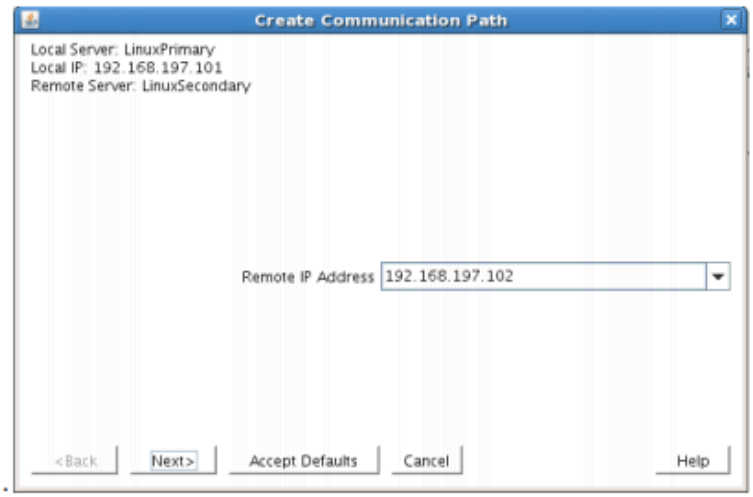


8. Select TCP for Device Type and Click Next.



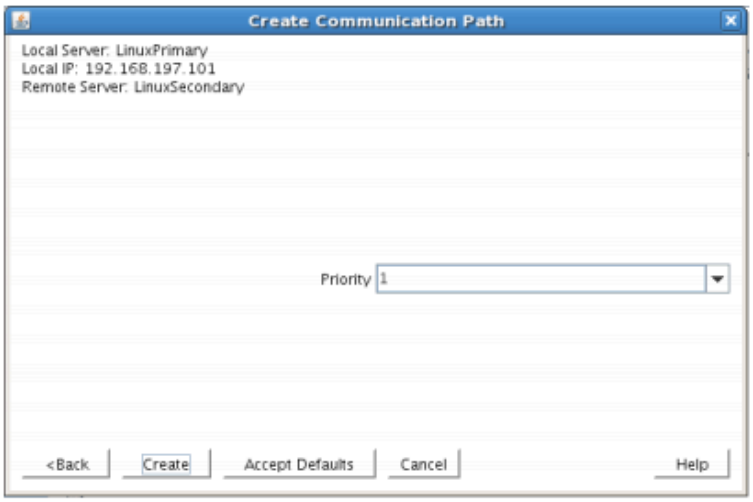
9. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

Field	Tips
For TCP/IP Comm Path...	
Local IP Address	Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation 
Remote IP Address	Choose the IP address to be used by the remote server for this comm path

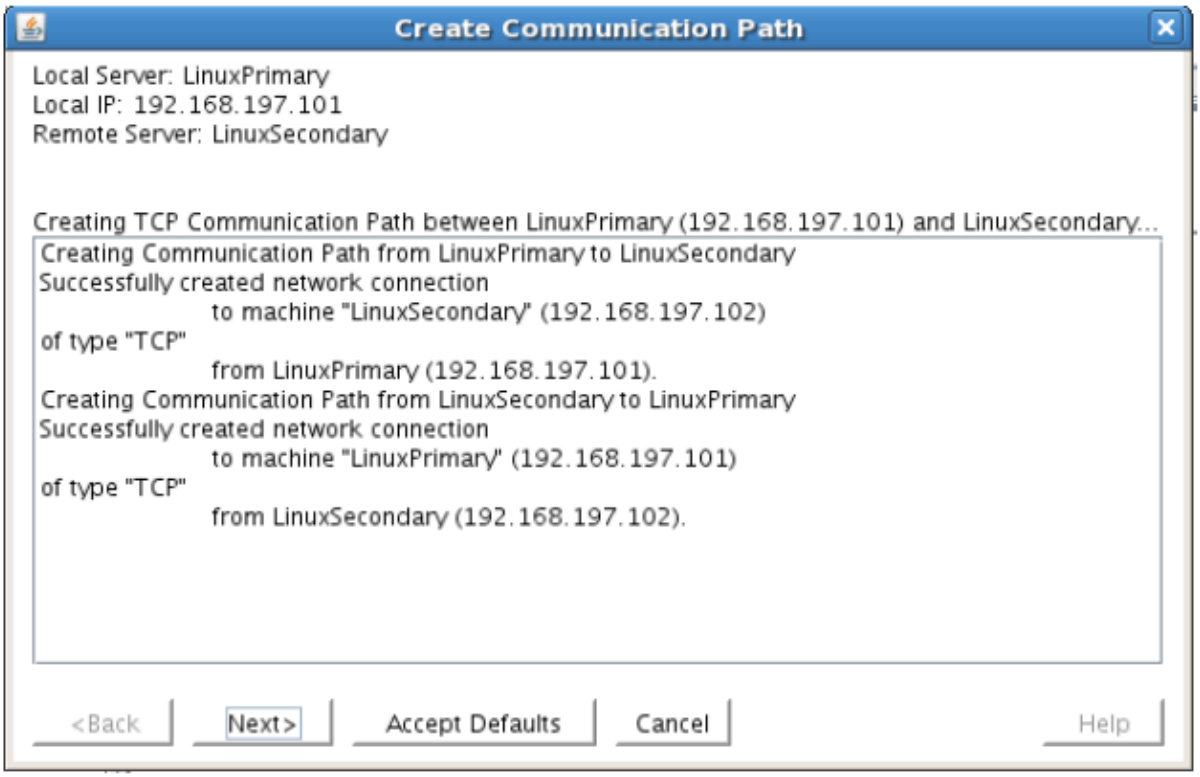


Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority



10. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



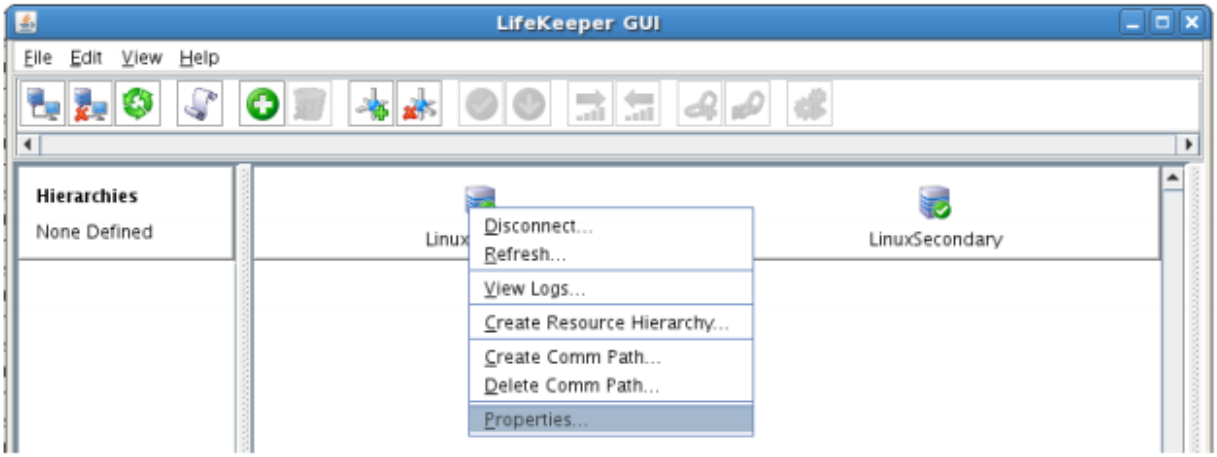
If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

11. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

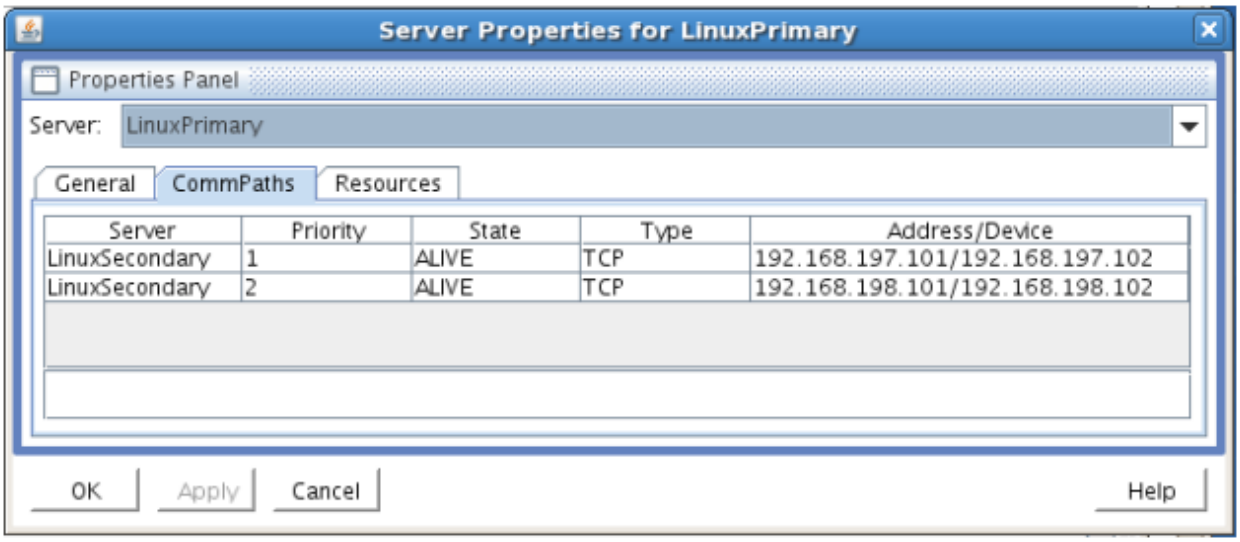
Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.



2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.



Create the LifeKeeper Hierarchy

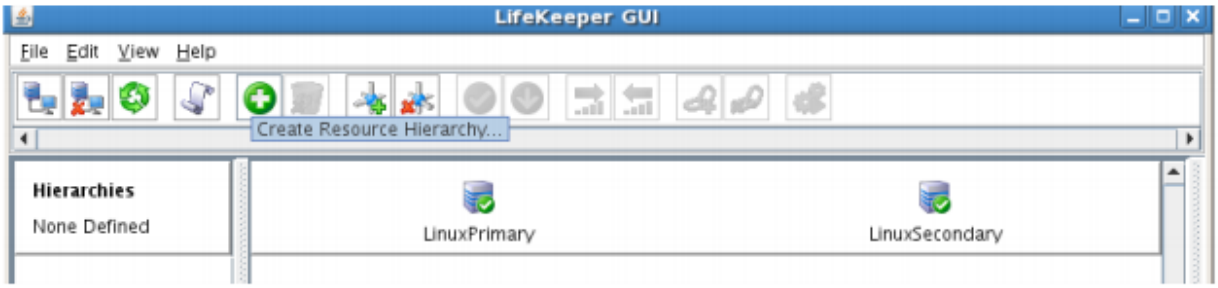
Create a Mirror and Begin Data Replication

In this section we will setup and configure the Data Replication resource, which be used to synchronize our MySQL's data between cluster nodes. The data we will replicate resides in the /var/lib/mysql partition on our Primary cluster node

Please note:

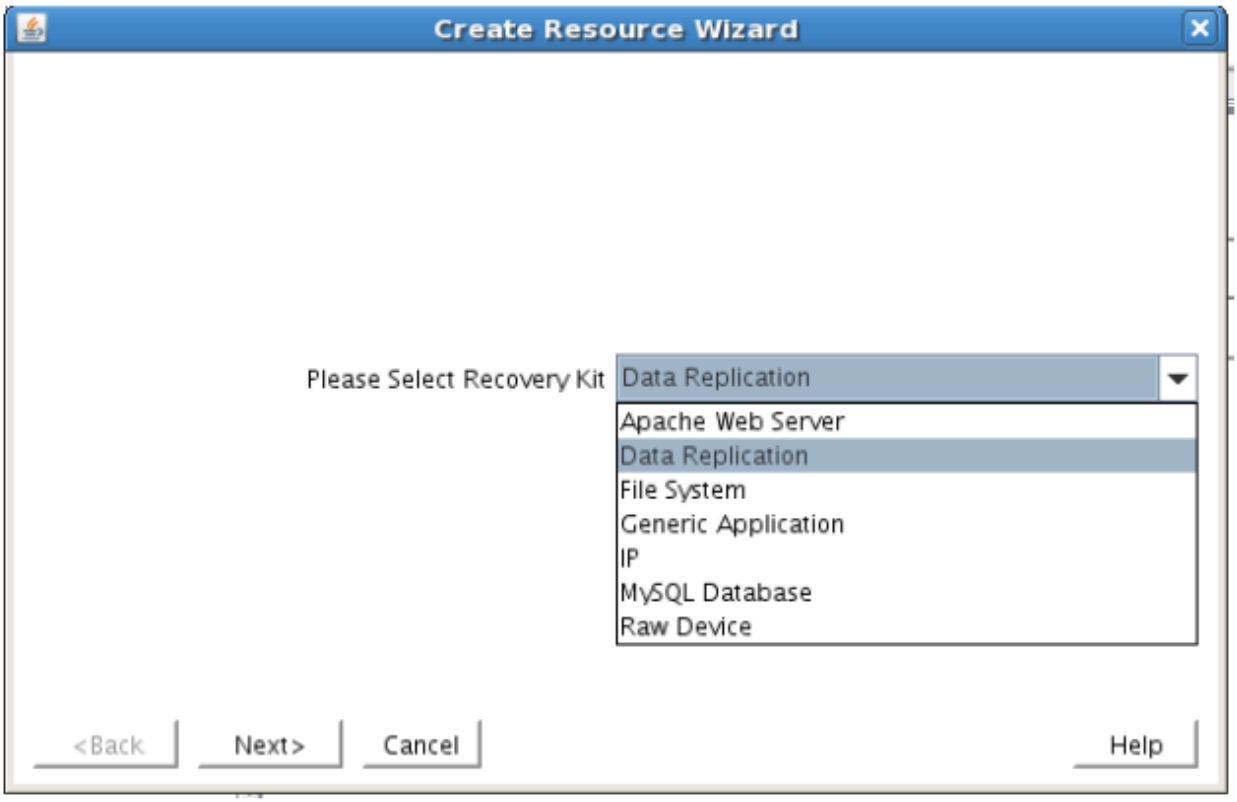
- The source volume to be replicated must be mounted on the Primary server
- The target volume, which will received replicated data, must NOT be mounted on the Secondary server.
- The target volume's size must equal to or larger than the size of its source volume.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.



The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select Data Replication and click Next.

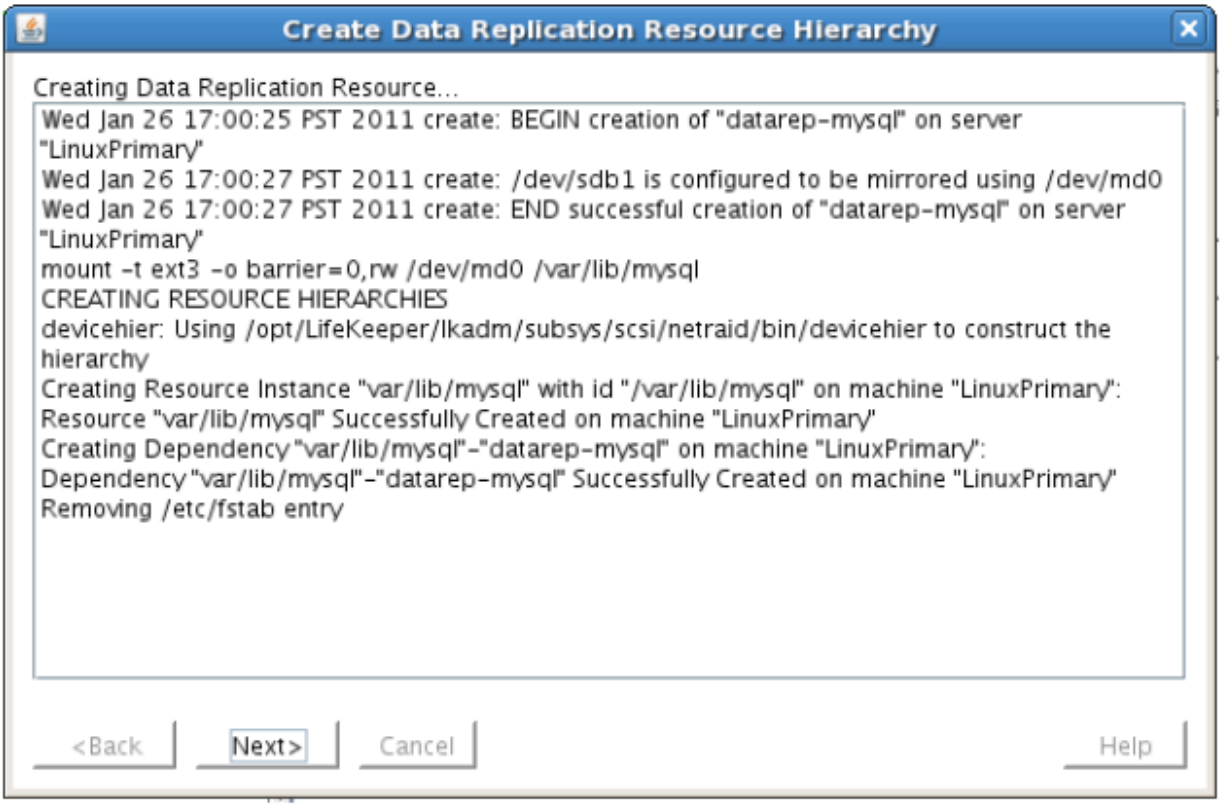


3. Follow the Data Replication wizard, and enter the following values:

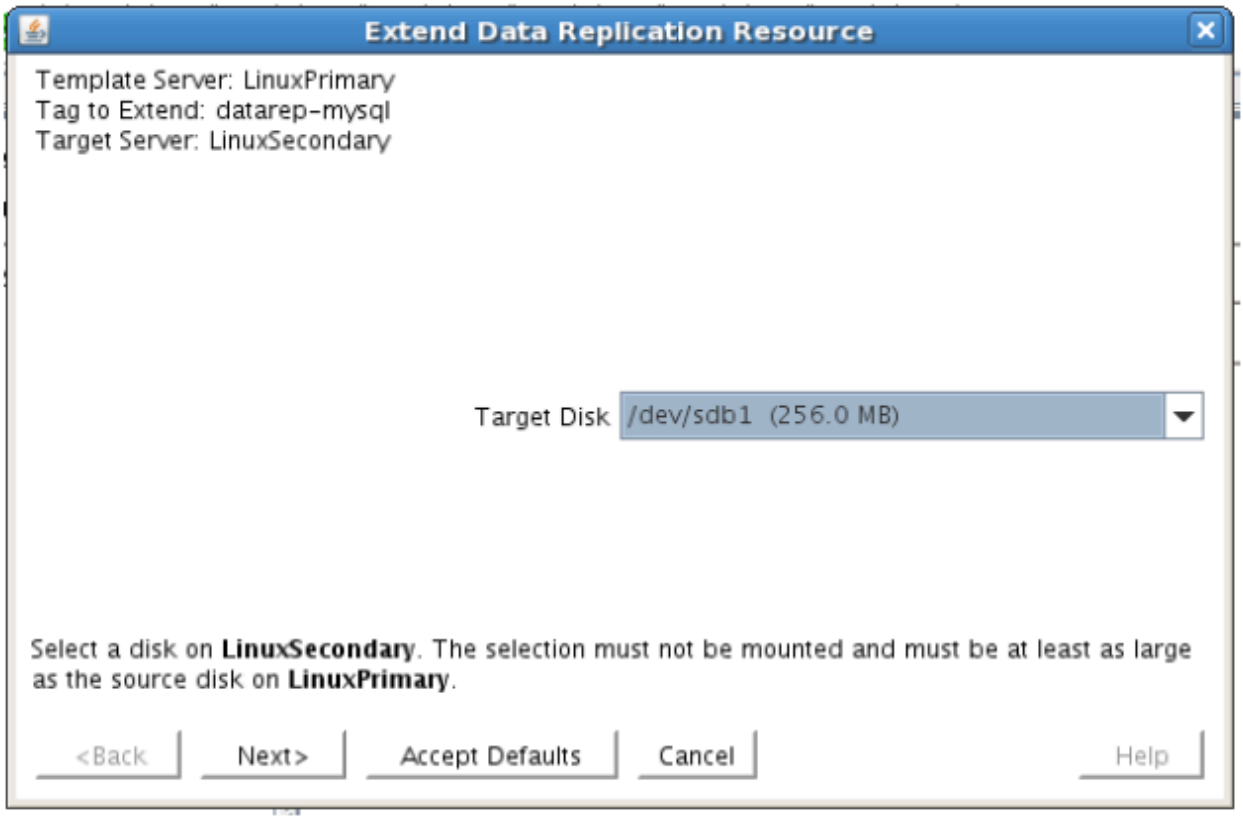
Field	Recommended Entries or Notes
Switchback Type Server	Intelligent LinuxPrimary (Primary Cluster Node, i.e. Mirror Source)
Hierarchy Type Existing Mount Point	Select: “**Replicate Existing Filesystem**” At this step you will select the mounted partition to replicate. In our example, select “/var/lib/mysql”
Data Replication Resource Tag File System	Leave as default Leave as default Leave as default (Note: if using high speed SSD storage you will want to create a

Resource Tag Bitmap File	small partition and use it for bitmap placement, i.e. /bitmaps)
Enable Asynchronous Replication	Leave as default (Yes)

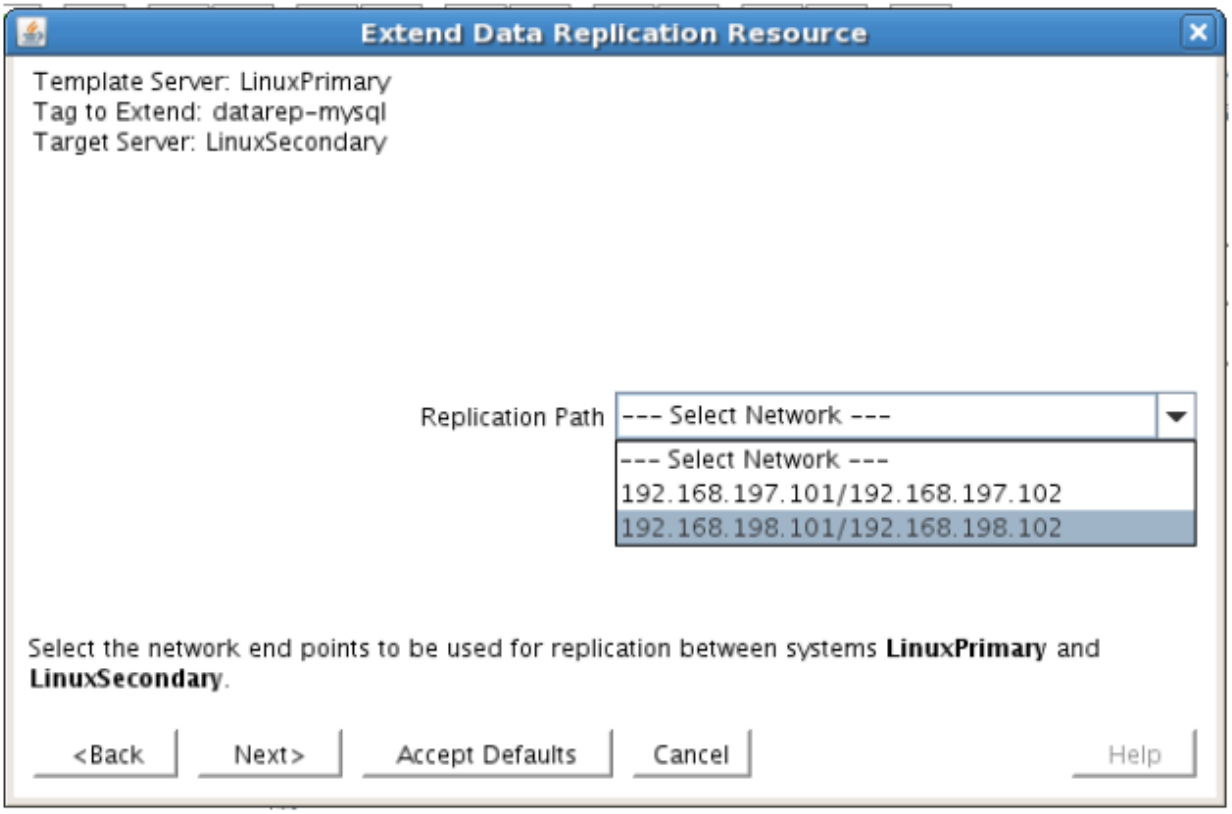
4. Click Next to begin creation of the Data Replication resource hierarchy. Status will be displayed in the GUI as follows:



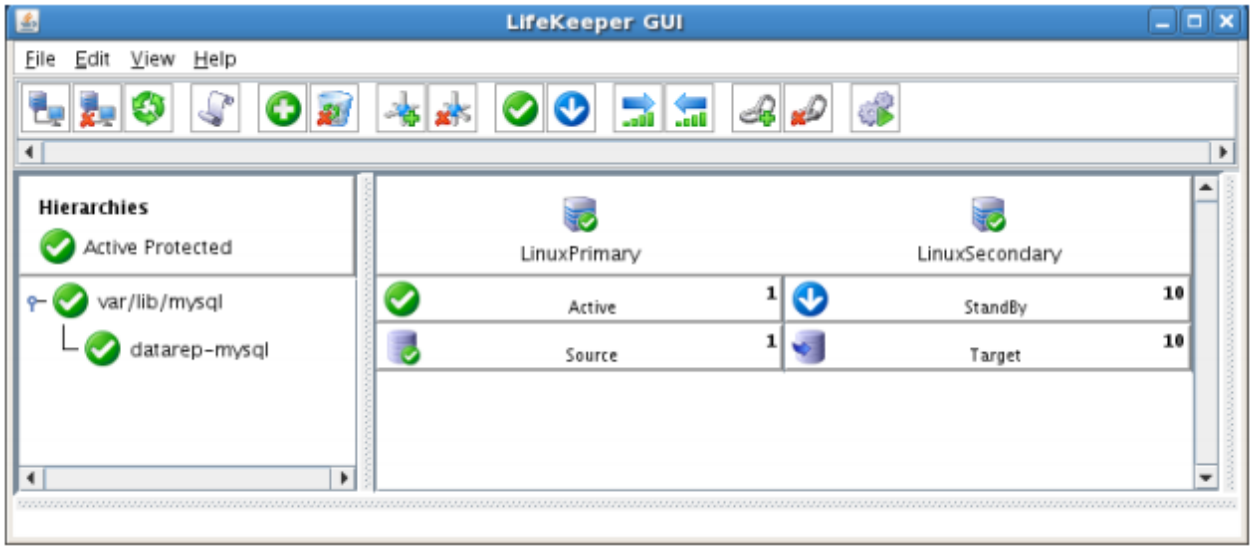
5. Click Next to begin the process to Extend the Data Replication Resource. Select all default settings. When it asks for the target disk, select a free partition on your Target server which is the same size (or greater) than the Source Volume we are replicating. This partition should NOT be mounted on the Target system.



6. Continue through the wizard, and you will be prompted to select the network you would like replication to take place over. In general, it's a best practice to separate your user/application and your replication traffic. In our example setup we will replicate over our backend network, 192.168.198.X



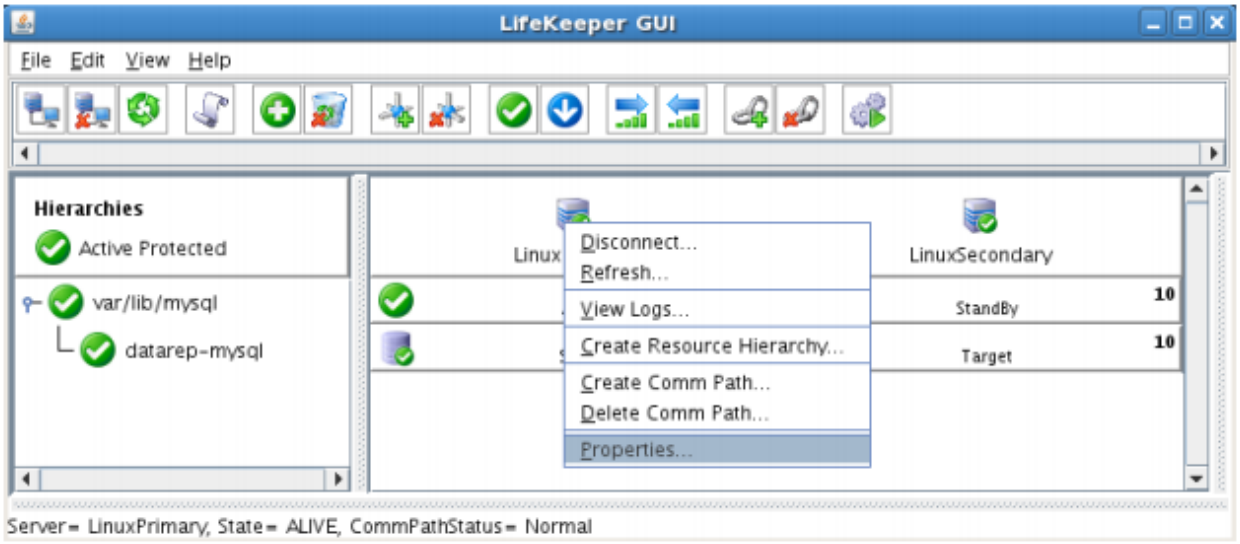
7. Click Next and continue through the wizard. Once completed, your resource hierarchy will look as follows



Disable Automatic Failover

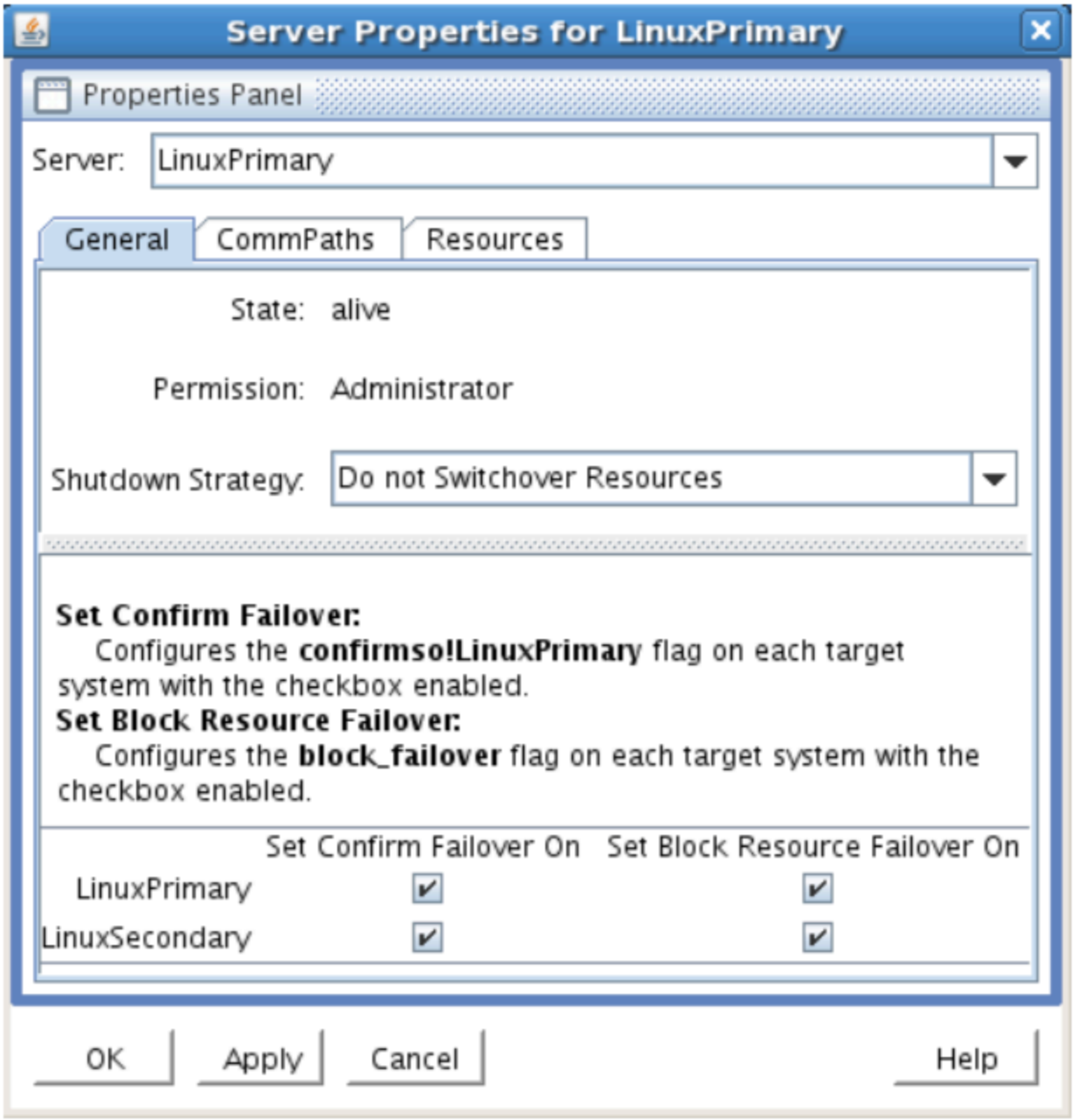
In this section we will review the procedure for disabling automatic failover to the standby server.

1. From the LifeKeeper GUI, right click on one of the cluster nodes and select Properties.

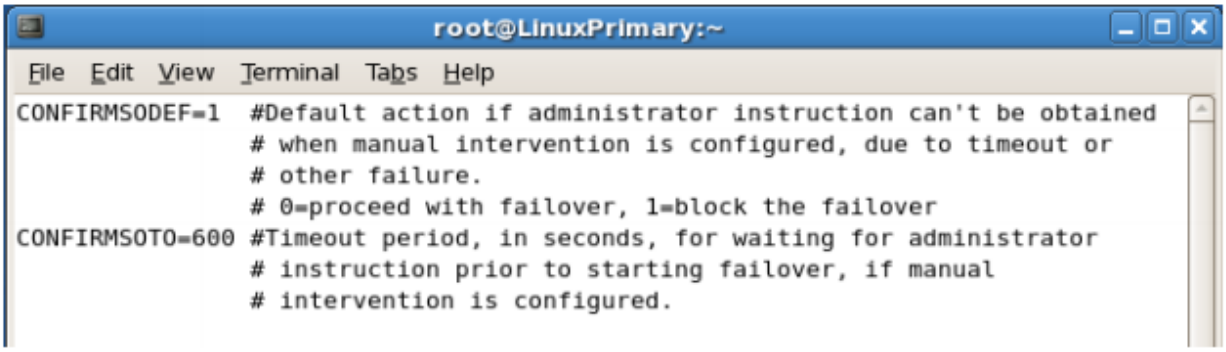


2. Select the Source server from the “Server:” drop down at the top of the window

3. Once the Server Properties window loads, check all boxes at the bottom of the page. This will prevent any automatic failovers from happening.



4. Click Apply
5. Repeat steps 2-4, this time selecting the Target server from the “Server:” drop down
6. Next, edit /etc/default/LifeKeeper on both nodes
    a. Set CONFIRMSODEF=1 (change from 0 to 1)


root@LinuxPrimary:~
File Edit View Terminal Tabs Help
CONFIRMSODEF=1 #Default action if administrator instruction can't be obtained
when manual intervention is configured, due to timeout or
other failure.
0=proceed with failover, 1=block the failover
CONFIRMSOTO=600 #Timeout period, in seconds, for waiting for administrator
instruction prior to starting failover, if manual
intervention is configured.

11.5.7. Test Your DK for Linux Environment

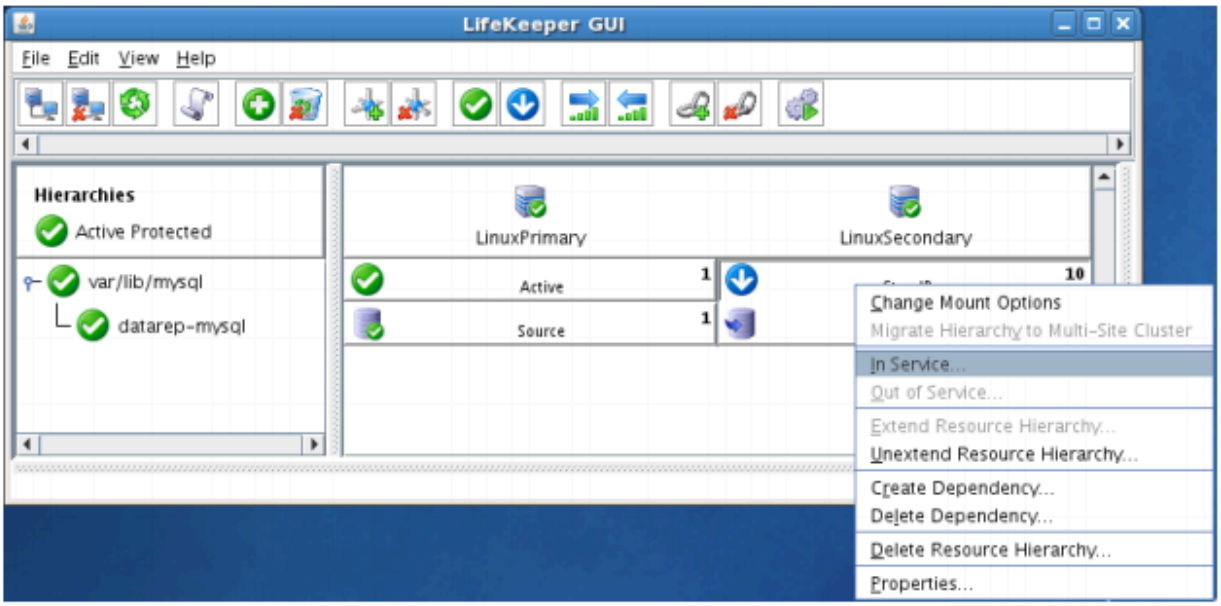
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

> **Note:** For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

1. Manual Switchover of the Mirror to Secondary Server

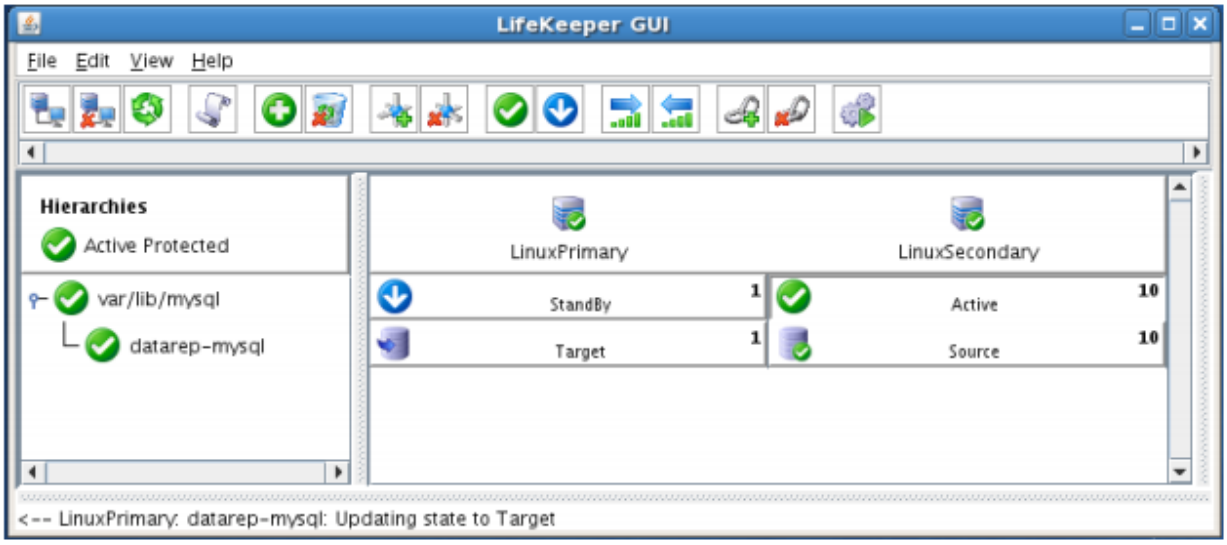
   Procedure:

- From the LifeKeeper GUI, right click on the top of the resource hierarchy on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



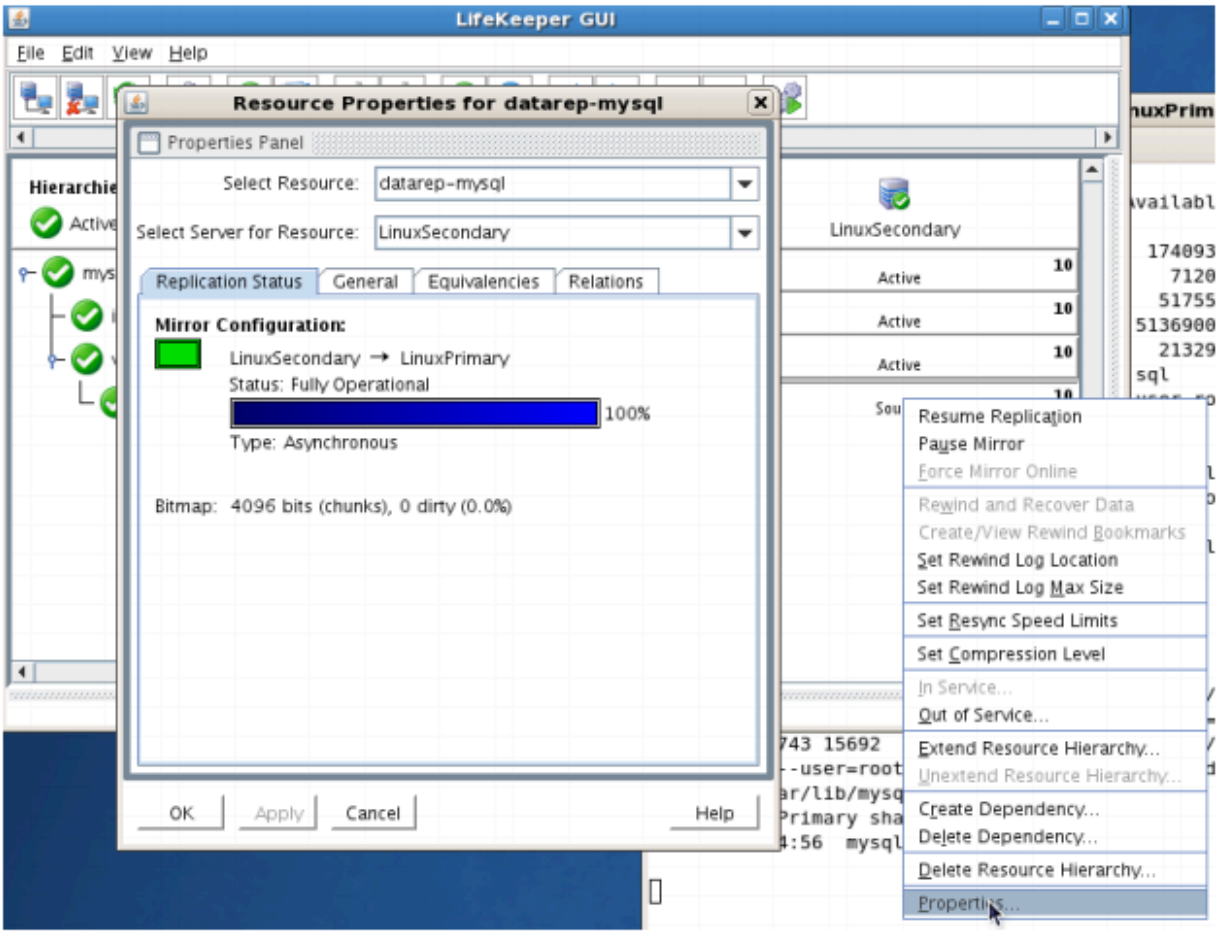
Expected Result:

- All resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources, all resources will be brought in service on LINUXSECONDARY.
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXSECONDARY -> LINUXPRIMARY
- At this point, all resources are now active on LINUXSECONDARY.



Tests/Verification:

- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties



- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXSECONDARY

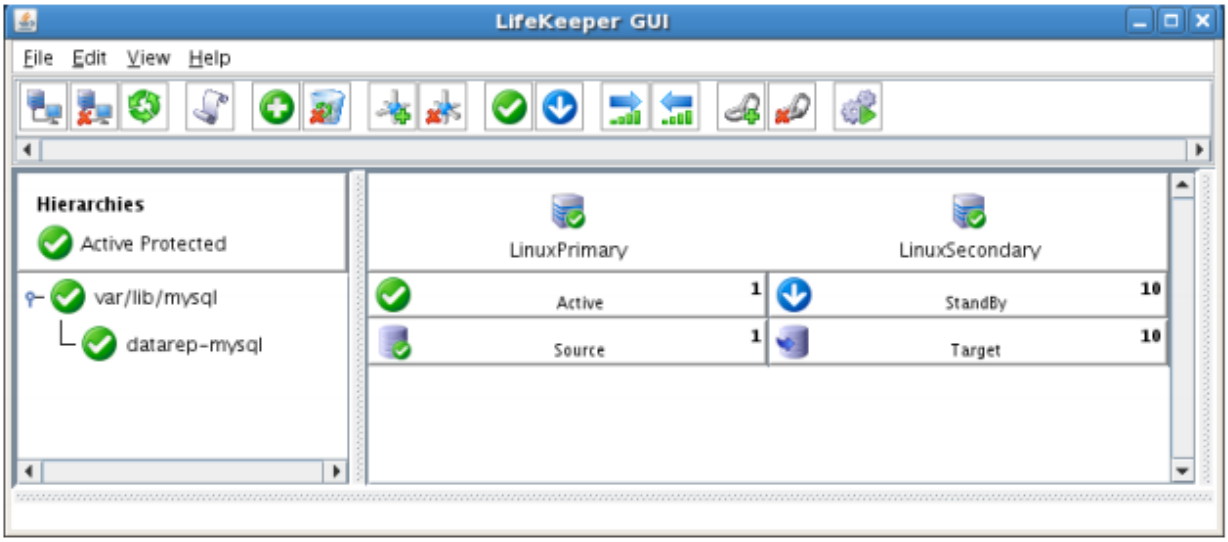
2. Manual Switchover of the Mirror back to Primary Server

Procedure:

- From the LifeKeeper GUI, right click on the top level of the resource hierarchy on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

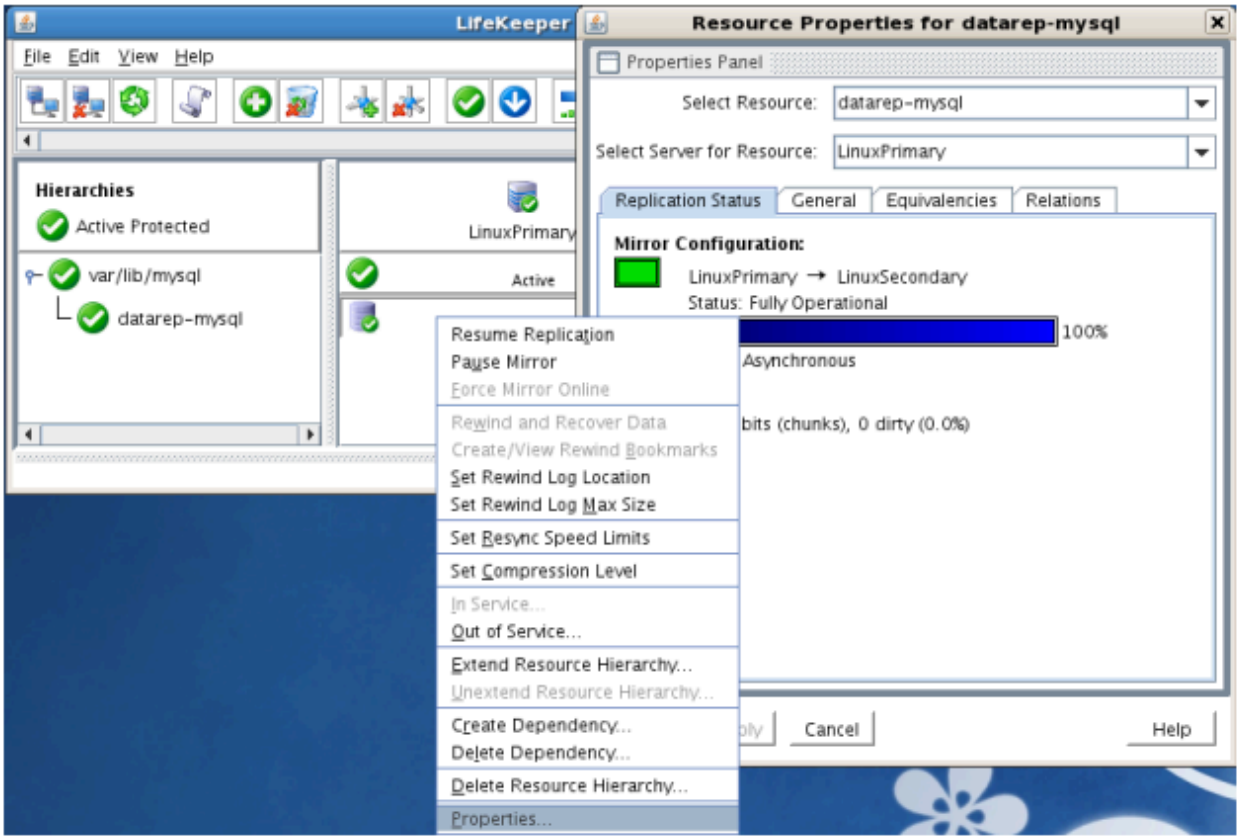
Expected Result:

- All resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources, all resources will be brought in service on LINUXPRIMARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXPRIMARY -> LINUXSECONDARY



Tests/Verification:

- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties



- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY

11.6. MySQL Cluster with Data Replication (“Shared Nothing” Cluster)

Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product to make MySQL highly available. If MySQL is not already installed, please allocate some time to install it on your servers. Once this task has been completed, you may install and configure SIOS Protection Suite for Linux.

There are five phases in this process:

- Prepare to Install
- Configure Storage
- Install and Configure MySQL
- Install SIOS Protection Suite for Linux
- Configure your LifeKeeper Cluster
- Test Your Environment

11.6.1. Terms to Know

The following terms are used throughout this document and, while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

Network Communication Terms

Crossover cable – A cable used to directly connect computing devices together, instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

Types of LifeKeeper Servers

Server – A computer system dedicated to running software application programs.

Active Server – This is the server where the resource hierarchy is currently running (IN SERVICE).

Standby Server – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

Primary Server – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

Secondary Server – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

Source Server – In a LifeKeeper cluster, using data replication, this is the Active Server. It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

Target Server – In a LifeKeeper cluster, using data replication, this is the Standby Server. The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

SIOS Data Replication Terms

Replication – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

Synchronous – A replication scheme in which the data is confirmed written and valid on the target before the write operation occurs on the source disk through a series of information exchanges. Synchronous mirrors should only be implemented on high speed (100Mbps+) networks due to the network overhead involved.

Asynchronous – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there and can be written on them.

Rate of Change – A measure of the amount of data which is changing over a set period of time.

Compression – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

Throttling – An optionally implemented mechanism to limit the bandwidth used for replication.

LifeKeeper Product Terms

Communications Path – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

Heartbeat – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

Split Brain – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

Failover – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

Switchover – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

Switchback – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

Resource – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

Extend a Resource – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously.

Resource Hierarchy – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

Shared Storage – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally called I/O fencing.

Data Replication (Disk Mirroring) – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

Source – The partition on the source server used for replication. The “gold” copy of the data.

Target – The partition on the target server used for replication.

Switchable IP Address – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

11.6.2. The Evaluation Process – MySQL Cluster

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux in a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

Important: Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

11.6.3. Prepare to Install

Hardware Requirements

Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system (/) and boot (/boot) partitions are not eligible for replication.

Note: You may use more than one partition for replicated data, allowing for separation of multiple database files and/or log directories for performance reasons.

Client

This system is not required but is recommended for testing the cluster environment.

- A standard linux terminal running the MySQL client can be used to test the configuration.

Software Requirements

Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
  -
    - RedHat Enterprise Linux 5 (5.4+ recommended) or 6.x
    - CentOS Linux 5 (5.4+ recommended) or 6.x
    - Oracle Enterprise Linux 5 (5.4+ recommended), 6.3, 6.4 (**RedHat Compatibility Kernel Only**)

- º SuSE Linux Enterprise Server 10 or 11 (11 recommended)
- º See Linux Release Notes for a full list of supported Operating Systems

Current patches / security updates are recommended Satisfied dependencies; especially if the Linux installation package selection was base/minimal you will need to refer to the dependencies documentation at Linux Dependencies It is recommended that IPtables is disabled

- º # /etc/init.d/iptables off
- º # chkconfig iptables off
- º See

Running LifeKeeper With a Firewall for information regarding the ports SIOS Protection Suite for Linux uses. Disable SELinux :

- º Edit /etc/selinux/config<
- º Set

SELINUX=disabled (note: permissive mode is also acceptable) Check the configuration of your /etc/hosts file

- º localhost.localdomain and localhost are the only entries that can be on 127.0.0.1
- º Create a separate entry for your hostname with a static address

GUI Authentication with PAM

- º

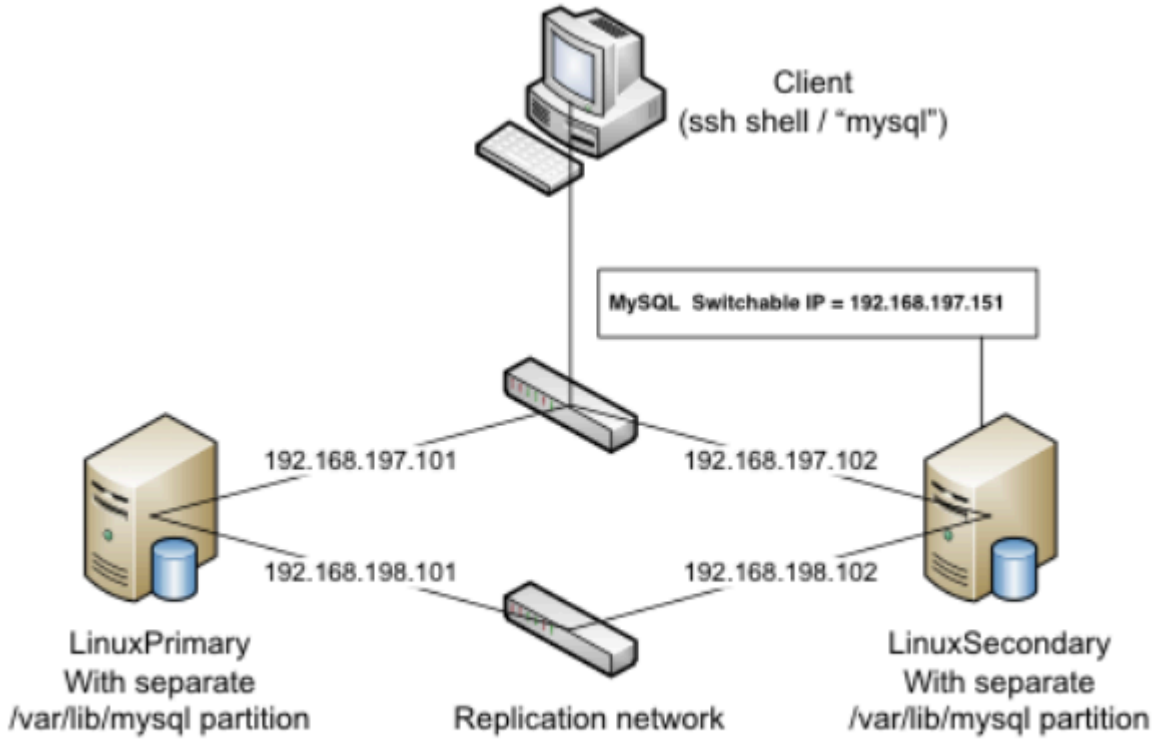
SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).
º Users are identified and authenticated against the system's PAM configuration. Privilege levels are determined from group membership as provided through PAM.
º In order to access the GUI, a user must be a member in one of the three LifeKeeper groups: lkadmin, lkoper or lkguest.
º See the following URL for more information on this topic:
See Configuring GUI Users for more information.

Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi-homed, between two LAN segments (the second NIC in each server could even be connected via a cross-over cable if a second physical network is not available). The second NIC is optional in this configuration, but highly recommended in production environments to avoid a single point of failure.

In this example evaluation scenario we will be leveraging local, replicated storage.



Network Configuration Example

Primary Server and Secondary Servers

- Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

  Example:
  192.168.197.101 LinuxPrimary
  192.168.197.102 LinuxSecondary

- See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.
- Public Network connection(s) configured with:
  - º Static IP address
  - º Correct subnet mask
  - º Correct gateway address
  - º Correct

  DNS server address(es)

Private Network connection(s) configured with:

- º Static IP address (on a different subnet from the public network)
- º Correct network mask
- º No gateway IP address
- º No

DNS server addresses

Client

Must be able to communicate on the same subnet/network as the servers Public interface addresses. In our example, this is the 192.168.197.0/24 network.

11.6.4. Configure Storage

Before You Begin

Ensure the following:

- You have an extra disk/partition on both servers that can be used for data replication. A target volume's size must be equal to or larger than the size of its source disk/partition.

