# SIOS Protection Suite for Linux

9.5.2 — 最終更新日時: 27 May 2022

SIOS TECHNOLOGY CORP.

Copyright © 2022 SIOS TECHNOLOGY CORP.

# 目次

**1. SIOS Protection Suite for Linux** ..... **24**

**2. SIOS Protection Suite for Linux** リリースノート ..... **25**

**3. LifeKeeper for Linux** スタートアップガイド ..... **52**

**4. SIOS Protection Suite** インストレーションガイド ..... **53**
4.1. SIOS Protection Suite ソフトウェアのパッケージ ..... 54
4.2. SPS 環境のプランニング ..... 56
4.2.1. サーバ構成のマッピング ..... 57
4.2.2. ストレージとアダプタの要件 ..... 59
4.2.3. ストレージとアダプタのオプション ..... 60
4.3. SPS 環境のセットアップ ..... 61
4.3.1. Linux OS および関連する通信パッケージのインストール ..... 62
4.3.2. Linux の依存関係 ..... 63
4.3.3. サーバと共有ストレージの接続 ..... 67
4.3.4. 共有ストレージの設定 ..... 68
4.3.5. ネットワーク設定の確認 ..... 69
4.3.6. 切り替え可能な IP アドレスの作成 ..... 71
4.3.7. データベースアプリケーションのインストールとセットアップ ..... 72
4.3.8. GUI ユーザの設定 ..... 73
4.3.9. ライセンスの取得とインストール ..... 75
4.3.9.1. インターネットホスト ID の取得 ..... 79
4.4. SIOS Protection Suite ソフトウェアのインストール ..... 80
4.5. セットアップスクリプトの操作 ..... 83
4.6. SPS LifeKeeper インストールの確認 ..... 89
4.7. SPS のアップグレード ..... 90

**5. SIOS Protection Suite for Linux** テクニカルドキュメンテーション ..... **92**
5.1. はじめに ..... 93
5.2. ドキュメンテーションとトレーニング ..... 94
5.3. lkbackup ..... 96
5.4. LifeKeeper ..... 98
5.4.1. はじめに ..... 99
5.4.1.1. 保護対象のリソース ..... 101
5.4.1.2. LifeKeeper Core ..... 102
5.4.1.3. 設定の概念 ..... 105
5.4.1.3.1. 共通のハードウェアコンポーネント ..... 106
5.4.1.3.2. システムのグループ化の配置 ..... 108
5.4.1.3.3. アクティブ - アクティブのグループ化 ..... 109
5.4.1.3.4. アクティブ - スタンバイのグループ化 ..... 110
5.4.1.3.5. Intelligent スイッチバックと Automatic スイッチバックの違い ..... 112
5.4.1.3.6. syslog によるログの記録 ..... 113
5.4.1.3.7. リソース階層 ..... 114

5.4.1.3.7.1. リソースタイプ ..... 115
5.4.1.3.7.2. リソースの状態 ..... 116
5.4.1.3.7.3. 階層の関係 ..... 117
5.4.1.3.7.4. イクイバレンシ情報 ..... 118
5.4.1.3.7.5. リソース階層の情報 ..... 119
5.4.1.3.7.6. リソース階層の例 ..... 120
5.4.1.3.7.7. ステータスの詳細表示 ..... 121
5.4.1.3.7.8. ステータスの簡略表示 ..... 127
5.4.1.4. 障害検出とリカバリのシナリオ ..... 129
5.4.1.4.1. IP ローカルリカバリ ..... 130
5.4.1.4.2. リソースのエラーリカバリのシナリオ ..... 131
5.4.1.4.3. サーバの障害リカバリのシナリオ ..... 134
5.4.2. インストールと設定 ..... 136
5.4.2.1. SPS の設定手順 ..... 137
5.4.2.1.1. TTY 接続のセットアップ ..... 139
5.4.2.2. SNMP による LifeKeeper イベント転送 ..... 140
5.4.2.2.1. SNMP による LifeKeeper イベント転送の概要 ..... 141
5.4.2.2.2. LifeKeeper イベント転送の設定 ..... 145
5.4.2.2.3. SNMP のトラブルシューティング ..... 147
5.4.2.3. LifeKeeper イベントメール通知 ..... 148
5.4.2.3.1. LifeKeeper イベントメール通知の概要 ..... 149
5.4.2.3.2. LifeKeeper イベントメール通知の設定 ..... 151
5.4.2.3.3. メール通知のトラブルシューティング ..... 153
5.4.2.4. 任意の設定作業 ..... 154
5.4.2.4.1. [Confirm Failover] と [Block Resource Failover] の設定 ..... 155
5.4.2.4.2. サーバーのシャットダウンストラテジーの設定 ..... 163
5.4.2.4.3. LifeKeeper ハートビートの調整 ..... 164
5.4.2.4.4. SPS API でカスタム証明書を使用する ..... 166
5.4.2.5. Linux の設定 ..... 168
5.4.2.6. データレプリケーションの設定 ..... 172
5.4.2.7. ネットワーク設定 ..... 173
5.4.2.8. アプリケーションの設定 ..... 174
5.4.2.9. ストレージとアダプタの設定 ..... 175
5.4.2.10. LifeKeeper I-O フェンシングの概要 ..... 216
5.4.2.10.1. SCSI リザベーション ..... 217
5.4.2.10.2. リザベーションの無効化 ..... 219
5.4.2.10.2.1. I/O フェンシング表 ..... 221
5.4.2.10.3. Quorum/Witness ..... 223
5.4.2.10.3.1. majority モード ..... 228
5.4.2.10.3.2. tcp_remote モード ..... 233
5.4.2.10.3.3. storage モード ..... 234
5.4.2.10.4. STONITH ..... 242
5.4.2.10.5. Watchdog ..... 246
5.4.2.10.6. I/O フェンシング機能組み合わせ情報 ..... 249
5.4.2.10.6.1. フェンシング機能利用可否リスト（物理構成） ..... 251

5.4.2.10.6.2. フェンシング機能利用可否リスト（仮想構成 VMware編） ..... 255
5.4.2.11. リソースポリシー管理 ..... 260
5.4.2.12. 認証情報の設定 ..... 265
5.4.2.13. Standby Node Health Check ..... 267
5.4.2.13.1. Node監視 ..... 269
5.4.2.13.2. OSU リソース監視 ..... 270
5.4.3. LifeKeeper 管理 ..... 272
5.4.3.1. エラーの検出および通知 ..... 274
5.4.3.2. N-Way リカバリ ..... 275
5.4.3.3. 管理作業 ..... 276
5.4.3.3.1. サーバプロパティの編集 ..... 277
5.4.3.3.2. コミュニケーションパスの作成 ..... 278
5.4.3.3.3. コミュニケーションパスの削除 ..... 280
5.4.3.3.4. サーバのプロパティ - フェイルオーバ ..... 281
5.4.3.3.5. リソース階層の作成 ..... 283
5.4.3.3.5.1. ファイルシステムリソース階層の作成 ..... 285
5.4.3.3.5.2. Generic Application リソース階層の作成 ..... 287
5.4.3.3.5.3. Raw デバイスリソース階層の作成 ..... 289
5.4.3.3.5.4. Quick Service Protection (QSP) Recovery Kit ..... 291
5.4.3.3.6. リソースのプロパティの編集 ..... 297
5.4.3.3.7. リソースの優先順位の編集 ..... 298
5.4.3.3.8. リソース階層の拡張 ..... 300
5.4.3.3.8.1. ファイルシステムリソース階層の拡張 ..... 302
5.4.3.3.8.2. Generic Application リソース階層の拡張 ..... 303
5.4.3.3.8.3. Raw デバイスリソース階層の拡張 ..... 304
5.4.3.3.9. 階層の拡張解除 ..... 305
5.4.3.3.10. リソース依存関係の作成 ..... 306
5.4.3.3.11. リソース依存関係の削除 ..... 308
5.4.3.3.12. すべてのサーバからの階層の削除 ..... 309
5.4.4. LifeKeeper User Guide ..... 310
5.4.4.1. LifeKeeper for Linux の使用 ..... 311
5.4.4.1.1. GUI ..... 312
5.4.4.1.1.1. GUI の概要 – 全般 ..... 314
5.4.4.1.1.1.1. LifeKeeper GUI ソフトウェアパッケージ ..... 315
5.4.4.1.1.2. メニュー ..... 316
5.4.4.1.1.2.1. リソースのコンテキストメニュー ..... 317
5.4.4.1.1.2.2. サーバのコンテキストメニュー ..... 319
5.4.4.1.1.2.3. [File] メニュー ..... 320
5.4.4.1.1.2.4. [Edit] メニュー - [Resource] ..... 321
5.4.4.1.1.2.5. [Edit] メニュー - [Server] ..... 322
5.4.4.1.1.2.6. [View] メニュー ..... 323
5.4.4.1.1.2.7. [Help] メニュー ..... 325
5.4.4.1.1.3. ツールバー ..... 326
5.4.4.1.1.3.1. GUI のツールバー ..... 327
5.4.4.1.1.3.2. リソースのコンテキストツールバー ..... 329

5.4.4.1.1.3.3. サーバのコンテキストツールバー ..... 330
5.4.4.1.1.4. GUI の実行の準備 ..... 331
5.4.4.1.1.4.1. LifeKeeper の GUI の設定 ..... 332
5.4.4.1.1.4.2. GUI サーバの開始 / 停止 ..... 335
5.4.4.1.1.4.3. Java のセキュリティポリシー ..... 337
5.4.4.1.1.4.4. LifeKeeper サーバでの GUI の実行 ..... 340
5.4.4.1.1.4.5. LifeKeeper の GUI – 概要 ..... 341
5.4.4.1.2. ステータスの表 ..... 343
5.4.4.1.3. プロパティパネル ..... 344
5.4.4.1.4. 出力パネル ..... 345
5.4.4.1.5. メッセージバー ..... 346
5.4.4.1.6. GUI の終了 ..... 347
5.4.4.1.7. 共通の作業 ..... 348
5.4.4.1.7.1. LifeKeeper の起動 ..... 349
5.4.4.1.7.2. LifeKeeper の停止 ..... 350
5.4.4.1.7.3. LifeKeeper プロセスの表示 ..... 351
5.4.4.1.7.4. LifeKeeper GUI サーバプロセスの表示 ..... 353
5.4.4.1.7.5. LifeKeeper の制御プロセスの表示 ..... 354
5.4.4.1.7.6. サーバのクラスタへの接続 ..... 355
5.4.4.1.7.7. クラスタからの切断 ..... 356
5.4.4.1.7.8. 接続サーバの表示 ..... 357
5.4.4.1.7.9. サーバのステータスの表示 ..... 358
5.4.4.1.7.10. サーバのプロパティの表示 ..... 359
5.4.4.1.7.11. サーバのログファイルの表示 ..... 360
5.4.4.1.7.12. リソースのタグと ID の表示 ..... 361
5.4.4.1.7.13. リソースのステータスの表示 ..... 362
5.4.4.1.7.14. リソースのプロパティの表示 ..... 364
5.4.4.1.7.15. Resource Labels ..... 365
5.4.4.1.7.16. メッセージ履歴の表示 ..... 366
5.4.4.1.7.17. リソース階層ツリーの展開と折り畳み ..... 367
5.4.4.1.7.18. [Cluster Connect] ダイアログ ..... 369
5.4.4.1.7.19. [Cluster Disconnect] ダイアログ ..... 370
5.4.4.1.7.20. [Resource Properties] ダイアログ ..... 371
5.4.4.1.7.21. [Server Properties] ダイアログ ..... 373
5.4.4.1.8. オペレータの作業 ..... 377
5.4.4.1.8.1. リソースを In Service にする ..... 378
5.4.4.1.8.2. リソースを Out of Service にする ..... 379
5.4.4.1.9. 高度な作業 ..... 380
5.4.4.1.9.1. LCD ..... 381
5.4.4.1.9.1.1. LCDI のコマンド ..... 382
5.4.4.1.9.1.2. LCD の設定データ ..... 386
5.4.4.1.9.1.3. LCD のディレクトリ構造 ..... 387
5.4.4.1.9.1.4. LCD のリソースタイプ ..... 388
5.4.4.1.9.1.5. LifeKeeper のフラグ ..... 389
5.4.4.1.9.1.6. リソースのサブディレクトリ ..... 390

5.4.4.1.9.1.7. /opt/LifeKeeper の LCD のディレクトリ構造 ..... 392
5.4.4.1.9.2. LCM ..... 393
5.4.4.1.9.2.1. 通信ステータスの情報 ..... 394
5.4.4.1.9.2.2. LifeKeeper の警報とリカバリ ..... 395
5.4.4.1.9.3. LifeKeeper API for Monitoring ..... 397
5.4.4.1.10. メンテナンス作業 ..... 407
5.4.4.1.10.1. LifeKeeper の設定値の変更 ..... 408
5.4.4.1.10.2. ファイルシステムの健全性の監視 ..... 411
5.4.4.1.10.3. LifeKeeper が保護するシステムのメンテナンス ..... 413
5.4.4.1.10.4. リソース階層のメンテナンス ..... 414
5.4.4.1.10.5. フェイルオーバ後の復旧 ..... 415
5.4.4.1.10.6. LifeKeeper の削除 ..... 416
5.4.4.1.10.7. ファイアウォールを使用した状態での LifeKeeper の実行 ..... 417
5.4.4.1.10.8. ファイアウォール経由での LifeKeeper GUI の実行 ..... 419
5.4.4.1.10.9. リソース階層の転送 ..... 421
5.4.4.1.11. テクニカルノート ..... 422
5.4.4.2. クラスタの例 ..... 427
5.4.4.3. ダイアログ ..... 428
5.4.5. トラブルシューティング ..... 438
5.4.5.1. ソリューション ..... 440
5.4.5.2. SPS が開始するフェイルオーバの一般的な原因 ..... 443
5.4.5.3. 既知の問題と制限 ..... 448
5.4.5.3.1. インストール - 既知の問題と制限 ..... 449
5.4.5.3.2. LifeKeeper Core – 既知の問題と制限 ..... 452
5.4.5.3.3. インターネット / IP ライセンス - 既知の問題と制限 ..... 460
5.4.5.3.4. GUI – 既知の問題と制限 ..... 461
5.4.5.3.5. データレプリケーション – 既知の問題と制限 ..... 464
5.4.5.3.6. IPv6 - 既知の問題と制限 ..... 467
5.4.5.3.7. Apache - 既知の問題と制限 ..... 469
5.4.5.3.8. Oracle Recovery Kit - 既知の問題と制限 ..... 470
5.4.5.3.9. MySQL - 既知の問題と制限 ..... 471
5.4.5.3.10. NAS Recovery Kit – 既知の問題と制限 ..... 472
5.4.5.3.11. NFS Server Recovery Kit – 既知の問題と制限 ..... 473
5.4.5.3.12. SAP Recovery Kit – 既知の問題と制限 ..... 476
5.4.5.3.13. LVM Recovery Kit – 既知の問題と制限 ..... 478
5.4.5.3.14. Multipath Recovery Kits (DMMP / HDLM / PPATH /NECSPS) - 既知の問題と制限 ..... 479
5.4.5.3.15. DMMP Recovery Kit – 既知の問題と制限 ..... 480
5.4.5.3.16. DB2 Recovery Kit – 既知の問題と制限 ..... 481
5.4.5.3.17. Sybase ASE Recovery Kit – 既知の問題と制限 ..... 482
5.4.5.3.18. WebSphere MQ Recovery Kit – 既知の問題と制限 ..... 485
5.4.5.3.19. SAP HANA – 既知の問題と制限 ..... 486
5.4.5.4. GUI トラブルシューティング ..... 487
5.4.5.4.1. ネットワーク関連トラブルシューティング (GUI) ..... 488
5.4.5.5. コミュニケーションパスの稼働と停止 ..... 493

5.4.5.6. 不完全なリソースの作成 ..... 494
5.4.5.7. 不完全なリソースの優先順位の変更 ..... 495
5.4.5.8. 階層の設定中に共有ストレージが見つからない ..... 497
5.4.5.9. LifeKeeper サーバ障害からの復旧 ..... 499
5.4.5.10. 停止できないプロセスからの復旧 ..... 500
5.4.5.11. 手動リカバリ時のパニックからの復旧 ..... 501
5.4.5.12. Out-of-Service 階層の復旧 ..... 502
5.4.5.13. リソースタグ名の制限 ..... 503
5.4.5.14. シリアル (TTY) コンソールの警告 ..... 504
5.4.5.15. システムが init 状態 S に遷移しているという警告 ..... 505
5.4.5.16. 共有ストレージでスレッドがハングしているというメッセージ ..... 506
5.5. SIOS DataKeeper for Linux ..... 507
5.5.1. SIOS DataKeeper for Linux によるミラーリング ..... 508
5.5.2. SIOS DataKeeper の仕組み ..... 510
5.5.3. インストールと設定 ..... 521
5.5.3.1. ハードウェアとソフトウェアの要件 ..... 522
5.5.3.2. 全般的な設定 ..... 524
5.5.3.3. ネットワーク設定 ..... 525
5.5.3.4. データレプリケーションパスの変更 ..... 526
5.5.3.5. ネットワーク帯域幅の要件の特定 ..... 527
5.5.3.5.1. Linux システム（物理または仮想）での変化率の測定 ..... 528
5.5.3.6. WAN の設定 ..... 538
5.5.3.7. SIOS DataKeeper for Linux のリソースタイプ ..... 539
5.5.3.8. DataKeeper 構成での I/O フェンシング ..... 541
5.5.3.9. リソースの設定作業 ..... 542
5.5.3.9.1. DataKeeper リソース階層の作成 ..... 543
5.5.3.9.1.1. Replicate New File System ..... 546
5.5.3.9.1.2. Replicate Existing File System ..... 549
5.5.3.9.1.3. DataKeeper Resource ..... 552
5.5.3.9.2. リソース階層の拡張 ..... 555
5.5.3.9.3. リソース階層の拡張解除 ..... 559
5.5.3.9.4. リソース階層の削除 ..... 560
5.5.3.9.5. DataKeeper リソースを Out of Service にする ..... 561
5.5.3.9.6. DataKeeper リソースを In Service にする ..... 562
5.5.3.9.7. リソース階層のテスト ..... 563
5.5.4. 管理 ..... 564
5.5.4.1. ミラーのステータスの表示 ..... 565
5.5.4.2. GUI からのミラーの管理 ..... 567
5.5.4.2.1. 一時停止と再開 ..... 569
5.5.4.2.2. 圧縮レベルの設定 ..... 570
5.5.4.3. コマンドラインからのミラー管理 ..... 571
5.5.4.4. コマンドラインからのミラーステータスの監視 ..... 575
5.5.4.5. サーバの障害 ..... 577
5.5.4.6. 再同期 ..... 578
5.5.4.7. 全同期の回避 ..... 579

5.5.4.8. 再同期前のデータ検証 (Wait to Resync) ..... 584
5.5.5. DataKeeper で LVM を使用する ..... 591
5.5.6. Fusion-io を使用するクラスタ化 ..... 592
5.5.7. DataKeeper for Linux が保護している領域に対する、スナップショット機能の利用についての注意事項 ..... 595
5.5.8. トラブルシューティング ..... 596
5.6. コマンドラインインターフェース ..... 603
5.6.1. コマンド ..... 605
5.6.1.1. lklogmsg ..... 608
5.6.1.2. SYS – LifeKeeper クラスタ内のシステムに関する LifeKeeper のコマンド ..... 610
5.6.1.3. NET – コマンドに関するコミュニケーションパス ..... 612
5.6.1.4. FLAG – LifeKeeper の内部フラグに関するコマンド ..... 614
5.6.1.5. TYP – リソース階層タイプに関する LifeKeeper のコマンド ..... 615
5.6.1.6. APP – リソースアプリケーションに関する LifeKeeper のコマンド (関連タイプのグループ) ..... 616
5.6.1.7. DEP – リソースアプリケーションを相互に関連付ける方法に関する LifeKeeper のコマンド ..... 617
5.6.1.8. INS – 個々の LifeKeeper 階層インスタンスに関するコマンド ..... 624
5.6.1.8.1. 階層の拡張解除 ..... 626
5.6.1.9. マニュアルページへのアクセス ..... 627
5.6.2. LKCLI（LifeKeeper コマンドラインインターフェース） ..... 629
5.6.2.1. ARK 毎のサブコマンド ..... 644
5.6.3. Ansible による LifeKeeper のセットアップ ..... 665
5.6.4. LKCLI ガイド ..... 674
5.6.4.1. コミュニケーションパスの作成･削除 ..... 675
5.6.4.2. リソースの作成 ..... 681
5.6.4.2.1. ファイルシステムリソースの作成 ..... 682
5.6.4.2.2. IP リソースの作成 ..... 685
5.6.4.2.3. PostgreSQL リソースの作成 ..... 688
5.6.4.2.4. DataKeeper リソースの作成 ..... 691
5.6.4.3. ステータスを確認するには？ ..... 695
5.6.4.4. スイッチオーバーの動作確認 ..... 696
5.6.4.5. メンテナンス ..... 698
5.6.4.6. クラスターの設定を引き継いで複製する ..... 703

**6. アプリケーションリカバリーキット ..... 707**
6.1. Apache Recovery Kit 管理ガイド ..... 708
6.1.1. ドキュメンテーションとリファレンス ..... 709
6.1.2. 要件 ..... 710
6.1.3. LifeKeeper における Apache Web Server の設定 ..... 711
6.1.3.1. 設定の定義と例 ..... 712
6.1.3.1.1. アクティブ / スタンバイ設定とアクティブ / アクティブ設定 ..... 717
6.1.3.2. Apache Web Server の設定に関する考慮事項 ..... 718
6.1.4. LifeKeeper の設定作業 ..... 722
6.1.4.1. Apache Web Server リソース階層の作成 ..... 723
6.1.4.2. リソース階層の拡張 ..... 725

6.1.4.3. リソース階層の拡張解除 ... 728
6.1.4.4. リソース階層の削除 ... 729
6.1.4.5. リソース階層のテスト ... 730
6.1.5. トラブルシューティング ... 731
6.1.5.1. 階層作成時のエラー ... 732
6.1.5.2. 階層拡張時のエラー ... 736
6.1.5.3. 階層リソースのリストア、削除、リカバリ時のメッセージとエラー ... 738
6.2. DB2 Recovery Kit 管理ガイド ... 741
6.2.1. ドキュメンテーションとリファレンス ... 742
6.2.2. ハードウェア要件とソフトウェア要件 ... 743
6.2.3. 概要 ... 744
6.2.4. LifeKeeper for Linux DB2 Recovery Kit の設定 ... 745
6.2.4.1. DB2 で RAW I/O を使用する場合 ... 746
6.2.4.2. DB2 の使用 ... 747
6.2.4.3. DB2 シングル・パーティションの設定の検討事項 ... 748
6.2.4.4. DB2 マルチ・パーティションの設定の検討事項 ... 749
6.2.4.4.1. DB2 EEE またはマルチ・パーティション ESE と NFS に関する問題 ... 750
6.2.4.4.2. 設定条件 ... 752
6.2.4.5. すべての DB2 構成に共通する設定の検討事項 ... 755
6.2.4.6. 構成例 ... 756
6.2.5. LifeKeeper の設定作業 ... 761
6.2.5.1. リソース階層の作成 ... 762
6.2.5.2. リソース階層の削除 ... 765
6.2.5.3. 階層の拡張 ... 767
6.2.5.4. 階層の拡張解除 ... 770
6.2.5.5. リソース階層のテスト ... 771
6.2.6. トラブルシューティング ... 772
6.2.7. 付録： DB2 で RAW I/O を使用する設定 ... 773
6.3. Recovery Kit for EC2 管理ガイド ... 776
6.3.1. 運用の原則 ... 777
6.3.2. 要件 ... 781
6.3.3. 設定 ... 783
6.3.3.1. Recovery Kit for EC2 のチューニング ... 785
6.3.3.2. リソース階層の作成 ... 786
6.3.3.3. リソース階層の削除 ... 788
6.3.3.4. リソース階層の拡張 ... 789
6.3.3.5. ローカルリカバリと設定に関する考慮事項 ... 791
6.3.3.6. リソース監視と設定に関する考慮事項 ... 793
6.3.3.7. リソース階層の拡張解除 ... 794
6.3.3.8. ユーザシステムのセットアップ ... 795
6.3.4. トラブルシューティング ... 797
6.4. Generic ARK for Load Balancer probe reply ... 798
6.4.1. 環境構成例 ... 799
6.4.2. 基本動作 ... 802
6.4.3. スクリプト仕様 ... 803

6.4.4. スクリプトのパラメーター一覧 ..... 805
6.4.5. リソースの作成/拡張 ..... 806
6.4.6. メッセージ一覧 ..... 807
6.5. LVM Recovery Kit 管理ガイド ..... 808
6.5.1. ドキュメンテーションとリファレンス ..... 809
6.5.2. 要件 ..... 810
6.5.2.1. ハードウェア要件とソフトウェア要件 ..... 811
6.5.3. 概要 ..... 812
6.5.3.1. LVM Recovery Kit の注意事項と制約事項 ..... 814
6.5.4. SPS LVM 階層の作成と管理 ..... 816
6.5.4.1. 階層作成の手順 ..... 817
6.5.4.2. DataKeeper で LVM を使用する ..... 818
6.5.4.3. ボリュームグループの再設定 ..... 819
6.5.5. トラブルシューティング ..... 823
6.6. IP Recovery Kit 管理ガイド ..... 825
6.6.1. 運用の原則 ..... 826
6.6.2. 要件 ..... 829
6.6.3. 設定 ..... 830
6.6.3.1. IP Recovery Kit のチューニング ..... 832
6.6.3.2. 設定例 ..... 833
6.6.3.3. IP リソース階層の作成 ..... 840
6.6.3.4. リソース階層の削除 ..... 842
6.6.3.5. リソース階層の拡張 ..... 843
6.6.3.6. 一般的な IP プランニングの考慮事項 ..... 846
6.6.3.7. IP 依存関係の作成時のガイドライン ..... 847
6.6.3.8. インターフェースの選択 ..... 848
6.6.3.9. IP ローカルリカバリと設定に関する考慮事項 ..... 849
6.6.3.10. IP のリソース監視と設定に関する考慮事項 ..... 850
6.6.3.11. リソース階層のテスト ..... 851
6.6.3.12. リソース階層の拡張解除 ..... 852
6.6.3.13. ユーザシステムのセットアップ ..... 853
6.6.3.14. IP 構成の確認および編集 ..... 854
6.7. MySQL Recovery Kit 管理ガイド ..... 866
6.7.1. 要件 ..... 867
6.7.2. 設定上の考慮事項 ..... 868
6.7.2.1. MySQL の設定上の考慮事項 ..... 869
6.7.2.2. クライアント設定の考慮事項 ..... 872
6.7.2.3. 設定の要件 ..... 873
6.7.2.4. 設定例 ..... 874
6.7.2.5. アクティブ / スタンバイ設定 ..... 875
6.7.2.6. アクティブ / アクティブ設定 ..... 877
6.7.2.7. 複数データベースサーバ環境に関する考慮事項 ..... 882
6.7.2.8. LifeKeeper で mysqld グループを使用する ..... 883
6.7.2.9. Network Attached Storage の使用 ..... 888
6.7.2.10. systemd 環境で使用する場合の考慮事項 ..... 892

6.7.3. インストール ..... 893
6.7.3.1. LifeKeeper の設定作業 ..... 894
6.7.3.2. MySQL リソース階層の作成 ..... 895
6.7.3.3. リソース階層の削除 ..... 899
6.7.3.4. リソース階層の拡張 ..... 901
6.7.3.5. リソース階層の拡張解除 ..... 905
6.7.4. 管理 ..... 907
6.7.4.1. GUI による手動スイッチオーバの実行 ..... 908
6.7.5. トラブルシューティング ..... 909
6.8. WebSphere MQ Recovery Kit 管理ガイド ..... 912
6.8.1. 略語 ..... 914
6.8.2. 要件 ..... 915
6.8.2.1. ハードウェア要件とソフトウェア要件 ..... 916
6.8.2.2. LifeKeeper MQ クラスタのアップグレード ..... 918
6.8.3. WebSphere MQ Recovery Kit の概要 ..... 919
6.8.3.1. WebSphere MQ リソース階層 ..... 920
6.8.3.2. Recovery Kit の機能 ..... 921
6.8.4. WebSphere MQ を設定する際の考慮事項 ..... 922
6.8.4.1. 設定の要件 ..... 923
6.8.4.1.1. サポートされるファイルシステムレイアウト ..... 926
6.8.4.1.1.1. 設定 1 – 共有ストレージ上の /var/mqm ..... 927
6.8.4.1.1.2. 設定 2 – 直接マウント ..... 928
6.8.4.1.1.3. 設定 3 – シンボリックリンク ..... 929
6.8.4.1.2. LifeKeeper 使用時の WebSphere MQ の設定 ..... 930
6.8.4.1.3. リソース作成後の設定変更 ..... 935
6.8.4.1.3.1. QMDIR と QMLOGDIR の再配置 ..... 936
6.8.4.1.3.2. リスナーポートの変更 ..... 937
6.8.4.1.3.3. キューマネージャの IP の変更 ..... 938
6.8.4.1.4. WebSphere MQ の設定例 ..... 939
6.8.4.1.4.1. 共有ストレージ上の /var/mqm によるアクティブ / スタンバイ設定 ..... 940
6.8.4.1.4.2. NAS ストレージにおけるアクティブ / スタンバイ設定 ..... 941
6.8.4.1.4.3. 共有ストレージにおけるアクティブ / アクティブ設定 ..... 943
6.8.4.1.4.4. NAS ストレージにおけるアクティブ / アクティブ設定 ..... 945
6.8.5. LifeKeeper の設定作業 ..... 947
6.8.5.1. WebSphere MQ リソース階層の作成 ..... 949
6.8.5.2. WebSphere MQ 階層の拡張 ..... 951
6.8.5.3. WebSphere MQ 階層の拡張解除 ..... 953
6.8.5.4. WebSphere MQ リソース階層の削除 ..... 954
6.8.5.5. WebSphere MQ リソース階層のテスト ..... 955
6.8.5.5.1. クライアント接続のテスト ..... 957
6.8.5.6. リソースのプロパティの表示 ..... 959
6.8.5.7. リソースプロパティ設定の編集 ..... 960
6.8.5.7.1. リスナー保護の有効化 / 無効化 ..... 964
6.8.5.7.2. LifeKeeper テストキュー名の変更 ..... 966
6.8.5.7.3. ログレベルの変更 ..... 967

6.8.5.7.4. シャットダウンタイムアウト値の変更 ..... 968
6.8.5.7.5. サーバ接続チャネルの変更 ..... 970
6.8.5.7.6. コマンドサーバ保護設定の変更 ..... 971
6.8.5.7.7. LifeKeeper WebSphere MQ Recovery Kit のデフォルト設定の変更 ..... 973
6.8.6. WebSphere MQ のトラブルシューティング ..... 975
6.8.6.1. エラーメッセージ ..... 976
6.8.7. 付録 A – mqs.ini 設定ファイルの例 ..... 984
6.8.8. 付録 B – qm.ini 設定ファイルの例 ..... 986
6.8.9. 付録 C – WebSphere MQ 設定シート ..... 987
6.9. NAS Recovery Kit 管理ガイド ..... 990
6.9.1. ドキュメンテーションとリファレンス ..... 991
6.9.2. ハードウェア要件とソフトウェア要件 ..... 992
6.9.3. 概要 ..... 993
6.9.4. LifeKeeper for Linux NAS Recovery Kit の設定 ..... 995
6.9.4.1. 設定上の考慮事項 ..... 996
6.9.4.2. 構成例 ..... 998
6.9.5. LifeKeeper の設定作業 ..... 1000
6.9.5.1. リソース階層の作成 ..... 1001
6.9.5.2. リソース階層の削除 ..... 1004
6.9.5.3. 階層の拡張 ..... 1006
6.9.5.4. 階層の拡張解除 ..... 1009
6.9.5.5. リソース階層のテスト ..... 1010
6.9.6. トラブルシューティング ..... 1011
6.9.6.1. エラーメッセージ ..... 1012
6.9.6.2. LifeKeeper GUI 関連のエラー ..... 1017
6.10. NFS Recovery Kit 管理ガイド ..... 1018
6.10.1. 概要 ..... 1019
6.10.2. 要件 ..... 1020
6.10.3. 設定上の考慮事項 ..... 1021
6.10.3.1. 特定の設定上の考慮事項 ..... 1022
6.10.3.2. LifeKeeper における NFS サーバーの設定 ..... 1024
6.10.3.3. 設定例 ..... 1028
6.10.3.3.1. NFS v2/v3 でのアクティブ / アクティブ ..... 1029
6.10.3.3.2. NFS v2/v3 でのアクティブ / スタンバイ ..... 1031
6.10.4. 設定作業 ..... 1033
6.10.4.1. NFS リソース階層の作成 ..... 1034
6.10.4.2. リソース階層の削除 ..... 1037
6.10.4.3. リソース階層の拡張 ..... 1039
6.10.4.4. リソース階層のテスト ..... 1043
6.10.4.5. リソース階層の拡張解除 ..... 1045
6.10.5. トラブルシューティング ..... 1047
6.10.5.1. HA nfs-utils のインストールと設定 ..... 1048
6.10.5.2. 階層削除のメッセージとエラー ..... 1049
6.10.5.3. 階層のリストア、削除、リカバリ時のメッセージ ..... 1050
6.10.5.4. NFS 階層拡張時のエラー ..... 1053

6.10.5.5. NFS 階層作成時のエラー ..... 1054
6.11. Oracle Recovery Kit 管理ガイド ..... 1057
6.11.1. 要件 ..... 1058
6.11.2. LifeKeeper での Oracle の設定 ..... 1059
6.11.2.1. Oracle 特有の設定上の考慮事項 ..... 1060
6.11.2.2. LifeKeeper 保護用の Oracle Net Listener の設定 ..... 1065
6.11.2.3. LifeKeeper での透過的アプリケーションフェイルオーバの設定 ..... 1069
6.11.2.4. Oracle Multitenant 構成による Pluggable Database の設定 ..... 1071
6.11.2.5. 設定例 ..... 1076
6.11.2.5.1. 設定の要件 ..... 1077
6.11.2.5.2. アクティブ / スタンバイ設定 ..... 1078
6.11.2.5.3. アクティブ / アクティブ設定 ..... 1081
6.11.3. LifeKeeper の設定作業 ..... 1086
6.11.3.1. リソース階層の作成 ..... 1087
6.11.3.2. リソース階層の削除 ..... 1089
6.11.3.3. リソース階層の拡張 ..... 1091
6.11.3.4. リソース階層の拡張解除 ..... 1095
6.11.3.5. Oracle 設定の参照 ..... 1097
6.11.3.6. Oracle データベースアカウントに対するユーザ名 / パスワードの変更 ..... 1098
6.11.3.7. リソース階層のテスト ..... 1100
6.11.4. トラブルシューティング ..... 1101
6.11.4.1. Oracle 既知の問題と制限 ..... 1102
6.11.4.1.1. データベース作成時の問題 ..... 1107
6.11.4.1.2. データベース起動時の問題 ..... 1108
6.11.4.1.3. LifeKeeper の inqfail エラーログ ..... 1109
6.11.5. 付録 ..... 1110
6.11.5.1. Raw I/O を使用するための Oracle のセットアップ ..... 1111
6.11.5.1.1. 階層作成後のテーブルスペースの追加 ..... 1114
6.11.5.2. 複数のリソースに対する共有 Oracle Listener の作成 ..... 1115
6.11.5.2.1. Listener 保護レベルのアップデート ..... 1119
6.11.5.2.2. Listener リカバリレベルのアップデート ..... 1120
6.11.5.2.3. 保護された Listener のアップデート ..... 1121
6.11.5.3. Pluggable Database の移行 ..... 1122
6.12. PostgreSQL Recovery Kit 管理ガイド ..... 1125
6.12.1. PostgreSQL リソース階層 ..... 1126
6.12.2. 要件 ..... 1127
6.12.3. 設定上の考慮事項 ..... 1128
6.12.3.1. Postgre SQL の保護: ベストプラクティス ..... 1129
6.12.3.2. DataKeeper でのミラーファイルシステムの使用 ..... 1130
6.12.4. インストール ..... 1131
6.12.4.1. PostgreSQL ソフトウェアのインストール ..... 1133
6.12.4.2. PostgreSQL データベースの作成 ..... 1135
6.12.4.3. LifeKeeper ソフトウェアのインストール ..... 1137
6.12.4.4. PostgreSQL リソースのための LifeKeeper 設定項目 ..... 1138
6.12.4.5. PostgreSQL リソース階層の作成 ..... 1140

6.12.4.6. PostgreSQL リソース階層の削除 ..... 1142
6.12.4.7. PostgreSQL リソース階層の拡張 ..... 1143
6.12.4.8. PostgreSQL リソース階層の拡張解除 ..... 1145
6.12.4.9. PostgreSQL 設定の参照 ..... 1146
6.12.4.10. PostgreSQL Recovery Kit の旧バージョンからのアップグレード ..... 1147
6.12.5. 管理 ..... 1150
6.12.5.1. LifeKeeper GUI からの手動スイッチオーバの実行 ..... 1151
6.12.5.2. EnterpriseDB Postgres Plus Advanced Server の保護 ..... 1152
6.12.5.3. Symfoware Server/Enterprise Postgres の保護 ..... 1153
6.12.5.4. データベース管理者ユーザの更新 ..... 1154
6.12.6. トラブルシューティング ..... 1155
6.12.6.1. 全般項目 ..... 1156
6.12.6.2. 設定項目 ..... 1158
6.13. Postfix Recovery Kit 管理ガイド ..... 1160
6.13.1. ハードウェア要件とソフトウェア要件 ..... 1162
6.13.1.1. Recovery Kit のインストール ..... 1163
6.13.2. LifeKeeper for Linux Postfix Recovery Kit の設定 ..... 1164
6.13.2.1. 保護対象 ..... 1165
6.13.2.2. 設定要件 ..... 1166
6.13.2.3. ポート定義とPostfix Recovery Kit ..... 1167
6.13.2.4. DNS、Postfix とLifeKeeper ..... 1168
6.13.2.5. 構成例 ..... 1169
6.13.3. Postfix の構成確認 ..... 1173
6.13.3.1. LifeKeeper の設定作業 ..... 1175
6.13.3.1.1. リソース階層の作成 ..... 1176
6.13.3.1.2. リソース階層の拡張 ..... 1178
6.13.3.1.3. リソース階層の拡張解除 ..... 1180
6.13.3.1.4. リソース階層の削除 ..... 1181
6.13.3.1.5. スプール (メールボックス) の依存関係作成 ..... 1182
6.13.3.1.6. リソース階層のテスト ..... 1183
6.13.4. トラブルシューティング ..... 1185
6.13.4.1. リソース階層作成時のエラーメッセージ ..... 1186
6.13.4.2. リソース階層拡張時のエラーメッセージ ..... 1187
6.13.4.3. サービスを起動、停止、回復時に出力されるエラーメッセージ ..... 1188
6.14. Recovery Kit for Route 53 管理ガイド ..... 1189
6.14.1. 要件 ..... 1190
6.14.2. 設定 ..... 1191
6.14.2.1. リソース階層の作成 ..... 1193
6.14.2.2. リソース階層の削除 ..... 1195
6.14.2.3. リソース階層の拡張 ..... 1196
6.14.2.4. リソース階層の拡張解除 ..... 1198
6.14.2.5. Recovery Kit for Route 53 のチューニング ..... 1199
6.14.2.6. リソースの監視とリカバリ動作 ..... 1200
6.14.2.7. ユーザシステムのセットアップ ..... 1201
6.14.3. トラブルシューティング ..... 1202

6.15. Samba Recovery Kit 管理ガイド ..... 1203
6.15.1. 要件 ..... 1205
6.15.2. Recovery Kit のインストール ..... 1206
6.15.3. Samba Recovery Kit の概要 ..... 1207
6.15.4. LifeKeeper for Linux Samba Recovery Kit の設定 ..... 1208
6.15.4.1. Samba 設定ファイル ..... 1209
6.15.4.2. 設定ファイルの [global] セクション ..... 1210
6.15.4.3. 設定ファイルの [homes] セクション ..... 1212
6.15.4.4. 設定ファイルの [printers] セクション ..... 1213
6.15.4.5. 設定ファイルの共用定義セクション ..... 1214
6.15.4.6. Samba の複数インスタンスの実行 ..... 1215
6.15.4.7. 構成例 ..... 1217
6.15.5. Samba の設定手順 ..... 1222
6.15.6. LifeKeeper の設定作業 ..... 1224
6.15.6.1. リソース階層の作成 ..... 1225
6.15.6.2. リソース階層の拡張 ..... 1227
6.15.6.3. リソース階層の拡張解除 ..... 1229
6.15.6.4. リソース階層の削除 ..... 1230
6.15.6.5. リソース階層のテスト ..... 1231
6.15.7. Samba 階層の管理 ..... 1232
6.15.7.1. Samba 設定ファイルの修正 ..... 1233
6.15.7.2. smbpasswd ファイルの管理 ..... 1236
6.15.8. トラブルシューティング ..... 1237
6.15.8.1. エラーメッセージ ..... 1238
6.15.8.2. リソース階層作成時のエラーメッセージ ..... 1239
6.15.8.3. リソース拡張時のエラーメッセージ ..... 1243
6.15.8.4. リソース復元時のエラーメッセージ ..... 1244
6.15.8.5. リソース削除時のエラーメッセージ ..... 1245
6.15.8.6. リソース監視時のエラーメッセージ ..... 1246
6.15.8.7. 設定ファイル同期ユーティリティのエラーメッセージ ..... 1247
6.16. SAP Recovery Kit 管理ガイド ..... 1248
6.16.1. 略語と定義 ..... 1250
6.16.2. LifeKeeper – SAP アイコン ..... 1252
6.16.3. SAP Recovery Kit の概要 ..... 1253
6.16.4. SIOS Protection Suite for SAP ソリューションページ ..... 1256
6.16.5. ハードウェア / ソフトウェア要件 ..... 1258
6.16.6. 設定上の考慮事項 ..... 1260
6.16.6.1. ABAP+Java の設定 (ASCS および SCS) ..... 1262
6.16.6.2. ABAP SCS (ASCS) ..... 1265
6.16.6.3. Java のみの構成 (SCS) ..... 1267
6.16.6.4. ディレクトリ構造 ..... 1269
6.16.6.5. 仮想サーバ名 ..... 1276
6.16.6.6. SAP ヘルスモニタリング ..... 1277
6.16.6.7. SAP ライセンス ..... 1279
6.16.6.8. 自動スイッチバック ..... 1280

6.16.6.9. その他の注記 ..... 1281
6.16.7. インストール ..... 1282
6.16.7.1. SAP構成を計画する ..... 1285
6.16.7.2. コアサービスのインストール ..... 1287
6.16.7.3. データベースのインストール ..... 1288
6.16.7.4. プライマリアプリケーションサーバインスタンスのインストール ..... 1289
6.16.7.5. 追加のアプリケーションサーバインスタンスのインストール ..... 1290
6.16.7.6. バックアップサーバへのインストール ..... 1291
6.16.7.7. SPS のインストール ..... 1292
6.16.7.8. ファイルシステムとディレクトリ構造の作成 ..... 1294
6.16.7.9. 共有ディスクおよび LifeKeeper へのデータ移動 ..... 1296
6.16.7.10. ASCSおよびERSインスタンスプロファイル設定の変更 ..... 1303
6.16.7.11. SAP Recovery Kit の旧バージョンからのアップグレード ..... 1306
6.16.7.12. IP リソース ..... 1308
6.16.7.13. SAP リソース階層の作成 ..... 1309
6.16.7.14. リソース階層の削除 ..... 1320
6.16.7.15. 共通の Recovery Kit 作業 ..... 1321
6.16.7.16. コマンドラインによる SAP の設定 ..... 1322
6.16.7.17. SAP SIOS HA Cluster Connector（SHCC）のアクティブ化 ..... 1324
6.16.7.18. テストの準備 ..... 1326
6.16.7.19. テストの実行 ..... 1327
6.16.8. 管理 ..... 1329
6.16.8.1. NFS の考慮事項 ..... 1330
6.16.8.2. クライアントの再接続 ..... 1332
6.16.8.3. SAP Recovery Kit のチューニング ..... 1333
6.16.8.4. SAP リソース階層と NFS リソース階層の分離 ..... 1335
6.16.8.5. 保護レベルの更新 ..... 1336
6.16.8.6. リカバリレベルの更新 ..... 1338
6.16.8.7. プロパティの表示 ..... 1339
6.16.8.8. Oracle に関する特別な考慮事項 ..... 1340
6.16.8.9. SSHCC HA アクション ..... 1341
6.16.8.10. LifeKeeperのERSリソースタイプ ..... 1342
6.16.8.11. ENSAv1からENSAv2へのアップグレード ..... 1348
6.16.8.12. ERSv1からERSv2へのアップグレード ..... 1350
6.16.9. トラブルシューティング ..... 1354
6.16.9.1. ERSプロファイルでAutostartを無効にする ..... 1355
6.16.9.2. ASCS + ERS Restart_Program パラメーター ..... 1356
6.16.9.3. 階層削除時のエラー ..... 1357
6.16.9.4. SAP 階層のリストアエラー ..... 1358
6.16.9.5. フェイルオーバ中または In Service の SAP エラーメッセージ ..... 1360
6.16.9.6. SAP インストールエラー ..... 1361
6.16.9.7. sapinit のトラブルシューティング ..... 1362
6.16.9.8. LifeKeeper ログファイルに出現する「tset」エラー ..... 1363
6.16.9.9. SAP SIOS HA Cluster Connector Errors ..... 1365
6.16.10. LifeKeeper が保護するシステムのメンテナンス ..... 1367

6.16.10.1. SAP メンテナンスモード ..... 1368
6.16.10.2. リソースポリシー管理 ..... 1372
6.17. SAP HANA Recovery Kit管理ガイド ..... 1377
6.17.1. SAP HANA Gen/AppからSAP HANA Recovery Kitへのアップグレード ..... 1379
6.17.2. SAP HANA Recovery Kitハードウェアおよびソフトウェアの要件 ..... 1382
6.17.3. SAP HANA Recovery Kit概要 ..... 1384
6.17.3.1. SAP HANA GUI ステータス ..... 1386
6.17.3.2. SAP HANAリソース階層 ..... 1389
6.17.3.3. マルチテナントデータベースのサポート ..... 1390
6.17.4. SPSでのSAP HANAの設定 ..... 1391
6.17.4.1. SAP HANAソフトウェアのインストール ..... 1392
6.17.4.2. SAP HANAシステムレプリケーションの構成 ..... 1393
6.17.4.3. SAP HANAインスタンスプロファイルの変更 ..... 1394
6.17.4.4. SPSソフトウェアのインストール ..... 1395
6.17.5. SAP HANAリソースの設定タスク ..... 1396
6.17.5.1. SAP HANAリソース階層の作成 ..... 1397
6.17.5.2. SAP HANAリソース階層の拡張 ..... 1400
6.17.5.3. SAP HANAリソース階層の拡張解除 ..... 1403
6.17.5.4. SAP HANAリソース階層の削除 ..... 1405
6.17.5.5. SAP HANAリソース階層のテスト ..... 1407
6.17.6. SAP HANAリソース階層の管理 ..... 1411
6.17.6.1. レプリケーションモードおよびオペレーションモードの変更 ..... 1415
6.17.6.2. スプリットブレインシナリオの解決 ..... 1418
6.17.6.3. Takeover with Handshake ..... 1421
6.17.7. SAP HANAトラブルシューティング ..... 1426
6.18. SAP MaxDB Recovery Kit 管理ガイド ..... 1427
6.18.1. SAP MaxDB Recovery Kit の要件 ..... 1429
6.18.2. 概要 ..... 1430
6.18.2.1. SAP MaxDB リソース階層 ..... 1431
6.18.3. 設定上の考慮事項 ..... 1432
6.18.3.1. Raw I/O の使用 ..... 1433
6.18.3.2. DataKeeper でのミラーファイルシステムの使用 ..... 1434
6.18.3.3. 内部向けロードバランサーの使用 ..... 1435
6.18.3.4. アクティブ/スタンバイの場合の考慮事項 ..... 1436
6.18.3.4.1. アクティブ/スタンバイ設定の例 ..... 1438
6.18.3.5. アクティブ/アクティブの場合の考慮事項 ..... 1439
6.18.3.5.1. アクティブ/アクティブ設定の例 ..... 1440
6.18.4. SPS での SAP MaxDB の設定 ..... 1441
6.18.4.1. SAP MaxDB ソフトウェアのインストール ..... 1442
6.18.4.2. SAP MaxDB データベースの作成 ..... 1443
6.18.4.3. User_Key の作成 ..... 1444
6.18.4.4. SPS ソフトウェアのインストール ..... 1446
6.18.5. リソース設定作業 ..... 1447
6.18.5.1. SAP MaxDB リソース階層の作成 ..... 1448
6.18.5.2. SAP MaxDB リソース階層の拡張 ..... 1450

6.18.5.3. SAP MaxDB リソース階層の拡張解除 ..... 1452
6.18.5.4. SAP MaxDB リソース階層の削除 ..... 1453
6.18.5.5. リソース階層のテスト ..... 1454
6.18.6. リソース階層の管理 ..... 1455
6.18.6.1. User_Key の変更 ..... 1456
6.18.6.2. OS User の変更 ..... 1457
6.18.6.3. パラメーターの更新 ..... 1458
6.18.7. トラブルシューティング ..... 1459
6.18.7.1. SAP MaxDB Recovery Kit エラーメッセージ ..... 1461
6.18.8. 付録：Raw I/O によるデバイス領域を作成する ..... 1464
6.18.8.1. Raw I/O セットアップ手順 ..... 1465
6.18.8.2. 階層作成後のデバイス領域の追加 ..... 1466
6.19. Sybase Recovery Kit 管理ガイド ..... 1467
6.19.1. 概要 ..... 1468
6.19.2. ハードウェア要件とソフトウェア要件 ..... 1470
6.19.3. 設定上の考慮事項 ..... 1472
6.19.3.1. raw I/O の使用 ..... 1473
6.19.3.2. DataKeeper でのミラーファイルシステムの使用 ..... 1474
6.19.3.3. interfaces ファイルの考慮事項 ..... 1475
6.19.3.4. Sybase Software Asset Manager (SySAM) ..... 1476
6.19.3.5. アクティブ / スタンバイの考慮事項 ..... 1477
6.19.3.6. アクティブ / アクティブの考慮事項 ..... 1479
6.19.3.7. Sybase Monitor Server と Backup Server ..... 1481
6.19.3.8. Network Attached Storage の使用 ..... 1482
6.19.4. インストール ..... 1485
6.19.4.1. Sybase ASE ソフトウェアのインストール ..... 1487
6.19.4.2. Sybase ASE サーバの作成 ..... 1488
6.19.4.3. LifeKeeper ソフトウェアのインストール ..... 1490
6.19.4.4. Sybase ASE リソース階層の作成 ..... 1491
6.19.4.5. Sybase ASE リソース階層の拡張 ..... 1493
6.19.4.6. Sybase ASE リソース階層の拡張解除 ..... 1495
6.19.4.7. Sybase ASE リソース階層の削除 ..... 1496
6.19.4.8. リソース階層のテスト ..... 1497
6.19.5. 管理 ..... 1498
6.19.5.1. Sybase Backup Server の保護の変更 ..... 1499
6.19.5.2. Sybase Monitor Server の保護の変更 ..... 1501
6.19.5.3. パラメータの更新 ..... 1503
6.19.6. トラブルシューティング ..... 1504
6.19.7. Appendix ..... 1508
6.19.7.1. 要件 ..... 1509
6.19.7.2. 命名規則 ..... 1510
6.19.7.3. raw I/O の設定手順 ..... 1511
6.19.7.4. 階層作成後のデータベースデバイスの追加 ..... 1512
6.19.7.5. ASE と OCS のリンクの作成 ..... 1513
6.20. VMDK as Shared Storage Recovery Kit 管理ガイド ..... 1516

6.20.1. ドキュメンテーションとリファレンス ..... 1517
6.20.2. ハードウェア要件とソフトウェア要件 ..... 1518
6.20.3. 概要 ..... 1519
6.20.4. LifeKeeper for Linux VMDK Recovery Kit の設定 ..... 1520
6.20.4.1. 設定上の考慮事項 ..... 1521
6.20.4.2. 構成例 ..... 1522
6.20.5. LifeKeeper 設定作業 ..... 1524
6.20.5.1. ESXiホストの登録 ..... 1525
6.20.5.2. 仮想マシンオプションの設定変更 ..... 1526
6.20.5.3. リソース階層の作成 ..... 1528
6.20.5.4. リソース階層の削除 ..... 1531
6.20.5.5. 階層の拡張 ..... 1533
6.20.5.6. 階層の拡張解除 ..... 1536
6.20.5.7. リソース階層のテスト ..... 1537
6.20.5.8. メンテナンス ..... 1538
6.20.6. トラブルシューティング ..... 1539
6.20.6.1. エラーメッセージ ..... 1540

**7. パラメーター一覧 ..... 1546**
7.1. EC2パラメーター一覧 ..... 1552
7.2. IPパラメーター一覧 ..... 1555
7.3. MQパラメーター一覧 ..... 1558
7.4. NFSパラメーター一覧 ..... 1563
7.5. Oracleパラメーター一覧 ..... 1564
7.6. PostgreSQLパラメーター一覧 ..... 1565
7.7. Quorumパラメーター一覧 ..... 1567
7.8. Route53 パラメーター一覧 ..... 1571
7.9. SAPパラメーター一覧 ..... 1573
7.10. DataKeeperパラメーター一覧 ..... 1575
7.11. Standby Node Health Check パラメーター一覧 ..... 1580
7.12. SAP HANA パラメーターリスト ..... 1583
7.13. SAP MaxDB パラメーター一覧 ..... 1585