Partition local storage for use with SIOS Data Replication

Primary Server

On your Primary server, perform the following actions:

1. Identify an existing free, unused disk partition to use as our Apache repository. Alternatively, create a new partition. Use the "gdisk" utility to partition your disk appropriately. In this example /dev/sdb is an unused disk where we will create a single partition

   a. gdisk /dev/sdb

   b. Press "n" to create a new partition

   c.This example uses a new disk, so we will use all default values (Partition 1, entire disk and Linux filesystem partition type) Hit Enter four times to confirm these parameters.

   d. Press "w" to write the partition table

   e. Press "Y" to confirm to overwrite existing partitions

Example

[root@LinuxPrimary ~]# **gdisk /dev/sdb**

GPT fdisk (gdisk) version 0.8.10

Partition table scan:
MBR: protective
BSD: not present
APM: not present
GPT: present

Found valid GPT with protective MBR; using GPT.

Command (? for help): **n**

Partition number (1-128, default 1): **<enter>**
First sector (34-2047, default = 34) or {+-}size{KMGTP}: **<enter>**
Last sector (34-2047, default = 2047) or {+-}size{KMGTP}: **<enter>**
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): **<enter>**
Changed type of partition to 'Linux filesystem'

Command (? for help): **w**

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING PARTITIONS!!

Do you want to proceed? (Y/N): **Y**
OK; writing new GUID partition table (GPT) to /dev/sdb.
Warning: The kernel is still using the old partition table.
The new table will be used at the next reboot.
The operation has completed successfully.

[root@LinuxPrimary ~]#

2. Format the newly created disk partition

   # mkfs.ext3 /dev/sdb1

3. Mount the partition temporarily at /mnt

   # mount /dev/sdb1 /mnt

4. Move any existing data from /var/lib/mysql/ into this new disk partition (assumes a default MySQL configuration)

   # cd /var/lib/mysql
   # mv * /mnt

5. Remount /dev/sdb1 at /var/lib/mysql

   # cd /root
   # umount /mnt
   # mount /dev/sdb1 /var/lib/mysql

6. Note: there is no need to add this partition to /etc/fstab. Lifekeeper will take care of mounting this automatically.

Result:

[root@LinuxPrimary ~]# df /var/lib/mysql
Filesystem 1K-blocks Used Available Use% Mounted on

/dev/sdb1 253855 11083 229666 5% /var/lib/mysql<

Secondary Server

7. On your Secondary server, only perform Step #1 above, where you partition the disk. The size of the Target disk/partition needs to be the same size, or greater, than our Source disk/partition.

11.6.5. Install, Configure, and Start MySQL

Primary Server

On your Primary server, perform the following actions:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

   ```
   # yum install mysql mysql-server
   ```

2. Verify that your local disk partition is still mounted at /var/lib/mysql via the “df” command

3. If this is a fresh MySQL install, initialize a sample MySQL database:

   ```
   # /usr/bin/mysql_install_db – –datadir=”/var/lib/mysql” – –user=mysql
   ```

4. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

   ```
   # chown –R mysql:mysql /var/lib/mysql
   ```

   ```
   # chmod 755 /var/lib/mysql
   ```

5. Finally, manually start the MySQL daemon from the command line.

Do Not start it via the “service” command, or the /etc/init.d/ scripts

```
# mysqld_safe – –user=root – –socket=/var/lib/mysql/mysql.sock – –port=3306 –
–datadir=/var/lib/mysql – –log &
```

6. Verify MySQL is running by connecting with the mysql client:

   ```
   [root@LinuxPrimary mysql]# mysql
   ```

   ```
   Welcome to the MySQL monitor. Commands end with ; or \g.
   ```

   ```
   Your MySQL connection id is 2
   ```

   ```
   Server version: 5.0.77-log Source distribution
   ```

   ```
   Type ‘help;’ or ‘\h’ for help. Type ‘\c’ to clear the buffer.
   ```

   ```
   mysql> exit
   ```

Bye

[root@LinuxPrimary mysql]#

7. Update the root password for your mysql configuration. In this example we set the MySQL root password to “SteelEye”

   # echo “update user set Password=PASSWORD where User=‘root’; flush privileges” | mysql mysql

8. Verify your new password:

   # mysql mysql –u root –p

   (Enter “SteelEye” as the password)

   #exit

9. Create a MySQL configuration file. We will place this in the same shared directory (/var/lib/mysql/my.cnf)

   # vi /var/lib/mysql/my.cnf

Example

cat /var/lib/mysql/my.cnf

[mysqld]

datadir=/var/lib/mysql

socket=/var/lib/mysql/mysql.sock

pid-file=/var/lib/mysql/mysqld.pid

user=root

port=3306

Default to using old password format for compatibility with mysql 3.x

clients (those using the mysqlclient10 compatibility package).

old_passwords=1

Disabling symbolic-links is recommended to prevent assorted security risks;

```
# to do so, uncomment this line:

# symbolic-links=0

[mysqld_safe]

log-error=/var/log/mysqld.log

pid-file=/var/run/mysqld/mysqld.pid

[client]

user=root

password=SteelEye
```

10. Delete the original MySQL configuration file, located in /etc

```
# rm /etc/my.cnf
```

Secondary Server

On your Secondary Server:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

```
# yum install mysql mysql-server
```

2. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

```
# chown –R mysql:mysql /var/lib/mysql

# chmod 755 /var/lib/mysql
```

3. There is no need to perform any of the additional steps taken on the Primary Server

11.6.6. Install SIOS Protection Suite for Linux – MySQL Cluster

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

SPS for Linux Component	Install Location
LifeKeeper Software	/opt/LifeKeeper
LifeKeeper Config File	/etc/default/LifeKeeper

Perform the following actions on **both** Primary and Secondary server.

Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.

2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After you have successfully downloaded the software you will have downloaded contents similar to the follow directory listing:

   [root@LinuxPrimary ~]# ls -l <directory> total 63680

   -rw-r--r-- 1 root root 23163 May 30 14:03 EULA.pdf

   -rw-r--r-- 1 root root 536 May 30 14:03 readme.txt

   -rw-r--r-- 1 root root 65179648 May 30 14:03 sps.img

3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

   # mount -o loop sps.img /mnt

```
# cd /mnt

# ./setup
```

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point.

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkSQL-<version>.noarch.rpm

   b. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

```
# cd /root

# umount /mnt
```

Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

```
# /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic
```

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

```
# /opt/LifeKeeper/bin/lklicmgr
License File: 20101230.lic
```

Product	Type	Expiry

LifeKeeper for Linux	Eval	27 Mar 2013 (87 days)
SIOS Data Replication ARK	Eval	27 Mar 2013 (87 days)
MySQL Recovery Kit	Eval	27 Mar 2013 (87 days)

Start the SIOS Protection Suite for Linux

1. Start

```
# /opt/LifeKeeper/bin/lkstart
```

11.6.7. Configure the Cluster

Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

Important Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously.

Access the LikeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application, or as an applet within your Java-Enabled Web Browser.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 error.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

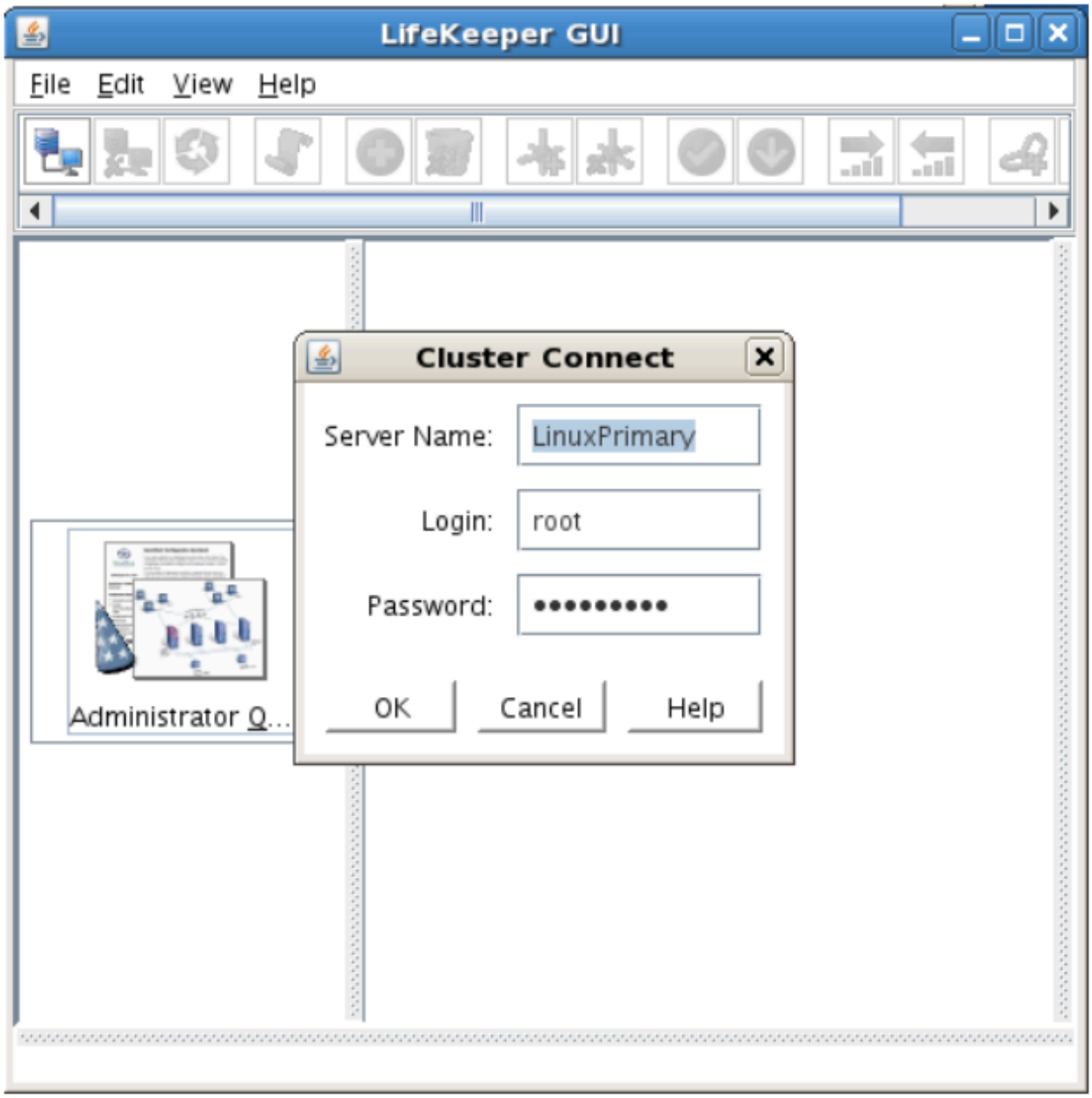
   /opt/LifeKeeper/bin/lkGUIapp &

3. To Connect to the LifeKeeper GUI Applet from a Web Browser, go to:

   http://<hostname>:81

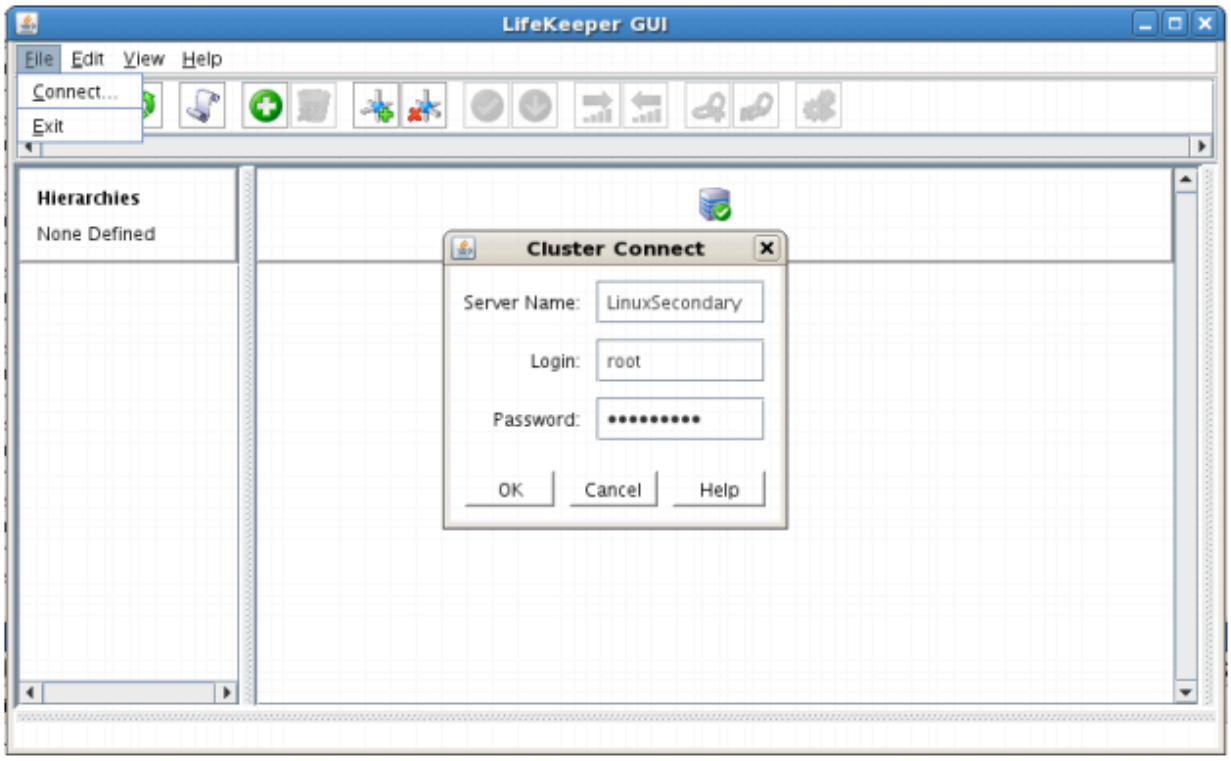
4. Enter the name of the server you wish to connect to (this field will be populated with the name of the server you are on, if you are running the GUI from a server with LifeKeeper installed) along

with your root credentials and click OK.



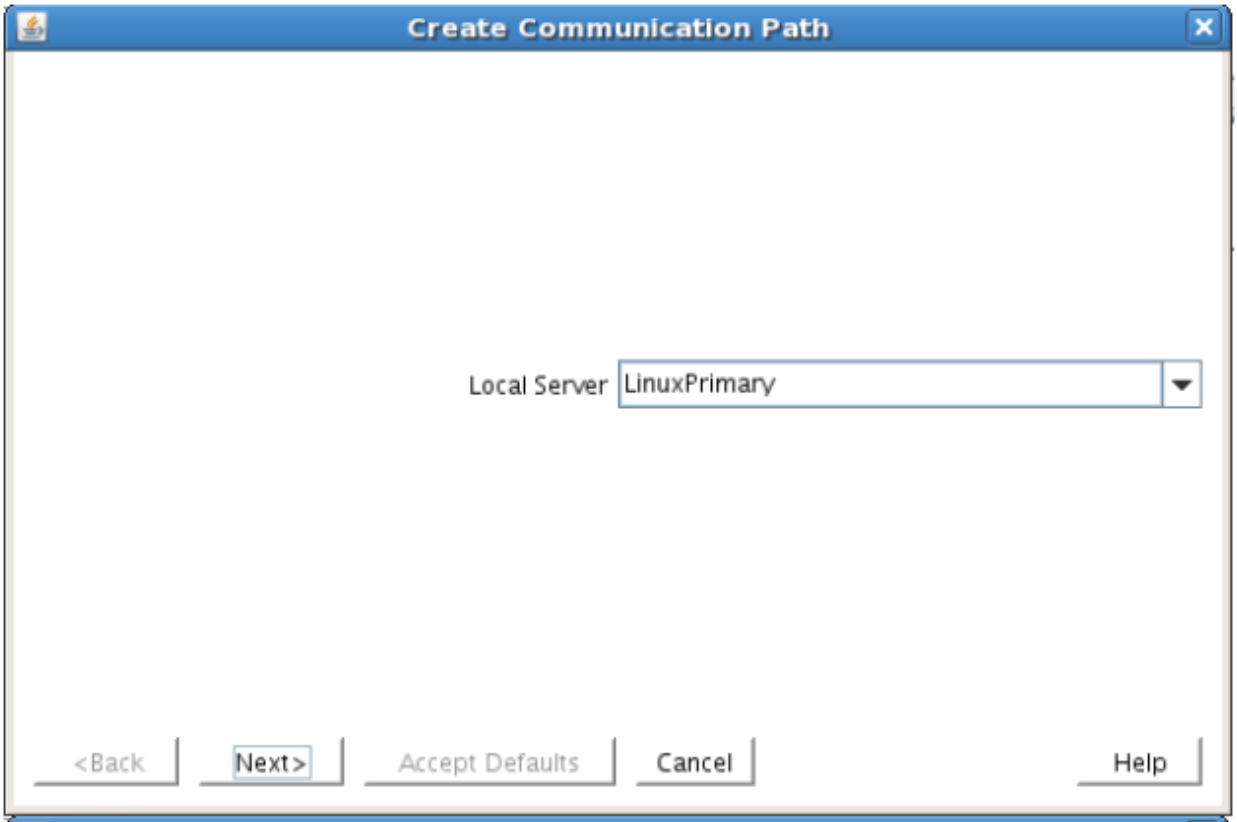
Create Communication (Comm) Paths

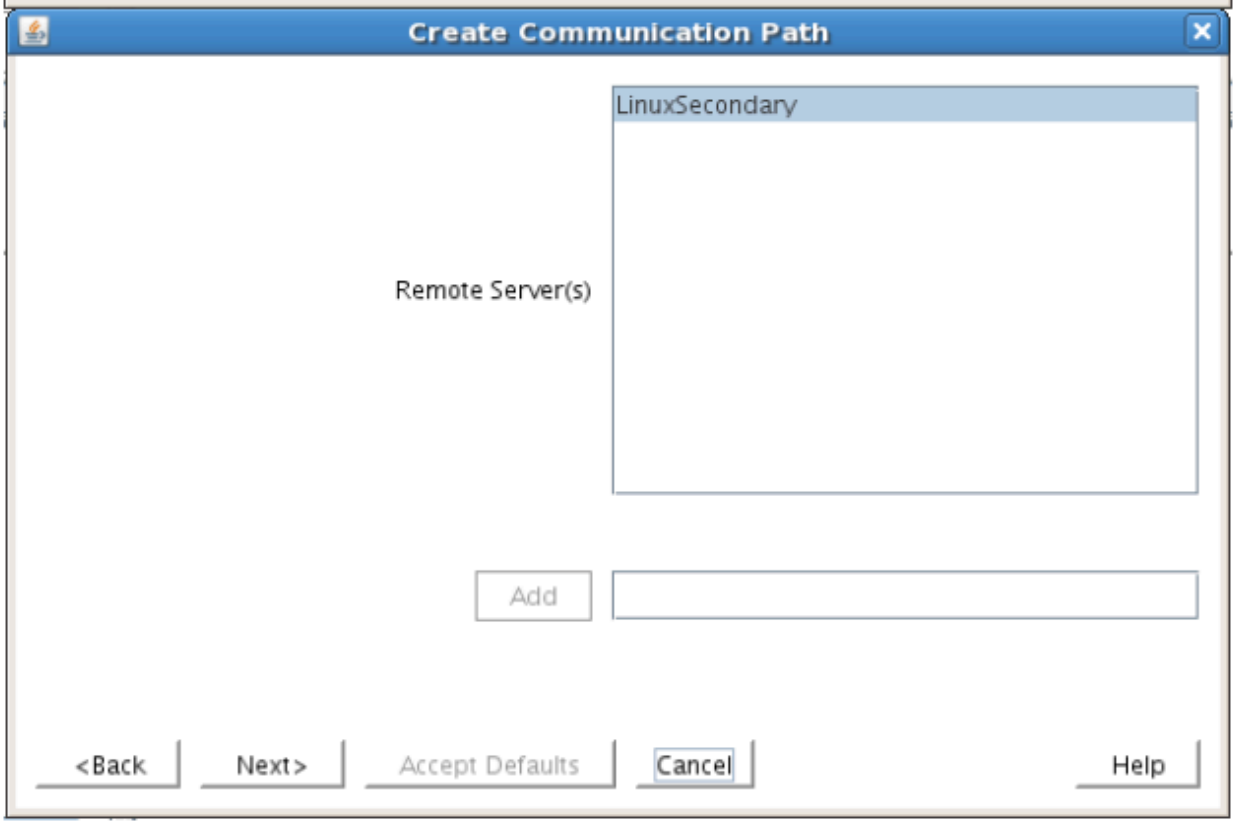
5. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



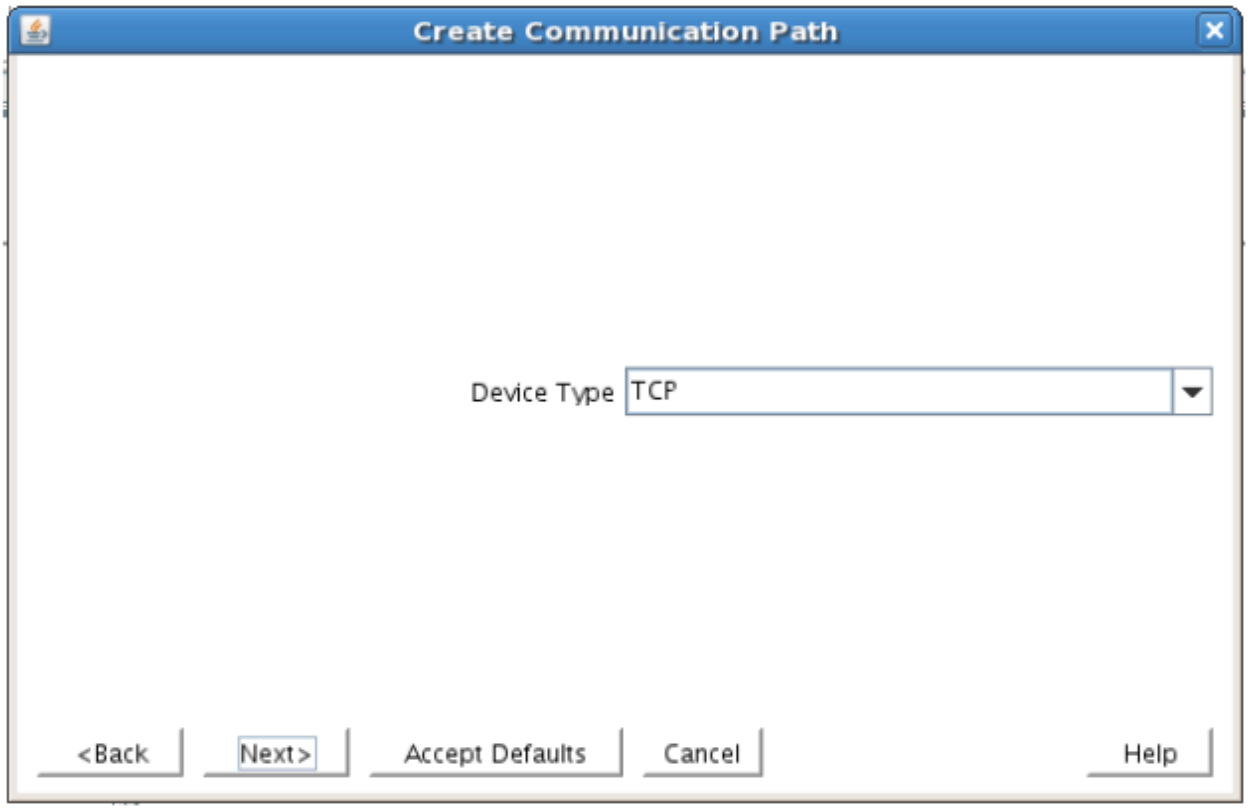
6. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

7. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

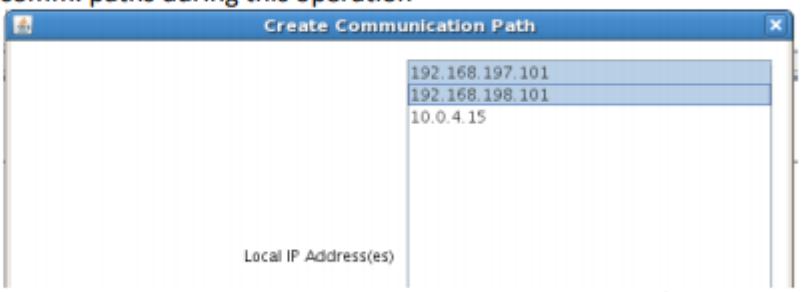


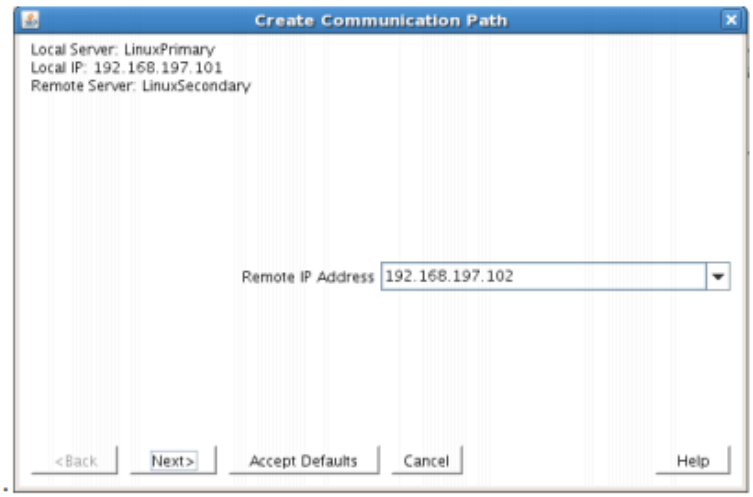


8. Select TCP for Device Type and Click Next.



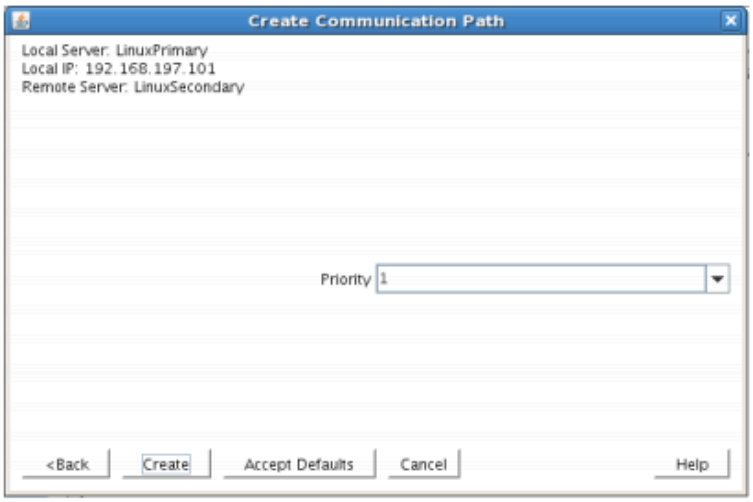
9. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

Field	Tips
For TCP/IP Comm Path...	
Local IP Address	Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation 
Remote IP Address	Choose the IP address to be used by the remote server for this comm path

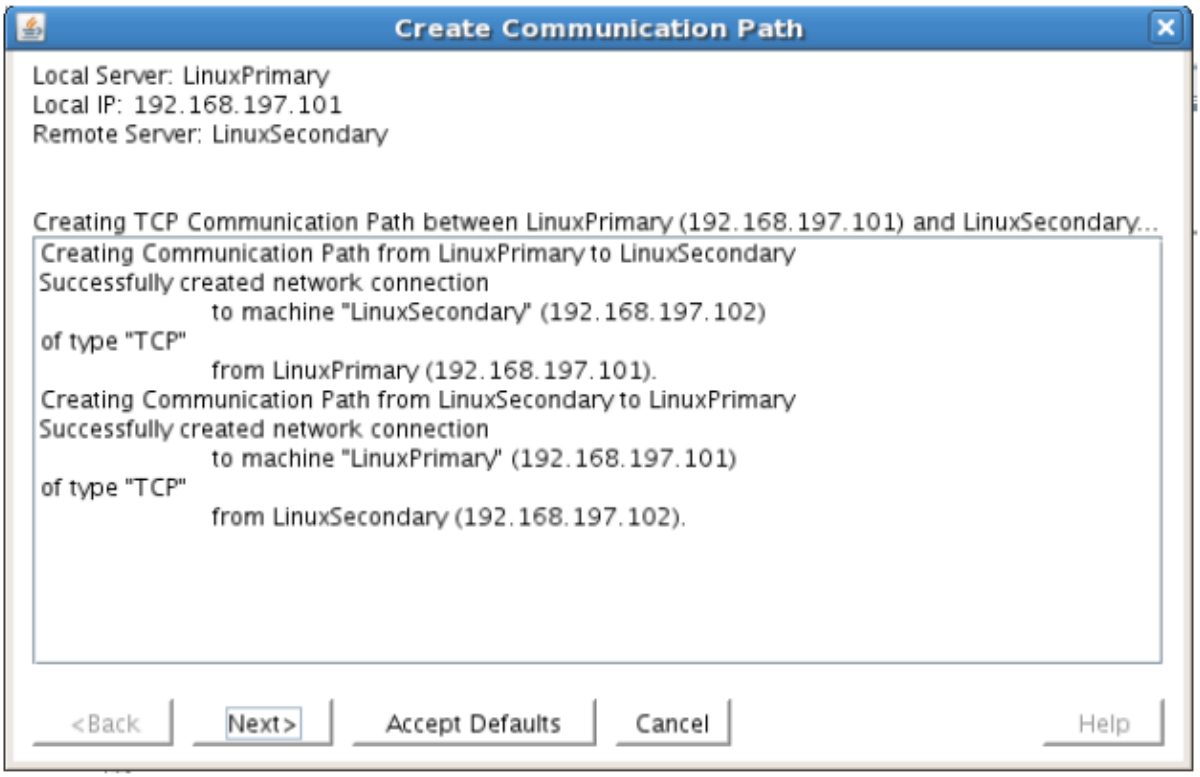


Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority



10. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



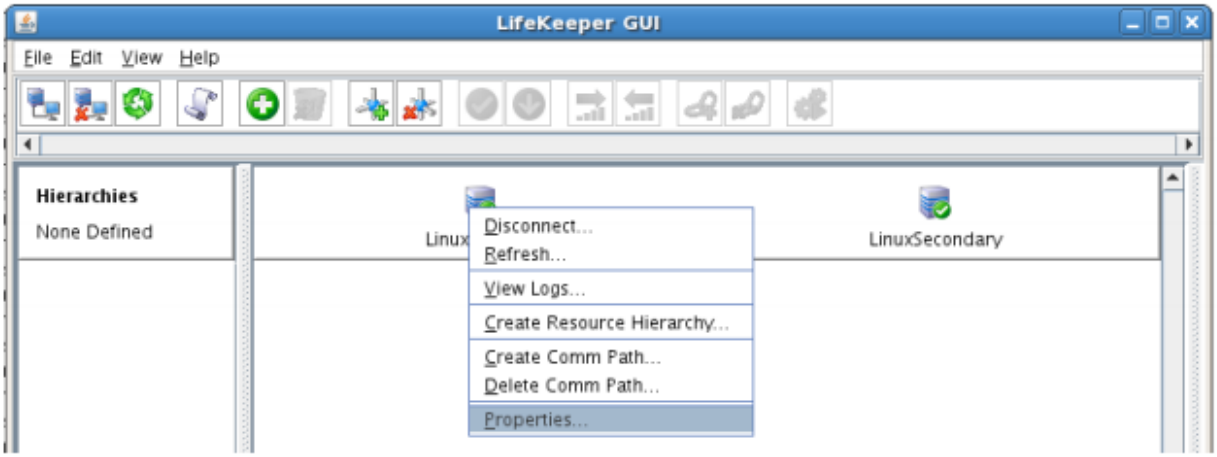
If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

11. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

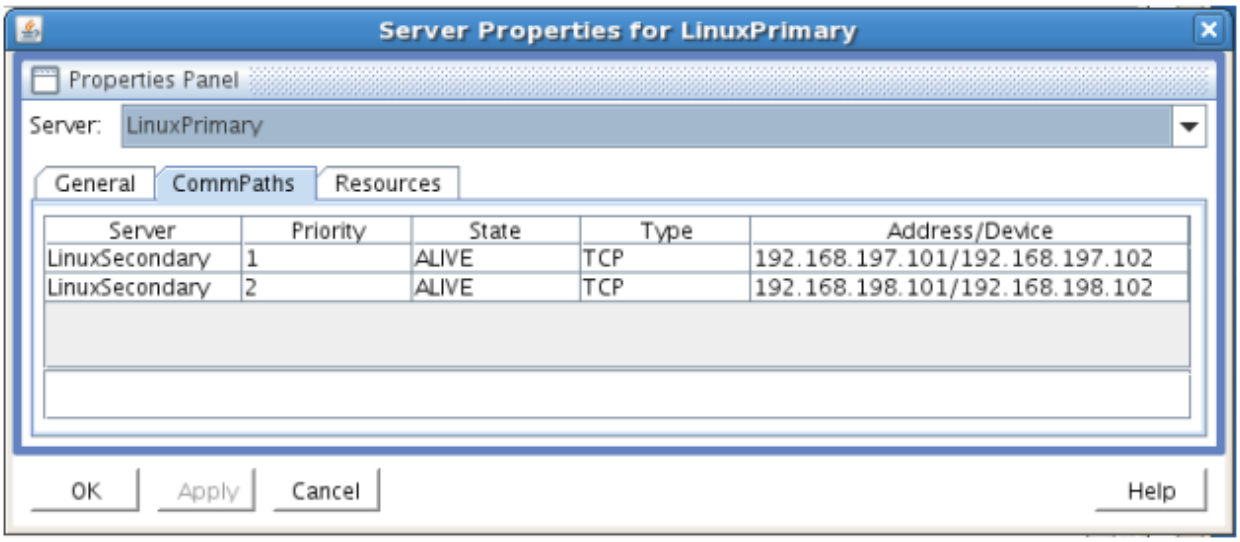
Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.



2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.

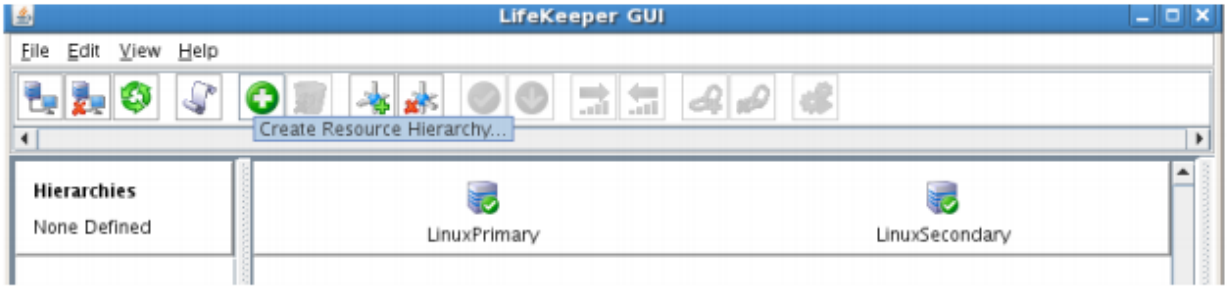


Create the LifeKeeper Hierarchy

Create and Extend an IP Resource

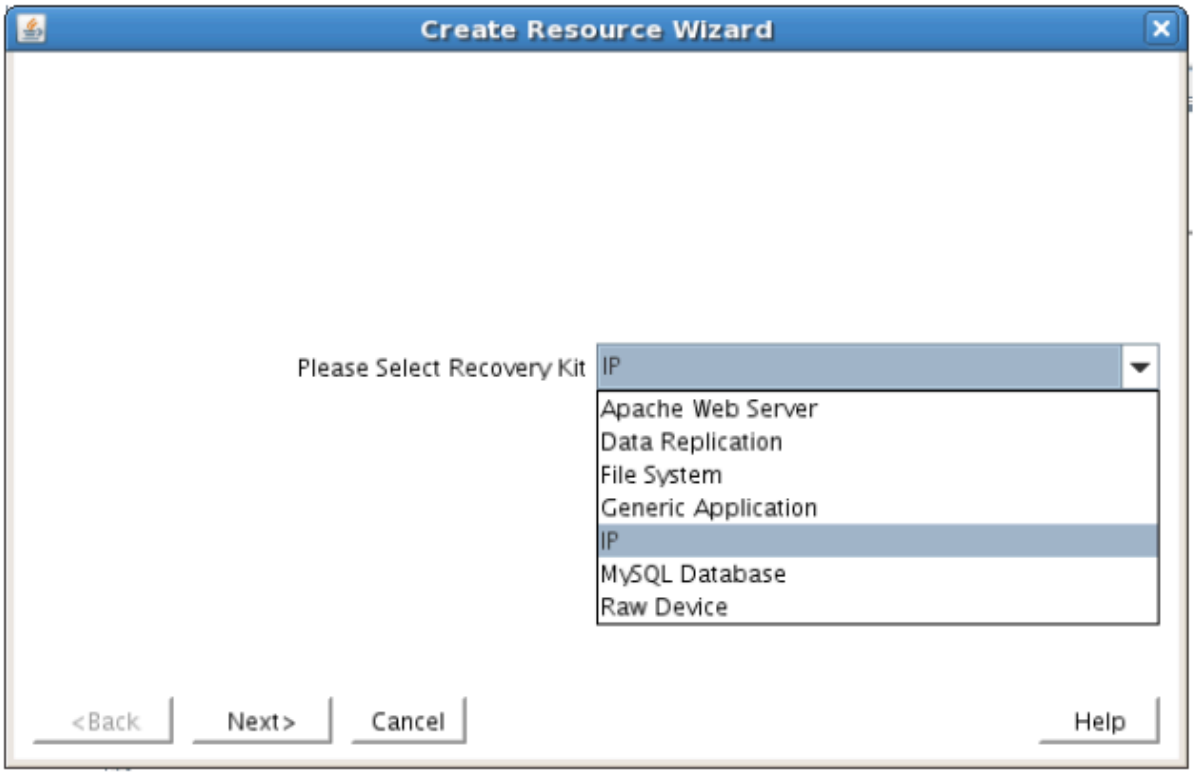
In LifeKeeper, create an IP resource and extend it to the secondary server by completing the following steps. This Virtual IP will have the ability to move between cluster nodes along the application that depends on it.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.



The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select IP Address and click Next.



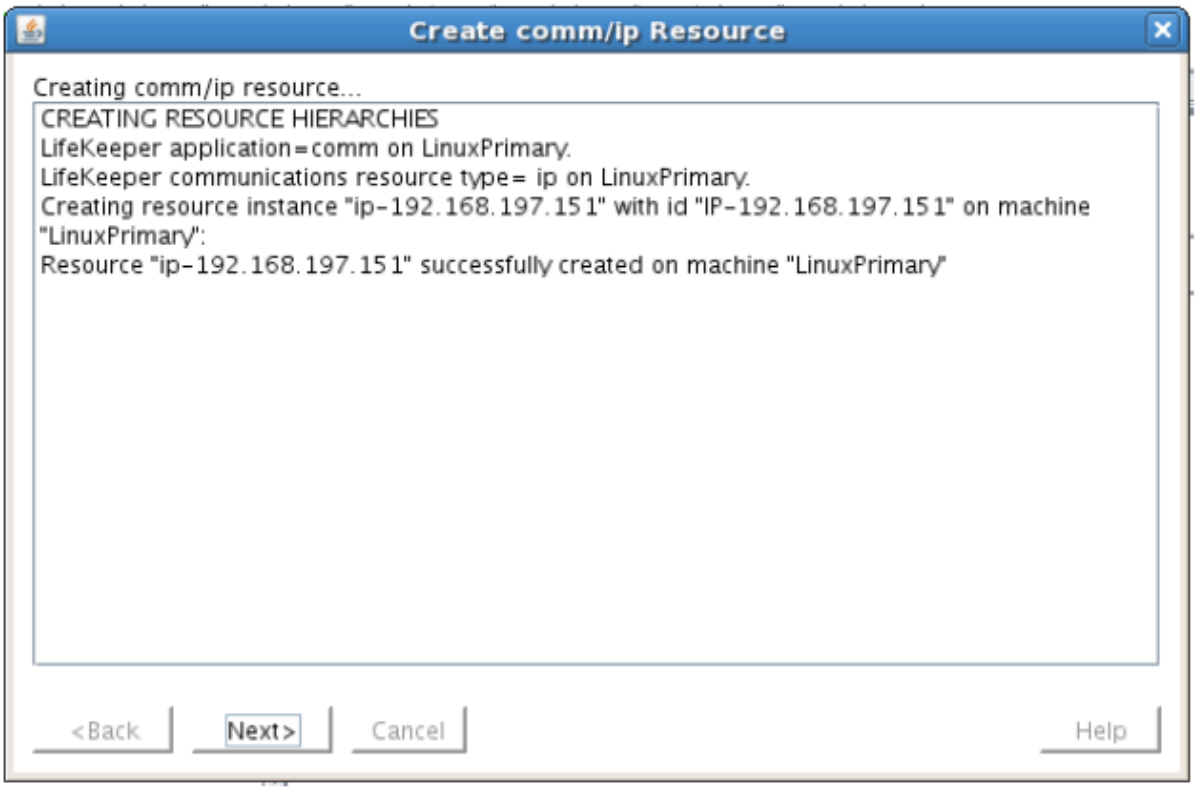
3. Enter the appropriate information for your configuration. The table below contains a list of the fields that display and additional information to assist you as you complete this procedure. Recommended values are also show below. You can also click the Help button for further information. Press Next to continue after entering the required information.

IP Creation Field Definitions

Field	Tips
Resource Type	Select IP Address as the resource type and click Next.
Switchback Type	Select Intelligent and click Next.
Server	Select the Server where the IP resource will be created. Select your Primary server and click Next.
IP Resource	Enter the virtual IP information and click Next Example **192.168.167.151** **Note** This is an IP address that is not currently in use anywhere on your network. This is the address that all clients will use to connect to the protected resources.

Netmask	The IP subnet mask that your TCP/IP resource will use on the target server. Any standard netmask for the class of the specific TCP/IP resource address is valid. In our sample configuration 255.255.255.0 is used for a subnet mask on both networks. **Note:** The subnet mask you choose, combined with the IP address, determines the subnet that will be used by the TCP/IP resource and should be consistent with the network configuration.
Network Connection	This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. Select the correct NIC and click Next.
IP Resource Tag	Accept the default value and click Next. This value only affects how the IP is displayed in the GUI. The IP resource will be created on our Primary server.

4. LifeKeeper will create and validate your resource. After receiving the message that the resource has been created successfully, click Next when the following dialog box appears so that you can complete the process of Extending the IP Resource to our Secondary server, below.



Extending the IP resource will start automatically after you have finished creating an IP address resource if you clicked Next in the dialog box displayed above. You can also start this from an existing IP address resource by right clicking on the active resource and selecting Extend Resource Hierarchy.

Refer to the table below to complete the Extend IP Resource procedure.

Field	Recommended Entries or Notes
Switchback Type	Leave as “intelligent” and click Next
Template Priority	Leave as default (1)
Target Priority	Leave as default (10)
Network Interface	This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. The correct physical NIC should be selected by default. Please verify and then click Next
IP Resource Tag	Leave as default.
Target Restore Mode	Select Enable and click Next.
Target Local Recovery	Select Yes to enable Local Recovery for the SQL resource on the Target server.
Backup Priority	Accept the default value.

5. After receiving the message Hierarchy extend operations completed, click Finish and then click Done

6. Your IP resource (192.168.197.151) is now fully protected and has the ability to “float” between cluster nodes as needed. Looking at the LifeKeeper GUI you will notice that the IP resource is Active on the Primary cluster node and Standby on the Secondary cluster node

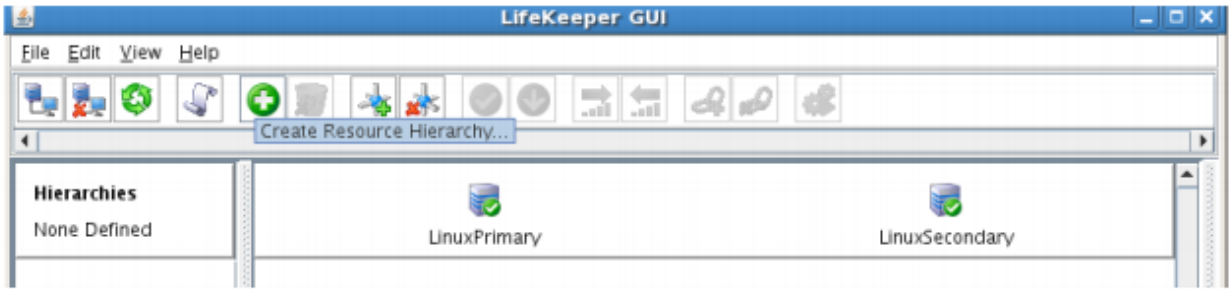
Create a Mirror and Begin Data Replication

In this section we will setup and configure the Data Replication resource, which be used to synchronize our MySQL’s data between cluster nodes. The data we will replicate resides in the /var/lib/mysql partition on our Primary cluster node

Please note:

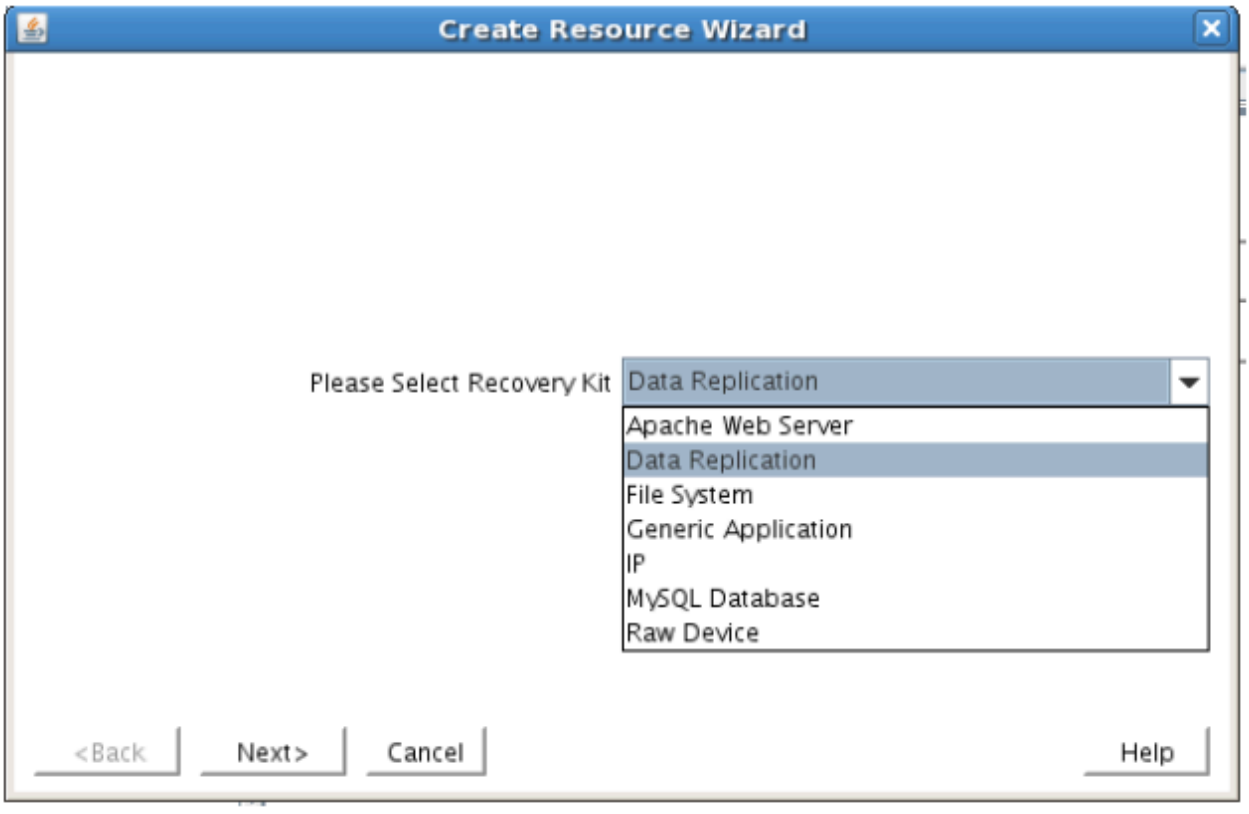
- The source volume to be replicated must be mounted on the Primary server
- The target volume, which will received replicated data, must **NOT** be mounted on the Secondary server.
- The target volume's size must equal to or larger than the size of its source volume.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.



The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select Data Replication and click Next.

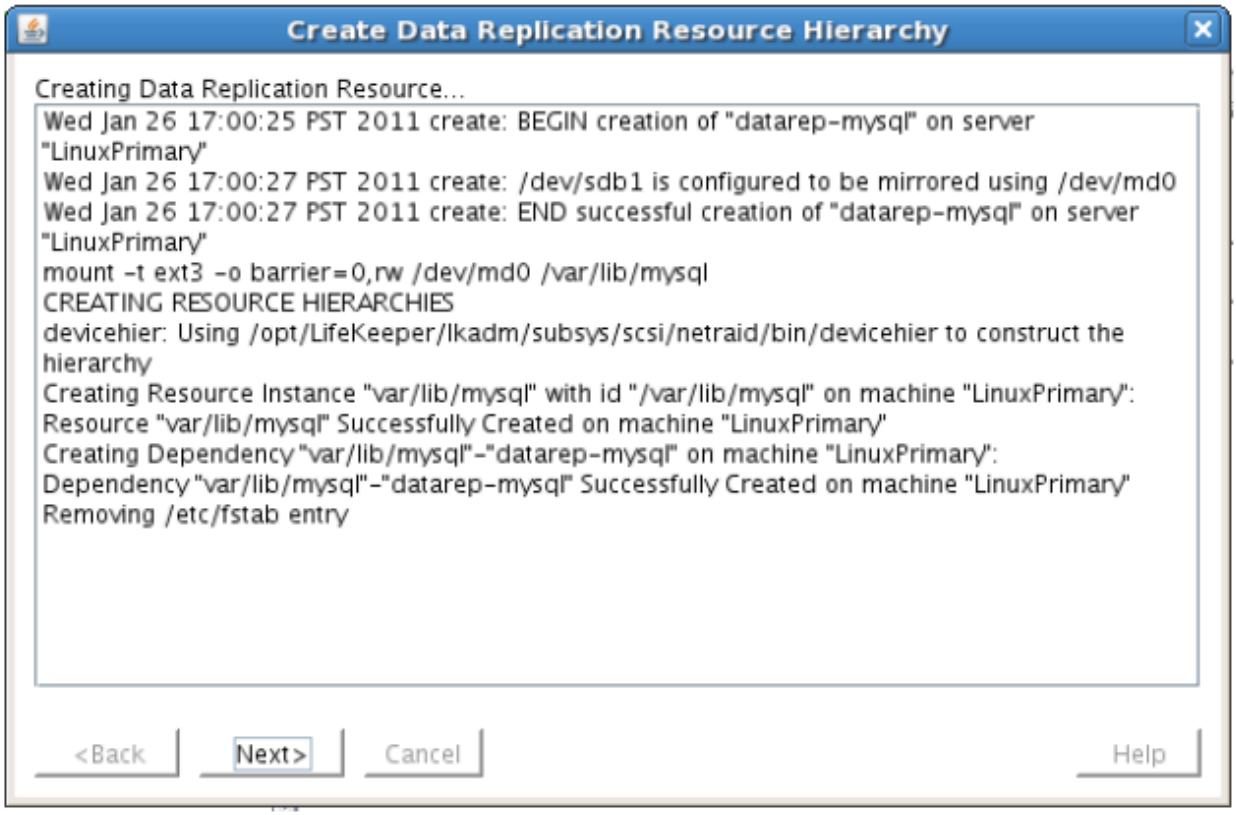


3. Follow the Data Replication wizard, and enter the following values:

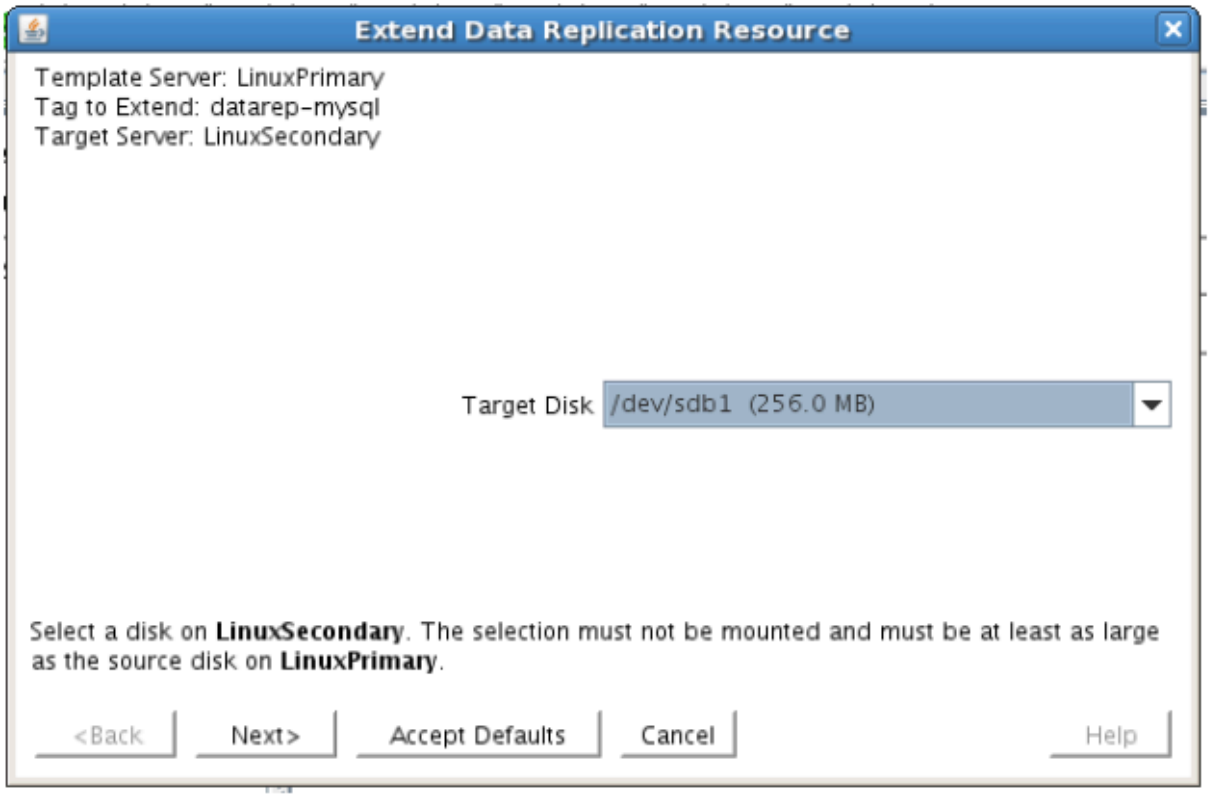
Field	Recommended Entries or Notes
Switchback Type	Intelligent

Server	LinuxPrimary (Primary Cluster Node, i.e. Mirror Source)
Hierarchy Type	Select: “**Replicate Existing Filesystem**”
Existing Mount Point	At this step you will select the mounted partition to replicate. In our example, select “/var/lib/mysql”
Data Replication Resource Tag	Leave as default
File System Resource Tag	Leave as default
Bitmap File	Leave as default (Note: if using high speed SSD storage you will want to create a small partition and use it for bitmap placement, i.e. /bitmaps)
Enable Asynchronous Replication	Leave as default (Yes)

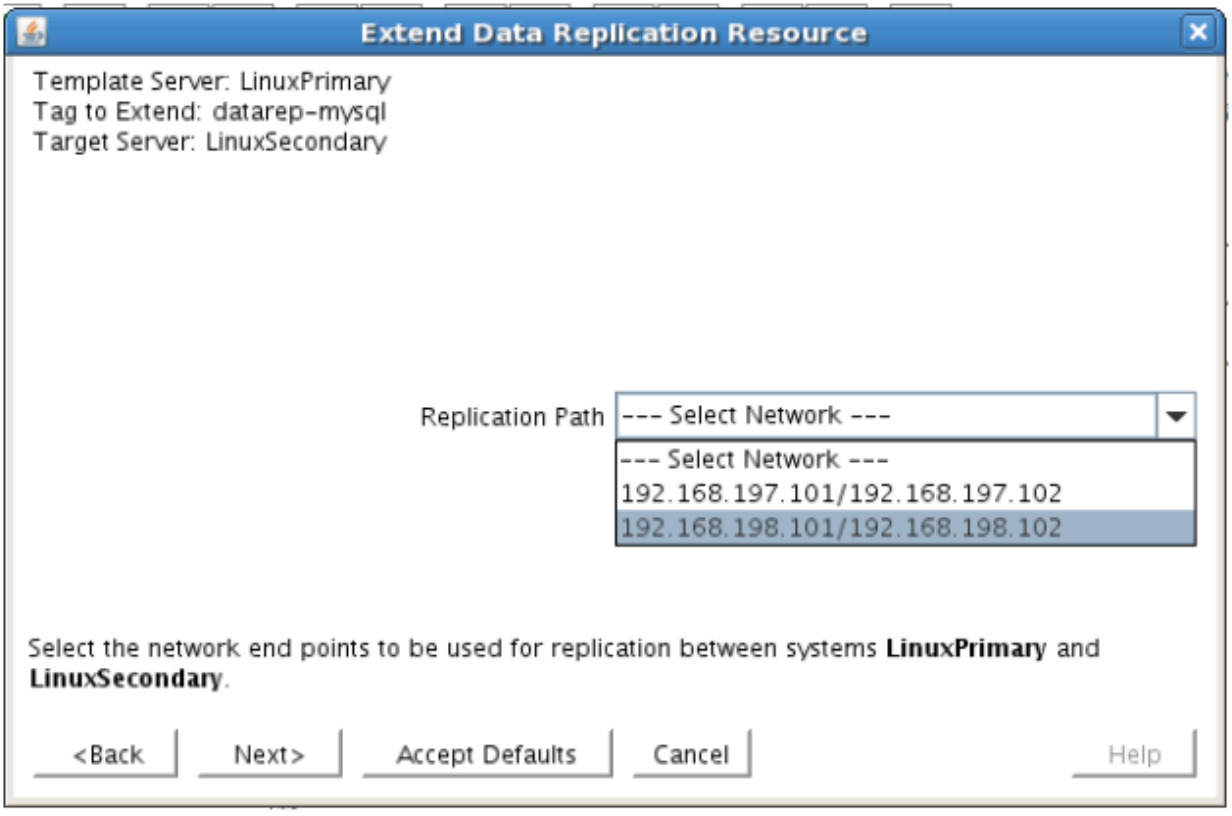
4. Click Next to begin creation of the Data Replication resource hierarchy. Status will be displayed in the GUI as follows:



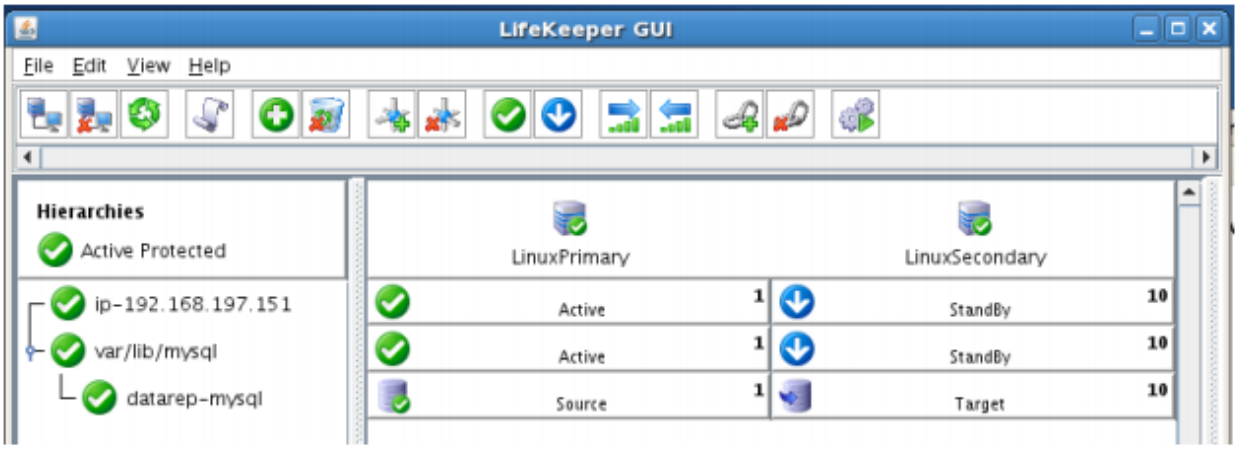
5. Click Next to begin the process to Extend the Data Replication Resource. Select all default settings. When it asks for the target disk, select a free partition on your Target server which is the same size (or greater) than the Source Volume we are replicating. This partition should NOT be mounted on the Target system.