**8. エラーコード一覧 ..... 1587**
8.1. 総合メッセージカタログ ..... 1588
8.1.1. DataKeeper メッセージカタログ ..... 1859
8.1.2. DB2 Kit メッセージカタログ ..... 1883
8.1.3. DMMP Kit メッセージカタログ ..... 1897
8.1.4. EC2 Kit メッセージカタログ ..... 1912
8.1.5. ファイルシステムキットメッセージカタログ ..... 1922
8.1.6. Gen/App キットメッセージカタログ ..... 1944
8.1.7. IP Kit メッセージカタログ ..... 1956
8.1.8. Oracle キットメッセージカタログ ..... 1962
8.1.9. Oracle Listener キットメッセージカタログ ..... 1992
8.1.10. Oracle PDB キットメッセージカタログ ..... 2010
8.1.11. SCSI キットメッセージカタログ ..... 2015

8.1.12. Quick Service Protection Recovery Kit メッセージカタログ ..... 2017
8.1.13. GUI メッセージカタログ ..... 2023
8.1.14. SAP メッセージカタログ ..... 2025
8.1.15. SAP HANA Recovery Kit メッセージカタログ ..... 2049

**9. SIOS Protection Suite for Linux サポートマトリックス ..... 2075**

**10. サポートストレージ ..... 2085**

**11. Evaluation Guides ..... 2106**
11.1. DataKeeper for Linux Evaluation Guide ..... 2107
11.1.1. DataKeeper for Linux Terms to Know ..... 2108
11.1.2. The Evaluation Process ..... 2111
11.1.3. Prepare to Install DK for Linux ..... 2112
11.1.4. Configure Storage for DK for Linux ..... 2116
11.1.5. Install SIOS Protection Suite for Linux ..... 2118
11.1.6. Configure the Cluster – DK for Linux ..... 2121
11.1.7. Test Your DK for Linux Environment ..... 2135
11.2. SIOS Protection Suite for Linux Evaluation Guide for Cloud Environments ..... 2139
11.2.1. Before Starting an Evaluation of SIOS Protection Suite for Linux ..... 2140
11.2.1.1. High Availability, RTO, and RPO ..... 2141
11.2.1.2. SIOS Protection Suite for Linux – Integrated Components ..... 2142
11.2.1.3. Benefits of SIOS Protection Suite for Linux ..... 2143
11.2.1.4. How Workloads Should be Distributed when Migrating to a Cloud Environment ..... 2144
11.2.1.5. Public Cloud Platforms and their Network Structure Differences ..... 2146
11.2.1.6. How a Client Connects to the Active Node ..... 2147
11.2.1.6.1. AWS Route Table Scenario ..... 2149
11.2.1.6.2. AWS Elastic IP Scenario ..... 2151
11.2.1.6.3. AWS Route53 Scenario ..... 2152
11.2.1.6.4. Azure Internal Load Balancer Scenario ..... 2153
11.2.1.6.5. GCP Internal Load Balancer Scenario ..... 2155
11.2.1.7. How does Data Replication between Nodes Work? ..... 2156
11.2.1.8. What is “Split Brain” and How to Avoid It ..... 2157
11.2.2. Documentation Style Used in this Guide ..... 2159
11.2.3. Configuring Network Components and Creating Instances ..... 2161
11.2.3.1. Network Structure Used in this Tutorial ..... 2162
11.2.3.2. Computing Resources Used in this Tutorial ..... 2164
11.2.3.3. Creating an Instance in AWS from Scratch ..... 2167
11.2.3.3.1. Switching between AWS Services ..... 2170
11.2.3.3.2. Deciding on an AWS Region ..... 2171
11.2.3.3.3. Creating the VPC ..... 2172
11.2.3.3.4. Creating a Subnet ..... 2175
11.2.3.3.5. Creating an Internet Gateway and Assigning it to the VPC ..... 2180
11.2.3.3.6. Creating the Route Table ..... 2182
11.2.3.3.7. Creating a Security Group ..... 2186
11.2.3.3.8. Creating the First EC2 Instance ..... 2191
11.2.3.3.9. Creating the Second and Third Instances ..... 2199

11.2.3.4. Creating an Instance in Azure from Scratch ..... 2200
11.2.3.4.1. Switching between Azure Services ..... 2203
11.2.3.4.2. Deciding on an Azure Region ..... 2204
11.2.3.4.3. Creating the Resource Group ..... 2205
11.2.3.4.4. Creating a Network Security Group ..... 2207
11.2.3.4.5. Creating the First Azure Virtual Machine ..... 2213
11.2.3.4.6. Creating the Second and Third Virtual Machines ..... 2223
11.2.3.5. Creating an Instance in GCP from Scratch ..... 2225
11.2.3.5.1. Switching between GCP Services ..... 2228
11.2.3.5.2. Deciding on a GCP Region ..... 2229
11.2.3.5.3. Creating the Project ..... 2230
11.2.3.5.4. Creating a VPC Network ..... 2233
11.2.3.5.5. Creating a New SSH Key ..... 2235
11.2.3.5.6. Creating the First GCP VM ..... 2237
11.2.3.5.7. Configuring the Firewall Rules ..... 2246
11.2.3.5.8. Creating the Second and Third VM ..... 2251
11.2.4. Configure Linux Nodes to Run SIOS Protection Suite for Linux ..... 2252
11.2.4.1. Connecting to a Linux Node from Windows Client Using ssh ..... 2253
11.2.4.2. Set a Hostname for Each Instance ..... 2257
11.2.4.3. Disable SELinux ..... 2259
11.2.4.4. Disable the Firewall ..... 2261
11.2.4.5. Set a Password for the Root User ..... 2262
11.2.4.6. Install x11 ..... 2263
11.2.5. Install SIOS Protection Suite for Linux ..... 2265
11.2.5.1. Install AWS CLI ..... 2270
11.2.5.2. Assign Permission to Use EC2 Recovery Kit ..... 2271
11.2.5.3. Disable PING Broadcasting ..... 2272
11.2.6. Login and Basic Configuration Tasks ..... 2273
11.2.6.1. Connecting to a Linux Node with “X11 Forwarding” ..... 2275
11.2.6.2. Setup X Window Client Software on Microsoft Windows ..... 2277
11.2.6.3. Connecting to the First Node (node-a) ..... 2281
11.2.6.4. Connecting to Other Nodes (node-b and node-c) ..... 2283
11.2.6.5. Define Communication Paths ..... 2285
11.2.7. Protecting Our Resources ..... 2289
11.2.7.1. Creating an IP Resource ..... 2292
11.2.7.2. Switching between Nodes in a Cloud Environment ..... 2297
11.2.7.2.1. Creating an AWS EC2 Resource (RouteTable Scenario) ..... 2298
11.2.7.2.2. Creating an AWS EC2 Resource (Elastic IP Scenario) ..... 2303
11.2.7.2.3. Creating an AWS Route53 Resource ..... 2307
11.2.7.2.4. Azure – Using an Internal Load Balancer ..... 2318
11.2.7.2.5. GCP – Using an Internal Load Balancer ..... 2325
11.2.7.2.6. Responding to Load Balancer Health Checks ..... 2336
11.2.7.3. Switch to Standby Node to Confirm Switchover is Working ..... 2343
11.2.7.4. How to Create Data Replication of a File System ..... 2346
11.2.7.4.1. How to Prepare Disks for Replication on AWS ..... 2353

11.2.7.4.2. How to Prepare Disks for Replication on Azure ..... 2356
11.2.7.4.3. How to Prepare Disks for Replication on GCP ..... 2358
11.2.7.5. How to Protect Other Resources (Databases or Applications) ..... 2360
11.2.7.5.1. Protecting an Oracle Resource (non-PDB) ..... 2361
11.2.7.5.1.1. Install Oracle ..... 2366
11.2.7.5.1.2. Create an Oracle Database (non-PDB) ..... 2367
11.2.7.5.1.3. Stop the Oracle Instance ..... 2369
11.2.7.5.1.4. Rename /datakeeper/oradata/ORCL ..... 2371
11.2.7.5.1.5. Update Config File for Oracle Listener on Both Nodes ..... 2373
11.2.7.5.1.6. Start Database and Listener on node-a ..... 2375
11.2.7.5.1.7. Configure Oracle LISTENER Resource ..... 2376
11.2.7.5.1.8. Configure the Oracle Resource ..... 2383
11.2.7.5.1.9. Test Switchover of the Oracle Resource ..... 2388
11.2.7.5.2. Protecting MSSQL Using Quick Service Protection ..... 2392
11.2.7.5.2.1. Install MSSQL 2017 ..... 2396
11.2.7.5.2.2. Relocate Master Database and Log Files to Replicated Storage ..... 2397
11.2.7.5.2.3. Rename Folders Under /dataKeeper/MSSQL ..... 2399
11.2.7.5.2.4. Configure MSSQL Resource ..... 2401
11.2.7.5.2.5. Customize LocalRecovery Parameter on Both Nodes ..... 2407
11.2.7.5.2.6. Update Dependency between Resources ..... 2409
11.2.7.5.3. Protecting a PostgreSQL Resource ..... 2412
11.2.7.5.3.1. Install Postgres 12 on Linux Nodes ..... 2416
11.2.7.5.4. Protecting an NFS Resource ..... 2418
11.2.7.5.5. Protecting SAP Resources ..... 2422
11.2.7.5.5.1. Create ASCS and ERS Virtual IPs ..... 2426
11.2.7.5.5.1.1. GCP – Creating ASCS and ERS Internal Load Balancers ..... 2427
11.2.7.5.5.2. Create SAP NFS Shared File Systems ..... 2433
11.2.7.5.5.2.1. GCP – Creating SAP NFS Shared File Systems ..... 2434
11.2.7.5.5.3. Install SAP Instances ..... 2438
11.2.7.5.5.4. Create LifeKeeper SAP Resources ..... 2446
11.2.7.5.5.5. Test Switchover and Failover ..... 2454
11.2.7.5.6. Protecting SAP HANA Resources ..... 2460
11.2.7.5.6.1. Create SAP HANA Primary Database Virtual IP ..... 2463
11.2.7.5.6.1.1. GCP – Create SAP HANA Primary Database Load Balancer ..... 2464
11.2.7.5.6.2. Attach Disks for SAP HANA File Systems ..... 2468
11.2.7.5.6.3. Install SAP HANA and Configure System Replication ..... 2470
11.2.7.5.6.4. Create LifeKeeper SAP HANA Resource ..... 2476
11.2.7.5.6.5. Test Switchover and Failover ..... 2480
11.2.7.6. Common Tasks ..... 2489
11.2.7.6.1. How to Confirm if the Data Storage is Available on a Node ..... 2490
11.2.7.6.2. Switchover the Data Storage to the Other Node ..... 2491

**12. Quick Start Guides ..... 2493**
12.1. AWS Direct Connect クイックスタートガイド ..... 2494
12.1.1. 利用のための要件 ..... 2495
12.1.2. AWS Direct Connect 構築手順 ..... 2497

12.1.2.1. AWS Direct Connect 準備 ..... 2498
12.1.2.2. Direct Connect リソースの作成 ..... 2499
12.1.2.3. ルートテーブルの設定 ..... 2500
12.1.3. 本構成における設定および運用上の留意点 ..... 2501
12.2. Microsoft Azure Quick Start Guide ..... 2502
12.2.1. Microsoft Azure Overview ..... 2503
12.2.2. 構成情報 ..... 2504
12.2.3. LifeKeeper-Specific Configurations in Azure ..... 2507
12.2.4. Building a Virtual Machine and Starting the OS ..... 2509
12.2.4.1. Creating a Resource Group ..... 2510
12.2.4.2. Creating a Virtual Network ..... 2511
12.2.4.3. Creating a Virtual Machine ..... 2513
12.2.4.3.1. Creating a Cluster Node (Active) ..... 2514
12.2.4.3.2. Creating a Cluster Node (Standby) ..... 2525
12.2.4.3.3. Creating a Client and Witness Server ..... 2528
12.2.4.4. Creating a Load Balancer ..... 2531
12.2.4.5. Configuring the OS ..... 2538
12.2.5. Building an HA Cluster with LifeKeeper ..... 2545
12.2.5.1. Configuring Quorum/Witness ..... 2546
12.2.5.2. Creating a Communication Path ..... 2547
12.2.5.3. Disable Broadcast Ping ..... 2548
12.2.5.4. Creating IP Resources ..... 2549
12.2.5.5. Creating a Data Replication Resource Hierarchy ..... 2552
12.2.5.6. Creating an Oracle Resource Hierarchy ..... 2558
12.2.5.6.1. Installing Oracle DB ..... 2562
12.2.5.6.2. Configuring a Listener ..... 2566
12.2.5.6.3. Creating the DB ..... 2571
12.2.5.6.4. Configuring Oracle ..... 2572
12.2.5.6.5. Creating an Oracle Database Listener Resource Hierarchy ..... 2574
12.2.5.6.6. Creating an Oracle Resource Hierarchy ..... 2579
12.2.5.7. Connectivity Check ..... 2584
12.2.6. Availability Zone (High Availability Zone) ..... 2587
12.2.6.1. Azure Configuration ..... 2589
12.2.6.2. Checking the Client Redirection ..... 2593
12.2.7. References and Acknowledgements ..... 2595
12.3. SIOS Protection Suite for Linux in the AWS Cloud (SAP) ..... 2596
12.3.1. SPS 上で SAP を設定する際の追加手順 ..... 2597
12.3.2. NFS なしの ASCS 構成 ..... 2603
12.3.2.1. 一般的なセットアップのステップ手順 ..... 2604
12.3.2.2. SAP のインストール ..... 2605
12.3.2.3. SAP リソース階層の作成 ..... 2606
12.3.3. NFS を使用している ASCS + ERS 構成 ..... 2611
12.3.3.1. 一般的なセットアップのステップ手順 ..... 2612
12.3.3.2. SAP のインストール ..... 2614
12.3.3.3. NFS のセットアップ ..... 2616

12.3.3.4. NFS リソース階層の作成 ... 2621
12.3.3.5. SAP リソース階層の作成 ... 2624
12.3.3.6. ERS リソースの作成 ... 2629
12.3.3.7. ENSAv2/ERSv2使用時にASCS/ERSに回避動作を強制する ... 2631
12.3.4. スイッチオーバーおよびフェールオーバーのテスト ... 2640
12.3.4.1. その他のリソース ... 2642
12.4. Transit Gatewayを使用したLifeKeeperクラスターとクライアント間接続クイックスタートガイド ... 2643
12.4.1. 利用のための要件 ... 2645
12.4.1.1. LifeKeeper ソフトウェアの要件 ... 2647
12.4.2. AWS VPC ピア接続構築手順 ... 2648
12.4.3. ルートテーブルの設定 ... 2649
12.4.4. 本構成における設定および運用上の留意点 ... 2650
12.4.4.1. LifeKeeper I/O フェンシングの利用を検討してください ... 2651
12.4.5. 既知の問題とトラブルシューティング ... 2652
12.5. AWS VPC ピアリングによる LifeKeeper クラスター間接続クイックスタートガイド ... 2653
12.5.1. 利用のための必要要件 ... 2655
12.5.1.1. LifeKeeper ソフトウェアの要件 ... 2657
12.5.1.2. その他 ... 2658
12.5.2. 構築手順 ... 2659
12.5.3. 関連する LifeKeeper リソースについて ... 2661
12.5.4. 本構成における設定および運用上の留意点 ... 2662
12.5.4.1. LifeKeeper I-O フェンシングの利用を検討してください ... 2663
12.6. MySQL Cluster with Data Replication (“Shared Nothing” Cluster) ... 2664
12.6.1. Terms to Know ... 2665
12.6.2. The Evaluation Process – MySQL Cluster ... 2668
12.6.3. Prepare to Install ... 2669
12.6.4. Configure Storage ... 2673
12.6.5. Install, Configure, and Start MySQL ... 2676
12.6.6. Install SIOS Protection Suite for Linux – MySQL Cluster ... 2679
12.6.7. Configure the Cluster ... 2682
12.6.8. Test Your Environment ... 2700
12.7. PostgreSQL Cluster with Shared Storage (ISCSI) ... 2707
12.7.1. Terms to Know – PostgreSQL ... 2708
12.7.2. The Evaluation Process – PostgreSQL ... 2711
12.7.3. Prepare to Install – PostgreSQL ... 2712
12.7.4. Configure Storage – PostgreSQL ... 2716
12.7.5. Install, Configure, and Start PostgreSQL ... 2718
12.7.6. Install SIOS Protection Suite for Linux – PostgreSQL ... 2720
12.7.7. Configure the Cluster – PostgreSQL ... 2723
12.7.8. Test Your Environment – PostgreSQL ... 2738
12.8. Apache/MySQL Cluster Using Both Shared and Replicated Storage ... 2743
12.8.1. Terms to Know – Apache ... 2744
12.8.2. The Evaluation Process – Apache ... 2747
12.8.3. Prepare to Install – Apache ... 2748

12.8.4. Configure Storage – Apache ..... 2752
12.8.5. Install and Configure Apache and PHP ..... 2756
12.8.6. Install, Configure, and Start MySQL – Apache ..... 2758
12.8.7. Install SIOS Protection Suite for Linux – Apache ..... 2761
12.8.8. Configure the Cluster – Apache ..... 2764
12.8.9. Test Your Environment – Apache ..... 2787

**13. LifeKeeper Single Server Protection ..... 2798**
13.1. LifeKeeper Single Server Protection リリースノート ..... 2799
13.2. LifeKeeper Single Server Protection インストレーションガイド ..... 2815
13.2.1. はじめに ..... 2817
13.2.2. LifeKeeper Single Server Protection ソフトウェアのインストール ..... 2819
13.2.3. セットアップスクリプトの操作 ..... 2821
13.2.4. SSP のアップグレード ..... 2827
13.2.5. ライセンスの取得とインストール ..... 2828
13.2.6. リソースポリシー管理 ..... 2831
13.2.7. LifeKeeper Single Server Protection のインストールの検証 ..... 2836

**14. LifeKeeper Single Server Protection テクニカルドキュメンテーション ..... 2837**
14.1. ドキュメンテーションとトレーニング ..... 2839
14.2. VMware HA との連携 ..... 2841
14.3. LifeKeeper Single Server Protection の管理の概要 ..... 2842
14.3.1. VMware HA と LifeKeeper Single Server Protection の連携を有効にする ..... 2843
14.3.2. VMware HA を有効化した障害検出およびリカバリシナリオ ..... 2844
14.3.3. LifeKeeper Single Server Protection ハートビートと VMware HA ..... 2845
14.3.4. LifeKeeper Single Server Protection で保護するシステムのメンテナンス ..... 2846
14.3.5. Quick Service Protection (QSP) Recovery Kit ..... 2847
14.3.6. LifeKeeper API for Monitoring ..... 2848
14.3.7. Watchdog ..... 2849
14.3.8. LKCLI（LifeKeeper コマンドラインインターフェース） ..... 2852
14.4. トラブルシューティング ..... 2854
14.4.1. 既知の問題と回避策 ..... 2855
14.5. アプリケーションリカバリーキット ..... 2859

# 1. SIOS Protection Suite for Linux

# 2. SIOS Protection Suite for Linux リリースノート

バージョン **9.5.2**

リリース日：**11 August, 2021**

> ✻ 注意：本製品をインストールまたは使用するまえに、必ずこのドキュメントをお読みください。本ドキュメントには、インストール時とその前後に留意すべき重要な情報が記載されています。

## はじめに

このリリースノート（以下本文書）は、 SIOS Protection Suite (SPS) for Linux 製品のインストール、設定、管理を行うユーザーを対象として記述されています。本文書には、製品のテクニカルドキュメンテーション には詳細に記述されていない重要な情報（製品の最終テスト時に明らかになったパッケージのバージョン、指示や手順に関する最終段階での変更点、トラブルシューティングセクションへのリンク、製品の制限、トラブル解決のヒントなど）が記載されています。SPS for Linux をインストールして設定する前に、必ず本文書の内容を確認してください。

## SPS for Linuxの製品説明

### LifeKeeper for Linux

LifeKeeper for Linux（以下LifeKeeper）は、Linux 上で動作する様々なリソースの高可用性を実現する、アクティブ･スタンバイ型の HAクラスターソフトウェアです。LifeKeeper によって保護されたリソースは、障害発生時やメンテナンス時、稼働するサーバーを切り替えてサービスを継続することが可能になります。稼働するサーバーの切り替えが行われた際も、リソースは切り替え前と同等の環境･同等のパフォーマンスでサービスを提供し続けることができます。これはアクティブ･スタンバイ型クラスターの特性です。クラスターを構成するサーバーのうち、主として起動するサーバーを稼働系、その他のサーバーを待機系と呼びます。LifeKeeperでは、待機系サーバーの OS を常時起動しておくことで、切り替えに要する時間を短縮し、結果としてシステムの高い可用性を実現することができます。

LifeKeeper が保護できるリソースには、次の種類があります。

- アプリケーション
- 共有ストレージデバイス（VMWareの仮想ハードディスクを含む）
- Network Attached Storage デバイス
- LVM ボリュームグループと論理ボリューム
- ファイルシステム (ext3、ext4、vxfs、xfs、nfs)

> ✻ 注意：btrfsは、現在 SIOS Protection Suite for Linux でサポートされていません。 bind マウントもサポートされません。詳細は、トラブルシューティング > 既知の問題と制限 をご確認ください。

- 通信リソース（TCP/IP）
- データベースアプリケーション (Oracle 、 MySQL 、 DB2 、 SAP MaxDB 、 PostgreSQL 、 EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server 、 Sybase )
- Web サーバリソース
- Samba リソース
- DataKeeper for Linux
- SAP アプリケーション環境リソース
- WebSphere MQ リソース
- Postfix リソース

### DataKeeper for Linux

DataKeeper for Linux（以下DataKeeper）は、ボリュームベースの同期および非同期のデータレプリケーション機能を提供するソフトウェアです。LifeKeeper によって手動のリソース切り替え（スイッチオーバー）や障害を契機とした自動切り替え（フェイルオーバー）が行われた際、本製品のレプリケーション機能によって、サーバー間のデータを自動的に同一に保ちます。レプリケーションの管理および監視は、LifeKeeper Graphical User Interface（LifeKeeper GUI）より統合的に行えます。

## SPS for Linux のコンポーネント

### SPS コア

SPS for Linux の中核となるソフトウェアです。64bit システム（AMD64、EM64T システム）でのみ稼動します。
SPS コアは、次のパッケージが含まれ、クラスターを構成する各サーバーにインストールして使用されます。

| パッケージ | パッケージ名 | 説明 |
|---|---|---|
| LifeKeeper Core | steeleye-lk-9.5.2-7301.x86_64.rpm | メモリー、CPU 、OS 、SCSI ディスクサブシステム 、ファイルシステムなど、のシステムの中核となるコンポーネントの障害回復機能を提供します。 |
| LifeKeeper GUI | steeleye-lkGUI-9.5.2-7301.x86_64.rpm | LifeKeeper および Datakeeper の管理および健全性監視用のグラフィカルユーザーインターフェースです。 |
| SPS IP Recovery Kit | steeleye-lkIP-9.5.2-7301.noarch.rpm | IP アドレスの障害回復機能を提供します。 |
| SPS Raw | steeleye- | raw I/O を使用するアプリケーションを保護します。 |

| | | |
|---|---|---|
| I/O Recovery Kit | lkRAW-9.5.2-7301.noarch.rpm | |
| Quick Service Protection | steeleye-lkQSP-9.5.2-7301.noarch.rpm | 様々なサービスに対して、簡易的な障害回復機能を提供します。 |
| LifeKeeper Man Page | steeleye-lkMAN-9.5.2-7301.noarch.rpm | SPS製品のリファレンスマニュアルのページが含まれます。 |

## SPS のオプションリカバリーソフトウェア

次のオプションリカバリーソフトウェア（以下リカバリーキット）は、アプリケーションのリソース定義を行うとともに、リカバリー機能を提供します。各リカバリーキットがサポートするアプリケーションのバージョンや、必要な要件については、 サポートマトリックス および各リカバリーキットの管理ガイドを参照してください。

| パッケージ | パッケージ名 | 説明 |
|---|---|---|
| SPS Apache Web Server Recovery Kit | steeleye-lkAPA-9.5.2-7301.noarch.rpm | Apache Web Server ソフトウェアの障害回復機能を提供します。 |
| SPS SAP Recovery Kit | steeleye-lkSAP-9.5.2-7301.noarch.rpm | SAP NetWeaver の障害回復機能を提供します。また 、 他のいくつかの SPS Recovery Kit と連携して包括的にフェイルオーバーを行います。 |
| SPS SAP MaxDB Recovery Kit | steeleye-lkSAPDB-9.5.2-7301.noarch.rpm | SAP MaxDB データベースの障害回復機能を提供します。 |
| SPS DB2 Recovery Kit | steeleye-lkDB2-9.5.2-7301.noarch.rpm | DB2 データベースインスタンスの障害回復機能を提供します。 |
| SPS Oracle Recovery Kit | steeleye-lkORA-9.5.2-7301.noarch.rpm | Oracle ソフトウェアの障害回復機能を提供します。 |
| SPS MySQL Recovery Kit | steeleye-lkSQL-9.5.2-7301.noarch.rpm | MySQL の障害回復機能を提供します。 |
| SPS PostgreSQL Recovery Kit | steeleye-lkPGSQL-9.5.2-7301.noarch.rpm | PostgreSQL インスタンスの障害回復機能を提供します。 |
| SPS Sybase ASE Recovery Kit | steeleye-lkSYBASE-9.5.2-7301.noarch.rpm | Sybase ASE コンポーネントのAdaptive Server、Monitor Server、Backup Server の障害回復機能を提供します。 |
| SPS Postfix Recovery Kit | steeleye-lkPOSTFIX-9.5.2-7301.noarch.rpm | Postfixの障害回復機能を提供します。 |
| SPS Samba Recovery Kit | steeleye-lkSMB-9.5.2-7301.noarch.rpm | 異種ネットワーク内の Linux サーバー上の Samba によるファイルと印刷の共有に対して障 |

| | | |
|---|---|---|
| | | 害回復機能を提供します。 |
| SPS NFS Server Recovery Kit | steeleye-lkNFS-9.5.2-7301.noarch.rpm | Network File System(NFS) ソフトウェアの障害回復機能を提供します。 |
| SPS Network Attached Storage Recovery Kit | steeleye-lkNAS-9.5.2-7301.noarch.rpm | Network Attached Storage（NAS）によりエクスポートされた領域を使用するアプリケーションを保護します。 |
| SPS Logical Volume Manager (LVM) Recovery Kit | steeleye-lkLVM-9.5.2-7301.noarch.rpm | 他の SPS Recovery Kit に対して論理ボリュームのサポートを提供します。SPS で保護されたアプリケーションは、ストレージ管理の簡素化、要件変更に応じた動的なボリュームのサイズ変更など、Logical Volume Manager が提供するメリットを活用できます。 |
| SPS PowerPath Recovery Kit | steeleye-lkPPATH-9.5.2-7301.noarch.rpm | EMC PowerPath マルチパス I/O デバイスを使用するアプリケーションおよびファイルシステムを保護します。 |
| SPS Device Mapper Multipath (DMMP) Recovery Kit | steeleye-lkDMMP-9.5.2-7301.noarch.rpm | DMMP デバイスを使用するアプリケーションおよびファイルシステムを保護します。 |
| Hitachi Dynamic Link Manager Software (HDLM) Recovery Kit | steeleye-lkHDLM-9.5.2-7301.noarch.rpm | Hitachi Dynamic Link Manager Software デバイスを使用するアプリケーションおよびファイルシステムを保護します。 |
| SPS NEC iStorage StoragePathSavior (NECSPS) Recovery Kit | steeleye-lkSPS-9.5.2-7301.noarch.rpm | NEC iStorage StoragePathSavior v3.3 以降のマルチパス I/O デバイスを使用するアプリケーションおよびファイルシステムを保護します。 |
| SIOS DataKeeper | steeleye-lkDR-9.5.2-7301.noarch.rpm | SPS 環境に統合データミラーリング機能を提供します。物理的な共有ストレージを持たない環境でも、 SPSリソースの運用が可能になります。 |
| SPS WebSphere MQ Recovery Kit | steeleye-lkMQS-9.5.2-7301.noarch.rpm | WebSphere MQ キューマネージャーおよびキューマネージャーストレージサイトの障害回復機能を提供します。 |
| Quorum/Witness Package | steeleye-lkQWK-9.5.2-7301.noarch.rpm | SCSI リザベーションによる排他制御を利用できない 共有ストレージ（DataKeeperやNASなど）をもつ環境で、スプリットブレイン状態の発生を抑制してより安全なクラスターを構築することができます。 |
| Recovery Kit for EC2™ | steeleye-lkECC-9.5.2-7301.noarch.rpm | AWS環境で、SPSでクラスター化されたサービスに対して、クライアントから切り替え前後で同一のIPアドレスを使ってアクセスを行うことが |

| | | |
|---|---|---|
| | | できます。 |
| Recovery Kit for Route 53™ | steeleye-lkROUTE53-9.5.2-7301.noarch.rpm | AWS Route 53 上のDNSレコードを更新し、接続先の切り替えを行う機能を提供します。AWS環境で、クライアントがSPSでクラスター化されたサービスに対して、Direct Connect 接続のクライアントまたは異なるAmazon VPC上のクライアントからアクセスを行うことができます。 |
| VMDK as Shared Storage Recovery Kit | steeleye-lkVMDK-9.5.2-7301.noarch.rpm | 共有ディスクとして使用している VMware の仮想ハードディスクおよびそのファイルシステムを LifeKeeperリソースとして保護することが可能になります。 |

# SPS for Linux Version 9 の新機能

| 製品 | 機能 |
|---|---|
| バージョン **9.5.2** の新機能 | |
| LifeKeeper Core | Red Hat Enterprise Linux 8.4 をサポートしました。 |
| | CentOS 8.3 をサポートしました。 |
| | Oracle Linux 8.3 をサポートしました。 |
| | Oracle Linux 7 UEK 6をサポートしました。<br>注意：カーネルバージョンが5.4.17-2102.202.5 以降のみサポートされます。 |
| | Oracle Linux 8 UEK 6をサポートしました。<br>注意：カーネルバージョンが5.4.17-2102.202.5 以降のみサポートされます。 |
| | 以下のARKでもLKCLIを利用できるようになりました。<br>• SAP RK<br>• SAP MaxDB RK<br>• MQ RK (デモ動画 (英語版) はこちら)<br>• DMMP RK<br>• HDLM RK<br>• NEC SPS RK |
| | 以下のARKにおいてスタンバイノード監視機能が利用できるようになりました。<br>• Filesystem RK<br>• NAS RK |
| | lk group に所属したroot以外のユーザーからでも lkcliを実行できるようになりました。 |
| | lkcli stop でも lkstop 同様に「-i」オプションを使用できるようになりました。 |
| | バグの修正 |

| | |
|---|---|
| MySQL | MariaDB 10.5 をサポートしました。 |
| SAP HANA | SAP HANA support for RHEL 8.2 をサポートしました。 |
| | SAP HANA support for SLES 15.2 をサポートしました。 |
| | SAP HANA データベースの Takeover with Handshake機能をサポートしました。 |
| | /etc/default/LifeKeeperに調整可能なパラメーターを追加しました。 |
| | バグの修正 |
| MQ | WebSphere MQ 9.2 for RHEL 7.9 をサポートしました。 |
| | WebSphere MQ 9.2 for RHEL 8.3 をサポートしました。 |
| DataKeeper、Install、NFS、IP、Oracle、SAP、Quorum/Witness、Filesystem、Generic、lksupport、DMMP | バグの修正 |
| **バージョン 9.5.1 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 7.9 をサポート。（2020年12月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | CentOS 7.9 をサポート。（2020年12月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | Oracle Linux 7.9 をサポート。（2020年12月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | Microsoft Azure の Load Balancer を利用した構成において、任意のポートでヘルスチェックプローブに応答する機能を追加。（2020年12月認定）<br>詳細は こちら をご参照ください。 |
| | Google Cloud の Cloud Load Balancing を利用した構成において、任意のポートでヘルスチェックプローブに応答する機能を追加。（2020年12月認定）<br>詳細は こちら をご参照ください。 |
| | 以下の ARK で LKCLI の利用をサポート。<br>• DB2 RDBMS RK<br>• RAW RK<br>• Postfix RK<br>• VMDK as Shared Storage RK<br>• SAP ASE RK<br>• Samba RK<br>• HULFT RK / HULFT HUB RK |

| | |
|---|---|
| | • Oracle RK の Oracle PDB 保護機能 |
| | Red Hat Enterprise Linux 8.3 をサポート。（2021年1月認定） |
| | Red Hat Enterprise Linux 8.2 をサポート。 |
| | CentOS 8.2 をサポート。 |
| | Oracle Linux 8.2（UEK6 は除く）をサポート。 |
| | SLES15 SP2 をサポート。 |
| | LifeKeeper 内部で使用している OpenSSL パッケージを 1.1.1g に更新。 |
| | LifeKeeper 内部で使用している cURL パッケージを 7.68.0 に更新。 |
| | lkstop コマンドに -i オプション を追加。このオプションは保護するリソースを停止せずに LifeKeeper のみを停止します。また、停止処理を続行するために「yes/no」の確認を行う必要があります。 |
| | バグの修正 |
| MQ | SIOS Protection Suite for Linux は IBM MQ 9.2 のサポートを開始しました。(2021年1月) |
| QSP | Quick Service Protection(QSP) RK によるApache Tomcat の保護をサポート。 |
| install | LifeKeeper の 構成のみ行うセットアップのモード をサポート。その他、セットアップに関する改修。 |
| | バグの修正 |
| PostgreSQL | PostgreSQL 13 をサポートをサポート。（2020年12月認定） |
| | EDB Postgres Advanced Server 13.0 をサポート。（2020年12月認定） |
| | FUJITSU Software Enterprise Postgres 12 をサポート。（2020年12月認定） |
| SAP HANA | SAP HANA2 SP5 をサポート。（2020年12月認定） |
| | SAP HANA support for RHEL 8.1 をサポート。（2020年12月認定） |
| | SAP HANA support for SLES 12.5 をサポート。（2020年12月認定） |
| | バグの修正 |
| SAP | SAP S/4HANA 2020をサポート。（2020年12月認定） |
| | バグの修正 |
| SAP MaxDB、DB2、DataKeeper、Route 53、Filesystem、SAP、GenericApplication、IP | バグの修正 |
| バージョン **9.5.0** の新機能 | |

| | |
|---|---|
| SAP HANA | SAP HANA Recovery Kitの提供を開始。 詳細については、 SAP HANA Recovery Kit管理ガイド を参照ください。<br><br>注意：<br>• SIOS Protection Suite for Linux v9.5以降をご利用の場合は、新しい（ビルトインの）SAP HANA Application Recovery Kitを使用する必要があります。<br>• 既存のSAP HANA gen/appベースのRecovery Kitはv9.5.0では サポートされていません 。SIOS Protection Suite for Linux v9.5.0へのアップグレードを希望するユーザーは、既存のSAP HANA gen/appベースのRecovery Kitから新しいSAP HANA Recovery Kitに 移行する必要があります 。詳細については、 SAP HANA Gen/AppからSAP HANA Recovery Kitへのアップグレード を参照してください。<br>• SIOSは 2022年3月31日 まで、9.4.xリリースでのSAP HANA gen/appベースのRecovery Kitのサポートを継続します。<br>• SAP HANA Recovery Kit は、HANA v1 をサポートしていません。 |
| LifeKeeper Core | Red Hat Enterprise Linux 7.8 をサポート。（2020年7月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | CentOS 7.8 をサポート。（2020年7月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | Oracle Linux 7.8 をサポート。（2020年7月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | SUSE Linux Enterprise Server 12 SP5 をサポート。（2020年7月認定） |
| | VMware vSphere 7.0 をサポート。（2020年7月認定） |
| | CentOS 8.0 をサポート。 |
| | Oracle Linux 8.0 をサポート。 |
| | Red Hat Enterprise Linux 8.1 をサポート。 |
| | CentOS 8.1 をサポート。 |
| | Oracle Linux 8.1 をサポート。 |
| | CLI の強化。CLI で LifeKeeper を制御できるようになりました。詳細は「LKCLI」を参照してください。 |
| | バグの修正 |
| Oracle | Multitenant 構成による PDB の保護をサポート。詳細は「Oracle Multitenant 構成による Pluggable Database の設定」を参照してください。 |
| PostgreSQL | EDB Postgres Advanced Server 12.0 をサポート。（2020年7月認定） |
| | PostgreSQL 12 をサポート。 |

| | |
|---|---|
| DataKeeper | ミラーボリュームのサイズのオンライン変更をサポート。詳細は「ミラーのサイズ変更」を参照してください。 |
| | 障害時の復帰処理を並列で行うよう変更。 |
| | LKDR_CONNECT_NBD_DURING_RESTORE パラメーターを追加。詳細については「DataKeeper パラメーター一覧」を参照してください。 |
| | バグの修正 |
| Filesystem, LVM, NFS, IP, DB2, MaxDB, SAP, Sybase, Sybase ASE, Quorum/ Witness | バグの修正 |
| **バージョン 9.4.1 の新機能** | |
| LifeKeeper Core | OS 同梱の OpenJDK をインストールするようになりました。詳細につきましてはテクニカルドキュメンテーションの LifeKeeper の GUI の設定 のページを参照してください。<br><br>（2020年12月追記）<br>一部の環境では LifeKeeper のインストールイメージに同梱の OpenJDK をインストールしています。テクニカルドキュメンテーションの LifeKeeper の GUI の設定 のページを参照してください。 |
| | SUSE Linux Enterprise Server 15 SP1 をサポートしました。 |
| | Oracle Linux 7.7 をサポートしました。 |
| | CentOS 7.7をサポートしました。 |
| | AWS Nitro systemをサポートしました。 |
| | AWS Transit Gatewayをサポートしました。 |
| | バグの修正 |
| DataKeeper | DataKeeperがNVMeデバイスに対応しました。 |
| | バグの修正 |
| VMDK as Shared Storage | LifeKeeper for Linux VMDK as Shared Storage Recovery Kitがリリースされました。詳細につきましては VMDK as Shared Storage 管理ガイド を参照してください。 |
| PostgreSQL | PostgreSQL Recovery Kitを使用してPowerGres Plus (Linux 版) v10とPowerGres on Linux v11を保護できるようになりました。 |
| | FUJITSU Software Symfoware Server (Postgres) V12.4 がサポートされました。<br>（2020年3月認定）<br>オプションの SPS リカバリソフトウェアの要件 、 PostgreSQL Recovery Kit 管理ガイド > 管理 をご参照ください。 |

| | |
|---|---|
| Install、IP、MaxDB、EC2 | バグの修正 |
| **バージョン 9.4 の新機能** | |
| LifeKeeper Core | スタンバイノード監視機能を追加しました。ユーザは、スタンバイノードのCPUとメモリーの使用率を監視し、Out-of-Service（OSU）リソースのヘルスを監視して、スタンバイノードのエラーを検出できます。 |
| | Red Hat Enterprise Linux 8.0 をサポートしました。<br>注意：RHEL 7 から RHEL 8 のようにカーネルをメジャーバージョンアップすることはサポートされません。（RHEL 7 から RHEL 8 にカーネルをアップグレードした環境では、DataKeeper リソースは動作しません。） |
| | Red Hat Enterprise Linux 7.7 をサポートしました。（2019年11月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら“をご参照ください。 |
| | Oracle Linux 7 UEK5をサポートしました。 |
| | バグの修正 |
| MySQL | MariaDB 10.3 をサポートしました。 |
| DB2 | DB2 11.5をサポートしました。 |
| PostgreSQL | FUJITSU Software Enterprise Postgres 11 をサポートしました。（2019年11月認定） |
| SAP | SAP-certified support of SAP S/4HANA Platform via SAP High Availability Clustering Certification S/4-HA-CLU-1.0<br>SAP S/4HANA 1809 Platform がサポートされました。<br>SAP S/4HANA 1909 Platform がサポートされました。（2019年11月サポート追加） |
| | スタンドアロンエンキューサーバー 2（ENSAv2） および エンキューレプリケーションサーバー（ERSv2） をサポートしました |
| | SAPリソースUIの機能 を強化しました |
| | LifeKeeper SAP ERSリソース を最適化しました |
| | バグの修正 |
| DataKeeper、Filesystem、EC2、Oracle、Quorum/Witness、Install | バグの修正 |
| **バージョン 9.3.2 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 7.6 をサポートしました。<br>ただし DataKeeper の非同期モードはサポートされません。 |
| | CentOS 7.6 をサポートしました。<br>ただし DataKeeper の非同期モードはサポートされません。 |

| | |
|---|---|
| | Oracle Linux 7.6 をサポートしました。<br>ただし DataKeeper の非同期モードはサポートされません。<br>Unbreakable Enterprise Kernel Release 5 (UEK R5) はサポートされません。<br>（DataKeeper リソースはUEK R5 上では動作しません。） |
| | SUSE Linux Enterprise Server 12 SP4 をサポートしました。 |
| | SUSE Linux Enterprise Server 15 をサポートしました。<br>ただし 、 DataKeeper にセクターサイズが奇数のディスクは利用できません。<br>SLES 12 から SLES 15 のように Kernel をメジャーバージョンアップすることはサポートされません。 （SLES 12 から SLES 15 に Kernel をアップグレードした環境では 、 DataKeeper リソースは動作しません。） |
| Install | 現在の setup コンフィグレーションを保存するための -s オプションを setup コマンドに追加しました。 |
| DataKeeper | マルチターゲットミラーの場合に前のソースを待機します。<br><br>マルチターゲットミラーの場合 、 DataKeeper はミラーが In-service になっている最後のサーバー（前のソース）を追跡します。フェールオーバーが発生した場合 、 すべてのターゲットを同期させるために 、 前のソースのビットマップが必要になります。 DataKeeper はターゲットへのレプリケーションを再開する前に 、 前のソースがクラスタに参加するのを自動的に待機します。これにより前のソースのビットマップがマージされ 、 部分再同期のみ行えばよくなります。 |
| PostgreSQL | PostgreSQL 11 をサポートしました。 |
| | EDB Postgres Advanced Server v11 をサポートしました。 |
| | FUJITSU Software Enterprise Postgres 10 をサポートしました。 |
| MQ | IBM MQ v9.1 をサポートしました。 |
| Oracle | Oracle 19c をサポートしました。 (2019年8月に認定) |
| SELinux | SELinux の permissive モードは、SAP 環境でのみサポートされています。詳細については、「テクニカルノート 」を参照してください。 |
| Core, DataKeeper, Install, SAP, PostgreSQL, Generic, EC2, Apache, GUI, lksupport, Filesystem | バグの修正 |
| **バージョン 9.3.1 の新機能** | |
| LifeKeeper Core | OpenSSLパッケージを 1.0.2p に更新しました。 |
| | Red Hat Enterprise Linux 6.10 をサポートしました。 |
| | CentOS 6.10 をサポートしました。 |
| | Oracle Linux 6 Update 10 をサポートしました。 |

| | |
|---|---|
| MySQL | MySQL 8.0 をサポートしました。 |
| Oracle | Oracle 18c をサポートしました。 (2019年3月に認定) |
| Install 、 EC2 、 Route 53 | バグの修正 |
| **バージョン 9.3 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 7.5 をサポートしました。<br>ただし DataKeeper の非同期モードはサポートされません。 |
| | CentOS 7.5 をサポートしました。<br>ただし DataKeeper の非同期モードはサポートされません。 |
| | Oracle Linux Version 7.5 をサポートしました。<br>ただし DataKeeper の非同期モードはサポートされません。 |
| | VMware vSphere 6.7 をサポートしました。（2018年10月認定） |
| | バグの修正 |
| EC2, Route 53 | EC2 、 Route 53 RKがHTTPプロキシに対応しました。 |
| | バグの修正 |
| Quorum/Witness | ストレージ QWK をサポートしました。詳細はこちら をご参照ください。 |
| | バグの修正 |
| Install | インストールスクリプトが刷新されました。詳細はこちら をご参照ください。 |
| SAP, Oracle, Samba, MQ, Sybase, Filesystem, Generic Application, QSP, SAP MaxDB, DataKeeper | バグの修正 |
| **バージョン 9.2.2 の新機能** | |
| EC2,Route 53 | IAM ロールに対応しました。<br>（Openswan Recovery Kit は IAM ロール未対応のため 、 Cross Region 構成の場合は 、 v9.2.1 をご利用ください。） |
| DataKeeper | 保護対象ディスクを識別する方法として 、 GUID Partition Table（GPT）をサポートしました<br>（Linux kernel 2.6.27以前の場合 、 サポートされるディスクはSCSIハードディスクとXen仮想ディスク(xvd)です。） |
| PostgreSQL | PostgreSQL 10 をサポートしました |
| | EDB Postgres Advanced Server v10.0をサポートしました（2018年4月に認定） |
| SAP,NAS,EC2 | バグの修正 |
| **バージョン 9.2.1 の新機能** | |

| | |
|---|---|
| LifeKeeper Core | Oracle Linux Version 7.4 をサポートしました |
| | CentOS 7.4 をサポートしました |
| | SUSE Linux Enterprise Server 12 SP3 をサポートしました<br>※ SUSE Linux Enterprise Server 12 SP3 を使用する場合は 、 カーネルを 4.4.82-6.9.1 にアップグレードしてください |
| | Recovery Kit for EC2 、 Route 53 Recovery Kit 、 Openswan Recovery Kit をセットアップメニューからインストールできるようになりました。なお Openswan Recovery Kit は CrossRegion 構成でのご利用のみサポート対象となります |
| | バグの修正 |
| PostgreSQL | EDB Postgres Advanced Server 9.6 をサポートしました |
| MQ | IBM MQ 9.0 をサポートしました |
| **バージョン 9.2 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux Version 7.4 をサポートしました |
| | SNMP トラップを複数のターゲットに送信することができるようになりました |
| | バグの修正 |
| IP | 実 IP（NIC に設定されたプライマリ IP アドレス）を用いた IP リソースを作成できるようになりました |
| PostgreSQL | PostgreSQL 9.6 をサポートしました |
| | FUJITSU Software Enterprise Postgres 9.6 をサポートしました<br>詳細はオプションの SPS リカバリソフトウェアの要件 、 PostgreSQL Recovery Kit 管理ガイド > 管理 をご参照ください。 |
| MQ | IBM MQ 9.0 をサポートしました (2017 年 12 月に認定) |
| MD, SAP, SAP MaxDB, Quorum/ Witness, Route 53, Install | バグの修正 |
| **バージョン 9.1.2 の新機能** | |
| LifeKeeper Core | SUSE Linux Enterprise Server 12 SP2 をサポートしました。 |
| | CentOS7.3をサポートしました。 |
| | Red Hat Enterprise Linux Version 6.9をサポートしました。 |
| | Oracle Linux Version 7.3の UEK カーネルをサポートしました。 |
| | バグの修正 |
| PostgreSQL | PostgreSQL 9.6 をサポートしました |
| | FUJITSU Software Enterprise Postgres 9.6 をサポートしました<br>詳細はオプションの SPS リカバリソフトウェアの要件 、 PostgreSQL Recovery Kit |

| | |
|---|---|
| | 管理ガイド > 管理 をご参照ください。 |
| Oracle | Oracle 12c R2をサポートしました。 |
| DB2 | DB2 11.1をサポートしました。 |
| IP, QSP, MySQL, NFS | バグの修正 |
| **バージョン 9.1.1 の新機能** | |
| LifeKeeper Core | SUSE Linux Enterprise Server 12 SP1 のサポート<br>※ SP1 を適用していない SLES12 の利用はサポートされません。<br>※ Btrfs の利用はサポートされません。 |
| | Red Hat Enterprise Linux Version 7.3 のサポート |
| | Oracle Linux Version 7.3 のサポート<br>※ UEK の利用はサポートされません。 |
| | vSphere 6.5 のサポート |
| | バグの修正 |
| PostgreSQL | PostgreSQL 9.5 のサポート<br>EDB Postgres Advanced Server v9.5 のサポート<br>FUJITSU Software Symfoware Server (Openインタフェース) V12.2 のサポート<br>FUJITSU Software Symfoware Server (Postgres) V12.3 のサポート<br>FUJITSU Software Enterprise Postgres 9.5 のサポート<br>詳細はオプションの SPS リカバリソフトウェアの要件 、 PostgreSQL Recovery Kit 管理ガイド > 管理 をご参照ください。 |
| Sybase ASE | Sybase ASE 16.0 のサポート |
| MySQL | RHEL 7.x/CentOS 7.x/OEL 7.x での MySQL 5.7 のサポート<br>※ 上記以外の環境での　MySQL 5.7 は既にサポートされています。 |
| SAP | SAP 7.5 のサポート |
| **バージョン 9.1.0 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 6.8のサポート(2016年9月に認定)<br>CentOS 6.8, Oracle Linux 6.8についてもサポートいたします (2016年9月に認定)。<br>*注意: これらのOS上でのMD RecoveryKitの利用はサポートされません。 |
| | LifeKeeper API for Monitoringのサポート<br>LifeKeeperのステータスやログの情報を提供するAPIを設けました。 |
| | Quick Service Protectionのサポート<br>OSサービスの簡易保護機能を提供します。 |
| | バグの修正 |
| **バージョン 9.0.2 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux Version 7.2のサポート |

| | |
|---|---|
| | ※MySQL RKはRHEL 7.x/CentOS 7.x/OEL 7.xをサポートしていません。 |
| | OpenSSLパッケージを1.0.1qに更新 |
| | バグの修正 |
| MQ | WebSphere MQ – マルチバージョン の WebSphere MQ のサポートが追加されました。本サポートにより 、 バージョン 7.1 、 7.5 、 および 8.x のキューマネージャのすべてを同クラスタノードで保護できるよう になりました。 |
| | mqm userだけがMQ commandを実行するために使えるというRecovery Kitの制限を除外。この変更により 、 mqm groupのどのuser でもRecovery KitでMQ commandを実行することが出来る。 |
| | バグの修正 |
| IP, Filesystem, DMMP, DataKeeper, EC2, PostgreSQL, Power Path, SAP, SAP DB/MaxDB, Oracle | バグの修正 |
| Licensing | FlexNetの新しいバージョンへ更新 |
| **バージョン 9.0.1 の新機能** | |
| LifeKeeper Core | バグの修正 |
| DataKeeper | バグの修正 |
| **バージョン 9.0 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux Version 6 Update 7 サポート (2015 年 10 月に認定) |
| | Community ENTerprise Operating System (CentOS) Version 6 Update 7 サポート (2015 年 10 月に認定) |
| | Oracle Linux Version 6 Update 7 サポート (2015 年 10 月に認定) |
| | SUSE SLES 11 SP4 サポート (2015 年 10 月に認定) |
| | Chefサポート （2ノードのDataReplication構成で 、 IP/FileSystem/Apache/MySQL/PostgreSQLのARKが対象） |
| | 詳細な説明を記載したSPS for Linux のパラメータ一覧 を追加し 、 lkchkconfコマンド を追加 |
| | vSphere 6のサポート |
| | reiserfsファイルシステムタイプのサポートを廃止 |
| | Oracle Linux Version 7.0/7.1でサポートされるARKは 、 LifeKeeper for Linux v8.4.1 と同じです (対象ARK：PostgreSQL, MySQL, Oracle, DB2, Apache, Postfix, DMMP, LVM, NFS, NAS, Samba, MD, EC2, Route 53, Openswan) |
| | バグの修正 |

| | |
|---|---|
| DataKeeper | DataKeeper for LinuxのRewind機能を廃止（バージョンアップする場合は 、 先にRewind機能を停止してください） |
| | バグの修正 |
| GUI | JRE 8u51のサポート（JRE 7はサポートされません） |
| | Chromeブラウザのサポートを廃止 |
| | バグの修正 |

## バグの修正

下記は、本バージョンで修正されたバグおよび拡張機能の一覧です。

| バグ | 説明 |
|---|---|
| PL-3177 | SAP RKにおいてエラーメッセージが表示されないことがある問題を修正しました。 |
| PL-3822 | サポートされていないカーネルバージョンにアップグレードした後にLifeKeeperのアップグレードを開始しようとした時に、アップグレードを中止するようにしました。 |
| PL-4122 | quickCheckの失敗やrecoverの際に取得するログが多くなる問題を修正しました。 |
| PL-4169 | quickCheck実行時に、SAPのエラーメッセージがブロックされることがある問題を修正しました。 |
| PL-4317 | 顧客がシステムをスナップショットからロールバックしたとき、DataKeeper が再同期を行えないバグを修正しました。 |
| PL-4318 | システムの応答に時間がかかるときにログメッセージを出力するようにしました。 |
| PL-4446 | 再起動または再稼働時に複数リソースのミラーリングが復旧に失敗することがある問題を修正しました。 |
| PL-4769 | setupスクリプトでパッケージのアンインストールを行う際に使用中のリソースに対して警告を出すように修正しました。 |
| PL-5140 | block failoverオプションを簡単に設定できるようにしました。 |
| PL-5163 | lkcli resource infoコマンドの終了ステータスがエラー時にも0になることがある問題を修正しました。 |
| PL-5402 | lksupportでgrub2の情報を取得できるようにしました。 |
| PL-5415 | quickCheck実行前にrecoverスクリプトが実行されているか確認するように修正しました。 |
| PL-5594 | 一部環境であったLifeKeeperとOS同梱のOpenJDKで発生するメモリ管理不整合の問題を修正しました。 |
| PL-5596 | lkcli importコマンドでリソースの依存関係が正しく構築されないことがある問題を修正しました。 |

| | |
|---|---|
| PL-5870 | HANA RKにおいてGUIのステータスが正しく表示されないことがある問題を修正しました。 |
| PL-6013 | SAPのremoveスクリプトのログを修正しました。 |
| PL-6050 | Quorumとフェイルオーバーの処理が競合する問題を修正しました。 |
| PL-6608 | Filesystemリソースをimportする際にdiskをマウントするように修正しました。 |
| PL-6616 | システムのシャットダウン時にlifekeeper.serviceとmultipathd.serviceが競合することがある問題を修正しました。 |
| PL-6691 | Oracleのコマンドを実行時にoracle userにセットされている環境変数PERL5LIBをリセットするようにしました。 |
| PL-6706 | アップグレード時にmodprobeがリセットされることでnbdに障害が発生することがある問題を修正しました。 |
| PL-6727 | LKCLIのexport時にクラスターノードの状態を確認するように修正しました。 |
| PL-6780 | lkscsidがメモリ/スレッドリークしてしまう問題を修正しました。 |
| PL-6850 | SAPで共有ディスクにNFSを使用している場合、ログにすべてのマウント情報が見えてしまうことがある問題を修正しました。 |
| PL-7066 | リソースの拡張時に生じるpriorityの問題を修正しました。 |
| PL-7077 | admユーザーのログインシェルがbashでない場合のsr_register, sr_takeover, および出力リダイレクトの問題を修正しました。 |
| PL-7178 | 既にリソースが存在する状態でLKCLI import を行うと、警告文とともに既存リソースがリストアップされるように修正しました。 |
| PL-7185 | IPv6でスイッチオーバー時にルートテーブルを削除してしまうことがある問題を修正しました。 |
| PL-7308 | lkcliのconfigコマンドにおいてallオプションが必須になっている問題を修正しました。 |
| PL-7309 | lkcliのgenappにあるconfigコマンドで追加したスクリプトに実行権限が付与されない問題を修正しました。 |
| PL-7310 | lkcliのcleanコマンドにおいて一部のリソースが削除されないことがある問題を修正しました。 |
| PL-7363 | パラメータWAITFORLCMのデフォルト値を20から30に変更しました。 |
| PL-7858 | IPリソースのrestoreまたはrecover時に物理IPが0.0.0.2に変わることがある問題を修正しました。 |
| PL-8035 | LKのセットアップ時に、オプションとしてインストール対象となるHADRパッケージを指定できるようにしました。 |
| PL-8152 | NFSリソースにおいて、LKCLI importが成功しない問題を修正しました。 |
| PL-8971 | IPリソースのネットマスクが32bitで設定できるようにしました。 |
| PL-9012 | LKCLI importにおける、deviceのvalidationの問題を修正しました。 |
| PL-9033 | NFSリソースを作成する際、依存関係を作成するIPリソースがプライマリとなっているサーバー上でなければならない問題を修正しました。 |

| | |
|---|---|
| PL-9036 | LKCLIのdeleteコマンドでhanfsリソースが残ることがある問題を修正しました。 |
| PL-9056 | ディスクリソースのIDが不適切に設定されることがある問題を修正しました。 |
| PL-9146 | Kernelの不具合を起因として、レプリケーションの部分再同期時にtarget側のデータ破損が起こることがある問題を回避するためのモジュールを作成しました。 |
| PL-9320 | ログメッセージの書き込み時に処理がループすることでシステムがハングすることがある問題を修正しました。 |
| PL-9900 | SLESとUEKでインストールの際にSecureBootが有効になっていた場合、警告を出すようにしました。 |

## SPS for Linux Version 9 で廃止された機能

| 機能 | 説明 |
|---|---|
| **本リリースで廃止された機能** | |
| Core | Red Hat Enterprise Linux 6 はサポートされなくなりました。 |
| | CentOS 6 はサポートされなくなりました。 |
| | Oracle Linux 6 はサポートされなくなりました。 |
| Software RAID | SPS Software RAID (md) Recovery Kit はサポートされなくなりました。 |
| Oracle | Oracle virtual machine (OVM) 3.4.6はサポートされなくなりました。 |
| PostgreSQL | PostgreSQL 9.5はサポートされなくなりました。 |
| | EnterpriseDB Postgres Plus Advanced Server 9.5はサポートされなくなりました。 |
| VMware | VMware vSphere 5.5 , 6.0はサポートされなくなりました。 |
| **バージョン 9.5.1 で廃止された機能** | |
| Core | Chefを用いた環境構築のサポートを終了。 |
| **バージョン 9.5.0 で廃止された機能** | |
| LifeKeeper Core | syslog-ng を用いた system ログ管理のサポートを終了。rsyslog を使用してください。 |
| | SUSE Linux Enterprise Server (SLES) 11.0 ～ SP4 のサポートを終了。 |
| | IAM ロール移行ツール /opt/LifeKeeper/lkadm/bin/aws_iam_migration のサポートを終了。IAM ロールは以前のバージョンで移行を済ませてから使用してください。 |
| DataKeeper | DataKeeper for Linux（DK リソース）を利用して、ディスク識別に DEVNAME を用いる環境のサポートを終了。GPT パーティション（GUID Partition Table）環境を使用してください。 |
| Oracle | Oracle Database Enterprise Edition 11g R2 のサポートを終了。 |

| | |
|---|---|
| | Oracle Database Standard Edition 11g R2 のサポートを終了。 |
| | Oracle Database Standard Edition One 11g R2 のサポートを終了。 |
| MySQL | MariaDB 5.5, 10.0 はサポートのサポートを終了。 |
| PostgreSQL | PostgreSQL 9.4 のサポートを終了。 |
| | EnterpriseDB Postgres Plus Advanced Server 9.4 のサポートを終了。 |
| **バージョン 9.4.1で廃止された機能** | |
| | なし |
| **バージョン 9.4で廃止された機能** | |
| DataKeeper | マルチサイトクラスター機能の廃止。 |

# システム要件

## SPS の製品要件

| 説明 | 要件 |
|---|---|
| Linux オペレーティングシステム | クラスターを構成する全てのサーバーは、同一のオペレーティングシステムを利用する必要があります。個々のオペレーティングシステム情報については、「Linux の設定」を参照してください。 |
| 仮想化環境 | 仮想マシン内で起動するゲスト OS は「Linux の設定」に記載のバージョンを利用する必要があります。<br>サポート対象の仮想化環境の詳細については、サポートマトリックス を参照ください。<br>SPS for Linux が利用可能なクラウド環境については こちら を参照してください。<br>以下の仮想化環境は SPSuite for Linux がサポートする仮想化環境の一例です。<br><br>• KVM<br>• Oracle VM Server for x86<br>• VMware vSphere v6.5、v6.7、v7.0<br>• Amazon EC2<br>• Microsoft Azure<br>• Nutanix Acropolis Hypervisor<br>• Google Cloud<br><br>VMware vSphere v6.5 以降 では vSAN の利用をサポートします。ただし RDM は、vSAN で非サポートとなるため、利用できません。<br><br>KVM および Oracle VM Server for x86 仮想化環境上の SPS for Linux では、ファイバーチャネルSANおよびSCSIディスクを使用するクラスター構成をサポートしません。<br><br>**注意：** AWSやAzure環境で動作するSLESにおいて、クラウドネットワークプラグインによ |

| | |
|---|---|
| | って仮想 IP アドレスの動的な変更が行われることにより、LifeKeeper クラスターの動作に影響を与えることがあります。詳細や回避策は トラブルシューティング > 既知の問題と制限 >「LifeKeeper Core」をご確認ください。 |
| メモリー | システムの最小メモリー要件は、OS が必要とする最小要件を満たしてください。システムのメモリーは SPS が保護するシステム上で動作するアプリケーションに対してサイジングする必要があります。 |
| ディスク容量 | SPS for Linux の導入に必要なディスク容量は次のとおりです。<br>• /opt – 約 100MB (インストールするキットに依存します)<br>• / – 約 110MB |
| Java Runtime Environment | OpenJDK 1.8、10以降 |

## SPSのオプションリカバリーソフトウェアの要件

SPS が保護するアプリケーションに適用される追加の要件や制限については、アプリケーション設定 を参照してください。

| 製品 | 要件 |
|---|---|
| SPS Apache Web Server Recovery Kit | Apache Web Server v2.4 |
| SAP Recovery Kit | • SAP NetWeaver 7.0 、 Enhancement Package 1,2 および 3 を含む<br>• SAP NetWeaver 7.3 、 Enhancement Package 1 を含む<br>• SAP NetWeaver 7.4<br>• SAP NetWeaver 7.5<br>• SAP NetWeaver AS for ABAP 7.51 innovation package |
| SPS SAP MaxDB Recovery Kit | • SAP MaxDB v7.9<br>• SPS for Linux v6 以降 |
| SPS Postfix Recovery Kit | • Postfix ソフトウェアは、SPS for Linuxがサポートするディストリビューションに付属しているものを使用し、（クラスターを構成する）各サーバー上に構成します。Postfixのバージョンは各サーバーで同一である必要があります。<br>• SPS for Linux v6 以降 |
| SPS Oracle Recovery Kit | • Oracle Database Enterprise Edition v12c 、 v12c R2 、 v18c、 v19c (ASM 機能除く)<br>• Oracle Database Standard Edition 2 (SE2) v12c 、 v12c R2 、 v18c、 v19c (ASM 機能除く) |
| SPS DB2 Recovery Kit | • IBM Db2 Universal Database v10.5、v11.1<br>• IBM Db2 Enterprise Server Edition (ESE) v10.5 、 v11.1、v11.5<br>• IBM Db2 Workgroup Server Edition (WSE) v10.5 、 v11.1、v11.5 |

| | |
|---|---|
| | • IBM Db2 Express Edition v10.5 、 v11.1、v11.5<br><br>• SPS for Linux v6 以降<br>• SPS NFS Server Recovery Kit v5.1 以降 (マルチパーティションを持つ DB2 EEE 用および DB2 ESE 用のみ) |
| SPS MySQL Recovery Kit | • MySQL および MySQL Enterprise v5.7 、 v8.0<br>• MariaDB v10.3、v10.4、v10.5 |
| SPS PostgreSQL Recovery Kit | • PostgreSQL v9.6、v10、v11、v12、v13<br>• EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server v9.6、v10.0、v11.0、v12.0、v13.0<br>• PowerGres Plus (Linux 版) v10<br>• PowerGres on Linux v11<br><br>• FUJITSU Software Symfoware Serverの以下のエディション<br>◦ Symfoware Server V12.2<br>▪ Symfoware Server (Openインタフェース) V12.2 Enterprise Edition<br>▪ Symfoware Server (Openインタフェース) V12.2 Standard Edition<br>◦ Symfoware Server V12.3<br>▪ Symfoware Server (Postgres) V12.3 Enterprise Edition<br>▪ Symfoware Server (Postgres) V12.3 Standard Edition<br>▪ Symfoware Server (Postgres) V12.3 Lite Edition<br>◦ Symfoware Server V12.4<br>▪ Symfoware Server (Postgres) V12.4 Enterprise Edition<br>▪ Symfoware Server (Postgres) V12.4 Standard Edition<br>• FUJITSU Software Enterprise Postgres 以下のエディション<br>◦ FUJITSU Software Enterprise Postgres 9.5<br>▪ FUJITSU Software Enterprise Postgres 9.5 Advanced Edition<br>▪ FUJITSU Software Enterprise Postgres 9.5 Standard Edition<br>◦ FUJITSU Software Enterprise Postgres 9.6<br>▪ FUJITSU Software Enterprise Postgres 9.6 Standard Edition<br>◦ FUJITSU Software Enterprise Postgres 10の以下のエディション<br>▪ FUJITSU Software Enterprise Postgres 10 Advanced Edition<br>▪ FUJITSU Software Enterprise Postgres 10 Standard Edition<br>▪ FUJITSU Software Enterprise Postgres 10 Community Edition<br>◦ FUJITSU Software Enterprise Postgres 11<br>▪ FUJITSU Software Enterprise Postgres 11 Advanced Edition<br>▪ FUJITSU Software Enterprise Postgres 11 Standard Edition<br>◦ FUJITSU Software Enterprise Postgres 11 Community Edition<br>◦ FUJITSU Software Enterprise Postgres 12<br>▪ FUJITSU Software Enterprise Postgres 12 Advanced Edition<br>▪ FUJITSU Software Enterprise Postgres 12 Standard Edition<br>▪ FUJITSU Software Enterprise Postgres 12 Community Edition |
| SPS Sybase ASE Recovery Kit | Sybase ASE 15.7、16.0 |

| | |
|---|---|
| SPS Samba Recovery Kit | サポート対象の Linux ディストリビューションに付属の標準 samba ファイルサービス |
| SPS NFS Server Recovery Kit | • Linux kernel version 2.6 以降<br>• SLES を利用する場合、NFS サーバーパッケージおよびNFSクライアントパッケージを追加でインストールする必要があります。<br><br>注意：<br>Red Hat Enterprise Linux 7以降、CentOS 7以降、Oracle Linux 7以降では NFSv2 はサポートされません。<br>Red Hat Enterprise Linux 8以降では NFS over UDP はサポートされません。<br>一部の環境では追加の設定を必要とする場合があります。詳細は「特定の設定上の考慮事項」を参照してください。 |
| SPS Network Attached Storage Recovery Kit | NFS バージョン v2、v3、v4 のいずれかでのマウントをサポートする NFS サーバーまたは NAS デバイス<br><br>注意：<br>Red Hat Enterprise Linux 7以降、CentOS 7以降、Oracle Linux 7以降では NFSv2 はサポートされません。<br>Red Hat Enterprise Linux 8以降では NFS over UDP はサポートされません。<br>一部の環境では追加の設定を必要とする場合があります。詳細は「特定の設定上の考慮事項」を参照してください。 |
| SPS Logical Volume Manager (LVM) Recovery Kit | Linux Logical Volume Manager (LVM) バージョン 1 または 2 のボリュームグループおよび論理ボリューム |
| EMC PowerPath | • PowerPath for Linux v5.3 またはそれ以降<br>• sg3_utils パッケージがインストールされている必要があります。 |
| Device Mapper Multipath (DMMP) | • オペレーティングシステム付属の device-mapper-multipath パッケージがインストールされている必要があります<br>• sg3_utils パッケージがインストールされている必要があります |
| Hitachi Dynamic Link Manager Software (HDLM) | • 日立 HDLM のマルチパス I/O 設定を参照してください。<br>• sg3_utils パッケージがインストールされている必要があります |
| NEC iStorage Storage Path Savior (NECSPS) | • iStorage StoragePathSavior for Linux v3.3 以降<br>• 対応するLinux カーネルおよびディストリビューションは StoragePathSavior for Linux のサポート情報を参照してください<br>• Red Hat および SLES では sg3_utils パッケージがインストールされている必要があります<br>• SPS for Linux v6 以降 |
| WebSphere MQ リソース | • IBM MQ v8.0, v9.0, v9.1 および v9.2<br>• 既知の問題と制限 > インストール を参照してください。 |

| | |
|---|---|
| Quorum/Witness Package | quorum/witness モードのクラスターに参加するすべてのノード (witness 専用のノードを含む) には 、 Quorum/Witness Server Support Package for SPS をインストールする必要があります。 |

### ストレージとアダプターのオプション

共有ストレージ構成において SPS for Linux がサポートするストレージモデルやアダプターについては、「ストレージとアダプターのオプション」を参照してください。また、ドライバーのバージョンやその他の設定要件の詳細については、「ストレージとアダプターの設定」を参照してください。

## オープンソースソフトウェア

本バージョンの SPS for Linux のインストールイメージに同梱されるオープンソースソフトウェアとそのバージョン、ライセンスの一覧です。

| 名称 | バージョン | ライセンスのタイプとバージョン |
|---|---|---|
| curl-7.68.0-1 | 7.68.0 | MIT |
| libcurl-7.68.0-1 | 7.68.0 | MIT |
| gnutls-2.8.6-3.1 | 2.8.6 | GPLv3+ and GPLv2+ |
| gnutls-utils-2.8.6-3.1 | 2.8.6 | GPLv3+ |
| libgcrypt-1.5.0-2.1 | 1.5.0 | LGPv2+ |
| libgpg-error-1.10-2.1 | 1.1 | LGPv2+ |
| libvirt-0.9.3 | 0.9.3 | LGPLv2+ |
| libxml2-2.7.8-7.1 | 2.7.8 | MIT |
| libxml2-static-2.7.8-7.1 | 2.7.8 | MIT |
| lighttpd-1.4.41-1 | 1.4.41 | BSD |
| lighttpd-fastcgi-1.4.41-1 | 1.4.41 | BSD |
| openjdk-12.0.2 | 12.0.2 | GPLv2+ |
| java-1_8_0-openjdk-1.8.0.222-351.4 | 1.8.0 | GPLv2+ |
| java-1_8_0-openjdk-headless-1.8.0.222-351.4 | 1.8.0 | GPLv2+ |
| openssl-1.1.1g-1 | 1.1.1g | BSDish |
| openssl-perl-1.1.1g-1 | 1.1.1g | BSDish |
| pcre-4.5-2.1 | 4.5 | distributable |
| pdksh-5.2.14 | 5.2.14 | GPL, distributable |

| | | |
|---|---|---|
| perl-5.8.8-8.2 | 5.8.8 | Artistic License or GPL |
| perl-addons-5.8.8-24.1 | 5.8.8 | Various(GPL, artistic v2, BSD, Open Market, GPLv2+, MIT) |
| powercli-11.5.0-1 | 11.5.0 | Various |
| powershell-6.2.3-1 | 6.2.3 | MIT |
| readline-4.3-14.1 | 4.3 | GPL |
| runit-2.0.0-4.11 | 2.0.0 | BSD |
| util-linux-2.31.1-2 | 2.31.1 | GPLv2 and GPLv2+ and LGPLv2+ and BSD with advertising and Public Domain |
| xenserver-5.6.100 | 5.6.100 | LGPL |
| Perl Config::IniFiles (CPAN module) | 2.27 | GPL/Artistic (Same as Perl) |
| HADR-CentOS-2.6.32 | 2.6.32 | GPLv2 |
| HADR-CentOS-3.10.0 | 3.10.0 | GPLv2 |
| HADR-CentOS-4.18.0 | 4.18.0 | GPLv2 |
| HADR-RHAS-2.6.32 | 2.6.32 | GPLv2 |
| HADR-RHAS-3.10.0 | 3.10.0 | GPLv2 |
| HADR-RHAS-4.18.0 | 4.18.0 | GPLv2 |
| HADR-OEL-2.6.32 | 2.6.32 | GPLv2 |
| HADR-OEL-3.10.0 | 3.10.0 | GPLv2 |
| HADR-OEL-4.18.0 | 4.18.0 | GPLv2 |
| HADR-OEL.UEK-4.14.35 | 4.14.35 | GPLv2 |
| HADR-SuSE-4.12.14 | 4.12.14 | GPLv2 |
| HADR-SuSE-5.3.18 | 5.3.18 | GPLv2 |

## クライアントのプラットフォームとブラウザー

SPS web クライアントは 、 Java Runtime 環境 JRE 8 update 51 をサポートするすべてのプラットフォームで動作します。現在動作が確認されている環境は、 JRE 8 update 51 を使用した Linux、Windows Server 2008 R2 、 Windows Server 2012 R2 、 Windows Server 2016、 Windows 7、 Windows8、 Windows 10 上のFirefox (Firefox 51 まで) 、 および Internet Explorer です。 その他の最近のプラットフォームやブラウザーも 、 SPS web クライアントが動作する可能性がありますが、 それらの環境はテストされていません。また 、 各ブラウザ固有の機能についても 、 テストされていません。
クライアントマシンのローカル host ファイル（通常は */etc/hosts* または *C:\windows\system32\drivers\etc\hosts* にあります）には、クラスターを構成するすべてのサーバーのホスト名とIPアドレスを設定する必要があります。クライアントのサーバーへの接続時間を最小にするとともに、 DNS 停止時であっても

サーバーへの接続を可能にします。

## インストールおよび構成

インストールおよび構成の詳しい情報については 、 SIOS Protection Suite インストレーションガイド を参照してください。

## アップグレード

SPS for Linux v9.5.x へは、は、v9.3.x および v9.4.x からアップグレードすることができます。これらより前のバージョンからアップグレードする場合、それまでのバージョンをアンインストールし、SPS for Linux を再インストールする必要があります。それまでのバージョンをアンインストールしないで v9.5.x へアップグレードしたい場合には、一度 v9.3.x または v9.4.x にアップグレードした後、さらに v9.5.x にアップグレードする方法をとることもできます。

## テクニカルノート

SPS 環境に関連した構成および動作上の問題に関して、テクニカルノート の参照を強く推奨します。

## 既知の問題

「既知の問題と制限 」と「DataKeeper トラブルシューティング 」をそれぞれ参照してください。

商標について：「Amazon Web Services」、「Powered by Amazon Web Services」のロゴ、「AWS」、「Amazon EC2」、「EC2」、「Amazon Elastic Compute Cloud」、「Amazon Route 53」、「Amazon Virtual Private Cloud」および「Amazon VPC」は、米国その他の国における Amazon.com, Inc. またはその関連会社の商標です。なお、本文および図表中では、「™」、「®」は明記しておりません。

# 重要なお知らせ: SIOS DataKeeper for Linux とカーネルの問題

**重要なお知らせ:** 本事象に関するパッチは現在入手可能です。対象となるお客様は速やかにパッチを適用してください。
v9.5.2ではパッチが適用された raid1 カーネルモジュールが配布され、v9.5.2 のインストールまたはアップグレード時に自動的にインストールされます。

以下のカーネルおよびバージョンにおいて影響を受ける事象は、全ての開発、品質保証作業を終えたことをご案内いたします。

| OS | 影響を受けるカーネルの範囲 | 左記OSをサポートしている SIOS Protection Suite for Linux(LifeKeeper/DtaKeeper) のバージョン |
|---|---|---|
| RHEL/CentOS/ OEL 8.2 | 全て | v9.5.1 |
| RHEL 8.3 | 全て | v9.5.1 |
| SLES 12 SP4 | 4.12.14-95.51 以上 | v9.3.2 ～ v9.5.1 |
| SLES 12 SP5 | 4.12.14-122.20 ～ 4.12.14-122.74.0 | v9.5.0 ～ v9.5.1 |
| SLES 15 SP1 | 4.12.14-197.37.1 以上 | v9.4.1 ～ v9.5.1 |
| SLES 15 SP2 | 5.3.18-22.2 ～ 5.3.18-24.67.1 | v9.5.1 |
| OEL 7.x UEK 5 | 4.14.35-2025.400.8 ～ 4.14.35-2047.504.1 | v9.4.0 ～ v9.5.1 |

上記のオペレーティングシステム上で SIOS Protection Suite for Linux v9.5.1以前バージョンと DataKeeper を起動させているお客様は速やかに弊社LifeKeeper / DataKeeper ユーザーポータルの下記 URL をご参照の上、パッチをダウンロードしてください。

LifeKeeper + DataKeeper構成のシステムにおける データ破損問題の対策についてのご案内

このパッチを実行しているシステム上で SIOS Protection Suite for Linuxをアップグレードするための必要な手順については、パッチダウンロード時に入手可能な readme 参照してください。

パッチ適用後は、カーネルをパッチが適用されたraid1 kernel moduleを含まない別のバージョンにアップデートすることをお勧めしません。カーネルの自動アップデートを制限する方法については、ご使用のOSのマニュアルを参照してください。カーネルアップデートする際の注意点に関しては、以下のページも併せて参照ください。

カーネルアップデートする際の注意点

最終的な解決策については、引き続き情報を提供してまいります。

パッチ適用後 (セキュリティ問題やレギュレーションの理由で) カーネルのアップグレードが必要な場合は、パッチ申請時にダウンロード可能な readme ファイルに記載されている「パッチ適用後の計画されたカーネルのアップグレード」のセクションを参照してください。

ご不明な点がございましたら、SIOSテクニカルサポートまたはSIOS営業担当者までお問い合わせください。

# 3. LifeKeeper for Linux スタートアップガイド

SIOS Protection Suite for Linux、および Single Server Protection for Linux のスタートアップガイドにつきましては、LifeKeeper/DataKeeper User Portal の記事 をご参照ください。

# 4. SIOS Protection Suite インストレーションガイド

「SIOS Protection Suite (SPS) インストレーションガイド」には、SPS 環境をプランニングおよびインストールする方法が記載されています。サーバ、ストレージデバイス、ネットワークコンポーネントをセットアップするために必要な手順の他、LifeKeeper のグラフィカルユーザインターフェース（GUI）の詳しい設定についても示されます。

このガイドの手順をすべて終えれば、LifeKeeper および DataKeeper リソースを設定できる状態になります。SPS を設定するために必要な情報は、SPS for Linux テクニカルドキュメンテーション で提供されています。

## システム要件

ハードウェアおよびソフトウェアの要件およびバージョンに関する総合的なリストについては、SPS for Linux リリースノート を参照してください。

また、SPS をインストールする前に、本書に記載されているプランニング作業およびハードウェア構成作業を完了していることを確認してください。

## テクニカルノート

本ソフトウェアに関する問題のトラブルシューティングや制限事項などの詳細については、SPS for Linux テクニカルドキュメンテーションのテクニカルノート およびトラブルシューティング セクションを参照してください。

# 4.1. SIOS Protection Suite ソフトウェアのパッケージ

SIOS Protection Suite (SPS) for Linux ソフトウェア (Optional SPS Recovery Kit を含む) は、単一のイメージファイル (sps.img) に格納されています。

## SPS for Linux のインストールイメージファイル

SPS for Linux のイメージファイル (sps.img) では、システムにSPSをインストールする際に必要なユーザー対話型システムセットアップタスクを実行するように設計された、一連のインストールスクリプトを提供しています（対話モード を参照してください）。 非ユーザー対話型インストールを実行することもできます。非ユーザー対話型インストールについての詳細は、非対話モード を参照してください。
SPSのインストールプロセスは、次の3つのステップに分けられます。

- 情報収集
- 選択（ユーザー対話型のみ）
- インストールと設定

このプロセスの最初のステップは情報収集で、使用中のLinuxディストリビューションなどのシステムに関する情報を収集して、システムがインストールに必要な要件を満たしていることを確認します。2番目のステップは選択で、メニューベースの選択プロセスを通じてユーザーと対話し、インストールするSPSパッケージとそのサポートに必要な構成を決定します。最後のステップは実際のインストールと設定で、SPS Coreパッケージクラスタとオプションのリカバリーソフトウェアのインストール、およびSPS用のシステムの設定を実施します。このステップでは、システムにまだインストールされていない、必要なOSサポートパッケージもインストールされます。

SPS for Linux のイメージファイルには、次のソフトウェアパッケージを含む Core パッケージクラスタが格納されています。

## SPS Core パッケージクラスタ

- LifeKeeper (**steeleye-lk**)。LifeKeeper Core パッケージには、メモリ、CPU、OS、SCSI ディスクサブシステム、ファイルシステムなどの中核システムコンポーネント用のリカバリソフトウェアがあります。
- LifeKeeper GUI (**steeleye-lkGUI**)。LifeKeeper GUI パッケージは、LifeKeeper の管理および監視用のグラフィカルユーザインターフェースです。
- DataKeeper (**steeleye-lkDR**)。DataKeeper パッケージは、インテントログ記録を使用するデータ複製 (同期ミラーまたは非同期ミラー) を実行します。
- IP Recovery Kit (**steeleye-lkIP**)。LifeKeeper IP Recovery Kit は、IP アドレスの自動リカバリ用のスイッチオーバソフトウェアです。

- Raw I/O Recovery Kit (**steeleye-lkRAW**)。LifeKeeper Raw I/O Recovery Kit は、ロー I/O を使用してカーネルのバッファリングを迂回するアプリケーションをサポートします。

- マニュアルページ (**steeleye-lkMAN**)。LifeKeeper マニュアルページパッケージは、LifeKeeper 製品のリファレンスマニュアルです。

## オプションのリカバリソフトウェア

また、Recovery Kit は、SPS Core ソフトウェアとともにリリースされています。インストール中、提供されている最新のリカバリキットを選択可能な全リストが表示されます。リカバリキットの詳細については、SPS テクニカルドキュメンテーションのアプリケーション Recovery Kit セクションを参照してください。

# 4.2. SPS 環境のプランニング

次のトピックは、SPS for Linux のクラスタ環境の定義に役立ちます。

---

サーバ構成のマッピング

ストレージとアダプタの要件

ストレージとアダプタのオプション

# 4.2.1. サーバ構成のマッピング

以下のガイドラインを使用して、サーバ構成を文書化してください。

1. 使用する構成に対して、サーバ名、プロセッサの種類、メモリ、およびその他の I/O デバイスを決定してください。バックアップサーバを指定した場合には、プライマリサーバに障害が発生したときに、選択したサーバに処理を実行する能力があることを確認する必要があります。

2. 通信接続要件を決定してください。

   **重要**:クラスタ化された構成には、可能性として、2 種類の通信要件 (クラスタ要件とユーザ要件) があります。

   - クラスタ - LifeKeeper クラスタでは、サーバ間に少なくとも 2 つのコミュニケーションパス (「ハートビート」とも呼ばれます) が必要になります。この冗長性により、通信障害が原因で発生する「スプリットブレイン」シナリオを回避することができます。独立した 2 つのサブネットを使用する 2 つの分離した LAN ベースの (TCP) コミュニケーションパスが推奨され、これらの 1 つ以上をプライベートネットワークとして構成する必要があります。TCP と TTY のの組み合わせもサポートされています。TTY コミュニケーションパスは、サーバのシリアルポート間で RS-232 ヌルモデム通信を使用します。

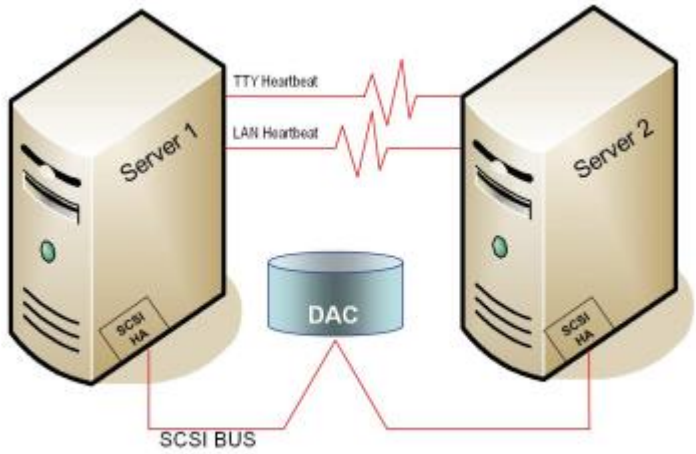
     コミュニケーションパスを 1 つしか使用しない場合、互いに通信する LifeKeeper クラスタ内のシステムの機能に支障をきたす可能性があります。単一のコミュニケーションパスを使用しているときに、そのコミュニケーションパスで障害が発生した場合、複数のシステム上で同時に LifeKeeper の階層が使用可能になることがあります。これは、偽のフェイルオーバまたは「スプリットブレイン」シナリオと呼ばれます。「スプリットブレイン」シナリオでは、各サーバが、アプリケーションを制御できると認識しているため、データにアクセスしようとしたり共有ストレージデバイスにデータを書き込もうとする場合があります。スプリットブレインシナリオを解決するために、LifeKeeper では、サーバの電源をオフにしたり、再起動したり、階層を使用できなくすることで、すべての共有データに対するデータの整合性を保証することができます。また、TCP コミュニケーションパス上のネットワークトラフィックが大きくなると、偽のフェイルオーバや LifeKeeper が適切に初期化できなくなるなど、予期せぬ動作が生じる可能性があります。

   - ユーザ - ユーザトラフィックに対する代替の LAN 接続、つまり、クラスタハートビートに使用するものとは別の LAN 接続を用意することをお勧めします。ただし、(推奨通りに) 2 つの TCP コミュニケーションパスを構成した場合、これらのコミュニケーションパスのいずれかが、サーバに出入りするその他のトラフィックとネットワークアドレスを共有することができます。
   - 注記: リソースが必要なときのみに in service になるようにするには、Quorum/Witness Server Support Package for LifeKeeper を使用することができます。

3. 共有リソースアクセス要件を確認して理解してください。共有ストレージを使用するクラスタは、共有 SCSI バスまたはファイバチャネルループを利用できます。 LifeKeeper ではリソースが 1 つのサーバにロックされるため、ロックされたすべてのリソースへのアクセスが必要になるサーバは常時 1 つだけであることを確認する必要があります。LifeKeeper デバイスのロックは、論理ユニット (LUN) レベルで行われます。アクティブ/アクティブ構成では、各階層が独自の一意の LUN にアクセスする必要があります。共通の LUN にアクセスするすべての階層は、同じサーバ上でアクティブ (稼働中) である必要があります。

4. 共有メモリ要件を決定してください。共有メモリおよびセマフォパラメータを設定する場合は、LifeKeeper だけでなくサードパーティ製アプリケーションの共有メモリ要件も考慮に入れてください。LifeKeeper の共有メモリ要件については、 テクニカルノート のチューニング を参照してください。

# LifeKeeper ペアに対する構成マップの例




この構成マップの例は、ディスクアレイサブシステムを共有する LifeKeeper サーバのペアを図示しています。通常は、Server 1 がアプリケーションを実行し、Server 2 がバックアップサーバまたはセカンダリサーバになります。このケースでは、同時に 1 つのサーバがディスクアレイのディスクストレージスペース全体を保有しているので、ディスクリソースの競合はありません。ディスクアレイコントローラは「DAC」、SCSI ホストアダプタ (パラレル SCSI、ファイバチャネルなど) は「SCSI HA」と表記されています。

サーバのペアが、最も単純な LifeKeeper 構成となります。3 つ以上のサーバで構成されるクラスタを計画する場合、複数のサーバ間が適切に接続されるようにマッピングすることが非常に重要になります。たとえば、多方向フェイルオーバ構成では、物理的な接続が存在しない場合でも LifeKeeper 内のコミュニケーションパスを定義することが可能です。カスケーディングフェイルオーバ機能を実現するために、各サーバがクラスタ内の他のすべてのサーバへの物理的な接続パスを持つ必要があります。

# 4.2.2. ストレージとアダプタの要件

以下のガイドラインを使用して、ストレージとホストアダプタの要件を決定してください。

**ストレージデバイス** - アプリケーションのデータストレージ要件に基づいて、構成に必要なデータストレージデバイスの種類と数を決定する必要があります。共有ファイルは、ディスクアレイサブシステム (RAID : Redundant Array of Inexpensive Disks) 上に置く必要があります。LifeKeeper は、構成に使用できるハードウェア RAID 周辺装置を多数サポートしています。サポートされている周辺装置のリストについては、ストレージとアダプタのオプションを参照してください。

ストレージデバイスの構成を計画する際には、以下の点を考慮してください。

- LifeKeeper では物理ディスクまたは論理ユニット (LUN) レベルでリソースを管理し、その構成内では同時に 1 つのサーバのみが各物理ディスクまたは LUN 上のリソースを利用できます。そのため、LifeKeeper の構成を始める前に、ディスク割り当ての計画を立てることをお勧めします。たとえば、アクティブ/アクティブ構成の各階層は、独自の一意の LUN にアクセスする必要があるので、2 ノードアクティブ/アクティブ構成の場合は最低 2 つの LUN が必要になります。

- 一部のモデル固有の問題およびハードウェア設定の詳細は、サポートストレージ一覧 で管理されています。

**アダプタ** - 構成の種類および周辺装置の数に基づいて、必要な SCSI またはファイバチャネルホストアダプタの種類と数を決定してください。 選択するアダプタは、ドライバが使用できるように、LifeKeeper だけでなく使用している Linux ディストリビューションでもサポートされていることが重要です。サポートされているホストアダプタのリストについては、サポートされているアダプタモデルを参照してください。 参照用に、構成マップにホストアダプタを追加する必要があります。

# 4.2.3. ストレージとアダプタのオプション

共有ストレージ構成で LifeKeeper が現在サポートしているディスクアレイのストレージモデルの一覧については、サポートストレージ一覧 を参照してください。ドライバのバージョンと、これらのアレイとアダプタに関するその他の設定要件については、ストレージとアダプタの設定 を参照してください。

IP フェイルオーバのみを使用する非共有ストレージを含む LifeKeeper 構成あるいは SIOS Data Replication または Network Attached Storage の使用時には、サポートされているディスクアレイおよびアダプタは必要ありません。

SIOS Technology Corp. では、ファイバチャネルのハブとスイッチを特に認定していません。これは、これらのデバイスに対する LifeKeeper 固有の既知の制限事項や要件がないためです。ストレージとアダプタの設定 の指定されたアレイで特に記載のない限り、LifeKeeper は、ディスクアレイベンダがサポートするハブとスイッチを推奨します。

# 4.3. SPS 環境のセットアップ

これまでのセクションでは、要件を決定し、LifeKeeper 設定をマッピングしました。これで、SPS 環境のコンポーネントをセットアップできるようになりました。

**注記:** 一部のセットアップ作業は異なる順序で実行することが可能ですが、このリストは推奨された順序で示されています。

---

Linux OS および関連する通信パッケージのインストール

Linuxの依存関係

サーバと共有ストレージの接続

共有ストレージの設定

ネットワーク設定の確認

切り替え可能な IP アドレスの作成

データベースアプリケーションのインストールとセットアップ

GUI ユーザの設定

# 4.3.1. Linux OS および関連する通信パッケージのインストール

SPS for Linux ソフトウェアをインストールする前にLinux オペレーティングシステムが正常にインストールされ動作可能であることを確認する必要があります。完全なインストールの詳細については、Linux のディストリビューションに付属の Linux インストール手順書を参照してください。

注記:

- 必須パッケージに必要となる場合がある依存関係については、Linux の依存関係 トピックを参照してください。
- 共有ストレージを接続して設定した後にLinux をインストールすることも可能ですが、 新しい周辺装置を導入する前に、Linux をインストールして実行する方が簡単な場合があります。
- SPS for Linux のインストールイメージファイルは、SPS をシステムにインストールするときに必要なユーザインタラクティブなシステムセットアップ作業とインストール作業を実行するよう設計されたインストールスクリプト一式を提供します。

# 4.3.2. Linux の依存関係

SPS for Linux を正常にインストールするには、いくつかの前提条件パッケージをインストールする必要があります。スクリプトが失敗しないように、これらのパッケージはインストールセットアップスクリプトを実行する前にインストールしてください。

前提条件パッケージは以下の 3 つのグループに分類されます。

- 一般的なパッケージの依存関係
- オプションのリカバリキットパッケージの依存関係

オペレーティングシステムのバージョン と、選択したオペレーティングシステムの種類 (最小、デフォルトなど) に基づいてインストールされるパッケージによっては、追加の従属パッケージが必要な場合があります。

依存関係は LifeKeeper がサポートする OS のバージョンが基本となります。詳細については SIOS Protection Suite for Linux サポートマトリックス を参照してください。

注記: **yum** や **zypper** など、リポジトリーベースのパッケージマネージャーの使用を検討することを推奨します。これらのパッケージマネージャーは、定義済みのソフトウェアリポジトリーを検索して依存関係を自動的に解決するように設計されているので、これらの必須パッケージのインストールが容易になります。依存するパッケージのインストールを行いやすくするため、SPS for Linux のインストーラーは、SPS for Linux のパッケージをインストールに **yum** もしくは **zypper** を使用します。 そのため、OS パッケージのリポジトリーをセットアップしておくことを強く推奨します。それにより SPS for Linux をインストールする前に下記に記載の OS に依存するパッケージをインストールする必要性がなくなります。

**rpm** のインストール例

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

**yum** のインストール例

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

**Zypper** のインストール例

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

## yum/Zypper パッケージリスト

列挙されたディストリビューションごとに、以下のリストの rpm パッケージが対応するパッケージインストーラーによってインストールされます。これらは、SIOS Protection Suite for Linux に必要なすべての依存関係を解決するパッケージの最小リストです。

**Red Hat Enterprise Linux、CentOS、Oracle Linux**

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

**SLES**

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

## 一般的なパッケージの依存関係

SPS for Linux を正常にインストールするには、以下のパッケージが必要です。インストールするパッケージのアーキテクチャーバージョンは、必ずオペレーティングシステムのアーキテクチャーと一致させてください (x86 または x86_64)。

**Red Hat Enterprise Linux、CentOS、Oracle Linux**

- bzip2
- iproute
- iputils
- patch (バージョン 2.5 以降)
- redhat-lsb
- ncurses-libs

**注記:** オペレーティングシステムのインストール時の選択によっては、これらのパッケージの一部またはすべてが既にインストールされている場合があります。

**SLES**

- bzip2
- iproute2
- iptables
- iputils
- insserv
- patch (バージョン 2.5 以降)
- lsb
- libncurses5
- libXtst6 (SLES15のみ)
- libXi6 (SLES15のみ)

**注記:** オペレーティングシステムのインストール時の選択によっては、これらのパッケージの一部またはすべてが既にインストールされている場合があります。

## syslogデーモンとの依存関係

LifeKeeper のログは syslog デーモンを利用して記録されます。LifeKeeper では、rsyslog をサポートしています。LifeKeeper をインストールする前に、rsyslog デーモンをインストールして動作させておく必要

があります。

**注記:** RHEL7 や SLES12 など systemd を採用しているディストリビューションでは、journald がログを一括して管理しています。LifeKeeper は syslog デーモンを利用してログ出力を行っていますので、syslog デーモンも動作させる必要があります。したがって、LifeKeeper を使用する場合は syslog デーモンを動作させるように設定してください。

**注記:** journald はデフォルトで /run/log/journal の tmpfs にログを記録します。したがって、OS シャットダウン時のシステムログなどが記録されません。journald のログが永続化する様に設定を変更してください。 journald のログを永続化するためには、/etc/systemd/journald.conf で「Storage=persistent」と設定するか、「Storage=auto」(デフォルト) の設定で /var/log/journal ディレクトリーを作成してください。設定変更後、systemd-journald.service を再起動してください。

## オプションのリカバリキットパッケージの依存関係

さらに、SIOS Protection Suite for Linux のオプションの Application Recovery Kit (ARK) の一部はサポートパッケージをインストールする必要があります。

SIOS Protection Suite for Linux NFS Application Recovery Kit を用いて NFS エクスポートを保護する場合は、以下の従属パッケージが必要です。

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

Device Mapper Multipath (DMMP)、Hitachi Dynamic Link Manager Software (HDLM)、Power Path、または NEC iStorage StoragePathSavior (NECSPS) を用いてマルチパスデバイスを保護する場合は、以下の従属パッケージが必要です。

- sg3_utils (すべてのマルチパスキット)
- sg3_utils-libs (すべてのマルチパスキット)
- HDLM (Hitachi Dynamic Link Manager Software Kit)
- EMCpower.LINUX (Power Path Kit)
- sps (NEC iStorage StoragePathSavior Kit **4.2.0** 以前)
- sps-utils および sps-driver (NEC iStorage StoragePathSavior Kit **4.2.1** 以降)

SIOS Protection Suite for Linux Websphere MQ/MQSeries Application Recovery Kit を用いて Websphere MQSeries キューマネージャーを保護する場合は、以下の従属 Websphere MQ パッケージが必要です。

- MQSeriesServer
- MQSeriesSamples
- MQSeriesClient
- MQSeriesRuntime
- MQSeriesSDK

SIOS Protection Suite for Linux Software RAID (md) Recovery Kit を用いて Software RAID デバイスを保

護する場合は、以下の従属パッケージが必要です。

- mdadm

# 4.3.3. サーバと共有ストレージの接続

非共有ストレージ環境で LifeKeeper を使用することを計画している場合は、この情報をスキップできます。データレプリケーション (ミラーリング) 環境で LifeKeeper を使用している場合は、この文書の DataKeeper セクションを参照してください。ネットワーク接続ストレージ環境で LifeKeeper を使用している場合は、LifeKeeper Network Attached Storage Recovery Kit 管理ガイド を参照してください。

Linux がインストールされたら、ホストアダプタおよび共有周辺装置のアドレスを設定する必要があります。具体的な詳細については、アダプタおよびストレージデバイスに付属のドキュメンテーションを参照してください。

# 4.3.4. 共有ストレージの設定

LifeKeeper では、共有 SCSI (Small Computer System Interface) ホストアダプタおよび共有ディスクハードウェアの機能を使用して、障害が発生したサーバから指定のバックアップサーバにリソースを切り替えるように設定できます。ファイバチャネルのストレージエリアネットワーク (SAN) も、障害の発生したサーバから指定のバックアップサーバにリソースを切り替えるのに使用できます。

以下の作業を実行して、ディスクベースのアプリケーションリソース階層を作成し、LifeKeeper でフェイルオーバ保護を提供できるようにしてください。

1. ディスクおよび LUN のパーティションを分割してください。LifeKeeper の保護下にあるすべてのディスクのパーティションを分割する必要があるので、共有ディスクアレイは論理ユニット (LUN) に設定する必要があります。ディスクアレイ管理ソフトウェアを使用して、この設定を実行してください。詳細な手順については、ディスクアレイソフトウェアのマニュアルを参照してください。

   注記: LifeKeeper では LUN レベルで **ディスク** をロックすることに注意してください。したがって、アクティブ/スタンバイ設定では 1 つの LUN が適切と考えられます。ただし、アクティブ/アクティブ設定を使用する場合は、少なくとも 2 つの別個の LUN を設定して、各階層が **独自の一意の** LUN にアクセスできるようにする必要があります。

2. 両方のサーバが共有ディスクを認識することを確認してください ( **gdisk** コマンドなどを使用)。作成した LUN を Linux が認識しない場合は、LifeKeeper も認識しません。

3. LifeKeeper 階層でプライマリサーバとして使用するシステムから共有ディスク上のファイルシステムを作成してください。ファイルシステムの管理に関する完全な手順については、Linux のマニュアルを参照してください。

# 4.3.5. ネットワーク設定の確認

LifeKeeper をインストールする *前に* 、ネットワークが適切に設定され動作することを確認することが重要です。ネットワークの動作を確認するために、この時点で実行する必要のある作業を以下に示します。

1. サーバインストールでファイアウォールが有効の場合、LifeKeeper のポートを調整するか、ファイアウォールを無効にする必要があります。ファイアウォール下で LifeKeeper を実行する 。

2. 各サーバから、ローカルサーバを ping し、クラスタ内の他のサーバを ping してください。ping が失敗した場合には、必要なトラブルシューティングを行い、修正処置を実行してください。

3. サーバに複数のネットワークアダプタがある場合は、アダプタが異なるサブネット上にあるように設定する必要があります。アダプタが同じサブネット上にある場合、TCP/IP は 2 つ目のアダプタを有効に利用できません。

4. *localhost* がクラスタ内の各サーバで解決可能であることを確認してください。DNS が実装されていない場合は、 */etc/hosts* ファイルを編集して、 *localhost* 名のエントリを追加してください。このエントリは、ローカルサーバの IP アドレスをリストしたり、デフォルトのエントリ (127.0.0.1) をリストすることができます。 *localhost* が解決できない場合、LifeKeeper GUI が正常に機能しない可能性があります。

5. DNS が実装されている場合は、LifeKeeper クラスタ内のサーバが DNS を使用して解決できるように設定されていることを確認してください。

6. 各サーバのホスト名が正しく、LifeKeeper をインストールした後に変更されないことを確認してください。後で LifeKeeper システムのホスト名を変更するように決定した場合は、 *クラスタ内のすべてのサーバについて* 以下の手順に従う必要があります。

   a. 次のコマンドを使用して、クラスタ内のすべてのサーバ上の LifeKeeper を停止してください。

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

   b. Linux **hostname** コマンドを使用して、サーバのホスト名を変更してください。

   c. 続行する前に、新しいホスト名がクラスタ内の各サーバで解決可能であることを確認する必要があります (前の項目を参照)。

   d. クラスタ内の各サーバで次のコマンドを実行して、LifeKeeper のホスト名を更新してください。(詳細については、lk_chg_value(1M) を参照してください。)

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

   e. 次のコマンドを使用して LifeKeeper を起動してください。

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

LifeKeeper for Linux v7.x では、コミュニケーションパスおよび IP リソース用の VLAN インターフェース

がサポートされています。VLAN インターフェースの種類は、以下に示すように選択することができます。

# VLAN インターフェースサポートマトリックス

- サポートなし \ x サポートあり

**LK Linux v7.1 以前のバージョン**

| VLAN_NAME_TYPE | コミュニケーションパス | IP リソース |
|---|---|---|
| DEV_PLUS_VID (eth0.0100) | - | x |
| DEV_PLUS_VID_NO_PAD (eth0.100) | - | x |
| VLAN_PLUS_VID (vlan0100) | x | x |
| VLAN_PLUS_VID_NO_PAD (vlan100) | x | x |

**LK Linux v7.2 以降のバージョン**

| VLAN_NAME_TYPE | コミュニケーションパス | IP リソース |
|---|---|---|
| DEV_PLUS_VID (eth0.0100) | x | x |
| DEV_PLUS_VID_NO_PAD (eth0.100) | x | x |
| VLAN_PLUS_VID (vlan0100) | x | x |
| VLAN_PLUS_VID_NO_PAD (vlan100) | x | x |

# 4.3.6. 切り替え可能な IP アドレスの作成

切り替え可能な IP アドレスとは、サーバ間で切り替えることができる「仮想的な」IP アドレスです。各サーバのネットワークインターフェースカードに関連付けられた IP アドレスとは別個のものです。LifeKeeper の保護下にあるアプリケーションは、切り替え可能な IP アドレスに関連付けられています。これにより、プライマリサーバで障害が発生した場合、その IP アドレスがバックアップサーバに「切り替わり」ます。

切り替え可能な IP アドレスに対してリソース階層を設定することを計画している場合は、クラスタの各サーバで以下の作業を実行する必要があります。

- コンピュータ名が正しく、変更されないことを確認する。
- ping コマンドを使用して切り替え可能な IP アドレスが一意であることを確認する。
- */etc/hosts* ファイルを編集して、切り替え可能な IP アドレスごとにエントリを追加する。

詳細については、LifeKeeper for Linux IP Recovery Kit テクニカルドキュメンテーション を参照してください。

# 4.3.7. データベースアプリケーションのインストールとセットアップ

お使いの環境に Oracle、DB2、MySQL などの保護されたデータベースアプリケーションが含まれている場合は、データベースに付属のドキュメンテーションを使用してアプリケーションをインストールする必要があります。データベースが共有ファイルシステム上にあり、設定ファイルが共有ファイルシステム上にあることを確認してください。実行可能ファイルは、ローカルまたは共有のファイルシステム上に置くことができます。

LifeKeeper をインストールした *後に* アプリケーションをインストールすることも可能ですが、LifeKeeper の保護下に置く前に、適切に設定され稼動することを確認するためにアプリケーションをテストする必要があります。その他のインストールおよびセットアップの特定の考慮事項については、LifeKeeper データベースリカバリキットドキュメンテーション を参照してください。

# 4.3.8. GUI ユーザの設定

## PAM による GUI の認証

SPS for Linux は、Linux Standard Base (LSB) で提供されている Pluggable Authentication Module (PAM) を利用するようになりました。SPS では、以前 /opt/LifeKeeper/website/passwd に置かれていた個別のパスワードファイルを現在は使用していません。それに代わり、ユーザは、システムの PAM 設定に対して識別および認証されます。権限のレベルは、PAM が提供するグループメンバシップから決定されます。

GUI にアクセスするには、ユーザは LifeKeeper の次の 3 つのグループのいずれかのメンバーである必要があります: lkadmin、lkoper または lkguest。これらのグループのメンバシップは、クラスタ全体で使用されているユーザアカウントデータベースのタイプに応じた適切な方法により、システム管理者が設定してください。

LifeKeeper のこれら 3 つのグループには、それぞれの権限セットが用意されています (権限テーブル を参照)。

1. クラスタ全体にわたって ***Administrator (管理者)*** の権限を持つユーザ (`lkadmin`) は、GUI から可能な動作のすべてを実行できます。
2. 1 台のサーバ上で ***Operator (オペレータ)*** の権限を持つユーザ (`lkoper`) は、LifeKeeper の設定やステータスの情報を表示でき、そのサーバ上のリソースを in service や out of service にできます。
3. 1 台のサーバ上で ***Guest (ゲスト)*** の権限を持つユーザ (`lkguest`) は、そのサーバの LifeKeeper の設定やステータスの情報を表示できます。

GUI パッケージのインストール時に、そのシステムの *root* ユーザはシステムのローカルグループデータベースの `lkadmin` グループに自動的に追加されるので、 *root* は、そのサーバから GUI アプリケーションまたは Web クライアント経由で LifeKeeper のすべての作業を実行できます。 *root* 以外のユーザに LifeKeeper GUI クライアントの使用を許可するように計画している場合は、ユーザを適切なグループに追加することにより LifeKeeper GUI のユーザを設定する必要があります。

NIS、LDAP、AD などの非ローカルデータベースを使用するように PAM を設定している場合は、それらのデータベースでアカウントが正しく設定されていることをシステム管理者は確認する必要があります。上記のグループが存在していなければなりません。また、LifeKeeper GUI へのログインが許可されているユーザがそれらのグループの 1 つに属するメンバでなければなりません。これらのグループは、リモートデータベース内のみに作成する必要があり、ローカルの `/etc/group` ファイルからは削除する必要があります。

クラスタ内のシステムで異なる「root」パスワードを使用している場合、LK GUI のログインは失敗します。クラスタ内の各システムの root パスワードを同一にすれば、LK GUIに「root」でログインできます。