6. Continue through the wizard, and you will be prompted to select the network you would like replication to take place over. In general, it's a best practice to separate your user/application and your replication traffic. In our example setup we will replicate over our backend network, 192.168.198.X



7. Click Next and continue through the wizard. Once completed, your resource hierarchy will look as follows



Create the MySQL Resource Hierarchy

Create a MySQL resource to protect the MySQL database and make it high available between cluster nodes.

> **Important** At this point, MySQL needs to be already running on the Primary Server. It should NOT be running on the Secondary Server. See section titled “Install, Configure, and Start MySQL” above to review the process to configure and start MySQL as needed.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select **MySQL Database** and click **Next**.
3. Proceed Through the Resource Creation wizard, providing the following values

Field	Recommended Entries or Notes
Switchback Type	Intelligent
Server	LinuxPrimary (Primary Cluster Node)
Location of my.cnf	Enter “/var/lib/mysql”. Note that earlier in the MySQL
executables	configuration process we created a my.cnf file in this directory
Location of MySQL	Leave as default (/usr/bin) since we are using a standard MySQL install/configuration in this example
Database tag	Leave as default

4. Select Create to define the MySQL resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the MySQL resource is now protected on both cluster nodes. Click Finish to exit the

Extend wizard.

8. Note: LifeKeeper will automatically identify that the MySQL resource has a dependency on the FileSystem (Data Replication) resource (/var/lib/mysql). The Filesystem Resource will appear underneath the MySQL resource in the GUI
9. Your resource hierarchy should look as follows:

Create the MySQL IP Address Dependency

In this step will define an additional dependency: that MySQL depends on a Virtual IP (192.168.197.151) so that the IP address follows the MySQL database as it moves.

1. From the LifeKeeper GUI toolbar, right-click on the “mysql” resource
2. Select “Create Dependency” from the right-click context menu
3. In the Child Resource Tag dropdown menu, select “ip-192.168.197.151”
4. Click Next
5. Click Create Dependency
6. Click Done
7. The Virtual IP address resource (192.168.197.151) will now appear underneath the MySQL resource in the LifeKeeper user interface. This ensures that resources move together, and are started/stopped in the proper order.
8. Your resource hierarchy should look as follows

At this point in the Evaluation, we have fully protected MySQL, and its dependent resources: IP addresses, and replicated storage.

11.6.8. Test Your Environment

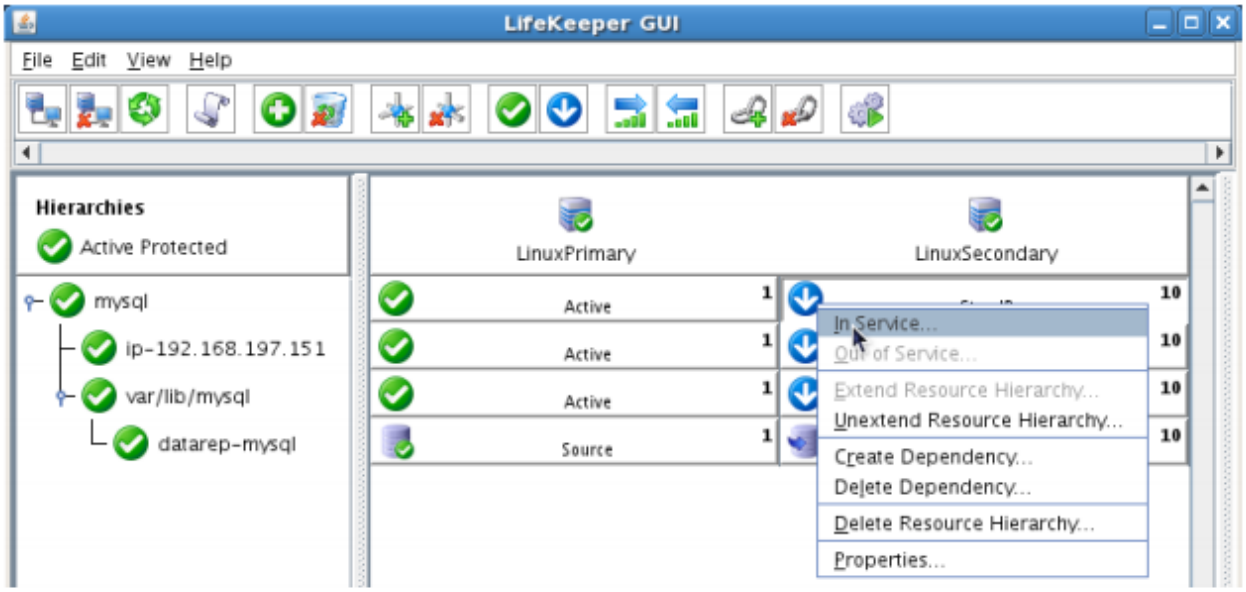
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

> **Note:** For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

Manual Switchover of the MySQL Hierarchy to Secondary Server

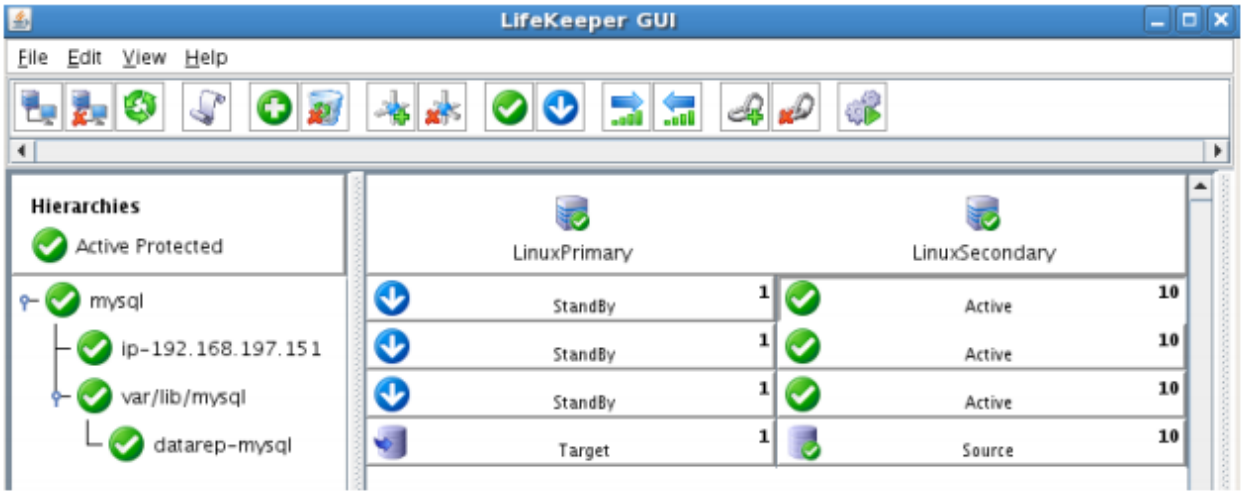
Procedure:

- From the LifeKeeper GUI, right click on the MySQL resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



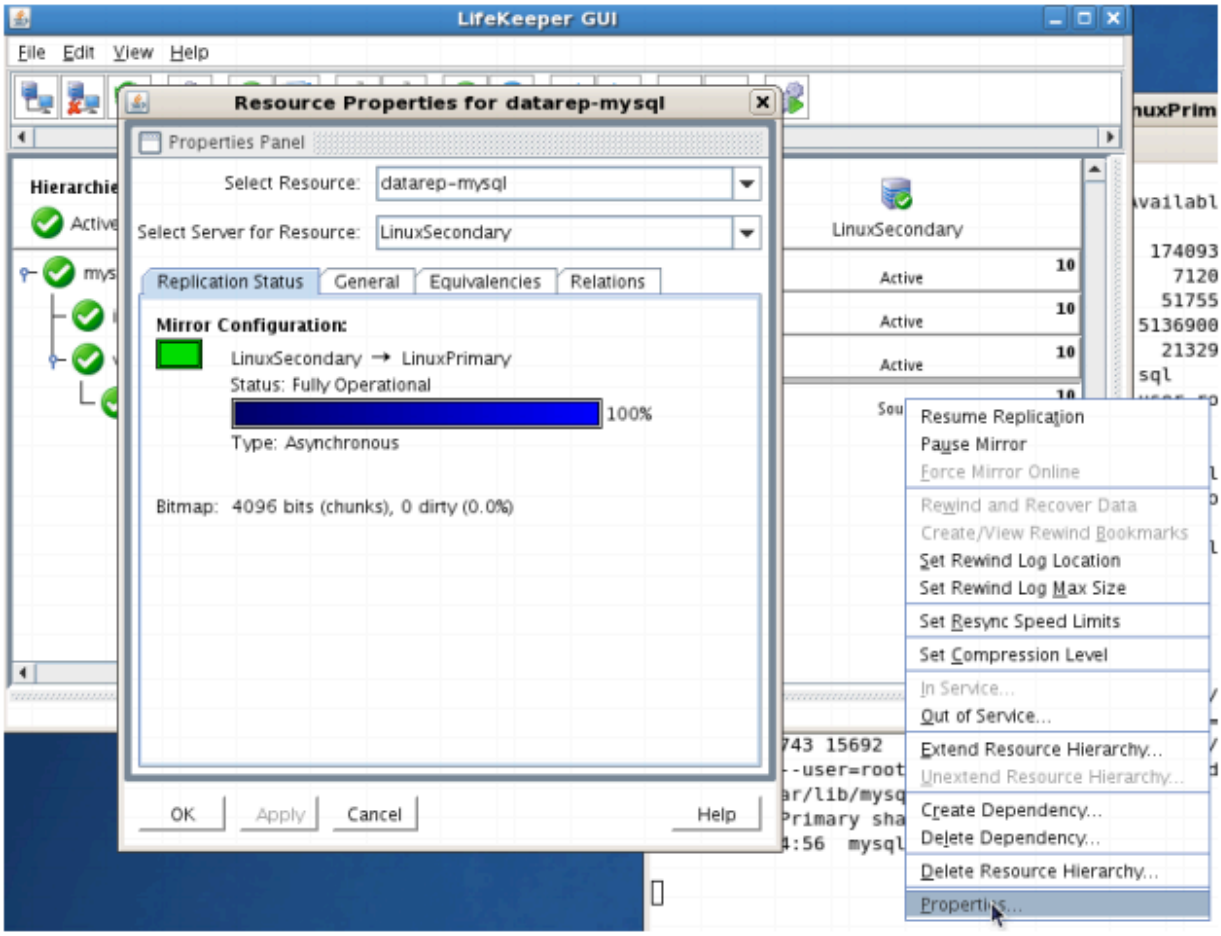
Expected Result:

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXSECONDARY.
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXSECONDARY -> LINUXPRIMARY
- At this point, all resources are now active on LINUXSECONDARY.



Tests/Verification:

- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXSECONDARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties



- Run “ifconfig –a” on LINUXSECONDARY to validate that the IP Address 192.168.197.151 is active on LINUXSECONDARY
- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device

(example: /dev/md0”) on LINUXSECONDARY
- Verify the MySQL services are running on LINUXSECONDARY by running “ps –ef | grep –i mysql”
- On LINUXSECONDARY run the following command to verify client connectivity to the MySQL database:
    -
        - # mysql –S /var/lib/mysql/mysql.sock –u root –p
        - (enter password “SteelEye”)

Manual Switchover of the MySQL Hierarchy back to Primary Server

Procedure:

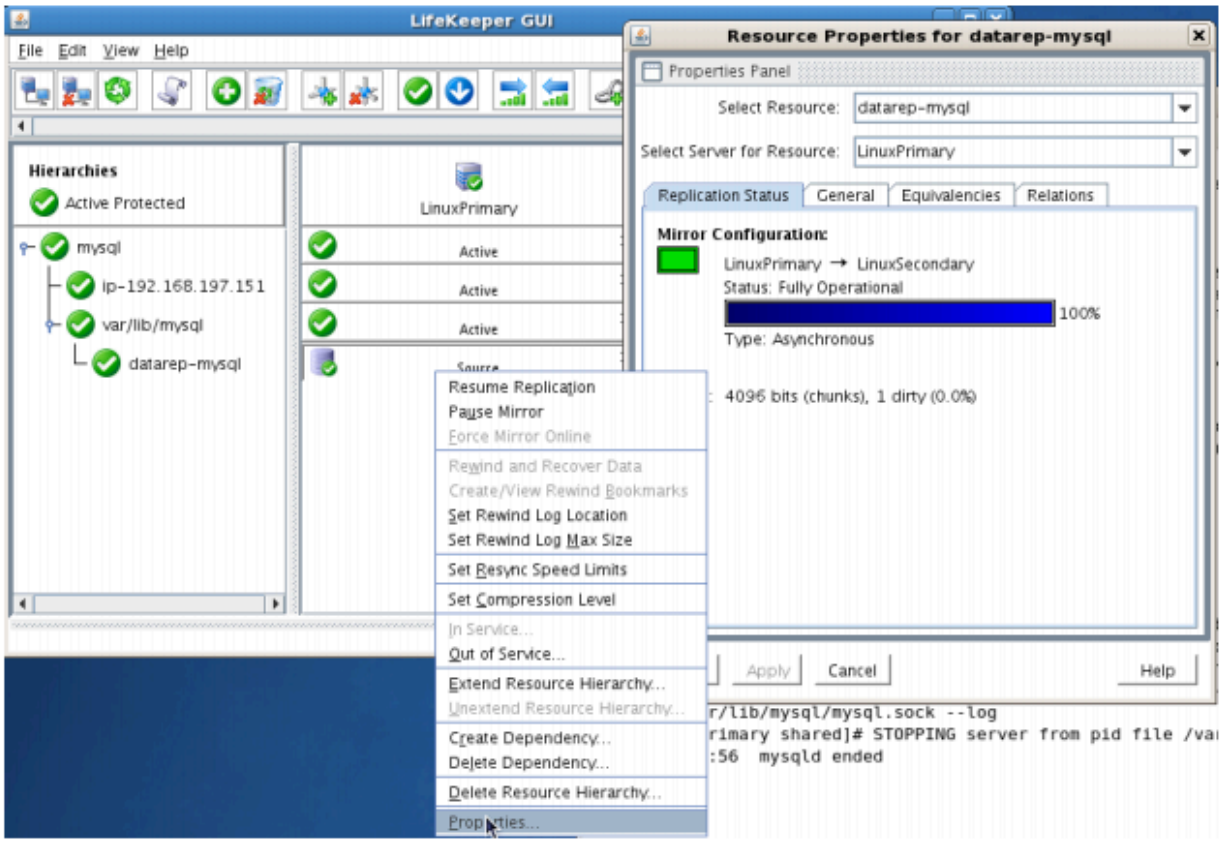
- From the LifeKeeper GUI, right click on the MySQL resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

Expected Result:

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXPRIMARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXPRIMARY -> LINUXSECONDARY

Tests/Verification:

- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXPRIMARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties



- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.151 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY
- Verify the MySQL services are running on LINUXPRIMARY by running “ps –ef | grep –i mysql”
- On LINUXPRIMARY run the following command to verify client connectivity to the MySQL database:
    -
        - # mysql –S /var/lib/mysql/mysql.sock –u root –p
        - (enter password “SteelEye”)

Simulate a network failure on the Primary Server by failing the IP resource

> **IMPORTANT:** Only perform this test if you have more than one communications path configured.

If you perform this test and have only one communications path configured, your system will enter a split-brain scenario as described in the LifeKeeper Administration Guide found here. Refer to this document for more information or contact SIOS presales technical support for assistance in resolving this condition.

Procedure:

- On LINUXPRIMARY, pull the network cable attached to the NIC that the virtual IP address is configured on

Expected Result:

- The IP Resource should fail first.
- The entire hierarchy should failover to LINUXSECONDARY

Tests/Verification:

- Check the LifeKeeper Log to verify the IP resource failed – "/opt/LifeKeeper/bin/lk_log log"
- Using the LifeKeeper GUI, verify the MySQL and Apache resource hierarchies fail over successfully to LINUXSECONDARY

Hard failover of the resource from the Secondary Server back to the Primary Server

Procedure:

- Pull the power cord on LINUXSECONDARY, as this is the server with all resources currently In Service.

Expected Result:

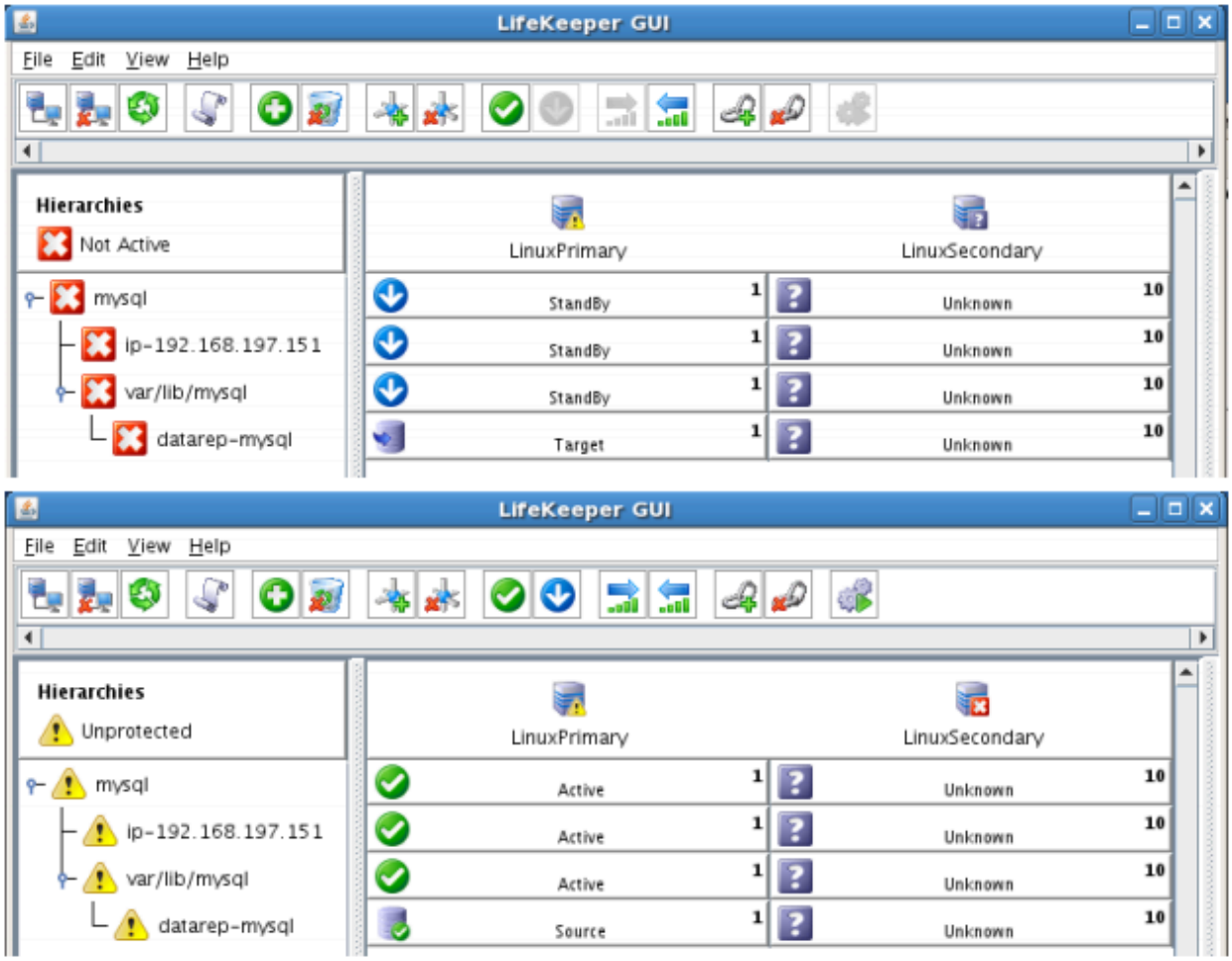
- After failure has been detected, beginning with the dependent resources (IP and Volume), all resources will be brought in service on LINUXPRIMARY.

Tests/Verification:

- Using the LifeKeeper GUI, verify the mirror has reversed and is in a Resync Pending state waiting

for LINUXSECONDARY to come back on line.

- Verify the Apache and MySQL Server services are running on LINUXPRIMARY.
- Verify that the client can still connect to the Webserver and database running on LINUXPRIMARY.
- Verify you can write data to the replicated volume, /var/lib/mysql on LINUXPRIMARY.



Bring Failed Server back on line

Procedure:

- Plug the power cord back into LINUXSECONDARY and boot it up.

Expected Result:

- Using the LifeKeeper GUI, verify that LINUXSECONDARY is coming back up and has become the Standby Server.

Tests/Verification:

- Verify the mirror performs a quick partial resync and moves to the Mirroring state
- Verify the Apache and MySQL Hierarchy are in service on LINUXPRIMARY and standby on LINUXSECONDARY.

Verify Local Recovery of MySQL Server

Procedure:

- Kill the MySQL processes via the command line:
- # ps –ef | grep sql
- # killall mysqld mysqld_safe
- run “ps –ef | grep sql once again to verify that the processes no longer exist

Expected Result: (Assumes Local Recovery for MySQL resource is set to YES)

- The MySQL Server service should stop.
- The MySQL quickcheck process will automatically restart the MySQL Server Service when it runs periodically.
- No failure of MySQL should occur.

Tests/Verification:

- Execute “ps –ef | grep sql” once again to verify that the mysql processes have been restored locally on LINUXPRIMARY.

- Verify connectivity to the MySQL database by running:
  ↑
- # mysql –S /var/lib/mysql/mysql.sock –u root –p
- # (Enter password “SteelEye”)

- If you inspect the LifeKeeper logs, you will see information indicating that LifeKeeper dectected the failure of the MySQL service and recovered it locally. Run /opt/LifeKeeper/bin/lk_log log for more information.

11.7. PostgreSQL Cluster with Shared Storage (ISCSI)

Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product, to make PostgreSQL highly available. If PostgreSQL is not already installed, please allocate some time to install it on your servers. Once this task has been completed, you may install and configure SIOS Protection Suite for Linux.

There are five phases in this process:

- Phase 1 – Prepare to Install
- Phase 2 – Configure Storage
- Phase 3 – Install and Configure PostgreSQL
- Phase 4 – Install SIOS Protection Suite for Linux
- Phase 5 – Configure your LifeKeeper Cluster
- Phase 6 – Test Your Environment

11.7.1. Terms to Know – PostgreSQL

The following terms are used throughout this document and, while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

Network Communication Terms

Crossover cable – A cable used to directly connect computing devices together, instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

Types of LifeKeeper Servers

Server – A computer system dedicated to running software application programs.

Active Server – This is the server where the resource hierarchy is currently running (IN SERVICE).

Standby Server – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

Primary Server – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

Secondary Server – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

Source Server – In a LifeKeeper cluster, using data replication, this is the Active Server. It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

Target Server – In a LifeKeeper cluster, using data replication, this is the Standby Server. The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

SIOS Data Replication Terms

Replication – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

Synchronous – A replication scheme in which the data is confirmed written and valid on the target before the write operation occurs on the source disk through a series of information exchanges. Synchronous mirrors should only be implemented on high speed (100Mbps+) networks due to the network overhead involved.

Asynchronous – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there and can be written on them.

Rate of Change – A measure of the amount of data which is changing over a set period of time.

Compression – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

Throttling – An optionally implemented mechanism to limit the bandwidth used for replication.

LifeKeeper Product Terms

Communications Path – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

Heartbeat – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

Split Brain – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

Failover – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

Switchover – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

Switchback – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

Resource – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

Extend a Resource – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously.

Resource Hierarchy – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

Shared Storage – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally called I/O fencing.

Data Replication (Disk Mirroring) – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

Source – The partition on the source server used for replication. The “gold” copy of the data.

Target – The partition on the target server used for replication.

Switchable IP Address – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

11.7.2. The Evaluation Process – PostgreSQL

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux in a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

Important Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

11.7.3. Prepare to Install – PostgreSQL

Hardware Requirements

Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional disk partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system (/) and boot (/boot) partitions are not eligible for replication.

Note: You may use more than one partition for replicated data, allowing for separation of multiple database files and/or log directories for performance reasons.

Client

This system is not required but is recommended for testing the cluster environment.

- A standard linux terminal running the MySQL client can be used to test the configuration.

Software Requirements

Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
  - º RedHat Enterprise Linux 5 (5.4+ recommended) or 6.x
    º CentOS Linux 5 (5.4+ recommended) or 6.x
    º Oracle Enterprise Linux 5 (5.4+ recommended), 6.3, 6.4
- RedHat Compatibility Kernel Only
- º SuSE Linux Enterprise Server 10 or 11 (11 recommended)
  º See

Linux Release Notes for a full list of supported Operating SystemsCurrent patches / security updates are recommended. Satisfied dependencies; especially if the Linux installation package selection was base/ minimal you will need to refer to the dependencies documentation at Linux Dependencies Its recommended that IPtables is disabled

- º # /etc/init.d/iptables off
- º # chkconfig iptables off
- º See

here for information regarding the ports SIOS Protection Suite for Linux uses. Disable SELinux :

- º Edit /etc/selinux/config
- º Set

SELINUX=disabled (note: permissive mode is also acceptable) Check the configuration of your /etc/hosts file

- º localhost.localdomain and localhost are the only entries that can be on
  127.0.0.1
- º Create a separate entry for your hostname with a static address

GUI Authentication with PAM

- º

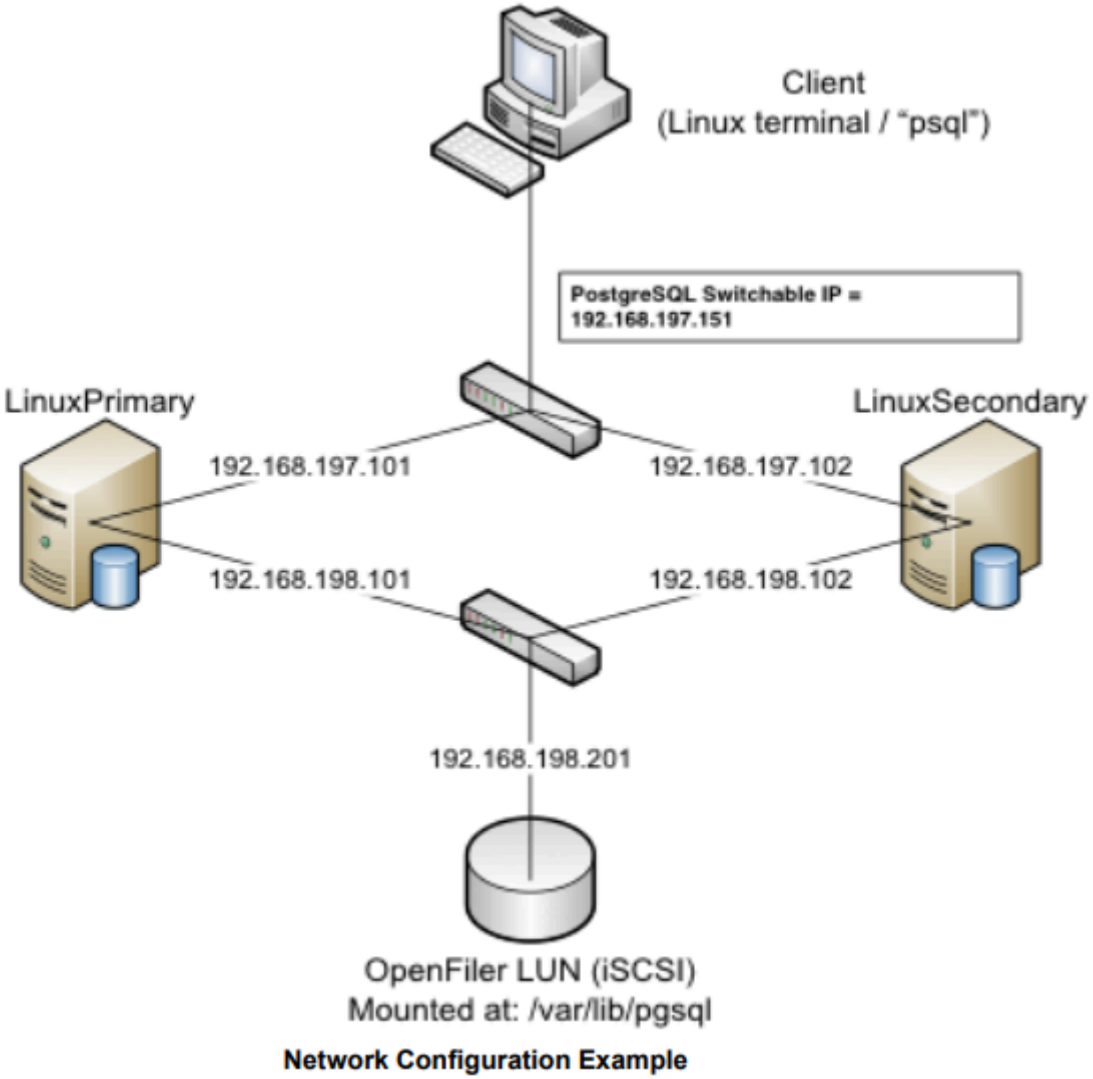
SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).
º Users are identified and authenticated against the system's PAM configuration. Privilege levels are determined from group membership as provided through PAM.
º In order to access the GUI, a user must be a member in one of the three LifeKeeper groups: **lkadmin**, **lkoper** or **lkguest**.
º See Configuring GUI Users for more information.

Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi- in production environments to avoid a single point of failure.

In this example evaluation scenario we will be leveraging Shared (iSCSI) Storage with our PostgreSQL database. OpenFiler is a storage appliance server that will serve an iSCSI target to LinuxPrimary and LinuxSecondary.



Network Configuration Example

Primary Server and Secondary Servers

- Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

  Example:

  192.168.197.101 LinuxPrimary

  192.168.197.102 LinuxSecondary

- See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.
- Public Network connection(s) configured with:
  - º Static IP address
  - º Correct subnet mask

- ◦
  - º Correct gateway address
- ◦
  - º Correct

DNS server address(es)

Private Network connection(s) configured with:

- º Static IP address (on a different subnet from the public network)
- º Correct network mask
- º No gateway IP address
- º No

DNS server addresses

Client

Must be able to communicate on the same subnet/network as the servers Public interface addresses. In our example, this is the 192.168.197.0/24 network.

11.7.4. Configure Storage – PostgreSQL

Before you Begin

Ensure the following:

- If planning to use replicated storage, have an extra volume/partition on both servers that can be used for data replication. A target volume's size must equal to or larger than the size of its source volume.
- If planning to use shared storage, as in this example, ensure the Shared storage is configured and accessible to your cluster nodes. This can either be Fiber Channel SAN, iSCSI, NAS, etc. In this example we will review configuration of an iSCSI target for use as our PostgreSQL database storage repository.

Configure iSCSI initiator, discover and login to iSCSI target

This Evaluation guide will not cover how to setup an iSCSI Target Server. It is assumed that the shared storage already exists in your environment. If you don't have shared storage and wish to configure it, a simple solution is to use OpenFiler (http://www.openfiler.com/), an Open Source storage management appliance, which can be run on physical hardware or as a virtual machine.

On both Primary and Secondary servers, perform the following functions:

1. If not already installed, ensure that the **iscsi-initiator-utils** rpm package is installed:

   ```
   # yum install iscsi-initiator-utils
   ```

2. Start the iscsid service and enable it to automatically start when the system boots

   ```
   # service iscsid start
   ```

   ```
   # chkconfig iscsid on
   ```

3. Configure the iscsi service to automatically start, which logs into iSCSI targets needed at system start up.

   ```
   # chkconfig iscsi on
   ```

4. Use the iscsiadm command to discover all available targets on the network storage server (OpenFiler)

   ```
   # iscsiadm -m discovery -t sendtargets -p <name or IP of iSCSI server>
   ```

Example
[root@LinuxPrimary init.d]# iscsiadm -m discovery -t sendtargets -p 192.168.198.201

iqn.2006-01.com.openfiler:tsn.postgres

5. Manually Login to the iSCSI Target

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.postgres –p 192.168.198.201 – login

6. Configure Automatic Login

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.postgres –p 192.168.198.201 —op update -n node.startup -v automatic

7. Use the “gdisk” command to format your iSCSI LUN, if needed

   # gdisk /dev/sdc

8. Create a filesystem on your new iSCSI LUN Partition, sdc1

   # mkfs.ext3 /dev/sdc1

9. Mount your iSCSI LUN at /var/lib/pgsql (assuming a default postgres configuration). If data already exists in this directory, make sure to move it into the shared iSCSI LUN

   # mount mount /dev/sdc1 /var/lib/pgsql

10. At this point you now have an iSCSI shared LUN, /dev/sdc1, mounted at /var/lib/pgsql. Our disk layout now look as follows (example):

Example

```
[root@LinuxPrimary pgsql]# df
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sda2 25967432 3683016 1976400 66% /
/dev/sda1 101086 24659 71208 26% /boot
tmpfs 517552 0 517552 0% /dev/shm
/dev/sdc1 966644 38944 878596 5% /var/lib/pgsql
```

11.7.5. Install, Configure, and Start PostgreSQL

Primary Server

On your Primary server, perform the following actions:

1. Install both the “postgresql-server” and “postgresql” rpm packages if they do not exist on your system. Apply any required dependencies as well

   # yum install postgresql postgresql-server

2. Verify that your Shared iSCSI LUN is still mounted at /var/lib/pgsql via the “df” command

3. If this is a fresh PostgreSQL install, initialize a sample PostgreSQL database:

   # su – postgres

   # initdb —pgdata=/var/lib/pgsql/data

4. Ensure that all files in your PostgreSQL data directory (/var/lib/pgsql) have correct permissions and ownership

   # chown –R postgres:postgres /var/lib/pgsql

   # chmod 755 /var/lib/pgsql

5. Finally, manually start the PostgreSQL daemon from the command line. Note: **Do Not** start it via the “service” command, or the /etc/init.d/ scripts

   # su – postgres

   # pg_ctl start -D /var/lib/pgsql/data -l /var/lib/pgsql/pgstartup.log -o “-p 5432” -w

6. Verify PostgreSQL is running by connecting with the psql client (ensure you are still running as the “postgres” linux user):

   -bash-3.2$ psql

   Welcome to psql 8.1.22, the PostgreSQL interactive terminal.

   Type: \copyright for distribution terms

   \h for help with SQL commands

```
\? for help with psql commands

\g or terminate with semicolon to execute query

\q to quit

postgres=# \q

-bash-3.2$
```

Secondary Server

On your Secondary Server:

1. install both the “postgresql” and “postgresql-server” rpm packages if they do not exist on your system. Apply any required dependencies as well

   ```
   # yum install postgresql postgresql-server
   ```

2. Ensure that the PostgreSQL data directory (/var/lib/pgsql) has correct permissions and ownership

   ```
   # chown –R postgres:postgres /var/lib/pgsql

   # chmod 755 /var/lib/pgsql
   ```

3. There is no need to perform any of the additional steps taken on the Primary Server

11.7.6. Install SIOS Protection Suite for Linux – PostgreSQL

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

SPS for Linux Component	Install Location
LifeKeeper Software	/opt/LifeKeeper
LifeKeeper Config File	/etc/default/LifeKeeper

Perform the following actions on **both** Primary and Secondary server.

Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.

2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After successful download you will have downloaded contents similar to the follow directory listing:

      [root@LinuxPrimary ~]# ls -l <directory> total 63680

      -rw-r—r— 1 root root 23163 May 30 14:03 EULA.pdf

      -rw-r—r— 1 root root 536 May 30 14:03 readme.txt

      -rw-r—r— 1 root root 65179648 May 30 14:03 sps.img

3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

      # mount -o loop sps.img /mnt

```
# cd /mnt

# ./setup
```

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

```
# cd /root

# umount /mnt
```

Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

```
# /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic
```

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

```
# /opt/LifeKeeper/bin/lklicmgr

License File: 20101230.lic
```

Product	Type	Expiry
LifeKeeper for Linux	Eval	27 Mar 2013 (87 days)

SIOS Data Replication ARK	Eval	27 Mar 2013 (87 days)
PostgreSQL Recovery Kit	Eval	27 Mar 2013 (87 days)

Start the SIOS Protection Suite for Linux

1. Start:

```
# /opt/LifeKeeper/bin/lkstart
```

11.7.7. Configure the Cluster – PostgreSQL

Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

Important Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously.

Access the LikeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application, or as an applet within your Java-Enabled Web Browser.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 error.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

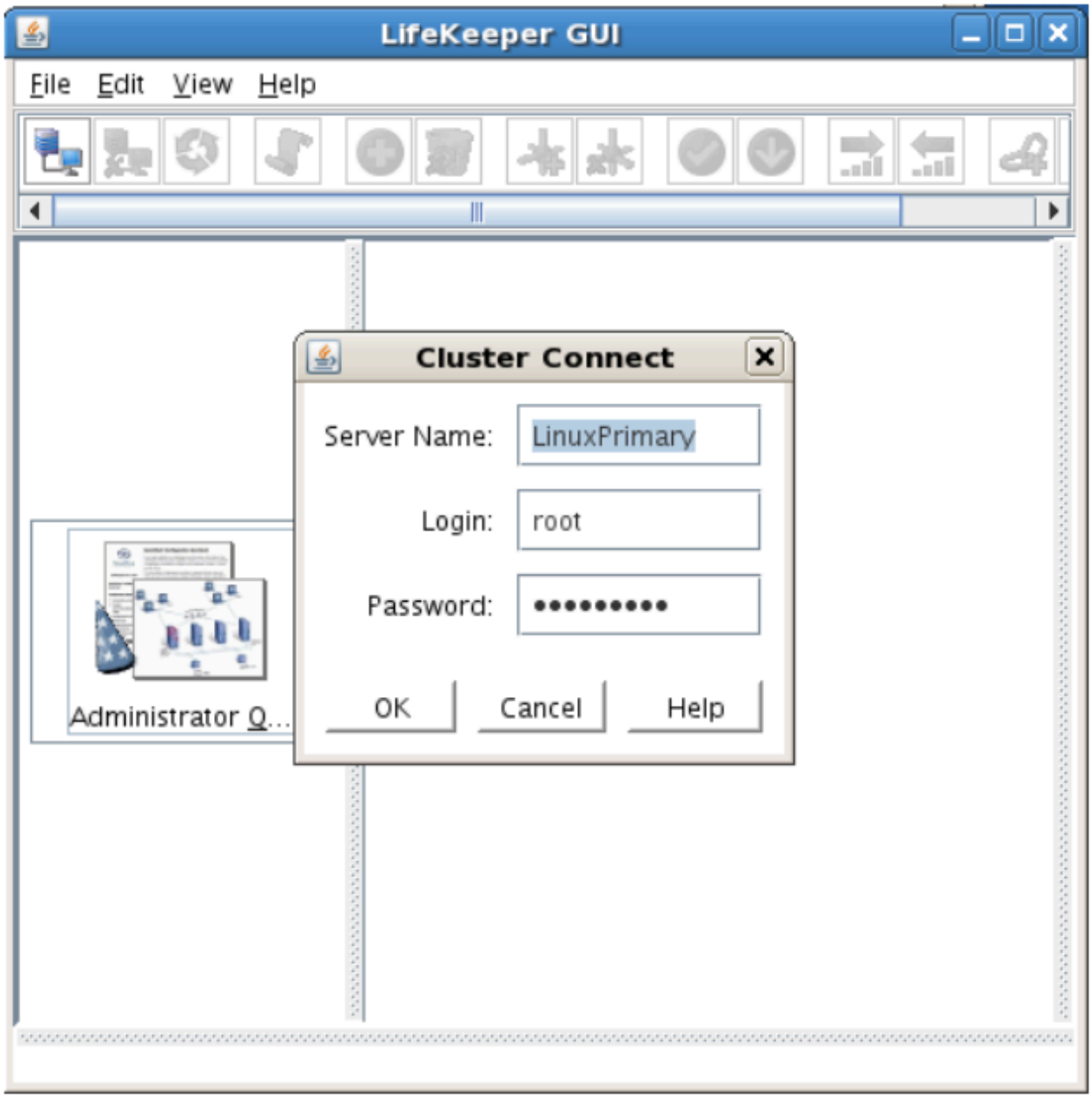
   a. /opt/LifeKeeper/bin/lkGUIapp &

3. To Connect to the LifeKeeper GUI Applet from a Web Browser, go to:

   a. http://<hostname>:81

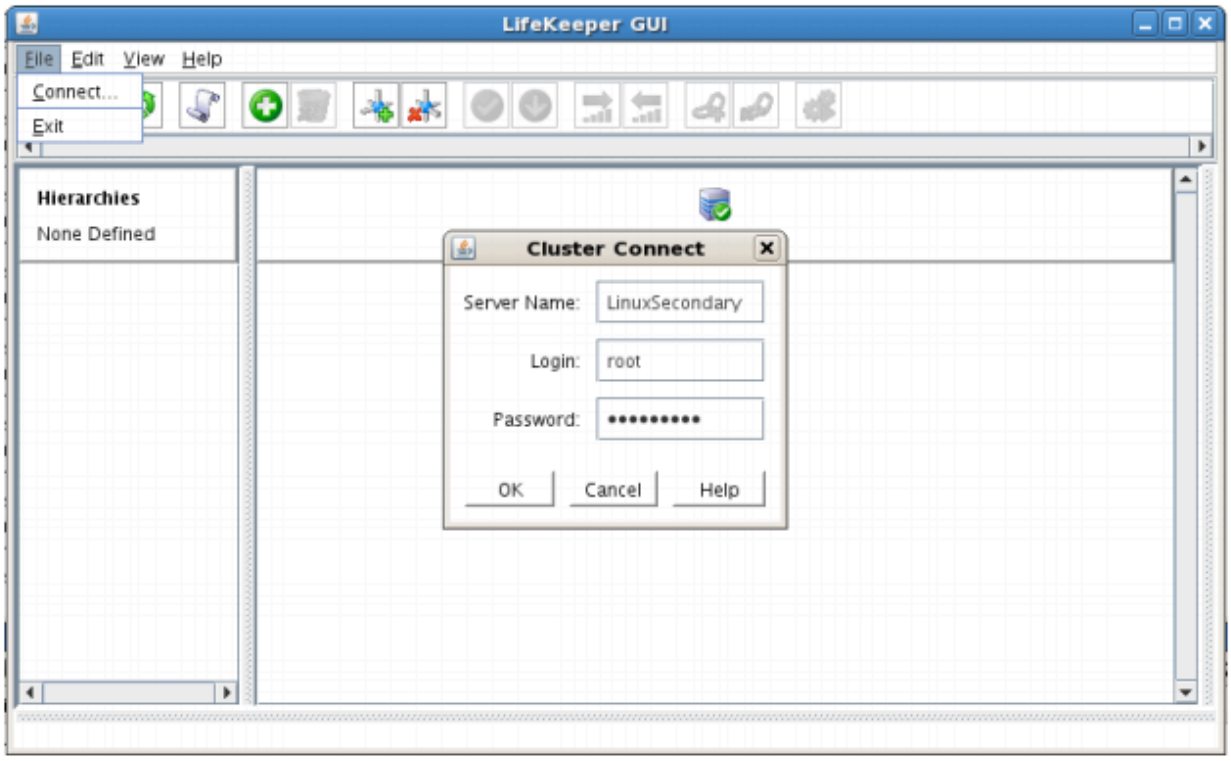
4. Enter the name of the server you wish to connect to (this field will be populated with the name of the server you are on, if you are running the GUI from a server with LifeKeeper installed) along

with your root credentials and click OK.



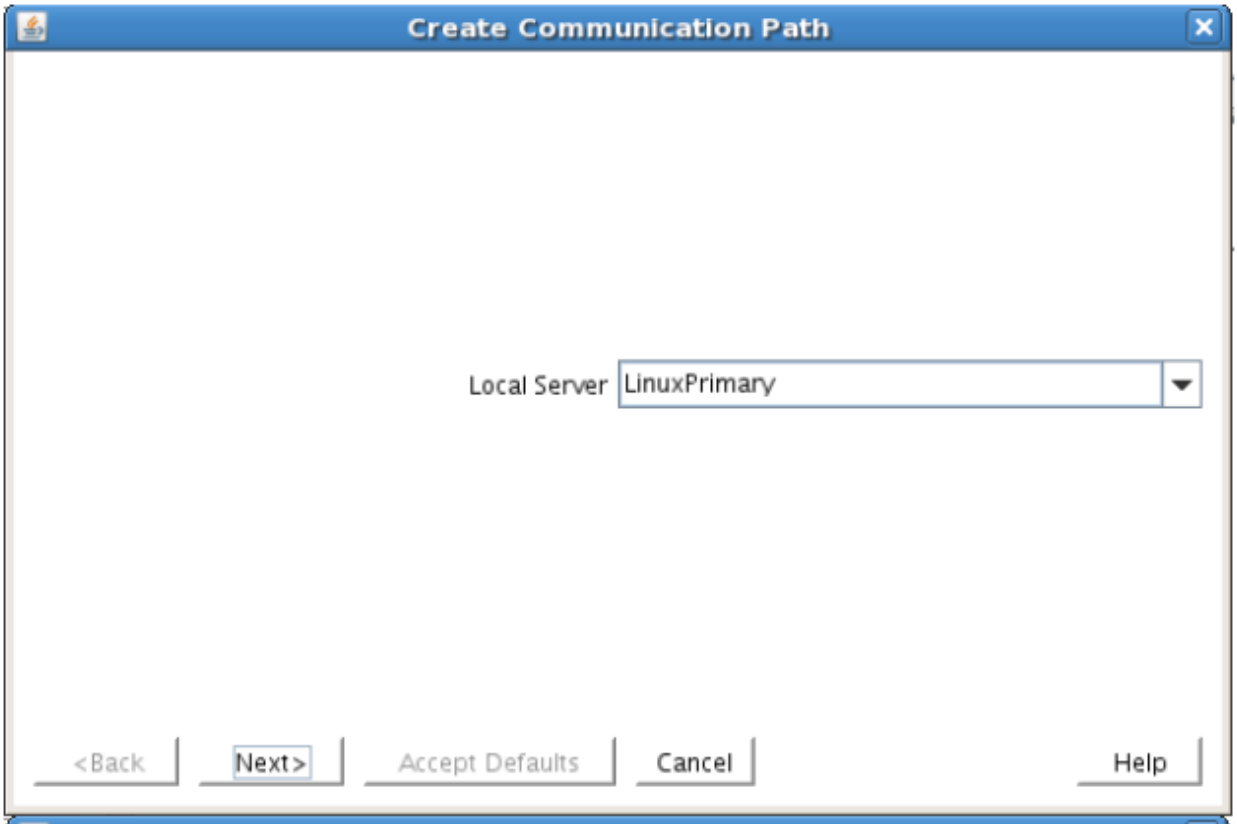
Create Communication (Comm) Paths

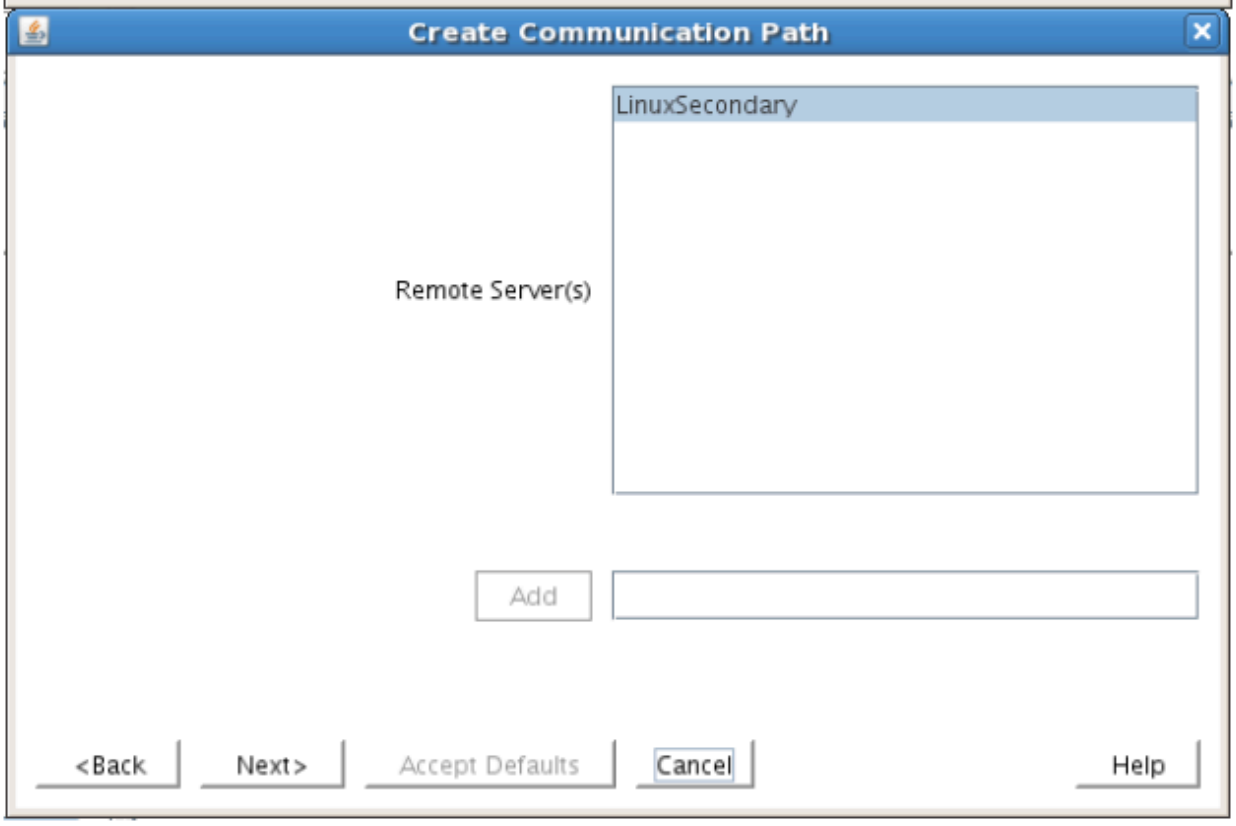
5. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



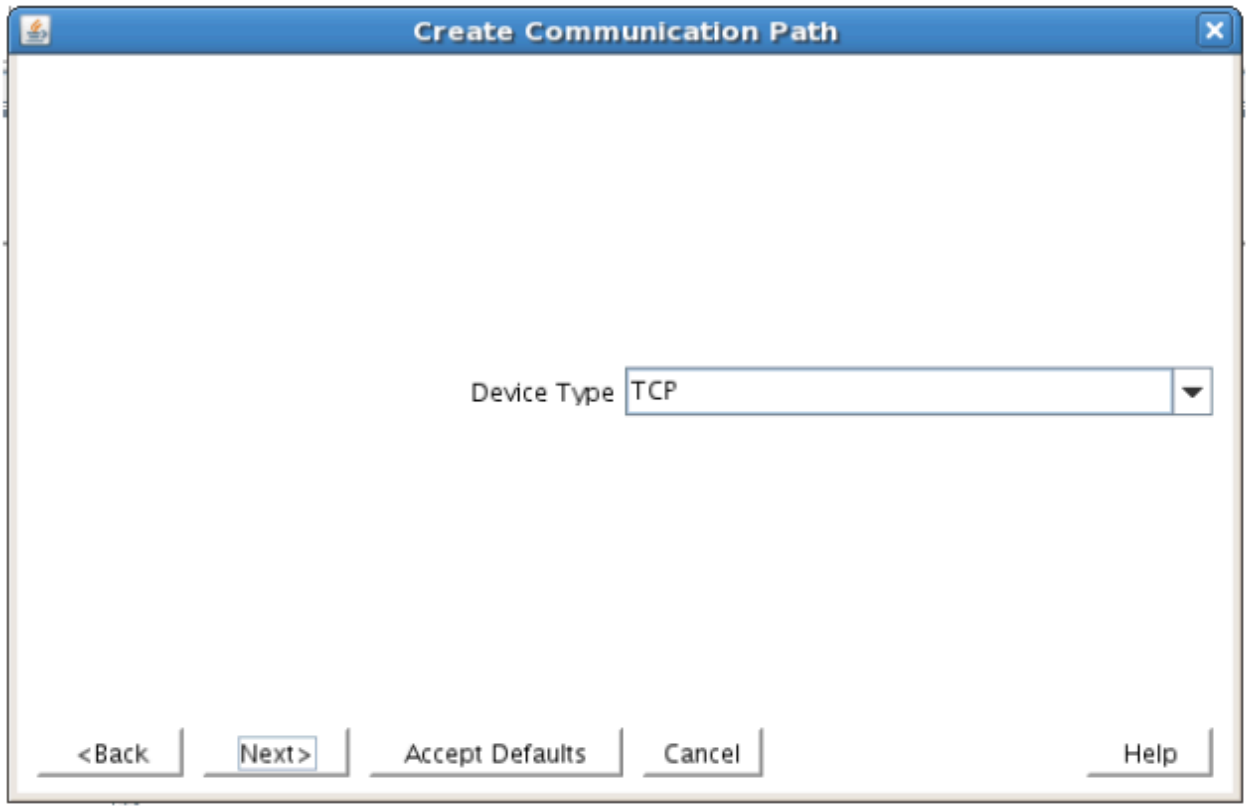
6. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

7. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

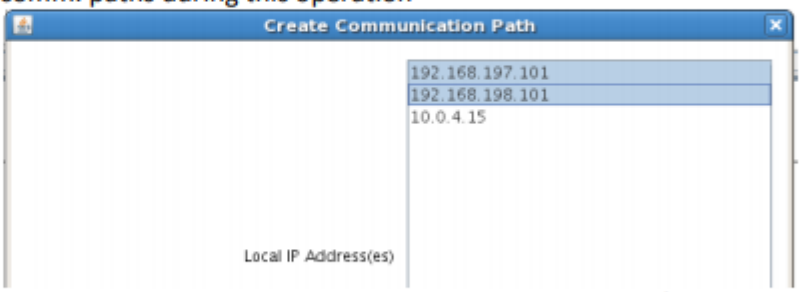


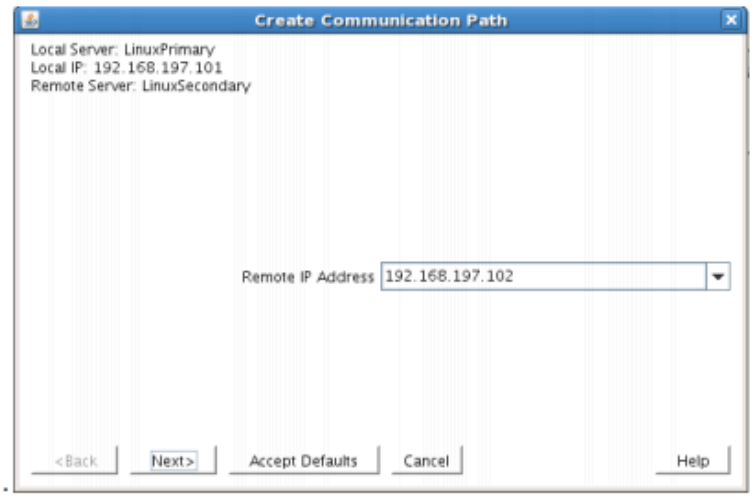


8. Select TCP for Device Type and Click Next.



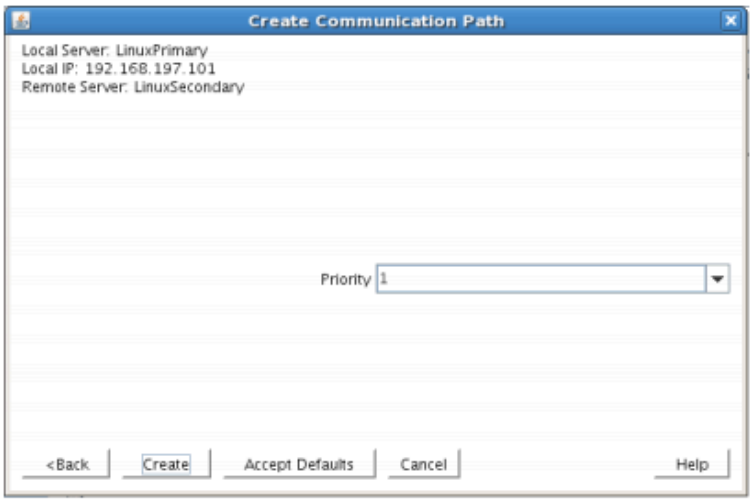
9. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

Field	Tips
For TCP/IP Comm Path...	
Local IP Address	Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation 
Remote IP Address	Choose the IP address to be used by the remote server for this comm path

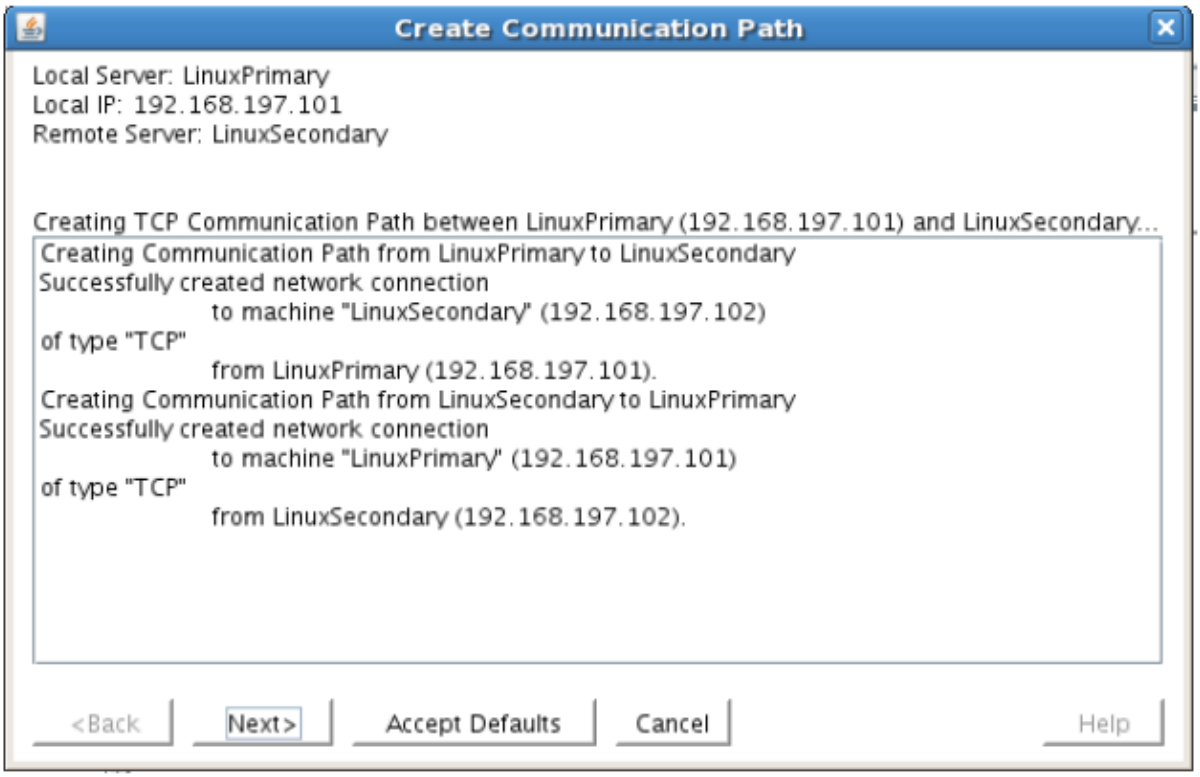


Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority



10. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



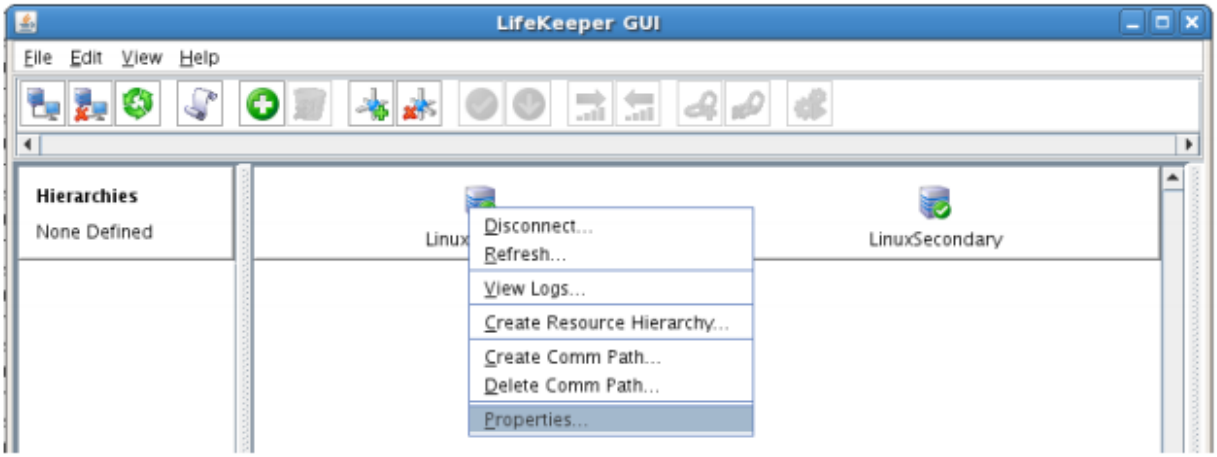
If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

11. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

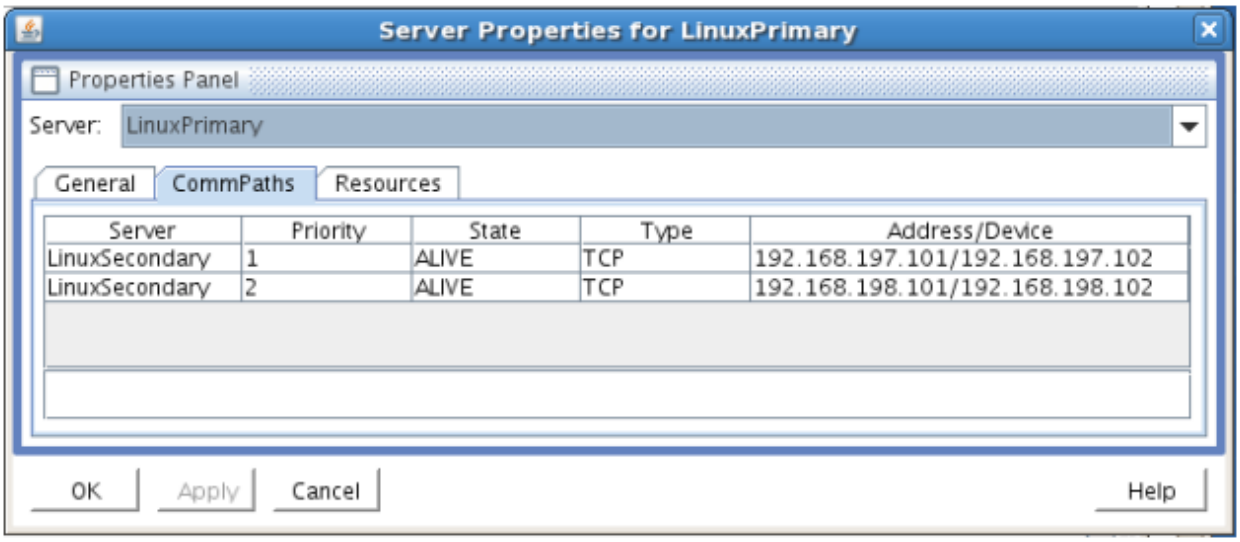
Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.



2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.

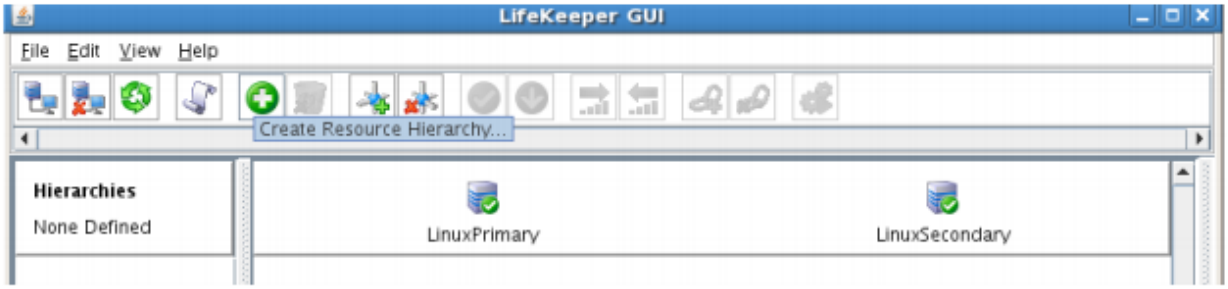


Create the LifeKeeper Hierarchy

Create and Extend an IP Resource

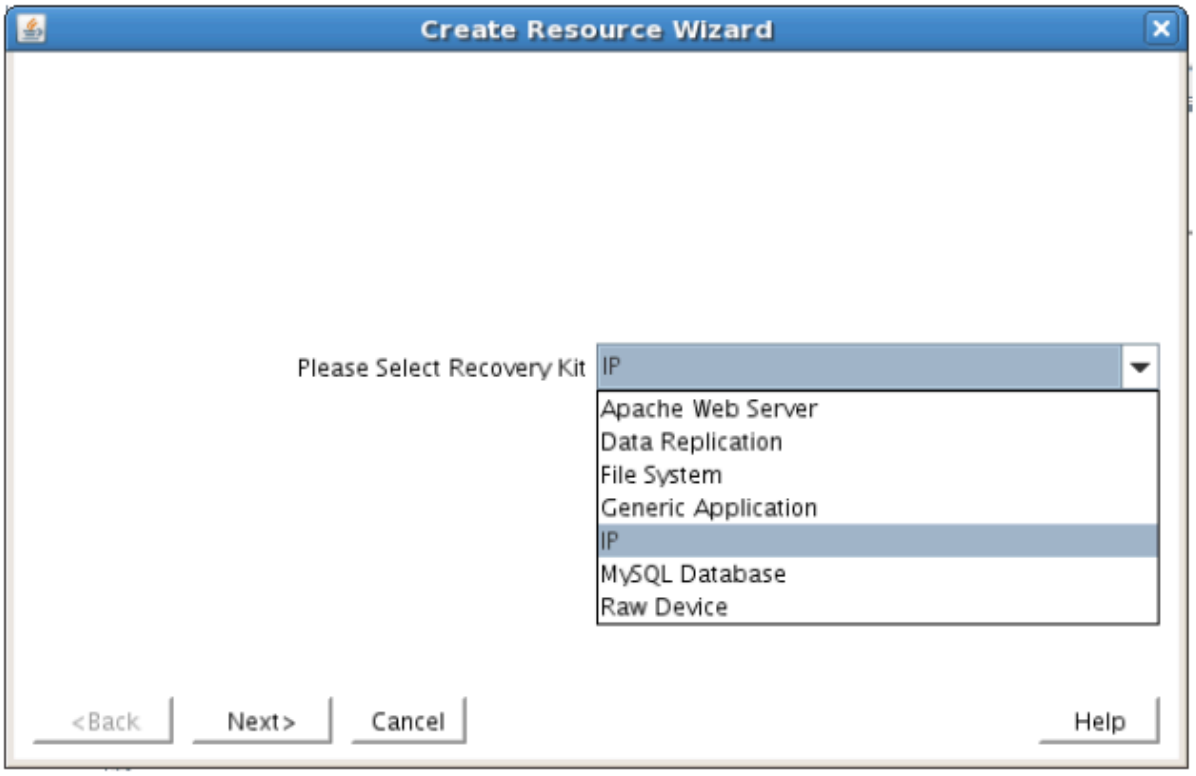
In LifeKeeper, create an IP resource and extend it to the secondary server by completing the following steps. This Virtual IP will have the ability to move between cluster nodes along the application that depends on it.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.



The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select IP Address and click Next.



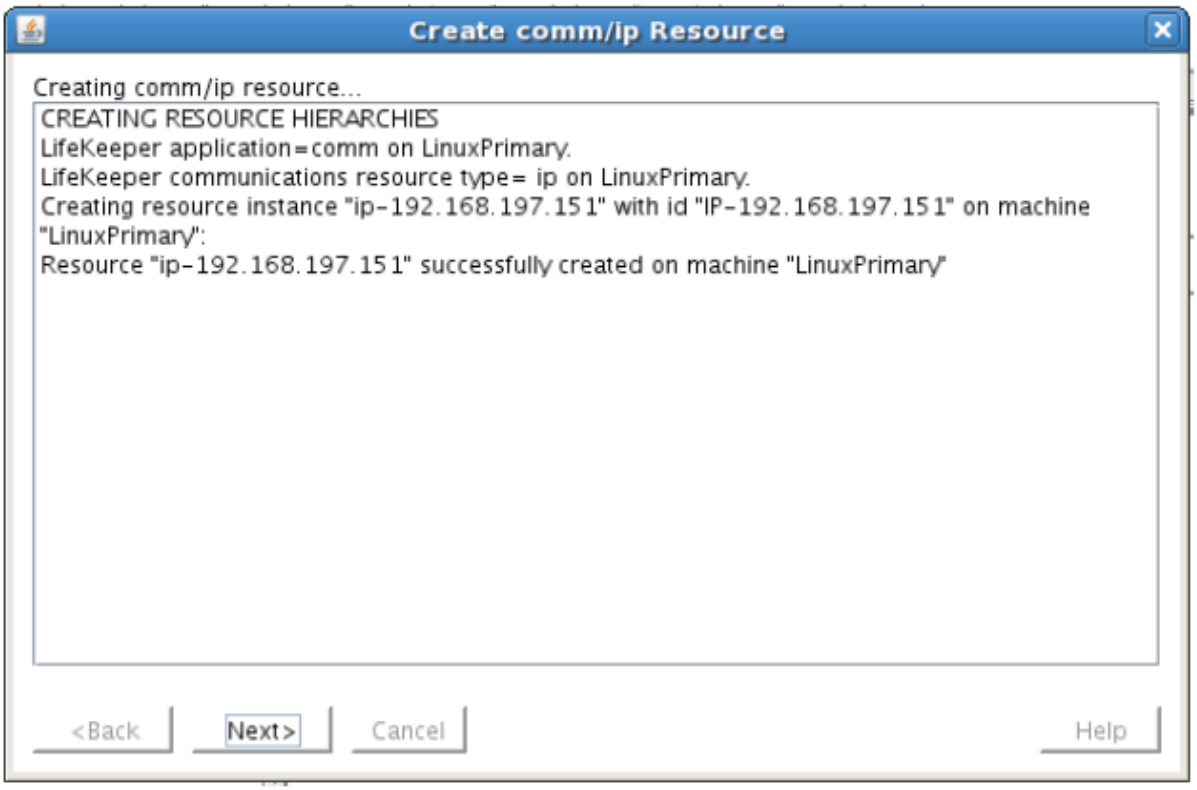
3. Enter the appropriate information for your configuration. The table below contains a list of the fields that display and additional information to assist you as you complete this procedure. Recommended values are also show below. You can also click the Help button for further information. Press Next to continue after entering the required information.

IP Creation Field Definitions

Field	Tips
Resource Type	Select IP Address as the resource type and click Next.
Switchback Type	Select Intelligent and click Next.
Server	Select the Server where the IP resource will be created. Select your Primary server and click Next.
IP Resource	Enter the virtual IP information and click Next Example **192.168.167.151** **Note** This is an IP address that is not currently in use anywhere on your network. This is the address that all clients will use to connect to the protected resources.

Netmask	The IP subnet mask that your TCP/IP resource will use on the target server. Any standard netmask for the class of the specific TCP/IP resource address is valid. In our sample configuration 255.255.255.0 is used for a subnet mask on both networks. **Note:** The subnet mask you choose, combined with the IP address, determines the subnet that will be used by the TCP/IP resource and should be consistent with the network configuration.
Network Connection	This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. Select the correct NIC and click Next.
IP Resource Tag	Accept the default value and click Next. This value only affects how the IP is displayed in the GUI. The IP resource will be created on our Primary server.

4. LifeKeeper will create and validate your resource. After receiving the message that the resource has been created successfully, click Next when the following dialog box appears so that you can complete the process of Extending the IP Resource to our Secondary server, below.



Extending the IP resource will start automatically after you have finished creating an IP address resource if you clicked Next in the dialog box displayed above. You can also start this from an existing IP address resource by right clicking on the active resource and selecting Extend Resource Hierarchy.

Refer to the table below to complete the Extend IP Resource procedure.

Field	Recommended Entries or Notes
Switchback Type	Leave as “intelligent” and click Next
Template Priority	Leave as default (1)
Target Priority	Leave as default (10)
Network Interface	This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. The correct physical NIC should be selected by default. Please verify and then click Next
IP Resource Tag	Leave as default.
Target Restore Mode	Select Enable and click Next.
Target Local Recovery	Select Yes to enable Local Recovery for the SQL resource on the Target server.
Backup Priority	Accept the default value.

5. After receiving the message Hierarchy extend operations completed, click Finish and then click Done

6. Your IP resource (192.168.197.151) is now fully protected and has the ability to “float” between cluster nodes as needed. Looking at the LifeKeeper GUI you will notice that the IP resource is Active on the Primary cluster node and Standby on the Secondary cluster node

Create the Shared Filesystem Resource Hierarchy

Create a Filesystem resource to protect the shared iSCSI filesystem and make it high available between cluster nodes. LifeKeeper leverages SCSI Persistent Group Reservations (PGR) to lock the LUN, ensuring that only the active cluster node for the storage resource can access it.

Important At this point, the shared iSCSI LUN needs to already be mounted on the Primary Server. It should NOT be mounted on the Secondary Server. See section titled “Configure iSCSI initiator, discover and login to iSCSI target” above to review the steps involved.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select File System and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

Field	Recommended Entries or Notes
Switchback Type	Intelligent
Server	LinuxPrimary (Primary Cluster Node)
Mount Point	Select /var/lib/pgsql . Note that LifeKeeper scans the system for LUNS that are sharable between cluster nodes. The list of possible shared LUNS is presented automatically in this step of the wizard.

4. Select Create Instance to define this resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the File System resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Your resource hierarchy should look as follows:

Create the PostgreSQLSQL Resource Hierarchy

Create a PostgreSQL resource to protect the PostgreSQL database and make it high available between cluster nodes.

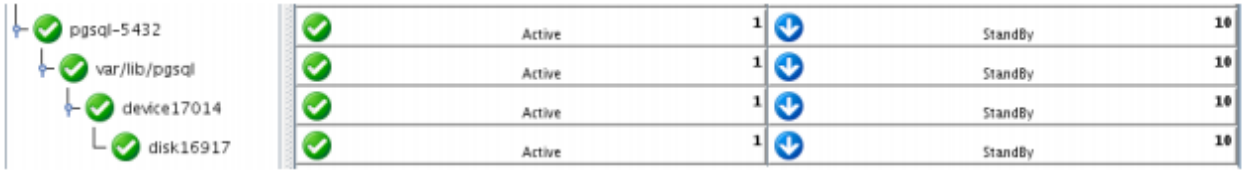
Important At this point, PostgreSQL needs to be already running on the Primary Server. It should NOT be running on the Secondary Server. See section titled “Install, Configure, and Start PostgreSQL” above to review the process to configure and start PostgreSQL as needed.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select PostSQL Database and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

Field	Recommended Entries or Notes
Switchback	Intelligent

Type	
Server	LinuxPrimary (Primary Cluster Node)
PostgreSQL Executable	Leave as default (/usr/bin) since we are using a standard PostgreSQL
Location	install/configuration in this example. This field is used to specify the directory path containing the PostgreSQL executables.
PostgreSQL Client	Leave as default (/usr/bin/psql) . This field is used to specify the directory
Executable Location	path containing the PostgreSQL executable psql.
PostgreSQL Administration	Leave as default (/usr/bin/pg_ctl). This field is used to specify the
Executable Location	directory path containing the PostgreSQL executable pg_ctl.
PostgreSQL Data Directory	/var/lib/pgsql/data. This field is used to specify the location of the PostgreSQL data directory (datadir) that will be placed under LifeKeeper protection. The specified directory must exist and reside on a shared or replicated file system.
PostgreSQL Port 5432.	This field is used to specify the TCP/IP port number on which the postmaster daemon is listening for connections from client applications
PostgreSQL Socket Path	Leave as default (/tmp/.s.PGSQL.5432) . This field is used to specify the full path to the Unix- domain socket on which the postmaster daemon is listening for connections from client applications.
Enter Database Administrator User	Enter “postgres” . This field is used to specify a PostgreSQL Database Administrator User name for the specified database instance with connection and administrator privileges for the instance
PostgreSQL Logfile	Leave as default (/tmp/pgsql-5432.lk.log) . This field is used to specify the log file path that will be used for the PostgreSQL log file.
Database tag	Leave as default

4. Select Create to define the PostgreSQL resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the PostgreSQL resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Note: LifeKeeper will automatically identify that the PostgreSQL resource has a dependency on the FileSystem resource (/var/lib/pgsql). The Filesystem Resource will appear underneath the PostgreSQL resource in the GUI
9. Your resource hierarchy should look as follows:



Create the PostgreSQL IP Address Dependency

In this step will define an additional dependency: that PostgreSQL depends on a Virtual IP (192.168.197.151) so that the IP address follows the PostgresSQL database should it move.

1. From the LifeKeeper GUI toolbar, right-click on the “pgsql-5432” resource
2. Select “Create Dependency” from the right-click context menu
3. In the Child Resource Tag dropdown menu, select “ip-192.168.197.151”
4. Click Next
5. Click Create Dependency
6. Click Done
7. The Virtual IP address resource (192.168.197.151) will now appear underneath the PostgreSQL resource in the LifeKeeper user interface. This ensures that resources move together, and are started/stopped in the proper order.
8. Your resource hierarchy should look as follows

At this point in the Evaluation, we have fully protected PostgreSQL, and its dependent resources: IP addresses, and Shared Storage.

11.7.8. Test Your Environment – PostgreSQL

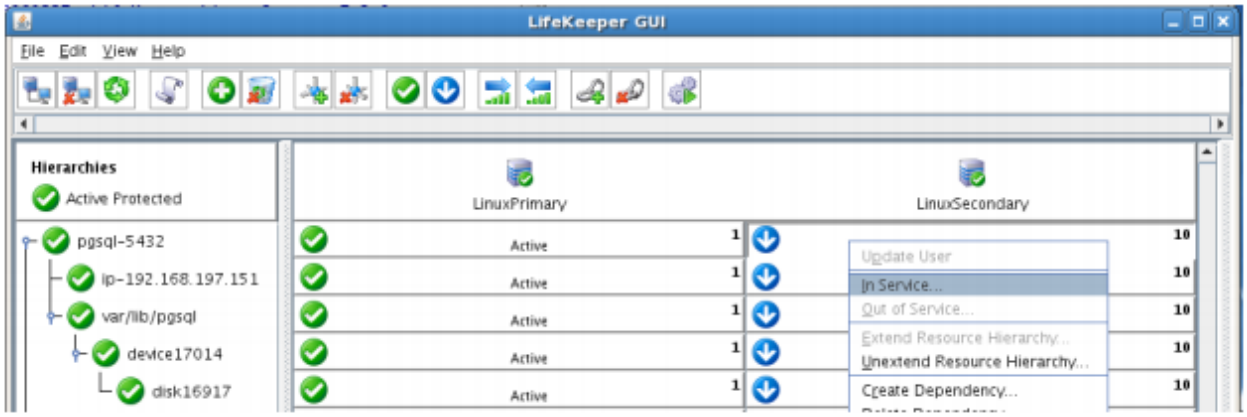
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

Note: For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

Manual Switchover of the PostgreSQL Hierarchy to Secondary Server

Procedure:

- From the LifeKeeper GUI, right click on the PostgreSQL resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



Expected Result:

- Beginning with the PostgreSQL resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXSECONDARY

Tests/Verification:

- Using the LifeKeeper GUI, verify that the PostgreSQL and dependent resources are active on LINUXSECONDARY.
- Run “ifconfig –a” on LINUXSECONDARY to validate that the IP Address 192.168.197.151 is active on LINUXSECONDARY
- Run “df –h” to verify that the /var/lib/pgsql shared iSCSI filesystem is mounted on LINUXSECONDARY

- Verify the PostgresSQL services are running on LINUXSECONDARY by running “ps –ef | grep –i postgres”
- On LINUXSECONDARY run the following command to verify client connectivity to the PostgreSQL database:
    -
        - # su – postgres
        - # psql

- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXPRIMARY, run “mount /dev/sdc1 /var/lib/pgsql”. This should FAIL because LINUXPRIMARY does not own the SCSI reservation on this LUN.

Manual Switchover of the PostgreSQL Hierarchy back to Primary Server

Procedure:

- From the LifeKeeper GUI, right click on the PostgreSQL resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

Expected Result

- Beginning with the PostgreSQL resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXPRIMARY

Tests/Verification:

- Using the LifeKeeper GUI, verify that the PostgreSQL and dependent resources are active on LINUXPRIMARY.
- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.151 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/lib/pgsql shared iSCSI filesystem is mounted on LINUXPRIMARY
- Verify the PostgreSQL services are running on LINUXPRIMARY by running “ps –ef | grep –i postgres”

- On LINUXPRIMARY run the following command to verify client connectivity to the PostgreSQL database:
    -
        - # su – postgres
        - # psql

- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXSECONDARY, run “mount /dev/sdc1 /var/lib/pgsql”. This should FAIL because LINUXSECONDARY does not own the SCSI reservation on this LUN.

Simulate a network failure on the Primary Server by failing the IP resource

> **IMPORTANT NOTE:** Only perform this test if you have more than one communications path configured.

If you perform this test and have only one communications path configured, your system will enter a split-brain scenario as described in the LifeKeeper Administration Guide found here. Refer to this document for more information or contact SIOS presales technical support for assistance in resolving this condition.

Procedure

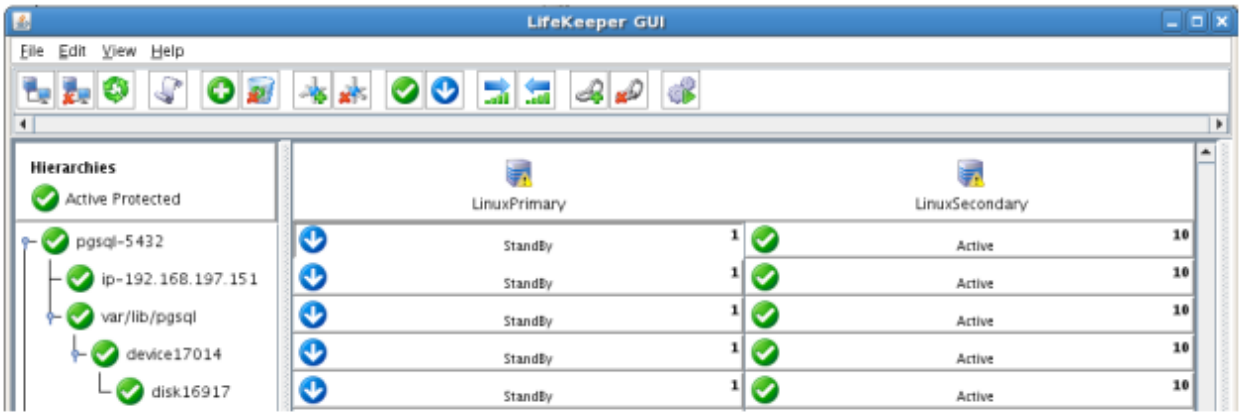
- On LINUXPRIMARY, pull the network cable attached to the NIC that the virtual IP address is configured on

Expected Result:

- The IP Resource should fail first.
- The entire hierarchy should failover to LINUXSECONDARY

Tests/Verification:

- Check the LifeKeeper Log to verify the IP resource failed – “/opt/LifeKeeper/bin/lk_log log”
- Using the LifeKeeper GUI, verify the PostgreSQL resource hierarchy fails over successfully to LINUXSECONDARY
- After this test has been completed, re-connect the network cable on LINUXPRIMARY



Hard failover of the resource from the Secondary Server back to the Primary Server

Procedure:

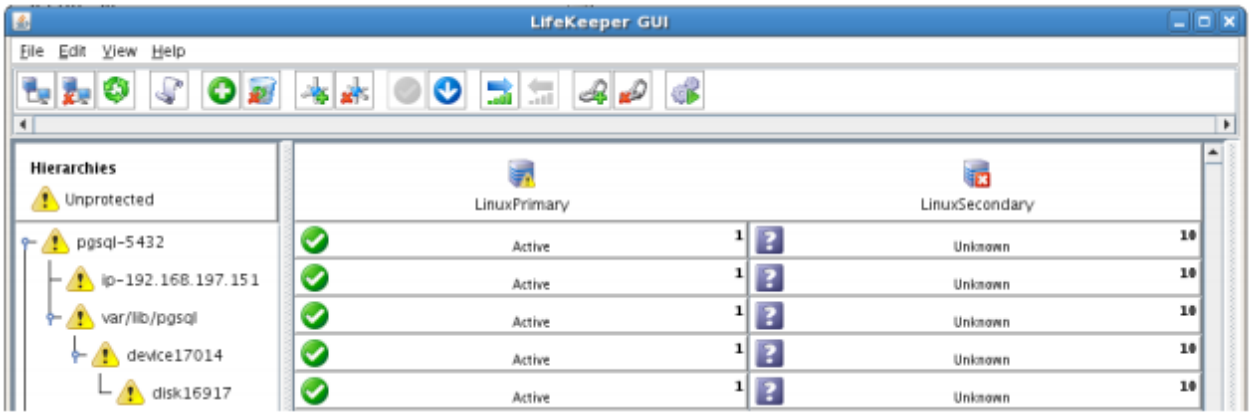
- Pull the power cord on LINUXSECONDARY, as this is the server with all resources currently In Service.

Expected Result:

- After failure has been detected, beginning with the dependent resources (IP and Volume), all resources will be brought in service on LINUXPRIMARY.

Tests/Verification:

- Using the LifeKeeper GUI, verify the mirror has reversed and is in a Resync Pending state waiting for LINUXSECONDARY to come back on line.
- Verify the PostgreSQL Server services are running on LINUXPRIMARY.



Bring Failed Server back on line

Procedure:

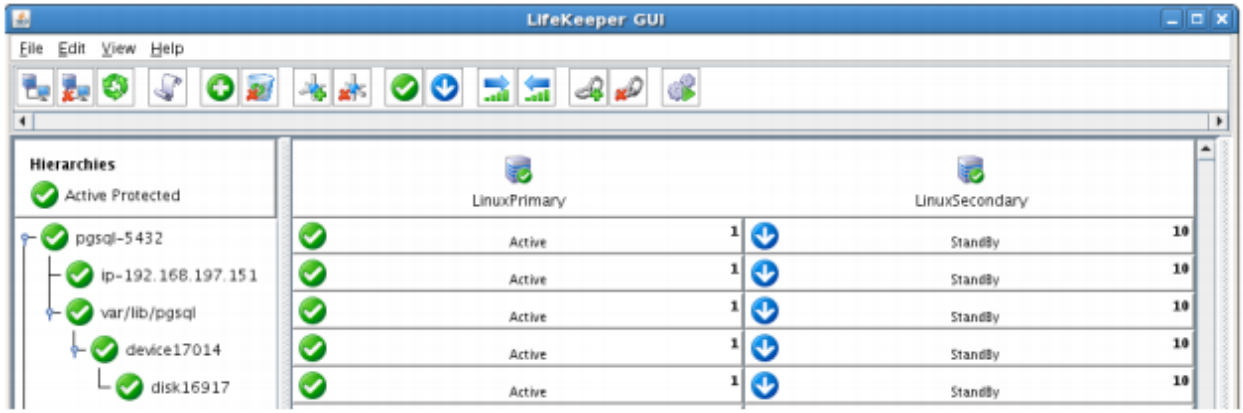
- Plug the power cord back into LINUXSECONDARY and boot it up.

Expected Result:

- Using the LifeKeeper GUI, verify that LINUXSECONDARY is coming back up and has become the Standby Server.

Tests/Verification:

- Verify the PostgreSQL Hierarchy is in service on LINUXPRIMARY and standby on LINUXSECONDARY.



Verify Local Recovery of PostgreSQL Server

Procedure:

- Kill the PostgreSQL processes via the command line:
- # ps –ef | grep postgres
- # (kill -9 the PIDs returned)
- run “ps –ef | grep postgres once again to verify that the processes no longer exist

Expected Result: (Assumes Local Recovery for SQL resource is set to YES)

- The PostgreSQL Server service should stop.
- The PostgreSQL quickcheck process will automatically restart the PostgreSQL Server Service when it runs periodically.
- No failure of PostgreSQL should occur.

Tests/Verification:

- Execute “ps –ef | grep postgres” once again to verify that the postgresql processes have been restored locally on LINUXPRIMARY.
- Verify connectivity to the postgreSQL database by running:
  -

    - # su – postgres
    - # psql

- If you inspect the LifeKeeper logs, you will see information indicating that LifeKeeper dectected the failure of the PostgreSQL service and recovered it locally. Run /opt/LifeKeeper/bin/lk_log log for more information.

11.8. Apache/MySQL Cluster Using Both Shared and Replicated Storage

Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product to make Apache and MySQL highly available. If Apache and MySQL are not already installed, please allocate some time to install it on your servers. Once this task has been completed, you may install and configure SIOS Protection Suite for Linux.

There are five phases in this process:

- Phase 1 – Prepare to Install
- Phase 2 – Configure Storage
- Phase 3 – Install and Configure Apache/PHP
- Phase 4 – Install and Configure MySQL
- Phase 5 – Install SIOS Protection Suite for Linux
- Phase 6 – Configure your LifeKeeper Cluster
- Phase 7 – Test Your Environment

11.8.1. Terms to Know – Apache

The following terms are used throughout this document and while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

Network Communication Terms

Crossover cable – A cable used to directly connect computing devices together instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

Types of LifeKeeper Servers

Server – A computer system dedicated to running software application programs.

Active Server – This is the server where the resource hierarchy is currently running (IN SERVICE).

Standby Server – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

Primary Server – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

Secondary Server – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

Source Server – This is the server in a LifeKeeper cluster that is using data replication (Active Server). It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

Target Server – This is the server in a LifeKeeper cluster using data replication (Standby Server). The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

SIOS Data Replication Terms

Replication – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

Synchronous – A replication scheme in which the data is confirmed written and valid on the target before the write operation occurs on the source disk through a series of information exchanges. Synchronous mirrors should only be implemented on high speed (100Mbps+) networks due to the network overhead involved.

Asynchronous – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there andcan be written on them.

Rate of Change – A measure of the amount of data which is changing over a set period of time.

Compression – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

Throttling – An optionally implemented mechanism to limit the bandwidth used for replication.

LifeKeeper Product Terms

Communications Path – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

Heartbeat – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

Split Brain – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

Failover – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

Switchover – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

Switchback – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

Resource – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

Extend a Resource – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously.

Resource Hierarchy – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

Shared Storage – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally called I/O fencing.

Data Replication (Disk Mirroring) – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

Source – The partition on the source server used for replication. The “gold” copy of the data.

Target – The partition on the target server used for replication.

Switchable IP Address – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

11.8.2. The Evaluation Process – Apache

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux in a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

Important Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

11.8.3. Prepare to Install – Apache

Hardware Requirements

Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional disk partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system (/) and boot (/boot) partitions are not eligible for replication.

Note: You may use more than one partition for replicated data, allowing for separation of multiple database files and/or log directories for performance reasons.

Client

This system is not required but is recommended for testing the cluster environment.

- A standard linux terminal running the MySQL client can be used to test the configuration.

Software Requirements

Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
  - º RedHat Enterprise Linux 5 (5.4+ recommended) or 6.x
  - º CentOS Linux 5 (5.4+ recommended) or 6.x
  - º Oracle Enterprise Linux 5 (5.4+ recommended), 6.3, 6.4
- RedHat Compatibility Kernel Only

- º SuSE Linux Enterprise Server 10 or 11 (11 recommended)

- º See

Linux Release Notes for a full list of supported Operating SystemsCurrent patches / security updates are recommended. Satisfied dependencies; especially if the Linux installation package selection was base/ minimal you will need to refer to the dependencies documentation at Linux Dependencies Its recommended that IPtables is disabled

- º # /etc/init.d/iptables off

- º # chkconfig iptables off

- º See

here for information regarding the ports SIOS Protection Suite for Linux uses. Disable SELinux :

- º Edit /etc/selinux/config

- º Set

SELINUX=disabled (note: permissive mode is also acceptable) Check the configuration of your /etc/hosts file

- º localhost.localdomain and localhost are the only entries that can be on 127.0.0.1

- º Create a separate entry for your hostname with a static address

GUI Authentication with PAM

- º

SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).

- º Users are identified and authenticated against the system's

PAM configuration. Privilege levels are determined from group membership as provided through PAM.

-

- In order to access the

GUI, a user must be a member in one of the three LifeKeeper groups: lkadmin, lkoper or lkguest.

- See the following

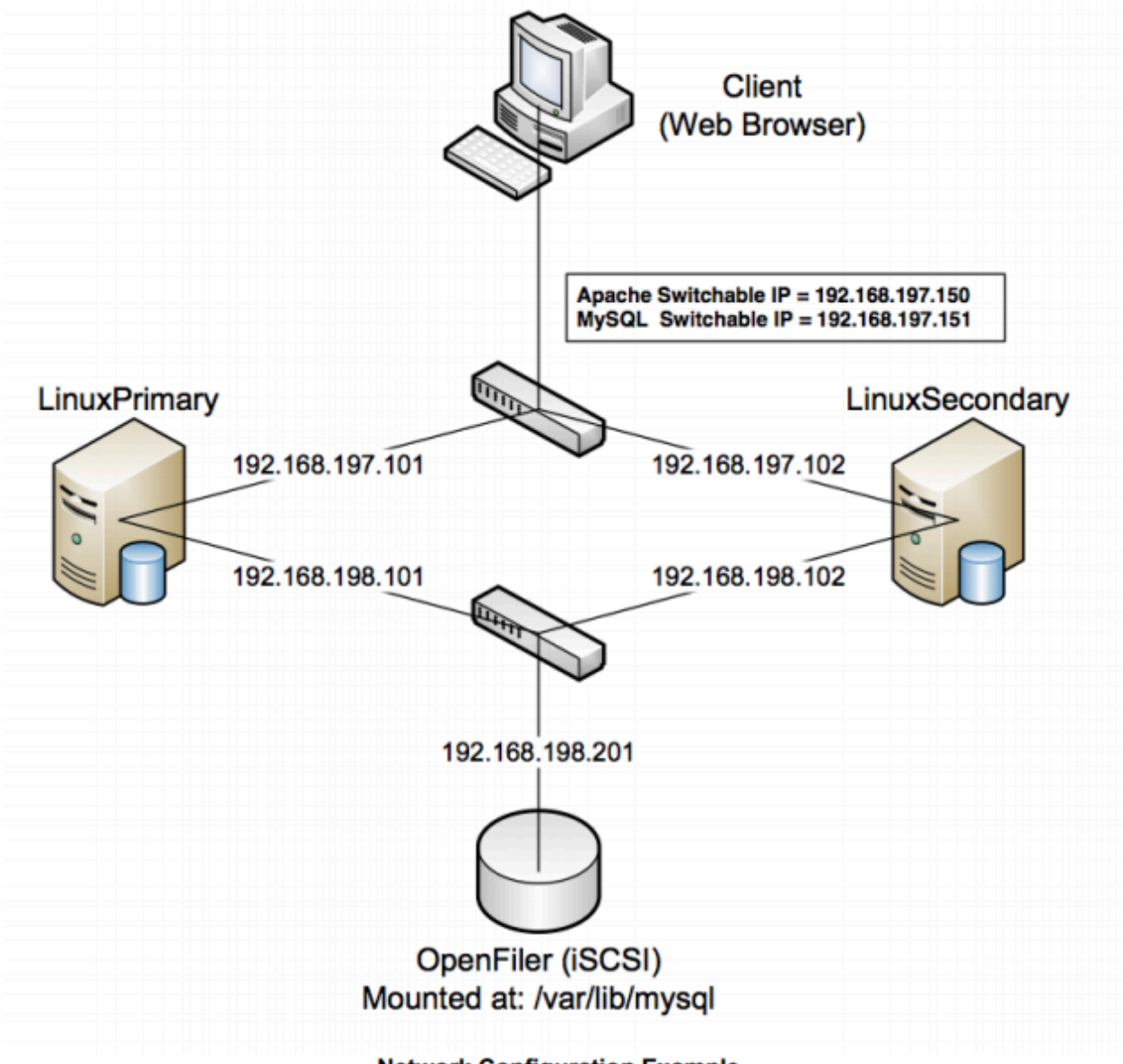
URL for more information on this topic:
Configuring GUI Users

Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi-homed, between two LAN segments (the second NIC in each server could even be connected via a cross-over cable if a second physical network is not available). The second NIC is optional in this configuration, but highly recommended in production environments to avoid a single point of failure.

In this example evaluation scenario we will be leveraging both local, replicated storage (with the Apache configuration) as well as Shared Storage (iSCSI, for the MySQL configuration). OpenFiler is a storage appliance server that will serve an iSCSI target to LinuxPrimary and LinuxSecondary.



Network Configuration Example

Primary Server and Secondary Servers

- Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

  Example:

  192.168.197.101 LinuxPrimary

  192.168.197.102 LinuxSecondary

- See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.
- Public Network connection(s) configured with:
  -
    - Static IP address
  -

- º Correct subnet mask
- º Correct gateway address
- º Correct DNS server address(es)

Private Network connection(s) configured with:

- º Static IP address (on a different subnet from the public network)
- º Correct network mask
- º No gateway IP address
- º No

DNS server addresses

Client

Must be able to communicate on the same subnet/network as the servers Public interface addresses. In our example, this is the 192.168.197.0/24 network.

11.8.4. Configure Storage – Apache

Before you Begin

Ensure the following:

- You have an extra disk/partition on both servers that can be used for data replication. A target volume's size must equal to or larger than the size of its source disk/partition.
- Shared storage is available. This can either be Fiber Channel SAN, iSCSI, NAS, etc. In this example we will review configuration of an iSCSI target for use as our MySQL database storage repository.

Partition local storage for use with SIOS DataKeeper for Linux

Primary Server

On your Primary server, perform the following actions:

1. Identify an existing free, unused disk partition to use as our Apache repository. Alternatively, create a new partition. Use the "gdisk" utility to partition your disk appropriately. In this example /dev/sdb is an unused disk where we will create a single partition

    a. gdisk /dev/sdb

    b. Press "n" to create a new partition

    c. Press "p" to create a primary partition

    d. This example uses a new disk, so we will use all default values (Partition 1, entire disk) Hit Enter twice to confirm these parameters

    e. Press "w" to write the partition table and exit gdisks

Example

[root@LinuxPrimary ~]# **gdisk /dev/sdb**

Command (m for help): **n**

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): **1**

First cylinder (1-256, default 1): **<enter>**

Using default value 1

Last cylinder or +size or +sizeM or +sizeK (1-256, default 256): **<enter>**

Using default value 256

Command (m for help): **w**

The partition table has been altered! Calling ioctl() to re-read partition table.

Syncing disks.

[root@LinuxPrimary ~]#

[root@LinuxPrimary ~]# df /var/www

Filesystem 1K-blocks Used Available Use% Mounted on

/dev/sdb1 253855 11083 229666 5% /var/www

[root@LinuxPrimary ~]# ls /var/www

cgi-bin error html icons lost+found manual usage

2. Format the newly created disk partition

   # mkfs.ext3 /dev/sdb1

3. Mount the partition temporarily at /mnt

   # mount /dev/sdb1 /mnt

4. Move any existing data from /var/www/ into this new disk partition (assumes a default apache configuration)

   # cd /var/www

   # mv * /mnt

5. Remount /dev/sdb1 at /var/www

```
# cd /root

# umount /mnt

# mount /dev/sdb1 /var/www
```

6. Note: there is no need to add this partition to /etc/fstab. Lifekeeper will take care of mounting this automatically.

Result

```
[root@LinuxPrimary ~]# df /var/www
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sdb1 253855 11083 229666 5% /var/www
[root@LinuxPrimary ~]# ls /var/www
cgi-bin error html icons lost+found manual usage
```

Secondary Server

7. On your Secondary server, only perform Step #1 above, where you partition the disk. The size of the Target volume needs to be the same size, or greater, than our Source volume.

Configure iSCSI initiator, discover and login to iSCSI target

This Evaluation guide will not cover how to setup an iSCSI Target Server. It is assumed that the shared storage already exists in your environment. If you don't have shared storage and wish to configure it, a simple solution is to use OpenFiler (http://www.openfiler.com/), an Open Source storage management appliance, which can be run on physical hardware or as a virtual machine.

On both Primary and Secondary servers, perform the following functions:

1. If not already installed, ensure that the **iscsi-initiator-utils** rpm package is installed:

```
# yum install iscsi-initiator-utils
```

2. Start the iscsid service and enable it to automatically start when the system boots

```
# service iscsid start

# chkconfig iscsid on
```

3. Configure the iscsi service to automatically start, which logs into iSCSI targets needed at system start up.

```
# chkconfig iscsi on
```

4. Use the iscsiadm command to discover all available targets on the network storage server (OpenFiler)

iscsiadm -m discovery -t sendtargets -p <name or IP of iSCSI server>

Example
[root@LinuxPrimary init.d]# iscsiadm -m discovery -t sendtargets -p 192.168.198.201
iqn.2006-01.com.openfiler:tsn.mysql

5. Manually Login to the iSCSI Target

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.mysql –p 192.168.198.201 –login

6. Configure Automatic Login

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.mysql –p 192.168.198.201 —op update -n node.startup -v automatic

7. Use the “gdisk” command to format your iSCSI LUN, if needed

   # gdisk /dev/sdc

8. Create a filesystem on your new iSCSI LUN Partition, sdc1

   # mkfs.ext3 /dev/sdc1

9. Mount your iSCSI LUN at /var/lib/mysql (assuming a default mysql configuration). If data already exists in this directory, make sure to move it into the shared iSCSI LUN

   # mount /dev/sdc1 /var/lib/mysql

10. At this point you now have a local partition, /dev/sdb1 mounted at /var/www and an iSCSI shared LUN, /dev/sdc1, mounted at /var/lib/mysql. Our disk layout now look as follows (example):

Example
[root@LinuxPrimary mysql]# df
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sda2 25967432 3683016 1976400 66% /
/dev/sda1 101086 24659 71208 26% /boot
tmpfs 517552 0 517552 0% /dev/shm
/dev/sdb1 253855 11132 229617 5% /var/www
/dev/sdc1 966644 38944 878596 5% /var/lib/mysql

11.8.5. Install and Configure Apache and PHP

Before you Begin

Ensure the following:

1. If you are not familiar with installing and configuring Apache, please refer to the Apache documentation.
2. Apache should not be running at the time you attempt to protect it with LifeKeeper.

Important: If Apache and PHP are already installed, skip to step #2 in the section below. You will need to verify the Apache configuration and/or relocate the location of the website data.

Primary Server

1. Install Apache.

   a. This example assumes that you are running a RHEL or CentOS 5.X based Linux distribution. For other Linux distros, please refer to your apache documentation for syntax differences.

   b. If the Apache and PHP packages are not already installed, from the command line, , run: “yum install httpd php”

   c. A number of dependencies will most likely be discovered. Install those as well.

2. Apache should be configured so that it will not automatically start when the server boots. LifeKeeper will control the start/stop of the webserver once its protected

   a. Check the status of the webserver: “/etc/init.d/httpd status

   b. If running, please stop of: /etc/init.d/httpd stop

   c. Disable automatic startup: chkconfig httpd off

3. Apache Configuration. In this example, we will assume default Apache settings and directory locations, specifically:

   a. Validate the following settings in /etc/httpd/conf/httpd.conf:

      i. ServerRoot “/etc/httpd”

      ii. DocumentRoot “/var/www/html”

iii. Listen 192.168.197.150:80 (note 192.168.197.150 is the Apache Switchable IP we will configure later in the LifeKeeper user interface)

b. Edit the Listen parameter in the /etc/httpd/conf.d/ssl.conf configuration file

i. Listen 192.168.197.150:443

4. Create a sample Index page:

a. vi /var/www/html/index.php

b. Insert the following single line of code into this file:

<? phpinfo(); ?>

Secondary Server

1. Install Apache/PHP exactly as you did on the primary server, making all of the same configuration changes.
2. There is no need to perform Step #4 in which you create a sample index page. All data in /var/www will be replicated from LinuxPrimary to LinuxSecondary
3. Stop all Apache services on the secondary server

11.8.6. Install, Configure, and Start MySQL – Apache

Primary Server

On your Primary server, perform the following actions:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

   ```
   # yum install mysql mysql-server
   ```

2. Verify that your Shared iSCSI LUN is still mounted at /var/lib/mysql via the “df” command

3. If this is a fresh MySQL install, initialize a sample MySQL database:

   ```
   # /usr/bin/mysql_install_db —datadir=”/var/lib/mysql” —user=mysql
   ```

4. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

   ```
   # chown –R mysql:mysql /var/lib/mysql
   ```

   ```
   # chmod 755 /var/lib/mysql
   ```

5. Finally, manually start the MySQL daemon from the command line. Note: **Do Not** start it via the “service” command, or the /etc/init.d/ scripts

   ```
   # mysqld_safe —user=root —socket=/var/lib/mysql/mysql.sock —port=3306 — datadir=/var/lib/mysql —log &
   ```

6. Verify MySQL is running by connecting with the mysql client:

   ```
   [root@LinuxPrimary mysql]# mysql

   Welcome to the MySQL monitor. Commands end with ; or \g.

   Your MySQL connection id is 2

   Server version: 5.0.77-log Source distribution

   Type ‘help;’ or ‘\h’ for help. Type ‘\c’ to clear the buffer.

   mysql> exit
   ```

Bye

[root@LinuxPrimary mysql]#

7. Update the root password for your mysql configuration. In this example we set the MySQL root password to “SteelEye”

   # echo “update user set Password=PASSWORD where User=‘root’; flush

   privileges” | mysql mysql

8. Verify your new password:

   # mysql mysql –u root –p

   (Enter “SteelEye” as the password)

   #exit

9. Create a MySQL configuration file. We will place this in the same shared directory (/var/lib/mysql/my.cnf)

   # vi /var/lib/mysql/my.cnf

Example

```
# cat /var/lib/mysql/my.cnf
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
pid-file=/var/lib/mysql/mysqld.pid
user=root
port=3306
# Default to using old password format for compatibility with mysql 3.x
# clients (those using the mysqlclient10 compatibility package).
old_passwords=1
# Disabling symbolic-links is recommended to prevent assorted security risks;
# to do so, uncomment this line:
# symbolic-links=0
[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
[client]
user=root
password=SteelEye
```

10. Delete the original MySQL configuration file, located in /etc

```
# rm /etc/my.cnf
```

Secondary Server

On your Secondary Server:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

   ```
   # yum install mysql mysql-server
   ```

2. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

   ```
   # chown –R mysql:mysql /var/lib/mysql
   ```

   ```
   # chmod 755 /var/lib/mysql
   ```

3. There is no need to perform any of the additional steps taken on the Primary Server

11.8.7. Install SIOS Protection Suite for Linux – Apache

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

SPS for Linux Component	Install Location
LifeKeeper Software	/opt/LifeKeeper
LifeKeeper Config File	/etc/default/LifeKeeper

Perform the following actions on **both** Primary and Secondary server.

Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.

2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After successful download you will have downloaded contents similar to the follow directory listing:

      [root@LinuxPrimary ~]# ls -l <directory> total 63680

      -rw-r—r— 1 root root 23163 May 30 14:03 EULA.pdf

      -rw-r—r— 1 root root 536 May 30 14:03 readme.txt

      -rw-r—r— 1 root root 65179648 May 30 14:03 sps.img

3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

      # mount -o loop sps.img /mnt

```
# cd /mnt

# ./setup
```

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkAPA-<version>.noarch.rpm

   b. steeleye-lkSQL-<version>.noarch.rpm

   c. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

```
# cd /root

# umount /mnt
```

Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

```
# /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic
```

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

```
# /opt/LifeKeeper/bin/lklicmgr

License File: 20101230.lic
```

Product	Type	Expiry
LifeKeeper for Linux	Eval	27 Mar 2013 (87 days)
Apache Recovery Kit	Eval	27 Mar 2013 (87 days)
SIOS Data Replication ARK	Eval	27 Mar 2013 (87 days)
MySQL Recovery Kit	Eval	27 Mar 2013 (87 days)

Start the SIOS Protection Suite for Linux

1. Start:

```
# /opt/LifeKeeper/bin/lkstart
```

11.8.8. Configure the Cluster – Apache

Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

Important: Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously.

Access the LikeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application, or as an applet within your Java-Enabled Web Browser.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 error.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

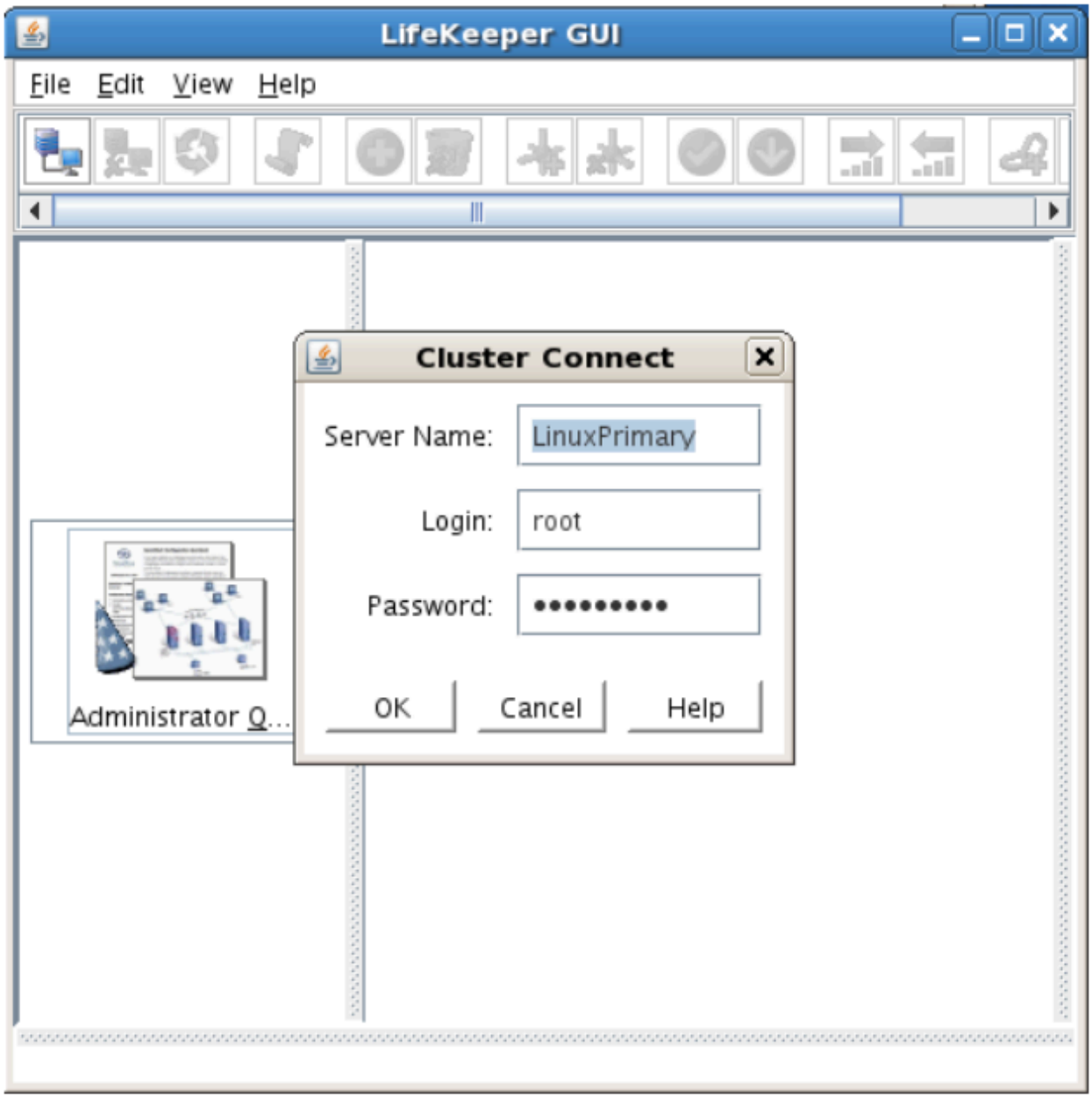
   a. /opt/LifeKeeper/bin/lkGUIapp &

3. To Connect to the LifeKeeper GUI Applet from a Web Browser, go to:

   a. http://<hostname>:81

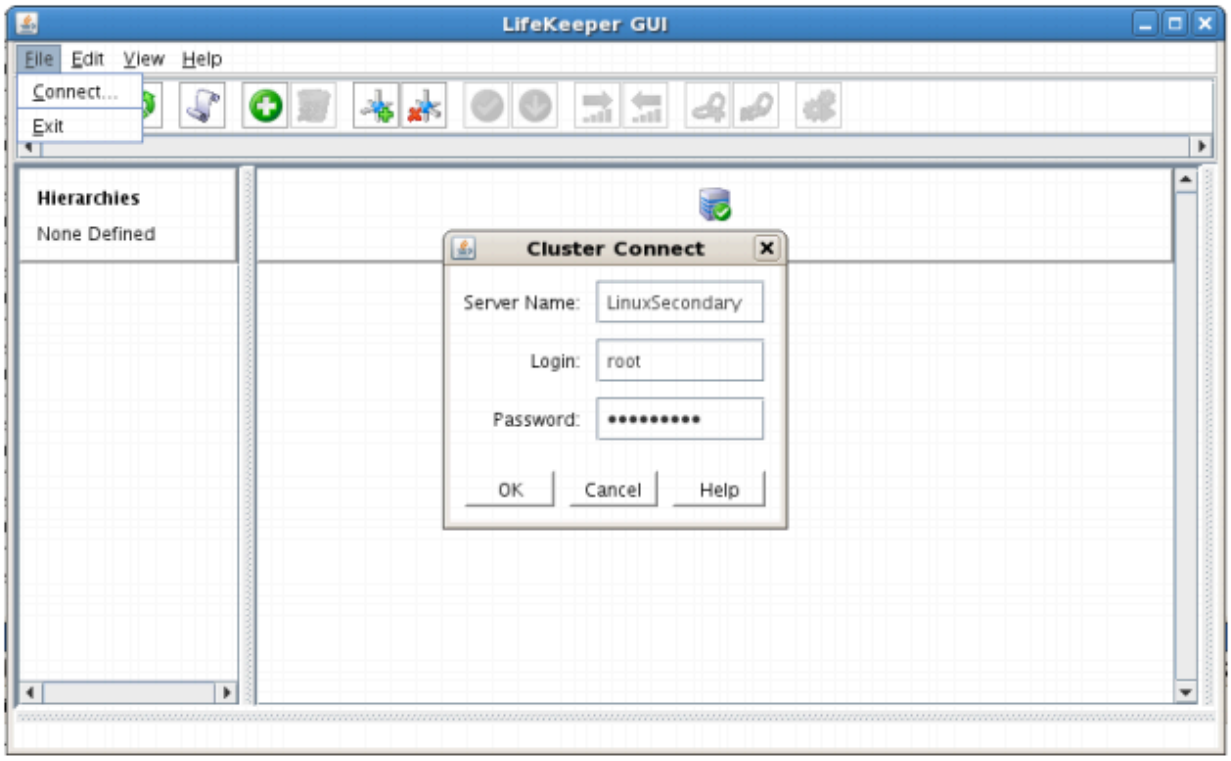
4. Enter the name of the server you wish to connect to (this field will be populated with the name of the server you are on, if you are running the GUI from a server with LifeKeeper installed) along

with your root credentials and click OK.