> ✻ 注記 : 混乱を避け整合性を保つため、LDAP、NIS、AD などのより複雑な PAM 設定を利用する場合は、SPS のインストールまたはアップグレードを実行する前に、すべてのユーザアカウントおよび LifeKeeper グループアカウントを準備しておいてください。

最良の方法は、常にクラスタ全体を単位として許可を付与することです。サーバ単位で許可を付与することもできますが、ユーザを混乱させてしまい、管理作業が実行できなくなります。

# 4.3.9. ライセンスの取得とインストール

SPS for Linux では、サーバーごとに別々のライセンスが必要です。ライセンスは、ランタイムライセンスです。つまり、SPS のインストールはライセンスなしでも可能ですが、正常に製品を起動して実行するためには、事前にライセンスをインストールする必要があります。

https://fast.wistia.net/embed/iframe/76bo8epo9g

参考動画 ライセンスのアクティベートについて（英語）

https://fast.wistia.net/embed/iframe/mtl4wubwfk

参考動画 ライセンスのインストールについて（英語）

インストールスクリプトによってインストールされるライセンスユーティリティーパッケージは、SIOS Protection Suite ソフトウェアの初期インストール時にサーバーの使用可能なすべての Host ID を取得して表示します。ライセンスがインストールされると、このユーティリティーは Entitlement ID (使用可能な場合) または Host ID (使用できない場合) を返します。

> 注記: Host ID が表示される場合は常に NIC の MAC アドレスに基づいています。

リリース 8.2.0 以降に SIOS Technology Corp. ライセンス管理ポータル から取得した新しいライセンスには Entitlement ID が含まれ、クラスター内の特定のノードにロックされることはありません。8.2.0 より前にライセンスを取得した既存ユーザーは、以下の既存ユーザー向けのライセンスリホストセクションを参照してください。SIOS Protection Suite ソフトウェアと一緒に提供された Entitlement ID (認証コード) は、SIOS Protection Suite ソフトウェアを実行するために必要なパーマネントライセンスを取得するために使用されます。このプロセスを以下の図に示します。

> 注記: ソフトウェアパッケージごとに、サーバーごとのライセンスが必要になります。

各サーバーのライセンスを取得してインストールするには、次の手順を実行します。

1. **Host ID** を取得します。 SPSのインストールの最後に、 **ライセンスキーインストーラーユーティリティ** に表示される **Host ID** をメモしてください。

2. **Host ID** をノートにメモするか、ファイルに保存します。ファイルに保存した場合は、そのファイルをインターネットにアクセスできるシステムにコピーしてください。それ以外の場合は、インターネットにアクセスできるシステムまでノートを持っていってください。

3. SIOS Protection Suite **Entitlement ID** (認証コード) を所有していることを確認します。ライセンスを取得するために必要な Entitlement ID が記載された E メールをソフトウェアとともに受け取っているはずです。

4. SIOS Technology Corp. の Licensing Operations Portal からライセンスを取得します。

a. インターネットにアクセスできるシステムから SIOS Technology Corp. の Licensing Operations Portal にアクセスし、 **ユーザー名** と **パスワード** を入力してログインします。（アカウントをお持ちでない場合は登録してください。）
**注記:** 新規ユーザーの場合、出荷メールに記載されているEntitlement IDを入力してください。

b. **[Activation and Entitlements]** ドロップダウンから、 **[List Entitlements]** を選択します。

c. ライセンスを取得したい製品の左側にあるチェックボックスにチェックを入れます。

d. **[Action]** ドロップダウンから **[Activate]** を選択し、必要な情報 （システムのHOSTNAMEなど）を入力して **[Next]** を選択します。

e **[Gray Plus Sign]** をクリックして定義済みのホストを選択するか、 **[Green Plus Sign]** を選択して新しいホストを作成します。

f. [Node Locked Host] の選択肢に [ANY] が表示されている場合には **[ANY]** を選択して [OK] をクリックしてください。[ANY] が表示されていない場合には **[ETHERNET MAC ADDRESS]** を選択し、Host ID（MACアドレス）を入力して **[OK]** をクリックし、次に **[Generate]** をクリックします。

> 注記: Host IDは12文字で、スペース、コロン、ダッシュ、区切り文字は使用できません。

g. **[Fulfillment ID]** の左側のボックスをオンにして、 **[Complete]** を選択します。

h. **[License Support]** ドロップダウンから **[List Licenses]** を選択します。 **[Fulfillment ID]** の左側のボックスをオンにして、 **[View]** ドロップダウンから **[Email]** を選択します。

i. ライセンス送信先の有効なメールアドレスを入力して、 **[Send]** を選択します。

j. 受信したメールを確認します。

k. ファイルを該当するシステムにコピーします。

5. ライセンスをインストールします。

- 各システム上で、ライセンスファイルを /var/LifeKeeper/licenseにコピーします。/opt/LifeKeeper/bin/lkkeyins を実行し、ファイル名（フルパスを含む）を指定します。

6. その他のサーバーについて、上記の手順を繰り返します。他のSPSサーバー上では、サーバーごとに一意の Host ID を使用してライセンスをインストールする必要があります

7. SIOS Protection Suite for Linuxを再起動します。

# プライマリーネットワークインターフェースを変更すると、ライセンスの Rehost が必要になる場合がある

ライセンスキーインストーラーユーティリティー で使用されるホスト ID は、 SIOS Protection Suite for Linux サーバーのプライマリーネットワークインターフェースカード (NIC) から取得されます。 SIOS Protection Suite for Linux は、起動されるたびに有効なライセンスを確認します。将来、 SIOS Protection Suite for Linux サーバーの NIC を交換してホスト ID が変わった場合は、次に SIOS Protection Suite for Linux を停止したとき、再起動する前にライセンスの Rehost を実行する必要があります。 SIOS Technology Corp. の Licensing Operations Portal にログインして、画面上メニューを [License Support] > [List Licenses] とクリックして画面遷移させた後、目的のライセンスを選択して [Action] > [Rehost] の順に選択し、Rehost を実行してください。

**注記:** 製品サポートへのご連絡なしにお客様ご自身で Rehost を行うことができるのは 6 か月に 4 回です。

# 4.3.9.1. インターネットホスト ID の取得

マシンのインターネットホスト ID を取得するには、lmutil を使用します。インターネットホスト ID は、通常、システムのプライマリネットワークインターフェースのプライマリ IP アドレスです。インターネットホスト ID は、Ethernet (または MAC) ホスト ID の代替として使用することができ、VM クローンのために MAC アドレスが変更される可能性がある仮想環境において望ましいと考えられます。

1. 次のコマンドを実行します。

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

2. プログラムから返される ID を記録してください。

例:

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

“INTERNET=172.17.100.161”
```

注記: この情報は、SIOS Technology Corp. から取得したパーマネントライセンスキーに記載されている情報と一致する必要があります。

# 4.4. SIOS Protection Suite ソフトウェアのインストール

本ページでは Linux OS について基本的な知識がある方にむけて SIOS Protection Suite for Linux (SPS) のインストールについて説明しています。 その他詳細については SIOS Protection Suite for Linux テクニカルドキュメンテーション を参照してください。

## インストール前の要件

SPS for Linux をインストールする前に、下記の要件について確認してください。

- **SIOS Protection Suite for Linux リリースノート** -SIOS Protection Suite for Linux リリースノートにはサポート対象のプラットフォーム、オペレーティングシステム、アプリケーション、ストレージについて記載されています。最新のバグフィックスについても記載されています。
- **TCP/IP 接続と名前解決** – GUI の機能を使用するためにクラスターノードの双方で名前解決ができる必要があります。DNS サービスまたは */etc/hosts* を使用してください。 ローカルホストは 127.0.0.1 となります。
- ファイアウォール – 以下のポートを使用します。
  - コミュニケーションパス (TCP): 7365/tcp
  - GUI サーバーとの通信: 81/tcp、82/tcp
  - GUI サーバーとクライアントとの RMI 通信: 1024/tcp 以降の全ポート
  - DataKeeper の同期 (DataKeeper 使用時): 10001+<mirror number>+<256 * i>

ファイアウォールに関して

- GUI サーバー、クライアントの通信に使用するポートは SPS がインストールされるクラスターノードと、GUI クライアントを動作させる全てのシステムでポートが開放されている必要があります。
- DataKeeper で使用するポートは上記の計算式で割り出すことができます。i の値は 0から開始し、使用されていないポートが見つかるとそのポートを使用します。例えば mirror number が 0 の DataKeeper リソースが存在している環境で、10001 番ポートが別のアプリケーションによって使用されていた場合は、10257 番ポートを使用します。
- GUI サーバー、クライアント間の通信では Java の RMI にて 1024 番以降のポートをランダムに使用します。クラスターシステムにアクセス制御などを適用する場合は、これらのポートを考慮し、パケットフィルタリングを行う必要があります。セキュリティー対策の観点で本仕様が問題となる場合は、ssh の Xフォワーディングを使用して対応することもできます。設定方法等は テクニカルドキュメント をご参照ください。
- 使用しているポート番号に以下を追加します。 WebGUI サーバーのプロセスと lkpolicy コマンドによるポリシー設定で使用：778(SSL)/tcp

- ◦ **SELinux の設定確認** – SELinux のモードによっては、SPS for Linux はインストールできません。
  - enforcing モード – SPS for Linux をインストールできません。
  - permissive モード – SPS for Linux をインストールできます。(いくつかのテスト済み ARK 環境を除いて推奨できません。)
    - SAP 環境で必要にならない限り、SELinux を permissive モードにすることは推奨されていません。クラスターでの動作が必要となるアプリケーションが permissive モードをサポートしているか確認してください。SELinux の permissive モードは次の

ARK でテスト済みです。: SAP / SAP MaxDB / Sybase / Oracle / DB2 / NFS / DataKeeper / NAS / EC2 / IP / FileSystem / MQ。必要となるパッケージについては Linux の依存関係 を参照してください。

- disabled モード – SPS for Linux をインストールできます。
  - SELinux を無効にする方法は、OS 側のドキュメントを確認してください。
- ご利用のディストリビューションから提供される適切なパッケージをインストールしてください。
- sg3_utils パッケージは DMMP Recovery Kit、PowerPath Recovery Kit 等のマルチパス用のリカバリーキットを使用する環境に必要です。マルチパス用のリカバリーキット を使用しない環境には必要ありません。

- ◦ 既知の問題 を確認してください。– ご利用の環境に既知の問題がないか確認してください。

## SPS ソフトウェアのインストール

SPS ソフトウェアを、SPS 設定内の各サーバにインストールします。

LifeKeeper インストールセットアップスクリプトは依存するすべてのパッケージを確実にインストールするためにパッケージマネージャーツール（yum または zypper）を使用するため、LifeKeeper が依存するパッケージは自動的にインストールされます。

> **重要**：LifeKeeper を正しくインストールするには、yum または zypper の機能構成が必要です。非機能構成では、インストールが失敗する可能性があります（詳細については、インストールに関する既知の問題 を参照してください）。さらに、インストールがハングする可能性があるため、パッケージマネージャーレポジトリまたは rpm データベースはロックしないでください。依存パッケージをパッケージマネージャーから自動的にインストールできない場合は、事前にLinux の依存関係 を参照して必要なパッケージをインストールしてください。

SPS for Linux のイメージファイル (sps.img) では、システムに SPS をインストールする際に必要なユーザー対話型システムセットアップタスクを実行するように設計された、一連のインストールスクリプトを提供しています（対話モード を参照してください）。 非ユーザー対話型インストールを実行することもできます。非ユーザー対話型インストールについても同様です（非対話モード を参照してください）。

サーバの Host ID または Entitlement ID を取得および表示するユーティリティを提供するライセンシングユーティリティパッケージもインストールされます。Host ID/Entitlement ID は、SPS を実行するための有効なライセンスを取得するために使用されます。ライセンスの取得とインストール方法については ライセンスの取得とインストール を参照してください。

> **注記:** これらのインストール手順は、読者がサーバにインストールされた Linux オペレーティングシステムに精通していることを前提としています。

> **重要**：
> • SPS の共有ストレージへのインストールはサポートされていません。各サーバのローカ

ルディスクに独自のコピーをインストールする必要があります。
• すべての SPS パッケージは、 /opt/LifeKeeper ディレクトリにインストールされます。

SPS は、使用している Linux ディストリビューションに関わらず、コマンドラインでインストールされます。

- インストール作業は セットアップスクリプトの操作 を参照してください。
- アップグレード方法については、SPS のアップグレード を参照してください。

# 4.5. セットアップスクリプトの操作

LifeKeeper をインストールするために、セットアップスクリプトを使用して以下の作業を行います。

## セットアップスクリプトの機能

セットアップスクリプトは以下の2つの使用方法があります。

1. 対話型インストール
   メニュー画面からLifeKeeperの構成を設定して、インストールまで行います。
   このとき構成情報を保存すると、後述の非対話型インストールで使用できます。
2. 非対話型インストール
   保存した構成情報を用いて、LifeKeeperをインストールします。
   ユーザーへの問い合わせは発生しないので、構築ツール（Ansible）などから実行することも可能です。

## セットアップの手順

### 対話型インストール

1. root ユーザーでログイン後、次のコマンドを使用して、sps.img ファイルをマウントします。

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

   ここで、PATH はイメージへのパスです
   IMAGE_NAME はイメージの名前です
   MOUNT_POINT はマウント位置へのパスです

2. sps.img がマウントされたディレクトリに移動して、次のコマンドを入力します。

   ```
   ./setup [-s <response_file>]
   ```

   -s オプションを指定すると、設定した構成情報をresponse_fileに保存できます。このファイルは非対話型インストールで使用します。

3. スクリプトを実行すると、まず環境情報の収集を行います。

   インストールまたはアップグレードのシステム要件が満たされていない場合は、エラーメッセージが表示され、インストール/アップグレードはキャンセルされます。

   また、何らかの問題が発生した場合や設定の変更が必要な場合は、警告メッセージが表示され、インストールを続行するか中止するかをユーザーに決定するように求めます。

4. メニュー画面にてインストールするSPSの機能やアプリケーションリカバリーキット（ARK）を選

択します。詳しくは「対話画面の操作」を参照してください。

5. インストールするSPSの機能やアプリケーションリカバリーキット（ARK）を選択し、**<Done>** を選択すると、インストールが始まります。

   インストール完了後に何らかの通知が出力された場合は、その指示に従い必要な対応を行ってください。

## 非対話的インストール

1. root ユーザーでログイン後、次のコマンドを使用して、sps.img ファイルをマウントします。

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

   ここで、PATH はイメージへのパスです
   IMAGE_NAME はイメージの名前です
   MOUNT_POINT はマウント位置へのパスです

2. LifeKeeperのインストールを行うシステムへ、構成情報ファイルをコピーした後、次のコマンドを実行します。

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

   「-q y」オプションを指定すると、警告に対して了解したと渡します。

> **注記:** 非対話モードでのインストールで、-sオプションで保存した設定ファイルを使用する場合、そのファイルが使用されるシステムは、そのファイルが生成されたシステムと同じように構成されている必要があります。システムの違いが多すぎると、非対話モードでのインストールは失敗する可能性があります。
> create_response_fileスクリプトで作成した設定ファイルはそのような制限がありません。

## 構成情報の作成

非対話型インストールで使用する構成情報ファイルは、`setup -s <response_file>`でセットアップ作業中に作成するか、create_response_fileスクリプトで事前に作成することが出来ます。

1. root ユーザーでログイン後、次のコマンドを使用して、sps.img ファイルをマウントします。

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

   ここで、PATH はイメージへのパスです
   IMAGE_NAME はイメージの名前です
   MOUNT_POINT はマウント位置へのパスです

2. sps.img がマウントされたディレクトリに移動して、次のコマンドを入力します。

```
./create_response_file <response_file>
```

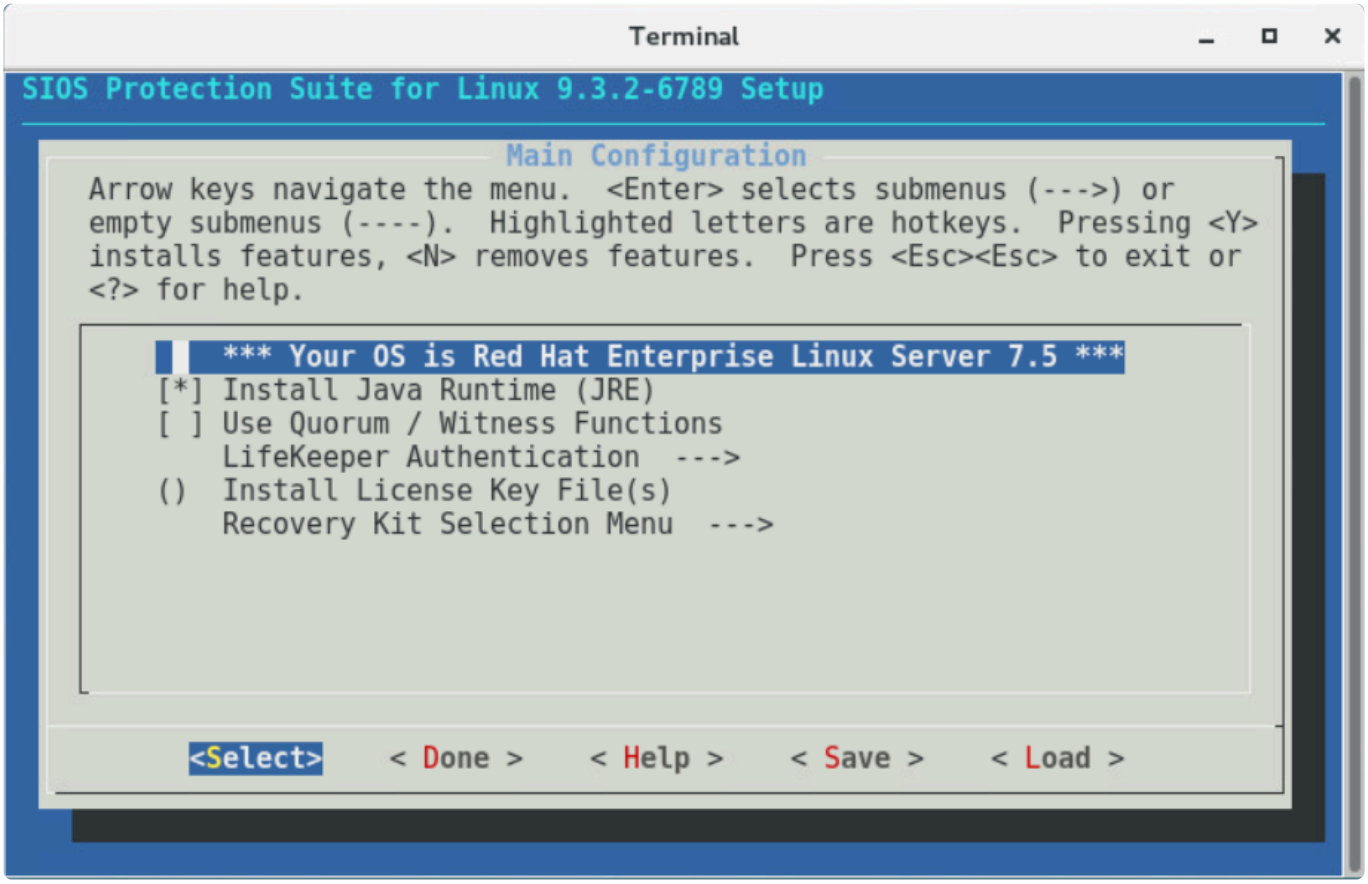
3. メニュー画面にてインストールするSPSの機能やアプリケーションリカバリーキット（ARK）を選択します。詳しくは「対話画面の操作」を参照してください。

4. インストールするSPSの機能やアプリケーションリカバリーキット（ARK）を選択し、**<Done>** を選択すると、構成をresponse_fileへ保存してスクリプトが終了します。response_fileはインストール先のシステムへコピーして使用します。

   保存された構成情報を変更する場合は、再度手順2～4を実行してください。

## 対話画面の操作

以下のようなメニュー画面が表示されます。




以下のキーで項目選択を行います。

| | |
|---|---|
| ↑↓ | 選択項目の移動 |
| ← → | 最下行ボタンの移動 |
| ENTER | 選択したサブメニューを開く |
| Y / N / Enter | 選択した項目のオン、オフ、または反転 |

最下行ボタンは以下の動作を行います。

| | |
|---|---|
| Select | 選択項目の ON / OFF ・子画面へ移動 |
| Done | 現画面を終了して親画面に戻ります<br>Main Configuration 画面の場合は、メニュー画面を終了して、インストールの確認画面へ移行します |
| Help | 選択項目のヘルプ画面を表示します |
| Save | 設定した構成情報をファイルに保存します<br>保存した構成情報ファイルは非対話モードでのインストールに使用できます<br>(create_response_fileでは無効です) |
| Load | 保存済み構成情報ファイルの読み込み<br>(create_response_fileでは無効です) |

メニューボタン <Save> および <Load> をクリックすると、現在の設定を保存するとき、または保存した設定を読み込むときに使用する設定ファイル名を尋ねるダイアログ画面が表示されます。デフォルトのファイル名を変更したい場合は、[TAB] キーを使用してファイル名フィールドに移動し、新しいファイル名を入力してください。 **注記:** <Save> をクリックした場合、同じ名前でファイルを上書きする前に確認を求められます。

各項目では以下の機能を設定できます。

アップグレード中は、設定可能な項目のみが一覧表示されます。 ホットキー <Z> を使用すると、アップグレード時は変更されない項目が表示されます。

- **Install Java Runtime (JRE)**

  LifeKeeper GUI が使用する Java 実行環境をインストールします。

- **Restart NFS service**

  NFS ARK を使用する場合に選択可能です。NFS サービスの設定変更を即時反映させたい場合に選択します。

  注記： NFSサービスを自動的に再起動したくない場合は、NFS Recovery Kit を使用する前に設定変更を反映させるための再起動を行う必要があります。

- **Use Quorum / Witness functions**

  Quorum/Witness を使用します。機能の詳細はテクニカルドキュメンテーションのQuorum/Witnessを参照してください。

- **LifeKeeper Authentication**

  SPS for Linux GUI へのログインを許可するユーザーとその権限を指定します。 複数のユーザー名を空白で区切って指定できます。 詳細は、 GUIユーザー設定 を参照してください。

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

LifeKeeper のライセンスファイルを指定します。ファイル名を入力できるダイアログに切り替わるので、ライセンスファイルのフルパス名を入力して下さい。スペース区切りで複数のライセンスファイルを指定することが出来ます。

ライセンスの取得方法については[ライセンスの取得とインストール]を参照してください。

- **Recovery Kit Selection Menu**

  インストールするアプリケーションリカバリーキットを選択します。 アプリケーションリカバリーキットは、機能に基づいていくつかのカテゴリーに分類されています。

  詳細については、[アプリケーションリカバリーキットの分類]を参照してください。

- **LifeKeeper Startup After Install**

  インストールもしくはアップデート完了後に LifeKeeper を起動します。

> **注記:** SPS-L データレプリケーションパッケージは、サポートされているOSディストリビューションによってはカーネルモジュールをインストールする場合があるため、カーネルのアップグレード時に SPS-L の再インストールが必要になることがあります。これは、カーネルバージョン 3.10.0-514 以降を実行している RedHat、CentOS、Oracle Enterprise Linux（UEK カーネル含む）、及び 4.12.14-95 以降の SUSE Linux Enterprise Server の OS ディストリビューションに適用されます。

## アプリケーションリカバリーキットの追加と削除

インストール完了後にアプリケーションリカバリーキットを追加するには、setup を実行し、[Recovery Kit] でリカバリーキットを選択し、次にアプリケーションリカバリーキットのカテゴリーを選択して、ご希望のキットを選択します。不要になったアプリケーションリカバリーキットの選択を解除すると、そのキットは削除されます。
ただし、使用中のリソースはキットを削除できないので、事前にリソースを削除してください。

## 修復インストール

SPS for Linux インストールを修復するには、「`--force`」オプションを使用してセットアップを実行します。 修復インストールでは、失われたファイルや破損したファイルを置き換えてインストールを更新します。

## setup スクリプトのオプション

setup スクリプトは以下のオプションを指定して実行できます。

- `-f <response_file>`

  非対話モードでインストールを行います。<response_file>には、インストール中に使用する設定情報が含まれています。

- `-s <response_file>`

  メニューで選択した内容を含む設定ファイルを保存します。このファイルを「-f」オプションと一緒に使用して、同じ LifeKeeper 構成を別のシステムにインストールすることができます。たとえば、次のコマンドを実行します。

  ```
  ./setup -s <response_file>
  ```

  必要なパッケージとオプションを選択し、セットアップを完了します。

  その後、以下のコマンドを実行し、最初にセットアップを実行したときに選択したのと同じオプションで LifeKeeper のサイレントインストールを（別のシステムで）実行します。

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

- `--force`

  SPS for Linux を強制的に再インストールします。

- `-q <y/n>`

  非対話モードでのインストール中に発生する可能性のある確認への回答を指定します。

## リカバリーキットの分類

| カテゴリー | 説明 |
|---|---|
| Application Suite | SAPやIBM MQ などアプリケーションを保護するリカバリーキットが所属します |
| Networking | EC2やRoute53など、クラウド内のネットワークサービスを保護するリカバリーキットが所属します |
| Database | Oracle、PostgreSQL、MaxDBなどのデータベースアプリケーションを保護するリカバリーキットが所属します |
| File Sharing | NFSやSambaなどのファイル共有サービスを保護するリカバリーキットが所属します |
| Mail Server | Postfixなどのメールサービスを保護するリカバリーキットが所属します |
| Storage | DataKeeper（レプリケーション）、Device Mapper（DM）マルチパス（DMMP）、Network Attached Storage（NAS）など（これらに限らない）のデータ保管方法を保護するリカバリーキットが所属します |
| Web Server | Apache などのウェブサービスを保護するリカバリーキットが所属します |

# 4.6. SPS LifeKeeper インストールの確認

SPS LifeKeeper パッケージが正しくインストールされていることを確認するには、コマンドラインで次のコマンドを入力してください。

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

注記: パッケージが正しくインストールされている場合、このコマンドは何も出力しません。

コマンドラインからクエリを実行するには、次のコマンドを入力してください。

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

注記: このコマンドの予想される出力は、パッケージ情報です。

# 4.7. SPS のアップグレード

v9.5.0より、SIOSは新しいSAP HANA Application Recovery Kitをリリースしました。SIOSは **2022年3月31日** まで、9.4.xリリースでのSAP HANA gen/appベースのRecovery Kitのサポートを継続します。
• 既存のSAP HANA gen/appベースのRecovery Kitはv9.5.0では **サポートされていません** 。
• SIOS Protection Suite for Linux v9.5以降をご利用の場合は、新しい（ビルトインの）SAP HANA Application Recovery Kitを使用する必要があります。
• SIOS Protection Suite for Linux v9.5.0へのアップグレードを希望するユーザーは、既存のSAP HANA gen/appベースのRecovery Kitから新しい SAP HANA Recovery Kit に **移行する必要があります** 。詳細については、 SAP HANA Gen/AppからSAP HANA Recovery Kitへのアップグレード を参照してください。

SPS for Linux は、既存の階層を保ちながら将来のリリースにアップグレードすることができます。この情報をよく検討して、アプリケーションのダウンタイムを最小限に抑えるようにしてください。

注記：最新バージョンの LifeKeeper にアップグレードできるのは、2 世代前までのバージョンです。(2 世代間アップデートの例： **9.1.x -> 9.3.x**) 。それより前のバージョンからアップグレードする場合、古いバージョンをアンインストールし、SIOS Protection Suite for Linux を再インストールする必要があります。旧バージョンをアンインストールする代わりに、旧バージョンを 1 世代前もしくは 2 世代前のバージョンにアップグレードした上で、最新バージョンにアップグレードする方法もあります。

注記：アップグレード中に `lkbackup` を使用する場合の詳細については、lkbackupの既知の問題 を参照してください。

1. クラスタをアップグレードする場合、そのアップグレードされるサーバからすべてのアプリケーションを切り替えてください。この操作を実行するには、手動で行うか、またはLifeKeeperのシャットダウンストラテジーを「**Switchover**」に設定します。これにより、LifeKeeper が停止したり、サーバがシャットダウンしたときにアプリケーションが切り替えられます。

2. 必要な場合、SPS をアップグレードする前に、Linux オペレーティングシステムをアップグレードしてください。オペレーティングシステムのアップグレードを実行する前に、アップグレードするサーバのすべてのリソースを拡張解除することをお勧めします。

3. セットアップスクリプトの操作 を参照してLifeKeeperをアップグレードします

4. アップグレードされたサーバにすべてのアプリケーションを切り替えてください。

5. この手順を SPS クラスタ内のアップグレード対象の各サーバに対して実行します。

注意：同じクラスタ内のすべてのシステム上に、同じバージョンおよびリリースの SPS

をインストールする必要があります。通常、バージョンまたはリリースの異なる SPS には互換性がありません。ローリングアップグレード以外の状況で、異なるバージョンまたはリリースが存在し、クラスタ内の別のシステムで実行されている場合には、LifeKeeper を起動しないでください。

## OS をアップグレードする際の注意点

OS をアップグレードする際には、現在インストールされている LifeKeeper がアップグレード後の OS をサポートしているかを確認する必要があります。サポートしていない場合は、LifeKeeper のアップグレードも必要になります。 サポート対象のオペレーティングシステム を参照して確認してください。

OS をアップグレードする前に、lkbackup コマンド にて LifeKeeper 設定のバックアップを実施することを推奨します。

注記：lkbackup を使用する場合、 lkbackup の既知の問題 を参照してください。

1. クラスタをアップグレードする場合、そのアップグレードされるサーバからすべてのアプリケーションを切り替えてください。この操作を実行するには、手動で行うか、または LifeKeeper のシャットダウンストラテジーを「Switchover」に設定します。これにより、LifeKeeper が停止したり、サーバがシャットダウンしたときにアプリケーションが切り替えられます
2. LifeKeeper を停止させてください
3. OS アップグレードを実施して、OS を再起動してください
4. LifeKeeperのアップグレードが必要な場合は、LifeKeeperのアップグレードを実施してください。LifeKeeper のアップグレード手順については、SPS のアップグレード を参照してください。アップグレードを実施しない場合は、setup を再実行して新しいOSに対応した設定を行ってください。
5. LifeKeeper を起動してください
6. アップグレードされたサーバにすべてのアプリケーションを切り替えてください
7. この手順を SPS クラスタ内の全てのサーバに対して実行します

注意：同じクラスタ内のすべてのシステム上に、同じバージョンの OS および同じバージョンの LifeKeeper をインストールする必要がありますが、アップデート作業中に発生する一時的なバージョン差に関してはサポート対象です。

# 5. SIOS Protection Suite for Linux テクニカルドキュメンテーション

SIOS Protection Suite (SPS) for Linuxは、単一のエンタープライズクラス・ソリューションで、高可用性クラスタリングと革新的なデータ複製機能を統合します。

## SPS for Linuxで統合されたコンポーネント

**SIOS LifeKeeper** は、完全な耐障害性ソフトウェアソリューションを提供し、サーバーのファイルシステム、アプリケーション、およびプロセスの高可用性を実現します。LifeKeeperは、カスタマイズされたフォールトトレラントハードウェアを必要としません。LifeKeeperでは、ネットワーク内で2つ以上のシステムをグループ化するだけでサイト固有の設定データが作成され、自動的に障害を検出して回復します。

障害が発生した場合、LifeKeeperは保護されているリソースを障害が発生したサーバーから指定されたバックアップサーバに移行します。実際のスイッチオーバー中にユーザーの操作は一瞬中断されますが、LifeKeeperはオペレーターの介入なしにバックアップサーバーの操作を復元します。

**SIOS DataKeeper** は、LifeKeeper環境用の統合データミラーリング機能を提供します。この機能により、LifeKeeperリソースを共有ストレージを使用している環境および共有ストレージをしていない環境で運用できます。

---

ドキュメンテーションとトレーニング

# 5.1. はじめに

## SIOS Protection Suite for Linux について

SIOS Protection Suite (SPS) for Linux は、高可用性のクラスタリングと革新的なデータ複製機能をエンタープライズクラスのソリューションに統合したものです。

### SPS for Linux の統合コンポーネント

**SIOS LifeKeeper** は、障害回復性の高いソフトウェアソリューションであり、お使いのサーバのファイルシステム、アプリケーション、およびプロセスの高い可用性を維持します。LifeKeeper には、カスタマイズした耐障害性のハードウェアは不要です。LifeKeeper を使用するには、ネットワーク内にある 2 台以上のシステムをグループ化するだけです。サイト固有の構成データが作成され、自動の障害検出とリカバリが実行されます。

障害が発生した場合、障害が発生したサーバから LifeKeeper が保護しているリソースを指定のバックアップサーバに移行します。実際のスイッチオーバ時に短時間の中断が発生します。ただし、オペレータの介入なしに LifeKeeper がバックアップサーバに動作をリストアします。

**SIOS DataKeeper** は、LifeKeeper 環境において統合データミラーリング機能を提供します。この機能により、LifeKeeper リソースが共有/非共有ストレージ環境で動作可能になります。

---

ドキュメンテーションとトレーニング

# 5.2. ドキュメンテーションとトレーニング

## ドキュメンテーション

SIOS Protection Suite for Linux をインストール、設定、管理、およびトラブルシューティングするための方法を説明する完全なリファレンスです。次のセクションは、SPS for Linux のあらゆる側面を網羅します。

| セクション | 説明 |
|---|---|
| はじめに | SIOS Protection Suite for Linux 製品の概要を説明します。ソフトウェアパッケージと設定コンセプトが含まれます。 |
| SPS for Linux インストールガイド | SPS 環境のプランニングと設定、SPS のインストールとライセンス、LifeKeeper のグラフィカルユーザインターフェース (GUI) の設定に役立つ情報を提供します。 |
| 設定 | クラスタ内の各サーバで LifeKeeper ソフトウェアを設定するための詳細情報と手順があります。 |
| 管理 | サーバのプロパティの編集やリソースの作成などのサーバレベルの作業、およびリソースの編集、拡張、削除などのリソースレベルの作業について説明します。 |
| User's Guide | LifeKeeper GUI で実行できる多数の作業を含めて、LifeKeeper の GUI に関する詳細情報があります。テクニカルノート セクション、および多数の高度なトピック もあります。 |
| DataKeeper | SIOS DataKeeper for Linux の計画とインストールの手順、および管理、設定、およびユーザの情報があります。 |
| トラブルシューティング | 既知の問題と制限について説明し、SIOS LifeKeeper for Linux のインストール、設定、および使用を行うときに発生する可能性がある問題に対する解決策を説明します。 |
| Recovery Kit | LifeKeeperで特定のアプリケーションを管理および制御するために必要なオプションの Recovery Kit のプランニングおよびインストール手順、管理、設定、およびユーザ情報が含まれます。 |
| エラーコードの検索 | SIOS Protection Suite for Linux の使用中に表示される可能性のあるすべてのメッセージの一覧が含まれます。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。この全一覧から、受信したエラーコードを検索できます。 |

## トレーニング

SPS トレーニングは、SIOS Technology Corp. または代理店から受講可能です。詳細については、営業担当者にお問い合わせください。

# テクニカルサポート

SIOS Technology Corp. と有効なサポート契約を結んだお客様は、SIOS Technology Corp. のセルフサービスサポートポータル にアクセスできます。

SIOS Technology Corp. のセルフサービスサポートポータル では、以下のことができます。

- 弊社の ソリューションナレッジベース から、問題の解決策と質問に対する回答を検索する。
- 次のメニューを選択して、年中無休の SIOS Technology Corp. のサポートチームにアクセスする。
  - **Log a Case** - 新しいインシデントを報告する。
  - **View Cases** - お客様の未解決と解決済みのインシデントをすべて表示する。
  - **Review Top Solutions** - 弊社のお客様が表示した、最も一般的な問題の解決策の情報を表示する。

セルフサービスポータルを設定してアカウントを有効にする方法については、SIOS Technology Corp. のサポート (support@us.sios.com) にお問い合わせください。

また、SIOS Technology Corp. のサポートには、以下の方法でも連絡できます。

1-877-457-5113 (通話料無料)

1-803-808-4270 (米国以外のお客様)

電子メール:support@us.sios.com

# 5.3. lkbackup

## lkbackup を使用したバックアップの作成

> ! LifeKeeper もしくは OS のアップグレードを実施する際は、全てのノードで LifeKeeper 階層のバックアップを作成してください。

以下のコマンドを実施してバックアップを作成してください。

```
/opt/LifeKeeper/bin/lkbackup -c
```

以下のファイルにバックアップが作成されます。

*/opt/LifeKeeper/config/archive.<date-time-stamp>.tar.gz*

### 自動 lkbackup

> ✻ 自動 **lkbackup** は 7.5 以上のバージョンで実行できます。バックアップファイルは以下の名前で保存されます。
> */opt/LifeKeeper/config/auto-backup.<x>.tgz* ( <x> は、シーケンス番号です。)

#### 自動 lkbackup の時間を変更するには

1. */etc/crontab* にアクセスします。

   エントリーは以下のようになっています。

   *0 3 * * * root /opt/LifeKeeper/bin/backupadm -c* (0 3 はバックアップが実施される時刻を表し、この例では毎日 3:00 にバックアップが実施されます。)

2. 時刻を変更します。

## バックアップからリストアを実行する

以下のコマンドを使用してリストアを実行します。

1. `/opt/LifeKeeper/bin/lkcli stop`
2. `/opt/LifeKeeper/bin/lkbackup -x -f <file-name>`
3. `/opt/LifeKeeper/bin/lkcli start`

クラスター内の全てのシステム上でバックアップからリストアを実行してください。

DataKeeper リソースで lkbackup を使用する場合は、 全同期 が必要です。

# 5.4. LifeKeeper

以下のSPS製品のマニュアルは、サイオステクノロジーのウェブサイトでご覧いただけます。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- アプリケーションリカバリーキットのマニュアル

# 5.4.1. はじめに

SIOS LifeKeeper for Linux は、さまざまなストレージ構成をサポートし、最大 32 ノードの高可用性クラスタリングを提供します。共有ストレージ（ファイバチャネル SAN、iSCSI）、ネットワーク接続ストレージ（NAS）、ホストベースの複製、HP Continuous Access などのアレイベースの SAN 複製との統合、VMware 仮想ハードディスク（VMDK）などをサポートします。

---

保護対象のリソース

LifeKeeper Core

設定の概念

- 共通のハードウェアコンポーネント
- システムのグループ化の配置
- アクティブ - アクティブのグループ化
- アクティブ - スタンバイのグループ化
- Intelligent スイッチバックと Automatic スイッチバックの違い
- syslog によるログの記録
- リソース階層
  - リソースタイプ
  - リソースの状態
  - 階層の関係
  - イクイバレンシ情報
  - リソース階層の情報
  - リソース階層の例
  - ステータスの詳細表示
  - ステータスの簡略表示

障害検出とリカバリのシナリオ

- IP ローカルリカバリ

リソースのエラーリカバリのシナリオ

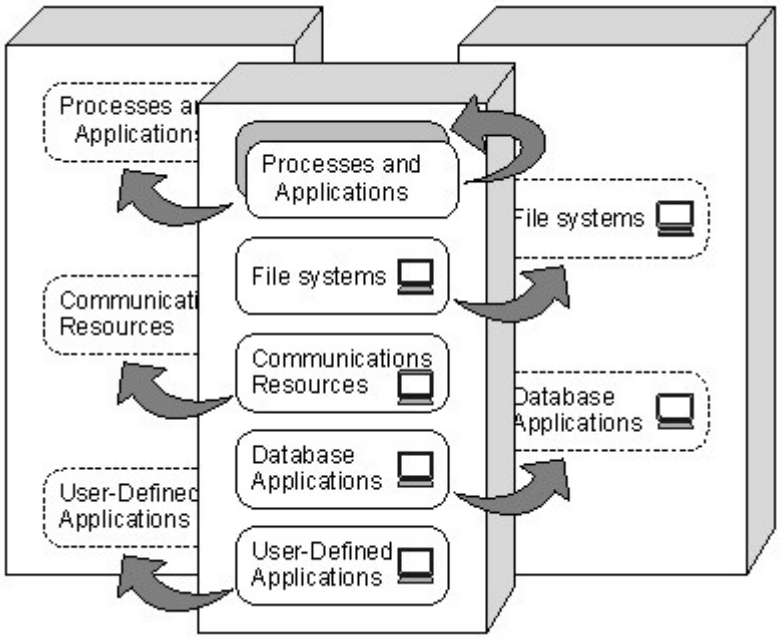
サーバの障害リカバリのシナリオ

# 5.4.1.1. 保護対象のリソース

LifeKeeper ファミリの製品には、多様なシステムリソースにフェイルオーバ保護を提供できるソフトウェアがあります。以下の図に、LifeKeeper の柔軟性、および自動リカバリを指定できるリソースタイプを示します。

- ファイルシステム 。LifeKeeper では、ext3、ext4、NFS、vxfs、xfs などのファイルシステムの指定とフェイルオーバができます。
- 通信リソース 。LifeKeeper には、TCP/IP のような通信リソースの通信 Recovery Kit が用意されています。
- インフラストラクチャリソース 。LifeKeeper には、NFS、Samba、LVM、WebSphere MQ、ソフトウェア RAID (md) など、Linux インフラストラクチャサービス用のオプションの Recovery Kit が用意されています。
- **Web** サーバリソース 。LifeKeeper には、Apache Web サーバリソース用のオプションの Recovery Kit が用意されています。
- データベースとその他のアプリケーション 。LifeKeeper には、Oracle、MySQL、PostgreSQL、Sybase、SAP DB/MaxDB および SAP HANA DB 1.0 / SAP HANA DB 2.0 などの主な RDBMS 製品、および SAP などのエンタープライズアプリケーション用のオプションの Recovery Kit が用意されています。
- クラウドリソース 。LifeKeeper には、EC2 EIP、Route53 などの通信リソース用のRecovery Kit が用意されています。

LifeKeeper は、多様なリソースタイプについて複数の回復方法 をサポートします。



# 5.4.1.2. LifeKeeper Core

LifeKeeper Core は、以下の 4 つの主要コンポーネントで構成されています。

- LifeKeeper Core ソフトウェア
- File System、Generic Application、Raw I/O、 IP、および Quick Service Protection (QSP) の Recovery Kit ソフトウェア
- LifeKeeper GUI ソフトウェア
- LifeKeeper のマニュアルページ

## LifeKeeper Core ソフトウェア

LifeKeeper Core ソフトウェアは、以下のコンポーネントで構成されます。

- LifeKeeper 構成データベース (LCD) - LCD は、LifeKeeper が保護するリソースの情報を保存します。リソースインスタンス、依存関係、イクイバレンシ情報、リカバリの方向、LifeKeeper の動作フラグに関する情報が含まれます。システムの起動後にデータが記憶されているように、データは共有メモリにキャッシュされ、ファイルに保存されます。
- LCD インターフェース (LCDI) - LCDI は、LCD に保存されているデータやデータの変更を要求するクエリを設定データベース (LCD) にクエリを送信します。また、リソースの状態や説明の情報を取得するために、Application Recovery Kit が LCDI を使用することもできます。
- LifeKeeper Communications Manager (LCM) - LCM は、クラスタ内にあるサーバのステータスの特定、および LifeKeeper のプロセス間通信 (ローカルとリモート) に使用されます。クラスタ内のあるサーバ上にあるすべてのコミュニケーションパスで LCM 通信がないことは、サーバに障害が発生したことを示します。
- LifeKeeper アラームインターフェース - LifeKeeper アラームインターフェースは、イベントを起動するためのインフラストラクチャです。LifeKeeper が保護するリソースに障害が検出された場合、アプリケーションデーモンにより sendevent プログラムが呼び出されます。sendevent プログラムが LCD と通信し、リカバリプロセスが使用可能かどうかを判断します。
- LifeKeeper のローカルリカバリ動作と制御のインターフェース （LRACI） - LRACI はリソースに適切なリカバリスクリプトを判断し、リソースに適切な restore / remove スクリプトを呼び出します。

## File System、Generic Application、IP、RAW I/O、および Quick Service Protection (QSP) の Recovery Kit ソフトウェア

LifeKeeper Core は、サーバ上の指定リソースを保護します。リソースを以下に示します。

- File Systems - LifeKeeper では、共有ストレージデバイス上にあるファイルシステムの指定とフェイルオーバができます。ファイルシステムは、共有 SCSI バス経由で 2 台のサーバからアクセス可能なディスク上に作成できます。LifeKeeper のファイルシステムリソースは、1 台目のサーバに作成されてから、2 台目のサーバに拡張されます。ファイルシステムの健全性監視 がディスクフルと不適切なマウント (またはアンマウント) のファイルシステム条件を検出します。検出した条件に従って、Recovery Kit が警告メッセージのログ記録、ローカルリカバリの試行、またはファイルシステムリソースのバックアップサーバへのフェイルオーバを実行できます。

  File System Recovery Kit に関連するヘルプトピックとして、ファイルシステムのリソース階層の作成 、拡張 、ファイルシステムの健全性の監視 などがあります。

- Generic Applications - Generic Application Recovery Kit は、リソースタイプに対して事前定義リカバリキットが指定されていない汎用アプリケーションやユーザ定義アプリケーションを保護できます。このキットを使用すると、特定アプリケーションについてカスタマイズした監視スクリプトやリカバリスクリプトを指定できます。

  Generic Application Recovery Kit に関連するヘルプトピックとして、汎用アプリケーションのリソース階層の作成 、拡張 などがあります。

- IP Addresses - IP Recovery Kit には、LifeKeeper 環境で、障害が発生したプライマリサーバから「切り替え可能な」IP アドレスをバックアップサーバにリカバリするメカニズムがあります。切り替え可能な IP アドレスとは、サーバ間で切り替えることができる仮想 IP アドレスであり、各サーバのネットワークインターフェースカードに関連付けられている IP アドレスとは別のものです。LifeKeeper で保護されているアプリケーションは切り替え可能な IP アドレスに関連付けられているので、プライマリサーバに障害が発生した場合、切り替え可能な IP アドレスはバックアップサーバに関連付けられます。LifeKeeper で保護されているリソースは、切り替え可能な IP アドレスです。

  特定の製品、構成、および管理に関する情報については、リカバリキットに含まれるIP Recovery Kit Technical Documentation を参照してください。

- RAW I/O - RAW I/O Recovery Kit は、カーネルのバッファリングを迂回するアプリケーションの RAW I/O デバイスをサポートします。RAW I/O Recovery Kit では、共有ストレージデバイスにボンディングされた RAW デバイスの指定とフェイルオーバができます。RAW デバイスは、リソースの作成前に、プライマリノードに設定する必要があります。RAW リソースを作成 した後、追加サーバに拡張 できます。

- Quick Service Protection (QSP) - QSP Recovery Kit は、OSのサービスを簡易的に保護する仕組みを提供します。OSのserviceコマンドで起動 (start)・停止 (stop) ができるサービスであれば、容易にリソースが作成できます。Generic Applicationを使っても同じことが実現できますが、QSPを使えばコードの開発が不要です。また、依存関係を作成することで、他のリソースで保護しているアプリケーションと連携してサービスを起動・停止させることができます。

  ただし、QSPのquickCheckでは簡易なチェックしか行っていません（serviceコマンドのstatusを利用しています）。実際にそのサービスが提供できているか、プロセスが動作しているかは保証されません。複雑な起動や停止処理、堅牢なチェックが必要な場合は、Generic Applicationの利用を検討してください。

QSPに関連するヘルプトピックスとして、QSPリソースの作成・拡張 があります。

## LifeKeeper GUI ソフトウェア

LifeKeeper GUI は、Java テクノロジを使用して開発されたクライアント/サーバアプリケーションであり、LifeKeeper およびその設定データ用のグラフィカルな管理インターフェースです。LifeKeeper GUI クライアントは、スタンドアロンの Java アプリケーション として実装されます。

## LifeKeeper のマニュアルページ

LifeKeeper 製品用の LifeKeeper Core のリファレンスマニュアルページです。

# 5.4.1.3. 設定の概念

LifeKeeper は、2 台以上のサーバを持つグループに対してユーザが定義したリソース階層に基づいて機能します。以下のトピックで、LifeKeeper のフェイルオーバ設定の概念を説明しています。

---

共通のハードウェアコンポーネント

システムのグループ化の配置

アクティブ - アクティブのグループ化

アクティブ - スタンバイのグループ化

Intelligent スイッチバックと Automatic スイッチバックの違い

syslog によるログの記録

リソース階層

# 5.4.1.3.1. 共通のハードウェアコンポーネント

LifeKeeper のすべての設定には、以下の共通コンポーネントが含まれます。

1. **サーバグループ** 。LifeKeeper が提供する障害回復機能は、2 台以上のサーバをクラスタにグループ化することを基礎にしています。サーバは、サポートする Linux のディストリビューションを実行するサポートするプラットフォームであれば、いずれでもかまいません。LifeKeeper には、複数の重なり合うグループにサーバを設定する柔軟性があります。ただし、リカバリ可能なリソースについての重要な要件は、リソースの役割と優先順位が定義されたサーバのグループをリンクすることです。リソースに対するサーバの優先順位は、現在実行中のサーバに障害が発生した場合に、どのサーバがそのリソースを復旧するかを決定するために使用されます。最高の優先順位を示す値は 1 です。特定のリソースについて、最高の優先順位の値 (通常は 1) を持つサーバが通常、そのリソースのプライマリサーバと呼ばれます。その他のサーバは、そのリソースのバックアップサーバとして定義されます。

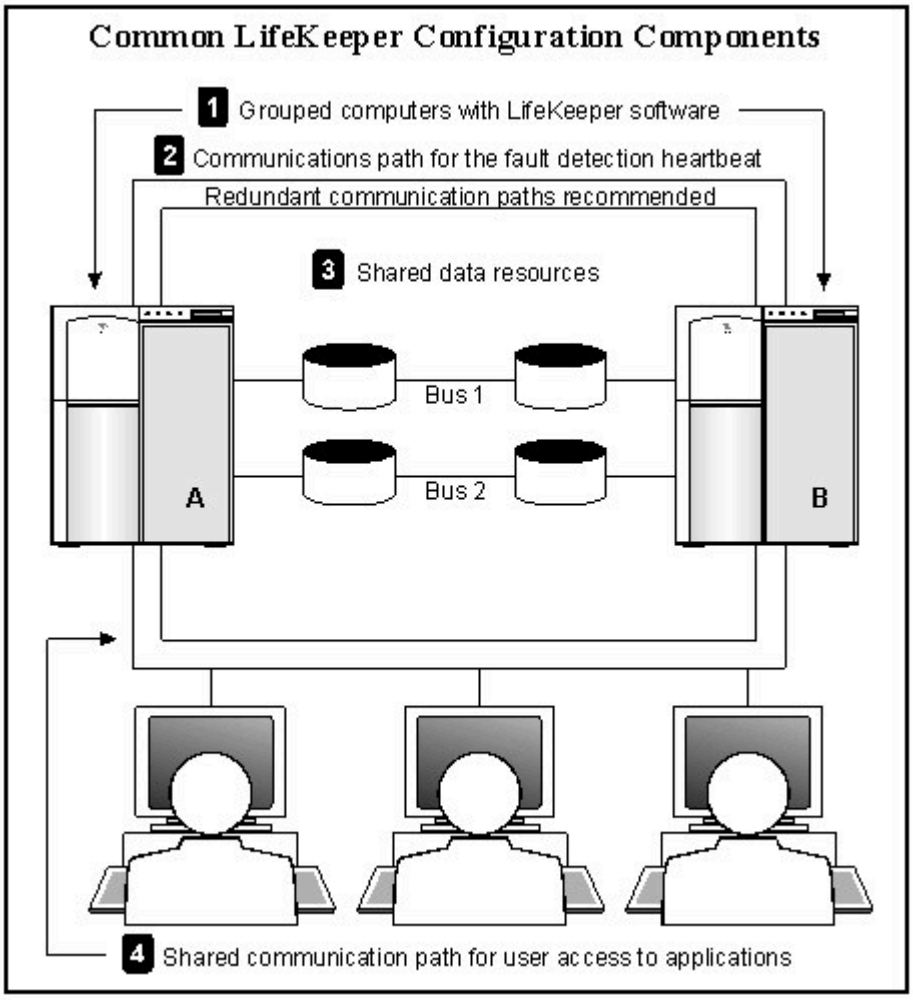
2. **コミュニケーションパス** 。LifeKeeper のハートビートは、LifeKeeper クラスタ内にあるサーバ間の定期的なメッセージで、主要な障害検出機能です。クラスタ内のすべてのサーバには、単純な通信障害でシステムに障害が発生しないように、冗長なハートビートコミュニケーションパス (comm パス) が必要です。2 つの独立したサブネットを使用する LAN ベース (TCP) の個別な 2 つのコミュニケーションパスが推奨されます (少なくとも 1 つのコミュニケーションパスをプライベートネットワークとして設定してください)。ただし、TCP と TTY のコミュニケーションパスの組み合わせの使用もサポートしています。TCP コミュニケーションパスは、他のシステムの通信にも使用できます。また、クラウド環境では一般にネットワークの内部構成が公開されていないので、物理的に別経路の LAN 回線を 2 つ用意することは困難です。しかしながら、基本的にクラウド側で物理的なネットワークの冗長化が行われている事が予想されるため、コミュニケーションパスが一つであっても動作上の信頼性を確保することができます。

   **注記:** TTY コミュニケーションパスは、クラスタ内の他のサーバがアクティブかどうかを検出するためにのみ LifeKeeper で使用されます。LifeKeeper の GUI は、TCP/IP を使用して、保護するリソースに関するステータス情報を通信します。TCP コミュニケーションパスが 2 つ設定されている場合、LifeKeeper はパブリックネットワークのコミュニケーションパスをリソースステータスの通信に使用します。このため、LifeKeeper の GUI が使用しているネットワークがダウンすると、TTY (または他の TCP) コミュニケーションパスが動作可能な場合でも、GUI には他のサーバのステータスが UNKNOWN として表示されます。

3. **共有データリソース** 。共有ストレージの構成では、LifeKeeper クラスタ内のサーバは同一セットのディスクに対するアクセスを共有します。プライマリサーバに障害が発生した場合、LifeKeeper は障害が発生したサーバ上にあるディスクのロック解除、および次に使用可能なバックアップサーバのディスクのロックを自動管理します。

4. **共有通信** 。LifeKeeper は TCP/IP アドレスのような通信リソースの切り替えを自動管理できるので、アプリケーションが現在どのサーバでアクティブになっているかには無関係に、ユーザはアプリケーションに接続できます。

# すべての LifeKeeper 設定に共通するコンポーネント



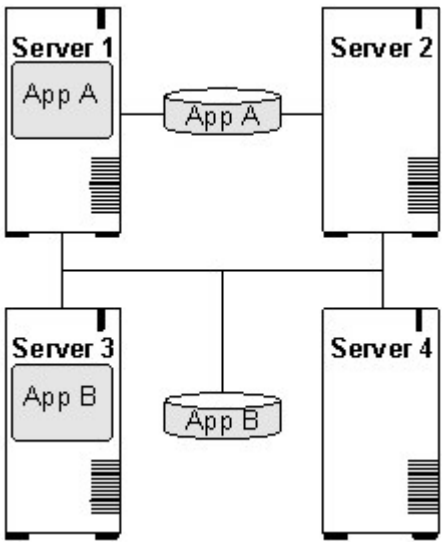
# 5.4.1.3.2. システムのグループ化の配置

リソース階層は、LifeKeeper サーバのクラスタに対して定義されます。ある階層について、各サーバに優先順位が割り当てられます。1 が最高の優先順位です。プライマリ、つまり優先順位が最高のサーバが、それらのリソースの通常動作に使用するコンピュータです。2 番目に高い優先順位を持つサーバがバックアップサーバであり、プライマリサーバに障害が発生した場合に、LifeKeeper がリソースを切り替える先のサーバです。

アクティブ / アクティブのグループ では、すべてのサーバがプロセスをアクティブに実行します。ただし、他のサーバのリソース階層ではバックアップサーバとしても機能します。アクティブ / スタンバイのグループ では、プライマリサーバは処理を実行し、バックアップサーバはプライマリサーバに障害が発生した場合に備えてスタンバイするように設定できます。スタンバイシステムは小型でパフォーマンスの低いシステムでもかまいませんが、プライマリサーバに障害が発生した場合にリソースの可用性を確保できるだけの処理能力が必要です。

共有リソースに対する物理的な接続とアクセスにより、グループ化のオプションが決まります。グループ化するサーバには、通信とハートビートパスがインストールされ、動作可能である必要があり、すべてのサーバが共有 SCSI またはファイバチャネルインターフェース経由で、ディスクリソースにアクセスできる必要があります。例えば、以下の図では、サーバ 1 のリソース *AppA* にはグループ化オプションが 1 つのみあります。この構成で *AppA* データベースへの共有アクセスを持つ他のサーバはサーバ 2 のみです。

ただし、サーバ 3 のリソース *AppB* は、その他 3 台のいずれを含むグループにも属するように設定できます。これは、この例の共有 SCSI バスが、構成内の 4 台すべてのサーバに *AppB* データベースへのアクセスを提供しているからです。



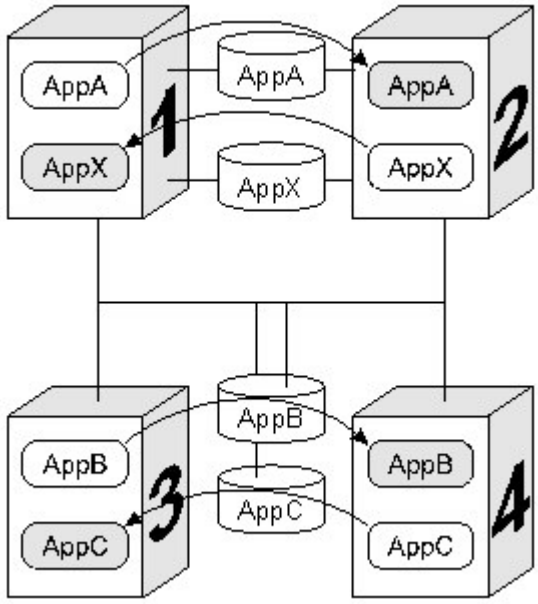
# 5.4.1.3.3. アクティブ - アクティブのグループ化

アクティブ / アクティブペアの設定では、すべてのサーバがプロセスをアクティブに実行します。また、他のサーバのリソース階層ではバックアップサーバとして機能します。

以下の設定例に、2 つのアクティブ / アクティブペアのサーバを示します。サーバ 1 は *AppA* を処理していますが、サーバ 2 で実行中の *AppX* のバックアップサーバとして機能します。この逆も当てはまります。サーバ 2 は *AppX* を処理していますが、サーバ 1 で実行中の *AppA* のバックアップサーバとして機能します。サーバ 3 とサーバ 4 の間には、同じタイプのアクティブ/アクティブの関係があります。

サーバ 1 とサーバ 2 の設定と、サーバ 3 とサーバ 4 の設定は似ていますが、大きな違いがあります。*AppA* と *AppX* のアプリケーションについて、サーバ 1 とサーバ 2 のみをグループ化できます。これらのサーバのみが、共有リソースにアクセスできます。

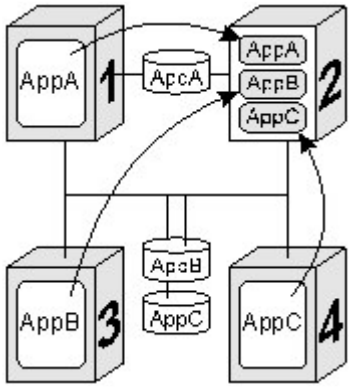
ただし、 *AppB* と *AppC* は、複数のグループ化オプションを持ちます。これは、4 台のサーバすべてが *AppB* と *AppC* の共有リソースにアクセスできるからです。 *AppB* と *AppC* は、第 3、第 4 のバックアップシステムとしてサーバ 1 や サーバ 2 にフェイルオーバするように設定することもできます。




**注記:** LifeKeeper はディスクレベルでロックを適用するので、 *AppB* と *AppC* のディスクリソースに接続する 4 つのシステムのうち、任意の時点でそれらにアクセスできるのは 1 つのみです。このため、サーバ 3 がアクティブに *AppB* を処理しているときには、サーバ 1、サーバ 2、および サーバ 4　は物理的に接続していても *AppB* のディスクリソースを使用できません。

# 5.4.1.3.4. アクティブ - スタンバイのグループ化

アクティブ/スタンバイのペア設定では、プライマリサーバは処理を実行し、バックアップサーバはプライマリサーバに障害が発生した場合に備えてスタンバイします。スタンバイシステムは小型でパフォーマンスの低いシステムでもかまいませんが、プライマリサーバに障害が発生した場合にリソースの可用性を確保できるだけの処理能力が必要です。




スタンバイサーバは、複数のアクティブサーバにバックアップを提供します。例えば、上の図では、3 つのアクティブ / スタンバイのリソースペアでサーバ 2 がスタンバイサーバです。LifeKeeper のリソース定義が、以下のアクティブ / スタンバイのペアの関係を指定します。

- サーバ *1* の *AppA* が サーバ *2* にフェイルオーバする。
- サーバ *3* の *AppB* が サーバ *2* にフェイルオーバする。
- サーバ *4* の *AppC* が サーバ *2* にフェイルオーバする。

複数のアクティブ / スタンバイグループを持つ設定を検討するときには、以下の 3 つの重要な設定概念を念頭に置いてください。

- **ディスクの所有権** 。複数の異なるアクティブなアプリケーションは、異なる複数のサーバから、同じ共有ディスクまた LUN にあるディスクパーティションを使用できません。LifeKeeper は、ディスクまたは LUN のレベルでロックを適用します。SCSI ロックが適用された場合、共有 SCSI バス上にあるシステム 1 台のみが、ディスクまたは LUN のパーティションにアクセスできます。このため、同一ディスク上の異なるパーティションにアクセスする複数のアプリケーションは、同一サーバ上でアクティブにする必要があります。この例では、サーバ 3 が *AppB* のディスクリソースを所有し、サーバ 4 が *AppC* のリソースを所有します。
- **処理能力** 。サーバ 1、サーバ 3、およびサーバ 4 に同時に障害が発生する可能性は非常に低いですが、複数のリソース関係をサポートするスタンバイサーバを指定するときには、複数の障害が発生した場合にスタンバイサーバが重要な処理のすべてを処理できるように注意する必要があります。
- **LifeKeeper の管理** 。この例では、サーバ 2 がその他 3 台のサーバをバックアップします。一般的に、LifeKeeper のデータベースを複数の論理グループで同時に管理することは望ましくありません。はじめに、予備システムと 1台のアクティブなシステムとの間でリソースを作成し、次に予備

システムと別のアクティブなシステムとの間、という手順を繰り返してリソースを作成する必要があります。

# 5.4.1.3.5. Intelligent スイッチバックと Automatic スイッチバックの違い

デフォルトでは、リソースのスイッチバック設定は *Intelligent* です。これは、そのリソースについて サーバ *A* から サーバ *B* にフェイルオーバが発生すると、別の障害が発生するか、管理者がリソースを別のサーバに *Intelligent* に 切り替えるまで、リソースは サーバ *B* に残ります。このため、 サーバ *A* が In Service に戻った後も、リソースは サーバ *B* で動作を続行します。この時点では、 サーバ *A* はリソースのバックアップとして機能します。

状況によっては、障害が発生したサーバが復旧したときに、リソースをそのサーバに自動でスイッチバックすることが望ましい場合があります。LifeKeeper には、前述したデフォルトの *Intelligent* スイッチバック 動作に代わる選択肢として、 *Automatic* スイッチバック オプションがあります。このオプションは、各サーバの個々のリソース階層に設定できます。特定のサーバ上にあるリソース階層に *Automatic* スイッチバック を選択し、そのサーバに障害が発生した場合、そのリソース階層はバックアップシステムにフェイルオーバします。障害が発生したサーバが復旧したときに、リソース階層は元のサーバに自動的にスイッチバックします。

**注記:**

- Automatic の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。
- LifeKeeper は、優先順位が上位のサーバから下位のサーバへの *Automatic* スイッチバック を実行しません。

# 5.4.1.3.6. syslog によるログの記録

LifeKeeper 8.0 から、標準の syslog 機能を使用してログの記録が行われます。LifeKeeper では、syslog の実装の1つである rsyslog をサポートしています。パッケージのインストール時には、すべての LifeKeeper ログメッセージに対して「local6」機能を使用するように syslog が設定されます（もし、local6 が既に使用されている場合は、他の local を使用します）。すべての LifeKeeper ログメッセージを */var/log/lifekeeper.log* に送信する LifeKeeper 固有のルーティングを含むように、 syslog 設定ファイル */etc/rsyslog.conf* が変更されます。(元の設定ファイルは、「~」で終わる同じ名前を使用してバックアップされます。)

> **重要:** LifeKeeper固有の設定を手動で編集しないでください。アップグレードやアンインストールが正しく機能しないことがあります。

この機能は、インストール後に /opt/LifeKeeper/bin にある lklogconfig ツールを使用して変更することができます。 例えば facility を local5 に変更する場合は、以下のコマンドを実行してください。

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

このツールの詳細については、LifeKeeper がインストールされているシステム上の lklogconfig(8) マニュアルページを参照してください。

Generic Applicationのリソーススクリプトが /opt/LifeKeeper/out/log に直接メッセージを送ると、LifeKeeperはデフォルトで */var/log/lifekeeper.log* にERRORレベルのログメッセージを送ります。 */etc/default/LifeKeeper* に **LOGMGR_LOGLEVEL=LK_INFO** パラメータを追加すれば、ERRORレベルをINFOレベルに変更することができます。

> **注意:** LifeKeeper がサーバから削除されると、LifeKeeper 固有の syslog 設定が削除されます。

> **注記:** ログローテンションをコントロールする LifeKeeper 設定ファイルは /etc/logrotate.d にあり、lifekeeper.log ファイルについては /etc/logrotate.d/lifekeeper 、 lifekeeper.err ファイルについては /etc/logrotate.d/lifekeeper-err で確認することができます。デフォルトの設定ではサイズが 100MB を越えるとローテーションされます。

# 5.4.1.3.7. リソース階層

LifeKeeper の GUI を使用すると、あるサーバにリソース階層を作成し、次にその階層を 1 台以上のバックアップサーバに拡張できます。その後、LifeKeeper により、指定したすべてのサーバに指定階層が自動作成されます。LifeKeeper は、各サーバのデータベースで階層情報を管理します。コマンドラインインターフェースを使用する場合は、各サーバの階層を明示的に指定する必要があります。

リソース階層の作成後、LifeKeeper が階層内のリソースの停止と開始を管理します。以下の関連トピックで、階層の指定作業の基本情報を説明しています。

---

リソースタイプ

リソースの状態

階層の関係

イクイバレンシ情報

リソース階層の情報

リソース階層の例

ステータスの詳細表示

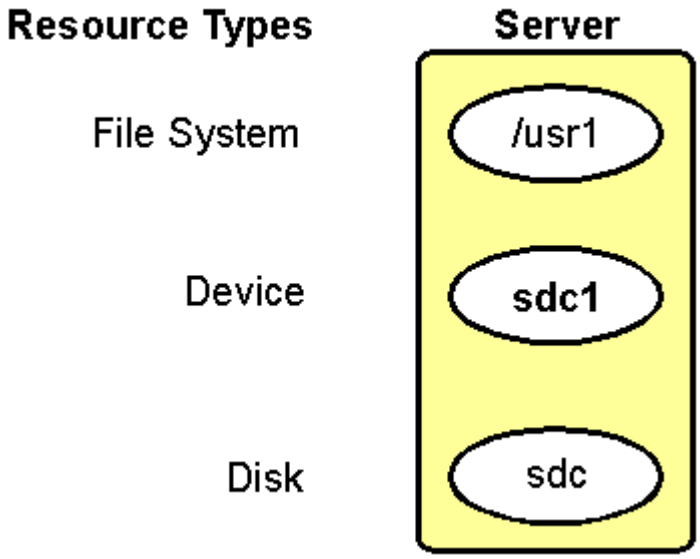
ステータスの簡略表示

# 5.4.1.3.7.1. リソースタイプ

リソースはハードウェアとソフトウェアのいずれかであり、リソースタイプ別に分類できます。LifeKeeper はファイルシステムと SCSI のリソースタイプに処理を提供し、リカバリキットは通信、RDBMS、その他のアプリケーションのリソースタイプに処理を提供します。

例えば、保護するファイルシステムの階層には、以下のタイプのリソースインスタンスが含まれます。

- **filesys** - Linux のファイルシステムリソースオブジェクトで、マウントポイントにより識別されます。
- **device** - SCSI ディスクパーティションと仮想ディスクで、デバイスファイル名で識別されます (例: *sdc1*)。
- **disk** - SCSI ディスクまたは RAID システム論理ユニットで、SCSI デバイス名で識別されます (例: *sd*)。

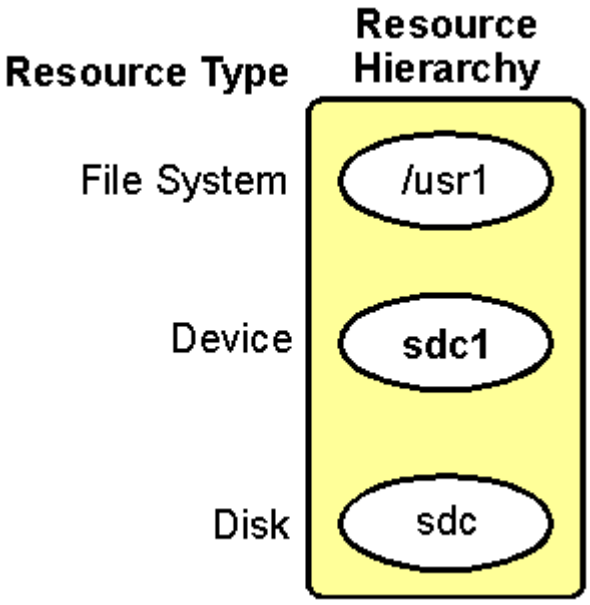


# 5.4.1.3.7.2. リソースの状態

| 状態 | 意味 |
|---|---|
| In Service、保護 (ISP) | リソースが動作可能です。LifeKeeper のローカルリカバリが正常に動作しています。LifeKeeper のサーバ間リカバリと障害検出が動作可能です。 |
| In Service、未保護 (ISU) | リソースは動作可能です。しかし、LifeKeeperのリソースヘルス監視は無効の状態であるため、ローカルリカバリーやフェイルオーバーは行われません。<br><br>**注記:** ファイルシステムリソース(filesys)で保護されたファイルシステムの使用容量が90%に（閾値のデフォルト）達した時、リソースステータスをISUにすることでそれを通知します。この場合、監視処理自体は継続されます。ファイルシステムリソースと使用容量の監視では、他のリソースタイプとは違って意味合いでISUステータスを使用しています。ファイルシステムの使用容量が一度閾値を下回れば、リソースステータスはISPに戻ります。 |
| Out of Service、障害 (OSF) | リソースが、障害により Out of Service になっています。リカバリは完了していないか、失敗しました。このリソースについて、LifeKeeper の警告機能は動作不能です。 |
| Out of Service、障害なし (OSU) | リソースは Out of Service ですが、別のサーバからリソースを引き継ぐことができます。 |
| 不正 (未定義) 状態 (ILLSTATE) | この状態は、リソースインスタンスについて状態が設定されていない場合に表示されます。通常の状況では、この不正状態が長く続くことはありません。ある状態から別の状態への移行が予測されます。LifeKeeper の情報テーブルがすべて更新される前 (LifeKeeper が初めて起動するときなど) にスイッチオーバが発生した場合に、この状態になります。 |

# 5.4.1.3.7.3. 階層の関係

LifeKeeper では、リソースインスタンス間の関係を作成できます。主な関係は依存関係で、例えばあるリソースインスタンスが動作するために、別のリソースインスタンスに依存します。リソースインスタンスと依存関係の組み合わせが、リソース階層です。



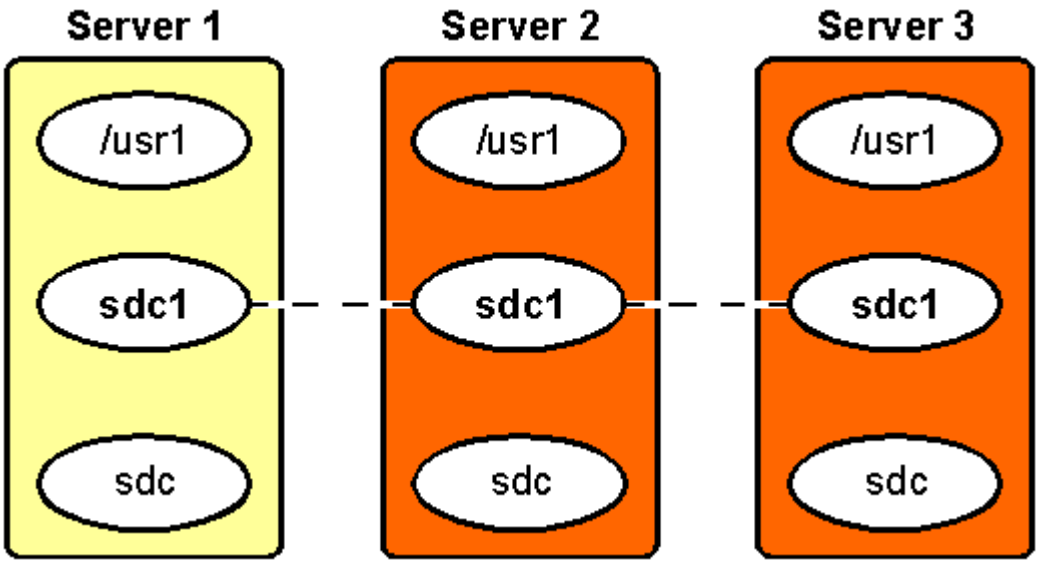

例えば、 */usr1* の動作はディスクサブシステムに依存するので、 */usr1* と、ディスクサブシステムを表すインスタンスとの間に順序付きの階層の関係を作成できます。

リソース階層により指定された依存関係は、リソースインスタンスを In Service と Out of Service にする適切な順序を LifeKeeper に示します。このリソース階層の例では、 *disk* と *device* のインスタンスを正常に In Service にするまで、LifeKeeper は */usr1* リソースを In Service にすることができません。

# 5.4.1.3.7.4. イクイバレンシ情報

LifeKeeper リソース階層を作成して拡張すると、そのリソース階層はプライマリサーバとセカンダリサーバの両方に存在します。ほとんどのリソースインスタンスは、1 台のサーバでのみ同時にアクティブにできます。このようなリソースについて、LifeKeeper は「イクイバレンシ情報」という第 2 の種類の関係を定義します。これにより、リソースがあるサーバで *In Service* になると、イクイバレンシ情報が定義されている他のサーバでは *Out of Service* になります。

以下の例に、各サーバのディスクパーティションのリソースインスタンス間のイクイバレンシ情報を示します。この例では、各リソースインスタンスが類似のイクイバレンシを持ちます。



# 5.4.1.3.7.5. リソース階層の情報

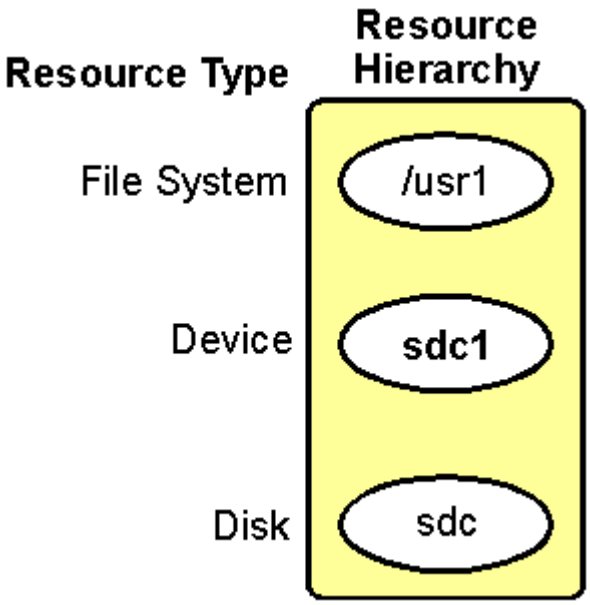
各リソースのステータスは、ステータスの詳細表示 とステータスの簡略表示 で表示されます。root リソースを表す LifeKeeper のタグ名は、[TAG] 列の左端から開始され、階層内のリソースのタグ名は適切にインデントされてリソース間の依存関係を表します。

以下の例は、ステータスの簡略表示のリソース階層セクションから取ったものです（デバイスとディスクの ID は、表示領域に収まるように切り詰められている）。

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

階層の図についてはリソース階層の例 のトピックを参照してください。詳細については、ステータスの詳細表示 とステータスの簡略表示 のトピックの「リソース階層の情報」セクションを参照してください。

# 5.4.1.3.7.6. リソース階層の例



# 5.4.1.3.7.7. ステータスの詳細表示

このトピックでは、 **lcdstatus** コマンドの出力例を使用してステータスの詳細表示で提供される情報のカテゴリについて説明します。この情報を表示する方法の詳細については、LCD (1M) のマニュアルページを参照してください。コマンドラインに、 **man lcdstatus** または **man LCD** を入力できます。LifeKeeper の GUI で使用できるステータス情報については、サーバーのステータスの表示 またはリソースのステータスの表示 を参照してください。

ステータスの詳細表示の例:

リソース階層の情報

```
Resource hierarchies for machine “wileecoyote”:

ROOT of RESOURCE HIERARCHY

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

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

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

reason=restore action has succeeded

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

Local priority = 1

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

FAILOVER ALLOWED

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

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

info=wileecoyote eth0 172.17.104.25 fffffc00

reason=restore action has succeeded

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

```
Local priority = 1

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

FAILOVER ALLOWED

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

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

info=wileecoyote eth0 172.17.106.10 fffffc00

reason=restore action has succeeded

these resources are dependent: apache-home.fred

Local priority = 1

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

FAILOVER ALLOWED

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

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

info=wileecoyote eth0 172.17.106.105 fffffc00

reason=restore action has succeeded

These resources are dependent: apache-home.fred

Local priority = 1

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

FAILOVER ALLOWED
```

通信ステータスの情報

```
The following LifeKeeper servers are known:

machine=wileecoyote state=ALIVE

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

The following LifeKeeper network connections exist:

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

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

LifeKeeper のフラグ

```
The following LifeKeeper flags are on:

shutdown_switchover
```

シャットダウンストラテジー

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

# リソース階層の情報

LifeKeeper は、リソースのステータスを root リソースから表示します。表示には、リソースのすべての依存関係についての情報が含まれます。

複数のリソースに共通する要素は、最初の root リソースの下に 1 回のみ表示されます。各リソース記述の第 1 行には、リソースタグとその後に続くコロン (:) が表示されます (例: `device13557:`)。階層内でリソースの記述に使用できる情報要素を以下に示します。

- **id** - LifeKeeper が使用する一意のリソース識別文字列。
- **app** - アプリケーションのタイプを示します。例えば、サンプルリソースは *Web* サーバアプリケーションです。
- **type** - リソースのクラスタイプを示します。例えば、サンプルリソースは *Apache* アプリケーションです。
- **state** - リソースの現在の状態。
  - ISP — ローカルで In Service であり、保護されています。

    - ISU — In Service であり、保護されていません。
    - OSF — Out of Service であり、障害が発生しています。
    - OSU — Out of Service であり、障害はありません。

- **initialize** - リソースの初期化方法を指定します。例えば、LifeKeeper はアプリケーションのリソースをリストアしますが、ホストアダプタは LifeKeeper なしで初期化します。

- **info** - オブジェクトの  remove と restore のスクリプトが使用する、オブジェクトに固有の情報があります。

- **reason** - 存在する場合、リソースが現在の状態にある原因を示します。例えば、あるアプリケーションが OSU の状態になった原因は、別のサーバでそのアプリケーションがIn Service (ISP または ISU) になったからです。共有リソースは、グループ内の 1 台のサーバでのみ同時にアクティブにできます。

- **depends on resources -** 存在する場合、このリソースが依存するリソースのタグ名がリストされます。

- **these resources are dependent -** このオブジェクトに直接依存するすべての親リソースのタグ名が示されます。

- **Local priority** - このリソースについて、ターゲットサーバのフェイルオーバの優先順位の値を示します。

- **SHARED equivalency** - このリソースが同等として定義されたリモートリソースのリソースタグとサーバ名、およびこのリソースについてのフェイルオーバの優先順位の値を示します。

- **FAILOVER ALLOWED** - 存在する場合、上の行で同等と指定されたリモートサーバで LifeKeeper が動作可能であること、およびアプリケーションが障害に対して保護されていることを示します。`FAILOVER INHIBITED` は、LifeKeeper がシャットダウンされているかリモートサーバが停止していることにより、アプリケーションが保護されていないことを示します。

## 通信ステータスの情報

ステータス表示のこのセクションには、LifeKeeper が認識しているサーバとその現在の状態、および各コミュニケーションパスの情報がリストされます。

これらの通信情報の要素は、ステータス表示にあります。

- **state** -コミュニケーションパスのステータス。通信ステータスの値は以下の値をとります。
    - ALIVE - 通常の動作中。
    - DEAD - 通常の動作をしていません。

- **priority** -コミュニケーションパスに割り当てられた優先順位の値。この項目は TCP パスについてのみ表示されます。

- **#comm_downs** - ポートに障害が発生してフェイルオーバが発生した回数。パスの障害によりフェ

イルオーバが発生するのは、障害発生時に「ALIVE」のコミュニケーションパスが他にない場合のみです。

さらに、ステータス表示では、TTY コミュニケーションパスについてのみ維持されている以下の統計値を提供できます。

- **wrpid** - 個々の TTY コミュニケーションパスが、一意の読み取りプロセスと書き込みプロセスを持ちます。wrpid フィールドには、書き込みプロセスのプロセス ID があります。書き込みプロセスは、以下の 2 つの条件のうちいずれかが発生するまでスリープ状態です。
  - ハートビートタイマの期限が切れ、書き込みプロセスにメッセージを送信させる。
  - ローカルプロセスが、LifeKeeper のメンテナンスメッセージを他のサーバに送信するように書き込みプロセスに要求する。書き込みプロセスは、関連付けられた TTY ポートを使用して、メッセージを他のシステムの TTY ポート上にある読み取りプロセスに送信します。

- **rdpid** - 個々の TTY コミュニケーションパスが、一意の読み取りプロセスと書き込みプロセスを持ちます。rdpid フィールドには、読み取りプロセスのプロセス ID があります。読み取りプロセスは、以下の 2 つの条件のうちいずれかが発生するまでスリープ状態です。
  - ハートビートタイマの期限が切れ、定義済みのハートビート間隔が期限切れになったかどうかを読み取りプロセスが判断する必要がある場合。期限切れの場合、読み取りプロセスはコミュニケーションパスに DEAD 状態のマークを付けます。これにより、ALIVE とマークされた他のコミュニケーションパスがない場合はフェイルオーバイベントが開始されます。
  - リモートシステムの書き込みプロセスが LifeKeeper のメンテナンスメッセージを送信し、読み取りプロセスがメッセージの受信に必要なプロトコルを実行します。

- **#NAKs** - 書き込みプロセスが negative acknowledgment （NAK）を受信した回数。NAK メッセージは、他のシステム上にある読み取りプロセスが、書き込みプロセスが送信したメッセージを受け取らず、書き込みプロセスがメッセージパケットを再送信する必要があったことを意味します。#NAKs の統計値は、回線ノイズに起因して、長期間にわたって集計できます。ただし、急激に数値が増加した場合、通信サブシステムで診断手順を実行する必要があります。

- **#chksumerr** - サーバ間のチェックサムメッセージが一致しなかった回数。この統計値は、回線ノイズに起因して、長期間にわたって集計できます。ただし、急激に数値が増加した場合、通信サブシステムで診断手順を実行する必要があります。

- **#incmpltmes** - 受信メッセージパケットが予測サイズに一致しなかった回数。不一致の回数が多い場合、コミュニケーションパスに関連付けられたハードウェアポートで診断手順の実行が必要な可能性があります。

- **#noreply** - 書き込みプロセスが肯定応答の待機中にタイムアウトし、メッセージを再送信しなければならなかった回数。肯定応答がない場合、サーバの過負荷、またはサーバの障害を意味することがあります。

- **#pacresent** - 読み取りプロセスが同一パケットを受診した回数。これは、送信サーバの書き込みプロセスがタイムアウトし、同一メッセージを再送信する場合に発生することがあります。

- **#pacoutseq** - 読み取りプロセスが、順序が不正のパケットを受診した回数。このフィールドの値が大きい場合、メッセージパケットの脱落を示すことがあり、通信サブシステムで診断手順の実行が必要な可能性があります。

- **#maxretrys** - 特定のメッセージについて、再送信の最大回数を超えたときに増加する指標 (NAK と noreply のメッセージ)。#maxretrys フィールドの値が大きい場合、通信サブシステムで診断手順を実行する必要があります。

# LifeKeeper のフラグ

ステータスの詳細表示の後部近くに、システムのフラグセットがあります。共通タイプは、プロセスのロックが動作を完了するまで他のプロセスを確実に待機させるために使用する LCD のロックフラグです。LCD のロックの標準フォーマットは以下のとおりです。

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

一般的な LCD のロックフラグの例を示します。

- `!action!02833!701236710!server1:` ***filesys*** 。ファイルシステム階層を作成すると、このフォーマットでステータス表示にフラグが生成されます。 ***filesys*** の指定は、他のアプリケーションリソース階層では別のリソースタイプである場合も、一般的なアプリケーションやユーザ定義アプリケーションでは　app　である場合もあります。

- 他の代表的なフラグとして、`!nofailover!`*`machine`*、`!notarmode!`*`machine`*、`shutdown_switchover` などがあります。`!nofailover!`*`machine`*と `!notarmode!`*`machine`* のフラグは、LifeKeeper が作成と削除を行う内部の一時フラグで、サーバのフェイルオーバを制御します。`shutdown_switchover` フラグは、このサーバのシャットダウンストラテジーが ***switchover*** に設定されたことを示し、サーバのシャットダウンによりスイッチオーバが発生します。使用可能なフラグの詳細については、依存関係の作成方法については、LCDI-flag (1M) を参照してください。

# シャットダウンストラテジー

ステータスの詳細表示の最後の項目は、このシステム用に選択された LifeKeeper のシャットダウンストラテジーを示します。詳細については、サーバのシャットダウンストラテジーの設定 を参照してください。

# 5.4.1.3.7.8. ステータスの簡略表示

このトピックでは、 **lcdstatus -e** コマンドの出力例を使用して、ステータスの簡略表示で提供される情報のカテゴリについて説明します。この情報を表示する方法の詳細については、LCD (1M) のマニュアルページを参照してください。コマンドラインに、 **man lcdstatus** または **man LCD** を入力できます。LifeKeeper の GUI で使用できるステータス情報については、サーバーのステータスの表示 またはリソースのステータスの表示 を参照してください。

ステータスの簡略表示の例（lcdstatus -e コマンドの出力例）：

リソース階層の情報

```
BACKUP   TAG                 ID          STATE     PRIO  PRIMARY
svr1     appfs3910-on-svr1   appfs4238   ISP          1  svr2
svr1      filesys4083        /jrl1       ISP          1  svr2
svr1       device2126        000…300-1   ISP          1  svr2
svr1        disk2083         000…300     ISP          1  svr2
```

通信ステータスの情報

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

## リソース階層の情報

LifeKeeper は、各リソースのステータスを表示します。root リソースを表す LifeKeeper のタグ名は、[TAG] 列の左端から開始され、階層内のリソースのタグ名は適切にインデントされてリソース間の依存関係を表します。

**BACKUP** 列は、フェイルオーバの優先順序内で、このステータス表示の対象システムの次にあるシステムを示します。指定したリソースについて、ターゲットシステムが優先順位の最も低いシステムである場合、そのリソースの **BACKUP** 列にはダッシュ (------) が表示されます。

- **TAG 列** - リソースの root タグがあります。
- **ID 列** -各リソースの識別文字列があります。
- **STATE 列** -各リソースの現在の状態があります (リソースの状態 を参照)。
- **PRIO 列** -各リソースについて、ローカルサーバのフェイルオーバの優先順位の値があります。

- **PRIMARY** 列 -各リソースについて、優先順位が最高のサーバ名があります。

## 通信ステータスの情報

表示のこのセクションには、ターゲットシステムで定義された各コミュニケーションパスのリストがあります。各パスについて、以下の情報が表示されます。

- **MACHINE** - コミュニケーションパスのリモートサーバ名。
- **NETWORK** - コミュニケーションパスのタイプ (TCP または TTY)。
- **ADDRESSES/DEVICE** - コミュニケーションパスの IP アドレスまたはデバイス名のペア。
- **STATE** - コミュニケーションパスの状態 (ALIVE または DEAD)。
- **PRIO** - TCP パスの場合、パスに割り当てられた優先順位。TTY パスの場合、優先順位が割り当てられていないので、この列にはダッシュ (----) が表示されます。

# 5.4.1.4. 障害検出とリカバリのシナリオ

障害検出とリカバリを実行するために、LifeKeeper のさまざまなコンポーネントがどのように連携しているかを調べるには、3 つのタイプのリカバリシナリオを説明する以下のトピックを参照してください。

---

IP ローカルリカバリ

リソースのエラーリカバリのシナリオ

サーバの障害リカバリのシナリオ

# 5.4.1.4.1. IP ローカルリカバリ

## ローカルリカバリーのシナリオ

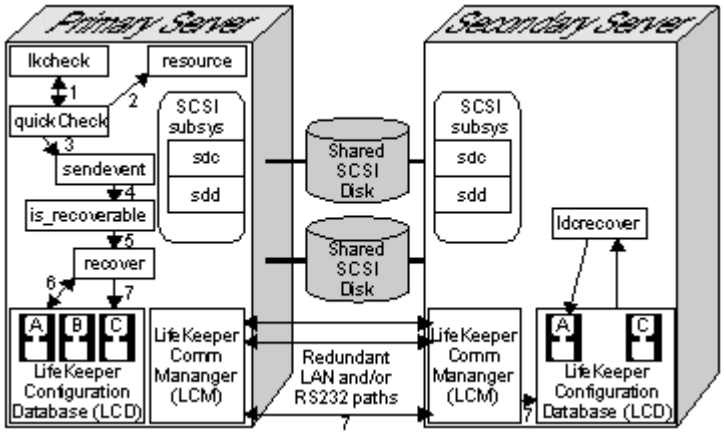
IP Recovery Kit によって IP アドレスの障害が検出されると、結果として生じる障害によって remove および restore を実行する **IP ローカルリカバリースクリプト** が起動されます。

> SIOS では、バックアップインターフェースが必要な場合、すべての LifeKeeper リリースに含まれる標準 Linux の NIC ボンディングメカニズムを使用してボンディングしたインターフェースを使用することを推奨しています。

# 5.4.1.4.2. リソースのエラーリカバリのシナリオ

LifeKeeper は、LifeKeeper が保護するリソースのステータスと健全性をチェックするリアルタイムデーモンモニタ **lkcheck** を装備しています。 In Service の各リソースについて、 **lkcheck** が定期的にそのリソースタイプの quickCheck スクリプトを呼び出します。 **quickCheck** スクリプトがリソースのクイック健全性チェックを実行し、リソースが障害のある状態にあると判断すると、 **quickCheck** スクリプトはイベント通知メカニズム sendevent を呼び出します。

以下の図に、 **lkcheck** がプロセスを開始したときのリカバリプロセスの作業を示します。




1. **lkcheck** が実行されます。デフォルトでは、 **lkcheck** プロセスは 2 分ごとに実行されます。 **lkcheck** が動作すると、システムで In Service の各リソースについて適切は **quickCheck** スクリプトを呼び出します。

2. **quickCheck** スクリプトがリソースをチェックします。 **quickCheck** スクリプトが実行するチェックの内容は、各リソースタイプによって異なります。通常、スクリプトは、リソースのクライアントの動作をシミュレートして、予測した応答を受信するかどうかを確認することにより、目的の作業を実行するためにリソースが使用可能かどうかを単純に確認します。

3. **quickCheck** スクリプトが **sendevent** を呼び出します。 **quickCheck** スクリプトが、リソースが障害のある状態にあると判断した場合、 **sendevent** を呼び出して、適切なクラスとタイプを持つイベントを開始します。

4. リカバリ手順の検索。システムイベント通知メカニズム **sendevent** は、はじめに、イベントタイプまたはコンポーネントについて、LCD がリソースまたはリカバリを持つかどうかを判断しようとします。この判断を行うために、is_recoverable プロセスは LCD のリソース階層をスキャンして、イベントに対応するリソースインスタンス (この例では filesys　の名前) を検索します。

   次の手順の動作は、スキャンでリソースレベルのリカバリ手順が検出されたかどうかによって異なります。

- ◦ 検出されない場合。リカバリ手順が見つからない場合、is_recoverable は **sendevent** に戻り、 **sendevent** は基本イベント通知を続行します。

- ◦ 検出された場合。スキャンでリソースが検出された場合、is_recoverable はリカバリプロセスをバックグラウンドに運びます。is_recoverable プロセスが戻り、 **sendevent** が基本イベント通知を続行します。推奨フラグ「 *-A* 」を基本警告イベント応答スクリプトに渡し、LifeKeeper がリカバリを実行することを示します。

5. リカバリプロセスが開始されます。リカバリが続行していると仮定して、is_recoverable はリカバリプロセスを開始し、はじめにローカルリカバリを試行します。

6. ローカルリカバリが試行されます。インスタンスが検出された場合、リカバリプロセスは、LCD 内のリソース階層にアクセスし、階層ツリーからイベントに応答する方法を知っているリソースを検索して、ローカルリカバリを試行します。各リソースタイプについて、イベントクラスにちなむ名前を持つサブディレクトリ (そのイベントタイプのリカバリスクリプトを持つ) を含むリカバリサブディレクトリを検索します。

   リカバリプロセスが、リソース階層で障害が発生しているリソースから上方向に最も離れたリソースに関連付けられている、リカバリスクリプトを実行します。リカバリスクリプトが正常に完了した場合、リカバリは停止します。スクリプトが失敗した場合、次のリソースに関連付けられたスクリプトが実行され、リカバリスクリプトが正常に完了するか、障害が発生したインスタンスに関連付けられたリカバリスクリプトが試行されるまで続行されます。

   *ローカルリカバリが正常に完了した場合、リカバリは停止します。*

7. サーバ間のリカバリが開始されます。ローカルリカバリに失敗した場合、イベントはサーバ間のリカバリにエスカレートします。

8. リカバリが続行されます。ローカルリカバリに失敗しているので、リカバリプロセスは失敗したインスタンスを *Out-of-Service-FAILED* (OSF) 状態にマークし、この失敗したリソースに依存するすべてのリソースを *Out-of-Service-UNIMPAIRED* (OSU) 状態にマークします。リカバリプロセスは次に、障害が発生したリソース、または障害が発生したリソースに依存するリソースが、他のシステム上にあるリソースとイクイバレンシー情報を持っているかどうかを判断し、優先順位が最高の動作可能なサーバを選択します。同時にアクティブにできるイクイバレンシー情報を持つリソースは 1 つのみです。

   *イクイバレンシー情報が存在しない場合、リカバリプロセスは停止します。*

   イクイバレンシー情報が検出されて選択された場合、LifeKeeper はサーバ間のリカバリを開始します。リカバリプロセスが LCM 経由で、イクイバレンシー情報を持つリソースを持つ選択されたバックアップシステムの LCD プロセスにメッセージを送信します。これは、LifeKeeper がサーバ間のリカバリを試行することを意味します。

9. **lcdrecover** プロセスが転送を調整します。バックアップサーバの LCD プロセスが **lcdrecover** プロセスを運び、同等リソースの転送を調整します。

10. バックアップサーバのアクティブ化。 **lcdrecover** プロセスが同等のリソースを検出し、そのリソースが Out of Service のリソースに依存しているかどうかを判断します。 **lcdrecover** が、必要な各リソースについて restore スクリプト (リソースリカバリ動作スクリプトの一部) を実行し、リソースを In Service にします。

バックアップサーバでリソースをリストアすることにより、プライマリシステムからより多くの共有リソースを転送することが必要になる場合があります。プライマリシステムとの間で、プライマリサーバ上でのサービスから削除する必要があるリソースを示すメッセージが送受信され、次に選択したバックアップサーバで In Service になり、重要なアプリケーションのすべての機能が提供されます。この動作は、転送する追加の共有リソースがなくなり、バックアップで必要なすべてのリソースインスタンスがリストアされるまで続行されます。

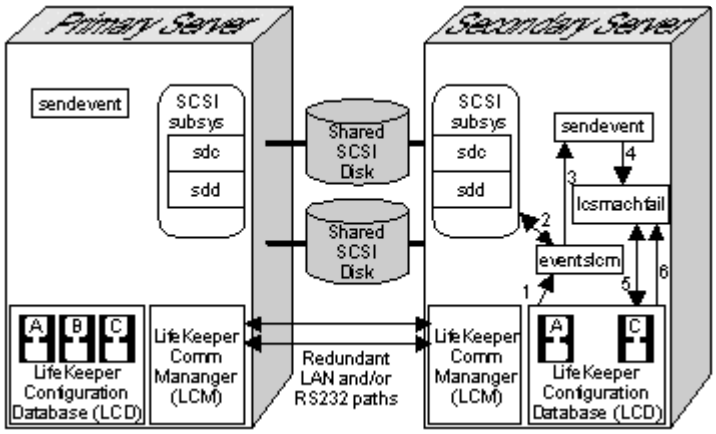
# 5.4.1.4.3. サーバの障害リカバリのシナリオ

LifeKeeper Communications Manager (LCM) には、2 つの機能があります。

- メッセージング。LCM は、LifeKeeper がリカバリ、設定、または監査の実行を行うときに送信するメッセージの経路として機能します。
- 障害検出。また、LCM には、サーバに障害が発生しているかどうかを検出する役割もあります。

LifeKeeper には、構成内の各サーバに、ペアのサーバが動作していることを定期的に通知する組み込みのハートビート信号があります。あるサーバが、いずれかのコミュニケーションパス経由でハートビートメッセージを受信しなかった場合、LifeKeeper はそのパスを DEAD としてマークします。

以下の図に、LCM ハートビートメカニズムがサーバの障害を検出したときのリカバリ作業を示します。




以下の手順では、上の図で、LifeKeeper があるサーバのすべての通信接続を DEAD としてマークした場合のリカバリシナリオを説明します。

1. LCM が **eventslcm** を起動します。LifeKeeper がすべてのコミュニケーションパスを DEAD としてマークすると、LCM は **eventslcm** プロセスを開始します。

   **eventslcm** プロセスを停止する活動は 1 つのみです。

   - ◦ コミュニケーションパスが、アクティブである。いずれかのコミュニケーションパスがハートビート信号の送信を再開した場合、LCM は **eventslcm** プロセスを停止します。

   通信障害に起因するフェイルオーバやシステムの障害を防止するために、各ペアのサーバ間に、物理的に独立した冗長なコミュニケーションパスを 2 つ以上設定することが重要です。

2. sendevent へのメッセージ送信。 **eventslcm** は、 **sendevent** をイベントタイプ *machfail* で呼び出してシステム障害アラームを送信します。

3. sendevent がフェイルオーバリカバリを開始します。sendevent プログラムが、LifeKeeper がシステム障害イベントを処理できることを判断し、LifeKeeper フェイルオーバリカバリプロセス **lcdmachfail** を実行します。

4. **lcdmachfail** のチェック。 **lcdmachfail** プロセスがはじめに、応答していないサーバがシャットダウンしていないことを確認します。システム障害の発生前に、他のシステムが正常にシャットダウンしている場合、フェイルオーバは禁止されます。次に、 **lcdmachfail** は、障害が発生したシステムと同等の共有を持つリソースをすべて特定します。これが、リカバリの関与ポイントです。

5. **lcdmachfail** がリソースをリストアします。 **lcdmachfail** が、障害が発生したシステムと同等のイクイバレンシ情報を持つ、バックアップサーバ上のリソースをすべて特定します。また、バックアップサーバが該当するリソースが構成されている、優先順位が最高のアクティブなサーバであるかどうかを判断します。すべてのバックアップサーバがこのチェックを実行するので、1 台のサーバのみが階層のリカバリを試行します。このチェックに合格した個々の同等リソースについて、 **lcdmachfail** が、関連付けられたリストアプログラムを呼び出します。次に、 **lcdmachfail** は、リストアしたリソースに依存する各リソースもリストアします。これは、バックアップサーバ上の階層全体がサービス中になるまで続行されます。

# 5.4.2. インストールと設定

## SPS for Linux のインストール

SPS for Linux ソフトウェアのインストールに関する詳細な手順については、SPS for Linux インストールガイド を参照してください。その他の情報については、 SPS for Linux リリースノート を参照してください。

## SPS for Linux の設定

SPS 環境がインストールされると、クラスタ内の各サーバ上で SPS ソフトウェアを設定することができます。以下の **SPS 設定手順** トピックに記載の手順に従ってください。ここには、詳細情報を含む各トピックへのリンクが記載されています。

---

SPS の設定手順

SNMP による LifeKeeper イベント転送

LifeKeeper イベントメール通知

任意の設定作業

Linux の設定

データレプリケーションの設定

ネットワーク設定

アプリケーションの設定

ストレージとアダプタの設定

LifeKeeper I-O フェンシングの概要

リソースポリシー管理

認証情報の設定

# 5.4.2.1. SPS の設定手順

SPS インストールガイドの説明に従って SPS 環境をインストールした場合、クラスタ内の各サーバーで SPS ソフトウェアを起動し、設定することができます。

詳細を説明するトピックへのリンクを含む以下の手順を実行してください。以下の手順は、クラスタ内の各サーバーで実行します。

1. 次のコマンドを root として実行して LifeKeeper を起動します。
   ```
   $LKROOT/bin/lkcli start
   ```

   このコマンドによって、管理対象のサーバー上のまだ起動していないすべての LifeKeeper デーモンプロセスを起動します。

   LifeKeeper の起動および停止の詳細については、LifeKeeper の起動 および LifeKeeper の停止 を参照してください。

2. TTY 通信接続をセットアップ します。LifeKeeper のハートビート用に TTY 通信 パスを利用する場合は、ハートビート用の物理的な接続をセットアップする必要があります。

3. GUI を設定します。GUI の設定には多くのタスクが含まれます。GUI の実行準備 のLifeKeeper GUI - 概要 トピックから始めてください。詳細な手順については、GUIの実行準備 に網羅されたリンクの順番に従ってください。

   **注記:** LifeKeeper GUI を初めて実行すると、QuickStart ボタンが表示され、これを押すと LifeKeeper のリソースの設定を案内する手順とリンクを含むウィンドウが開きます。QuickStart Configuration Assistant は [Help] メニュー からいつでもアクセスできます。

4. コミュニケーションパスを作成 します。LifeKeeper の保護を有効にする前に、コミュニケーションパス (ハートビート) の定義を作成する必要があります。

5. 以下の設定作業を任意で実行します。

   - サーバーのシャットダウンストラテジーの設定
   - 手動フェイルオーバ確認オプションの設定
   - LifeKeeper ハートビートの調整
   - SNMP イベント転送の設定
   - イベントメール通知の設定
   - クラスタで STONITH デバイスを使用する場合は、STONITH デバイスを制御するスクリプトを作成し、LifeKeeper の適切なイベントディレクトリに配置します。

6. SPS でアプリケーションを保護する準備ができました。以降の手順は、オプションの SPS

Recovery Kit のいずれかを使用するかどうかによって異なります。

- SPS Recovery Kit を使用する場合、リソース階層の作成と拡張の手順については Recovery Kit に関連するドキュメンテーションを参照してください。
- 関連する Recovery Kit がないアプリケーションを使用する場合、2 通りの選択肢があります。
    - シンプルなアプリケーションの場合、アプリケーションと LifeKeeper との間のインターフェースの作成方法を慎重に検討してください。LifeKeeper Core に含まれる Generic Application Recovery Kit を使用して保護することもできます。
    - OS が提供しているサービスの場合、LifeKeeper Core に含まれる Quick Service Protection (QSP) Recovery Kit を使用して容易にリソースが作成できます。ただし、quickCheck では簡易なチェックしかしていませんので、注意が必要です。

# 5.4.2.1.1. TTY 接続のセットアップ

LifeKeeper のハートビート用に TTY コミュニケーションパスを利用する場合は、ハートビート用の物理的な接続をセットアップする必要があります。単一の通信障害による誤ったフェイルオーバを抑止するためには、複数のコミュニケーションパスが必要です。2 つ以上の LAN ベース (TCP) のコミュニケーションパスも使用する必要があります。

使用する各サーバのシリアルポートにシリアルハートビート用の TTY ケーブルを接続します。

1. 次のコマンドを実行してシリアルパスをテストします。

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

   ここで、

   - ◦ **baud** は、シリアルパス用に選択したボーレート (通常は 9600)
   - ◦ **port** は、サーバ 1 でテスト中のシリアルポート。例えば、 */dev/ttyS0*
     - これでサーバ 1 は、サーバ 2 からの入力を待っている状態です。

2. サーバ 2 で **portio** コマンドを実行します。ペアの 2 番目のシステムで次のコマンドを実行します。

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

   ここで、

   - ◦ **baud** は、サーバ 1 に合わせて選択した同じボーレート
   - ◦ **port** は、サーバ 2 でテスト中のシリアルポート。例えば、 */dev/ttyS0* 。

3. コンソールを確認します。コミュニケーションパスが正常に動作する場合、サーバ 1 のコンソールには「Helloworld」が表示されます。表示されない場合は、診断、修正作業を終えてから LifeKeeper の設定を続けてください。

# 5.4.2.2. SNMP による LifeKeeper イベント転送

SNMP による LifeKeeper イベント転送の概要

LifeKeeper イベント転送の設定

SNMP のトラブルシューティング

# 5.4.2.2.1. SNMP による LifeKeeper イベント転送の概要

SNMP (Simple Network Management Protocol) は、ネットワークを管理するための、デバイスに依存しないフレームワークです。ネットワーク上のデバイスは、デバイスのベンダーが提供する MIB (Management Information Base) 変数によって記述されます。ネットワーク内の各ノード上では SNMP エージェントが実行され、ネットワークマネージャノードと通信を行います。ネットワークマネージャは、エージェントに対するクエリで MIB 変数の値を取得、設定することにより、エージェントノードを監視、制御します。エージェントは、トラップと呼ばれるメッセージを非同期に生成して例外イベントの発生をマネージャにしらせることもできます。SNMP (Simple Network Management Protocol) を使用してネットワークを監視および管理するアプリケーションは多数提供されています。

LifeKeeper のイベント通知機能では、特定のイベントが起きたときに通知を受信するアプリケーションを登録することができます (sendevent(5) マニュアルページを参照)。LifeKeeper は、LifeKeeper の動作を監視するサードパーティのネットワーク管理コンソールに向けて LifeKeeper の重要なイベントに関する SNMP トラップ通知を送信するように簡単に設定できます。

SNMP トラップを受信するリモート管理コンソールは、最初にそのシステムの管理用ソフトウェアを使用して設定する必要があります。LifeKeeper は、外部の SNMP の設定機能を提供していません。リモート管理サーバは、通常、LifeKeeper クラスタの外側に配置されます (つまり LifeKeeper のノードではありません)

## LifeKeeper イベントテーブル

以下の表では、LifeKeeper のイベントのリストと関連付けられているトラップ番号を示しています。オブジェクト ID (OID) は、プリフィックスとそれに続く個別のトラップ番号から、次のフォーマットで構成されます。

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

プリフィックスは「.1.3.6.1.4.1.7359」であり、これは MIB ツリーで **iso.org.dod.internet.private.enterprises.7359** に展開されます(7359 は、SteelEye (SIOS Technology) の企業番号です。LifeKeeper を表す「1」をこれに続けます)。例えば、「LifeKeeper Startup Complete」イベントは次の OID を生成します: **.1.3.6.1.4.1.7359.1.0.100** 。

| **LifeKeeper** イベント/説明 | トラップ番号 | オブジェクト **ID** |
|---|---|---|
| **LifeKeeper Startup Complete**<br><br>LifeKeeper が起動したノードから送信されます | 100 | .1.3.6.1.4.1.7359.1.0.100 |

| | | |
|---|---|---|
| **LifeKeeper Shutdown Initiated**<br><br>LifeKeeper のシャットダウンを開始したノードから送信されます | 101 | .1.3.6.1.4.1.7359.1.0.101 |
| **LifeKeeper Shutdown Complete**<br><br>LifeKeeper のシャットダウンを完了したノードから送信されます | 102 | .1.3.6.1.4.1.7359.1.0.102 |
| **LifeKeeper Manual Switchover Initiated on Server**<br><br>手動スイッチオーバを要求したノードから送信されます | 110 | .1.3.6.1.4.1.7359.1.0.110 |
| **LifeKeeper Manual Switchover Complete – recovered list**<br><br>手動スイッチオーバを完了したノードから送信されます | 111 | .1.3.6.1.4.1.7359.1.0.111 |
| **LifeKeeper Manual Switchover Complete – failed list**<br><br>手動スイッチオーバに失敗したクラスタ内の各ノードから送信されます | 112 | .1.3.6.1.4.1.7359.1.0.112 |
| **LifeKeeper Node Failure Detected for Server**<br><br>クラスタ内のノードに障害が発生したときに、クラスタ内の各ノードから送信されます | 120 | .1.3.6.1.4.1.7359.1.0.120 |
| **LifeKeeper Node Recovery Complete for Server – recovered list**<br><br>障害ノードからのリソースをリカバリしたクラスタ内の各ノードから送信されます | 121 | .1.3.6.1.4.1.7359.1.0.121 |
| **LifeKeeper Node Recovery Complete for Server – failed list**<br><br>障害ノードからのリソースのリカバリに失敗したクラスタ内の各ノードから送信されます | 122 | .1.3.6.1.4.1.7359.1.0.122 |
| **LifeKeeper Resource Recovery Initiated** | 130 | .1.3.6.1.4.1.7359.1.0.130 |

| | | |
|---|---|---|
| リソースをリカバリしているノードから送信されます。リカバリが完了したか失敗したかを示す 131 または 132 トラップが必ずこれに続きます。 | | |
| **LifeKeeper Resource Recovery Failed**<br><br>リカバリを試みたリソースを稼働できなかったときに、トラップ 130 を送信したノードから送信されます | 131* | .1.3.6.1.4.1.7359.1.0.131 |
| **LifeKeeper Resource Recovery Complete**<br><br>リソースのリカバリが完了したときに、トラップ 130 を送信したノードから送信されます | 132 | .1.3.6.1.4.1.7359.1.0.132 |
| **LifeKeeper Communications Path Up**<br><br>ノードへのコミュニケーションパスが確立されました | 140 | .1.3.6.1.4.1.7359.1.0.140 |
| **LifeKeeper Communications Path Down**<br><br>ノードへのコミュニケーションパスがダウンしました | 141 | .1.3.6.1.4.1.7359.1.0.141 |
| **LifeKeeper <Node monitoring> Failure**<br><br>Standby Node Health Check における Node監視で異常を検知したときに、異常を検知したノードから送信されます。<Node monitoring> には、検知した異常の内容が記載されます。 | 190 | .1.3.6.1.4.1.7359.1.0.190 |
| **LifeKeeper <OSUquickCheck> Failure**<br><br>Standby Node Health Check における OSUリソース監視で異常を検知したときに、異常を検知したノードから送信されます。<br><OSUquickCheck> には、異常が検知されたリソースのタグ名が記載されます。 | 200 | .1.3.6.1.4.1.7359.1.0.200 |
| **トラップ PDU に追加情報を含めるために以下の変数が使用されます。:** | | |
| Trap message | すべての | .1.3.6.1.4.1.7359.1.1 |

| | トラップ | |
|---|---|---|
| Resource Tag | 130 | .1.3.6.1.4.1.7359.1.2 |
| Resource Tag | 131 | .1.3.6.1.4.1.7359.1.2 |
| Resource Tag | 132 | .1.3.6.1.4.1.7359.1.2 |
| List of recovered resources | 111 | .1.3.6.1.4.1.7359.1.3 |
| List of recovered resources | 121 | .1.3.6.1.4.1.7359.1.3 |
| List of failed resources | 112 | .1.3.6.1.4.1.7359.1.4 |
| List of failed resources | 122 | .1.3.6.1.4.1.7359.1.4 |

* 複数のバックアップサーバでリカバリに失敗すると、このトラップは複数回表示されることがあります。

# 5.4.2.2.2. LifeKeeper イベント転送の設定

## 前提条件

SNMP のイベント転送機能は、LifeKeeper Core の機能の一部として含まれているので、LifeKeeper の追加パッケージをインストールする必要はありません。LifeKeeperはトラップを発行するためにsnmptrapコマンドを使用しているので、SNMP通知機能を使用するノードにsnmptrapコマンドをインストールしている必要があります。
snmptrapコマンドは以下のパッケージにて提供されています。
RHEL5以降および互換OS - net-snmp-utils
SLES11以降 - net-snmp

以前のバージョン (4.1 以前) の snmp の実装では、`defCommunity` ディレクティブがサポートされていないため、トラップは「public」コミュニティストリングを使用して送信されます。

LifeKeeper のノードで SNMP エージェント snmpd を起動しておく必要はありません。

ネットワーク管理コンソール上のトラップハンドラおよびトラップメッセージに対するハンドラの応答に関する設定は、LifeKeeper の本機能が提供する範囲ではありません。必要な手順については、お使いのシステム管理ツールが提供するドキュメンテーションを参照してください。

## 設定作業

LifeKeeper SNMP イベント転送を設定するには、以下の作業を実施します。SNMP トラップを生成する LifeKeeper クラスタの各ノードにおいて、最後の手順以外のすべてを 繰り返す必要があります。

1. 上述の `snmptrap` ユーティリティが利用できることを確認します。

2. SNMP トラップを受信するネットワーク管理ノードを指定します。指定するには、コマンドラインを使用するか、`/etc/default/LifeKeeper` ファイルを編集します。DNS の問題に影響されないように、ドメイン名ではなく IP アドレスを指定してください。

   - ◦ コマンドラインからは、`lk_configsnmp` を使用します (詳細については、`lk_configsnmp(1M)` のマニュアルページを参照)。このユーティリティでは、IP アドレスのみ使用できます。

   - ◦ または、デフォルトファイル`/etc/default/LifeKeeper` を編集して IP アドレスを追加します。`LK_TRAP_MGR=` エントリを見つけて「=」の右側に IP アドレスを１つまたは複数（カンマ区切り）入力します (「=」の前後やカンマの前後にはスペースを入れません)。

3. `defCommunity` をサポートとしない SNMP 実装の以前のバージョンをお使いの場合は、このステップを飛ばしてください。トラップは「public」コミュニティストリングを使用して送信されます。新しいバージョンの場合は次の手順を実行します。

   `/usr/share/snmp/snmp.conf` にデフォルトのコミュニティを指定してください。このファイルが存在しない場合は、十分な制限付きの権限で作成します。ディレクティブ

defCommunity を値と共に追加します。これにより、トラップの送信時に SNMP バージョン 2c のコミュニティストリングが指定されます。例えば、以下のような行を追加します。