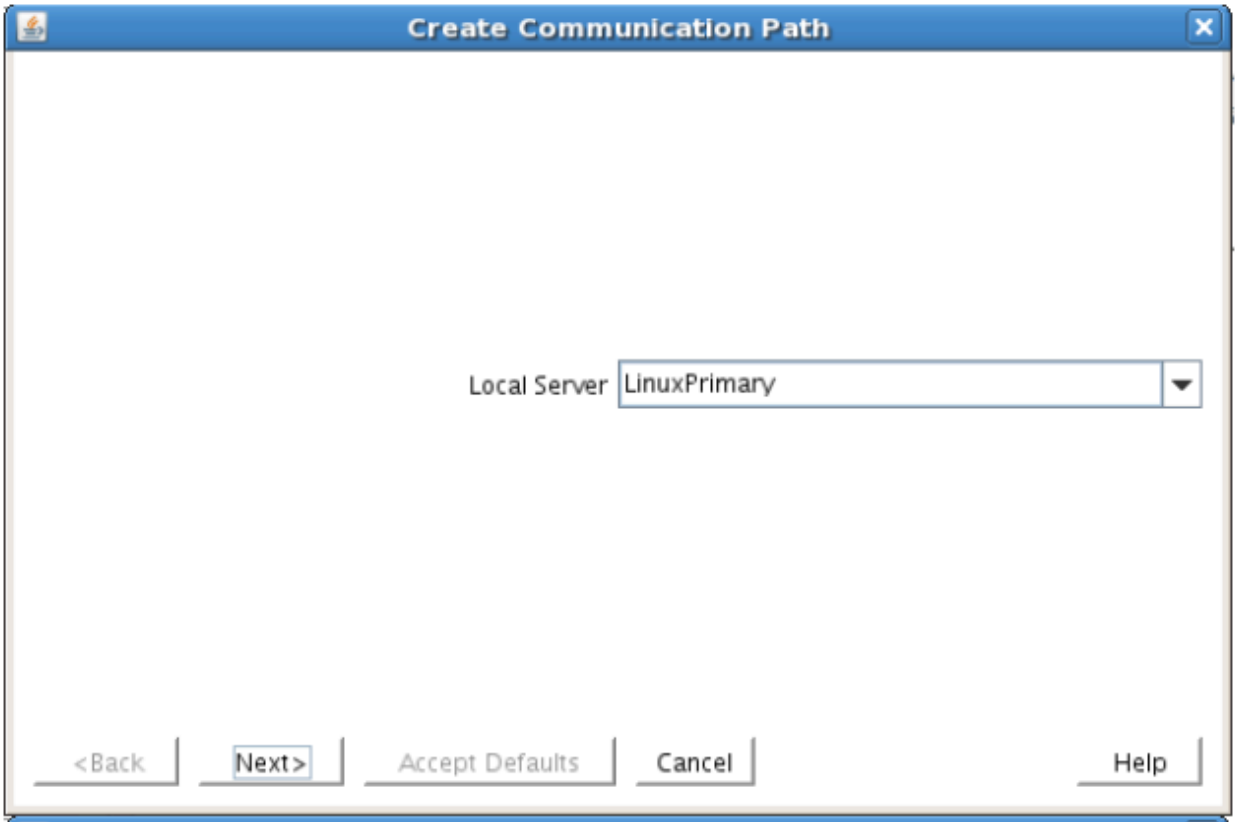
Create Communication (Comm) Paths

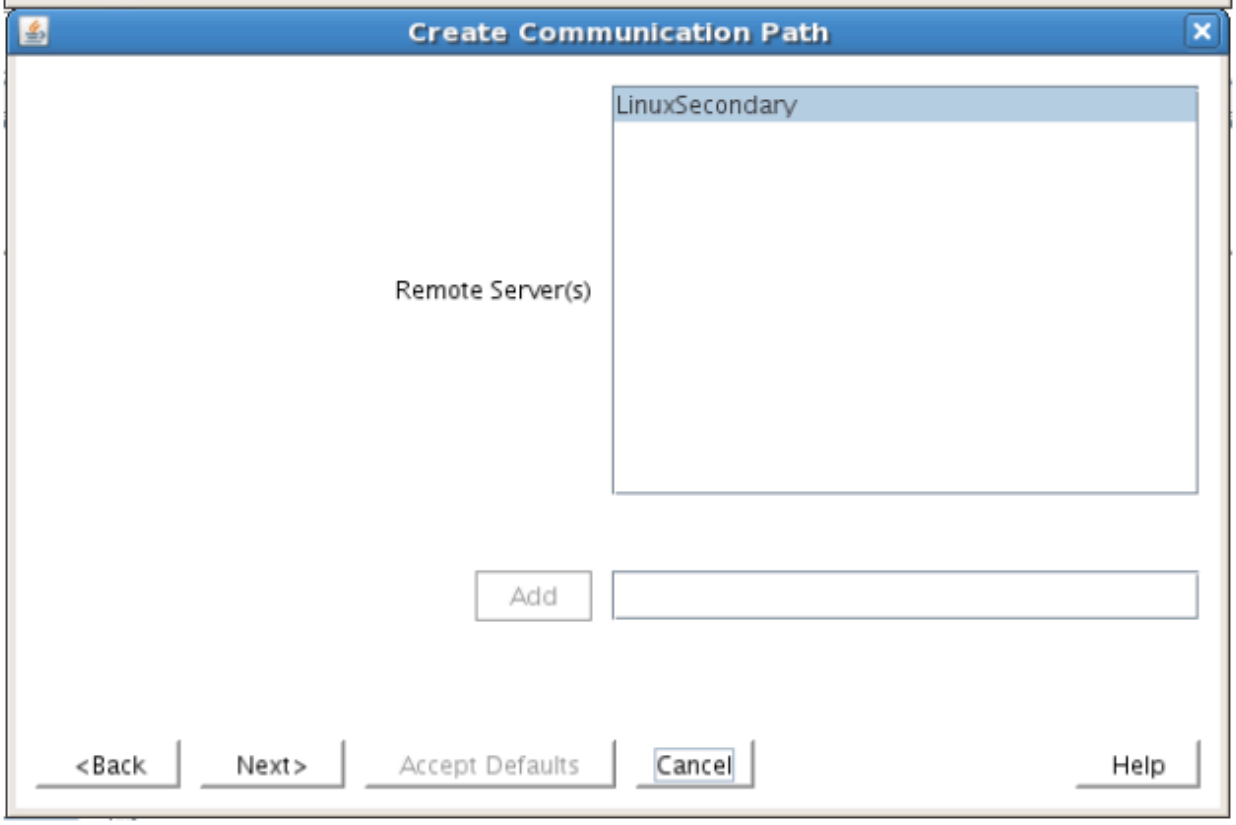
5. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



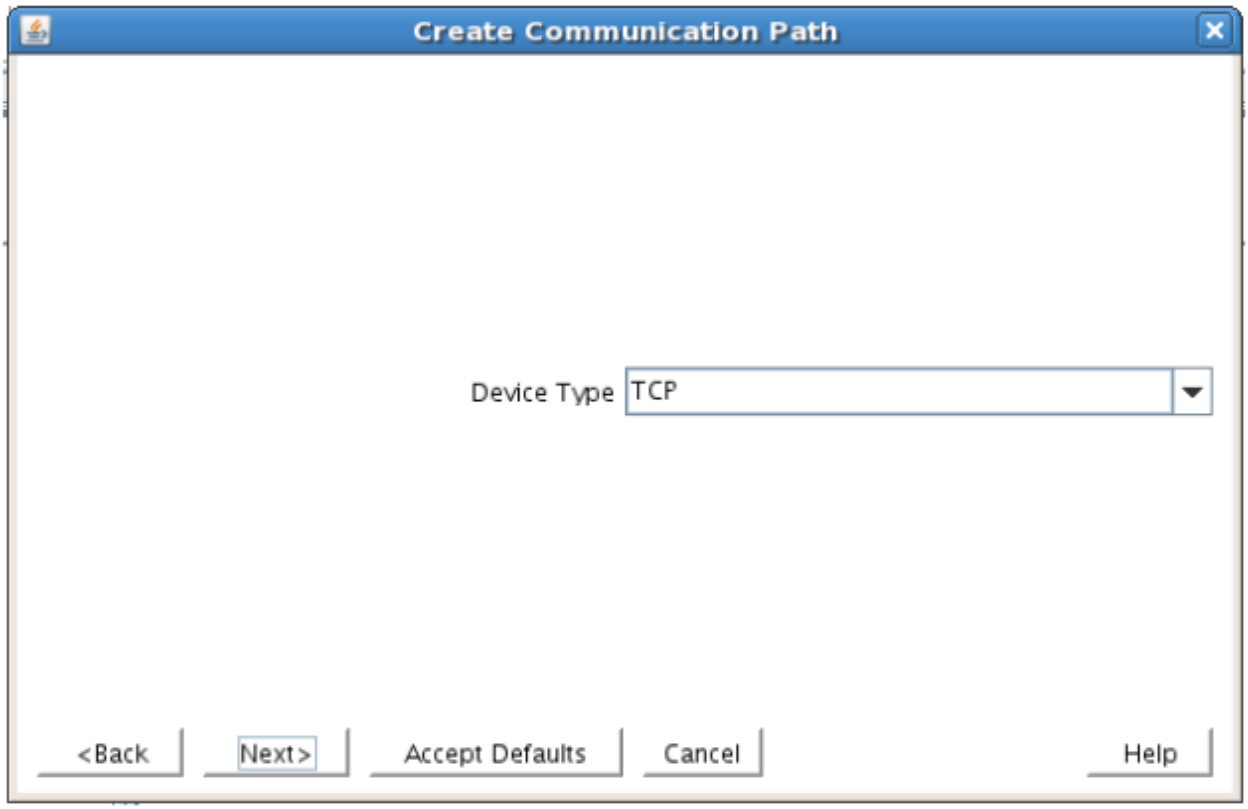
6. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

7. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

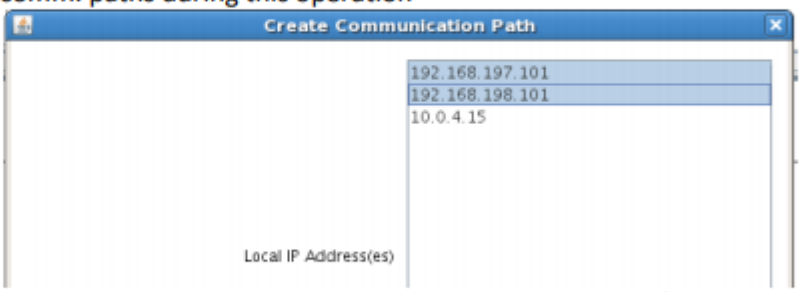


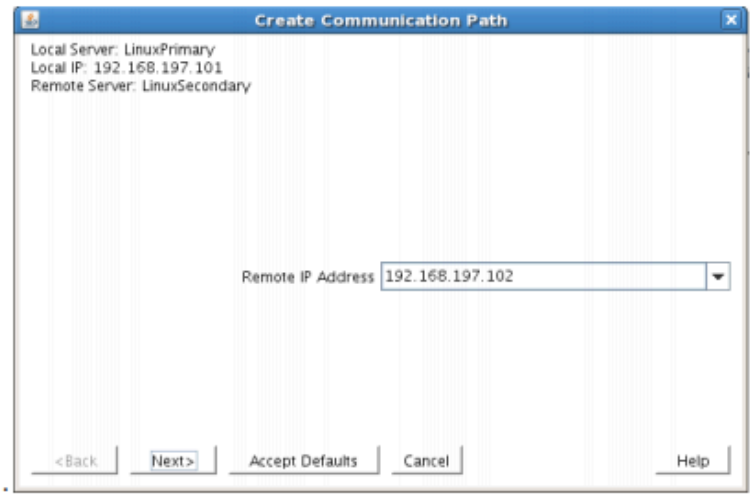


8. Select TCP for Device Type and Click Next.



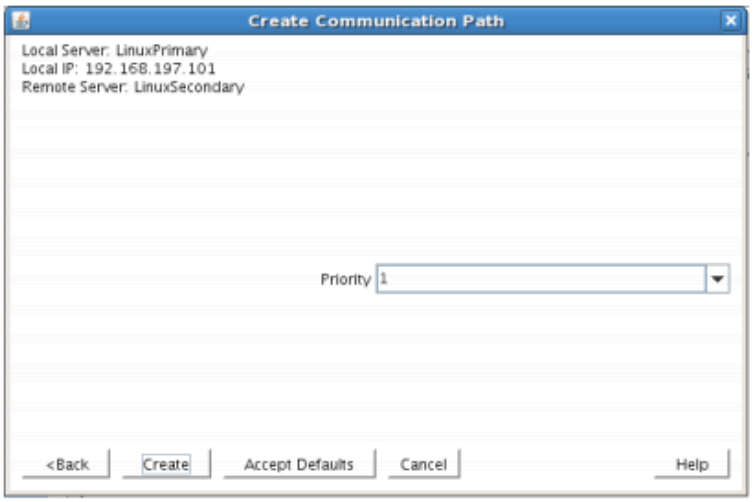
9. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

Field	Tips
For TCP/IP Comm Path...	
Local IP Address	Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation 
Remote IP Address	Choose the IP address to be used by the remote server for this comm path

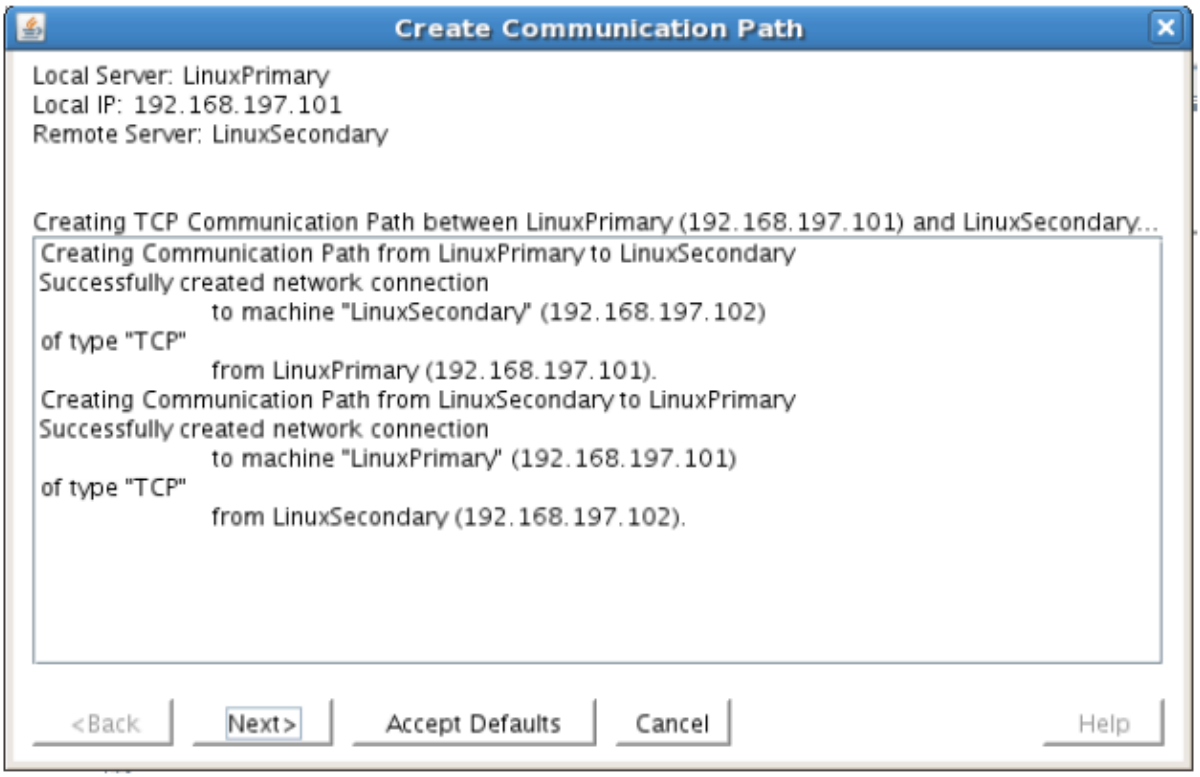


Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority



10. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



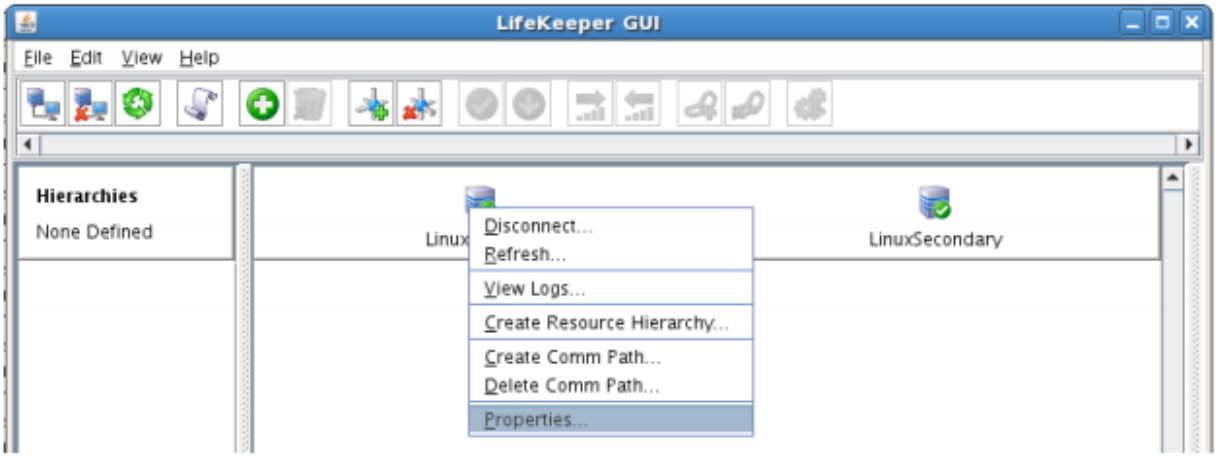
If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

11. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

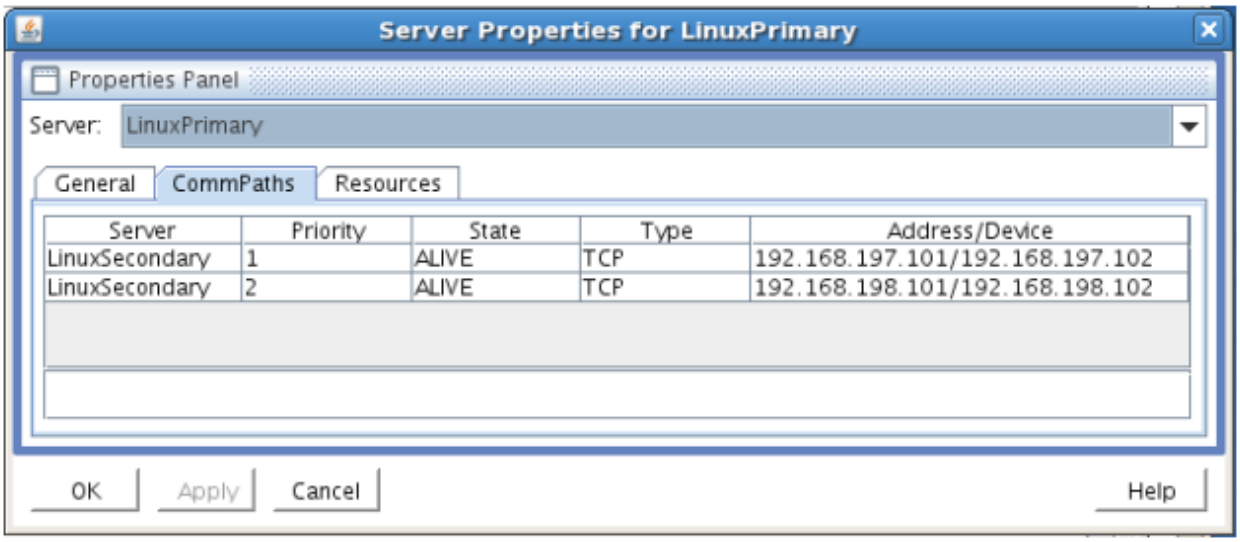
Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.



2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.

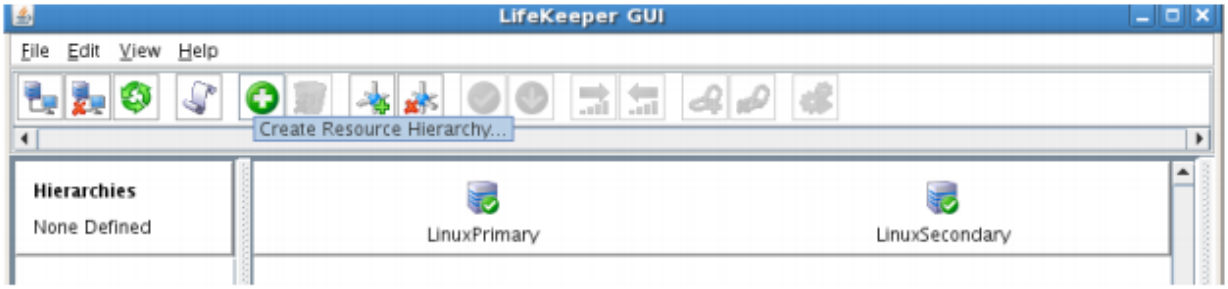


Create the LifeKeeper Hierarchy

Create and Extend an IP Resource

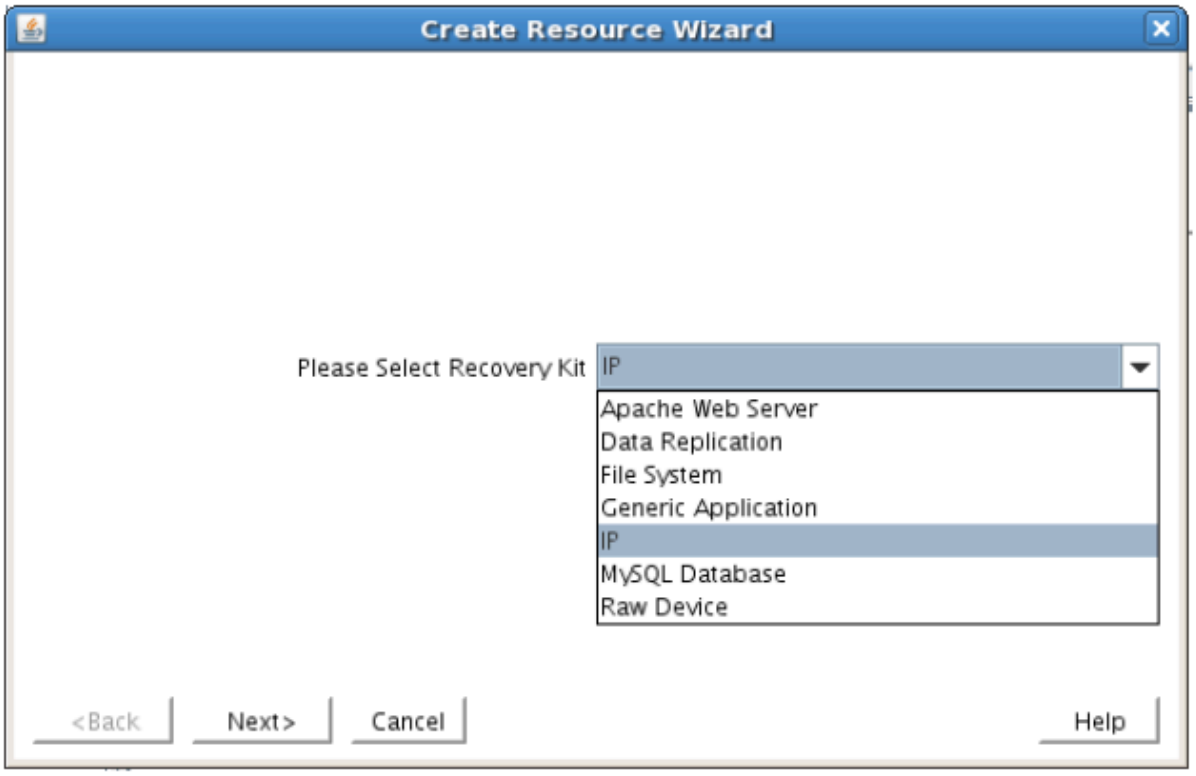
In LifeKeeper, create an IP resource and extend it to the secondary server by completing the following steps. This Virtual IP will have the ability to move between cluster nodes along the application that depends on it.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.



The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

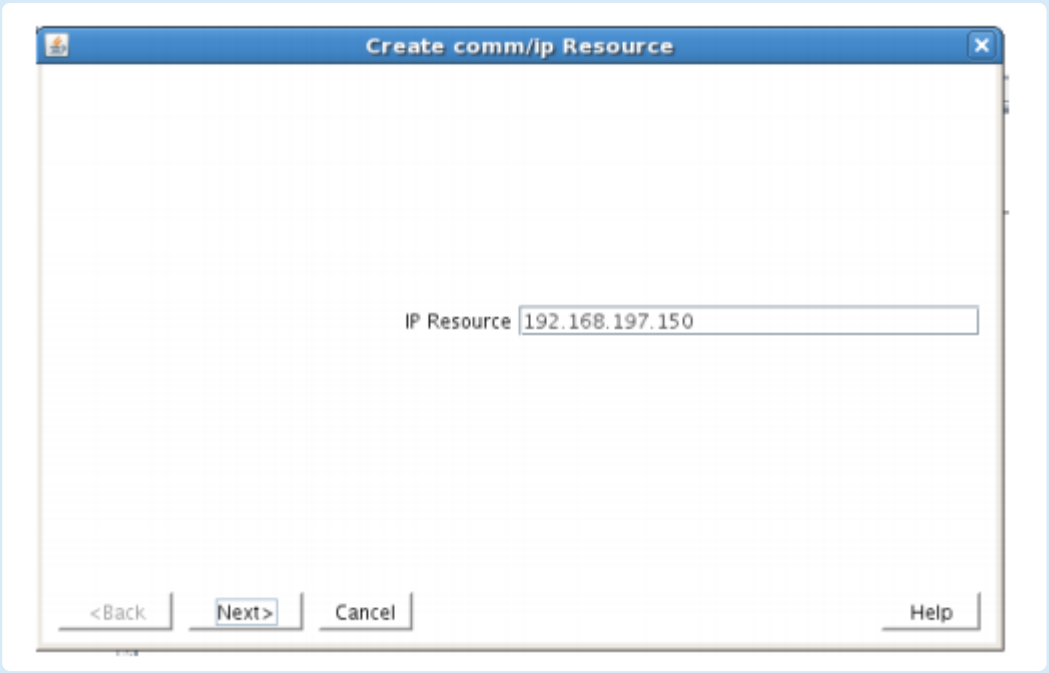
2. Select IP Address and click Next.

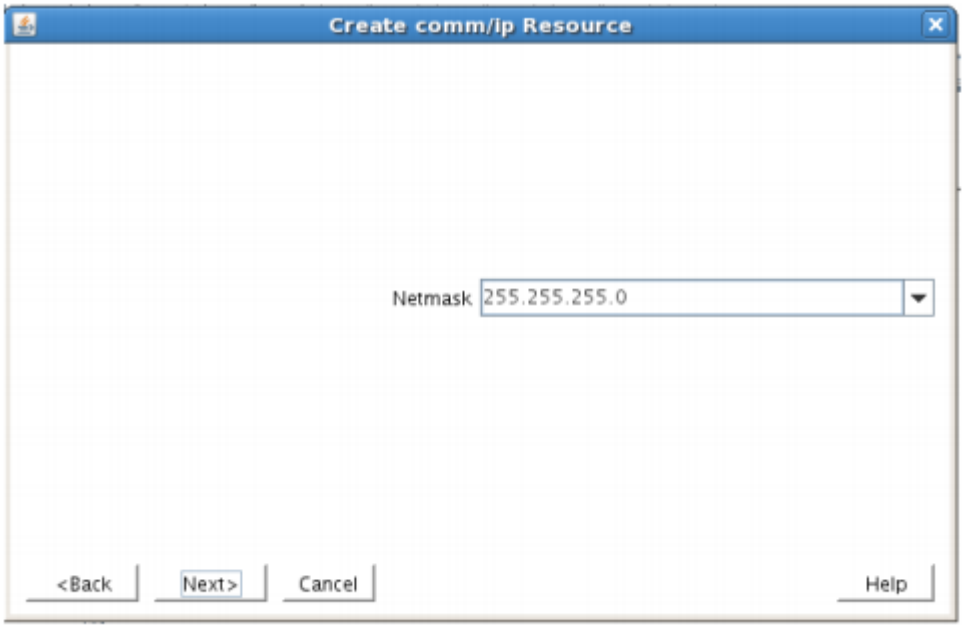


3. Enter the appropriate information for your configuration. The table below contains a list of the fields that display and additional information to assist you as you complete this procedure. Recommended values are also show below. You can also click the Help button for further information. Press Next to continue after entering the required information.

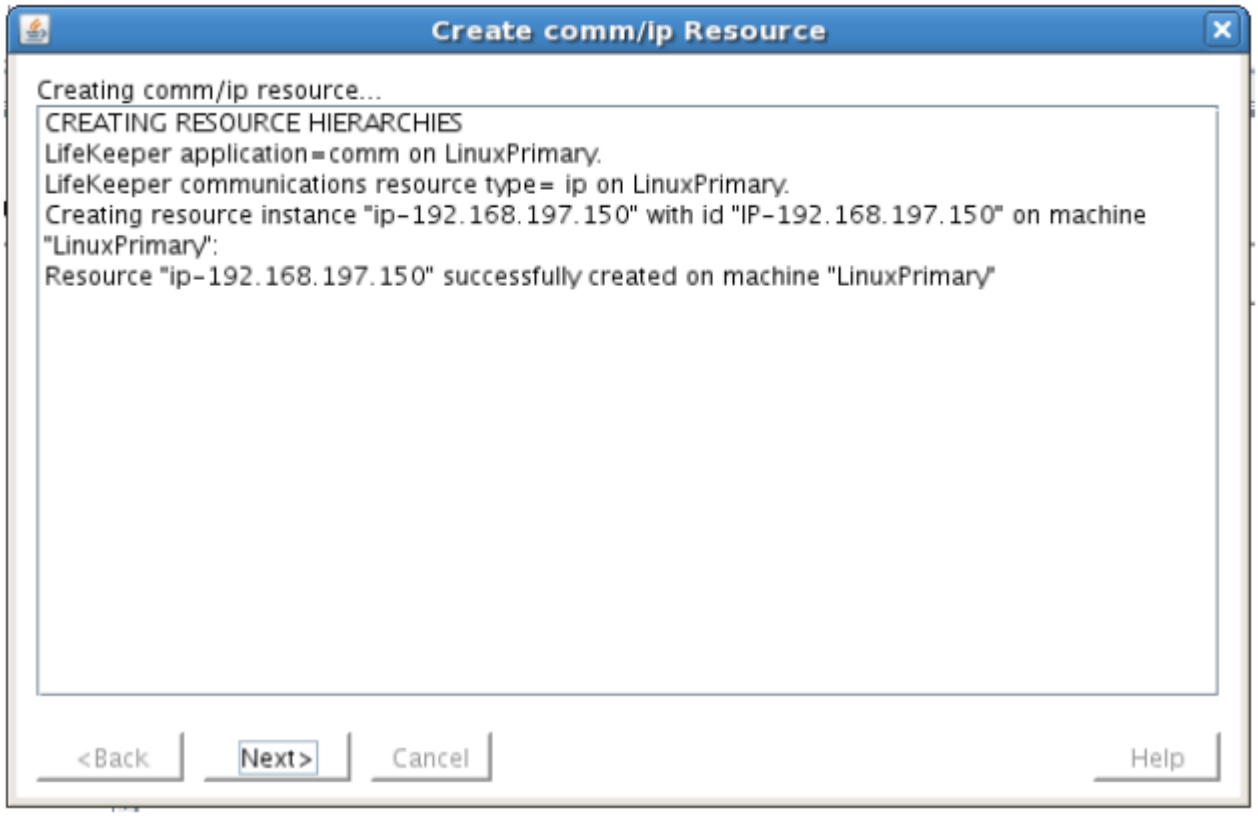
IP Creation Field Definitions

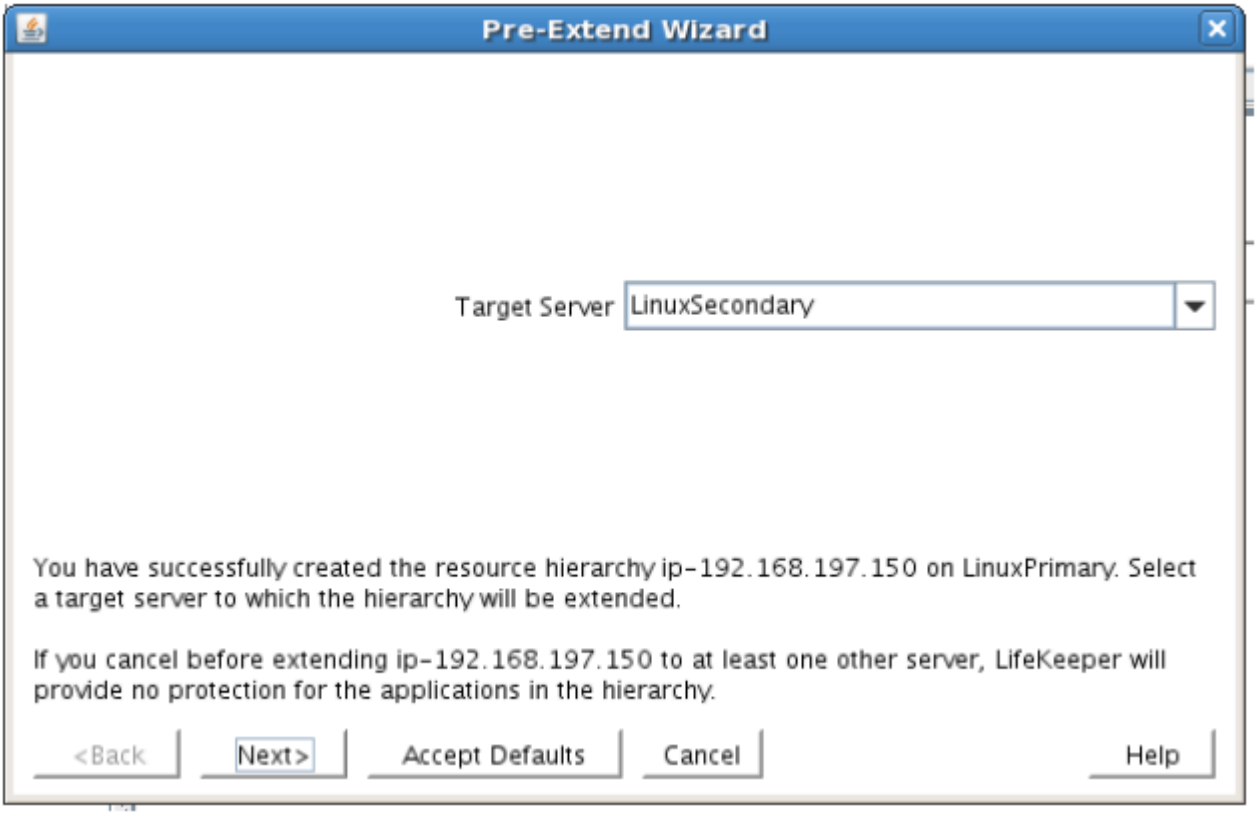
Field	Tips
Resource Type	Select IP Address as the resource type and click Next.
Switchback Type	Select Intelligent and click Next.
Server	Select the Server where the IP resource will be created. Select your Primary server and click Next.
IP Resource	Enter the virtual IP information and click Next Example **192.168.167.151** **Note** This is an IP address that is not currently in use anywhere on your network. This is the address that all clients will use to connect to the protected resources. In this configuration example, we will be protecting two (2) virtual IPs.

	First we will protect 192.168.197.150, which our Apache webserver will use.  The second time through this wizard we will protect 192.168.197.151, which will be used by MySQL
Netmask	The IP subnet mask that your TCP/IP resource will use on the target server. Any standard netmask for the class of the specific TCP/IP resource address is valid. In our sample configuration 255.255.255.0 is used for a subnet mask on both networks. **Note:** The subnet mask you choose, combined with the IP address, determines the subnet that will be used by the TCP/IP resource and should be consistent with the network configuration.

	
Network Connection	This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. Select the correct NIC and click Next.
IP Resource Tag	Accept the default value and click Next. This value only affects how the IP is displayed in the GUI. The IP resource will be created on our Primary server.

4. LifeKeeper will create and validate your resource. After receiving the message that the resource has been created successfully, click Next when the following dialog box appears so that you can complete the process of Extending the IP Resource to our Secondary server, below.





Extending the IP resource will start automatically after you have finished creating an IP address resource if you clicked Next in the dialog box displayed above. You can also start this from an existing IP address resource by right clicking on the active resource and selecting Extend Resource Hierarchy.

Refer to the table below to complete the Extend IP Resource procedure.

Field	Recommended Entries or Notes
Switchback Type	Leave as “intelligent” and click Next
Template Priority	Leave as default (1)
Target Priority	Leave as default (10)
Network Interface	This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. The correct physical NIC should be selected by default. Please verify and then click Next
IP Resource Tag	Leave as default.
Target Restore Mode	Select Enable and click Next.
Target Local Recovery	Select Yes to enable Local Recovery for the SQL resource on the Target server.
Backup Priority	Accept the default value.

5. After receiving the message Hierarchy extend operations completed, click Finish and then click Done

6. Your IP resource (192.168.197.151) is now fully protected and has the ability to “float” between cluster nodes as needed. Looking at the LifeKeeper GUI you will notice that the IP resource is Active on the Primary cluster node and Standby on the Secondary cluster node

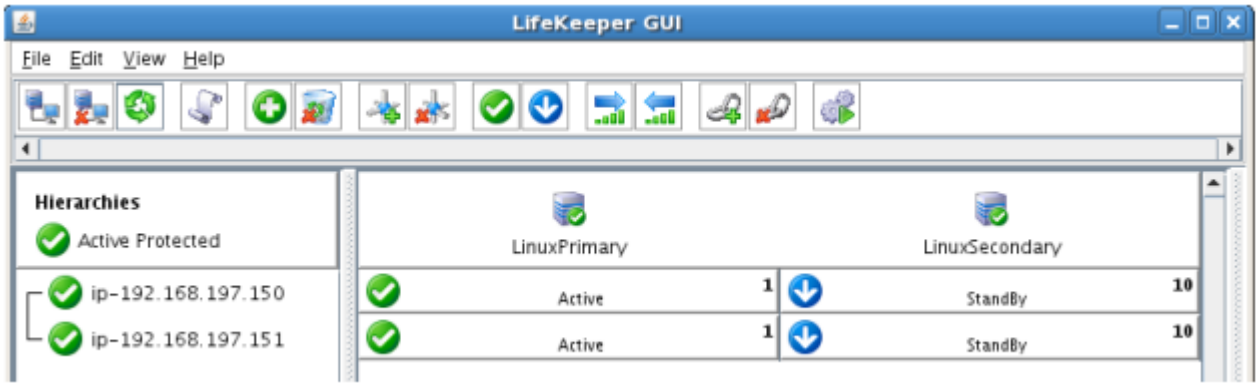
Create a Second IP Resource

Repeat the procedure above to protect a 2nd IP resource.

This second time, protect 192.168.197.151, which is the IP address our MySQL database will later use.

As a result, your LifeKeeper GUI will display as follows, with both IP resources Active and protected on

the Primary cluster node:



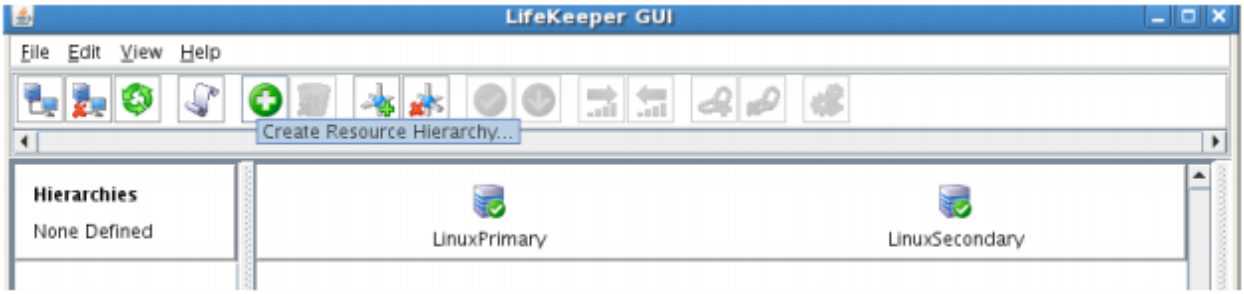
Create a Mirror and Begin Data Replication

In this section we will setup and configure the Data Replication resource, which be used to synchronize our Apache Webserver's data between cluster nodes. The data we will replicate resides in the /var/www partition on our Primary cluster node

Please note:

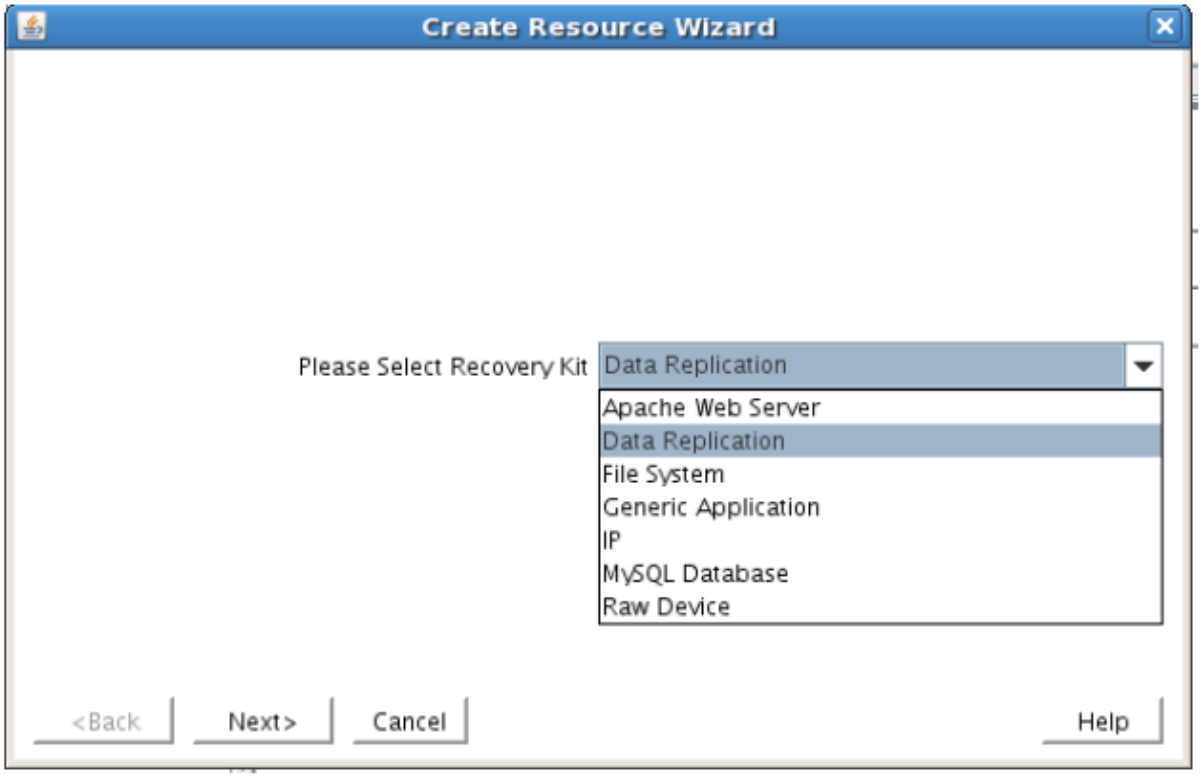
- The source volume to be replicated must be mounted on the Primary server
- The target volume, which will received replicated data, must NOT be mounted on the Secondaryserver.
- The target volume's size must equal to or larger than the size of its source volume.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.



The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

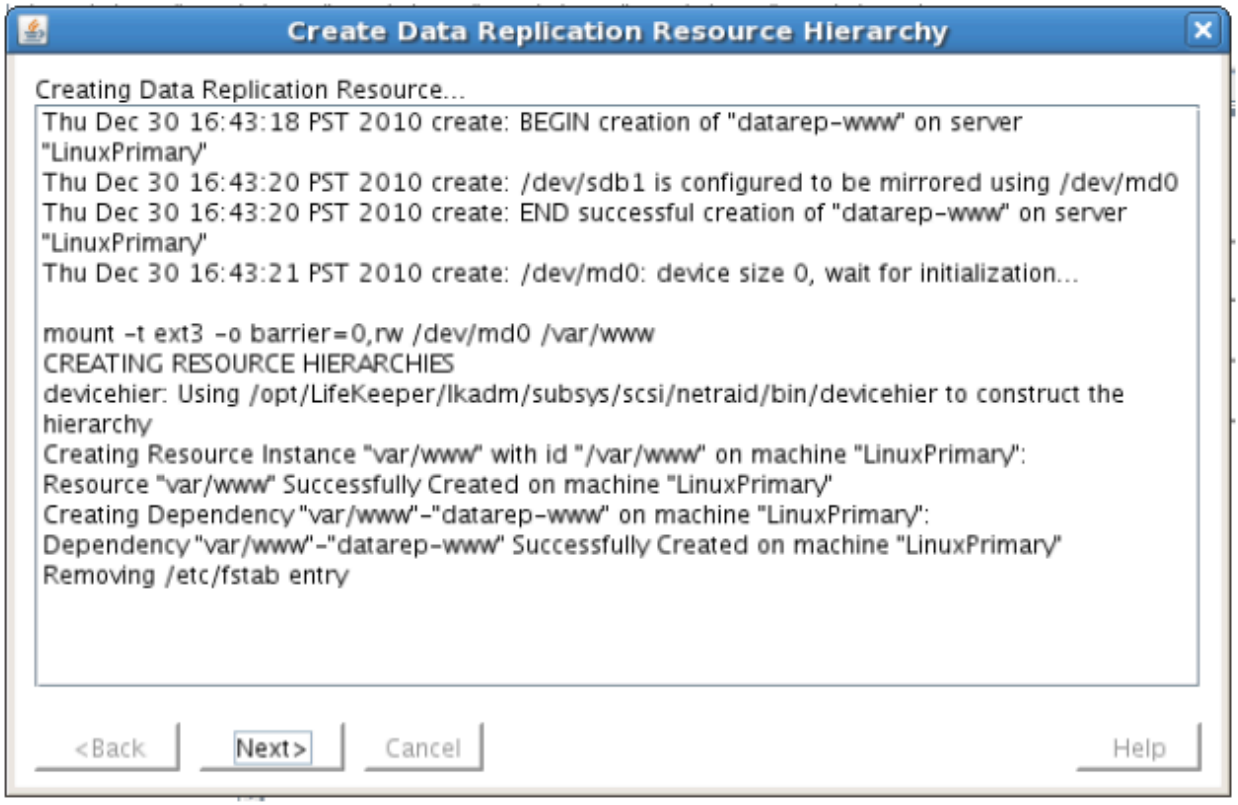
2. Select Data Replication and click Next.



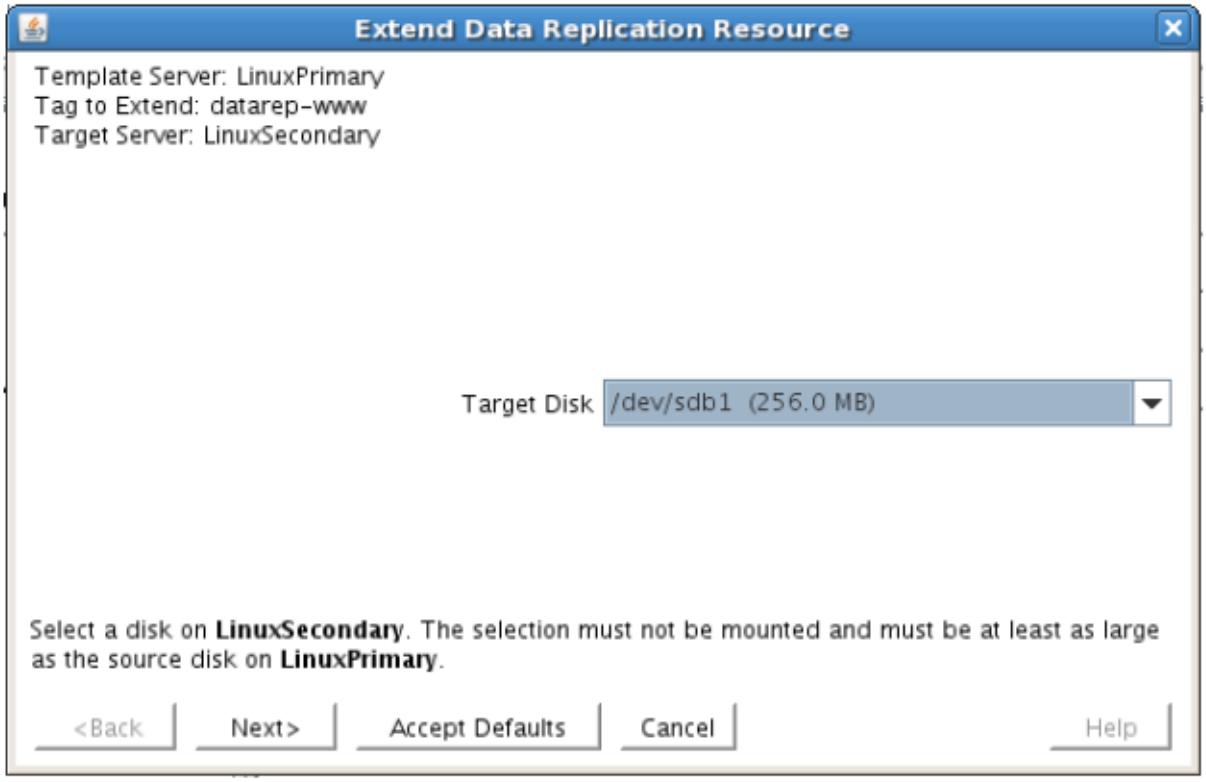
3. Follow the Data Replication wizard, and enter the following values:

Field	Recommended Entries or Notes
Switchback Type	Intelligent
Server	LinuxPrimary (Primary Cluster Node, i.e. Mirror Source)
Hierarchy Type	Select: “**Replicate Existing Filesystem**”
Existing Mount Point	At this step you will select the mounted partition to replicate. In our example, select “/var/lib/mysql”
Data Replication Resource Tag	Leave as default
File System Resource Tag	Leave as default
Bitmap File	Leave as default (Note: if using high speed SSD storage you will want to create a small partition and use it for bitmap placement, i.e. /bitmaps)
Enable Asynchronous Replication	Leave as default (Yes)

4. Click Next to begin creation of the Data Replication resource hierarchy. Status will be displayed in the GUI as follows:

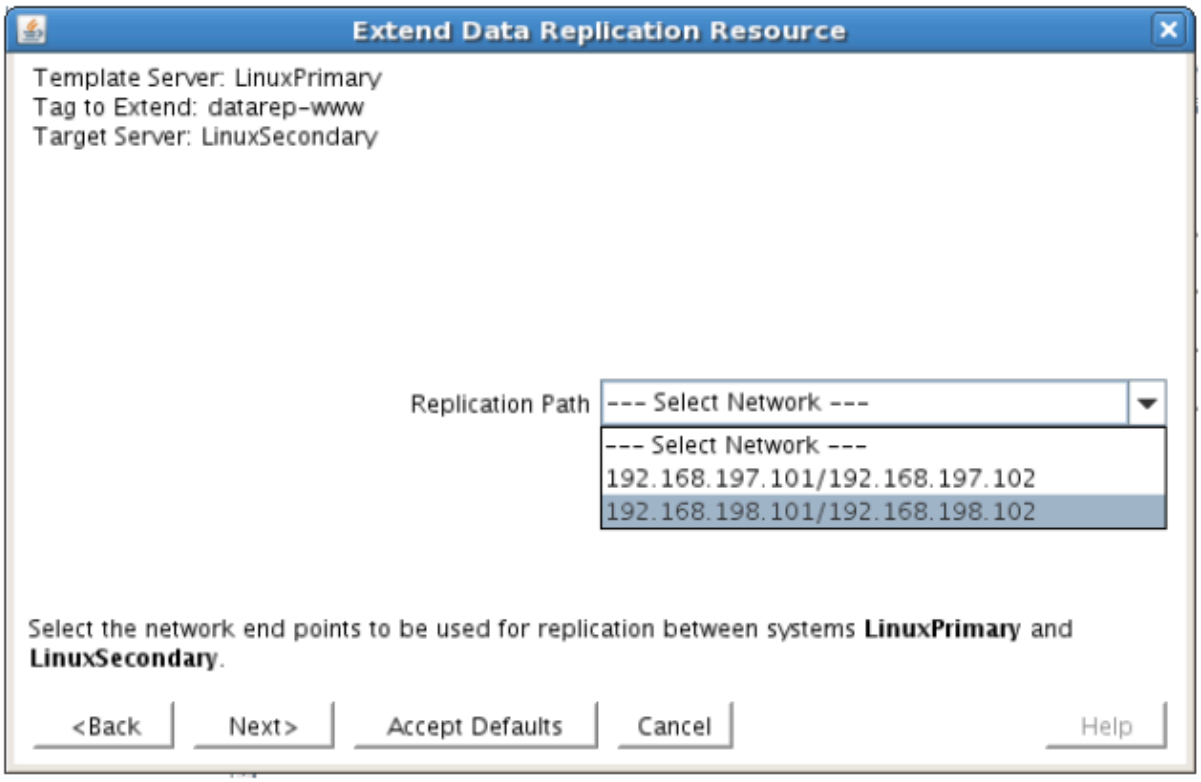


5. Click Next to begin the process to Extend the Data Replication Resource. Select all default settings. When it asks for the target disk, select a free partition on your Target server which is the same size (or greater) than the Source Volume we are replicating. This partition should NOT be mounted on the Target system.