```
defCommunity myCommunityString
```

この設定ファイルの詳細については、snmp.conf マニュアルページを参照してください。

4. リモート管理コンソール上で、LifeKeeper のイベントから送られて来るトラップ OID を検出し応答するために必要な設定手順をすべて実行します。管理ノードが Linux サーバの場合、この機能の検証を開始するために最低限必要なことは、-f -Lo オプション (メッセージを stdout に出力) を指定して snmptrapd デーモンを開始することです。

## 設定の確認

設定が正常に動作することを確認するには、LifeKeeper の処理を実行します (例えば、LifeKeeper を開始または停止する、または LifeKeeper GUI を使用して、あるリソースを手動で in service にするなど)。管理コンソールでトラップメッセージを受信していることを確認します。トラップを受信していない場合は、管理システムの適切なログファイルを調査し、管理ソフトウェアが提供する標準のトラブルシューティング手順を実行してください。LifeKeeper のログを調べると、トラップメッセージの送信に問題があるかどうかを判断することができます。詳細については、SNMP のトラブルシューティング を参照してください。

## SNMP イベント転送の無効化

LifeKeeper による SNMP トラップの生成を無効にするには、ファイル /etc/default/LifeKeeper の LK_TRAP_MGR 環境変数から IP アドレスの割り当てを削除します。削除するには、コマンドラインで「disable」オプションを指定して lk_configsnmp ユーティリティを実行します (例については lk_configsnmp(1M) マニュアルページを参照)。または、/etc/default/LifeKeeper を編集して、LK_TRAP_MGR のエントリを LK_TRAP_MGR= に変更します (または行全体を削除する)。この手順は、トラップメッセージの送信を無効にしたい各ノードで実行する必要があります。

# 5.4.2.2.3. SNMP のトラブルシューティング

SNMP によるイベント転送に関連して予想される問題とその解決策を以下に説明します。具体的なエラーメッセージについては、 LifeKeeper メッセージカタログ を参照してください。

**問題:** LifeKeeper から SNMP のトラップメッセージが送信されない。

**解決策:** snmptrap ユーティリティがインストールされていることを確認します (通常は /bin/bin にあります)。インストールされていない場合は、適切な SNMP パッケージをインストールします (前提条件 を参照)。別の場所にインストールされている場合は、ファイル /etc/default/LifeKeeper の PATH 変数に適切なパスを追加します。

**問題:** SNMP のエラーメッセージがログに記録されない。LifeKeeper サーバから SNMP のトラップメッセージが送信されていないように見える。

**解決策:** トラップを受信するネットワーク管理サーバの IP アドレスが LK_TRAP_MGR に設定されていることを確認します。コマンドラインで、lk_configsnmpを「--query」オプション付きで使用して設定を確認します (lk_configsnmp(1M) マニュアルページの例を参照してください)。または、ファイル /etc/default/LifeKeeper の LK_TRAP_MGRのエントリを確認します。この変数は、SNMP トラップメッセージを生成する LifeKeeper の各ノードで設定する必要があります。

# 5.4.2.3. LifeKeeper イベントメール通知

LifeKeeper イベントメール通知の概要

LifeKeeper イベントメール通知の設定

メール通知のトラブルシューティング

# 5.4.2.3.1. LifeKeeper イベントメール通知の概要

LifeKeeper イベントメール通知は、特定のイベントが LifeKeeper クラスタで発生したときに 1 人以上のユーザがメールによる通知を受信する仕組みです。LifeKeeper のイベント通知機能では、特定のイベントが起きたときに通知を受信するアプリケーションを登録することができます (sendevent(5) マニュアルページを参照)。LifeKeeper は、LifeKeeper の動作を監視したいユーザのグループに向けて LifeKeeper の重要なイベントに関するメール通知を送信するように簡単に設定できます。さらに、/var/log/lifekeeper.log ファイルまたは LifeKeeper GUI の サーバログファイルの表示 機能を使用すると、送信された各メール通知のログを参照することができます。メッセージは通常 NOTIFY ログに入ります。ログの内容をコマンドラインで表示する方法の詳細については、lk_log(8) マニュアルページを参照してください。

デフォルトでは、LifeKeeper イベントメール通知は無効になっています。この機能を有効にするには、/etc/default/LifeKeeper で指定する LK_NOTIFY_ALIAS 環境変数を設定する必要があります。LK_NOTIFY_ALIAS 環境変数には、メールアドレスまたはエイリアスを 1 つまたは複数個 (カンマ区切り) 設定できます。LK_NOTIFY_ALIAS を設定するには、コマンドラインから lk_confignotify alias (lk_configotifyalias(1M) マニュアルページで例を参照してください) を実行してイベントが発生したときにメールを受信するアドレスまたはアドレスリストを指定するか、デフォルトファイル /etc/default/LifeKeeper を編集してメールアドレスまたはアドレスリストを追加します。LK_NOTIFY_ALIAS= エントリを見つけて、アドレスまたはカンマ区切りのアドレスリストを入力します。選択した LifeKeeper イベントについてメールを送信する必要があるクラスタのすべてのノードで以上の手順を繰り返します。

メール通知を無効にするには、引数 *-disable* を付けて **lk_configotifyalias** (lk_configotifyalias(1M) マニュアルページで例を参照してください) を実行するか、デフォルトファイル /etc/default/LifeKeeper を編集して LK_NOTIFY_ALIAS の設定を削除します (この行を LK_NOTIFY_ALIAS= に変更)。

## メールが生成される LifeKeeper のイベント

以下の LifeKeeper イベントが発生するとメール通知が生成されます (LK_NOTIFY_ALIAS が設定されている場合)。

| LifeKeeper のイベント | イベントの説明 |
|---|---|
| LifeKeeper Startup Complete | LifeKeeper が起動したノードから送信されます。 |
| LifeKeeper Shutdown Initiated | LifeKeeper のシャットダウンを開始したノードから送信されます。 |
| LifeKeeper Shutdown Complete | LifeKeeper のシャットダウンを完了したノードから送信されます。 |
| LifeKeeper Manual | 手動スイッチオーバを要求されたノードから送信されます。 |

| | |
|---|---|
| Switchover Initiated on Server | |
| LifeKeeper Manual Switchover Complete - recovered list | 手動スイッチオーバが完了したノードから、リカバリに成功したリソースのリストと共に送信されます。 |
| LifeKeeper Manual Switchover Complete - failed list | 手動スイッチオーバが完了したノードから、切り替えに失敗したリソースのリストと共に送信されます。 |
| LifeKeeper Node Failure Detected | クラスタ内のノードに障害が発生したときに、クラスタ内の各ノードから送信されます。 |
| LifeKeeper Node Recovery Complete for Server - recovered list | 障害ノードからのリソースをリカバリしたクラスタ内の各ノードから、リカバリに成功したリソースのリストと共に送信されます。 |
| LifeKeeper Node Recovery Complete for Server - failed list | 障害ノードからのリソースのリカバリに失敗したクラスタ内の各ノードから、リカバリに失敗したリソースのリストと共に送信されます。 |
| LifeKeeper Resource Recovery Initiated | リソースをリカバリしているノードから送信されます。このメールに続いて、リカバリが完了したか失敗したかを示すメッセージ(「Resource Recovery Complete」または「Resource Recovery Failed」) が必ず送信されます。 |
| LifeKeeper Resource Recovery Complete | リソースのリカバリが成功した時点で、「LifeKeeper Resource Recovery Initiated」メッセージを送信したノードから、リカバリに成功したリソースのリストと共に送信されます。 |
| LifeKeeper Resource Recovery Failed | リソースが In Service の状態になることができない場合に、「LifeKeeper Resource Recovery Initiated」メッセージを送信したノードから、リカバリに成功したリソースのリストと共に送信されます。 |
| LifeKeeper Communications Path Up | ノードへのコミュニケーションパスが確立されました。 |
| LifeKeeper Communications Path Down | ノードへのコミュニケーションパスがダウンしました。 |
| LifeKeeper <Node monitoring> Failure Detected | Standby Node Health Check における Node監視で異常を検知したときに、異常を検知したノードから送信されます。<Node monitoring> には、検知した異常の内容が記載されます。 |
| LifeKeeper <OSUquickCheck> Failure Detected | Standby Node Health Check における OSUリソース監視で異常を検知したときに、異常を検知したノードから送信されます。<OSUquickCheck> には、異常が検知されたリソースのタグ名が記載されます。 |

# 5.4.2.3.2. LifeKeeper イベントメール通知の設定

## 前提条件

イベントメール通知機能は、LifeKeeper のコア機能の一部として含まれており、LifeKeeper の追加パッケージをインストールする必要はありません。ただし、LifeKeeper イベントのメール通知を生成する LifeKeeper の各ノードに電子メールソフトウェアがインストールされている必要があります。LifeKeeper は、mailx パッケージによってインストールされるメールユーティリティを使用して通知を送信します。

メールの設定は、LifeKeeper の本機能が提供する範囲ではありません。デフォルトでは、LifeKeeper イベントメール通知は無効になっています。

## 設定作業

LifeKeeper イベントメール通知を設定するには、以下の作業を実施します。

1. 上述のメールユーティリティが利用できることを確認します。
2. LifeKeeper のイベントのメール通知を受信するユーザ (1 人以上) を特定し、LifeKeeper のデフォルトファイル `/etc/default/LifeKeeper` の `LK_NOTIFY_ALIAS` を設定します。これを行うには、コマンドラインを使用するか、ファイル `/etc/default/LifeKeeper` を編集して通知を受信するメールアドレスまたはエイリアスを指定します。

- ◦ コマンドラインからは、`lk_confignotifyalias` を使用します (詳細については、`lk_confignotifyalias (1M)` のマニュアルページを参照してください)。このユーティリティでは、カンマ区切りのメールアドレスまたはエイリアスのみ使用できます。
- ◦ または、デフォルトファイル `/etc/default/LifeKeeper` を編集してメールアドレスまたはエイリアスを追加します。`LK_NOTIFY_ALIAS=` エントリを見つけて「=」の右側にメールアドレスまたはエイリアス (1 つまたはカンマ区切りのリスト) を入力します (「=」の前後にはスペースを入れません)。

## 設定の確認

設定が正常に動作することを確認するには、LifeKeeper の処理を実行します (例えば、LifeKeeper を 開始 または 停止 する、または LifeKeeper GUI を使用して、あるリソースを手動で In Service の状態にするなど)。ファイル `/etc/default/LifeKeeper` の `LK_NOTIFY_ALIAS` で指定したユーザがメールを受信していること、LifeKeeper のログファイルにメッセージが記録されていることを確認します。メールを受信していない場合は、メール障害に対する通常のトラブルシューティング手順を実行してください。
LifeKeeper のログを調べると、メール送信に問題があるかどうかを判断することができます。詳細については、メール通知のトラブルシューティング を参照してください。

# イベントメール通知の無効化

LifeKeeper によるメール通知の生成を無効にするには、ファイル /etc/default/LifeKeeper の LK_NOTIFY_ALIAS 環境変数からメールアドレスとエイリアスの割り当てを削除するだけです。コマンドラインで lk_confignotifyalias ユーティリティを「--disable」オプションを付けて実行します (k_confignotifyalias (1M) マニュアルページの例を参照してください)。または、/etc/default/LifeKeeper を編集して、LK_NOTIFY_ALIAS のエントリを LK_NOTIFY_ALIAS = に変更します。この手順は、メール送信を無効にしたい各ノードで実行する必要があります。

# 5.4.2.3.3. メール通知のトラブルシューティング

LifeKeeper イベントのメール通知に関連して予想される問題とその解決策を以下に説明します。具体的なエラーメッセージについては、 LifeKeeper メッセージカタログ を参照してください。

**問題:** LifeKeeper からのメールを受信しない。

**解決策:** メールユーティリティがインストールされていることを確認します (通常は /bin/mail にあります)。インストールされていない場合は、mailx パッケージをインストールします。別の場所にインストールされている場合は、ファイル /etc/default/LifeKeeperPATH 変数にメールユーティリティのパスを追加します。

**問題:** LifeKeeper からのメールを受信しない。

**解決策:** メール設定を確認し、配信用のキューにメールメッセージが滞留していないことを確認します。メール設定の問題が原因でメッセージが滞留することがあります。LK_NOTIFY_ALIAS で指定しているメールアドレスが有効なアドレスであり、カンマで区切られていることを確認します。

**問題:** ログファイルに「mail returned」というエラーメッセージがある。

**解決策:** メールコマンドがエラー「X」を返す場合、LifeKeeper イベントがメールを生成、送信する際に問題 (「node failure」など) が発生しています。メール設定を確認し、LK_NOTIFY_ALIAS に含まれるメールアドレスが有効であり、アドレスのリストがカンマで区切られていることを確認します。また、LK_NOTIFY_ALIAS で指定しているメールアドレスのフォーマットを使用してコマンドラインからそれらのアドレスにメールを送信できることを確認します。

**問題:** メッセージや成功または失敗がログに何も記録されず、ノードのフェイルなどの LifeKeeper イベントが発生したときもメールを受信するはずのユーザがメールを受信しない。

**解決策:** LK_NOTIFY_ALIAS にメールアドレスが設定されており、複数の場合はカンマで区切られていることを確認します。コマンドラインで、lk_configNotifyAlias を「--query」オプション付きで使用して設定を確認します (lk_configNotifyAlias(1M) マニュアルページの例を参照してください)。または、ファイル /etc/default/LifeKeeper の LK_NOTIFY_ALIAS で確認します。この変数は、メール通知メッセージを生成する LifeKeeper の各ノードで設定する必要があります。また、LifeKeeper イベントメール通知の概要 で、その LifeKeeper イベントがメールメッセージを生成するのかどうかを確認します (すべてのイベントがメールメッセージを生成するわけではありません)。

# 5.4.2.4. 任意の設定作業

[Confirm Failover] と [Block Resource Failover] の設定

サーバのシャットダウンストラテジーの設定

LifeKeeper ハートビートの調整

SPS API でカスタム証明書を使用する

# 5.4.2.4.1. [Confirm Failover] と [Block Resource Failover] の設定

LifeKeeperの通常の動作として、ノード障害やリソース障害によるバックアップノードへの切り替えは自動的に行われるようになっています。しかし、利用する環境によっては、障害が検出されたシステムのフェイルオーバーリカバリーをLifeKeeper が実行する前にシステム管理者の手動による確認を必須とすることが望ましいこともあります。そのような場合には、この **[Confirm Failover]** や **[Block Resource Failover]** 設定の利用することができます。この機能を利用することによって、リソース障害やノード障害が発生した際に、自動的に行われるフェイルオーバーをブロックしたり、フェイルオーバーが行われるまでの待ち時間を設けたりすることができます。

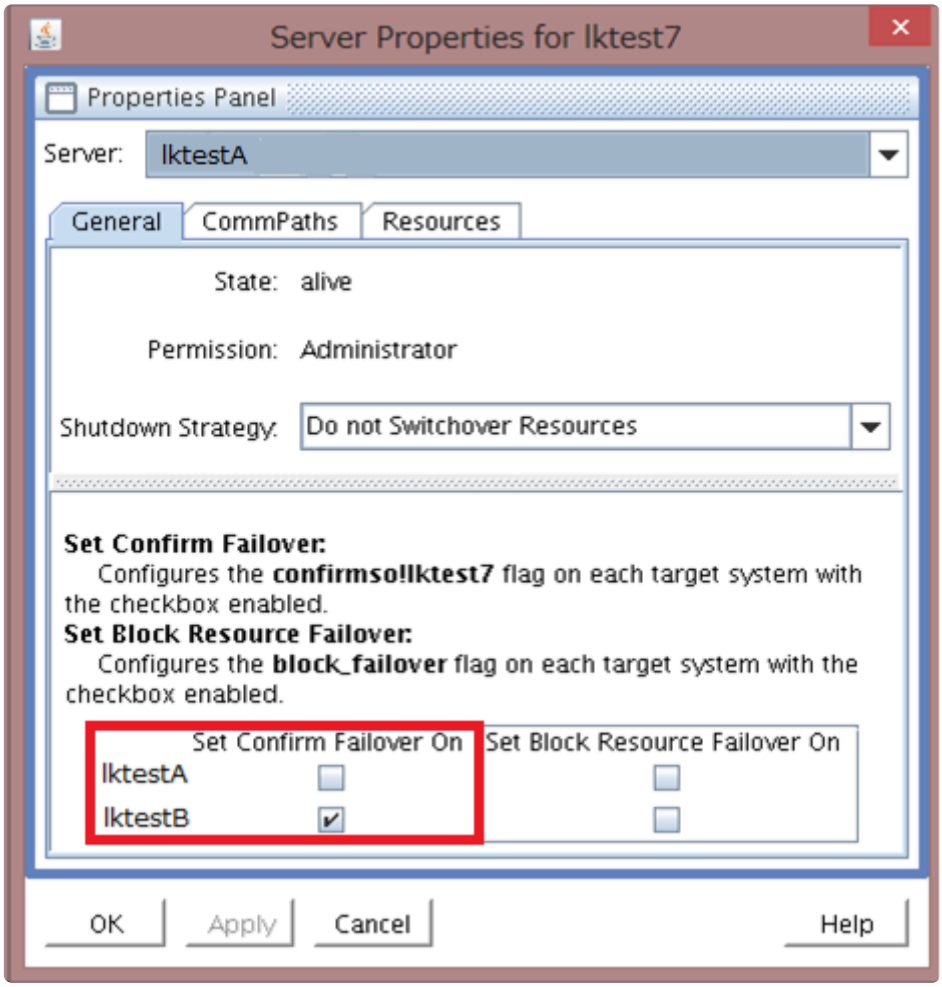
以下の説明、例、および考慮事項をよく読んで理解してから、お使いの SPS 環境で **[Confirm Failover]** または **[Block Resource Failover]** を設定してください。これらの設定は、コマンドライン、または **LifeKeeper の GUI の [Properties]** パネルから使用できます。

## [Set Confirm Failover On]

**[Confirm Failover]** を設定すると、LifeKeeper クラスターのノード障害によるフェイルオーバーが発生した時（注記: ノード障害は、システムへのすべてのLifeKeeperコミュニケーションパス障害によって識別されます。）、バックアップノードへの切り替えの実行に対して待ち時間を設けることができるようになります（後述のCONFIRMSODEF変数を参照して下さい）。また、その待ち時間の間にバックアップノードへの切り替えを実行するかしないかをユーザーが決定できるようになります。

> ✻ 注記: Set Confirm Failover On設定は、ノード障害が発生した場合にのみ有効です。一つ以上のコミュニケーションパスがアクティブであるリソース障害に対しては有効ではありません。

**[Confirm Failover]** 設定をGUIで有効にする場合は、サーバープロパティーのGeneralタブ画面を使用します。以下はサーバープロパティーのGeneral画面の例です。赤で囲まれた部分が **[Confirm Failover]** を設定する画面です。




> 注意： この設定は、SPS の管理者権限を持つユーザーのみが使用できます。

この画面の例では、lktestAという名前のホストから見た設定となります。上図の赤で囲まれた部分が **[Confirm Failover]** の設定部分で、HAクラスターを構成するノードの名前が縦に表示されています。この例では、lktestAの対向ノードがlktestBとなっています。

図の例ではlktestAを選択して設定状況を表示しており、lktestBの横にチェックが入った状態となっています。この場合、lktestB上にConfirm Failover フラグが作成され、lktestAからlktestBへのフェイルオーバーが行われる際に、フェイルオーバー実施までのタイムアウトと実施可否の確認が有効になります。このフラグの有無によって、 **[Confirm Failover]** の動作の実施がコントロールされます。

この時のフラグの作成状況については、コマンドで確認することができます。例にあるlktestAというホスト上でlktestBの欄にチェックを入れた場合には、lktestBに **[Confirm Failover]** フラグが作成されます。(例の通りの操作の場合lktestAにはフラグは作成されません)。具体的には以下のような出力になります。

```
[root@lktestB~]# flg_list

                confirmso!lktestA
```

「confirmso!lktestA」はflg_listコマンドの結果が出力され、 **[Confirm Failover]** フラグがlktestBノード上に設定されます。confirmsoフラグがセットされている状態で、フェイルオーバーが発生するとLifeKeeperのログファイルには以下のようなログが記録されます。

INFO:lcd.recover:::004113:chk_man_interv: Flag confirmso!hostname is set, issuing confirmso event and waiting for switchover instruction.

NOTIFY:event.confirmso:::010464:LifeKeeper: FAILOVER RECOVERY OF MACHINE lktestA requires manual confirmation! Execute '/opt/LifeKeeper/bin/lk_confirmso -y -s hostname ' to allow this failover, or execute '/opt/LifeKeeper/bin/lk_confirmso -n -s lktestA' to prevent it. If no instruction is provided, LifeKeeper will timeout in 600 seconds and the failover will be allowed to proceed.

このメッセージが出力されている間に、次のいずれかのコマンドを実行するとフェイルオーバーの可否をコントロールすることができます。

**フェイルオーバーを続行する場合**

#/opt/LifeKeeper/bin/lk_confirmso -y -s ホスト名

**フェイルオーバーをブロックする場合**

#/opt/LifeKeeper/bin/lk_confirmso -n -s ホスト名

コマンドを実行する際に指定するホスト名はConfirm Failoverフラグに書かれているホスト名、この例では、lktestAとなります。ログに具体的なコマンドの実行例が書かれていますので、内容を元にコマンドを実行してください。

設定された待ち時間を超えた場合、LifeKeeperのデフォルトの設定ではバックアップノードへのフェイルオーバー（あるいは、フェイルオーバーのブロック）を自動的に行うようになっています。タイムアウトを迎えた場合以下のようなログが記録されます。

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

待ち時間を経過した際の動作は、/etc/default/LifeKeeperファイルの「CONFIRMSODEF=1 or 0」の設定で制御されます。デフォルトでは”0”が設定されており、待ち時間を超えた場合にはフェイルオーバーが継続されます。この値を”1”に変更した場合には、待ち時間を経過した場合にフェイルオーバーしないようにすることができます。

フェイルオーバー時の待ち時間を変更したい場合には、/etc/default/LifeKeeperファイルの「CONFIRMSOTO＝秒」の値を変更してください。上記CONFIRMSODEFの値によって決定されたフェイルオーバーの実行、あるいはブロックをする前に、この変数はユーザーからの手動確認を待つための秒数を指定します。この設定をこれらの変数の影響を反映するために、LifeKeeperやOSを再起動する必要はありません。CONFIRMSOTOに０秒を設定すると待ち時間無しで、CONFIRMSODEFの設定に基づいた動作をさせることができます。

## [Confirm Failover]設定を選択するタイミング

この設定は、コミュニケーションパスが冗長化されていない環境で場合のディザスターリカバリーやWAN構成で使用されます。

- 通常のサイト（非マルチサイトクラスターおよび非XenServer）では、あるサーバーで **[Properties]**

ページを開き、 **[Confirm Failover]** フラグをオンに設定するサーバーを選択してください。

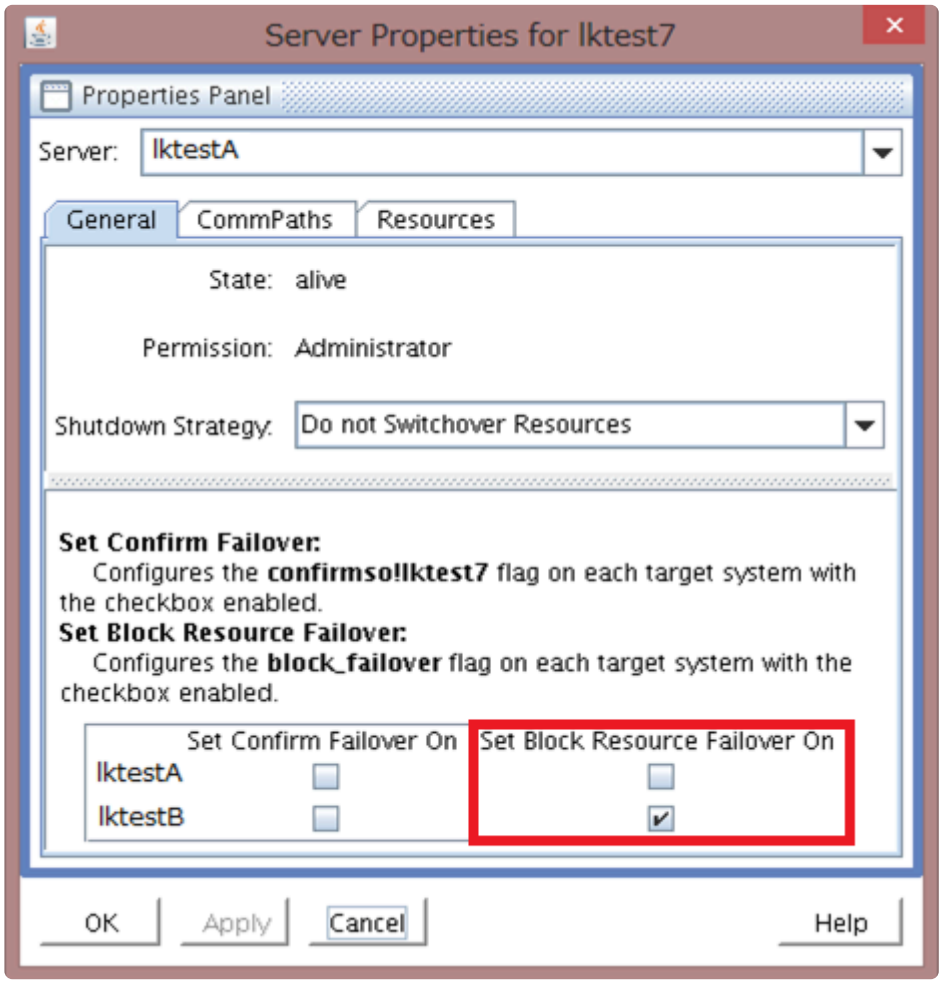
# [Block Resource Failover On]

**[Block Resource Failover]** 設定は、指定したシステムでのリソース障害に起因するフェイルオーバーをブロックします。

> 注記 : Block Resource Failover設定は、ノード障害が発生した場合のフェイルオーバー動作には影響しません。この設定は、ローカルリソースの回復に失敗し、クラスターで他ノードへリソースを転送する場合のみフェイルオーバーをブロックします。

デフォルトでは、リソース障害を検知した時、ローカルシステムでの障害リソースのリカバリー（ローカルリカバリー）を試行し、ローカルリカバリーが失敗した場合、または有効になっていない場合は、リソースが定義されている、優先順位が次に最も高いスタンバイノードにフェイルオーバーしようとします。Block Resource Failoverの設定は、この時のフェイルオーバーをブロックします。

Block Resource Failoverの設定をGUIで有効にする場合は、サーバープロパティーのGeneralタブ画面を使用します。以下はサーバープロパティーのGeneral画面の例です。赤で囲まれた部分が **[Block Resource Failover　On]** を設定する画面です。



> ✻ 注意 ：この設定は、SPS の管理者権限を持つユーザーのみが使用できます。

この画面の例は、lktestAという名前のホストから見た設定となります。上図の赤で囲まれた部分が **[Confirm Failover]** の設定部分で、HAクラスターを構成するノードの名前が縦に表示されています。この例では、lktestAの対向ノードがlktestBとなっています。

この場合、lktestB上にBlock Resource Failover フラグが作成されます。lktestBでflg_listコマンドを実行すると、「block_failover」というフラグが作成されていることを確認することができます。出力例は以下の通りです。

[root@lktestB~]# flg_list

block_failover

この結果lkestB上で、リソース障害が発生した場合には、他ノード （**lktestA**） へのフェイルオーバーがブロックされます。

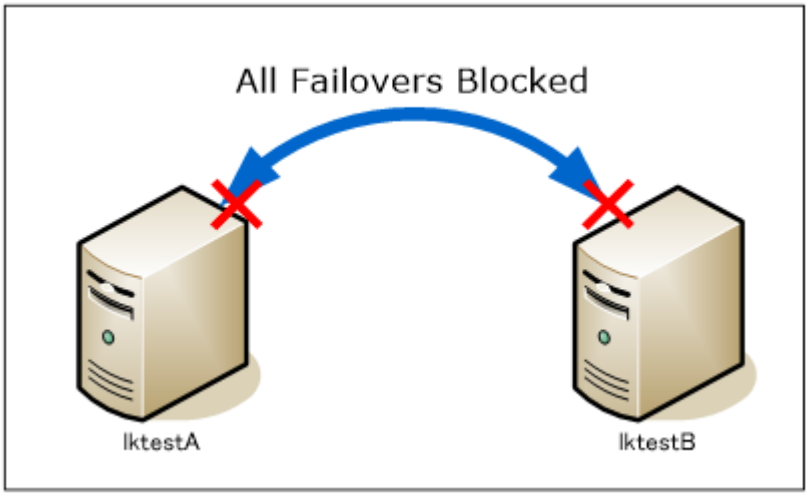
**block_failover**フラグは、それがセットされているノード上で発生したリソース障害によるフェイルオーバーをブロックします。 この設定によってフェイルオーバーがブロックされた場合には、以下のようなログが記録されます。

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

# 設定例

いくつかの具体的な設定例を解説します。

## 全ての自動フェイルオーバーをすべてブロックする




この例ではlktestAとlktestBどちらでノード障害、リソース障害が発生してもフェイルオーバーしないようにします。この設定にはConfirm Failover設定とBlock Resource failover設定を利用します。LKCLI ではコマンド1つで設定可能です。詳細は lkcli server block-all-failovers を参照ください。GUI を使った設定例は

次の通りです。

1. lktestAサーバーを選択し、 **[Server Properties]** を表示してください。

   **[General]** タブで、lktestBの行にある **[Set Confirm Failover On]** ボックスと、LktestAとLktestB両方の行にある **[Set Block Resource Failover on]** ボックスをオンにしてください。

   GUIのlktestAのサーバープロパティー設定例

| | Set Confirm Failover On | Set Block Resource Failover On |
|---|---|---|
| lktestA | (チェックなし) | ✔ |
| lktestB | ✔ | ✔ |

   ※GUIでサーバープロパティーをみる場合、ノード名は、プロパティーパネルの上部近くに表示されます。

2. lktestBサーバーを選択し、 **[Server Properties]** を表示してください。

   **[General]** タブで、lktestAの行にある **[Set Confirm Failover On]** ボックスをオンにしてください。

   GUIのlktestBのサーバープロパティー設定例

| | Set Confirm Failover On | Set Block Resource Failover On |
|---|---|---|
| lktestB | (チェックなし) | ✔ |
| lktestA | ✔ | ✔ |

   ※GUIでは表示している自ノードのホスト名が列の上部に表示されます

   GUIで設定した後、lktestBでflg_listコマンドを使用してフラグ設定を確認すると、「confirmso!lktestA」フラグが作成されていることが確認できます。

   ここまでの結果それぞれのノードにconfirmso!hostnameとblock_failoverフラグがセットされていることを確認してください。confirmsoフラグについては、lktestA上のlktestBからのフェイルオーバーをブロックするために、フェイルオーバー確認が実行されることになっているhostnameが、フラグ名の内容の一部として記載されることを確認して下さい。lktestAは、lktestA上でlktestBをconfirmsoフラグ名の内容に記載する必要があります。本設定例ではそれぞれのノードで以下のようにフラグが作成されます。

| | Confirm Failover フラグ | Block Resource Failoverフラグ |
|---|---|---|
| lktestA側 | confirmso!lktestB | block_failover |
| lktestB側 | confirmso!lktestA | block_failover |

3. 両ノードの/etc/default/LifeKeeper のCONFIRMSOTOとCONFIRMSODEFの設定値を以下のように設定してください。（LifeKeeperやOSの再起動は必要ありません。）

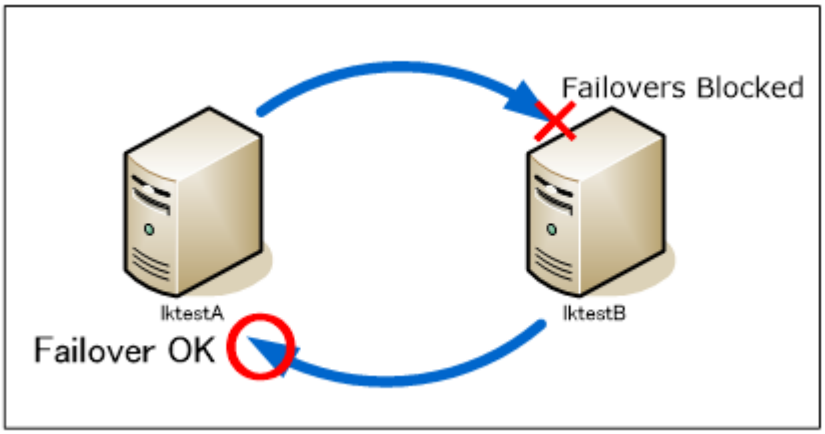
   CONFIRMSODEF = 1

   CONFIRMSOTO = 0

   CONFIRMSOTOの設定については待ち時間を設ける場合には時間を秒で指定することも可能です。ポイントはCONFIRMSODEFの設定が0(フェイルオーバーする)から1(フェイルオーバーしない)になっていることです。

   上記の設定により、オペレータが介在することなく、ノード障害はすぐにブロックされます。

## 一方向のフェイルオーバーをブロックする




この例ではlktestAで障害が検知された場合にはノード障害、リソース障害どちらの場合でもlktestBへのフェイルオーバーをブロックします。逆に、lktestBで障害が発生した場合にはノード障害、リソース障害どちらの場合でもServerAへフェイルオーバーします。

1. lktestAを選択し、 **[Server Properties]** を表示してください。

2. **[General]** タブで、lktestAの行にある **[Set Confirm Failover On]** ボックスと、lktestAの行にある **[Set Block Resource Failover On]** ボックスをオンにしてください。GUIでの設定状態は以下のようになります。

   GUIのlktestAのサーバープロパティー設定例

| | Set Confirm Failover On | Set Block Resource Failover On |
|---|---|---|
| lktestA | (チェックなし) | ✔ |
| lktestB | ✔ | (チェックなし) |

3. lktestBサーバーを選択し、 **[Server Properties]** を表示してください。

   **[General]** タブで、lktestAの行にある **[Set Confirm Failover On]** ボックスをオフ（チェックな

し）にしてください。GUIでの設定状態は以下のようになります。

GUIのlktestBのサーバープロパティー設定例

| | Set Confirm Failover On | Set Block Resource Failover On |
|---|---|---|
| lktestB | (チェックなし) | (チェックなし) |
| lktestA | (チェックなし) | ✔ |

※GUIでは表示している自ノードのホスト名が列の上部に表示されます。

この設定で、lktestB上に「confirmso!lktestA」フラグ、lktestA上に「Block failover」フラグが設定されていることを確認してください　(lktestAには、confirmsoフラグは設定されません)

| | Confirm Failover フラグ | Block Resource Failoverフラグ |
|---|---|---|
| lktestA | なし | block_failover |
| lktestB | confirmso!lktestA | なし |

4. lktestBの/etc/default/LifeKeeper を以下のように設定してください。LifeKeeperやOSの再起動は必要ありません。

   CONFIRMSODEF = 1

   CONFIRMSOTO = 0

   ここまでの設定によって、lktestAサーバーからlktestBのサーバーへのフェイルオーバーは許可されますが、lktestBからlktestAへのフェイルオーバーは許可されません。

# 5.4.2.4.2. サーバーのシャットダウンストラテジーの設定

シャットダウンストラテジーは、サーバーがシャットダウンするときにバックアップサーバーにリソースをスイッチオーバーするかどうかを制御する LifeKeeper の設定オプションです。以下のオプションがあります。

| | |
|---|---|
| Do Not Switch Over Resources (デフォルト) | LifeKeeper は、正常なシャットダウンではバックアップサーバーのリソースを起動しません。 |
| Switch Over Resources | LifeKeeper は、正常なシャットダウンでバックアップサーバーのリソースを起動します。 |

シャットダウンストラテジーは、デフォルトでは「Do Not Switch Over Resources」に設定されています。クラスター内の各サーバーでどちらのストラテジーを使用するかを決定し、必要に応じてシャットダウンストラテジーを「Switch Over Resources」に変更してください。

クラスター内の各サーバーで次のようにします。

1. [Edit] メニュー で **[Server]** を選択し、次に **[Properties]** をクリックします。
2. 修正するサーバーを選択します。
3. **[Server Properties]** ダイアログの [General] タブ で、 **[Shutdown Strategy]** を選択します。

**注意：** シャットダウンストラテジーが有効に機能するには、正常なシャットダウン時に LifeKeeper のプロセスが起動している必要があります。

# 5.4.2.4.3. LifeKeeper ハートビートの調整

## ハートビート設定項目の概要

LifeKeeper のハートビートは、各サーバが「生存」していることを確認するためにコミュニケーションパスを通じて LifeKeeper のサーバ間で送信される信号です。ハートビートに関しては、LifeKeeper が障害を検知する速さを決定する要素が 2 つあります。

- 間隔: ハートビート信号を送信する時間の間隔（単位は秒）。また、この時間内に対向サーバからハートビート信号を含むLCM信号を受信できなかった場合、ハートビート停止と判断する。
- ハートビート回数: コミュニケーションパスが切断していると判定するハートビートの連続失敗回数。

これらのハートビートの値は、LifeKeeper デフォルトファイル /etc/default/LifeKeeper 内の以下の 2 つの設定項目で指定します。デフォルト値を使用した場合よりも早期にサーバの障害を検知したい場合は、設定項目を変更することができます。

- LCMHBEATTIME (間隔)
- LCMNUMHBEATS (ハートビート回数)

次の表は、TCP および TTY 経由のハートビートの設定項目についてのデフォルト値と最小値の一覧です。TTY コミュニケーションパスは、媒体として通信速度が遅いため、間隔を 2 秒未満にすることはできません。

| 設定項目 | デフォルト値 | 最小値 |
|---|---|---|
| LCMHBEATTIME | 5 | 1 (TCP) 2 (TTY) |
| LCMNUMHBEATS | 3 | 2 (TCP、TTY) |

重要な注記: どちらの設定項目も、クラスタ内のすべてのサーバで必ず同じ値にする必要があります。

### 例

LifeKeeper のクラスタで両方の間隔がデフォルト値に設定されていると仮定します。LifeKeeper は、サーバ間で 5 秒ごとにハートビートを送信します。通信障害によって 2 回のハートビートが途絶し、3 回目のハートビートで再開した場合、LifeKeeper はアクションを実行しません。コミュニケーションパスの切断継続時間がハートビート 3 回分になった場合は、LifeKeeper はそのコミュニケーションパスを切断と判定します。ただし、他方の冗長的なコミュニケーションパスも切断と判定されるまではフェイルオーバを開始しません。

## ハートビートの設定

設定項目とその値を追加するには、/etc/default/LifeKeeper ファイルを手動で編集する必要があります。通

常、デフォルトファイルにはこれらの設定項目のエントリが含まれていません。設定する値を含む次の行を最終行以降に追加してください。

```
LCMHBEATTIME=x

LCMNUMHBEATS=y
```

最小値を下回る値を設定した場合、LifeKeeper はその値を無視して代わりに最小値を採用します。

## 設定上の考慮事項

- 間隔を 5 秒未満に設定すると、ネットワークの中断による誤ったフェイルオーバを発生させるリスクが高くなるため、5 秒未満に設定する場合は、コミュニケーションパスをプライベートネットワーク上で構成してください。

- 検証によると、ハートビート回数を 2 未満にした場合に誤ったフェイルオーバの発生リスクが高まります。このため、この値は 2 以上に制限されています。

- 誤ったフェイルオーバを回避するため、間隔およびハートビート回数の値はどちらもクラスタ内のすべてのサーバで必ず同じ値にする必要があります。このため、これらの値を編集する前に両方のサーバで LifeKeeper を停止しておく必要があります。LifeKeeper の稼働開始後、アプリケーションを保護している状態でハートビートの設定項目を編集する場合は、コマンド "`/opt/LifeKeeper/bin/lkstop -f`" を使用できます。このコマンドは LifeKeeper を停止しますが、保護下のアプリケーションは停止しません。

- LCMHBEATTIME および LCMNUMHBEATS の値に上限値はありません。ただし、非常に大きい数字に値を設定すると、LifeKeeper の障害検知能力は著しく損なわれます。例えば、両方の値を 25 に設定した場合、サーバ障害を検知するまで LifeKeeper は 625 秒間 (10 分間以上) 待機します。これはサーバをリブートしてクラスタに再参加させるのに十分な時間です。

**注記:** TTY および TCP コミュニケーションパスの両方を使用する場合、各設定項目の値は両方のコミュニケーションパスに適用されます。唯一の例外は、TTY コミュニケーションパスの最小値である 2 未満の値が間隔に設定された場合です。

例えば、障害をできるだけ早く検知するために、LifeKeeper で許容される最小値を指定したとします。

```
LCMHBEATTIME=1

LCMNUMHBEATS=2
```

このとき LifeKeeper は TCP コミュニケーションパスの間隔に 1 秒を採用し、TTY の間隔には 2 秒を採用します。サーバ障害が発生すると、LifeKeeper は間隔の短い TCP の障害 (1 秒間隔の 2 回のハートビート後) を先に検知します。ただし、TTY の障害 (2 秒間隔の 2 回のハートビート後) を検知するまでは何もしません。

# 5.4.2.4.4. SPS API でカスタム証明書を使用する

SIOS Protection Suite (SPS) API のリリース 7.5 以降では、異なるシステムとの通信に SSL/TLS が使用されます。現在、この API は一部のみ使用され、内部使用のみとして予約されていますが、将来のリリースではお客様とサードパーティが使用できるように公開される可能性があります。デフォルトでは、ノード間で一定の身元確認が可能なデフォルト証明書が SPS と共にインストールされます。このドキュメントでは、デフォルト証明書を組織独自の認証局 (CA) が作成した証明書に置き換える方法を説明します。

注記: 通常の SPS 通信では、これらの証明書は使用されません。

## 証明書の使用方法

データ転送を保護するために SPS サーバ間の通信に SSL/TLS が使用されている場合、システムがそれ自体を識別するために証明書を提供します。また、システムは、CA 証明書を使用して、SSL 接続経由で提示された証明書を検証します。

以下の 3 種類の証明書が使用されます。

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

最初の 2 つの証明書は、サーバが実行する検証に合格するために CA 証明書による署名が必要です。証明書の共通名は検証されません。証明書は CA によって署名されるのみということに注意してください。

## 独自の証明書の使用

運用環境によっては、デフォルト証明書を組織内部の CA または商用 CA が作成した証明書に置き換える必要がある場合があります。そのような場合は、上記の 3 種類の証明書を、 *同じ証明書ファイル名* を持つ新しい証明書に置き換えます。これらの証明書は PEM 形式です。`LK4LinuxValidNode.pem` および `LK4LinuxValidClient.pem` はそれぞれ、キーと証明書の両方を含んでいます。`LK4LinuxValidNode.pem` 証明書は、サーバタイプの証明書です。`LK4LinuxValidClient.pem` は、クライアントタイプの証明書です。

デフォルトの証明書を置換した場合、変更を反映するために SPS を再起動する必要があります。証明書の設定を間違えると、`steeleye-lighttpd` デーモンが起動に失敗し、LifeKeeper のログファイルにエラーが記録されます。問題が発生した場合、このログファイルを参照すると実行すべき完全なコマンドを見ることができます。

# 期限切れの証明書の更新

注意：v9.5.1およびそれ未満のバージョンの証明書は、2021年9月に期限切れとなります。上記すべての古い証明書から、同じファイル名の新しい証明書への置き換えを実施してください。最新の証明書につきましては 弊社LifeKeeper / DataKeeper ユーザーポータル をご参照の上、ダウンロードしてください。

# 5.4.2.5. Linux の設定

<table>
<tr><td>オペレーティングシステム</td><td>必要なすべてのパッケージをインストールするためには、オペレーティングシステムはデフォルトでインストールしてください。最小構成のオペレーティングシステムでは必要なすべてのパッケージが含まれないため、LifeKeeper で使用することはできません。</td></tr>
<tr><td>カーネルのアップデート</td><td>LifeKeeper クラスタの可用性を最大限に引き出すには、システムで使用するカーネルのバージョンが非常に重要です。次の表に、LifeKeeper 認定テストに合格した、サポート対象のディストリビューション、バージョン、およびカーネルを示します。<br><br>注記:カーネルのアップグレードを実行する際、インストールイメージから setup スクリプト(./setup) の再実行が必要になる場合があります。カーネルアップグレード後に DataKeeper が起動しない場合には、setup スクリプトを実行して適切なカーネルモジュールをインストールしてください。
<table>
<tr><th>ディストリビューション/バージョン</th><th>サポート対象のバージョン</th><th>サポート対象のカーネル</th></tr>
<tr><td>Red Hat Enterprise Linux for AMD64/EM64T<br><br>(*DataKeeper 非同期ミラーは、RHEL 7.4～7.6 の一部のカーネルではサポートされていません 詳細については、ここをクリックしてください。 )</td><td>7<br>7.1<br>7.2<br>7.3<br>7.4*<br>7.5*<br>7.6*<br>7.7<br>7.8<br>7.9</td><td>3.10.0-123.el7<br>3.10.0-229.el7<br>3.10.0-327.el7<br>3.10.0-514.el7<br>3.10.0-693.el7<br>3.10.0-862.el7<br>3.10.0-957.el7<br>3.10.0-1062.el7<br>3.10.0-1127.el7<br>3.10.0-1160.el7</td></tr>
<tr><td>Red Hat Enterprise Linux for AMD64/EM64T<br><br>(RHEL 7 から RHEL 8 のようにKernelをメジャーバージョンアップすることはサポートされません)</td><td>8.0<br>8.1<br>8.2<br>8.3<br>8.4</td><td>4.18.0-80.el8.x86_64<br>4.18.0-147.el8.x86_64<br>4.18.0-193.el8.x86_64<br>4.18.0-240.el8.x86_64<br>4.18.0-305.el8.x86_64</td></tr>
<tr><td>SUSE Linux Enterprise Server 12 for x86_64<br><br>(*SP3はカーネルを4.4.82-6.9.1にアップグレ</td><td>12 SP1<br>12 SP2<br>12 SP3*<br>12 SP4</td><td>3.12.49-11.1<br>4.4.21-69.1<br>4.4.82-6.9.1<br>4.12.14-94.41.1</td></tr>
</table>
</td></tr>
</table>

| ディストリビューション/バージョン | サポート対象のバージョン | サポート対象のカーネル |
|---|---|---|
| *ードしてください)* | 12 SP5<br>※SLES12.0はサポート対象外 | 4.12.14-120.1 |
| SUSE Linux Enterprise Server 15 for x86_64<br>*(*DataKeeper にセクターサイズが奇数のディスクは利用できません)*<br>*(*SLES12からSLES15のようにKernelをメジャーバージョンアップすることはサポートされません)* | 15*<br>15 SP1<br>15 SP2 | 4.12.14-23.1<br>4.12.14-195.1<br>5.3.18-14.1 |
| Oracle Linux<br>*(*DataKeeper 非同期ミラーは、OL 7.4～7.6 の一部のカーネルではサポートされていません 詳細については、ここをクリックしてください。 )*<br>*(**UEK6はカーネルを5.4.17-2102.202.5にアップグレードしてください)* | 7<br>7.1<br>7.2<br>7.3<br>7.4*<br>7.5*<br>7.6*<br>7.7<br>7.8<br>7.9<br>UEK R3<br>UEK R4<br>UEK R5<br>UEK R6** | 3.10.0-123.el7<br>3.10.0-229.el7<br>3.10.0-327.el7<br>3.10.0-514.el7<br>3.10.0-693.el7<br>3.10.0-862.el7<br>3.10.0-957.el7<br>3.10.0-1062.el7<br>3.10.0-1127.el7<br>3.10.0-1160.el7<br>3.8.13-16.2.1.el7uek<br>4.1.12-37.3.1.el7uek<br>4.14.35-1818.3.3.el7uek<br>5.4.17-2102.202.5.el7uek |
| Oracle Linux<br>*(OL 7 から OL 8 のようにKernelをメジャーバージョンアップすることはサポートされません)*<br>*(*UEK6はカーネルを5.4.17-2102.202.5にアップグレードしてください)* | 8.0<br>8.1<br>8.2<br>8.3<br>UEK R6* | 4.18.0-80.el8<br>4.18.0-147.el8.x86_64<br>4.18.0-193.el8.x86_64<br>4.18.0-240.el8.x86_64<br>5.4.17-2102.202.5.el8uek |

<table>
<tr>
<td></td>
<td>
<table>
<tr><th>ディストリビューション/バージョン</th><th>サポート対象のバージョン</th><th>サポート対象のカーネル</th></tr>
<tr><td>CentOS<br><br>(*DataKeeper 非同期ミラーは、CentOS 7.4～7.6 の一部のカーネルではサポートされていません 詳細については、ここをクリックしてください。 )</td><td>7<br>7.1<br>7.2<br>7.3<br>7.4*<br>7.5*<br>7.6*<br>7.7<br>7.8<br>7.9</td><td>3.10.0-123.el7<br>3.10.0-229.el7<br>3.10.0-327.el7<br>3.10.0-514.el7<br>3.10.0-693.el7<br>3.10.0-862.el7<br>3.10.0-957.el7<br>3.10.0-1062.el7<br>3.10.0-1127.el7<br>3.10.0-1160.el7</td></tr>
<tr><td>CentOS<br><br>(CentOS 7 から CentOS 8 のようにKernelをメジャーバージョンアップすることはサポートされません)</td><td>8.0<br>8.1<br>8.2<br>8.3</td><td>4.18.0-80.el8<br>4.18.0-147.el8.x86_64<br>4.18.0-193.el8.x86_64<br>4.18.0-240.el8.x86_64</td></tr>
</table>
<br>注記: このリストのサポート対象のディストリビューションおよびカーネルは、LifeKeeper のみを考慮したものです。お使いのサーバーおよびストレージハードウェアについては、各メーカーがサポートするディストリビューションおよびカーネルに従ってください。
</td>
</tr>
<tr>
<td>LUN のサポート</td>
<td>
Linux の SCSI ドライバには、論理ユニット (LUN) の検索対象とするデバイスを制御するいくつかのパラメータがあります。
<ul>
<li>LUN をサポートしないデバイスのリスト – このリストのデバイスは LUN をサポートしないことが既知であるため、SCSI ドライバはこれらのデバイスに対して LUN を検索することを許可しません。</li>
<li>LUN をサポートするデバイスのリスト – このリストのデバイスは LUN をサポートすることが既知であるため、必ず LUN を検索します。</li>
<li>Probe all LUNs on each SCSI device – デバイスがどちらのリストにも存在しない場合、検索するかどうかを指定します。このパラメータは、make config を使用して SCSI モジュールセクションで設定します。</li>
</ul>
(SUSE を含む) ほとんどのディストリビューションでは、Probe all LUNs 設定はデフォルトで有効になっていますが、Red Hat ではデフォルトで無効に設定されています。LifeKeeper の構成でデータ保護を目的として通常使用される外部 RAID コントローラに
</td>
</tr>
</table>

| | |
|---|---|
| | は、多くの場合、複数の LUN (論理ユニット) が設定されます。LUN のサポートを有効にするには、このフィールドを選択してカーネルを再構築する必要があります。<br><br>カーネルやモジュールを再構築せずに Probe all LUNs を有効にするには、変数 max_scsi_luns を 255 に設定します (これによって最大 255 個の LUN をスキャンするようになります)。SCSI ドライバがモジュールになっているカーネル (Red Hat など) で max_scsi_luns を設定するには、*/etc/modules.conf* に以下のエントリを追加し、初期 RAM ディスクを再構築して、再起動してからその RAM ディスクを読み込みます。<br><br>`options scsi_mod max_scsi_luns=255`<br><br>SCSI ドライバがコンパイルされるカーネル (SUSE など) で max_scsi_luns を設定するには、*/etc/lilo.conf* に以下のエントリを追加します。<br><br>`append="max_scsi_luns=255"`<br><br>**注記**: 255 個の LUN をスキャンすると、デバイスによってはブートのパフォーマンスに悪影響を与える可能性があります (特に、BLIST_SPARSELUN が指定されたデバイス)。Dell PV650F というアレイではそのような状況が発生しました。このパフォーマンスの問題を回避するには、アレイ上で設定した LUN の最大数 (16 または 32 など) を max_scsi_luns に設定します。例えば、以下のようになります。<br><br>`append="max_scsi_luns=16"` |
| **チャンネルボンディング、ネットワークチーミングの動作確認環境** | LifeKeeperでは、以下の設定のチャンネルボンディング、またはネットワークチーミングを用いた環境での動作確認を行なっています。<br><br>• チャンネルボンディングにおけるボンディングポリシー<br><br>+ balance-rr<br><br>+ active-backup<br><br>• ネットワークチーミングにおけるランナー<br><br>+ round-robin<br><br>+ active-backup |

# 5.4.2.6. データレプリケーションの設定

| 項目 | 説明 |
| --- | --- |
| **SIOS DataKeeper の機能** | SIOS DataKeeper は、Linux カーネルの機能を利用しています。そのため、最低要件としてカーネルバージョン 2.6 以降である必要があります。<br>利用可能な OS は サポートマトリックス を参照してください。 |
| **SIOS DataKeeper ドキュメンテーション** | SIOS DataKeeper のドキュメンテーションは、SIOS Technology Corp. の Web サイトにある「SIOS Protection Suite テクニカルドキュメンテーション」の中に収録されています。 |

# 5.4.2.7. ネットワーク設定

| 項目 | 説明 |
| --- | --- |
| ルーティングテーブルに対する IP Recovery Kit の影響 | LifeKeeper が保護する IP アドレスは、論理インターフェースとして、Linux 上で実装されます。Linux 上で論理インターフェースを設定すると、その論理インターフェースに関連付けられたサブネットへのルートが自動的にルーティングテーブルに追加されます。例えば、物理インターフェースによってそのサブネットへのルートがすでに存在する場合も同様です。この追加により、同じサブネットに対して複数のルーティングテーブルエントリが作成される可能性があります。<br><br>接続元のアドレスを検査して確認するアプリケーションの場合、複数のルーティングテーブルエントリがあると、LifeKeeper システムが (LifeKeeper がインストールされていない) 他のシステム上のそのようなアプリケーションに接続しようとしたときに問題が発生することがあります。複数のルーティングテーブルエントリによって、物理インターフェースからではなく論理インターフェースから接続が張られているように見えます。 |
| IP サブネットマスク | LifeKeeper 保護下の IP 設定では、物理インターフェースの IP アドレスと、LifeKeeper が保護するエイリアス IP アドレスのサブネットを同じにする場合、2 つのアドレスのサブネットマスクを同じにする必要があります。サブネットマスクの設定を間違えると、LifeKeeper GUI のクライアントとサーバ間の接続に遅延や障害が発生します。 |
| EEpro100 ドライバの初期化 | Intel Ethernet インターフェースを搭載するシステムでは、eepro100 ドライバの初期化の問題を解決するために、Intel e100 ドライバをインストールする必要があります。eepro100 ドライバを使用すると、ブート時にインターフェースが起動したときに以下のエラーが発生し、インターフェースをシャットダウンするまでエラーを出し続けることがあります。<br><br>eth0: card reports no Rx buffers<br><br>eth0: card reports no resources |

# 5.4.2.8. アプリケーションの設定

| 項目 | 説明 |
| --- | --- |
| データベース初期設定ファイル | データベースの初期設定ファイルは、共有デバイス上に置いてローカルファイルシステムの指定場所にシンボリックリンクを作成するか、または個別のシステム上に保持して変更を適用する必要がある場合に手動で両方のシステムを更新するかのいずれかに必要があります。 |
| **Oracle** のローカルマウントポイント | Oracle のローカル環境は、 *internal* として接続するか、 *sysdba* として接続するかによって異なります。データベースを LifeKeeper の保護下に置く場合、「`connect / as sysdba`」を使用してローカルマウントポイント上にデータベースを作成する必要があります。 |
| **Apache** のアップデート | Linux オペレーティングシステムのアップグレードの一環として、SPS が保護する Apache アプリケーションをアップグレードするには、起動時にデフォルトサーバインスタンスを無効にする必要があります。<br><br>設定ファイル *(httpd.conf)* がデフォルトのディレクトリ *(/etc/httpd/conf)* にある場合、Red Hat のアップグレードにより設定ファイルが上書きされます。したがって、アップグレードする前にファイルのコピーを作成し、アップグレードした後にファイルをリストアする必要があります。<br><br>また、*Apache Web Server Recovery Kit 管理ガイド* の「Apache Web Server の設定に関する考慮事項」セクションも参照してください。 |

# 5.4.2.9. ストレージとアダプタの設定

| 項目 | 説明 |
| --- | --- |
| マルチパス I/O 冗長コントローラ | マルチパス I/O のソリューションには数種類あり、すでに利用可能なものや Linux 環境向けに開発中のものなどがあります。SIOS Technology Corp. は、多くのサーバベンダ、アダプタベンダ、およびドライバ開発者と積極的に協力することで、LifeKeeper とマルチパス I/O ソリューションとの協調動作を実現しています。データの整合性を保護するために LifeKeeper が使用する SCSI リザベーションは、特殊な要件を必要とするため、マルチパス I/O ソリューションの最初の実装では多くの場合要件が満たされません。<br><br>ディスクアレイサポートに関する以下の技術情報を参照し、個別のアレイがマルチパスおよび特定のマルチパスソリューションでサポートされているかを判断してください。マルチパスおよび特定のマルチパスソリューションと共に動作する LifeKeeper のサポート対象として一覧に指定されていないアレイは、サポート対象ではないと考えてください。 |
| マルチパス構成での大量の I/O | マルチパス構成では、パスの操作中に大量の I/O を実行すると、システムが応答しなくなったように見えることがあります。マルチパスのソフトウェアが LUN のアクセスをあるパスから別のパスに移動する場合、処理中の I/O も新しいパスに移動させる必要があります。この I/O の経路変更は、その I/O の応答時間の遅延を発生させます。この間にさらに I/O が発行されると、それらはシステム内のキューとなり、システムはプロセス用のメモリを使い果たしてしまう可能性があります。非常に高負荷の I/O の下では、これらの遅延と低メモリ状態によってシステムが無応答になり、LifeKeeper がこれをサーバのダウンとして検知し、フェイルオーバを開始することがあります。<br><br>この問題が発生する頻度には多くの要因が影響を及ぼします。<br><br>• プロセッサの速度は、I/O がキューに保持される速さに影響します。高速なプロセッサでは、障害が検知される頻度が高くなります。<br>• システムメモリの搭載量は、システムが無応答になるまでにキューに保持できる I/O の数に影響します。 メモリが多いシステムでは、障害が検知される頻度が低くなります。<br>• 使用する LUN の数は、キューに保持できる I/O の量に影響します。<br>• I/O の特性は、キューに保持される I/O の量に影響します。問題が発生したテストケースでは、ディスクにデータを無制限に書き込んでいました。ほとんどのアプリケーションは、データの読み取りと書き込みの両方を行うはずです。フェイルオーバを待って読み取りがブロックされることで書き込みも抑制され、結果的に I/O 速度 |

| | |
|---|---|
| | が減少して障害検知の頻度が低くなります。<br><br>例えば、RDAC を使用した IBM DS4000 のマルチパス構成のテストでは、DS4000 への I/O スループットを毎秒 190 MB 以上にしてパス障害をシミュレーションした場合に LifeKeeper は約 12 回に 1 回サーバの障害を (誤) 検出しました。このテストでは、サーバとして IBM x345 (デュアル Xeon 2.8GHz プロセッサとメモリ 2 GB を搭載) を使用し、DS4400 に接続して使用サーバ当たり 8 ボリューム (LUN) にしました。フェイルオーバを抑止するために、LifeKeeper の LCMNUMHBEATS パラメータ (`/etc/default/LifeKeeper` 内) を 16 に増やしました。このパラメータの変更により、無応答のシステムが生存していないと判定するまでに LifeKeeper は、デフォルトの約 15 秒ではなく、約 80 秒間待機するようになります。 |
| 大規模ストレージ構成の場合のスイッチオーバに関する特別な考慮事項 | いくつかの大規模ストレージ構成 (例えば、複数の論理ボリュームグループがあり、各ボリュームグループ内に 10 以上の LUN を持つ構成) では、LifeKeeper は障害を検出したときにデフォルトの 300 秒のタイムアウト時間内に sendevent を完了することができない場合があります。その結果、バックアップシステムへのスイッチオーバが失敗します。サービス状態にならないリソースが生じ、LifeKeeper のログにエラーメッセージが記録されます。<br><br>大規模ストレージ構成では、/etc/default/LifeKeeper ファイルの SCSIERROR を「event」から「halt」に変更することを推奨します。これにより LifeKeeper は SCSI エラーの発生時に「halt」を実行します。LifeKeeper はバックアップシステムへのフェイルオーバに成功するようになります。 |
| HP 3PAR StoreServ 7200 FC | HP 3PAR StoreServ 7200 は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>QLogic QMH2572 8Gb FC HBA for HP BladeSystem c-Class(ファームウェアバージョン5.06.02 (90d5) を使用した)HP 3PAR StoreServ 7200 (ファームウェア (HP 3PAR OS) バージョン 3.1.2), ドライババージョン 8.03.07.05.06.2-k (RHELにバンドル) + DMMP (device-mapper-1.02.66-6, device-mapper-multipath-0.4.9-46.el6)<br><br>テストはRHEL 6.2 (x86_64) を使用したSPS for Linux v8.1.1 で実施されました。<br><br>**注記:** 3PAR StoreServ 7200 は、デフォルトのパスチェッカとのリザベーション競合を返します。この競合を回避するには、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。 |

| | |
|---|---|
| | **DMMP_REGISTRATION_TYPE=hba** |
| HP 3PAR StoreServ 7400 FC | HP 3PAR StoreServ 7400は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>HP 3PAR StoreServ 7400 (ファームウェア (HP 3PAR OS) バージョン 3.1.2) + HP DL380p Gen8 + Emulex LightPulse Fibre Channel SCSI HBA (ドライババージョン 8.3.5.45.4p) + DMMP (device-mapper-1.02.66-6, device-mapper-multipath-0.4.9-46.el6)<br><br>テストはRHEL 6.2 (x86_64) を使用したSPS for Linux v8.1.1 で実施されました。<br><br>**注記:** 3PAR StoreServ 7400 は、デフォルトのパスチェッカとのリザベーション競合を返します。この競合を回避するには、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。<br><br>**DMMP_REGISTRATION_TYPE=hba**<br><br>およびユーザーフレンドリーデバイスマッピングはサポートされません。“multipath.conf” に次のパラメータを設定してください。<br><br>**“user_friendly_names no”** |
| HP 3PAR StoreServ 7400 iSCSI (DMMP Recovery Kit を使用したマルチパス構成) | HP 3PAR StoreServ 7400は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>HP 3PAR StoreServ 7400 (ファームウェア (HP 3PAR OS) version 3.1.3) + HP Ethernet 10Gb 2-port 560SFP+ Adapter (Networkdriver ixgbe-3.22.0.2) + iSCSI (iscsi-initiator-utils-6.2.0.873-10.el6.x86_64)、DMMP (device-mapper-1.02.79-8.el6、device-mapper-multipath-0.4.9-72.el6).<br><br>**注記** : 3PAR StoreServ 7400 iSCSI は、リザベーション競合を返す場合があります。この競合を回避するため、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。<br><br>**DMMP_REGISTER_IGNORE=TRUE** |
| HP 3PAR StoreServ 7400 iSCSI(Quorum/Witness Kit 使用) | HP 3PAR StoreServ 7400は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。 |

| | |
|---|---|
| | iSCSI (iscsi-initiator-utils-6.2.0.872-21.el6.x86_64) + DMMP (device-mapper-multipath-0.4.9-41 + device-mapper-1.02.62-3) + nx_nic v4.0.588.<br><br>DMMP と DMMP Recovery Kit を RHEL 6.1 で利用する場合には、Quorum/Witness Server Kit と STONITH 機能を利用する必要があります。SCSI リザベーションを無効にするため、“/etc/default/LifeKeeper“に RESERVATIONS=none を追記する必要があります。<br><br>サーバは IPMI 2.0 に準拠したインタフェースを備えている必要があります。 |
| HP 3PAR StoreServ 10800 FC | HP 3PAR StoreServ 10800 FC は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>ファームウェア (HP 3PAR OS) バージョン3.1.2 + HP DL380p Gen8 + Emulex LightPulse Fibre Channel HBA (ドライババージョン 8.3.5.45.4p) + DMMP (device-mapper-1.02.66-6, device-mapper-multipath-0.4.9-46 el6)<br><br>テストはRHEL 6.2 (x86_64) を使用したSPS for Linux v8.1.2 で実施されました。<br><br>**注記:** 3PAR StoreServ 10800 FC は、デフォルトのパスチェッカとのリザベーション競合を返します。この競合を回避するには、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。<br><br>**DMMP_REGISTRATION_TYPE=hba**<br><br>およびユーザーフレンドリーデバイスマッピングはサポートされません。“multipath.conf” に次のパラメータを設定してください。<br><br>**“user_friendly_names no”** |
| HP MSA1040/2040fc | HP MSA 2040 Storage FC は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>HP MSA 2040 Storage FC (ファームウェアGL101R002) + HP SN1000Q 16Gb 2P FC HBA QW972A (ファームウェアバージョン 6.07.02, ドライババージョン 8.04.00.12.06.0-k2 (RHELにバンドル)) + DMMP (device-mapper-1.02.74-10, device-mapper-multipath-0.4.9-56).<br><br>テストは RHEL 6.3 (X86_64) を使用した LifeKeeper for Linux v8.1.2 で実施されました。 |

| | |
|---|---|
| HP P9500/XP | SIOS LifeKeeper for Linux v7.2 以降を使用する場合について、Hewlett-Packard 社により認定。テストに使用されたモデルは HP P9500/XP です。以下の環境の LifeKeeper で動作することが認定されています。<br><br>• Red Hat Enterprise for 32-bit, x64 (64-bit; Opteron および Intel EMT64)<br>RHEL 5.3, RHEL 5.4, RHEL 5.5<br>• SuSE Enterprise Server for 32-bit, x64 (64-bit; Opteron および Intel EMT64)<br>SLES 10 SP3, SLES 11, SLES 11 SP1<br>• ネイティブまたは内蔵のクラスタリングソリューション RHCS および SLE HA |
| HP StoreVirtual 4330 iSCSI (DMMP Recovery Kit を使用したマルチパス構成) | HP StoreVirtual 4330 は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>HP StoreVirtual 4330 (ファームウェアHP LeftHand OS 10.5) + HP Ethernet 1Gb 4-port 331FLR (Networkdriver tg3-3.125g) + iSCSI (iscsi-initiator-utils-6.2.0.872-41.el6.x86_64)、 DMMP (device-mapper-1.02.74-10.el6,device-mapper-multipath-0.4.9-56.el6) |
| HP StoreVirtual(LeftHand) シリーズ OS (SAN/iQ) バージョン 11.00 iSCSI | HP 社 StoreVirtual(LeftHand) ストレージにおいて、OS(SAN/iQ) バージョン 11.00 をサポートします。本バージョンが使われている全ての StoreVirtual シリーズがサポート対象であり、仮想ストレージ・アプライアンスである StoreVirtual VSA も含まれます。<br><br>テスト構成は次の通りです。<br><br>StoreVirtual VSA(11.0.00.1263.0) + RHEL 6.4(x86_64) + DMMP (device-mapper-1.02.77-9.el6.x86_64, device-mapper-multipath-0.4.9-64.el6.x86_64) |
| HP StoreVirtual 4730 iSCSI (DMMP Recovery Kit を使用したマルチパス構成) | HP StoreVirtual 4730 は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>HP StoreVirtual 4730 (ファームウェアHP LeftHand OS 11.5) + HP FlexFabric 10Gb 2-port 536FLB Adapter (Networkdriver bnx2x-1.710.40) + iSCSI (iscsi-initiator-utils-6.2.0.873-10.el6.x86_64)、 DMMP (device-mapper-1.02.79-8.el6,device-mapper-multipath-0.4.9-72.el6) |
| HP StoreVirtual(LeftHand) シリーズ LeftHand OS バージョン 11.5 iSCSI (DMMP Recove | HP 社 StoreVirtual(LeftHand) ストレージにおいて、LeftHand OS バージ |

| | |
|---|---|
| ry Kitを使用したマルチパス構成) | ョン 11.5 をサポートします。本バージョンが使われている全ての Store Virtual シリーズがサポート対象であり、仮想ストレージ・アプライアンスである StoreVirtual VSA も含まれます。<br><br>テスト構成は次の通りです。<br><br>StoreVirtual 4730(11.5.00.0673.0) + RHEL 6.5(x86_64) + DMMP (device-mapper-1.02.79-8.el6.x86_64, device-mapper-multipath-0.4.9-72.el6.x86_64) |
| HP StoreVirtual 4330 iSCSI (Quorum/Witness Kit　使用) | HP StoreVirtual 4330 は、 以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>HP StoreVirtual 4330 (ファームウェア HP LeftHand OS 10.5) + iSCSI (iscsi-initiator-utils-6.2.0.872-41.el6.x86_64)、 bonding(バージョン: 3.6.0)、tg3(バージョン: 3.125g) SCSI リザベーションを無効にするには、”/etc/default/LifeKeeper” に RESERVATIONS=noneを設定します。 |
| IBM San Volume Controller (SVC) | シングルパス構成において、パートナーテストにより認定。SDD Recovery Kit および Device Mapper Multipath Recovery Kit を使用するマルチパス設定において、SIOS Technology Corp. により認定。 |
| IBM Storwize V7000 iSCSI | パートナーテストによりiSCSI (iscsi-initiator-utils-6.2.0.872-34.el6.x86_64) と DMMP (device-mapper-1.02.66-6.el6、device-mapper-multipath-0.4.9-46.el6) を使用する IBM Storwize V7000 (Firmware Version 6.3.0.1)を 認定しています。 テストは、LifeKeeper for Linux v7.5 と RHEL 6.2 を使用して行われました。<br><br>**制限事項:** IBM Storwize V7000 は、Quorum/Witness Server Kit および STONITH と組み合わせて使用する必要があります。/etc/default/LifeKeeper 内で以下の設定により、SCSI リザベーションを無効にしてください。<br><br>RESERVATIONS=none |
| IBM Storwize V7000 FC | パートナテストにより Red Hat Enterprise Linux Server Release 6.2 (Tikanga)、HBA: QLE2562 DMMP: 0.4.9-46 を組み合わせたマルチパス構成で認定済み。 |
| IBM Storwize V3700 FC | パートナーテストにより Red Hat Enterprise Linux Server Release 6.5 (Santiago)、HBA: QLE2560、DMMP: 0.4.9-72 を組み合わせたマルチパス構成で認定済み。 |

| | |
|---|---|
| IBM XIV Storage System | パートナテストにより Red Hat Enterprise Linux Sever Release 5.6、HBA: NEC N8190-127 Single CH 4Gbps (Emulex LPe1150 相当品)、XIV Host Attachement Kit: バージョン 1.7.0 を組み合わせたマルチパス構成でのみ認定済み。<br><br>**注記:** LifeKeeper を使用する IBM XIV Storage System で 32 個以上の LUN を作成する必要がある場合、詳細については IBM の営業担当者にお問い合わせください。 |
| Dell EqualLogic PS4000/4100/4110/6000/6010/6100/6110/6500/6510 | Dell EqualLogic は、SIOS Technology Corp. パートナーによってテスト済みです。テスト構成は次の通りです。Dell EqualLogic PS4000/4100/6000/6100/6110/6500/6510 + DMMP + DMMP Recovery Kit + RHEL 5.3 + iscsi-initiator-utils-6.2.0.868-0.18.el5。LUN 数が大きい場合 (20 以上) は、/etc/default/LifeKeeper の REMOTETIMEOUT 設定を REMOTETIMEOUT=600 に変更してください。 |
| 富士通<br><br>ETERNUS DX60 S2 / DX80 S2 / DX90 S2<br><br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX410 S2 / DX440 S2I<br><br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX8100 S2/DX8700 S2<br><br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX100 S3/DX200 S3<br><br>• iSCSI、シングルパスおよびマルチパス構成 | マルチパス環境にてDMMP ARKを利用する際、 /etc/multipath.conf に以下の設定が必要となります。<br><br>`prio alua`<br><br>`path_grouping_policy group_by_prio`<br><br>`failback immediate`<br><br>`no_path_retry 10`<br><br>`Path_checker tur` |

| | |
|---|---|
| ETERNUS DX500 S3/DX600 S3<br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX200F<br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX60 S3<br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS AF250 / AF650<br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX60 S4 / DX100 S4 / DX200 S4<br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX500 S4 / DX600 S4 / DX8900S4<br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS AF250 S2 / AF650 S2<br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX60 S5 / DX100 | |

| | |
|---|---|
| S5 / DX200 S5<br><br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS DX500 S5 / DX600 S5 / DX900S5<br><br>• iSCSI、シングルパスおよびマルチパス構成<br><br>ETERNUS AF150 S3 / AF250 S3 / AF650 S3<br><br>• iSCSI、シングルパスおよびマルチパス構成 | |
| 富士通<br><br>ETERNUS DX60 S2 / DX80 S2 / DX90 S2<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX410 S2 / DX440 S2<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX8100 S2/DX8700 S2<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX100 S3/DX200 S3/DX500 S3/DX600S3 | マルチパス環境にてDMMP ARKを利用する際、 /etc/multipath.conf に以下の設定が必要となります。<br><br>`prio alua`<br><br>`path_grouping_policy group_by_prio`<br><br>`failback immediate`<br><br>`no_path_retry 10`<br><br>`Path_checker tur`<br><br>マルチパス環境にてETERNUSマルチパスドライバを利用する際、設定ファイルにパラメータを設定する必要はありません。 |

| | |
|---|---|
| • ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX200F<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX60 S3<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX8700 S3 / DX8900 S3<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS AF250 / AF650<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX60 S4 / DX100 S4 / DX200 S4<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX500 S4 / DX600 S4 / DX8900S4<br><br>• ファイバチャネル、シングルパスおよびマルチパス構成 | |

| | |
|---|---|
| ETERNUS AF250 S2 / AF650 S2<br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX60 S5 / DX100 S5 / DX200 S5<br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS DX500 S5 / DX600 S5 / DX900S5<br>• ファイバチャネル、シングルパスおよびマルチパス構成<br><br>ETERNUS AF150 S3 / AF250 S3 / AF650 S3<br>• ファイバチャネル、シングルパスおよびマルチパス構成 | |
| NEC iStorage M10e iSCSI (SPS Recovery Kit を使用したマルチパス構成) | NEC iStorage M10e iSCSI は、以下の構成で SIOS Technology Corp. のパートナーによってテスト済みです。<br><br>NEC iStorage M10e iSCSI + 1GbE NIC + iSCSI (iscsi-initiator-utils-6.2.0.873-10.el6.x86_64)、SPS (sps-utils-5.3.0-0.el6,sps-driver-E-5.3.0-2.6.32.431.el6) |
| NEC iStorage Storage Path Savior Multipath I/O | **マルチパスデバイスを使用するアプリケーションとファイルシステムの保護:** SPS デバイスを使用するアプリケーションやファイルシステムを SPS によって設定し保護するには、SPS Recovery Kit をインストールする必要があります。<br><br>SPS Kit のインストール後は、1 つ以上のマルチパスデバイスノードを使用するアプリケーション階層を作成するだけで、SPS Kit が提供する新 |

しいリソースタイプが自動的に組み込まれます。

**マルチパスデバイスノード:** SPS Kit を使用するには、すべてのファイルシステムおよび Raw デバイスをネイティブの /dev/sd* デバイスノードではなく、マルチパスデバイスノード (/dev/dd*) 上にマウントまたは設定する必要があります。

**SCSI-3 Persistent Reservations の使用:** SPS Kit は、リザベーションタイプを「書き込み専用」とする SCSI-3 Persistent Reservations を使用します。この場合、クラスタのあるノードが予約したデバイスは、クラスタの他のノードから読み取り可能のままですが、他のノードからデバイスへの書き込みはできなくなります。このことは、それらの他のノード上で進行中の読み取り専用アクセスのためにファイルシステムをマウントできるという意味ではないことに注意してください。

LifeKeeper では、sg_persist ユーティリティを使用してパーシステントリザベーションを発行し、監視します。必要であれば、LifeKeeper は sg_persist(8) ユーティリティをインストールします。

**テスト環境:** SPS Kit は、Emulex HBA および Emulex lpfc ドライバを使用する NEC iStorage ディスクアレイにおいて、テストおよび認定済みです。SPS Kit は、SPS がサポートする他の NEC iStorage D、S、および M でも同様に問題なく動作すると考えられます。

**[テストされた Emulex HBA]**

iStorage D-10
==========
LP952
LP9802
LP1050
LP1150
==========

iStorage M100
==========
LPe1150
LPe11002
LPe1250
LPe12002
LPe1105
LPe1205
==========

**マルチパスソフトウェアの要件:** SPS Kit は、SPS for Linux 3.3.001 を使用してテスト済みです。インストールされている SPS パッケージに対す

| | |
|---|---|
| | る既知の依存関係はありません。<br><br>**インストール要件:** SPS Recovery Kit をインストールする前に SPS ソフトウェアをインストールする必要があります。<br><br>**SPS パスの追加または修復:** LifeKeeper は、SPS リソース起動する場合に、パーシステントリザベーションを確立してその時点でアクティブなパスに登録します。最初のリザベーションの後に新しいパスが追加されるか、障害が起きたパスが修復されて SPS がそのパスを自動的に再度アクティブにした場合、そのパスは、LifeKeeper が SPS リソースに対する次の quickCheck を実行するまでリザベーションの一部として登録されません。その時点までに SPS がそのパスに対する書き込みを許可した場合、リザベーションコンフリクトが発生し、システムのメッセージファイルに競合が記録されます。SPS ドライバは、登録されたパスでそれらの I/O を再試行するため、アプリケーションにとっては検出可能な障害になりません。quickCheck によるパスの登録が完了すると、その後の書き込みは成功します。 |
| Pure Storage FA-400 シリーズ FC (DMMP Recovery Kit を使用したマルチパス構成) | パートナーテストにより、DMMP Recovery Kit を利用したFC接続のマルチパス構成で認定済みです。 |
| QLogic ドライバ | QLogic アダプタを使用するサポート対象の他のファイバチャネルアレイについては、qla2200 または qla2300 ドライバのバージョン 6.03.00 以降を使用してください。 |
| Emulex ドライバ | サポート対象の Emulex のファイバチャネル HBA については、lpfc ドライバ v8.0.16 以降を使用してください。 |
| Adaptec 29xx ドライバ | Adaptec 29xx を使用するサポート対象の SCSI アレイについては、OS ディストリビューションに付属の aic7xxx ドライバのバージョン 6.2.0 以降を使用してください。 |

## HP のマルチパス I/O 設定

| 項目 | 説明 |
|---|---|
| Secure Path を使用するマルチパスクラスタのインストール | Secure Path を使用するマルチパスクラスタを新規にインストール場合は、次の手順を実行します。<br><br>1. 選択した OS を各サーバにインストールします。<br>2. クラスタハードウェア (FCA2214 アダプタ、ストレージ、スイッチ、およびケーブル) をインストールします。<br>3. HP Platform Kit をインストールします。<br>4. HP Secure Path ソフトウェアをインストールします。ここでシステムをリブートする必要があります。Secure Path からストレージへのパスを適切に設定したこと |

| | |
|---|---|
| | を確認してください。詳細については、Secure Path のドキュメンテーションを参照してください。<br>5. LifeKeeper をインストールします。 |
| Secure Path による永続的デバイスノード | Secure Path は、/dev/spdev/spXX (XX はデバイス名) の形式の「永続的な」デバイスノードをサポートします。これらのノードは、特定の SCSI デバイスノード /dev/sdXX へのシンボリックリンクです。LifeKeeper はこれらのノードを「通常の」SCSI デバイスノード /dev/sdXX であるかのように認識します。LifeKeeper は、デバイスが /dev/sda1 か /dev/sdq1 かを直接検出し、その後正しいデバイスノードを直接使用することにより、リブートおよびクラスタノードをまたがってデバイス名の永続性を独自に維持しています。<br><br>**注記:** SCSI デバイスノードへのシンボリックリンクのサポートは、LifeKeeper v4.3.0 で追加されました。 |
| アクティブ / パッシブコントローラおよびコントローラスイッチオーバ | MSA1000 では、一方のコントローラをアクティブに、他方のコントローラをスタンバイモードにすることによってマルチパスを実装しています。アクティブなコントローラまたはアクティブなコントローラへのパスのいずれかに問題が起きた場合、スタンバイコントローラがアクティブ化されて処理を引き継ぎます。コントローラをアクティブにする場合、コントローラの準備ができるまでにある程度の時間がかかります。アレイ上で設定されている LUN の数に応じて、30 ~ 90 秒の時間を必要とします。この間、ストレージへの I/O は、新しくアクティブになるコントローラに経路変更できるようになるまでブロックされます。 |
| 起動時にシングルパスでも通知が発生しない | システムがロードされたときに、サーバがシングルパスでしかストレージにアクセスできない場合でも、この問題に関する通知が発生しません。この問題は、システムがリブートしたときに上記のような物理的なパスの障害が起きると発生しますが、一時的なパス障害でも発生しています。システムをロードするときは、管理者はストレージへのすべてのパスが正しく構成されたことを必ず確認し、構成されていない場合は、ハードウェアの問題を修復するか、システムをリロードして一時的な問題を解決するかいずれかのアクションを取ることを推奨します。 |

## 日立 HDLM のマルチパス I/O 設定

| 項目 | 説明 |
|---|---|
| マルチパスデバイスを使用するアプリケーションとファイルシステムの保護 | HDLM デバイスを使用するアプリケーションやファイルシステムを LifeKeeper によって設定し保護するには、HDLM Recovery Kit をインストールする必要があります。<br><br>HDLM Kit のインストール後は、マルチパスデバイスノードを 1 つ以上使用するアプリケーション階層を作成するだけで、HDLM Kit が提供する新しいリソースタイプが自動的に組み込まれます。 |
| マルチパスデバイスノード | HDLM Kit を使用するには、すべてのファイルシステムおよび Raw デバイスをネイティブの /dev/sd* デバイスノードではなく、マルチパスデバイスノード (/dev/sddlm*) 上にマウン |

| | |
|---|---|
| | トするか、設定する必要があります。 |
| SCSI-3 Persistent Reservations の使用 | HDLM Kit は、リザベーションタイプを「書き込み専用」とする SCSI-3 Persistent Reservations を使用します。この場合、クラスタのあるノードが予約したデバイスは、クラスタの他のノードから読み取り可能のままですが、他のノードからデバイスへの書き込みはできなくなります。このことは、それらの他のノード上で進行中の読み取り専用アクセスのためにファイルシステムをマウントできるという意味ではないことに注意してください。<br><br>LifeKeeper では、sg_persist ユーティリティを使用してパーシステントリザベーションを発行し、監視します。必要であれば、LifeKeeper は sg_persist(8) ユーティリティをインストールします。 |
| ハードウェア要件 | HDLM Kit は、QLogic qla2432 HBA と 8.02.00-k5-rhel5.2-04 ドライバ、および Silkworm3800 FC スイッチを使用する 日立 SANRISE AMS1000 ディスクアレイにおいて、テストおよび認定済みです。HDLM Kit は、他の日立ディスクアレイでも同様に問題なく動作すると考えられます。HDLM Kit は、SANRISE AMS シリーズ、SANRISE USP、および日立の VSP においても、テストおよび認定済みです。HBA および HBA ドライバは HDLM がサポートするものを使用してください。<br><br>BR1200 は、Hitachi Data Systems により認定済みです。シングルパスとマルチパスの両方の設定で、RDAC ドライバが必要です。RDAC ドライバを使用する BR1200 設定のみをサポートします。HDLM (HDLM ARK) を使用する BR1200 設定はサポートしません。 |
| マルチパスソフトウェアの要件 | HDLM Kit は、以下の HDLM for Linux でテスト済みです。<br><br>05-80, 05-81, 05-90, 05-91, 05-92, 05-93, 05-94, 6.0.0, 6.0.1, 6.1.0, 6.1.1, 6.1.2, 6.2.0, 6.2.1, 6.3.0, 6.4.0, 6.4.1, 6.5.0, 6.5.1, 6.5.2, 6.6.0, 6.6.2, 7.2.0, 7.2.1, 7.3.0, 7.3.1, 7.4.0, 7.4.1, 7.5.0, 7.6.0, 7.6.1, 8.0.0, 8.0.1, 8.1.0, 8.1.1, 8.1.2, 8.1.3, 8.1.4, 8.2.0, 8.4.0, 8.5.0, 8.5.1, 8.5.2, 8.5.3, 8.5.4, 8.6.0, 8.6.1, 8.6.2, 8.6.4, 8.6.5, 8.7.0, 8.7.1, 8.7.2, 8.7.3, 8.7,4, 8.7.6, 8.7.7, 8.7.8, 8.8.0, 8.8.1<br><br>インストールされている HDLM パッケージに対する既知の依存関係はありません。<br><br>**注記:** HDLM 6.0.0 以降から製品名が「Hitachi Dynamic Link Manager Software (HDLM)」に変更されました。6.0.0 (05-9X) より古いバージョンでは、「Hitachi HiCommand Dynamic Link Manager (HDLM)」という製品名です。<br><br>**注記:** HDLM version 6.2.1 以降は、HDLM Recovery Kit v6.4.0-2 でサポートされていません。このバージョンの HDLM を使用する必要がある場合は、HDLM Recovery Kit v7.2.0-1 以降と LifeKeeper Core v7.3 以降を使用できます。 |

| | |
|---|---|
| | **注記:** LVM と HDLM をともに使用する場合、HDLM がサポートするバージョンが必要です。また、フィルタ設定を /etc/lvm/lvm.conf に追加して、システムが /dev/sddlm* に対応する /dev/sd* を検出しないようにする必要があります。詳細については、HDLM のマニュアルの「LVM Configuration」を参照してください。 |
| Linux ディストリビューションの要件 | Linux ディストリビューションの要件<br><br>HDLM は以下のディストリビューションでサポートされています。<br><br>RHEL 4 (AS/ES) (x86 or x86_64) Update 1、2、3、4、Update 4 セキュリティフィックス (*2)、4.5、4.5 セキュリティフィックス(*4)、4.6、4.6 セキュリティフィックス(*8)、4.7、4.7セキュリティフィックス(*9)、4.8、4.8 セキュリティフィックス(*12) ×86/x86_64(*1)，国内、海外共)<br><br>RHEL 5、5.1、5.1 セキュリティフィックス(*5)、5.2、5.2 セキュリティフィックス(*6)、5.3、5.3 セキュリティフィックス(*10)、5.4、5.4 セキュリティフィックス(*11)、5.5、5.5 セキュリティフィックス (*13)、5.6、5.6　セキュリティフィックス (*14)、5.7 (x86/x86_64(*1)国内、海外共)、5.8,5.9,5.10,5.11 (x86/x86_64(*1)国内のみ)<br><br>RHEL 6、6.1、6.2、6.3、6.4、6.5、6.6、6.7、6.8、6.9、6.10 (x86/x86_64(*1)国内、海外共)(*15)<br><br>RHEL 7、7.1、7.2、7.3、7.4、7.5、7.6、7.7、7.8、7.9 (x86_64(*1)国内、海外共)<br><br>RHEL 8.1、8.2、8.3 (x86_64(*1)国内、海外共)<br><br>(*1) AMD Opteron(シングルコア、デュアルコア) あるいは Intel EM64T アーキテクチャ CPU + x86_64 カーネル<br><br>(*2) 次のカーネルがサポートされています。<br>x86：2.6.9-42.0.3.EL，2.6.9-42.0.3.ELsmp，2.6.9-42.0.3.ELhugemem<br>x86_64：2.6.9-42.0.3.EL，2.6.9-42.0.3.ELsmp，2.6.9-42.0.3.Ellargesmp<br><br>(*3) 日立 では、RHEL4 U2 の環境をサポートしていません。<br><br>(*4) 次のカーネルがサポートされています。<br>x86：2.6.9-55.0.12.EL，2.6.9-55.0.12.ELsmp，2.6.9-55.0.12.Elhugememx86_64：2.6.9-55.0.12.EL，2.6.9-55.0.12.ELsmp，2.6.9-55.0.12.Ellargesmp<br><br>(*5) 次のカーネルがサポートされています。<br>x86：2.6.18-53.1.13.el5，2.6.18-53.1.13.el5PAE，2.6.18-53.1.21.el5，2.6.18-53.1.21.el5PAE<br>x86_64：2.6.18-53.1.13.el5，2.6.18-53.1.21.el5 |

| | |
|---|---|
| | (*6) 次のカーネルがサポートされています。<br>x86：2.6.18-92.1.6.el5，2.6.18-92.1.6.el5PAE，2.6.18-92.1.13.el5，2.6.18- 92.1.13.el5PAE，2.6.18-92.1.22.el5，2.6.18-92.1.22.el5PAE<br>x86_64：2.6.18-92.1.6.el5，2.6.18-92.1.13.el5，2.6.18-92.1.22.el5<br><br>(*7) 次のカーネルがサポートされています。<br>x86：2.6.9-34.0.2.EL，2.6.9-34.0.2.ELsmp，2.6.9-34.0.2.ELhugemem<br>x86_64：2.6.9-34.0.2.EL，2.6.9-34.0.2.ELsmp，2.6.9-34.0.2.Ellargesmp<br><br>(*8) 次のカーネルがサポートされています。<br>x86：2.6.9-67.0.7.EL，2.6.9-67.0.7.ELsmp，2.6.9-67.0.7.ELhugemem，2.6.9- 67.0.22.EL，2.6.9-67.0.22.ELsmp，2.6.9-67.0.22.ELhugemem<br>x86_64：2.6.9-67.0.7.EL，2.6.9-67.0.7.ELsmp，2.6.9-67.0.7.ELlargesmp<br>2.6.9-67.0.22.EL，2.6.9-67.0.22.ELsmp，2.6.9-67.0.22.Ellargesmp<br><br>(*9) 次のカーネルがサポートされています。<br>x86：2.6.9-78.0.1.EL，2.6.9-78.0.1.ELsmp，2.6.9-78.0.1.ELhugemem，2.6.9-78.0.5.EL，2.6.9-78.0.5.ELsmp，2.6.9-78.0.5.ELhugemem，2.6.9-78.0.8.EL，2.6.9-78.0.8.ELsmp，2.6.9-78.0.8.ELhugemem, 2.6.9-78.0.17.EL，2.6.9-78.0.17.ELsmp，2.6.9-78.0.17.ELhugemem，2.6.9-78.0.22.EL，2.6.9-78.0.22.ELsmp，2.6.9-78.0.22.Elhugemem<br>x86_64：2.6.9-78.0.1.EL，2.6.9-78.0.1.ELsmp，2.6.9-78.0.1.ELlargesmp, 2.6.9-78.0.5.EL，2.6.9-78.0.5.ELsmp，2.6.9-78.0.5.ELlargesmp，2.6.9-78.0.8.EL，2.6.9-78.0.8.ELsmp，2.6.9-78.0.8.ELlargesmp，2.6.9-78.0.17.EL，2.6.9-78.0.17.ELsmp，2.6.9-78.0.17.ELlargesmp，2.6.9-78.0.22.EL，2.6.9-78.0.22.ELsmp，2.6.9-78.0.22.ELlargesmp<br><br>(*10) 次のカーネルがサポートされています。<br>x86：2.6.18-128.1.10.el5，2.6.18-128.1.10.el5PAE，2.6.18-128.1.14.el5, 2.6.18-128.1.14.el5PAE，2.6.18-128.7.1.el5，2.6.18-128.7.1.el5PAE<br>x86_64：2.6.18-128.1.10.el5，2.6.18-128.1.14.el5<br><br>(*11) 次のカーネルがサポートされています。<br>x86：2.6.18-164.9.1.el5，2.6.18-164.9.1.el5PAE，2.6.18-164.11.1.el5，2.6.18-164.11.1.el5PAE<br>x86_64：2.6.18-164.9.1.el5，2.6.18-164.11.1.el5<br><br>(*12) 次のカーネルがサポートされています。<br>x86：2.6.9-89.0.20.EL，2.6.9-89.0.20.ELsmp，2.6.9-89.0.20.Elhugemem<br>x86_64：2.6.9-89.0.20.EL，2.6.9-89.0.20.ELsmp，2.6.9-89.0.20.Ellargesmp<br><br>(*13) 次のカーネルがサポートされています。<br>x86：2.6.18-194.11.1.el5, 2.6.18-194.11.1.el5PAE, 2.6.18-194.11.3.el5, 2.6.18-194.11.3.el5PAE, 2.6.18-194.17.1.el5, 2.6.18-194.17.1.el5PAE, 2.6.18-194.32.1.el5, 2.6.18-194.32.1.el5PAE<br>x86_64：2.6.18-194.11.1.el5, 2.6.18-194.11.3.el5, 2.6.18-194.17.1.el5, 2.6.18-194.3 |

| | |
|---|---|
| | 2.1.el5<br><br>(*14) 次のカーネルがサポートされています。<br>x86：2.6.18-238.1.1.el5,2.6.18-238.1.1.el5PAE,2.6.18-238.9.1.el5,2.6.18-238.9.1.el5PAE,2.6.18-238.19.1.el5,2.6.18-238.19.1.el5PAE<br>x86_64：2.6.18-238.1.1.el5,2.6.18-238.9.1.el5,2.6.18-238.19.1.el5<br><br>(*15) 次のカーネルがサポートされています。<br>x86：2.6.32-71.el6.i686, 2.6.32-131.0.15.el6.i686, 2.6.32-220.el6.i686，2.6.32-279.el6.i686<br>x86_64：2.6.32-71.el6.x86_64, 2.6.32-131.0.15.el6.x86_64, 2.6.32-220.el6.x86_64，2.6.32-279.el6.x86_64<br><br>(*16) 次のカーネルがサポートされています。<br>x86：2.6.18-274.12.1.el5，2.6.18-274.12.1.el5PAE，2.6.18-274.18.1.el5，2.6.18-274.18.1.el5PAE<br>x86_64：2.6.18-274.12.1.el5，2.6.18-274.18.1.el5<br><br>(*17) 次のカーネルがサポートされています。<br>x86：2.6.18-308.8.2.el5，2.6.18-308.8.2.el5PAE<br>x86_64：2.6.18-308.8.2.el5<br><br>(*18) 次のカーネルがサポートされています。<br>x86：2.6.32-220.4.2.el6.i686, 2.6.32-220.17.1.el6.i686, 2.6.32-220.23.1.el6.i686, 2.6.32-220.31.1.el6.i686, 2.6.32-220.45.1.el6.i686<br>x86_64：2.6.32-220.4.2.el6.x86_64, 2.6.32-220.17.1.el6.x86_64, 2.6.32-220.23.1.el6.x86_64, 2.6.32-220.31.1.el6.x86_64, 2.6.32-220.45.1.el6.x86_64, 2.6.32-220.48.1.el6.x86_64，2.6.32-220.64.1.el6.x86_64，2.6.32-220.65.1.el6.x86_64, 2.6.32-220.72.2.el6.x86_64，2.6.32-220.73.1.el6.x86_64，2.6.32-220.75.1.el6.x86_64, 2.6.32-220.77.1.el6.x86_64<br><br>(*19) 次のカーネルがサポートされています。<br>x86：2.6.32-279.19.1.el6.i686<br>x86_64：2.6.32-279.19.1.el6.x86_64<br><br>(*20) 次のカーネルがサポートされています。<br>x86：2.6.32-358.6.2.el6.i686, 2.6.32-358.11.1.el6.i686, 2.6.32-358.14.1.el6.i686, 2.6.32-358.23.2.el6.i686<br>x86_64：2.6.32-358.6.2.el6.x86_64, 2.6.32-358.11.1.el6. x86_64, 2.6.32-358.14.1.el6. x86_64, 2.6.32-358.23.2.el6. x86_64, 2.6.32-358.28.1.el6. x86_64, 2.6.32-358.87.1.el6. x86_64<br><br>(*21) 次のカーネルがサポートされています。<br>x86：2.6.32-431.1.2.el6.i686, 2.6.32-431.3.1.el6.i686, 2.6.32-431.5.1.el6.i686, 2.6.32-431.17.1.el6.i686, 2.6.32-431.20.3.el6.i686, 2.6.32-431.23.3.el6.i686, 2.6.32-431.29.2.el |

| | |
|---|---|
| | 6.i686, 2.6.32-431.72.1.el6.i686，2.6.32-431.77.1.el6.i686<br>x86_64：2.6.32-431.1.2.el6.x86_64, 2.6.32-431.3.1.el6. x86_64, 2.6.32-431.5.1.el6. x86_64, 2.6.32-431.17.1.el6. x86_64, 2.6.32-431.20.3.el6. x86_64, 2.6.32-431.23.3.el6. x86_64, 2.6.32-431.29.2.el6. x86_64, 2.6.32-431.72.1.el6.x86_64, 2.6.32-431.77.1.el6.x86_64, 2.6.32-431.87.1.el6.x86_64<br><br>(*22) 次のカーネルがサポートされています。<br>x86：2.6.32-504.3.3.el6.i686，2.6.32-504.12.2.el6.i686，2.6.32-504.30.3.el6.i686<br>x86_64：2.6.32-504.3.3.el6.x86_64，2.6.32-504.12.2.el6.x86_64，2.6.32-504.16.2.el6.x86_64，2.6.32-504.30.3.el6.x86_64，2.6.32-504.40.1.el6.x86_64，2.6.32-504.43.1.el6.x86_64, 2.6.32-504.66.1.el6.x86_64<br><br>(*23) 次のカーネルがサポートされています。<br>x86：2.6.18-348.1.1.el5, 2.6.18-348.1.1.el5PAE, 2.6.18-348.6.1.el5, 2.6.18-348.6.1.el5PAE, 2.6.18-348.18.1.el5, 2.6.18-348.18.1.el5PAE<br>x86_64：2.6.18-348.1.1.el5, 2.6.18-348.6.1.el5, 2.6.18-348.18.1.el5<br><br>(*24) 下記のカーネルを示します。<br>x86_64:3.10.0-123.13.2.el7.x86_64, 3.10.0-123.20.1.el7.x86_64<br><br>(*25) 下記のカーネルを示します。<br>x86_64:3.10.0-229.4.2.el7.x86_64,3.10.0-229.20.1.el7.x86_64, 3.10.0-229.34.1.el7.x86_64<br><br>(*26) 下記のカーネルを示します。<br>x86_64:3.10.0-327.4.4.el7.x86_64, 3.10.0-327.4.5.el7.x86_64, 3.10.0-327.10.1.el7.x86_64, 3.10.0-327.18.2.el7.x86_64,<br>3.10.0-327.22.2.el7.x86_64, 3.10.0-327.36.1.el7.x86_64, 3.10.0-327.36.3.el7.x86_64,<br>3.10.0-327.44.2.el7.x86_64, 3.10.0-327.46.1.el7.x86_64, 3.10.0-327.49.2.el7.x86_64，3.10.0-327.55.2.el7.x86_64，3.10.0-327.55.3.el7.x86_64，3.10.0-327.58.1.el7.x86_64, 3.10.0-327.62.1.el7.x86_64，3.10.0-327.62.4.el7.x86_64, 3.10.0-327.64.1.el7.x86_64, 3.10.0-327.93.1.el7.x86_64, 3.10. 0 327. 9 6 .1 .el 7 .x86_64<br><br>(*27) 下記のカーネルを示します。<br>x86: 2.6.32-573.8.1.el6.i686, 2.6.32-573.12.1.el6.i686, 2.6.32-573.18.1.el6.i686, 2.6.32-573.53.1.el6.i686<br>x86_64: 2.6.32-573.8.1.el6.x86_64，2.6.32-573.12.1.el6.x86_64, 2.6.32-573.18.1.el6.x86_64, 2.6.32-573.53.1.el6.x86_64<br><br>(*28) 下記のカーネルを示します。<br>x86:2.6.32-642.1.1.el6.i686, 2.6.32-642.6.2.el6.i686, 2.6.32-642.13.1.el6.i686<br>x86_64:2.6.32-642.1.1.el6.x86_64, 2.6.32-642.6.1.el6.x86_64，2.6.32-642.6.2.el6.x86_64, 2.6.32-642.13.1.el6.x86_64, 2.6.32-642.15.1.el6.x86_64<br><br>(*29) 下記のカーネルを示します。<br>x86：2.6.18-416.el5，2.6.18-416.el5PAE, 2.6.18-419.el5，2.6.18-419.el5PAE, 2.6.1 |