6. Continue through the wizard, and you will be prompted to select the network you would like

replication to take place over. In general, it's a best practice to separate your user/application and your replication traffic. In our example setup we will replicate over our backend network, 192.168.198.X



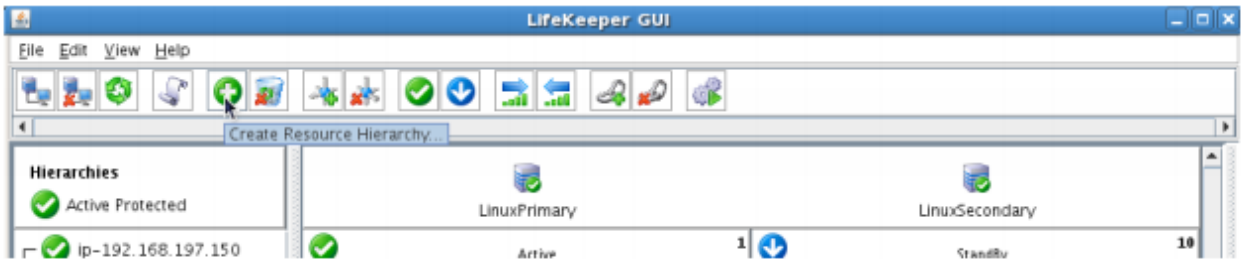
7. Click Next and continue through the wizard. Once completed, your resource hierarchy will look as follows

Create the Apache Hierarchy

In this section we will create an Apache resource hierarchy on the primary server and extend it to the backup server. This step will create a dependency on the IP resource created in previous the step.

Important: The Apache web server should not be running at this time.

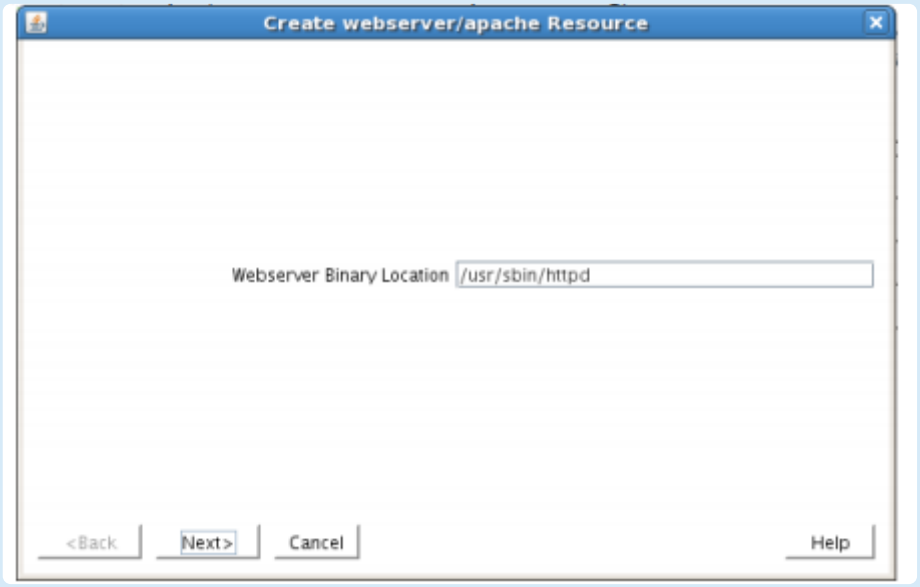
1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.

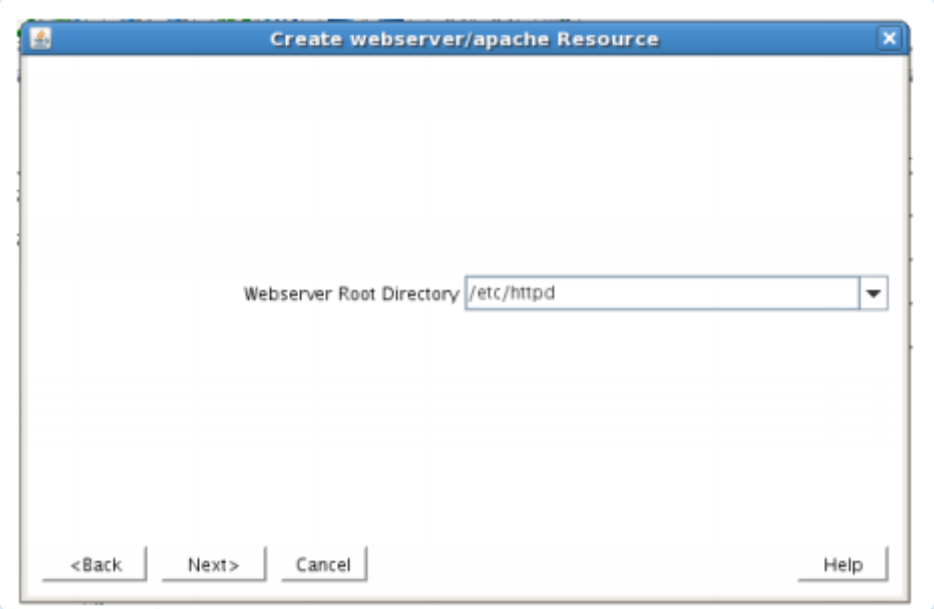


   The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select Apache Web Server and click Next.

3. Proceed Through the Resource Creation wizard, providing the following values

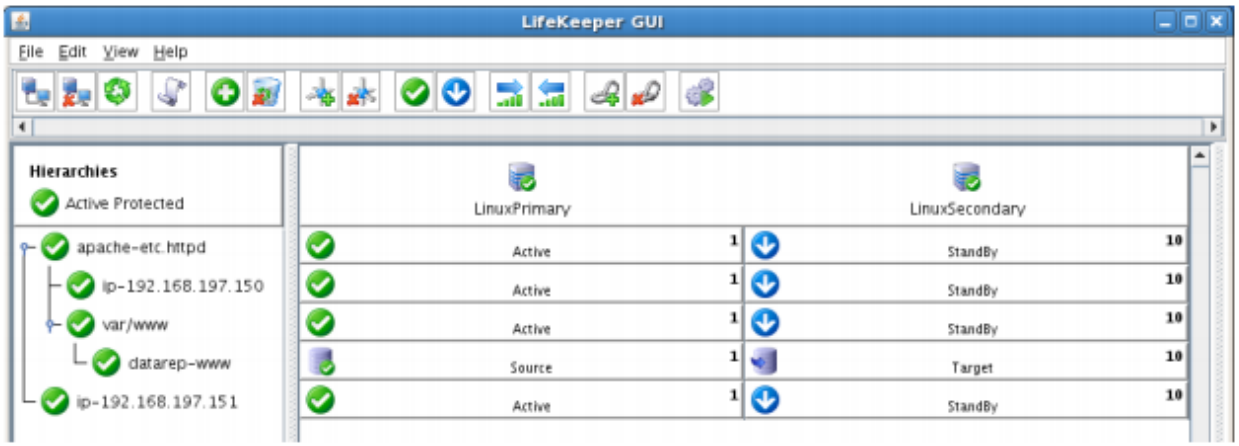
Field	Recommended Entries or Notes
Switchback Type	Intelligent
Server	LinuxPrimary (Primary Cluster Node, i.e. Mirror Source)
Webserver Binary Location	/usr/sbin/httpd (assumes a standard apache config) 
Webserver Root Directory	/etc/httpd (assumes a standard apache config)

	
Root Tag	Leave as default

4. Click “Create” to begin resource hierarchy creation on the primary server. Once complete, click “Next” to extend this resource to the secondary server.

5. During the Extend Resource wizard, leave all settings as default.

6. Note: during the resource creation process, LifeKeeper extracted the existing configuration of the existing Apache webserver, and identified that it depends on the IP resource (192.168.197.150) and the Data Replication resource (/var/www) that were created in previous steps. These resources now appear underneath the newly created Apache resource, to indicate the dependency relationship.



Create the Shared Filesystem Resource Hierarchy

Create a Filesystem resource to protect the shared iSCSI filesystem and make it high available between cluster nodes. SPS for Linux leverages SCSI Persistent Group Reservations (PGR) to lock the LUN,

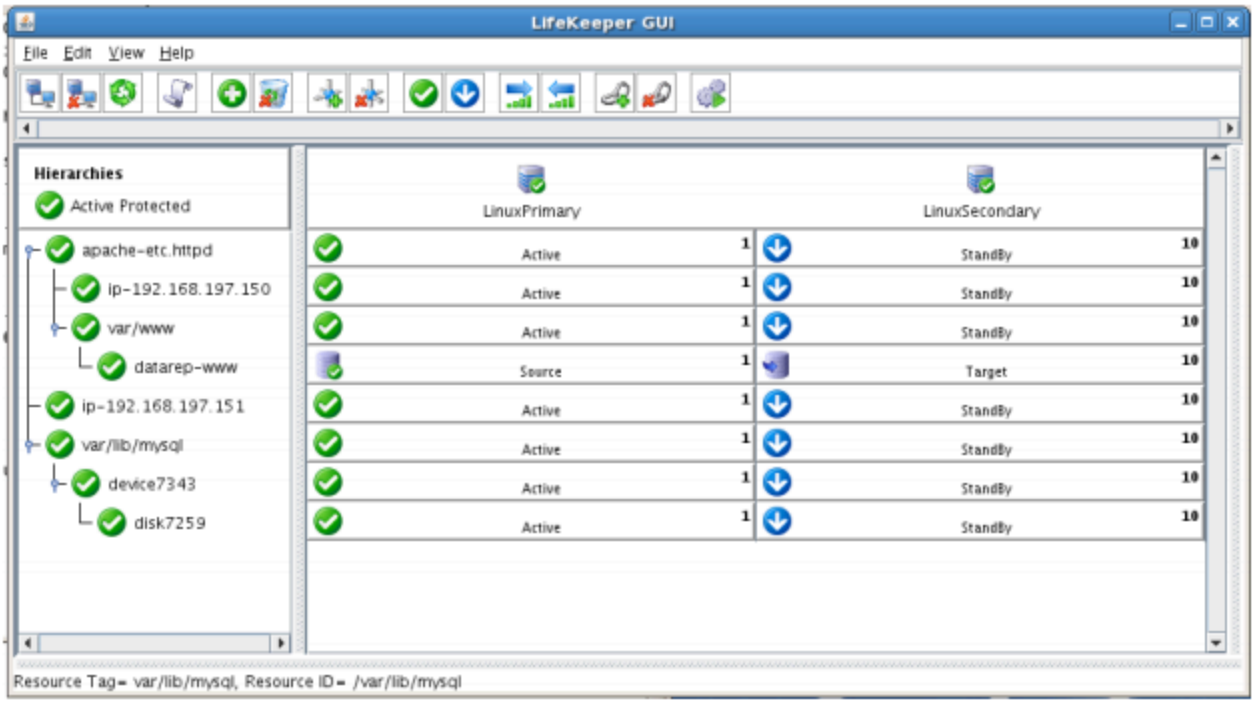
ensuring that only the active cluster node for the storage resource can access it.

> **Important:** At this point, the shared iSCSI LUN needs to already be mounted on the Primary Server. It should NOT be mounted on the Secondary Server. See section titled “Configure iSCSI initiator, discover and login to iSCSI target” above to review the steps involved.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select File System and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

Field	Recommended Entries or Notes
Switchback Type	Intelligent
Server	LinuxPrimary (Primary Cluster Node)
Mount Point	Select /var/lib/mysql. Note that LifeKeeper scans the system for LUNS that are sharable between cluster nodes. The list of possible shared LUNS is presented automatically in this step of the wizard.

4. Select Create Instance to define this resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the File System resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Your resource hierarchy should look as follows:



Create the MySQL Resource Hierarchy

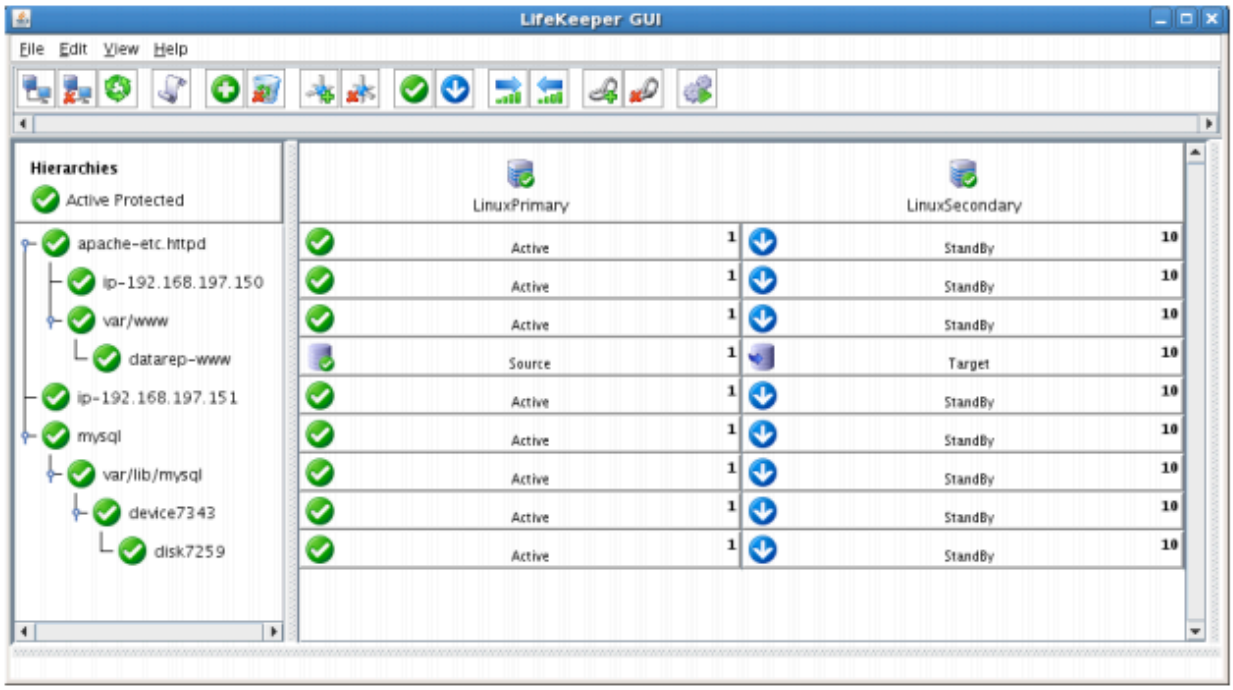
Create a MySQL resource to protect the MySQL database and make it high available between cluster nodes.

> **Important:** At this point, MySQL needs to be already running on the Primary Server. It should NOT be running on the Secondary Server. See section titled “Install, Configure, and Start MySQL” above to review the process to configure and start MySQL as needed.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select **MySQL Database** and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

Field	Recommended Entries or Notes
Switchback Type	Intelligent
Server	LinuxPrimary (Primary Cluster Node)
Location of my.cnf	Enter “/var/lib/mysql”. Note that earlier in the MySQL configuration process we created a my.cnf file in this directory.
Location of MySQL executables	Leave as default (/usr/bin) since we are using a standard MySQL install/ configuration in this example
Database tag	Leave as default

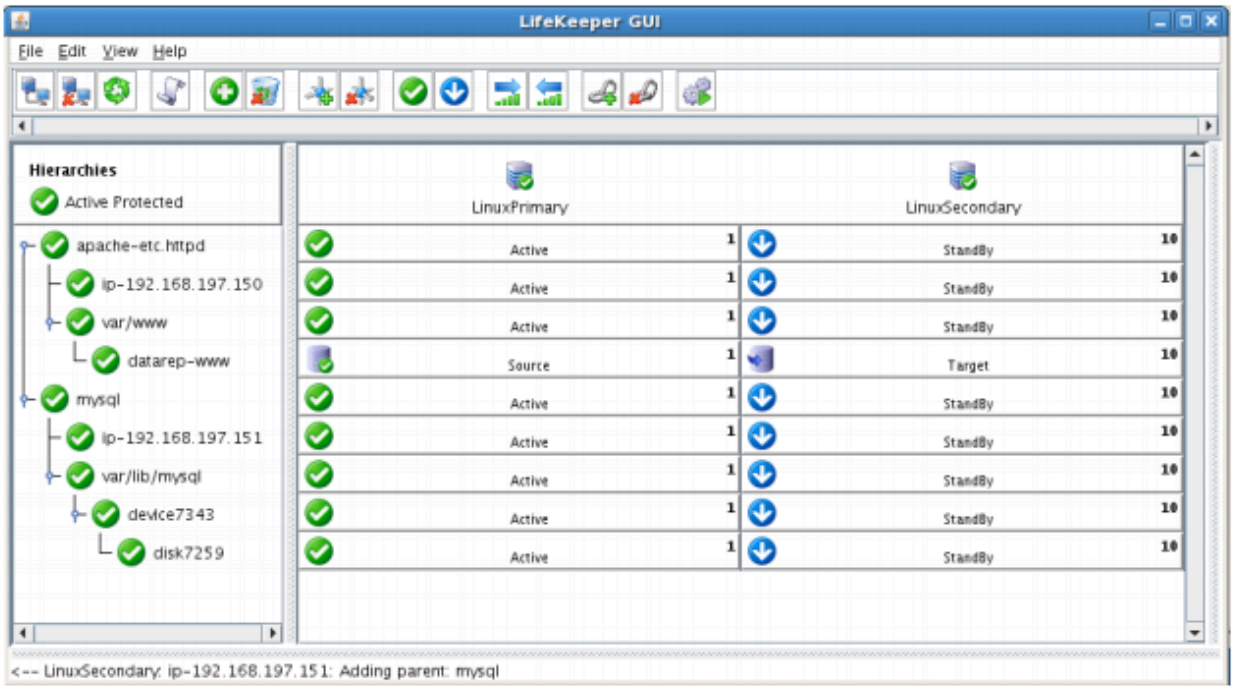
4. Select Create to define the MySQL resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the MySQL resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Note: LifeKeeper will automatically identify that the MySQL resource has a dependency on the FileSystem resource (/var/lib/mysql). The Filesystem Resource will appear underneath the MySQL resource in the GUI
9. Your resource hierarchy should look as follows:



Create the MySQL IP Address Dependency

In this step will define an additional dependency: that MySQL depends on a Virtual IP (192.168.197.151) so that the IP address follows the MySQL database should it move. This IP (.151) is the IP the webserver will use to access the MySQL database.

1. From the LifeKeeper GUI toolbar, right-click on the “mysql” resource
2. Select “Create Dependency” from the right-click context menu
3. In the Child Resource Tag dropdown menu, select “ip-192.168.197.151”
4. Click Next
5. Click Create Dependency
6. Click Done
7. The Virtual IP address resource (192.168.197.151) will now appear underneath the MySQL resource in the LifeKeeper user interface. This ensures that resources move together, and are started/stopped in the proper order.
8. Your resource hierarchy should look as follows



At this point in the Evaluation, we have fully protected Apache, MySQL, and their dependent resources: IP addresses, and Storage, both shared and replicated.

11.8.9. Test Your Environment – Apache

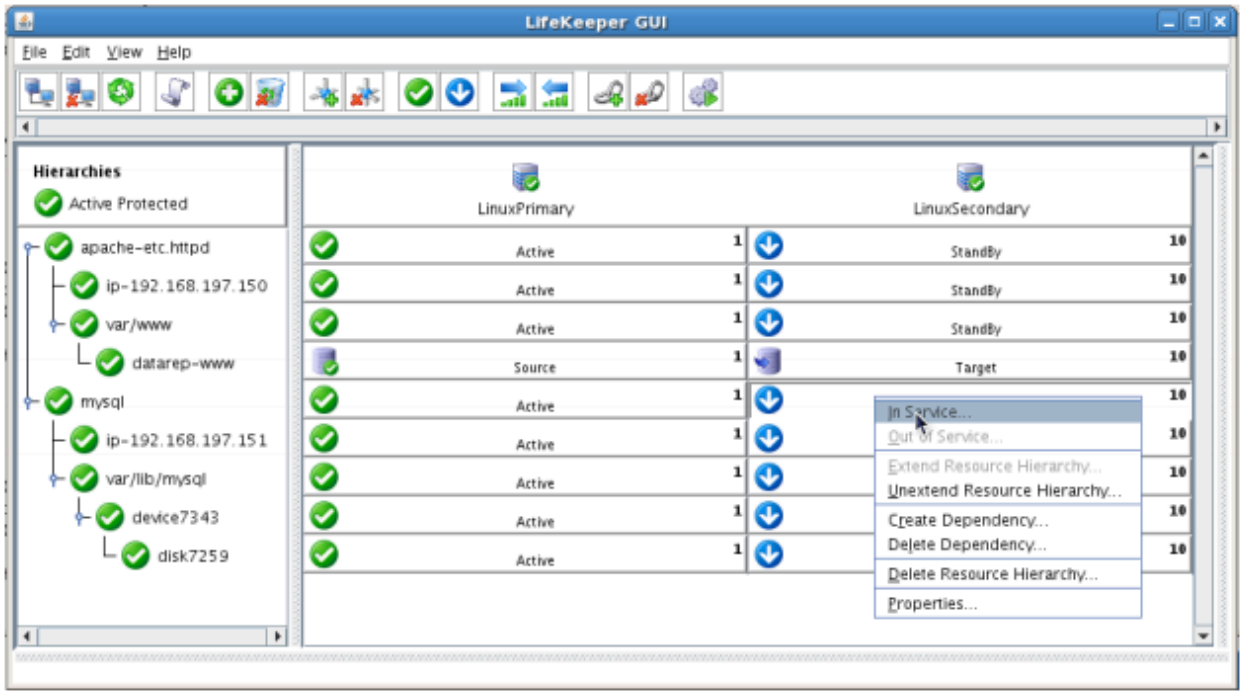
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

> **Note:** For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

Manual Switchover of the MySQL Hierarchy to Secondary Server

Procedure:

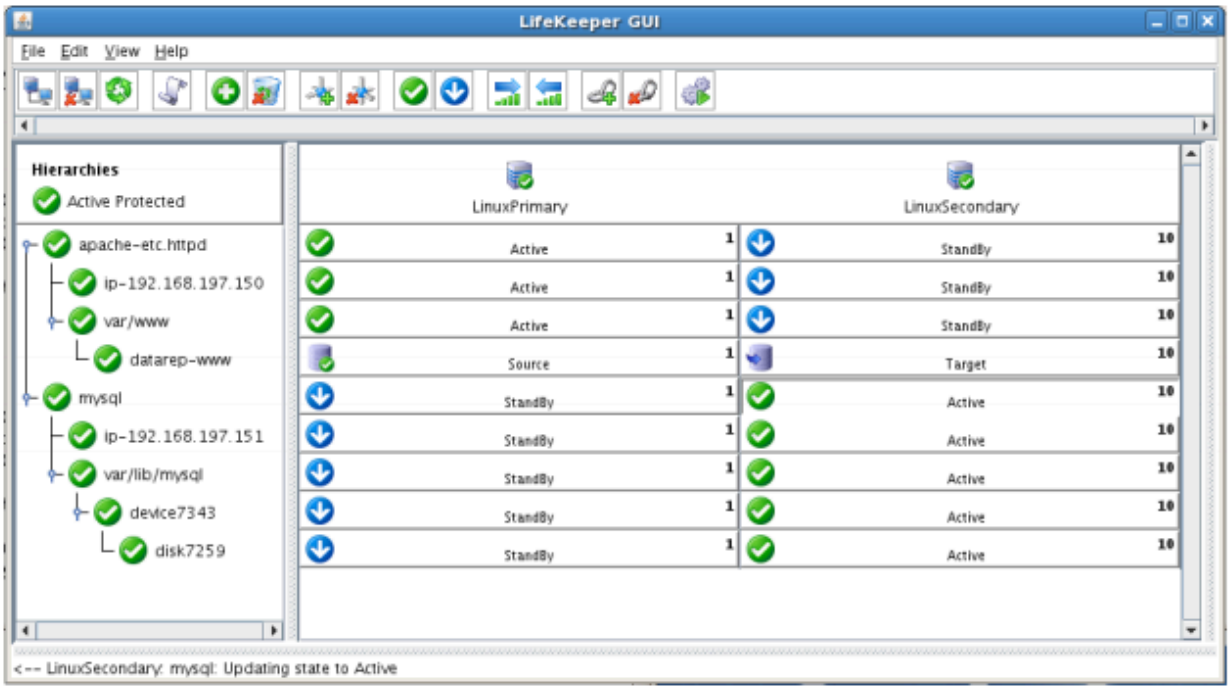
- From the LifeKeeper GUI, right click on the MySQL resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



Expected Result:

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXSECONDARY
- At this point, we now have an “Active/Active” cluster because both cluster nodes are actively

running resources.



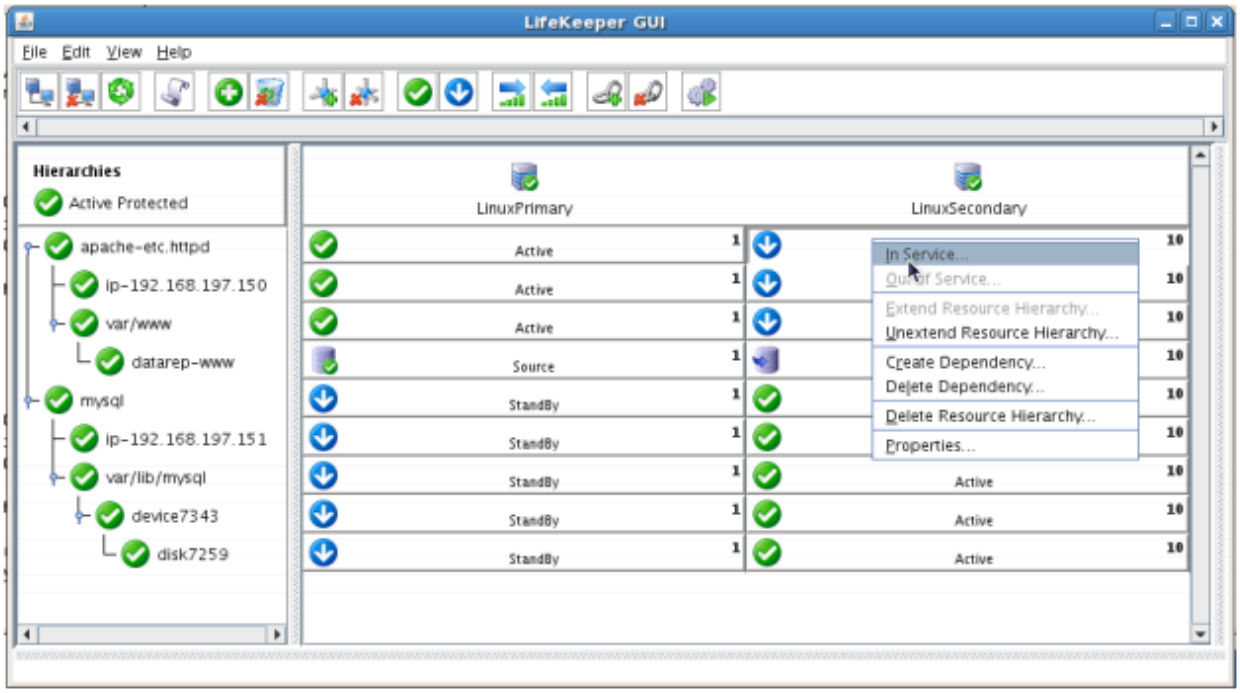
Tests/Verification:

- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXSECONDARY.
- Run “ifconfig –a” on LINUXSECONDARY to validate that the IP Address 192.168.197.151 is active on LINUXSECONDARY
- Run “df –h” to verify that the /var/lib/mysql shared iSCSI filesystem is mounted on LINUXSECONDARY
- Verify the MySQL services are running on LINUXSECONDARY by running “ps –ef | grep –i mysql”
- On LINUXSECONDARY run the following command to verify client connectivity to the MySQL database:
  - # mysql –S /var/lib/mysql/mysql.sock –u root –p
  - (enter password “SteelEye”)
- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXPRIMARY, run “mount /dev/sdc1 /var/lib/mysql”. This should FAIL because LINUXPRIMARY does not own the SCSI reservation on this LUN.

Manual Switchover of the Apache Hierarchy to Secondary Server

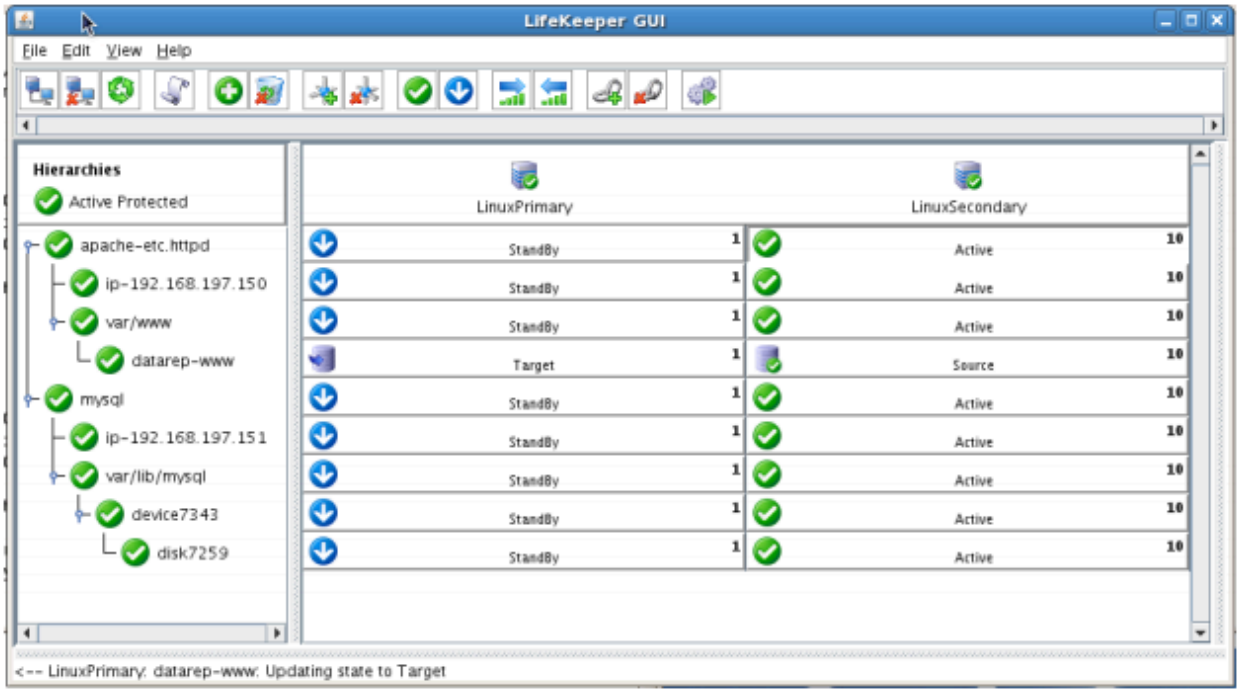
Procedure:

- From the LifeKeeper GUI, right click on the Apache resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



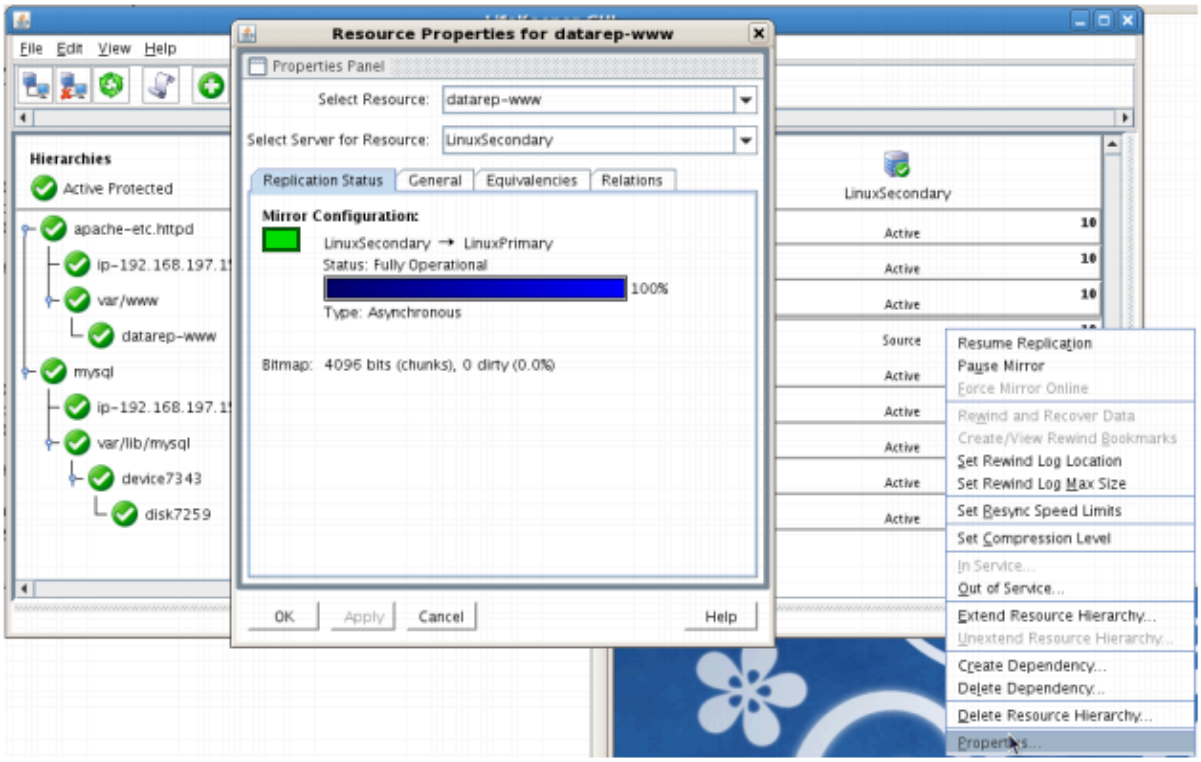
Expected Result:

- Beginning with the Apache resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXSECONDARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXSECONDARY -> LINUXPRIMARY
- At this point, we now back to an “Active/Passive” cluster because all services are now actively running on LINUXSECONDARY

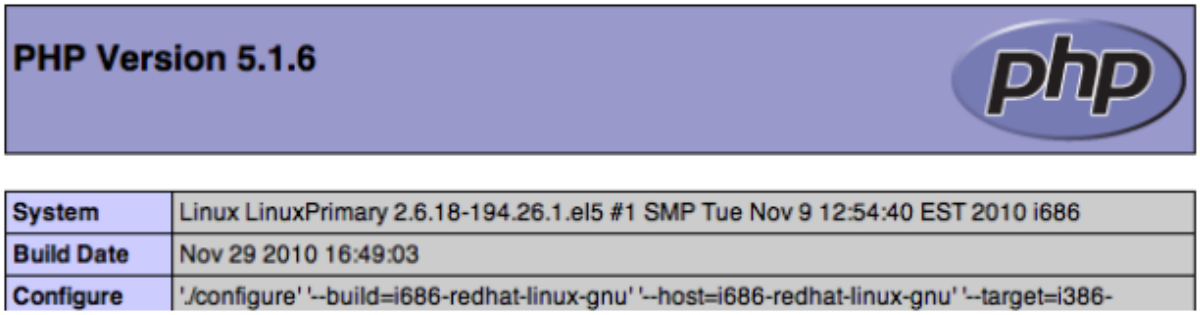


Tests/Verification:

- Using the LifeKeeper GUI, verify that the Apache and dependent resources are active on LINUXSECONDARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-www” resource and select Properties



- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.150 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/www replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY
- Verify the Apache services are running on LINUXPRIMARY by running “ps –ef | grep –i httpd”
- Open a Web Browser to http://192.168.197.150 and verify that it can successfully connect. The PHPInfo output should indicate that the system name is “LinuxPrimary



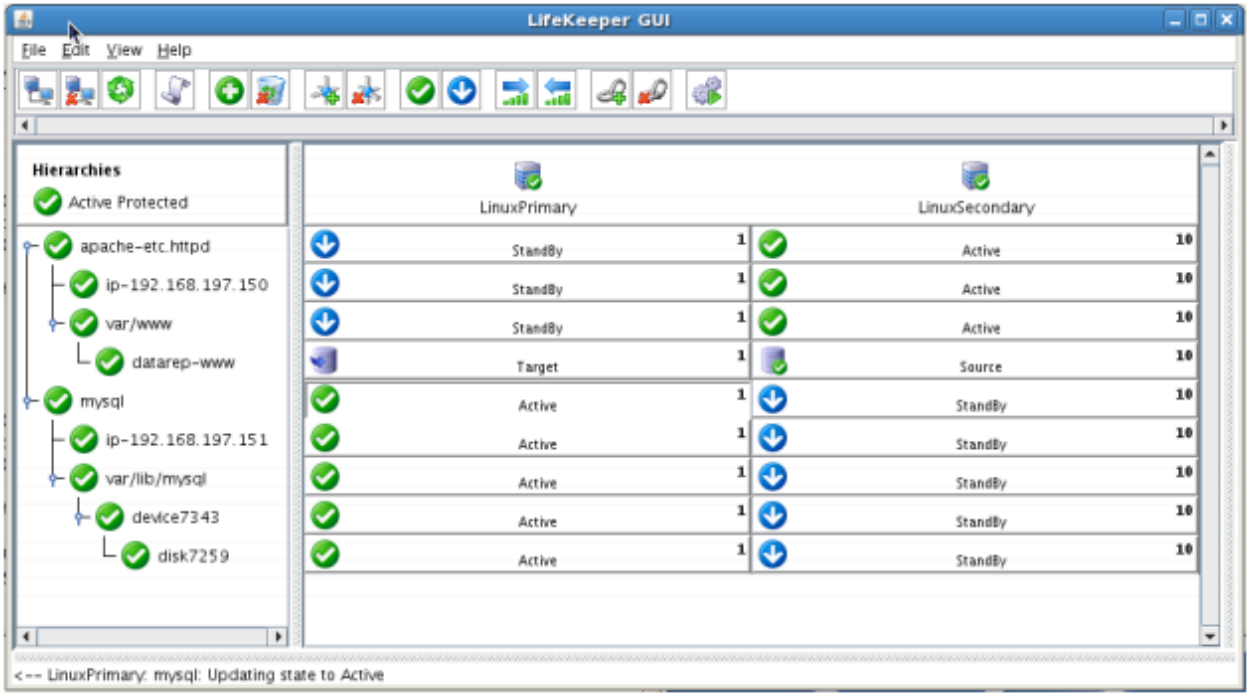
Manual Switchover of the MySQL Hierarchy back to Primary Server

Procedure:

- From the LifeKeeper GUI, right click on the MySQL resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

Expected Result:

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXPRIMARY



Tests/Verification:

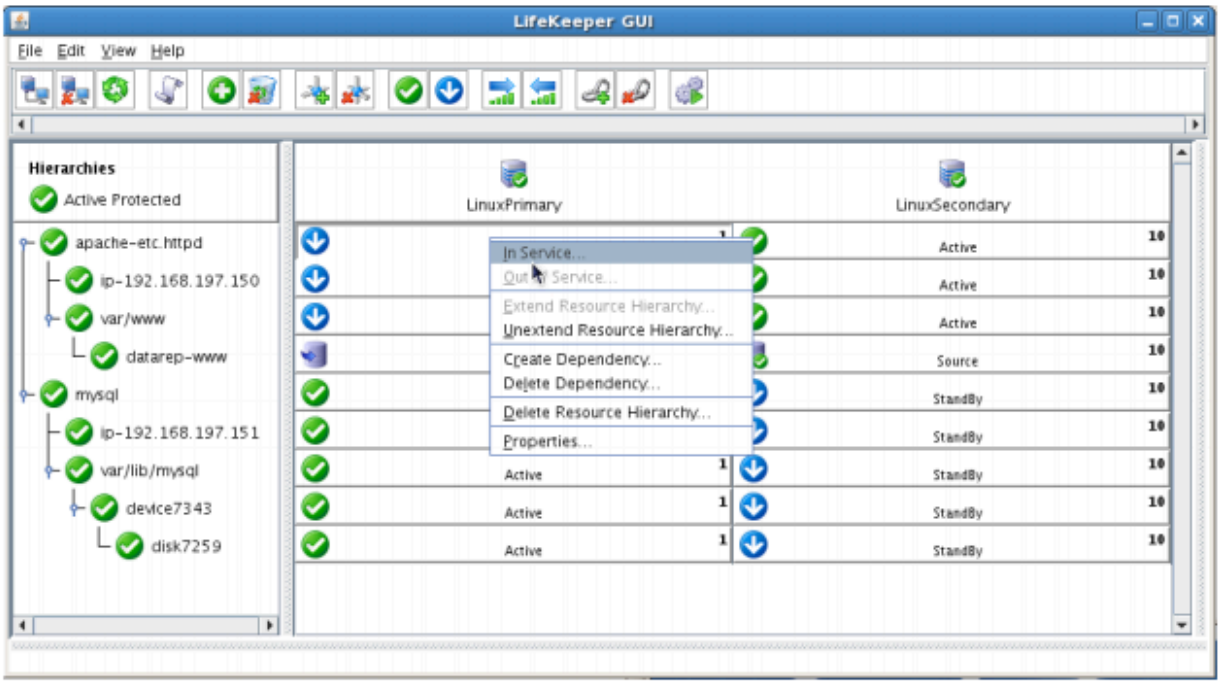
- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXPRIMARY.
- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.151 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/lib/mysql shared iSCSI filesystem is mounted on LINUXPRIMARY
- Verify the MySQL services are running on LINUXPRIMARY by running “ps –ef | grep –i mysql”
- On LINUXPRIMARY run the following command to verify client connectivity to the MySQL database:

    - # mysql –S /var/lib/mysql/mysql.sock –u root –p
    - (enter password “SteelEye”)
- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXSECONDARY, run “mount /dev/sdc1 /var/lib/mysql”. This should FAIL because LINUXSECONDARY does not own the SCSI reservation on this LUN.

Manual Switchover of the Apache Hierarchy back to the Primary Server

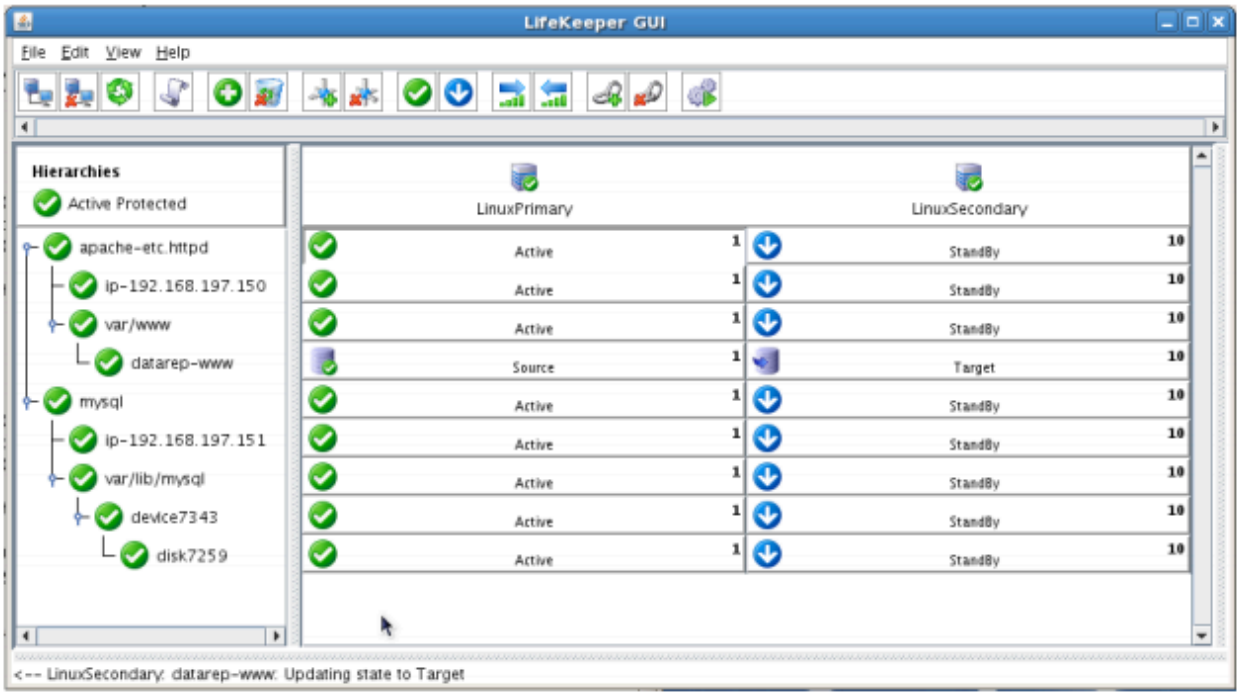
Procedure:

- From the LifeKeeper GUI, right click on the Apache resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



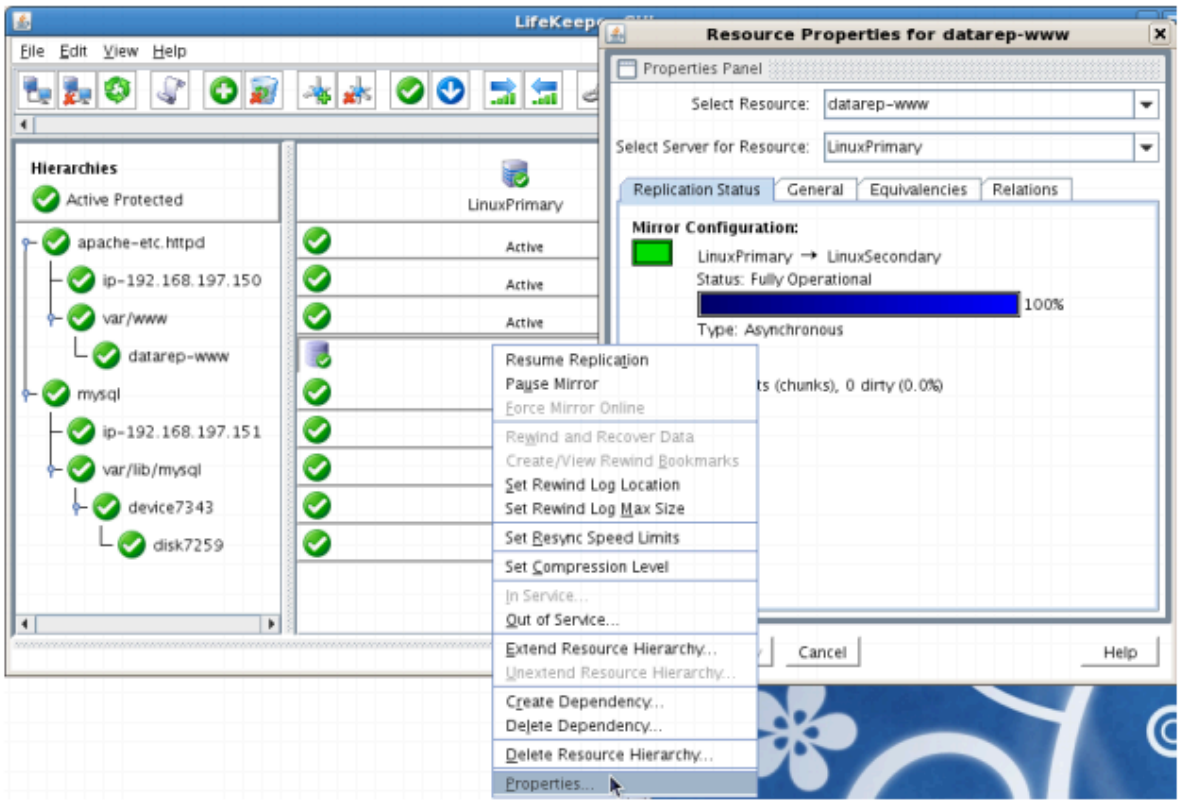
Expected Result:

- Beginning with the Apache resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXPRIMARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXPRIMARY -> LINUXSECONDARY

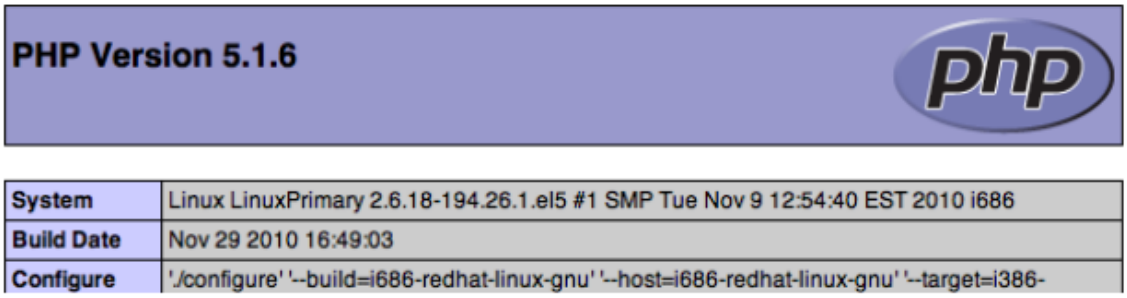


Tests/Verification:

- Using the LifeKeeper GUI, verify that the Apache and dependent resources are active on LINUXSECONDARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-www” resource and select Properties



- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.150 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/www replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY
- Verify the Apache services are running on LINUXPRIMARY by running “ps –ef | grep –i httpd”
- Open a Web Browser to http://192.168.197.150 and verify that it can successfully connect. The PHPInfo output should indicate that the system name is “LinuxPrimary



Simulate a network failure on the Primary Server by failing the IP resource

> **IMPORTANT:** Only perform this test if you have more than one communications path configured.

If you perform this test and have only one communications path configured, your system will enter a split-brain scenario as described in the LifeKeeper Administration Guide found here. Refer to this document for more information or contact SIOS presales technical support for assistance in resolving this condition.

Procedure:

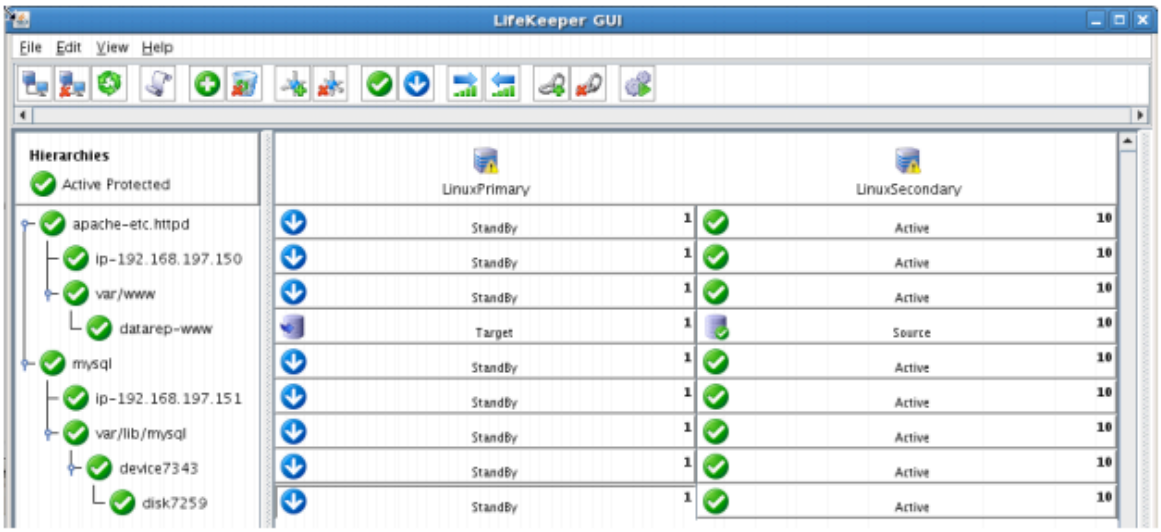
- On LINUXPRIMARY, pull the network cable attached to the NIC that the virtual IP address is configured on

Expected Result:

- The IP Resource should fail first.
- The entire hierarchy should failover to LINUXSECONDARY

Tests/Verification:

- Check the LifeKeeper Log to verify the IP resource failed – “/opt/LifeKeeper/bin/lk_log log”
- Using the LifeKeeper GUI, verify the MySQL and Apache resource hierarchies fail over successfully to LINUXSECONDARY



Hard failover of the resource from the Secondary Server back to the Primary Server

Procedure:

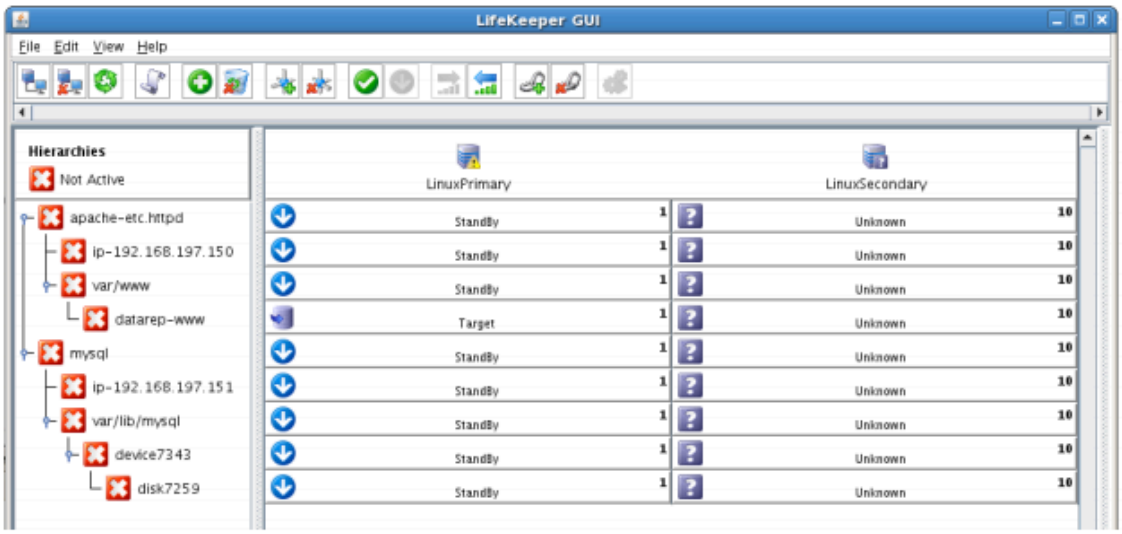
- Pull the power cord on LINUXSECONDARY, as this is the server with all resources currently In Service.

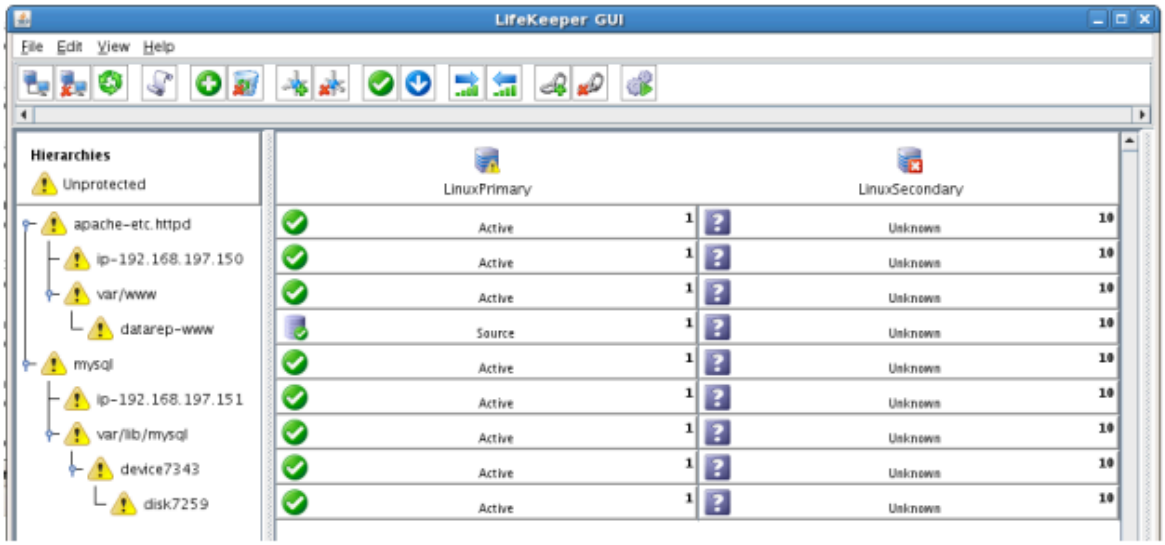
Expected Result:

- After failure has been detected, beginning with the dependent resources (IP and Volume), all resources will be brought in service on LINUXPRIMARY.

Tests/Verification:

- Using the LifeKeeper GUI, verify the mirror has reversed and is in a Resync Pending state waiting for LINUXSECONDARY to come back on line.
- Verify the PostgreSQL Server services are running on LINUXPRIMARY.
- Verify that the client can still connect to the Webserver and database running on LINUXPRIMARY.
- Verify you can write data to the replicated volume, /var/www on LINUXPRIMARY.





Bring Failed Server back on line

Procedure:

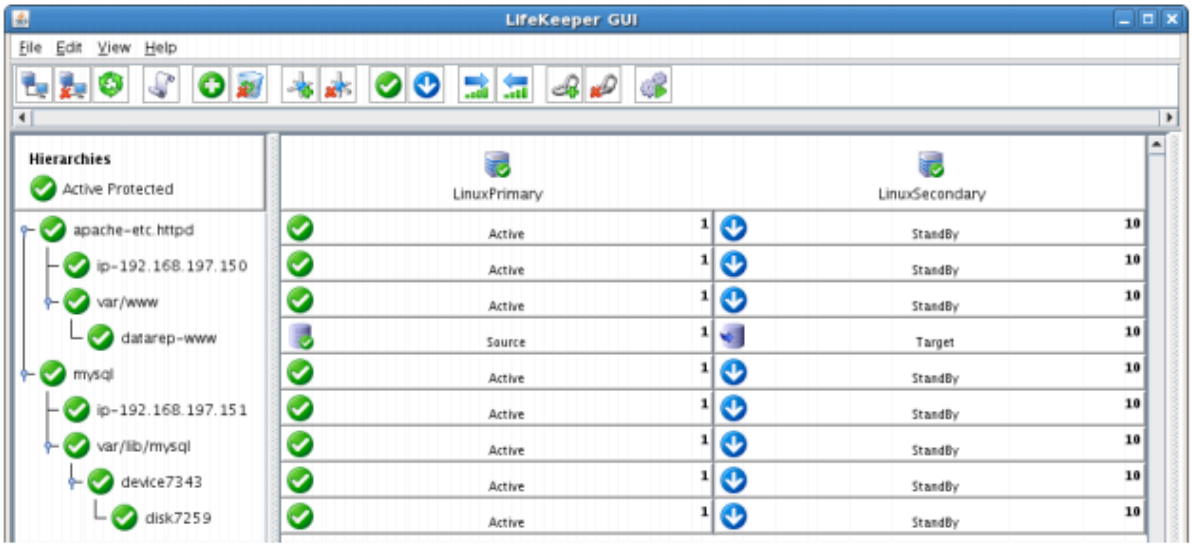
- Plug the power cord back into LINUXSECONDARY and boot it up.

Expected Result:

- Using the LifeKeeper GUI, verify that LINUXSECONDARY is coming back up and has become the Standby Server.

Tests/Verification:

- Verify the mirror performs a quick partial resync and moves to the Mirroring state
- Verify the Apache and MySQL Hierarchy are in service on LINUXPRIMARY and standby on LINUXSECONDARY.



Verify Local Recovery of MySQL Server

Procedure:

- Kill the PostgreSQL processes via the command line:
  - # ps –ef | grep sql
  - # killall mysqld mysqld_safe
  - run “ps –ef | grep sql once again to verify that the processes no longer exist

Expected Result: (Assumes Local Recovery for SQL resource is set to YES)

- The MySQL Server service should stop.
- The MySQL quickcheck process will automatically restart the MySQL Server Service when it runs periodically.
- No failure of MySQL should occur.

Tests/Verification:

- Execute “ps –ef | grep sql” once again to verify that the mysql processes have been restored locally on LINUXPRIMARY.
- Verify connectivity to the MySQL database by running:
  - # mysql –S /var/lib/mysql/mysql.sock –u root –p
  - (Enter password “SteelEye”)
- If you inspect the LifeKeeper logs, you will see information indicating that LifeKeeper dectected the failure of the MySQL service and recovered it locally. Run /opt/LifeKeeper/bin/lk_log log for more information.

12. LifeKeeper Single Server Protection

LifeKeeper Single Server Protection Release Notes

LifeKeeper Single Server Protection for Linux Installation Guide

12.1. LifeKeeper Single Server Protection for Linux Release Notes

Version 9.5.0

Released May 12, 2020

Important!!

Read This Document Before Attempting To Install Or Use This Product!
This document contains last minute information that must be considered before, during and after installation.

Introduction

This release notes document is written for the person who installs, configures and/or administers the LifeKeeper Single Server Protection for Linux product. The document contains important information not detailed in the formal LifeKeeper Single Server Protection documentation set such as system requirements, new features and links to product restrictions and troubleshooting hints and tips. It is important that you review this document before installing and configuring LifeKeeper Single Server Protection software.

LifeKeeper Single Server Protection Product Description

LifeKeeper Single Server Protection allows for application monitoring in a single node configuration (i.e., no cluster requirements or restraints). Single node environments may be physical or virtual (vSphere, KVM, Amazon EC2). LifeKeeper Single Server Protection is built on the proven and stable architecture of SIOS LifeKeeper. LifeKeeper Single Server Protection provides superior application monitoring and can perform recovery of failed applications and system infrastructure items (e.g., NFS share, IP address, File System). If an application cannot be recovered for some reason, LifeKeeper Single Server Protection will initiate a restart of the node via a system reboot or via a VMware HA restart for VMware virtual machines configured for VM and Application Monitoring.

Components

LifeKeeper Single Server Protection Software is bundled and runs on 64-bit systems (x86_64, AMD64) and is comprised of the following components:

- LifeKeeper Single Server Protection Software
- SteelEye Management Console with LifeKeeper Single Server Protection vSphere Client Plug-in (optional software for VMware environments only)

LifeKeeper Single Server Protection Optional Recovery Software

The following optional software provides resource definition and recovery software for the application versions listed. See the Support Matrix and Recovery Kit Administration Guides for the requirements for each recovery software.

Package	Package Name	Protected Applications
LifeKeeper Apache Web Server Recovery Kit	steeleye-lkAPA-9.5.0-7075.noarch.rpm	Apache Web Server v2.4
LifeKeeper SAP DB / MaxDB Recovery Kit	steeleye-lkSAPDB-9.5.0-7075.noarch.rpm	SAP MaxDB v7.9
LifeKeeper DB2 Recovery Kit	steeleye-lkDB2-9.5.0-7075.noarch.rpm	IBM DB2 Universal Database v10.5, v11.1 and v11.5 IBM DB2 Enterprise Server Edition (ESE) v10.5, v11.1 and v11.5 IBM DB2 Workgroup Server Edition (WSE) v10.5, v11.1 and v11.5 IBM DB2 Express Edition v10.5, v11.1 and v11.5
LifeKeeper Oracle Recovery Kit	steeleye-lkORA-9.5.0-7075.noarch.rpm	Oracle Database Enterprise Edition v12c, v12c R2, v18c and v19c (excluding ASM and pluggable database) Oracle Database Standard Edition 2 (SE2) v12c, v12c R2, v18c and v19c (excluding ASM and pluggable database)
LifeKeeper MySQL Recovery Kit	steeleye-lkSQL-9.5.0-7075.noarch.rpm	MySQL and MySQL Enterprise v5.7 and v8.0 MariaDB v10.3 and v10.4
LifeKeeper PostgreSQL Recovery Kit	steeleye-	PostgreSQL v9.5,v9.6, v10, v11 and v12

	lkPGSQL-9.5.0-7075.noarch.rpm	EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server v9.4, v9.5, v9.6, v10.0, v11.0 and v12.0
LifeKeeper Sybase ASE Recovery Kit	steeleye-lkSYBASE-9.5.0-7075.noarch.rpm	Sybase ASE 15.7 and 16.0
LifeKeeper Postfix Recovery Kit	steeleye-lkPOSTFIX-9.5.0-7075.noarch.rpm	Postfix software provided with the supported Linux distributions installed and configured on each server. The same version of Postfix should be installed on each server.
LifeKeeper Samba Recovery Kit	steeleye-lkSMB-9.5.0-7075.noarch.rpm	Standard Samba file services provided with the supported Linux distributions
LifeKeeper NFS Server Recovery Kit	steeleye-lkNFS-9.5.0-7075.noarch.rpm	Linux kernel version 2.6 or later The NFS Server and client packages must be installed on SLES systems.
LifeKeeper Network Attached Storage Recovery Kit	steeleye-lkNAS-9.5.0-7075.noarch.rpm	NFS version of Mounted NFS file systems from an NFS server or Network Attached Storage (NAS) device v2, v3 and v4
LifeKeeper WebSphere MQ Recovery Kit	steeleye-lkMQS-9.5.0-7075.noarch.rpm	IBM MQ v8.0, v9.0 and v9.1 See Known Issues and Restrictions > Installation.
Quick Service Protection	steeleye-lkQSP-9.5.0-7075.noarch.rpm	Quick Service Protection supplies functionality to easily protect OS services.

Features of LifeKeeper Single Server Protection

Product	Description
Temporal Recovery Logic	Local recovery attempt limits can be set providing improved application availability.
Multi-Level Policies	Recovery options can be specified at the server and resource levels allowing clients to define the most appropriate recovery strategy for each application.
Notification Only/ Maintenance Mode	Allows users to temporarily disable monitoring of one or more resources preventing LifeKeeper Single Server Protection from attempting to recover a resource that is undergoing maintenance.
VMware vSphere	Integrates with VMware's vSphere platform to improve application availability while allowing organizations to realize the full benefits of server virtualization and automation.

Integration	(VMware environments only)
vSphere Client Plug-in	Centralized management and monitoring through the vSphere Client. (VMware environments only)

Product	Feature
New in Version 9.5.0	
LifeKeeper Core	Supports Red Hat Enterprise Linux 7.8 (Certified in July 2020)
	Supports CentOS 7.8 (Certified in July 2020)
	Supports Oracle Linux 7.8 (Certified in July 2020) .
	Supports SUSE Linux Enterprise Server 12 SP5 (Certified in July 2020)
	Support VMware vSphere 7.0 (Certified in July 2020)
	Supports CentOS 8.0
	Supports Oracle Linux 8.0
	Supports Red Hat Enterprise Linux 8.1
	Supports CentOS 8.1
	Supports Oracle Linux 8.1
	The CLI has been enhanced to allow you to control LifeKeeper through the Command Line Interface. See LKCLI for details.
	Bug Fixes
PostgreSQL	Support PostgreSQL 12
	EDB Postgres Advanced Server v12.0 is supported. (Certified in July 2020)
LifeKeeper Core, Filesystem, NFS, DB2, MaxDB, Sybase ASE	Bug Fixes
New in Version 9.4.1	
LifeKeeper Core	OpenJDK included with OS is installed. See Configuring the LifeKeeper GUI for details.
	Supports SUSE Linux Enterprise Server 15 SP1
	Supports Oracle Linux 7.7
	Supports CentOS 7.7
	Supports AWS Nitro system
	Supports AWS Transit Gateway
	Bug Fixes
Install, IP, MaxDB	Bug Fixes
New in Version 9.4.0	
LifeKeeper Core	Oracle Linux 7 Unbreakable Enterprise Kernel Release 5 (UEK R5) is supported.

	Red Hat Enterprise Linux 8 is supported. **Note:** Upgrading from one kernel version to another major version such as from RHEL7 to RHEL8 is NOT supported. (i.e DataKeeper resource does NOT work when upgrading from RHEL7 to RHEL8.) Red Hat Enterprise Linux 7.7 is now supported. (Authorized in November 2019)
MySQL	MariaDB10.3 is supported.
DB2	DB2 11.5 is supported.
General maintenance	Bug Fixes
New in Version 9.3.2	
LifeKeeper Core	Red Hat Enterprise Linux 7.6 is supported.
	CentOS 7.6 is supported.
	Oracle Linux Version 7.6 is supported.
	SUSE Linux Enterprise Server 12 SP4 is supported.
	SUSE Linux Enterprise Server 15 is supported.
Install	The -s option for saving the current setup configuration has been added to the setup command.
PostgreSQL	PostgreSQL 11 is supported.
	EDB Postgres Advanced Server v11 is supported.
MQ	SIOS Protection Suite for Linux now supports IBM MQ 9.1
Oracle	Support Oracle 19c (Certified in August 2019).
General Maintenance	Bug fixes
New in Version 9.3.1	
LifeKeeper Core	Updated the OpenSSL package to 1.0.2p Support Red Hat Enterprise Linux 6.10 Support CentOS 6.10

	Support Oracle Linux 6 Update 10
MySQL	Support MySQL 8.0
Oracle	Support Oracle 18c (Certified in March 2019)
Install	Bug fixes
New in Version 9.3	
LifeKeeper Core	Red Hat Enterprise Linux Version 7.5 is supported. CentOS7.5 is supported. Oracle Linux Version 7.5 is supported. Support VMware vSphere 6.7. (Certified in October 2018) Bug fixes
Install	The installation script has been renewed. For details, please click here.
Oracle, Samba, MQ, Sybase, Filesystem, Generic Application, QSP	Bug fixes
New in Version 9.2.2	
PostgreSQL	Support PostgreSQL 10 EDB Postgres Advanced Server v10.0 is now supported. (Certified in April 2018)
NAS	Bug fixes
New in Version 9.2.1	
LifeKeeper Core	Support Oracle Linux 7.4 Support CentOS 7.4 Support SUSE Linux Enterprise Server 12 SP3 • The kernel should be updated to 4.4.82-6.9.1 for SUSE Linux Enterprise Server 12 SP3 Bug fixes

PostgreSQL	Support EDB Postgres Advanced Server 9.6
MQ	Support IBM MQ 9.0
New in Version 9.2	
LifeKeeper Core	Support Red Hat Enterprise Linux 7.4 SNMP trap can be sent to multiple targets Virtualization environment Nutanix Acropolis Hypervisor is supported. (SPS is not supported) Bug fixes
IP	IP resources using real IP (primary IP address configured for NIC) can be created
PostgreSQL	Support PostgreSQL 9.6
MQ	Support IBM MQ 9.0 (Certified in December 2017)
SAP MaxDB, Install	Bug fixes
New in Version 9.1.1	
LifeKeeper Core	SUSE Linux Enterprise Server 12 SP1 support. * SLES12.0 is not supported. * Btrfs is not supported. Red Hat Enterprise Linux Version 7.3 support. Oracle Linux Version 7.3 support. * UEK is not supported. vSphere 6.5 support (SMC feature is no longer supported with vSphere 6.5). Bug fixes
PostgreSQL	PostgreSQL 9.5 support EDB Postgres Advanced Server v9.5 support For the details, refer to the SPS Optional Recovery Software Requirements, PostgreSQL Recovery Kit Administration Guide > Administration.
Sybase ASE	Sybase ASE 16.0 support.
MySQL	MySQL 5.7 support on RHEL 7.x/CentOS 7.x/OEL 7.x.

	* MySQL 5.7 on other OS is already supported.
New in Version 9.1.0	
LifeKeeper Core	Red Hat Enterprise Linux 6.8 support (Certified in September 2016). CentOS 6.8, Oracle Linux 6.8 support (Certified in September 2016). *MD RecoveryKit is not supported on these OS. LifeKeeper API for Monitoring Added API to supply LifeKeeper status and log information. Quick Service Protection support Added functionality to easily protect OS services. Bug Fixes.
New in Version 9.0.2	
LifeKeeper Core	Support of Red Hat Enterprise Linux Version 7.2. ※MySQL RK is not supporting RHEL 7.x/CentOS 7.x/OEL 7.x. ※Support of RHEL 7.2 by each application must be confirmed by user. Update OpenSSL package to 1.0.1q Bug Fixes.
MQ	WebSphere MQ – Added support for Multi-version WebSphere MQ. With this support queue managers for 7.1, 7.5, and 8.x can all be protected on the same cluster node. Added the function that mqm group user can execute MQ command alternatively Bug Fixes.
IP, Filesystem, PostgreSQL, SAP DB/MaxDB, Oracle	Bug Fixes.

Licensing	Update the package of FlexNet
New in Version 9.0.1	
LikeKeeper Core	Bug Fixes
DataKeeper	Bug Fixes
New in Version 9.0	
LifeKeeper Core	Combined documents of Parameters List, and added the lkchkconf command. vSphere 6 support (SMC feature is no longer supported with vSphere 6.) reiserfs filesystem is no longer supported. Arks supported with Red Hat Enterprise Linux Version 7.0/7.1, Community ENTerprise Operating System (CentOS) Version 7.0/7.1, and Oracle Linux Version 7.0/7.1 are the same as LifeKeeper for Linux v8.4.1. (Arks to be applied: PostgreSQL, MySQL, Oracle, DB2, Apache, Postfix, NFS, NAS, Samba) Bug Fixes.
GUI	JRE 8u51 support. (JRE 7 is no longer supported.) Chrome Browser is no longer supported Bug Fixes.
FileSystem, PostgreSQL	Bug Fixes.

Bug Fixes

The following is a list of the latest bug fixes and enhancements.

Bug	Description
PL-114	Added a timer to the restore functionality in the MaxDB ARK
PL-118	NFS quickCheck fails to detect matching export option list if an option is listed more than once
PL-1215	Enhanced quickCheck for NFS resources to monitor fsid
PL-2300	The LifeKeeper logrotate configuration does not reload rsyslog, which results in lost LK logging after log rotation
PL-2310	Avoid using Sybase “shutdown with nowait”
PL-2320	Repaired DB2 ARK to create DB2 resource if character code is set to IBM-943
PL-3149	Added XFS sanity check

PL-3232	Fixed an issue where resources could not be created if Sybase was installed outside of /opt/sybase
PL-3518	Stopping LifeKeeper needs to clean up in-progress tasks such as machine failover, quorum_isp, and quickCheck daemons
PL-4164	lkstop fails if the process of removing all the resources takes more than 90 seconds
PL-4355	Fixed an issue where LifeKeeper also stopped if the rsyslog service stopped on systems employing systemd

Discontinued Features

Feature	Description
Discontinued in Version 9.5.0	
LifeKeeper Core	System log management using syslog-ng is no longer supported. Please use rsyslog.
	SUSE Linux Enterprise Server (SLES) 11.0 to SP4 is no longer supported.
Oracle	Oracle Database Enterprise Edition 11g R2 is no longer supported.
	Oracle Database Standard Edition 11g R2 is no longer supported.
	Oracle Database Standard Edition One 11g R2 is no longer supported.
MySQL	MariaDB 5.5, 10.0 is no longer supported.
PostgreSQL	PostgreSQL 9.4 is no longer supported.
	EnterpriseDB Postgres Plus Advanced Server 9.4 is no longer supported.
Discontinued in Version 9.4.1	
	None

System Requirements

LifeKeeper Single Server Protection Product Requirements

LifeKeeper Single Server Protection is supported on any Linux platform that satisfies the minimum requirements included in the table below.

Description	Requirement
Linux Operating System	See the Linux Configuration Table for specific operating system information.
Virtual Environments	The guest operating system running on the virtual machine must be one of the supported versions listed in the Linux Configuration Table. The following virtual environment is an example where SIOS Protection Suite for Linux is deployed. Please refer to the Support Matrix for detailed versions of supported virtualization environments.

	• KVM • Oracle VM Server for x86 • VMware vSphere v5.5, v6.0, v6.5, v6.7 and v7.0 • Amazon EC2 • Nutanix Acropolis Hypervisor vSAN configuration is supported for vSphere 6.5 or later except RDM which is not supported by VMWare. Fibre channel SAN and shared SCSI cluster configurations are not supported with SPS for Linux running in a KVM and Oracle VM Server for x86 virtual machine. **Note:** On SLES v12 or later running on AWS or Azure, the dynamic change of the virtual IP address by the cloud network plug-in may affect the operation of the LifeKeeper cluster. For detailed information, see LifeKeeper Core – Known Issues / Restrictions.
Memory	The minimum memory requirement for a system running LifeKeeper Single Server Protection is 512 MB. This is the minimum amount required by LifeKeeper Single Server Protection supported Linux distributions. System memory should be sized for the applications that will be running on the LifeKeeper Single Server Protection protected system as well.
Disk Space	The LifeKeeper Single Server Protection requires the following disk space: /opt – approx 100MB (depending on kits installed) / – approx 110MB
Java Runtime Environment	• OpenJDK 1.8, 10 or later

LifeKeeper Single Server Protection Support Software Requirements

The following table of Supporting Software is only required in VMware VMs configured for VM and Application Monitoring.

Product	Requirement(s)	Disk Space Required
VMware	VMware vSphere Client (for LifeKeeper Single Server Protection vSphere Client Plug-in functionality) VMware Tools installed and running on all protected virtual machines VMware Application HA monitoring must be enabled and set to VM and Application Monitoring for all protected virtual machines	Approximately 175 KB in */opt* (for VMware Tools

Client Platforms and Browsers

The LifeKeeper Single Server Protection web client can run on any platform that provides support for Java Runtime Environment JRE8 update 51. The currently supported configurations are Firefox (Firefox 51 or earlier) and Internet Explorer on Linux, Windows Server 2008 R2, Windows Server 2012 R2, Windows Server 2016, Windows 7, Windows 8 or Windows 10 with JRE8 update 51. Other recent platforms and browsers will likely work with the SPS web client, but they have not been tested by SIOS Technology Corp. In addition, particular features of each browser also have not been tested.

IP addresses for LifeKeeper Single Server Protection components (e.g. the SteelEye Management Console and vCenter when installed in a VMware configuration) and protected Linux guests must be resolvable via DNS or the local hosts file (usually/etc/hosts or C:\windows\system32\drivers\etc\hosts). Using the local hosts file minimizes the client connection time and allows the client to connect even in the event of a Domain Name Server (DNS) failure.

Known Issues

See the Known Issues and Workarounds section in LifeKeeper Single Server Protection for Linux Technical Documentation for known issues, workarounds and other troubleshooting information.

12.2. LifeKeeper Single Server Protection for Linux Installation Guide

About LifeKeeper Single Server Protection for Linux

LifeKeeper Single Server Protection (SSP) allows for application monitoring in a single node configuration (i.e., no cluster requirements or restraints). Single node environments may be physical or virtual (vSphere, KVM, Amazon EC2). LifeKeeper SSP is built on the proven and stable architecture of SIOS LifeKeeper. LifeKeeper SSP provides superior application monitoring and can perform recovery of failed applications and system infrastructure items (e.g., NFS share, IP address, File System). If an application cannot be recovered for some reason, LifeKeeper SSP will initiate a restart of the node via a system reboot or via a VMware HA restart for VMware virtual machines configured for VM and Application Monitoring.

Note: Because LifeKeeper SSP is built using the SIOS LifeKeeper technology, you will see references to LifeKeeper throughout the documentation as well as references to information found in the SIOS Protection Suite for Linux documentation for topics common to both products. When referencing these common topics the following subject items do not apply to LifeKeeper SSP:

- Clustering
- Communication Paths
- Shared Storage (requirements, configuration, …)
- Extending / Unextending resource hierarchies
- Storage Kits (DR, DMMP, HDLM, LVM, MD, PPATH and NEC SPS)

Note: Without the underlying storage kits in LifeKeeper SSP, steps must be taken to ensure the devices required to mount protected file systems are activated during system boot (e.g. if the file system is mounted on a logical volume the volume must be in the active state before LifeKeeper SSP starts)

- Resource/Machine failovers (by default with LifeKeeper SSP these result in a node restart)
- Resource switchovers
- Switchable IP Addresses (with LifeKeeper SSP Switchable IP addresses are required for some protected applications but since there is only a single node no switching actually takes place)

Note: When operating on Amazon EC2, assign a secondary private IP address to the NIC using the Amazon EC2 Management Console prior to creating the IP resource. Next, create the IP resource as the private IP address on the NIC that is using the LifeKeeper GUI client. An Elastic IP can now be associated with the IP resource

For more information on the SIOS LifeKeeper product, on which LifeKeeper SSP is built, please see the SIOS Protection Suite for Linux documentation for the common release number. This documentation will provide detailed information on resource hierarchy creation, resource types, states and relationships, Graphical User Interface (GUI), as well as common and advanced tasks.

12.2.1. LifeKeeper Single Server Protection for Linux Introduction

LifeKeeper Single Server Protection (SSP) allows for application monitoring in a single node configuration (i.e., no cluster requirements or restraints). Single node environments may be physical or virtual (vSphere, KVM, Amazon EC2). LifeKeeper SSP is built on the proven and stable architecture of SIOS LifeKeeper. LifeKeeper SSP provides superior application monitoring and can perform recovery of failed applications and system infrastructure items (e.g., NFS share, IP address, File System). If an application cannot be recovered for some reason, LifeKeeper SSP will initiate a restart of the node via a system reboot or via a VMware HA restart for VMware virtual machines configured for VM and Application Monitoring.

Note: Because LifeKeeper SSP is built using the SIOS LifeKeeper technology, you will see references to LifeKeeper throughout the documentation as well as references to information found in the SIOS Protection Suite for Linux documentation for topics common to both products. When referencing these common topics the following subject items do not apply to LifeKeeper SSP:

- Clustering
- Communication Paths
- Shared Storage (requirements, configuration, …)
- Extending / Unextending resource hierarchies
- Storage Kits (DR, DMMP, HDLM, LVM, MD, PPATH and NEC SPS)

Note: Without the underlying storage kits in LifeKeeper SSP, steps must be taken to ensure the devices required to mount protected file systems are activated during system boot (e.g. if the file system is mounted on a logical volume the volume must be in the active state before LifeKeeper SSP starts)

- Resource/Machine failovers (by default with LifeKeeper SSP these result in a node restart)
- Resource switchovers
- Switchable IP Addresses (with LifeKeeper SSP Switchable IP addresses are required for some protected applications but since there is only a single node no switching actually takes place)

Note: When operating on Amazon EC2, assign a secondary private IP address to the NIC using the Amazon EC2 Management Console prior to creating the IP resource. Next, create the IP resource as the private IP address on the NIC that is using the

LifeKeeper GUI client. An Elastic IP can now be associated with the IP resource

For more information on the SIOS LifeKeeper product, on which LifeKeeper SSP is built, please see the SIOS Protection Suite for Linux documentation for the common release number. This documentation will provide detailed information on resource hierarchy creation, resource types, states and relationships, Graphical User Interface (GUI), as well as common and advanced tasks.

12.2.2. Installing the LifeKeeper Single Server Protection Software

This document will guide you through the installation of the LifeKeeper Single Server Protection Software (SSP) and assumes the user has basic knowledge of the Linux operating system. Please refer to the LifeKeeper Single Server Protection Software for Linux product documentation for more information.

Pre-Installation Requirements

Before installing SSP for Linux, please check the following:

- **SSP for Linux Release Notes** -The Release Notes include supported platforms, operating systems, applications, and storage. They also include the latest features and Bug Fixes.
- **TCP/IP Connection and Name Resolution** – In order to use the GUI function, both cluster nodes need to be able to resolve the name. Use the DNS service or */etc/hosts* for name resolution. Also, localhost needs to be resolved to 127.0.0.1.
- **Firewall** – The following ports are used:
  - Communication Path (TCP): 7365/tcp
  - Communication of a GUI Server: 81/tcp、82/tcp
  - RMI Communication between the GUI Server and Client: all the ports after 1024/tcp

More Firewall Information

- The port used for communication with the GUI server and a client needs to be open on the cluster node where SSP is installed and on all systems where the GUI client runs.
- For communication between the GUI server and a client, Java RMI (Remote Method Invocation) randomly uses ports 1024 and above. Please refer to the Technical Documentation for the setting details.
- Add the following to the port numbers you are using: *WebGUI server process and policy setting with the lkpolicy command : 778(SSL) /tcp*

- ◦ **Check the SELinux Setting** – When the SELinux setting is enabled, SSP for Linux may not be able to be installed depending on the mode.
  - enforcing mode – SSP for Linux cannot be installed
  - permissive mode – SSP for Linux can be installed (not recommended except in some ARK environments)
    - It is not recommended to use SELinux permissive mode unless it is required in an SAP environment. Please make sure that the application to be run on the cluster supports permissive mode. SELinux permissive mode has been tested for following ARKs: SAP MaxDB / Sybase / Oracle / DB2 / NFS / NAS / IP / FileSystem / MQ. Refer to Linux Dependencies for required packages.
  - disabled mode – SSP for Linux can be installed
    - Please refer to the OS distribution documentation on how to disable SELinux.
  - Install the appropriate package provided by your distribution.

- **Check Known Issues** – Please make sure that there are no known issues for your environment.

Installing SSP for Linux

Install the LifeKeeper Single Server Protection software on each server in the LifeKeeper Single Server Protection configuration.

Packages that LifeKeeper is dependent on are installed automatically because the LifeKeeper installation setup script uses package manager tools (yum or zypper) to ensure installation of all dependent packages.

> **IMPORTANT**: A functional yum or zypper configuration is required for the successful installation of LifeKeeper. A non-functional configuration can result in an installation failure (see Installation Known Issues for more information). Additionally, the package manager repo or rpm database must not be locked as that could cause the install to hang. If the dependent packages cannot be installed automatically via the package manager, refer to Linux Dependencies and install the necessary packages in advance.

The LifeKeeper Single Server Protection image file (lkssp.img) provides a set of installation scripts designed to perform the user interactive system setup tasks that are necessary when installing SSP on your system (see Interactive Mode for more information). A non-user interactive install can be performed as well (see Non-interactive Mode for more information).

A licensing utilities package is also installed providing utilities for obtaining and displaying the Host ID or Entitlement ID of your server. Host IDs and/or Entitlement IDs are used to obtain valid licenses for running LKSSP.

> **IMPORTANT**: LifeKeeper Single Server Protection does not provide shared storage support or I/O fencing. Each server must use local disk storage for application data. All LifeKeeper Single Server Protection packages are installed in the directory */opt/LifeKeeper*.

Please refer to How to Use Setup Scripts for the installation activities.

For upgrading, please refer to Upgrading SSP.

12.2.3. How to Use Setup Scripts

How to Install / Upgrade LifeKeeper SSP Using the Setup Script

To install LifeKeeper SSP, perform the following activities using the setup script.

Interactive Mode

1. After logging in as the root user, use the following command to mount the sps.img file:

   ```
   mount <PATH/IMAGE_NAME> <MOUNT_POINT> -t iso9660 -o loop
   ```

   Where PATH is the path to the image
   IMAGE_NAME is the name of the image
   MOUNT_POINT is the path to mount location

2. Change to the directory where sps.img is mounted and enter the following:

   ```
   ./setup
   ```

3. The script collects information about the system environment and determines what you need to do to install LifeKeeper SSP.

   If the system requirements for installation or upgrade are not satisfied, then an error message is displayed and the installation / upgrade is cancelled.

   Also, if some restrictions arise or a configuration change is required, a warning message will be displayed requiring the user to decide whether to continue or abort the installation.

4. Select the LifeKeeper SSP features and Application Recovery Kits (ARKs) to install via the main dialog screen.

   Please refer to the Dialog Screen.

5. Once all the required LifeKeeper SSP features and ARKs have been selected, select <Done> to begin the installation.

   If any notifications are output when the installation completes, please take the necessary actions to correct them.

Non-interactive Mode

1. After logging in as root user, use the following command to mount the sps.img file:

```
mount <PATH/IMAGE_NAME> <MOUNT_POINT> -t iso9660 -o loop
```

PATH is the path to the image
IMAGE_NAME is the name of the image
MOUNT_POINT is the path to mount location

2. Change to the directory where sps.img is mounted and enter the following command. First you will need to run setup in Interactive Mode, with the “Save Configuration” (-s) option:

   `./setup -s` <response_file>

   Select the necessary packages and options and complete setup. A configuration file will be saved in the location you specified. This configuration file can be copied to other systems and used as follows.

3. On the system where you wish to perform a non-interactive install, enter the following command:

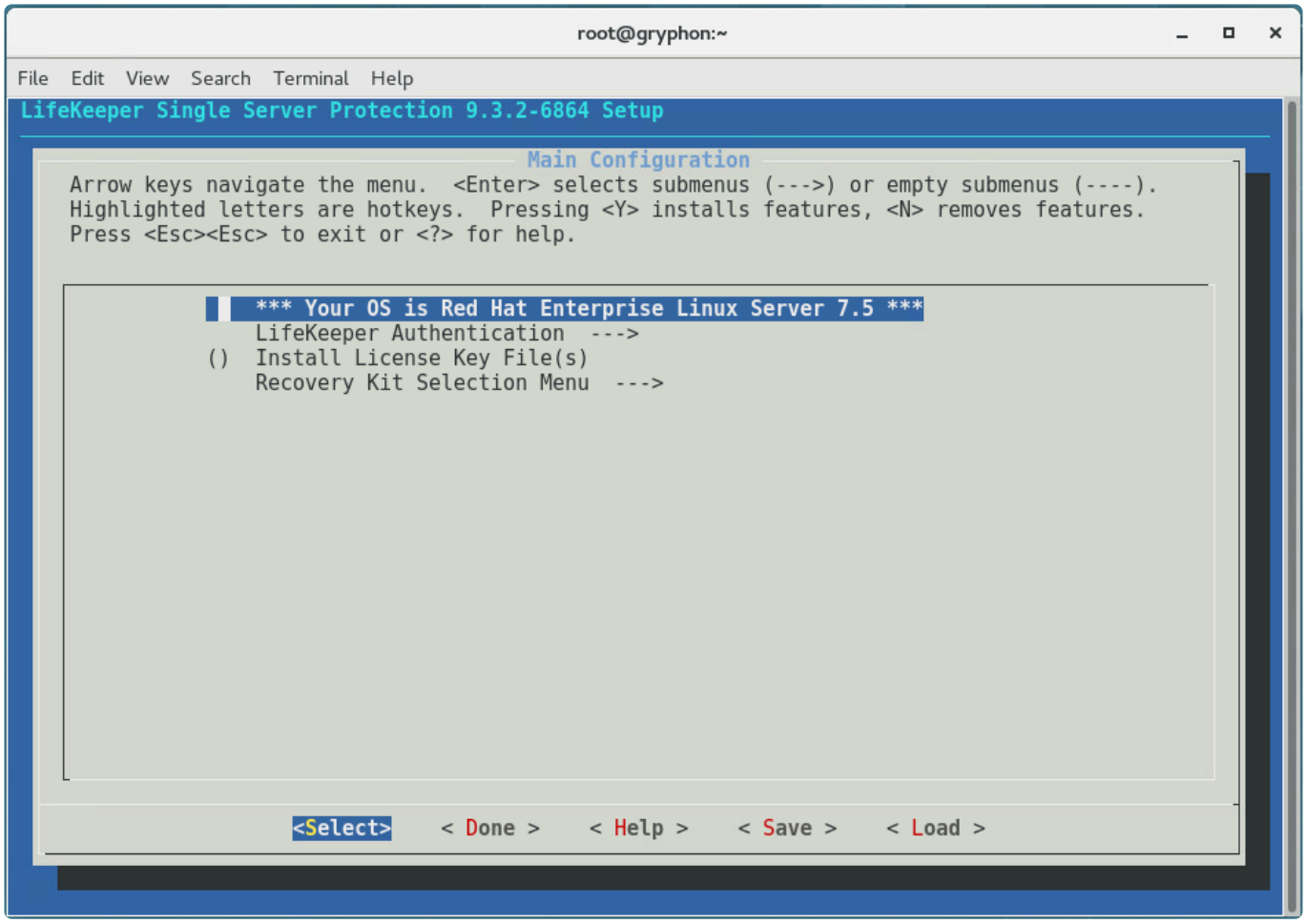
   `./setup -f` <response_file> -q y

   The “-q y” option ensures that user prompts are skipped, and the default answers given.

> **Note:** When using a configuration file for non-interactive installations the system on which the file is used must be configured the same as the system on which the file was generated. If the systems have too many differences the non-interactive installation may fail.

How to Use the Dialog Screen

The dialog screen is displayed below.



Use the following keys to navigate the menu.

↑ ↓	Navigate between menu items
← →	Navigate between the menu buttons at the bottom of the screen
ENTER	Open the selected sub menu
Y / N / SPACE	Turn on, turn off or invert the selected item

The menu buttons at the bottom of the screen are used for the following operations.

Select	Opens a sub menu dialog screen
Done	Closes the current screen and returns to the previous screen. Selecting this button on the main screen completes the configuration.
Help	Displays help for the highlighted item
Save	Saves the current settings in a configuration file. The saved configuration file can be used for non-interactive installations.
Load	Loads a saved configuration file

The “Save” and “Load” menu buttons display a dialog screen asking for a configuration file name for use in saving the current configuration or for loading a saved configuration. If you want to change the default file name provided, move to the file name field using the [TAB] key, and enter a new name. **Note:** The “Save” operation will prompt for confirmation before overwriting a file with the same name.

The items listed below are configurable during installation. During an upgrade only items that can be configured are listed. Using the hotkey <Z> will show those items that will remain unchanged during the upgrade.

- **Install Java Runtime (JRE)**

  Install the Java runtime environment used by the LifeKeeper GUI.

- **LifeKeeper Authentication**

  Specify the users allowed to log in to the LifeKeeper SSP GUI along with their privilege levels. Multiple user accounts can be specified by separating them with blanks. For details, refer to GUI User Settings.

- **Install License Key File(s)**

  Install the licenses required to start LifeKeeper SSP by entering the path name of the license file to install. Multiple files can be specified by separating them with spaces.

  Please refer to Licensing for details.

- **Recovery Kit Selection**

  Select the Application Recovery Kits to install.

  Application Recovery Kits are broken into several categories based on common functionality.

  Please refer to Categories for Application Recovery Kits for details.

- **LifeKeeper Startup After Install**

  When selected, SPS for Linux will be started when the installation is completed.

Adding / Removing Application Recovery Kits

To add Application Recovery Kits after completing an installation, simply execute setup, select the Recovery Kit in the Recovery Kit Selection, followed by the Application Recovery Kit Category and then select the desired kit.If you deselect an Application Recovery Kit which is no longer necessary, that kit will be removed.

Repair Installation

To repair a LifeKeeper SSP installation run setup with the “—force” option. A repair installation will update the installation replacing any lost or corrupted files.

setup Script Options

The setup script can be executed with the following options:

- `-f <file>`

  Install non-interactively. <file> contains the configuration information to use during the installation.

- `-s <file>`

  Save a configuration file containing your menu selections. This file can then be used with the “-f” option to install the same LifeKeeper configuration to another system. For example, run:

  ```
  setup -s <file>
  ```

  Select the necessary packages and options and complete setup.

  Then run:

  ```
  setup -f <file> -q y
  ```

  to run a silent installation of LifeKeeper (on another system) with the same options that were selected the first time setup was run.

- `—force`

  Forcibly reinstall SPS for Linux.

- `-q <y/n>`

  Specifies the response to any confirmation questions that may arise during non-interactive installation.

Categories for Application Recovery Kits

Category	Description
Application Suite	A group of recovery kits that protect applications such as IBM MQ.
Database	A group of recovery kits that protect database applications, including, but not limited to, Oracle, PostgreSQL, and MaxDB.
File Sharing	A group of recovery kits that protect file sharing services such as NFS and Samba.
Mail Server	A group of recovery kits that protect email services such as Postfix.
Storage	A group of recovery kits that protect data storage methods, including, but not limited to, DataKeeper (replication), Device Mapper (DM) Multipath (DMMP), and Network Attached Storage (NAS).

Web Server	A group of recovery kits that protect web services such as Apache.

12.2.4. Upgrading LKSSP

LifeKeeper Single Server Protection (SSP) can be upgraded to future releases while maintaining existing hierarchies.

Note: Only the previous two generations of LifeKeeper Single Server Protection can be upgraded to the latest version. If you are upgrading from older versions, you will need to uninstall the old version and reinstall LifeKeeper Single Server Protection. Instead of uninstalling the old version, you can also upgrade to the latest version after upgrading the older version to either of the previous two generations.

Note: If using `lkbackup` during your upgrade, see the lkbackup Known Issue for further information.

1. Upgrade your Linux operating system before upgrading SSP If necessary.
2. Upgrade LifeKeeper referring to How to Use Setup Scripts.

12.2.5. Obtaining and Installing the License for LKSSP

LifeKeeper Single Server Protection requires a unique license for each server. The license is a run-time license, which means that you can install LifeKeeper Single Server Protection without it, but the license must be installed before you can successfully start and run the product.

The Installation script installs the Licensing Utilities package which obtains and displays all of the available Host IDs for your server during the initial install of your LifeKeeper Single Server Protection Software. Once your licenses have been installed the utility will return the Entitlement ID if it is available or the Host IDs if it is not.

Note: Host IDs, if displayed will always be based on the MAC address of the NICs.

Any LifeKeeper Single Server Protection licenses obtained from the SIOS Technology Corp. Licensing Operations Portal will contain your Entitlement ID and will not be locked to a specific node in the cluster. The Entitlement ID (Authorization Code) which was provided with your LifeKeeper Single Server Protection Software, is used to obtain the permanent license required to run the LifeKeeper Single Server Protection Software. The process is illustrated below.

Note: Each software package requires a license for each server.

License Key Manager

In addition to installing product licenses, the **License Key Manager** allows you to perform the following functions:

- View all licenses currently installed on your system.
- View all expiration notifications (days remaining) for each time-expiring license.
- Identify invalid licenses that are currently installed.
- Delete any installed licenses (right-click on the license and select **Delete**).
- Delete all expired licenses as a group (press the **Delete Expired License** button).
- **Refresh** the Installed License list when installing software or upgrades.

Perform the following steps to obtain and install your licenses for each server:

1. Get your **Host ID**. At the end of the SSP installation, make note of the **Host ID** displayed by the **License Key Installer** utility.
2. Write the **Host IDs** in a notebook or save them in a file. If saved in a file, copy that file to a system with internet access. Otherwise, take your notebook with you to the system with internet access.
3. Ensure you have your LifeKeeper Single Server Protection **Entitlement ID** (Authorization Code). You should have received an email with your software containing the Entitlement ID needed to obtain the license.
4. Obtain your licenses from the SIOS Technology Corp. Licensing Operations Portal.

   a. Using the system that has internet access, navigate to the SIOS Technology Corp. Licensing Operations Portal and log in entering your **User Name** and **Password** (or register if you do not already have an account).
   Note: New users must enter the Entitlement ID that is included in the delivery email..

   b. From the **Activation and Entitlements** dropdown select **List Entitlements**.

   c. Check the box to the left of the product line item(s) that you wish to license.

   d. From the **Action** dropdown select **Activate** and enter the requested information (including your system HOSTNAME) then select **Next**.

   e Click on the **Gray Plus Sign** to choose an already defined host or create a new host by selecting the **Green Plus Sign**.

   f. Select **ANY** for the Node Locked Host choice if it is available, otherwise select **ETHERNET MAC**

ADDRESS and enter the Host ID (MAC address), click **OK** then click **Generate**.

> **Note:** The Host ID is 12 characters with no spaces, no colons, no dashes, and no separators.

g. Check the box to the left of the **Fulfillment ID** and select **Complete**.

h. From the **License Support** dropdown select **List Licenses**. Check the box to the left of the **Fulfillment ID** and select **Email** from the **View** dropdown.

i. Enter a valid email address to send the license to and select **Send**.

j. Retrieve the email(s).

k. Copy the file(s) to the appropriate system(s).

5. Install your license(s).

- On each system, copy the license file(s) to /var/LifeKeeper/license. Run /opt/LifeKeeper/bin/lkkeyins and specify the filename (including full path) to the file.

6. Repeat on all additional servers. You must install a license on the other SPS server(s) using the unique Host ID for each server.

7. Restart SIOS Protection Suite for Linux.

Primary Network Interface Change May Require a License Rehost

The Host ID used by the License Key Installer utility is obtained from the SSP for Linux server's primary network interface card (NIC). SSP for Linux will check for a valid license each time it starts. If your SSP server should require a NIC replacement in the future that would cause the Host ID to change, then the next time SSP for Linux is stopped, a License Rehost must be performed before starting either again. Log in to the SIOS Technology Corp. Licensing Operations Portal and select **License Support**, **List Licenses**, **Action**, **Rehost**. (**Note:** A rehost can be performed one time per six-month period without contacting support.)

Troubleshooting

If errors are encountered, please try the following before contacting Support:

- Verify credentials by logging in to the SIOS Technology Corp. Licensing Operations Portal. Enter **User ID** and **Password**. Run `%ExtMirrBase%\lmSubscribe.exe` again using the correct **User ID** and **Password**.

- To force a manual check for a license renewal, stop and restart the service. (**Note:** To find the service, bring up the view for all of the Linux services and search for "**SIOS Subscription Licensing**".)

- If ownership of the license certificate has changed, please contact SIOS Technology Corp. Support personnel to have the certificate moved to the new owner. Once ownership has been moved, the automatic license renewal service will need to be updated with these new credentials by running the above command again using the new **User ID** and **Password**.

12.2.6. Resource Policy Management

Resource Policy Management in LifeKeeper Single Server Protection (SSP) provides behavior management of resource local recovery and failover. Resource policies are managed with the **lkpolicy** command line tool (CLI).

LifeKeeper SSP Recovery Behavior

LifeKeeper SSP is designed to monitor individual applications and groups of related applications, periodically performing local recoveries or notifications when protected applications fail. Related applications, by example, are hierarchies where the primary application depends on lower-level storage or network resources. When an application or resource failure occurs, the default behavior is:

1. **Local Recovery**: First, attempt local recovery of the resource or application. An attempt will be made to restore the resource or application on the local server without external intervention. If local recovery is successful, then LifeKeeper SSP will not perform any additional action.

2. **Failover**: Second, if a local recovery attempt *fails* to restore the resource or application (or the *recovery kit* monitoring the resource has no support for local recovery), then a ***failover*** will be initiated (see Failover in the Standard Policies section below).

Please see LifeKeeper Single Server Protection Fault Detection and Recovery Scenario for more detailed information about our recovery behavior.

Custom and Maintenance-Mode Behavior via Policies

LifeKeeper SSP supports the ability to set additional policies that modify the default recovery behavior. There are four policies that can be set for individual resources (see the section below about precautions regarding individual resource policies) or for an entire server. ***The recommended approach is to alter policies at the server level***.

The available policies are:

Standard Policies

- **Failover** – For LifeKeeper SSP this policy setting can be used to turn on/off resource failover (which results in a reboot).

- **LocalRecovery** – LifeKeeper SSP by default, will attempt to recover protected resources by restarting the individual resource or the entire protected application prior to performing a failover (which would be a reboot). This policy setting can be used to turn on/off local recovery.

- **TemporalRecovery** – Normally, LifeKeeper SSP will perform local recovery of a failed resource. If local recovery fails, LifeKeeper SSP will perform a reboot. If the local recovery succeeds, failover (which would be a reboot) will not be performed.

There may be cases where the local recovery succeeds, but due to some irregularity in the server, the local recovery is re-attempted within a short time; resulting in multiple, consecutive local recovery attempts. This may degrade availability for the affected application.

To prevent this repetitive local recovery/failure cycle, you may set a temporal recovery policy. The temporal recovery policy allows an administrator to limit the number of local recovery attempts (successful or not) within a defined time period.

> *Example*: If a user sets the policy definition to limit the resource to three local recovery attempts in a 30-minute time period, LifeKeeper SSP will failover(reboot) when a third local recovery attempt occurs within the 30-minute period.

Defined temporal recovery policies may be turned on or off. When a temporal recovery policy is off, temporal recovery processing will continue to be done and notifications will appear in the log when the policy would have fired; however, no actions will be taken.

> **Note**: It is possible to disable failover and/or local recovery with a temporal recovery policy also in place. This state is illogical as the temporal recovery policy will never be acted upon if failover or local recovery are disabled.

Meta Policies

The “meta” policies are the ones that can affect more than one other policy at the same time. These policies are usually used as shortcuts for getting certain system behaviors that would otherwise require setting multiple standard policies.

- **NotificationOnly** – This mode allows administrators to put LifeKeeper SSP in a “monitoring only” state. **Both** local recovery **and** failover(reboot) **of a resource (or all resources in the case of a server-wide policy) are affected**. The user interface will indicate a **Failure** state if a failure is detected; but no recovery or failover(reboot) action will be taken. **Note**: The administrator will need to correct the problem that caused the failure manually and then bring the affected resource(s) back in service to continue normal LifeKeeper SSP operations.

Important Considerations for Resource-Level Policies

Resource level policies are policies that apply to a specific resource only, as opposed to an entire resource hierarchy or server.

> *Example*:
>
> app
>
> - IP
>
> - file system

In the above resource hierarchy, app depends on both IP and file system. A policy can be set to disable local recovery or failover of a specific resource. This means that, for example, if the IP resource's local recovery fails and a policy was set to disable failover of the IP resource, then the IP resource will not fail over or cause a failover of the other resources. However, if the file system resource's local recovery fails and the file system resource policy does not have failover disabled, then the entire hierarchy will failover causing a reboot.

Note: It is important to remember that resource level policies apply only to the specific resource for which they are set.

This is a simple example. Complex hierarchies can be configured, so care must be taken when setting resource-level policies.

The lkpolicy Tool

The `lkpolicy` tool is the command-line tool that allows management (querying, setting, removing) of policies on servers running LifeKeeper SSP. `lkpolicy` supports setting/modifying policies, removing policies and viewing all available policies and their current settings. In addition, defined policies can be set on or off, preserving resource/server settings while affecting recovery behavior.

The general usage is :

```
lkpolicy [—list-policies | —get-policies | —set-policy | —remove-policy] <name value pair data…>
```

`The <name value pair data…>` differ depending on the operation and the policy being manipulated, particularly when setting policies. For example: Most on/off type policies only require `-on` or —off switch, but the temporal policy requires additional values to describe the threshold values.

Example lkpolicy Usage

Authenticating With Local and Remote Servers

The `lkpolicy` tool communicates with LifeKeeper SSP servers via an API that the servers expose. This API requires authentication from clients like the `lkpolicy` tool. The first time the `lkpolicy` tool is asked to access a LifeKeeper SSP server, if the credentials for that server are not known, it will ask the user for credentials for that server. These credentials are in the form of a username and password and:

1. Clients must have LifeKeeper SSP admin rights. This means the username must be in the lkadmin group according to the operating system's authentication configuration (via pam). It is **not** necessary to run as **root**, but the root user can be used since it is in the appropriate group by default.

2. The credentials will be stored in the credential store so they do not have to be entered manually

each time the tool is used to access this server.

The lkpolicy tool communicates with LifeKeeper SSP servers via an API that the servers expose. This API requires authentication from clients like the lkpolicy tool. The first time the lkpolicy tool is asked to access a LifeKeeper SSP server, if the credentials for that server are not known, it will ask the user for credentials for that server. These credentials are in the form of a username and password and:

See Configuring Credentials for SIOS Protection Suite for more information on the credential store and its management with the `credstore` utility.

An example session with lkpolicy might look like this:

```
[root@thor49 ~]# lkpolicy -l -d v6test4
Please enter your credentials for the system 'v6test4'.
Username: root
Password:
Confirm password:
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]# lkpolicy -l -d v6test4
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]#
```

Listing Policies

```
lkpolicy –list-policy-types
```

Showing Current Policies

```
lkpolicy –get-policies

lkpolicy –get-policies tag=\*

lkpolicy –get-policies –verbose tag=mysql\* # all resources starting with
mysql

lkpolicy –get-policies tag=mytagonly
```

Setting Policies

```
lkpolicy –set-policy Failover –off
```

```
lkpolicy –set-policy Failover –on tag=myresource

lkpolicy –set-policy Failover –on tag=\*

lkpolicy –set-policy LocalRecovery –off tag=myresource

lkpolicy –set-policy NotificationOnly –on

lkpolicy –set-policy TemporalRecovery –on recoverylimit=5 period=15
```

lkpolicy —set-policy TemporalRecovery —on —force recoverylimit=5 period=10

Removing Policies

```
lkpolicy –remove-policy Failover tag=steve
```

Note: *NotificationOnly is a policy alias. Enabling NotificationOnly is the equivalent of disabling the corresponding LocalRecovery and Failover policies.*

12.2.7. Verifying LifeKeeper Single Server Protection Installation

You can verify that the LifeKeeper Single Server Protection packages were installed correctly by entering the following at the command line:

```
rpm -V <package name>
```

Note: If the package is installed correctly, no output will be displayed by this command.

To perform a query from the command line, type

```
rpm -qi <package name>
```

Note: The expected output for this command is the package information.

13. LifeKeeper Single Server Protection for Linux Technical Documentation

About LifeKeeper Single Server Protection for Linux

LifeKeeper Single Server Protection (SSP) allows for application monitoring in a single node configuration (i.e., no cluster requirements or restraints). Single node environments may be physical or virtual (vSphere, KVM, Amazon EC2). LifeKeeper SSP is built on the proven and stable architecture of SIOS LifeKeeper. LifeKeeper SSP provides superior application monitoring and can perform recovery of failed applications and system infrastructure items (e.g., NFS share, IP address, File System). If an application cannot be recovered for some reason, LifeKeeper SSP will initiate a restart of the node via a system reboot or via a VMware HA restart for VMware virtual machines configured for VM and Application Monitoring.

Note: Because LifeKeeper SSP is built using the SIOS LifeKeeper technology, you will see references to LifeKeeper throughout the documentation as well as references to information found in the SIOS Protection Suite for Linux documentation for topics common to both products. When referencing these common topics the following subject items neither apply to nor support LifeKeeper SSP:

- Clustering
- Communication Paths
- Shared Storage (requirements, configuration, …)
- Extending / Unextending resource hierarchies
- Quorum/Witness
- Application Recovery Kits
  - Recovery Kit for EC2
  - LVM Recovery Kit
  - MD Recovery Kit
  - Route53 Recovery Kit
  - SAP Recovery Kit
  - SAP HANA Recovery Kit
  - VMDK as Shared Storage Recovery Kit
- Storage Kits (DR, DMMP, HDLM, LVM, MD, PPATH and NEC SPS)

  Note: Without the underlying storage kits in LifeKeeper SSP, steps must be taken to ensure the devices required to mount protected file systems are activated during system boot (e.g. if the file system is mounted on a logical volume the volume must be in the active state before LifeKeeper SSP starts).

- Resource/Machine failovers (by default with LifeKeeper SSP these result in a node restart)
- Resource switchovers
- Switchable IP Addresses (with LifeKeeper SSP Switchable IP addresses are required for some protected applications but since there is only a single node no switching actually takes place)

  Note: When operating on Amazon EC2, assign a secondary private IP address to the NIC using the Amazon EC2 Management Console prior to creating the IP resource. Next, create the IP resource as the private IP address on the NIC that is using the LifeKeeper GUI client. An Elastic IP can now be associated with the IP resource.

For more information on the SIOS LifeKeeper product, on which LifeKeeper SSP is built, please see the SIOS Protection Suite for Linux documentation for the common release number. This documentation will provide detailed information on resource hierarchy creation, resource types, states and relationships, Graphical User Interface (GUI), as well as common and advanced tasks.

13.1. Documentation and Training

A complete reference providing instructions for installing, configuring, administering and troubleshooting SIOS LifeKeeper Single Server Protection for Linux is available in the LifeKeeper Single Server Protection Documentation. The following sections cover every aspect of SIOS LifeKeeper Single Server Protection for Linux:

Section	Description
Introduction and Installation	Provides useful information for planning and setting up your LifeKeeper Single Server Protection environment, installing and licensing LifeKeeper Single Server Protection and configuring the LifeKeeper graphical user interface (GUI).
Administration	Discusses server-level tasks such as editing server properties and creating resources and resource-level tasks such as editing, extending or deleting resources.
User's Guide	Contains detailed information on the LifeKeeper GUI, including the many tasks that can be performed within the LifeKeeper GUI.
Troubleshooting	Describes known issues and restrictions and suggests solutions to problems that may be encountered during installation, configuration and/or use of SIOS LifeKeeper Single Server Protection for Linux.
Recovery Kits	Contains planning and installation instructions as well as administration, configuration and user information for the Optional Recovery Kits that allow LifeKeeper Single Server Protection to manage and control specific applications.

Training

LifeKeeper Single Server Protection training is available through SIOS Technology Corp. or through your LifeKeeper Single Server Protection provider. Contact your sales representative for more information.

Technical Support

As a SIOS Technology Corp. customer with a valid Support contract, you are entitled to access the new SIOS Technology Corp. Support Self-Service Portal.

The SIOS Technology Corp. Support Self-Service Portal offers you the following capabilities:

- Search our Solution Knowledge Base to find solutions to problems and answers to questions
- Always on 24/7 service with the SIOS Technology Corp. Support team to:
- **Log a Case** to report new incidents
- **View Cases** to see all of your open and closed incidents
- **Review Top Solutions** provides information on the most popular problem resolutions being viewed by our customers.

Contact SIOS Technology Corp. Support at support@us.sios.com to set up and activate your Self-Service Portal account.

You can also contact SIOS Technology Corp. Support at:

1-877-457-5113 (Toll Free)

1-803-808-4270 (International)

Email: support@us.sios.com

13.2. Intergration with VMware HA

As noted in the Introduction Section, LifeKeeper Single Server Protection is designed for use in both physical and virtual environments. When LifeKeeper SSP is installed in a VMware VM the HA features of VMware can be used in conjunction with LifeKeeper SSP to monitor and recover from any protected resource or node failure. To enable these features see Enabling VMware HA Integration with LifeKeeper Single Server Protection. Additionally, LifeKeeper SSP provides an optional component that provides a management interface that integrates with VMware vCenter (see SteelEye Management Console topic).

13.2.1. SteelEye Management Console

The SteelEye Management Console, or SMC, is an optional piece of LifeKeeper Single Server Protection when running in VMware HA configurations. The SMC is a dedicated system that provides a management interface that is integrated with VMware vCenter.

The following topics will help in understanding the setup, installation and operation of a SteelEye Management Console for use with VMware vCenter Server and SIOS LifeKeeper Single Server Protection. The details are broken down into the following categories:

13.2.1.1. Installation Overview

The installation of the SteelEye Management Console, or SMC, consists of several important steps:

1. A server (virtual or physical) must be identified to host the SMC. This server must be a **dedicated system for running the SMC**. SIOS Technology Corp. does not currently support running the SMC on servers being used for other purposes. This server does not need to be very powerful, thus a small virtual machine should be adequate. See System Requirements for more details on server requirements.

2. The SMC software must be installed by running the setup script either from CD media or the `.img` file. This process will make the necessary modifications to the system, install the SMC software components and start the SMC services.

3. The VMware vSphere Client plug-in must be registered with the vCenter server. **Note**: The SMC can only integrate with a single vCenter instance; therefore, if deploying multiple vCenter instances, the SMC software must be installed for each vCenter instance.

4. The SMC must be configured with the credentials required to communicate with each SIOS LifeKeeper Single Server Protection server managed by the vSphere Client (or the subset that will be managed via the SMC). If a certain set of credentials is valid for multiple LifeKeeper Single Server Protection systems, they can be entered once and credentials for the remaining LifeKeeper Single Server Protection nodes will need to be added individually. More details can be found in Configuring Credentials.

13.2.1.2. System Requirements

The SteelEye Management Console, or SMC, must be installed on a dedicated server. The server can be physical or virtual but must not be used for any other purpose. SIOS Technology Corp. does not currently support running the SMC software on servers that are also being used for other purposes. This server must meet the following minimum requirements:

- It must be an Intel (or AMD) 64-bit system capable of running Red Hat Enterprise Linux/CentOS 6.4 ×86_64. This can either be a bare metal system or a virtual machine.
- It must have at least 512MB of RAM.
- It must have at least 8GB of disk space with at least 1GB available for the /opt filesystem.
- It must have at least one network adapter.
- It must be able to communicate directly with, via TCP/IP, a VMware vCenter Server (if using the vSphere Client plug-in) and any LifeKeeper Single Server Protection servers that will be viewed/managed by the SMC. There may be network segment/routing/firewall considerations in choosing a server to host the SMC.
- It must be installed with an openssl-devel (its version must coincide with the version of the openssl-devel included in the installing CD).

Setup Prerequisites

Once a system is chosen, the following prerequisites must be set up prior to installing the SMC software.

The system must be installed with the default, ***base*** software packages and does not require any special package selection.

The system must have the core operating system **yum repository enabled** and available. This means either the install media repository or a network repository for base system must be enabled in */etc/yum.repos.d/*.

Once the system is running a supported operating system, the instructions in Running Setup can be followed to install the SMC software components.

13.2.1.3. Running Setup

The SteelEye Management Console software components can be installed from either a CD/DVD or from the ISO img file containing the CD image. All software procedures from this point must be run as the *root* user.

In either case, the CD or ISO img file must be mounted. The CD can be mounted in the usual way and the img file can be mounted via the loopback device with a command like:

```
mount -o loop <path-to>/smc.img /mnt
```

(/mnt can be any location suitable for mounting the image)

Installing the Software

Once the image is mounted, the installation can be started via:

```
cd /mnt; ./setup
```

The setup tool will guide the installation process and the following will occur to set up the software components:

- System packages will be upgraded and/or removed to ensure that there are no packages on the system that conflict with the SMC components. This process includes removing the pre-installed webserver and components that depend on it. This process can take several minutes.
- Next, the setup tool will install/upgrade SIOS packages for all the SMC software components. This can also take several minutes and will include other required changes to the system, and in particular, changes to the iptables firewall configuration to allow clients to communicate with the SMC. The iptables configuration will be altered to allow HTTP traffic into the SMC server on TCP/IP port 80 and 443.
- Finally, the setup tool will install the required VMware SDK package. This will require agreeing to the VMware SDK end-user license. The SDK install will prompt for a file path for tool binaries to be installed. SIOS Technology Corp. recommends accepting the default location for these files.

Base vCenter and Credential Configuration

After the software components are installed, the setup tool will configure the vSphere Client plug-in and the default credentials for SIOS LifeKeeper Single Server Protection node communication. The following will happen to conclude the setup process:

- The setup tool will ask for the vCenter server name, user and password. This information will be used to register the vSphere Client plug-in provided by this SMC server with the given vCenter. After this step, the vCenter server should provide an additional tab for the plug-in. The plug-in can be re-registered or unregistered anytime after the initial installation and the details of that process

can be found on the Configuring the vSphere Client Plug-in page.

- Last, the setup tool will ask for the ***default*** credentials to use when communicating with SIOS LifeKeeper Single Server Protection nodes. These credentials will be stored on the SMC server and will be used to communicate with LifeKeeper Single Server Protection servers unless credentials specific to the given server have been configured. The ***default*** credentials must have admin access to the LifeKeeper Single Server Protection nodes (on a typical installation, the user must be in the *lkadmin* group in the local */etc/group* file) to allow the SMC to fully manage the LifeKeeper Single Server Protection systems. In general, the default credentials would be the root user and password of the installed LKSSP node. **Note**: While storage of passwords are base64 encoded they are **not** encrypted in the LifeKeeper credstore database. It is advised to use another LKSSP system account that has membership in the lkadmin group.The Configuring Credentials page has more details on how to manage the credentials used by the SMC.

The setup process should now be complete. Please look over the Verifying Installation page for more information on validating that the software was installed and configured correctly.

Credential Considerations

The final step in the setup process above is to store the default credentials for accessing LifeKeeper Single Server Protection systems. Default credentials in this case refers to credentials that will be used to authenticate with LifeKeeper Single Server Protection systems if there are no per-system credentials configured for a system. The SMC will always fall back to using the default credentials.

For this reason, SIOS Technology Corp. recommends using the same credentials on all LifeKeeper Single Server Protection systems whenever possible to simplify the SMC configuration. This typically means using the root user on the LifeKeeper Single Server Protection systems, but any user that is common across all the systems will suffice as long as the user is in the lkadmin group.

13.2.1.4. SIOS LifeKeeper Single Server Protection vSphere Client Plug-in

The SIOS LifeKeeper Single Server Protection vSphere Client plug-in integrates with the VMware vSphere client to provide application monitoring status for protected virtual machines. The plug-in must be registered with the vCenter Server in order to operate.

The plug-in uses HTTPS secure communications for all transfers. It is normal to receive a Security Warning for the **LK4Linux Valid SMC** SSL certificate when first loading the plug-in in the vSphere client. The SSL certificate can be inspected and installed in the local certificate store, and the Security Warning can safely be ignored.

Plug-in Requirements

- VMware vSphere Version 4 or Version 5
- VMware vSphere Client
- Javascript and cookies enabled on Client system

For more information, see the Configuring the vSphere Client Plug-in and the Addressing vSphere Client Plug-in Security Warnings topics.

13.2.1.5. Configuring the vSphere Client Plug-in

The vSphere Client plug-in can be re-registered, or the credentials can be changed, at any time after installation. This includes registering the plug-in with a different vCenter server if needed. If the plug-in is going to be installed on a different vCenter server, it should first be unregistered with the current server.

Registering the vSphere Client Plug-in

Registering the plug-in is done via the `/opt/LifeKeeper/bin/registerPlugin.pl` tool. This tool takes three arguments: the vCenter server, the username and the password. All are required to re-register the vSphere Client plug-in. An example of running this tool would look like (*all on one line*):

```
/opt/LifeKeeper/bin/registerPlugin.pl –server=myvcenter.mydomain.com
–username=vcuser –password=vcpassword
```

Note: Password characters that are also shell characters need to be escaped to avoid shell interpretation.

Unregistering the vSphere Client Plug-in

The plug-ins installed from the SMC server can be listed with the following command:

```
/opt/LifeKeeper/bin/registerPlugin.pl –action=list
```

To remove the plug-in, the following command can be run (all on one line):

```
/opt/LifeKeeper/bin/registerPlugin.pl –action=remove
–key=com.sios.us.lkssp
```

Note: The key used to unregister a plug-in must be the same as shown by the listaction.

Note: For information on security warnings, see Addressing vSphere Client Plug-in Security Warnings.

13.2.1.6. vSphere Client User Interface

SIOS LifeKeeper Single Server Protection Tab

When you register the vSphere Client plug-in, the **LifeKeeper Single Server Protection** tab will appear in the vSphere Client User Interface.

Context Menus

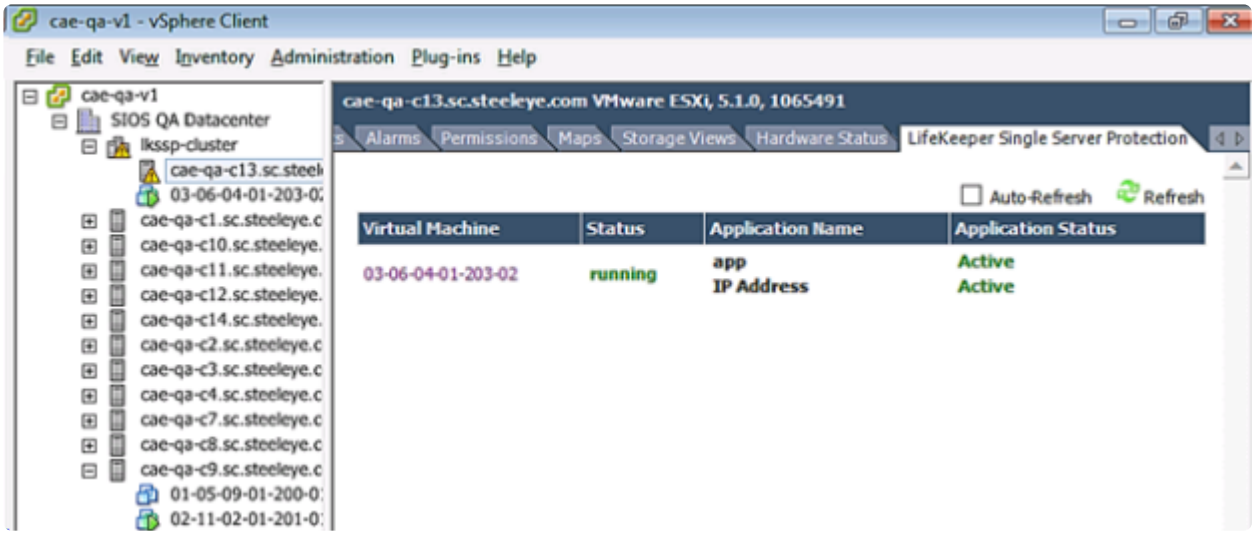
This vSphere Client plug-in offers several context levels for the LifeKeeper Single Server Protection tab depending on where you click in the left-side inventory tree:

Datacenter Level

Displays the virtual machines that are in the datacenter.

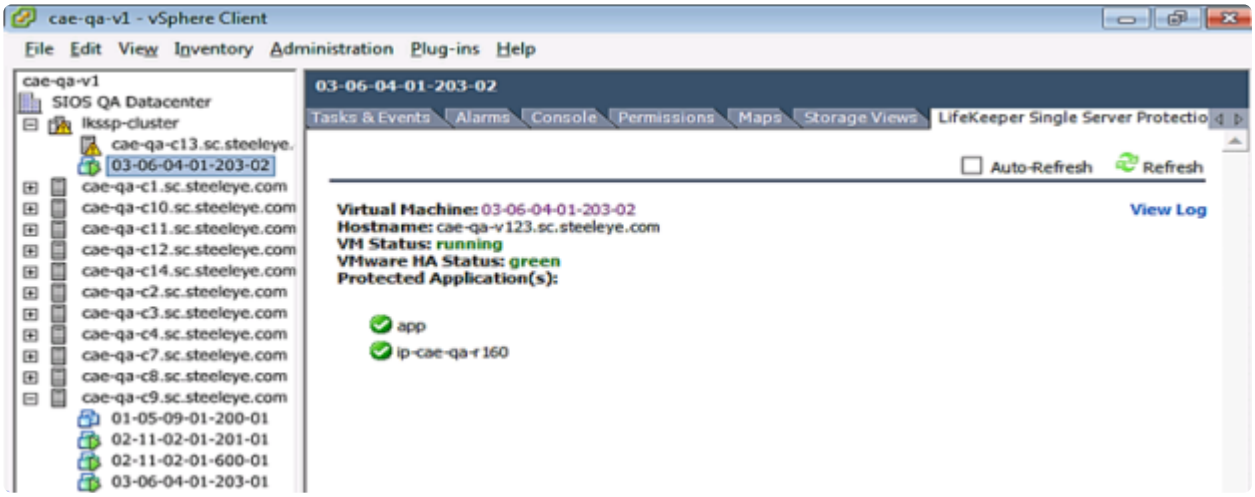
ESX or ESXi Level

Displays the status of the virtual machines running on the particular host.



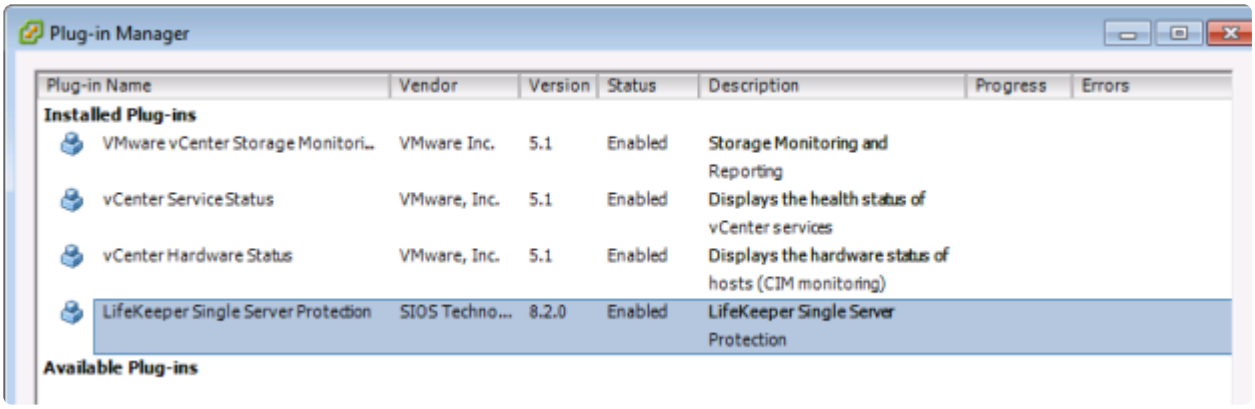
Virtual Machine Level

Displays specific node information: Virtual Machine Name, Hostname, VM Status, VMware HA Status, Protected Applications. There is also a **View Log** link, which allows you to view the LifeKeeper Single Server Protection log.



Manage Plug-Ins

Displays the status of the registered LifeKeeper Single Server Protection vCenter plug-in.



Other Views

The LifeKeeper Single Server Protection plug-in is also viewable at the **Datacenter**, **Virtual Application**, and **Resource Pool** levels.

13.2.1.7. Configuring Credentials

The SMC and LifeKeeper Single Server Protection software manages credentials for communicating with other systems (i.e., vCenter Server or SIOS LifeKeeper Single Server Protection) via a *credential store*. This store is used during plug-in registration (see Configuring the vSphere Client Plug-in) for example. This store can be managed, as needed, by the */opt/LifeKeeper/bin/credstore* command. This command allows server access credentials to be set, changed and removed – on a per server basis.

Adding or Changing Credentials

Adding and changing credentials are handled in the same way. A typical example of adding or changing credentials for a server, *lkssp-server.mydomain.com*, would look like this:

```
/opt/LifeKeeper/bin/credstore -k lkssp-server.mydomain.com myuser
```

In this case, *myuser* is the username used to access *lkssp-server.mydomain.com* and the password will be asked for via a prompt with confirmation (like *passwd*).

Note: The key name used to store LifeKeeper Single Server Protection server credentials on the SMC must match exactly the hostname of the LifeKeeper Single Server Protection server (as displayed in the **Hostname**: field of the vSphere Client Plug-in). If the hostname is an FQDN, then the credential key must also be the FQDN. If the hostname is a short name, then the key must also be the short name.

If following the 'Credential Considerations' suggested in Running Setup, a ***default*** key with an associated username and password will be used for authentication when no specific server keys exist. To add or change the *default* key, run:

```
/opt/LifeKeeper/bin/credstore -k default myuser
```

Listing Stored Credentials

The currently stored credentials can be listed by the following command:

```
/opt/LifeKeeper/bin/credstore -l
```

This will list the keys stored in the credential store and, in this case, the key indicates the server for which the credentials are used. (This command will not actually list the credentials, only the keys, since the credentials themselves may be sensitive.)

Removing Credentials for a Server

Credentials for a given server can be removed with the following command:

```
/opt/LifeKeeper/bin/credstore -d -k lkssp-server.mydomain.com
```

In this case, the credentials store for the server lkssp-server.mydomain.com will be removed from the store.

Additional Information

More information on the credstore utility can be found by running:

```
/opt/LifeKeeper/bin/credstore –man
```

This will show the entire man/help page for the command.

13.2.1.8. Verifying Installation

The SteelEye Management Console installation can be verified by connecting, via a web browser, to `https://<smcserver>/` which should show a page indicating that the SMC services are available.

> **Note:** It is normal to receive a security warning since the SMC uses a self-signed SSL certificate. This warning can safely be ignored.

If the browser indicates an error showing the page or fails to connect to the newly installed SMC server, please ensure all installation steps were completed without error and that the SMC server is network-accessible.

Troubleshooting

For troubleshooting, please see the SMC Troubleshooting section. Also, for information on security warnings, refer to the topic Addressing vSphere Client Plug-in Security Warnings.

13.2.1.9. Addressing vSphere Client Plug-in Security Warnings

The LifeKeeper Single Server Protection vSphere Client Plug-in uses a self-signed certificate to enable SSL communications. It is normal to receive a security warning when viewing the plug-in contents. To reduce or eliminate the security warnings, you should install the “LK4Linux Valid SMC” certificate in your vSphere Client system’s certificate store. Additionally, you can install the “SIOS Technology, Corp.” Certificate Authority (CA) certificate in your system’s “Trusted Root Certification Authorities” certificate store.

To install the “LK4Linux Valid SMC” certificate:

1. When a security warning is displayed, choose **View Certificate**.
2. Click on the **Install Certificate** button.
3. Follow the wizard steps to install the certificate.

To install the “SIOS Technology, Corp.” certificate authority (CA) certificate in the “Trusted Root Certification Authorities” store:

1. When a security warning is displayed, choose **View Certificate**.
2. Click the **Certification Path** tab.
3. Select the **SIOS Technology, Corp.** certificate.
4. Click **View Certificate** to view the CA cert.
5. Click on the **Install Certificate** button.
6. Click **Next**.
7. On the ***Certificate Store Wizard*** pane, select **Place all certificates in the following store** radio button.
8. The **Browse** button will now be enabled. Click it.
9. Select **Trusted Root Certification Authorities** from the **Select Certificate Store** list.
10. Click **OK**.
11. Click **Next** and complete the wizard.

13.2.1.10. LifeKeeper API

The LifeKeeper API is used to allow communications between LifeKeeper Single Server Protection servers and the SteelEye Management Console (SMC). Currently, this API is reserved for internal use only but may be opened up to customer and third party usage in a future release.

Network Configuration

Each LifeKeeper Single Server Protection server provides the API via an SSL Connection on port 778. This port may be changed using the configuration variable API_SSL_PORT in /etc/default/LifeKeeper. This variable is set in /etc/default/LifeKeeper.local.pl on the SMC. (**Note**: This setting controls API client communications to **LifeKeeper Single Server Protection servers**, not access to the SMC itself, which is always on port 443). Both LifeKeeper Single Server Protection and the SMC **must** use the same value for API_SSL_PORT.

Authentication

The LifeKeeper API uses PAM for authentication. Access to the API is only granted to users that are members of the group `lkadmin, lkoper or lkguest`. Depending on the PAM configuration of the system, this can be accomplished by using the local system files (i.e. `/etc/passwd` and `/etc/group`) or by including the user in an LDAP or Active Directory group.

> **Note**: The LifeKeeper API does not use the user database that is managed by the `lkpasswd` utility.

SMC Use of the API

The SMC uses the API to gather information from the LifeKeeper Single Server Protection servers. The SMC uses the credstore utility to manage user account info for LifeKeeper Single Server Protection servers. The SMC uses the LifeKeeper Single Server Protection server name as the key in the credential store, so the system name of the LifeKeeper Single Server Protection server should be passed as the -k option to the credstore utility when specifying credentials for a LifeKeeper Single Server Protection server. The SMC will also check for and use credentials stored in the ***default*** key if it does not find credentials for a specific server.

13.2.1.11. Using Custom Certificates

LifeKeeper Single Server Protection uses SSL/TLS to communicate between different systems. By default, the product is installed with default certificates that provide some assurance of identity between nodes. This document explains how to replace these default certificates with certificates created by your own Certificate Authority (CA).

How Certificates Are Used

Communication to the SteelEye Management Console (SMC) and LifeKeeper Single Server Protection servers uses SSL/TLS to protect the data being transferred. Both systems provide a certificate to identify themselves, and both systems use a CA certificate to verify the certificate that is presented to them over the SSL connection.

Four certificates are involved:

- `/opt/LifeKeeper/etc/certs/LK4LinuxValidNode.pem` (LifeKeeper Single Server Protection server certificate)
- `/opt/LifeKeeper/etc/certs/LK4LinuxValidSMC.pem` (SMC server certificate)
- `/opt/LifeKeeper/etc/certs/LK4LinuxClient.pem` (LifeKeeper Single Server Protection client certificate, installed on all servers)
- `/opt/LifeKeeper/etc/certs/LKCA.pem` (certificate authority, installed on all servers)

The first three certificates must be signed by the fourth certificate to satisfy the verification performed by the servers. Note that the common name of the certificates is not verified, only that the certificates are signed by the CA.

Using Your Own Certificates

In some installations, it may be necessary to replace the default certificates with certificates that are created by an organization's internal CA. If this is necessary, replace the four certificates listed above with new certificates using the same certificate file names. These certificates are of the PEM type. The LK4LinuxValidNode.pem, LK4LinuxValidSMC.pem and LK4LinuxValidClient.pem each contain both their respective key and certificate. The `LK4LinuxValidNode.pem` and `LK4LinuxValidSMC.pem` certificates are server type certificates. `LK4LinuxValidClient.pem` is a *client* type certificate.

If the default certificates are replaced, LifeKeeper Single Server Protection and SMC will need to be restarted to reflect the changes. If the certificates are misconfigured, `steeleye-lighttpd` daemon will not start successfully and errors will be received in the LifeKeeper Single Server Protection log file. If problems arise, refer to this log file to see the full command that should be run.

13.3. Administration

LifeKeeper Single Server Protection Administration Overview

LifeKeeper Single Server Protection does not require administration during operation. LifeKeeper Single Server Protection works automatically to monitor protected resources and to perform the specified recovery actions if a fault should occur. You use the LifeKeeper Single Server Protection GUI in these cases:

- **Resource and hierarchy definition**. LifeKeeper Single Server Protection provides these interface options:
    - º LifeKeeper Single Server Protection GUI.
    - º LifeKeeper Single Server Protection command line interface.
- **Resource monitoring**. The LifeKeeper Single Server Protection GUI provides access to resource status information and to the LifeKeeper Single Server Protection logs.
- **Manual intervention**. You may need to stop servers or specific resources for maintenance or other administrative actions. The LifeKeeper Single Server Protection GUI provides menu functions that allow you to bring specific resources in and out of service. Once applications have been placed under LifeKeeper Single Server Protection, they should be started and stopped only through these LifeKeeper Single Server Protection interfaces. Starting and stopping LifeKeeper Single Server Protection is done through the command line only.

See Administration Tasks, GUI Tasks, and Maintenance Tasks in the SPS for Linux documentation for detailed instructions on performing administration, configuration, and maintenance operations including the creation of resource hierarchies.

Note: All LifeKeeper executable scripts and programs run via the command line require super user authority. A super user (granted permissions by running the “su” or “sudo” command) is able to execute LifeKeeper commands. However, SIOS Technology Corp has tested executing LifeKeeper commands via the root user only.

13.3.1. Enabling VMware HA Integration with LifeKeeper Single Server Protection

By default LifeKeeper Single Server Protection integration with VMware HA is disabled when installed on a VMware VM. To enable integration requires the following steps:

1. Installation of VMware tools in the LifeKeeper Single Server Protection VM.
2. Edit */etc/default/LifeKeeper* and change the VMware HA integration tunable `HA_DISABLE` value form 1 to 0.
3. Restart LifeKeeper Single Server Protection. If LifeKeeper Single Server Protection is currently running, it must be stopped and restarted to pick up the above change in */etc/default/LifeKeeper.*
4. Installation of the SteelEye Management Console (optional step).

13.3.2. Enabled VMware HA Fault Detection and Recovery Scenario

The ability to provide detection and alarming for problems within an application is critical to building the best total fault resilient solution. Since every specific application varies on the mechanism and format of failures, no one set of generic mechanisms can be supplied. In general, however, many application configurations can rely on the Core system error detection provided within LifeKeeper Single Server Protection. This topic demonstrates the power of LifeKeeper Single Server Protection's core facilities.

Below is a recovery scenario to demonstrate how LifeKeeper Single Server Protection provides fault detection and recovery when an application fails.

1. LifeKeeper Single Server Protection will first attempt recovery by trying to restart the application.

2. If the recovery succeeds, the application should continue to run normally.

3. If the recovery attempt fails:

    a. If the LifeKeeper Single Server Protection recovery attempt fails, and LifeKeeper Single Server Protection is installed in a VMware guest OS with HA enabled (HA_DISABLE=0 in /etc/default/LifeKeeper), then LifeKeeper Single Server Protection will trigger VMware HA by withholding the heartbeat that LifeKeeper Single Server Protection sends down to the Application Monitoring Interface. VMware HA will then respond by restarting the server.

    b. If the LifeKeeper Single Server Protection recovery attempt fails, and LifeKeeper Single Server Protection is not installed in a VMware guest OS or is installed in a VMware guest OS but has HA disabled (HA_DISABLE=1 in /etc/default/LifeKeeper), then a system reboot will be forced.

Optionally, LifeKeeper Single Server Protection can be placed in **Notification Only** mode. In this mode the automatic triggering of a system reboot is disabled (see the section VMware HA and **Notification Only** Mode below). In **Notification Only** mode you must log into the system and correct the issue that caused failure.

VMware HA and Notification Only Mode

1. In **Notification Only** mode with HA enabled in the VMware guest OS and the LifeKeeper SSP vCenter plugin installed, when a failure is detected, LifeKeeper Single Server Protection will not attempt to restart the application. Instead, the resource will be marked as **Failed**. The vCenter plugin dashboard view status screen will show failure (**Application Status: Failed**).

Application Name	Application Status
Login failed: Couldn't resolve host name (6)	N/A
Apache	**Failed**
Login failed: Couldn't connect to server (7)	N/A
Login failed: Couldn't connect to server (7)	N/A
N/A	N/A
Login failed: Couldn't resolve host name (6)	N/A

2. Log in to the server and correct the issue that caused the failure.

3. Open the **LifeKeeper Admin Console** either through the CLI or by clicking on the protected virtual machine within the **vSphere Client User Interface**.

4. Bring the application back in service.

5. Go to the **dashboard** view within the **vSphere Client User Interface**.

6. Click **Refresh**. **Application status** goes back to **Active**.

Application Name	Application Status
Login failed: Couldn't resolve host name (6)	N/A
Apache	**Active**
Login failed: Couldn't connect to server (7)	N/A
Login failed: Couldn't connect to server (7)	N/A
N/A	N/A
Login failed: Couldn't resolve host name (6)	N/A

13.3.3. LifeKeeper Single Server Protection Heartbeat with VMware HA

The LifeKeeper Single Server Protection heartbeat is the signal sent to VMware HA (every 10 seconds if running in a VMware guest OS and if HA is enabled) indicating that the protected applications are OK. If an application fails, LifeKeeper Single Server Protection will first attempt to recover the application. If recovery fails, LifeKeeper Single Server Protection will withhold the heartbeat, which instructs VMware HA to reboot the VM.

13.3.4. Maintaining a LifeKeeper Single Server Protection Protected System

When performing system or application maintenance on a LifeKeeper Single Server Protection-protected server, you should either stop LifeKeeper Single Server Protection monitoring or place the protected resources into maintenance mode. This will stop LifeKeeper Single Server Protection from interfering with the system and application maintenance tasks by disabling both application recovery and triggering of VMware HA failure events.

To stop and restart LifeKeeper Single Server Protection:

1. **Stop LifeKeeper Single Server Protection**. Stop LifeKeeper Single Server Protection using the ***/opt/LifeKeeper/bin/lkstop -f*** command. The resources will remain running but will no longer be monitored by LifeKeeper Single Server Protection. Any failure will have to be handled manually.

2. **Perform maintenance**. Perform the necessary maintenance.

3. **Start LifeKeeper Single Server Protection**. Use the command ***/opt/LifeKeeper/bin/lkstart*** to start LifeKeeper Single Server Protection. Your resources are now protected.

Alternatively, place the resources in maintenance (a.k.a., notification only) mode:

1. **Place all resources in maintenance mode**. Use the command ***/opt/LifeKeeper/bin/lkpolicy -s NotificationOnly --On***. Resources will not be recovered and VMware HA failure events will not be triggered.

2. **Perform maintenance**. Perform the necessary maintenance.

3. **Turn off maintenance mode**. Use the command ***/opt/LifeKeeper/bin/lkpolicy -s NotificationOnly --Off***. Resources are now protected.

13.3.5. Quick Service Protection (QSP) Recovery Kit

The QSP Recovery Kit provides a simplified method to protect the OS service. With the QSP Recovery Kit users can easily create a LifeKeeper resource instance to protect an OS service provided that service can be started and stopped by the OS service command. The service can also be protected via the Generic Application Recovery Kit, but the use of that kit requires code development whereas the QSP Recovery Kit does not. Also, by creating a dependency relationship, protected services can be started and stopped in conjunction with the application that requires the service. The application is protected by another LifeKeeper resource instance not via the QSP resource.

However, the QSP Recovery Kit quickCheck can only perform simple health (using the “status” action of the service command). QSP doesn’t guarantee that the service is provided or the process is functioning. If complicated starting and/or stopping is necessary, or more robust health checking operations are necessary, using a Generic Application is recommended.

For details, please refer to the following URL:

- Quick Service Protection (QSP) Recovery Kit

13.3.6. LifeKeeper API for Monitoring

Introduction

The LifeKeeper API for Monitoring can obtain the operational status of LifeKeeper nodes and their protected resources by making status inquiries to the available nodes in the LifeKeeper cluster.

Summary

This document describes the LifeKeeper API for Monitoring (hereinafter referred to as the API) and is targeted for developers who manage the resource protected by LifeKeeper.

By using the API, the information supplied by the lcdstatus command is obtained through CGI script and the lighttpd module. By using this API, users can determine the current status of the LifeKeeper nodes and resources without logging-in to LifeKeeper servers.

The API can supply the following information.

- ◦ LifeKeeper node status is the node alive and processing or down
- ◦ Communication path status between nodes in the cluster, are communication path(s) up or down
- ◦ Status of protected resources

To get the detailed status of any abnormal condition requires logging-in to LifeKeeper GUI or checking the LifeKeeper log as necessary.

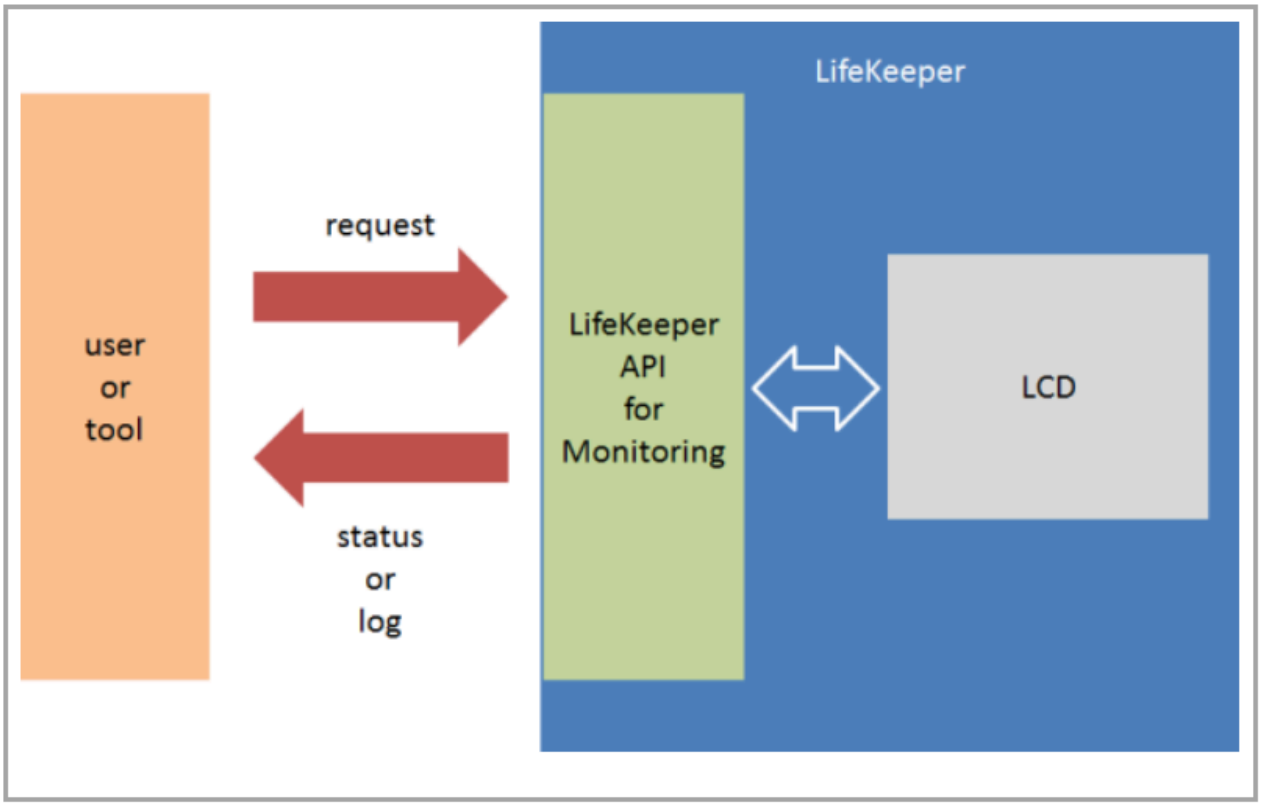


Chart 1. Overview of the LifeKeeper API for Monitoring

Information to be supplied with this API

The following information is supplied through this API when the user makes an inquiry to an active LifeKeeper node. The information supplied is about the specific LifeKeeper server to which the inquiry was directed even if the cluster consists of multiple servers.

- ◦ Status
- ◦ Operating status of each server
- ◦ Node name
- ◦ Operational status (ALIVE/DEAD)
- ◦ Operational status of communication path(s)
- ◦ Node name
- ◦ Operational status (ALIVE/DEAD)
- ◦ Address / device name
- ◦ Status of protected resources
- ◦ Node Name

- ◦ Tag
- ◦ Status (ISP, OSU, OSF, …)
- ◦ Dependency setting
- ◦ Mirror information for Data Replication resources (available only if status is ISP)
- ◦ Replication status (75%, 100%, …)
- ◦ Mirror status (Sync/Async, Paused, …)
- ◦ Log
- ◦ /var/log/lifekeeper.log
- ◦ Up to 1000 lines (when data output format is HTML)
- ◦ All (when data output format is plain text)
- ◦ Not supported if log file path is changed
- ◦ /var/log/lifekeeper.err
- ◦ Up to 1000 lines (when data output format is HTML)
- ◦ All (when data output format is plain text
- ◦ Not supported if log file path is changed

Communication format

The API uses HTTP to obtain the requested information. To obtain information, the user sends a HTTP GET request to the CGI scripts via lighttpd on the specific server.

Data format

The following 3 data formats are available.

- ◦ JSON
- ◦ To be used by an external tool to analyze the status information returned
- ◦ Status checking is possible
- ◦ Log output is not available

- ◦ HTML
- ◦ To be used to visually check via a browser
- ◦ Status checking is possible
- ◦ Log information is available up to 1000 lines
- ◦ plain text
- ◦ Used for regular log checking
- ◦ For logging purpose only and not for checking the status
- ◦ All contents of /var/log/lifekeeper.log and /var/log/lifekeeper.err are available

Available JSON format from the status in figure 2 is shown in figure 3, and HTML format is in figure 4

Figure 2. Example of active LifeKeeper configuration

```
{
   "resource": [
       {
           "replication": {},
           "child": [
```

```
            {
                "tag": "datarep-data"
            }
        ],
        "server": {
            "status": "ISP",
            "name": "lk01"
        },
        "tag": "/data"
    },
    {
        "replication": {
            "percent": "100%",
            "mirror": "Fully Operational"
        },
        "child": [],
        "server": {
            "status": "ISP",
            "name": "lk01"
        },
        "tag": "datarep-data"
    },
    {
        "replication": {},
        "child": [],
        "server": {
            "status": "ISP",
            "name": "lk01"
        },
```

```
            "tag": "ip-10.125.139.118"
        }
    ],
    "compath": [
        {
            "status": "ALIVE",
            "server": [
                {
                    "name": "lk01",
                    "term": "192.168.139.18"
                },
                {
                    "name": "lk02",
                    "term": "192.168.139.19"
                }
            ]
        },
        {
            "status": "ALIVE",
            "server": [
                {
                    "name": "lk01",
                    "term": "172.20.139.18"
                },
                {
                    "name": "lk02",
                    "term": "172.20.139.19"
                }
            ]
```

```
        }
    ],
    "server": [
        {
            "status": "ALIVE",
            "name": "lk01"
        },
        {
            "status": "ALIVE",
            "name": "lk02"
        }
    ]
```

}

Figure 3. Status output example using the JSON data format

RESOURCEs

tag	lk01
/data	ISP
datarep-data	ISP
ip-10.125.139.118	ISP

DATA REPLICATIONs

tag	nodes	mirror status	replication status
datarep-data	lk01 -> lk02	Fully Operational	100%

COMMUNICATION PATHs

communication path	status
192.168.139.18/192.168.139.19	ALIVE
172.20.139.18/172.20.139.19	ALIVE

Figure 4. Status output using the HTML format

How to use

Activate the API

The API is disabled by default. To activate, requires modification of /etc/default/LifeKeeper set the LKAPI_MONITORING configuration parameter to true (see figure 5). Setting of the configuration parameter only activates the API on that node and therefore must be set on each node on which the API will be used. Setting of this configuration parameter does not require a restart of LifeKeeper..

```
LKAPI_MONITORING=true
```

Figure 5. Enabling the LifeKeeper API for Monitoring

Port number

The API uses port 779 by default. To change the port number, the user needs to set the following in /etc/default/LifeKeeper.

```
LKAPI_WEB_PORT=<port number>
```

Figure 6. Change the port number for LifeKeeper API for Monitoring

Usage examples

To obtain information a request is made to a server with an active LifeKeeper API configuration. Basic example using curl.

```
curl http://<IPADDR>:779/Monitoring.cgi
```

If no arguments are given, the current status is obtained using the JSON data format.

Request for log information using HTML data format.

```
curl http://<IPADDR>:779/Monitoring.cgi?format=html&show=log
```

The list of available arguments can be found in the table below.

List 1. Arguments

Name	Explanation	Value	Comments
show	Specify the target information	status, log, log-err	show=status is the default
format	Specify data format	json, html, plain	format=json is the default. If the format is json an error will be displayed if show=log or show=log-err is set.

Security

All the users requesting information via the API must be authorized to get LifeKeeper status information.

For this reason, user security settings can limit the users who can get the status by, configuring SSL, and encrypting the information.

Basic Authentication

To obtain the information via the API, Basic Authentication is required. To setup the authentication requires modification to the lighttpd configuration file (Modify the part in red colored character.) plus a restart of the lighttpd module. See figure 7 for how to configure lightttp.conf.

After modification execute the command ”/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd” and reboot lighttpd to restart lighttpd using the new configuration.

/opt/LifeKeeper/etc/lighttpd/lighttpd.conf

```
server.modules              = (
    :
                            "mod_auth", # uncommenting
```

/opt/LifeKeeper/etc/lighttpd/conf.d/lkapi_user.conf

```
print qq/\$SERVER["socket"] == ":$lkapi_port" {\n/;
print qq/    server.document-root = "\/opt\/LifeKeeper\/api"\n/;
print qq/    auth.backend = "htpasswd"\n/;
print qq/    auth.backend.htpasswd.userfile =
"\/opt\/LifeKeeper\/etc\/lighttpd\/lighttpd.user.htpasswd"\n/;
print qq/    auth.require = ( "\/" =>\n/;
print qq/        (\n/;
print qq/                "method"  => "basic",\n/;
print qq/                "realm"   => "LifeKeeperAPI",\n/;
print qq/                "require" => "valid-user"\n/;
print qq/        )\n/;
print qq/    )\n/;
print qq/ }\n/;
```

Step to create htpasswd file.

htpasswd -c /opt/LifeKeeper/etc/lighttpd/lighttpd.user.htpasswd USERNAME

Figure 7. Basic Authentication setting example

SSL/TLS set up

SSL/TLS is available for the communication via this API. The lighttpd modifications for SSL/TLS is shown in the example in Figure 8. After modification execute the command ”/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd” and reboot lighttpd to restart lighttpd with the new configuration.

/opt/LifeKeeper/etc/lighttpd/include_ssl_port.pl

```
configAPI("0.0.0.0", 443);
if(socket($sock, AF_INET6, SOCK_STREAM, 0)) {
    configAPI("[::]", 443);
}
sub configAPI {
    my $addr = shift;
    my $port = shift;

    print qq/\$SERVER["socket"] == "$addr:$port" {\n/;
    print qq/    server.document-root = "\/opt\/LifeKeeper\/api"\n/;
    print qq/    ssl.engine  = "enable"\n/;
    print qq/    ssl.pemfile = "\/opt\/LifeKeeper\/etc\/certs\/LK4LinuxValidNode.pem"\n/;
    print qq/    ssl.use-sslv2 = "disable"\n/;
    print qq/    ssl.use-sslv3 = "disable"\n/;
    print qq/ }\n/;
}
```

Figure 8 SSL/TLS setting example

Modification to support SSL/TLS + Basic authentication

Using SSL/TLS, modification example to set up Basic authentication is below. After modification, execute the command “/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd“ and restart lighttpd to reflect the modified set up.

/opt/LifeKeeper/etc/lighttpd/lighttpd.conf

```
server.modules              = (
  :
                        "mod_auth", # uncommenting
```

/opt/LifeKeeper/etc/lighttpd/include_ssl_port.pl

```
configAPI("0.0.0.0", 443);
if(socket($sock, AF_INET6, SOCK_STREAM, 0)) {
  configAPI("[::]", 443);
}
sub configAPI {
  my $addr = shift;
  my $port = shift;

  print qq/\$SERVER["socket"] == "$addr:$port" {\n/;
  print qq/    server.document-root = "\/opt\/LifeKeeper\/api"\n/;
  print qq/    ssl.engine  = "enable"\n/;
  print qq/    ssl.pemfile = "\/opt\/LifeKeeper\/etc\/certs\/LK4LinuxValidNode.pem"\n/;
  print qq/    ssl.use-sslv2 = "disable"\n/;
  print qq/    ssl.use-sslv3 = "disable"\n/;
  print qq/    auth.backend = "htpasswd"\n/;
  print qq/    auth.backend.htpasswd.userfile =
"\/opt\/LifeKeeper\/etc\/lighttpd\/lighttpd.user.htpasswd"\n/;
  print qq/    auth.require = ( "\/" =>\n/;
  print qq/       (\n/;
  print qq/                "method"  => "basic",\n/;
  print qq/                "realm"   => "LifeKeeperAPI",\n/;
  print qq/                "require" => "valid-user"\n/;
  print qq/       )\n/;
  print qq/    )\n/;
  print qq/ }\n/;
}
```

Figure 9. SSL/TLS+Basic Authentication set up example

IP address access limitation

The lightppd configuration can also be setup to limit IP addresses that can be used to access data via the API. The lighttpd configuration to limit access is shown in Figure 9. The example will reject the connections from IP address other than 192.168.10.1. After modification execute the command ”/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd” to restart lighttpd with the new configuration.

/opt/LifeKeeper/etc/lighttpd/conf.d/lkapi_user.conf

```
$HTTP["remoteip"] != "192.168.10.1" {
  url.access-deny = ( "" )
}
```

Figure 10. IP access limit setting example

Error

Errors can occur during the usage of this API when enabled. Should this occur, the summary of the error is output. Error example when JSON format is shown below.

HTTP status code returned by lighttpd is not described here.

```
{
    "error" : {
    id : -1,
    message : "Failed to get LCD status"
  }
}
```

Figure 11. Error output example

Similar message is output in the case the output format is HTML.

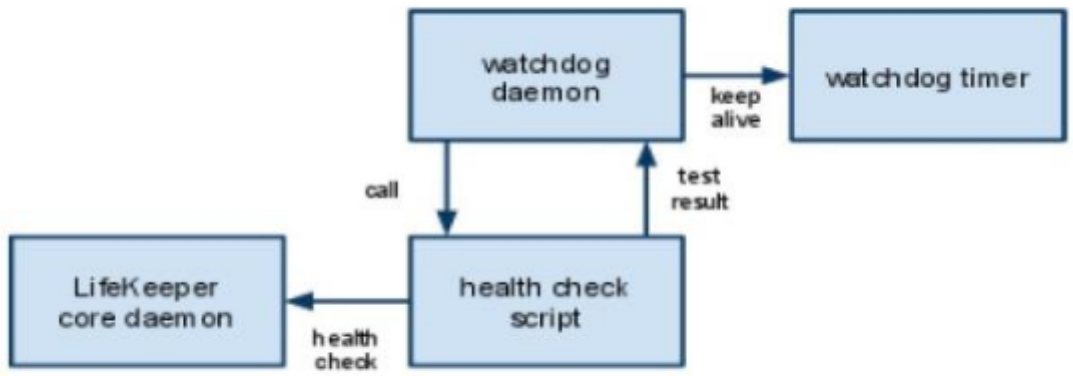
13.3.7. Watchdog

Watchdog is a method of monitoring a server to ensure that if the server is not working properly, corrective action (reboot) will be taken so that it does not cause problems. Watchdog can be implemented using special watchdog hardware or using a software-only option.

> **Note**: This configuration has only been tested with Red Hat Enterprise Linux Versions 6 and 7. No other operating systems have been tested; therefore, no others are supported at this time.

Components

- Watchdog timer – software driver or an external hardware component
- Watchdog daemon – rpm available through the Linux distribution
- LifeKeeper core daemon – installed with the LifeKeeper installation
- Health check script – Script to check the status of LifeKeeper SSP core



LifeKeeper Interoperability with Watchdog

Read the next section carefully. The daemon is designed to recover from errors and will reset the system if not configured carefully. Planning and care should be given to how this is installed and configured. This section is not intended to explain and configure watchdog, but only to explain and configure how LifeKeeper SSP interoperates in such a configuration.

Configuration

The following steps should be carried out by an administrator with root user privileges. The administrator should already be familiar with some of the risks and issues with watchdog.

The health check script (LifeKeeper monitoring script) is the component that ties the LifeKeeper configuration with the watchdog configuration *(/opt/LifeKeeper/samples/watchdog/LifeKeeper-watchdog*). This script can monitor the basic parts of LifeKeeper core components.

1. If watchdog has been previously configured, enter the following command to stop it. If not, go to Step 2.

   ```
   service watchdog stop (RHEL6)
   ```

   ```
   systemctl stop watchdog (RHEL7)
   ```

2. Edit the watchdog configuration file (/etc/watchdog.conf) supplied during the installation of watchdog software.

   - Modify test-binary:

     ```
     test-binary = /opt/LifeKeeper/samples/watchdog/LifeKeeper-watchdog
     ```

   - Modify test-timeout:

     ```
     test-timeout = 5
     ```

   - Modify interval:

     ```
     interval = 7
     ```

     The interval must be greater than or equal to the test-timeout value. The recommended value is between 5 and 10 because if the interval is too long, failure detection will be delayed.

3. Make sure LifeKeeper SSP has been started. If not, please refer to the Starting LifeKeeper topic.

4. Start watchdog by entering the following command:

   ```
   service watchdog start (RHEL6)
   ```

   ```
   systemctl start watchdog (RHEL7)
   ```

5. To start watchdog automatically on future restarts, enter the following command:

   ```
   chkconfig —levels 35 watchdog on (RHEL6)
   ```

   ```
   systemctl enable watchdog (RHEL7)
   ```

Note: Configuring watchdog may cause some unexpected reboots from time to time. This is the general nature of how watchdog works. If processes are not responding correctly, the watchdog feature will assume that LifeKeeper (or the operating system) is hung, and it will reboot the system (without warning).

Uninstall

Care should be taken when uninstalling LifeKeeper. The above steps should be done in reverse order as listed below.

> **WARNING**: IF UNINSTALLING LIFEKEEPER BY REMOVING THE RPM PACKAGES THAT MAKE UP LIFEKEEPER, **TURN OFF WATCHDOG FIRST!** In Step 2 above, the watchdog config file was modified to call on the LifeKeeper-watchdog script; therefore, if watchdog is not turned off first, it will call on that script that is no longer there. An error will occur when this script is not found which will trigger a reboot. This will continue until watchdog is turned off.

1. Stop watchdog by entering the following command:

   ```
   service watchdog stop (RHEL6)

   systemctl stop watchdog (RHEL7)
   ```

2. Edit the watchdog configuration file (/etc/watchdog.conf) supplied during the installation of watchdog software.

   - Modify test-binary and interval by commenting out those entries (add # at the beginning of each line):

     ```
     #test-binary =
     #interval =
     ```

> **Note**: If interval was used previously for other functions, it can be left as-is

3. Uninstall LifeKeeper. See the Removing LifeKeeper topic.

4. Watchdog can now be started again. If only used by LifeKeeper, watchdog can be permanently disabled by entering the following command:

   ```
   chkconfig –levels 35 watchdog off (RHEL6)

   systemctl disable watchdog (RHEL7)
   ```

13.3.8. LKCLI (LifeKeeper Command Line Interface)

LKCLI is not supported in Single Server Protection for Linux v9.5.0.

13.4. FAQs

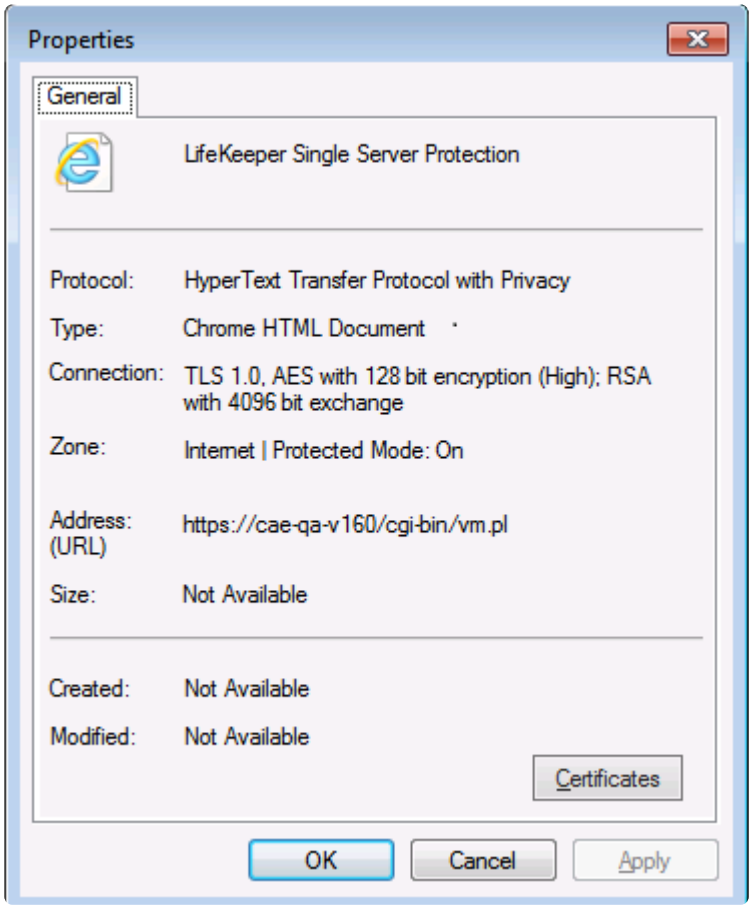
SMC

Question

Is there any way (from the plug-in) to tell which SMC I am using?

Answer

Right-click and view **Properties** of the plug-in web page.



13.5. Troubleshooting

This section contains restrictions and/or known issues open against LifeKeeper Single Server Protection as well as SMC troubleshooting hints and tips.

For more troubleshooting information, see the LifeKeeper Technical Notes and Troubleshooting topics in the LifeKeeper Technical Documentation.

13.5.1. Known Issues and Workarounds

Included below are the restrictions and/or known issues open against LifeKeeper Single Server Protection.

Product Incompatibility

SIOS AppKeeper

Because SIOS LifeKeeper monitors and remediates OS and application services in clustered environments in AWS EC2, the use of SIOS AppKeeper in addition to SIOS LifeKeeper in those environments is not recommended or supported.

Core

Bug 2257 **Access to LifeKeeper Single Server Protection and SIOS Protection Suite nodes via** **credstore requires proper credstore key** **Solution:** When storing credentials for a LifeKeeper Single Server Protection or SIOS Protection Suite node using `credstore`, you must use the proper form of the hostname for the credstore credentials key (`i.e. credstore -k`): For the LifeKeeper Single Server Protection	

plugin, `credstore` should be run using the hostname of the system as reported in the **Hostname**: field of the LifeKeeper Single Server Protection plugin display. For SIOS Protection Suite, the hostname used to store credentials must be the same as the one you plan to use in the command line tool's (e.g., `lkpolicy) -d` argument. For example, if you want to run **`lkpolicy -d mynode1`**, then you must store credentials using **`credstore -k mynode1`**. You cannot store credentials using the FQDN in this case. If you do, you must run **`lkpolicy -d`** *FQDN*. **Workaround:** If you've stored a default credential set (i.e., **`credstore -k default`**) that works for all your LifeKeeper Single Server Protection and/or SIOS Protection Suite nodes, then you will not be affected by this issue.	
Bug 2408 **HA hearbeat incorrectly**	

enabled lkvmhad incorrectly enables the HA hearbeat after second resource failure **Workaround:** Set `LKCHECKINTERVAL` in */etc/default/LifeKeeper* greater than the VMware HA, VM Monitoring Failure Interval. **Note:** The `LKCHECKINTERVAL` default is 120 seconds. This is also the default 'low' monitoring sensitivity for VMware HA, VM Monitoring.	
Cannot install the SMC when an openssl-devel has not been installed An openssl-devel must be installed in advance when installing SMC v8.3.2. If the openssl-devel has not been installed, the install of SMC will fail outputting the message as follows: ld -shared -o ./lib/Crypt-SSLeay-0.55-0.9.8/lib/auto/Crypt/SSLeay/SSLeay.so ./lib/Crypt-SSLeay-0.55-0.9.8/lib/auto/Crypt/SSLeay/SSLeay.o -lcrypto -lssl ld: cannot find -lcrypto	

Unable to link the Crypt::SSLeay Perl module. Secured connections will be unavailable until you install the Crypt::SSLeay module. So required libcrypt.a in system library. **Workaround:** When the message above is shown in executing the setup script, you must execute the setup script again after installing the openssl-devel included in the OS installer.	

GUI

Refresh problem with LifeKeeper Single Server Protection GUI The GUI may occasionally scramble the resource tree (i.e., resource dependencies may not be shown correctly). **Workaround:** Perform a refresh of the GUI.	

Apache

Apache resource	

creation fails **Example of Error message:** Error: valid_http_root: Since “/usr/sbin/httpd” is shareable on “/usr”, “/etc/httpd” must be also **Cause:** Due to a defect, files in mount point “/”(root) cannot be detected appropriately. For example, if “/etc/httpd” is in a same filesystem as the mount point “/”, a resource creation will fail. **Workaround:** Mount one of the below workarounds to avoid this issue. (a) Transfer such as “/etc/httpd” under the other mount point. (b) Mount “ /etc” to such as” /dev/sdb1”.	

Oracle

Bug 2387 **Cannot create an**	

Oracle hierarchy on root file system in LifeKeeper Single Server Protection environment **Workaround:** Using the following procedure, copy Oracle to a new file system. Create a new disk large enough for Oracle data (e.g. /dev/sdb). (Note: You can size up /oracle directory to get an idea how big this should be; multiply by at least 50% to allow for logs) Using gdisk, create a new partition on that disk. `gdisk /dev/sdb` Make a file system. mkfs -t ext3 /dev/sdb1 Mount this file system (example using /mnt/oracle). mkdir /mnt/oracle mount /dev/sdb1 /mnt/oracle Stop Oracle, Listener. Copy Oracle to new file system.	

cd /oracle cp -a * /mnt/oracle (**Note:** This step may take some time based on the amount of data) Unmount the new file system. umount /mnt/oracle Mount the new file system over /oracle. mount /dev/sdb1 /oracle Start Listener and then Oracle.	

SAP

Bug 2388 **For SAP, hierarchies cannot be created using the GUI** **Workaround:** Use the command line option to create hierarchies. However, at the end of the command line, specify the number 76 as follows: • $LKROOT/lkadm/subsys/appsuite/sap/bin/create <primary sys> <tag> <SAP SID>	

<SAP Instance> <switchback type> <IP Tag> <Protection Level> <Recovery Level> <Additional SAP Dependents> 76 See Setting Up SAP from the Command Line for further command line information.	

Also, refer to Known Issues and Restrictions.

13.5.2. SMC Troubleshooting

See below for troubleshooting hints and tips.

LifeKeeper Single Server Protection monitored guest virtual machines must have VMware Tools installed. LifeKeeper Single Server Protection is unable to completely connect to virtual machines that do not have working installations of VMware tools. SteelEye Management Console will not be able to connect to these machines to get complete status of the machine and its resources. The LifeKeeper Single Server Protection plugiin in the vSphere Client will show error messages for these guests (where the tools are not installed). **Solution:** Make sure the VMware Tools are installed on the guest machines that will be protected by LifeKeeper Single Server Protection.	

13.6. Application Recovery Kits

LifeKeeper Single Server Protection for Linux Application Recovery Kits (ARKs) include tools and utilities that allow SSP to manage and control a specific application. The following optional recovery kits are available with this release of SSP..

- Apache Recovery Kit Administration Guide
- DB2 Recovery Kit Administration Guide
- IP Recovery Kit Administration Guide
- MySQL Recovery Kit Administration Guide
- WebSphere MQ Recovery Kit Administration Guide
- NAS Recovery Kit Administration Guide
- NFS Recovery Kit Administration Guide
- Oracle Recovery Kit Administration Guide
- PostgreSQL Recovery Kit Administration Guide
- Postfix Recovery Kit Administration Guide
- Samba Recovery Kit Administration Guide
- SAP MaxDB Recovery Kit Administration Guide
- Sybase Recovery Kit Administration Guide