| | |
|---|---|
| | 8-426.el5，2.6.18-426.el5PAE<br>x86_64：2.6.18-416.el5, 2.6.18-419.el5, 2.6.18-426.el5<br><br>(*30)下記のカーネルを示します。<br>x86_64：3.10.0-514.6.1.el7.x86_64，3.10.0-514.10.2.el7.x86_64，3.10.0-514.16.1.el7.x86_64，3.10.0-514.21.1.el7.x86_64，3.10.0-514.26.2.el7.x86_64，3.10.0-514.36.5.el7.x86_64，3.10.0-514.44.1.el7.x86_64, 3.10.0-514.51.1.el7.x86_64, 3.10.0-514.55.4.el7.x86_64<br><br>(*31)下記のカーネルを示します。<br>x86：2.6.32-696.3.2.el6.i686, 2.6.32-696.6.3.el6.i686, 2.6.32-696.10.3.el6.i686, 2.6.32-696.18.7.el6.i686，2.6.32-696.20.1.el6.i686, 2.6.32-696.23.1.el6.i686<br>x86_64：2.6.32-696.3.2.el6.x86_64, 2.6.32-696.6.3.el6.x86_64, 2.6.32-696.10.3.el6.x86_64, 2.6.32-696.18.7.el6.x86_64, 2.6.32-696.20.1.el6.x86_64, 2.6.32-696.23.1.el6.x86_64<br><br>(*32)下記のカーネルを示します。<br>x86_64：3.10.0-693.1.1.el7.x86_64, 3.10.0-693.5.2.el7.x86_64, 3.10.0-693.11.1.el7.x86_64，3.10.0-693.11.6.el7.x86_64, 3.10.0-693.21.1.el7.x86_64, 3.10.0-693.43.1.el7.x86_64, 3.10.0-693.46.1.el7.x86_64, 3.10.0-693.62.1.el7.x86_64<br><br>(*33)下記のカーネルを示します。<br>x86_64：3.10.0-862.3.2.el7.x86_64, 3.10.0-862.14.4.el7.x86_64<br><br>(*34)下記のカーネルを示します。<br>x86：2.6.32-754.3.5.el6.i686，2.6.32-754.15.3.el6.i686，2.6.32-754.24.3.el6.i686<br>x86_64：2.6.32-754.3.5.el6.x86_64，2.6.32-754.15.3.el6.x86_64，<br>2.6.32-754.24.3.el6.x86_64<br><br>(*35)下記のカーネルを示します。<br>x86_64：3.10.0-957.10.1.el7.x86_64，3.10.0-957.12.2.el7.x86_64，<br>3.10.0-957.21.3.el7.x86_64，3.10.0-957.27.2.el7.x86_64<br><br>(*36)下記のカーネルを示します。<br>x86_64：3.10.0-1062.1.1.el7.x86_64, 3.10.0 1062 9 .1 .el 7.x86_64, 3.10.0-1062.18.1.el7.x86_64, 3.10.0 1062 56 .1 .el 7.x86_64，3.10.0-1062.60.1.el7.x86_64<br><br>(*37)下記のカーネルを示します。<br>x86_64：4.18.0-147.5.1.el8_1.x86_64，4.18.0-147.8.1.el8_1.x86_64<br><br>(*38)下記のカーネルを示します。<br>x86_64：4.18.0-193.13.2.el8_2.x86_64，4.18.0-193.28.1.el8_2.x86_64<br><br>(*39) iSCSI 環境をサポートです。<br><br>(*40)下記のカーネルを示します。 |

| | |
|---|---|
| | x86_64：4.18.0-240.22.1.el8_3.x86_6<br><br>(*41) 下記のカーネル を示します。<br>x86_64 3.10. 0 1 160 .1 5.2 .el 7.x86_6 4<br><br>(*42)下記のカーネルを示します。<br>x86_64：4.18.0-305.3.1.el8_4.x86_64，4.18.0-305.25.1.el8_4.x86_64 |
| インストール要件 | HDLM Recovery Kit をインストールする前に HDLM ソフトウェアをインストールする必要があります。また、SCSI デバイスから HDLM デバイスに環境を転送したい場合は、HDLM 環境を設定した後、インストールセットアップスクリプトを実行する必要があります。そのようにしないと、sg3_utils がインストールされません。 |
| HDLM パスの追加または修復 | LifeKeeper は、HDLM リソースを起動する場合、パーシステントリザベーションを確立してその時点でアクティブなパスに登録します。最初のリザベーションの後に新しいパスが追加されるか、障害が起きたパスが修復されて HDLM がそのパスを自動的に再度アクティブにした場合、そのパスは、LifeKeeper が HDLM リソースに対する次の quickCheck を実行するまでリザベーションの一部として登録されません。その時点までに HDLM がそのパスに対する書き込みを許可した場合、リザベーションコンフリクトが発生し、システムのメッセージファイルに競合が記録されます。HDLM ドライバは、登録されたパスでそれらの I/O を再試行するため、アプリケーションにとっては検出可能な障害になりません。quickCheck によるパスの登録が完了すると、その後の書き込みは成功します。quickCheck がリザベーションコンフリクトを検出すると、ステータスが「Offline(E)」に変更されます。ステータスが「Offline(E)」の場合、ユーザはオンラインの HDLM コマンドを使用して手動でステータスを「Online」に変更する必要があります。 |
| RHEL 7.xの追加設定 | HDLM RecoveryKitをRed Hat Enterprise Linux 7.0以降で使用する場合、 /etc/default/LifeKeeper に HDLM_DLMMGR=.dlmmgr_exe を追記する必要があります。 |

| | **OS のバージョン / アーキテクチャ** | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | **RHEL4** | | | | | | | | | | |
| | U1-U4 | U3 セキュリティフィックス (*7) | U4 セキュリティフィックス (*2) | 4.5 | 4.5 セキュリティフィックス (*4) | 4.6 | 4.6 セキュリティフィックス | 4.7 | 4.7 セキュリティフィックス (*9) | 4.8 | 4.8 セキュリティフィックス (*12) |

| | | | | | | | | (*8) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | x86/x86_64 | | | | | | | | | | |
| HDLM | 05-80<br>05-81<br>05-90 | X | | | | | | | | | | |
| | 05-91<br>05-92 | X | | X | | | | | | | | |
| | 05-93 | X(*3) | | X | X | | | | | | | |
| | 05-94 | X(*3) | | X | X | X | X | X | | | | |
| | 6.0.0 | X(*3) | | X | X | X | X | X | X | X | X | X |
| | 6.0.1 | X(*3) | | X | X | X | X | X | X | X | X | X |
| | 6.1.0 | X(*3) | | X | X | X | X | X | X | X | X | X |
| | 6.1.1 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.1.2 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.2.0 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.2.1 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.3.0 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.4.0 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.4.1 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.5.0 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.5.1 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.5.2 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.6.0 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 6.6.2 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 7.2.0 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 7.2.1 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| | 7.3.0 以降 | X(*3) | X | X | X | X | X | X | X | X | X | X |
| LifeKeeper | v6.0<br>v6.0(v6.0.1-2 以降) | X | X | X | | | | | | | | |
| | v6.1<br>(v6.1.0-5 以降) | X | X | X | | | | | | | | |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | v6.2<br>(v6.2.0-5 以降) | X | X | X | X | X | X | X | | | | |
| | v6.2<br>(v6.2.2-1以降) | X | X | X | X | X | X | X | | | | |
| | v6.3<br>(v6.3.2-1以降) | X | X | X | X | X | X | X | | | | |
| | v6.4<br>(v6.4.0-10 以降) | X | X | X | X | X | X | X | X | X | | |
| | v7.0<br>(v7.0.0-5 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | V 7.1<br>(v7.1.0-8 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | V7.2<br>(v7.2.0-10 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | V 7.3<br>(v7.3.0-21 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | V 7.4<br>(v7.4.0-63 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | V 7.5<br>(v7.5.0-3640 以降) | RHEL4 は、LifeKeeper の v7.5 以降ではサポートされません。 | | | | | | | | | | |
| HDLM ARK | 6.0.1-2 | X | X | X | X | X | X | X | | | | |
| | 6.1.0-4 | X | X | X | X | X | X | X | | | | |
| | 6.2.2-3 | X | X | X | X | X | X | X | | | | |
| | 6.2.3-1 | X | X | X | X | X | X | X | X | X | X | X |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 6.4.0-2 | X | X | X | X | X | X | X | X | X | X | X |
| | 7.0.0-1 | X | X | X | X | X | X | X | X | X | X | X |
| | 7.2.0-1 | X | X | X | X | X | X | X | X | X | X | X |
| | X = サポートあり、空白 = サポートなし | | | | | | | | | | | |

| | | RHEL5 | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 未更新 | 5.1 | 5.1 セキュリティフィックス (*5) | 5.2 | 5.2 セキュリティフィックス (*6) | 5.3X | 5.3 セキュリティフィックス (*10) | 5.4 | 5.4 セキュリティフィックス (*11) | 5.5 | 5.5 セキュリティフィックス (*13) | 5.6 | 5.6 セキュリティフィックス (*14) | 5.7 セキュリティフィックス (*16) | 5.8 セキュリティフィックス (*17) | 5.9 セキュリティフィックス (*23) | 5.10 | 5.1 セキュリティフィックス (*29 |
| | | x86/x86_64 | | | | | | | | | | | | | | | | | |
| HDLM | 05-80<br>05-81<br>05-90 | | | | | | | | | | | | | | | | | | |
| | 05-91<br>05-92 | | | | | | | | | | | | | | | | | | |
| | 05-93 | X | | | | | | | | | | | | | | | | | |
| | 05-94 | X | X | | | | | | | | | | | | | | | | |
| | 6.0.0 | X | X | X | X | X | | | | | | | | | | | | | |
| | 6.0.1 | X | X | X | X | X | | | | | | | | | | | | | |
| | 6.1.0 | X | X | X | X | X | | | | | | | | | | | | | |
| | 6.1.1 | X | X | X | X | X | | | | | | | | | | | | | |
| | 6.1.2 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 6.2.0 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 6.2.1 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 6.3.0 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 6.4.0 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 6.4.1 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

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

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 8.6.4 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.6.5 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.0 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.1 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.2 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.3 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| LifeKeeper | v6.0 (v6.0.1-2 以降) | | | | | | | | | | | | | | | | | | |
| | v6.1 (v6.1.0-5 以降) | X | X | | | | | | | | | | | | | | | | |
| | v6.2 (v6.2.0-5 以降) | X | X | | | | | | | | | | | | | | | | |
| | v6.2 (v6.2.2-1 以降) | X | X | X | | | | | | | | | | | | | | | |
| | v6.3 (v6.3.2-1 以降) | X | X | X | X | X | | | | | | | | | | | | | |
| | v6.4 (v6.4.0-10 以降) | X | X | X | X | X | X | X | | | | | | | | | | | |
| | v7.0 (v7.0.0-5 以降) | X | X | X | X | X | X | X | X | X | | | | | | | | | |
| | v7.1 (v7.1.0-8 以降) | X | X | X | X | X | X | X | X | X | X | X | | | | | | | |
| | v7.2 (v7.2.0-10 以降) | X | X | X | X | X | X | X | X | X | X | X | | | | | | | |
| | v7.3 (v7.3.0-21 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | | | | | |
| | v7.4 (v7.4.0-63 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | | | | |

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 以降) | | | | | | | | | | | | | | | | | | |
| | v7.5 (v7.5.0-3640 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.0 (v8.0.0-510 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.1 (v8.1.1-5620 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.2 (v8.2.0-6213 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.2.1 (v8.2.1-6353 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.3.0 (v8.3.0-6389 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.3.1 (v8.3.1-6397 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.3.2 (v8.3.2-6405 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.4.0 (v8.4.0-6427 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v8.4.1 (v8.4.1-6449 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.0.0 (v9.0.0-6488 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.0.1 (v9.0.1-6492 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.0.2 (v9.0.2-6513 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | v9.1.0 (v9.1.0-6538 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.1.1 (v9.1.1-6594 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.1.2 (v9.1. 2-6609 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.2.0 (v9.2. 0-6629 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.2.1 (v9.2. 1-6653 以降) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | v9.2.2 (v9.2. 2-6679 以降) (注 1) | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| HDLM ARK | 6.0.1-2 | | | | | | | | | | | | | | | | | | |
| | 6.1.0-4 | X | X | | | | | | | | | | | | | | | | |
| | 6.2.2-3 | X | X | X | | | | | | | | | | | | | | | |
| | 6.2.3-1 | X | X | X | X | X | | | | | | | | | | | | | |
| | 6.4.0-2 | X | X | X | X | X | X | X | | | | | | | | | | | |
| | 7.0.0-1 | X | X | X | X | X | X | X | X | X | X | X | | | | | | | |
| | 7.2.0-1 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.1.1-5620 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.2.0-6213 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.2.1-6353 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.3.0-6389 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.3.1-6397 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.3.2-6405 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 8.4.0-6427 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 8.4.1-6449 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.0.0-6488 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.0.1-6492 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.0.2-6513 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.1.0-6538 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.1.1-6594 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.1.2-6609 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.2.0-6629 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.2.1-6653 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| | 9.2.2-6679 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| X = サポートあり、空白 = サポートなし | | | | | | | | | | | | | | | | | | | |

(注 1):RHEL5 のサポートは LifeKeeper v9.2.2 までです。 v9.3 以降はサポートされません。

| | | **OS のバージョン / アーキテクチャ** | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **RHEL6** | | | | | | | | | | |
| | | 6 | 6.1 | 6.2 セキュリティフィックス (*18) | 6.3 セキュリティフィックス (*19) | 6.4 セキュリティフィックス (*20) | 6.5 セキュリティフィックス (*21) | 6.6 セキュリティフィックス (*22) | 6.7 セキュリティフィックス (*27) | 6.8 セキュリティフィックス (*28) | 6.9 セキュリティフィックス (*31) | 6.10 セキュリティフィックス (*34) |
| | | **x86/x86_64** | | | | | | | | | | |
| HDLM | 6.5.0 | | | | | | | | | | | |
| | 6.5.1 | | | | | | | | | | | |
| | 6.5.2 | X | | | | | | | | | | |
| | 6.6.0 | X | | | | | | | | | | |
| | 6.6.2 | X | | | | | | | | | | |
| | 6.6.2-01 | X | X | | | | | | | | | |
| | 7.2.0 | X | X | X | | | | | | | | |
| | 7.2.1 | X | X | X | | | | | | | | |
| | 7.3.0 | X | X | X | | | | | | | | |
| | 7.3.1 | X | X | X | | | | | | | | |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 7.4.0 | X | X | X | X | X | X | X | X | X | | |
| | 7.4.1 | X | X | X | X | X | X | X | X | X | | |
| | 7.5.0 | X | X | X | X | X | X | X | X | X | | |
| | 7.6.0 | X | X | X | X | X | X | X | X | X | | |
| | 7.6.1 | X | X | X | X | X | X | X | X | X | | |
| | 8.0.0 | X | X | X | X | X | X | X | X | X | X | |
| | 8.0.1 | X | X | X | X | X | X | X | X | X | X | |
| | 8.1.0 | X | X | X | X | X | X | X | X | X | X | |
| | 8.1.1 | X | X | X | X | X | X | X | X | X | X | |
| | 8.1.2 | X | X | X | X | X | X | X | X | X | X | |
| | 8.1.3 | X | X | X | X | X | X | X | X | X | X | |
| | 8.1.4 | X | X | X | X | X | X | X | X | X | X | |
| | 8.2.0 | X | X | X | X | X | X | X | X | X | X | |
| | 8.2.1 | X | X | X | X | X | X | X | X | X | X | |
| | 8.4.0 | X | X | X | X | X | X | X | X | X | X | |
| | 8.5.0 | X | X | X | X | X | X | X | X | X | X | |
| | 8.5.1 | X | X | X | X | X | X | X | X | X | X | |
| | 8.5.2 | X | X | X | X | X | X | X | X | X | X | |
| | 8.5.3 | X | X | X | X | X | X | X | X | X | X | |
| | 8.5.4 | X | X | X | X | X | X | X | X | X | X | |
| | 8.6.0 | X | X | X | X | X | X | X | X | X | X | |
| | 8.6.1 | X | X | X | X | X | X | X | X | X | X | |
| | 8.6.2 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.6.4 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.6.5 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.0 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.1 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.2 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.3 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.4 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.6 | X | X | X | X | X | X | X | X | X | X | X |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 8.7.7 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.7.8 | X | X | X | X | X | X | X | X | X | X | X |
| | 8.8.0 | X | X | X | X | X | X | X | X | X | X | X |
| LifeKeeper | v7.0 (v7.0.0-5 以降) | | | | | | | | | | | |
| | V 7.1 (v7.1.0-8 以降) | | | | | | | | | | | |
| | V7.2 (v7.2.0-10 以降) | | | | | | | | | | | |
| | V 7.3 (v7.3.0-21 以降) | X | | | | | | | | | | |
| | V 7.4 (v7.4.0-63 以降) | X | | | | | | | | | | |
| | V 7.5 (v7.5.0-3640 以降) | X | X | X | X | | | | | | | |
| | v8.0 (v8.0.0-510 以降) | X | X | X | X | | | | | | | |
| | v8.1 (v8.1.1-5620 以降) | X | X | X | X | | | | | | | |
| | v8.1.2 (v8.1.2-5795 以降) | X | X | X | X | X | | | | | | |
| | v8.2.0 (v8.2.0-6213 以降) | X | X | X | X | X | | | | | | |
| | v8.2.1 (v8.2.1-6353 以降) | X | X | X | X | X | X | | | | | |
| | v8.3.0 (v8.3.0-6389 以降) | X | X | X | X | X | X | | | | | |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | v8.3.1 (v8.3.1-6937 以降) | X | X | X | X | X | X | | | | | |
| | v8.3.2 (v8.3.2-6405 以降) | X | X | X | X | X | X | X | X | X | | |
| | v8.4.0 (v8.4.0-6427 以降) | X | X | X | X | X | X | X | X | X | | |
| | v8.4.1 (v8.4.1-6449 以降) | X | X | X | X | X | X | X | X | X | | |
| | v9.0.0 (v9.0.0-6488 以降) | X | X | X | X | X | X | X | X | X | | |
| | v9.0.1 (v9.0.1-6492 以降) | X | X | X | X | X | X | X | X | X | | |
| | v9.0.2 (v9.0.2-6513 以降) | X | X | X | X | X | X | X | X | X | | |
| | v9.1.0 (v9.1.0-6538 以降) | X | X | X | X | X | X | X | X | X | | |
| | v9.1.1 (v9.1.1-6594 以降) | X | X | X | X | X | X | X | X | X | | |
| | v9.1.2 (v9.1.2-6609 以降) | X | X | X | X | X | X | X | X | X | X | |
| | v9.2.0 (v9.2.0-6629 以降) | X | X | X | X | X | X | X | X | X | X | |
| | v9.2.1 (v9.2.1-6653 以降) | X | X | X | X | X | X | X | X | X | X | |
| | v9.2.2 (v9.2.2-6679 以降) | X | X | X | X | X | X | X | X | X | X | |
| | v9.3 | X | X | X | X | X | X | X | X | X | X | |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | (v9.3.0-6738 以降) | | | | | | | | | | | |
| | v9.3.1 (v9.3.1-6750 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | v9.3.2 (v9.3.2-6863 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | v9.4.0 (v9.4.0-6959 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | v9.4.1 (v9.4.1-6983 以降) | X | X | X | X | X | X | X | X | X | X | X |
| | v9.5.0 (v9.5.0-7075 以降) | X | X | X | X | X | X | X | X | X | X | X |
| HDLM ARK | 7.0.0-1 | | | | | | | | | | | |
| | 7.2.0-1 | X | X | X | X | | | | | | | |
| | 8.1.1-5620 | X | X | X | X | | | | | | | |
| | 8.1.2-5795 | X | X | X | X | X | | | | | | |
| | 8.2.0-6213 | X | X | X | X | X | | | | | | |
| | 8.2.1-6353 | X | X | X | X | X | X | X | X | X | | |
| | 8.3.0-6389 | X | X | X | X | X | X | X | X | X | | |
| | 8.3.1-6397 | X | X | X | X | X | X | X | X | X | | |
| | 8.3.2-6405 | X | X | X | X | X | X | X | X | X | | |
| | 8.4.0-6427 | X | X | X | X | X | X | X | X | X | | |
| | 8.4.1-6449 | X | X | X | X | X | X | X | X | X | | |
| | 9.0.0-6488 | X | X | X | X | X | X | X | X | X | | |
| | 9.0.1-6492 | X | X | X | X | X | X | X | X | X | | |
| | 9.0.2-6513 | X | X | X | X | X | X | X | X | X | | |
| | 9.1.0-6538 | X | X | X | X | X | X | X | X | X | | |
| | 9.1.1-6594 | X | X | X | X | X | X | X | X | X | | |
| | 9.1.2-6609 | X | X | X | X | X | X | X | X | X | X | |
| | 9.2.0-6629 | X | X | X | X | X | X | X | X | X | X | |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 9.2.1-6653 | X | X | X | X | X | X | X | X | X | X | |
| | 9.2.2-6679 | X | X | X | X | X | X | X | X | X | X | |
| | 9.3.0-6738 | X | X | X | X | X | X | X | X | X | X | |
| | 9.3.1-6750 | X | X | X | X | X | X | X | X | X | X | X |
| | 9.3.2-6863 | X | X | X | X | X | X | X | X | X | X | X |
| | 9.4.0-6959 | X | X | X | X | X | X | X | X | X | X | X |
| | 9.4.1-6983 | X | X | X | X | X | X | X | X | X | X | X |
| | 9.5.0-7075 | X | X | X | X | X | X | X | X | X | X | X |
| X = サポートあり、空白 = サポートなし | | | | | | | | | | | | |

| | | RHEL7 | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7.0 セキュリティフィックス (*24) | 7.1 セキュリティフィックス (*25) | 7.2 セキュリティフィックス (*26) | 7.3 セキュリティフィックス (*30) | 7.4 セキュリティフィックス (*32) | 7.5 セキュリティフィックス (*33) | 7.6 セキュリティフィックス (*35) | 7.7 セキュリティフィックス (*36) | 7.8 セキュリティフィックス | 7.9 セキュリティフィックス(*42) |
| | | x86/x86_64 | | | | | | | | | |
| HDLM | 8.0.1 | X | X | | | | | | | | |
| | 8.1.0 | X | X | | | | | | | | |
| | 8.1.1 | X | X | | | | | | | | |
| | 8.1.2 | X | X | | | | | | | | |
| | 8.1.3 | X | X | | | | | | | | |
| | 8.1.4 | X | X | | | | | | | | |
| | 8.2.0 | X | X | | | | | | | | |
| | 8.2.1 | X | X | | | | | | | | |
| | 8.4.0 | X | X | X | | | | | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 8.5.0 | X | X | X | | | | | | | |
| | 8.5.1 | X | X | X | X | X | | | | | |
| | 8.5.2 | X | X | X | X | X | | | | | |
| | 8.5.3 | X | X | X | X | X | | | | | |
| | 8.5.4 | X | X | X | X | X | | | | | |
| | 8.6.0 | X | X | X | X | X | | | | | |
| | 8.6.1 | X | X | X | X | X | X | | | | |
| | 8.6.2 | X | X | X | X | X | X | X（注3） | | | |
| | 8.6.4 | X | X | X | X | X | X | X | | | |
| | 8.6.5 | X | X | X | X | X | X | X | | | |
| | 8.7.0 | X | X | X | X | X | X | X | X | | |
| | 8.7.1 | X | X | X | X | X | X | X | X | | |
| | 8.7.2 | X | X | X | X | X | X | X | X | | |
| | 8.7.3 | X | X | X | X | X | X | X | X | X | X |
| | 8.7.4 | X | X | X | X | X | X | X | X | X | X |
| | 8.7.6 | X | X | X | X | X | X | X | X | X | X |
| | 8.7.7 | X | X | X | X | X | X | X | X | X | X |
| | 8.7.8 | X | X | X | X | X | X | X | X | X | X |
| | 8.8.0 | X | X | X | X | X | X | X | X | X | X |
| | 8.8.1 | X | X | X | X | X | X | X | X | X | X |
| LifeKeeper | v9.0.0 (v9.0.0-6488X 以降) | X | | | | | | | | | |
| | v9.0.1 (v9.0.1-6492X 以降) | X | | | | | | | | | |
| | v9.0.2 (v9.0.2-6513X 以降) | X | X | | | | | | | | |
| | v9.1.0 (v9.1.0-6538X 以降) | X | X | | | | | | | | |
| | v9.1.1 | X | X | X | X | | | | | | |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | (v9.1.1-6594 以降) | | | | | | | | | |
| | v9.1.2 (v9.1.2-6609X 以降) | X | X | X | | | | | | |
| | v9.2.0 (v9.2.0-6629X 以降) | X | X | X | X | | | | | |
| | v9.2.1 (v9.2.1-6653X 以降) | X | X | X | X | | | | | |
| | v9.2.2 (v9.2.2-6679X 以降) | X | X | X | X | | | | | |
| | v9.3 (v9.3.0-6738X 以降) | X | X | X | X | X | | | | |
| | v9.3.1 (v9.3.1-6750X 以降) | X | X | X | X | X | | | | |
| | v9.3.2 (v9.3.2-6863X 以降) | X | X | X | X | X | X | | | |
| | v9.4.0 (v9.4.0-6959X 以降) | X | X | X | X | X | X | X | | |
| | v9.4.1 (v9.4.1-6983X 以降) | X | X | X | X | X | X | X | | |
| | v9.5.0 (v9.5.0-7075X 以降) | X | X | X | X | X | X | X | X | |
| | v9.5.1 (v9.5.1-7154X 以降) | X | X | X | X | X | X | X | X | X |
| | v9.5.2 (v9.5.2-7301X 以降) | X | X | X | X | X | X | X | X | X |
| HDLM ARK | 9.0.0-6488 X | X | | | | | | | | |
| | 9.0.1-6492 X | X | | | | | | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 9.0.2-6513 | X | X | X | | | | | | | |
| | 9.1.0-6538 | X | X | X | | | | | | | |
| | 9.1.1-6594 | X | X | X | X | | | | | | |
| | 9.1.2-6609 | X | X | X | X | | | | | | |
| | 9.2.0-6629 | X | X | X | X | X | | | | | |
| | 9.2.1-6653 | X | X | X | X | X | | | | | |
| | 9.2.2-6679 | X | X | X | X | X | | | | | |
| | 9.3.0-6738 | X | X | X | X | X | X | | | | |
| | 9.3.1-6750 | X | X | X | X | X | X | | | | |
| | 9.3.2-6863 | X | X | X | X | X | X | X | | | |
| | 9.4.0-6959 | X | X | X | X | X | X | X | X | | |
| | 9.4.1-6983 | X | X | X | X | X | X | X | X | | |
| | 9.5.0-7075 | X | X | X | X | X | X | X | X | X | |
| | 9.5.1-7154 | X | X | X | X | X | X | X | X | X | X |
| | 9.5.2-7301 | X | X | X | X | X | X | X | X | X | X |
| X = サポートあり、空白 = サポートなし | | | | | | | | | | | |

(注): Lifekeeper v9.0.x と RHEL7/7.1/7.2 の組み合わせは、Bug7205 のパッチの適用が必要です。

(注 2): RHEL7/7.1/7.2/7.3/7.4/7.5/7.6/7.7/7.8 では、Raw デバイスを使用した構成は未サポートです。

(注 3): HDLM 8.6.2-02 以降でのサポートです。

| | | **OS のバージョン / アーキテクチャ** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **RHEL8** | | | | | | | | | | | |
| | | 8.1 セキュリティフィックス(*37) | 8.2 セキュリティフィックス(*38)(*39) | 8.3 セキュリティフィックス(*40) | | | | | | | | | |
| | | **x86/x86_64** | | | | | | | | | | | |
| HDLM | 8.7.2 | X | | | | | | | | | | | |
| | 8.7.3 | X | | | | | | | | | | | |
| | 8.7.4 | X | X | X | | | | | | | | | |
| | 8.7.6 | X | X | X | | | | | | | | | |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 8.7.7 | X | X | X | | | | | | |
| | 8.7.8 | X | X | X | | | | | | |
| | 8.8.0 | X | X | X | | | | | | |
| LifeKeeper | v9.5.1 (9.5.1-7154 以降) | X | X | X | | | | | | |
| HDLM ARK | 9.5.1-7154 | X | X | X | | | | | | |
| X = サポートあり、空白 = サポートなし | | | | | | | | | | |

(注):RHEL8.1, RHEL8.2, RHEL8.3 では，Raw デバイスを使用した構成は未サポートです。

## Device Mapper のマルチパス I/O 設定

| | |
|---|---|
| Device Mapper Multipath デバイスを使用するアプリケーションとファイルシステムの保護 | Device Mapper Multipath デバイスを使用するアプリケーションやファイルシステムを LifeKeeper によって設定し保護するには、Device Mapper Multipath (DMMP) Recovery Kit をインストールする必要があります。<br><br>DMMP Kit のインストール後は、1 つ以上のマルチパスデバイスノードを使用するアプリケーション階層を作成するだけで、DMMP Kit が提供する新しいリソースタイプが自動的に組み込まれます。 |
| マルチパスデバイスノード | DMMP Kit を使用するには、すべてのファイルシステムおよび Raw デバイスをネイティブの /dev/sd* デバイスノードではなく、マルチパスデバイスノード上にマウントまたは設定する必要があります。ディスク全体を利用できるサポート対象のマルチパスデバイスノードは、/dev/dm-#、/dev/mapper/<uuid>、/dev/mapper/<user_friendly_name> および /dev/mpath/<uuid> です。ディスクのパーティションに対応するには、/dev/mapper ディレクトリに作成される各パーティション用のデバイスノードを使用します。 |
| SCSI-3 Persistent Reservations の使用 | Device Mapper Multipath Recovery Kit は、リザベーションタイプを「書き込み専用」とする SCSI-3 Persistent Reservations を使用します。この場合、クラスタのあるノードが予約したデバイスは、クラスタの他のノードから読み取り可能のままですが、他のノードからデバイスへの書き込みはできなくなります。このことは、それらの他のノード上で進行中の読み取り専用アクセスのためにファイルシステムをマウントできるという意味ではないことに注意してください。<br><br>LifeKeeper では、sg_persist ユーティリティを使用してパーシステントリザベーションを発行、監視します。必要であれば、LifeKeeper は sg_persist(8) ユーティリティをインストールします。<br><br>EMC Symmetrix (VMAX を含む) アレイをマルチパスソフトウェアおよび LifeKeeper と |

| | |
|---|---|
| | 組み合わせて使用する場合は、SCSI-3 Persistent Reservations を LUN 単位で有効にする必要があります。これは、DMMP と PowerPath の両方に当てはまります。 |
| ハードウェア要件 | Device Mapper Multipath Kit は、EMC CLARiiON CX300、HP EVA 8000、HP MSA1500、HP P2000、IBM SAN Volume Controller (SVC)、IBM DS8100、IBM DS6800、IBM ESS、DataCore SANsymphony、HDS 9980V を使用して SIOS Technology Corp. によりテスト済みです。Device Mapper Multipath のサポートについては、ストレージベンダにお問い合わせください。<br><br>CX300 およびVNXシリーズでリザベーションのサポートを有効にするには、リザベーションに従うようにハードウェアハンドラに通知する必要があります。このアレイ用に */etc/multipath.conf* 内に次のパラメータを設定してください。<br><br>`hardware_handler "3 emc 0 1"`<br><br>HP MSA1500 の場合、デフォルトのパスチェッカ (tur) とのリザベーションコンフリクトを返します。これによりスタンバイノードは、すべてのパスを障害であると判定します。この状態を回避するには、このアレイ用に */etc/multipath.conf* 内に次のパラメータを設定してください。<br><br>`path_checker readsector0`<br><br>HP 3PAR F400 は、デフォルトのパスチェッカとのリザベーション競合を返します。この競合を回避するには、このアレイ用に */etc/default/LifeKeeper* 内に次のパラメータを設定 (追加) してください。<br><br>`DMMP_REGISTRATION_TYPE=hba`<br><br>HDS 9980V の場合、以下の設定が必要です。<br><br>• Host mode: 00<br>• System option: 254 (有効にする必要がある。すべてのサーバに影響を与えるグローバルな HDS 設定)<br>• Device emulation: OPEN-V<br><br>HDS の DMMP 設定の詳細については、HDS ドキュメンテーション「Suse Linux Device Mapper Multipath for HDS Storage」または「Red Hat Linux Device Mapper Multipath for HDS Storage」の v1.15 以降を参照してください。このドキュメンテーションでは、互換性のある multipath.conf ファイルも提供しています。<br><br>ファームウェアバージョン 6 以降を使用する EVA ストレージでは、DMMP Recovery Kit v6.1.2-3 以降が必要です。これ以前のバージョンの DMMP Recovery Kit は、バージョン 6 より前のファームウェアを使用する EVA ストレージでサポートされています。 |

| | |
|---|---|
| マルチパスソフトウェアの要件 | SUSE の場合、multipath-tools-0.4.5-0.14 以降が必要です。<br><br>Red Hat の場合、device-mapper-multipath-0.4.5-12.0.RHEL4 以降が必要です。<br><br>ベンダが提供する最新のマルチパスツールの組み合わせを使用することを推奨します。このマルチパス製品の機能と安定性は急速に向上しています。 |
| Linux ディストリビューションの要件 | IBM などの一部のストレージベンダは、現時点では SLES 11 を使用する DMMP を認定していません。<br><br>SIOS Technology Corp. は、DMMP、SLES 11、EMC CLARiiON および Symmetrix アレイの組み合わせで報告された問題を現在調査中です。 |
| 一時的なパス障害 | Device Mapper Multipath デバイスで I/O テストを実行中に、サーバのリブートなどの SAN 上の操作によって一時的なパスの障害が報告されることは珍しくありません。ほとんどの場合、結果として単に 1 つのパスだけが障害となり他のパスは I/O を送信するため、パフォーマンスへのわずかな影響以外に検出される障害はありません。ただし一部のケースでは、複数のパスが障害として報告され、機能するパスがまったくない状態になることがあります。この状態では、ファイルシステムやデータベースなどのアプリケーションからは I/O エラーが発生しているように見えます。このような障害を排除する上で、Device Mapper Multipath およびベンダのサポートはこれまでに大きく改善されました。ただし、まだ問題が発生することはあります。このような状況を回避するため、以下の措置を検討してください。<br><br>1. ディスクアレイベンダの手順に従ってマルチパス構成が正しく設定されていることを確認します。<br>2. 「failback」機能の設定を確認します。この機能は、パスの障害および修復後にパスを再度アクティブにするまでの時間を指定します。「immediate」に設定した場合、パスがオンラインに戻るとすぐに使用を再開することを意味します。整数に設定した場合、パスがオンラインに戻ってから使用を再開するまでの秒数を意味します。10 ~ 15 に設定すると、一般的に SAN 上のスラッシングを回避するのに十分な処理時間が得られます。<br>3. 「no_path_retry」機能の設定を確認します。この機能は、すべてのパスに障害が発生したときに Device Mapper Multipath がやるべきことを指定します。10 ~ 15 に設定することを推奨します。この機能によって、すべてのパスに障害が起きた一時的なイベントを「乗り切る」方法が提供され、復旧に必要な妥当な時間を稼ぐことができます。LifeKeeper の DMMP Kit はストレージへの I/O を監視しており、4 分以内に応答がなかった場合、LifeKeeper はスタンバイサーバにリソースをスイッチオーバします。注記: 簡単には削除できない I/O が発生するため、LifeKeeper では「no_path_retry」設定を「queue」に設定することは推奨されません。それらを削除するためのメカニズムは、新しいバージョンの DM に含まれており、デバイスの設定を次のように変更できます。<br><br>`/sbin/dmsetup message -u 'DMid' 0 fail_if_no_path` |

| | |
|---|---|
| | これによって no_path_retry の設定が一時的に「fail」に変更され、未処理の I/O がすべて失敗します。ただし、multipathd は、no_path_retry をいつでもデフォルトにリセットできます。失敗した I/O を消去するために設定が fail_if_no_path に変更されたときは、デバイスにアクセスする前に (手動または LifeKeeper によって) 設定をデフォルトにリセットする必要があります。<br><br>「no_path_retry」が「queue」に設定されていて障害が発生した場合、LifeKeeper はリソースをスタンバイサーバにスイッチオーバします。ただし、LifeKeeper は失敗した I/O を削除しません。この I/O を消去するための推奨の方法はリブートですが、上記の dmsetup コマンドを使用して管理者が消去することもできます。I/O を消去しておかないと、他方のサーバでリソースがサービス休止状態になってロックを解放した場合にこの「古い」I/O が発行される事態になってデータの破損が発生します。 |

# 5.4.2.10. LifeKeeper I-O フェンシングの概要

I/Oフェンシングは、障害ノードをデータから切り離すことにより、共有ストレージへの非協調的なアクセスを防止する機能です。複数のサーバが同じデータにアクセスできる環境では、データの破損を防ぐためにすべての書き込みを制御された方法で行うことが不可欠です。障害検知メカニズムが破綻した場合、この破綻によってノード障害に似た状況になるため問題が発生します。例えば、2 ノードクラスタで 2 つのノード間の接続に障害が発生した場合、各ノードは相手側に障害が発生したと「思い込む」ため、両方のノードがデータに対する制御を獲得しようと試みてデータの破損につながります。I/O フェンシングは、特定のノードからのデータアクセスをブロックすることによりこのデータ破損のリスクを排除します。

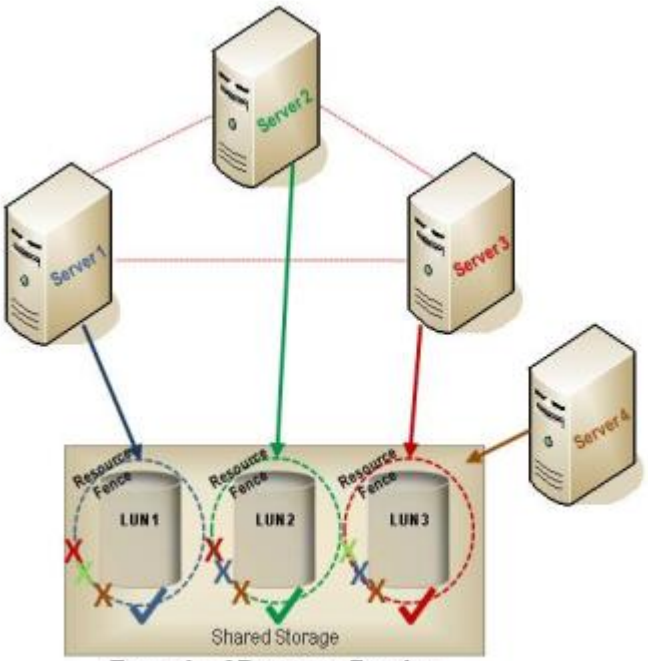
なお、DataKeeper 構成では、DataKeeper 構成での I/O フェンシング を参照して下さい。

# 5.4.2.10.1. SCSI リザベーション

## SCSI リザベーションを利用したストレージフェンシング

LifeKeeper for Linux は、リソースフェンシングとノードフェンシングの両方をサポートしますが、主要なフェンシングメカニズムは、SCSI リザベーションによるストレージフェンシングです。共有ストレージに対する最高レベルのデータ保護を提供するこのフェンシングを使用すると、非常に粒度の高い LUN レベルのロックによって最大限の柔軟性と最大限のセキュリティが可能になります。このアーキテクチャでベースとなる共有リソース (LUN) は、プライマリ quorum デバイスです。 quorum は、共有ストレージに対する排他的なアクセスと定義できます。つまり、この共有ストレージは 1 度に 1 台のサーバからしかアクセスできません。 quorum (排他的アクセス) を持つサーバは、「プライマリ」の役割を持ちます。 quorum の確立 (排他的アクセスをどのサーバに与えるか) は、「quorum デバイス」によって行われます。

上述の通り、リザベーションが有効の場合、quorum デバイスはその共有リソースです。共有リソースは、共有リソースに対するリザベーションを持つサーバを判断して quorum を確立します。これにより、ある 1 つのサーバがその LUN にアクセスできる限り、クラスタは実質的には単一のサーバで運用されることになります。



Example of Resource Fencing

SCSI リザベーションは、共有のユーザデータを保護し、LifeKeeper が指定するシステムのみデータを変更できるようにします。クラスタ内外の他のシステムがそのデータを変更することは許可されません。さらに SCSI リザベーションによって、クラスタ内の複数のサーバで障害が起きた場合に、LifeKeeper 保護下のアプリケーションは共有のユーザデータに安全にアクセスできます。サーバの多数派 quorum は必要ありません。唯一の要件は、共有データの所有権の帰属が確立していることです。

quorum/witness 機能を追加すると、 quorum のメンバーシップを確立することができます。このメンバーシップがない場合、スプリットブレインの状況で複数のサーバ (場合によっては全サーバ) がお互いを終了させることも考えられます。リザベーションが有効になっている構成に Watchdog を追加すると、部分的にサーバがハングしている状態からリカバリするメカニズムが提供されます。ハングしたサーバが

LifeKeeper に検出されないような場合に、Watchdog はリカバリを開始します。また、サーバがハングしてさらにリザベーションが奪われた場合に、Watchdog はそのサーバをリブートしてリカバリを開始することができます。

## I/O フェンシングのための代替方式

SCSI リザベーションを利用したリソースフェンシングに加えて、LifeKeeper for Linux はリザベーションの無効化もサポートします。リザベーションが有効か無効にかかわらず、以下の 2 つの点に注意すべきです。

- ストレージへのアクセスは LifeKeeper が制御する必要があります。
- ストレージへの意図しないアクセス (ファイルシステムのマウント、手動の fsck など) が発生しないように細心の注意を払う必要があります。

以上の 2 つのルールを順守してリザベーションを有効にすると、LifeKeeper はたいていのエラーを防止できます。リザベーションが (単独で) 無効になった状態は、保護がない状態です。したがって、この保護を実現するには、他の選択肢を検討する必要があります。以降のセクションでは、SCSI リザベーションなしでも LifeKeeper で非常に信頼性の高い構成を実現できる各種のフェンシングオプションと代替方式を説明します。

# 5.4.2.10.2. リザベーションの無効化

リザベーションを使用すると、共有ストレージに対する最高レベルのデータ保護が可能になりますが、場合によっては、リザベーションを使用できず LifeKeeper 内で無効にしなければならないことがあります。リザベーションを無効にすると、複数のシステムが意図的または意図せずにストレージにアクセスしようとする場合にストレージが調停役として動作することがなくなります。

そのため、システムハング、システムビジー、またはサーバが停止したように見えるあらゆる状況に対応できるように、クラスタメンバーシップによってストレージをフェンシングする別の方法を採用することを検討する必要があります。

リザベーションがなくても信頼性の高い構成を実現する鍵は、フェイルオーバが発生したとき、「他の」サーバの電源がオフになったこと、または電源が再投入されたことを「知る」ことです。この要件を満たすために利用可能なフェンシングオプションは 4 つあり、これらは SCSI リザベーションなしでも LifeKeeper で非常に信頼性の高い構成を実現できます。オプションは以下の通りです。

- STONITH (Shoot the Other Node in the Head) (高信頼性のインターコネクト、すなわちサーバと STONITH デバイスとの間のシリアル接続を使用) - STONITH は、サーバがクラスタの一部とみなされなくなったときに、そのサーバを物理的に停止させたり、電源を切断したりする技術です。LifeKeeper フェイルオーバのイベント時にサーバの電源を切断する機能をサポートしています。これにより共有データへの安全なアクセスを保証します。このオプションはリザベーションと同様の信頼性を提供できますが、利用できるのは物理的に同じ場所に配置された 2 つのノードに限定されます。

- Quorum/Witness - Quorum/Witness サーバは、特にクラスタサーバが異なる場所に配置されている場合に、クラスタ内でのメンバーシップを確認するために使用されます。このオプションはスプリットブレインに対応できるもののシステムハングに対応できないため、単独での使用は推奨されません。

- Watchdog - Watchdog は、サーバーの状態を監視します。問題が検出されると、問題のあるサーバは再起動または電源を切断されます。このオプションではサーバーのハングからのリカバリはできますが、スプリットブレインには対応できません。したがって、このオプションもまた単独での使用は推奨されません。

- CONFIRM_SO - このオプションでは自動フェイルオーバを無効にする必要があります。そのため、信頼性が非常に高い (管理者のスキルによって異なります) 一方で、可用性はあまり高くありません。

これらの代替フェンシング方式はどれも単独では十分とは言えませんが、組み合わせて使用することで非常に信頼性の高い構成を実現できます。

## 非共有ストレージ

非共有ストレージ環境で LifeKeeper を使用する計画の場合、共有ストレージに存在するデータ破損のリスクは問題にならないため、リザベーションは不要です。ただし、データの部分的または完全な再同期およびマージが必要な場合があります。信頼性と可用性を最適化するには、非共有ストレージでも上記のオプ

ションを検討する必要があります。

**注記:** 各オプションの信頼性と可用性の比較の詳細については、I/O フェンシング比較表 を参照してください。

完全なデータ保護を実現するオプションはないことを理解することは重要です。ただし、以下のように組み合わせるとリザベーションとほぼ同等レベルの保護を実現できます。

## リザベーションを使用しない I/O フェンシングの設定

ノードフェンシングをサポートするクラスタを構成するには以下の手順を実行します。

1. LifeKeeper を停止します。

2. LifeKeeper 内での SCSI リザベーションの使用を無効にします。無効にするには、クラスタのすべてのノードで LifeKeeper のデフォルトファイル /etc/default/LifeKeeper を編集します。Reservations 変数を追加または修正して「none」にします (`RESERVATIONS="none"`)。このオプションはリザベーションを利用できない場合のみ使用することに注意してください。

3. I/O フェンシングを提供する STONITH デバイスを用意して設定します。この設定では、STONITH デバイスが reboot コマンドではなく、poweroff コマンドをシステムに対して実行するようにします。LifeKeeper の通信が何らかの理由で中断したとき、手動操作によって同時に両ノード上のデバイス階層を In Service の状態にしないように注意してください。

4. 必要に応じて、quorum/witness サーバを用意して設定します。quorum/witness サーバを設定、使用する詳細な手順と情報については、Quorum/Witness Server Support Package トピックを参照してください。

   注記: サイトの障害時に最良の保護機能を提供するため、quorum/witness サーバはクラスタ内の別サーバから離れた場所にある必要があります。

5. 必要に応じて、Watchdog を設定します。詳細については、 Watchdog トピックを参照してください。

# 5.4.2.10.2.1. I/O フェンシング表

| | スプリットブレイン | ハングしたサーバ |
|---|---|---|
| **Reservations オン** | | |
| **単体** | | |
| **Quorum/Witness** | | |
| **Watchdog** | | |
| **Watchdog & Quorum/ Witness** | | |
| **STONITH (シリアル)** | | |
| **Reservations オフ** | | |
| **なし** | | |
| **STONITH (シリアル)** | | |
| **CONFIRM_SO*** | | |
| **Quorum/Witness** | | |
| **Watchdog** | | |
| **非共有ストレージ** | | |
| **デフォルトの機能** | | |
| **Quorum/Witness** | | |
| **CONFIRM_SO*** | | |
| **Watchdog** | | |
| **STONITH (シリアル)** | | |

* *CONFIRM_SO は信頼性が高いが（管理者の知識による）、自動フェイルオーバがオフになっているという事実により可用性が低い。*

# 5.4.2.10.3. Quorum/Witness

## Quorum/Witness Server Support Package for LifeKeeper

### 機能の概要

LifeKeeper Core の既存のフェイルオーバープロセスに Quorum/Witness Server Support Package for LifeKeeper（steeleye-lkQWK、以下　Quorum/Witness　パッケージ）を組み合わせることにより、ネットワーク全体にわたる障害の恐れがある環境において、より高い信頼度でシステムフェイルオーバーを実行できます。つまり、「スプリットブレイン」の発生リスクを大幅に軽減しながら、ローカルサイトのフェイルオーバーと WAN 越しのノードへのフェイルオーバーを実行することができます。

ネットワーク分断を考慮した分散システムにおいて、クラスター全体での合意を得るために Quorum と呼ばれる概念があります。Quorum を持っているノードはクラスター全体での合意が得られるノードであり、リソースを起動することが許可されます。一方、Quorumを持っていないノードはクラスター全体での合意が得られないノードで、リソースを起動することは許可されません。これにより、スプリットブレインを抑止します。そして、この Quorum を持っているかどうかを確認することを Quorum チェックと呼び、Quorum を持っている場合を Quorum チェック成功、Quorum を持っていない場合を Quorum チェック失敗と表現します。

通信障害が発生した時、障害が発生したノードと別の１台以上のノード（もしくは他のデバイス）を使って障害ノードのステータスについて「セカンドオピニオン」を取得することができます。このセカンドオピニオンを得るために使用されるノードを Witness ノード（もしくは Witness デバイス）と呼びます。また、このセカンドオピニオンを得ることを Witness チェックと呼びます。フェイルオーバー先となることができるノードは、Witness ノード（Witness デバイス）が障害となったノードのステータスに関して同じ意見である場合のみ、リソース起動が許可されます。これにより、ノード間で発生する単純な通信障害から発生するフェイルオーバーを回避し、全体のアクセスや、パフォーマンス、サービス中のノードに影響を与えないようにします。実際の運用では、LifeKeeperが起動した時やコミュニケーションパスが復旧した時にも、Witness ノード（Witness デバイス）に問い合わせを行います。なお、Witness チェックを行うのは、Quorum を持っていることが前提です。

> ✻ SCSI リザベーションによる排他制御を利用できない DataKeeper や NAS などの共有ストレージを使用する場合、Quorum/Witness 機能を使用することでスプリットブレイン状態の発生を抑制してより安全な HA クラスターを構築することができます。

> ✻ Witnessノードのシステム要件は LifeKeeper のシステム要件 と同じです。Witness ノードはレプリケーションに参加せず、保護するリソースもないため、最低要件を満たしていれば動作します。

### パッケージのインストールと設定

Quorum/Witness 機能を利用するクラスターに参加するすべてのノード（Witness 専用ノードを含む）に Quorum/Witness パッケージをインストールする必要があります。Witness ノードに必要な唯一の設定

は、適切なコミュニケーションパスを作成する ことです。Quorum モードを *tcp_remote* で使用する場合、/etc/default/LifeKeeper 設定ファイルの QUORUM_HOSTS に設定したホストには LifeKeeper をインストールする必要はありません。

Quorum/Witness 機能をセットアップする一般的なプロセスには以下の手順が含まれます。

1. サーバーをセットアップし、他のサーバーとネットワーク通信ができることを確認します。
2. サーバーに LifeKeeper をインストールします。その際、setupコマンドで「Use Quorum / Witness functions」を有効にして、Quorum/Witness パッケージもインストールしてください。
3. Witness 専用ノードを含め、すべてのノード間でコミュニケーションパスを作成します。
4. Quorum/Witnessの設定を行います。

上記の手順が完了すると、そのクラスターでは Quorum/Witness 機能が動作するようになり、フェイルオーバーが許可される前に、Quorum チェックおよび Witness チェックが行われます。
詳細な設定オプションについては、下の「設定可能なコンポーネント」セクションを参照してください。

**注記:** Quorum/Witness パッケージをインストールしたノードであれば、Quorum/Witness 機能のクラスターに参加できます。Witness 専用ノードとは、保護対象のリソースを持たず、他のすべてのノードとコミュニケーションパスが設定されているノードのことを指しています。

## 設定可能なコンポーネント

Quorum/Witness パッケージでは、Quorum モードと Witness モードの 2 つのモードを設定できます。デフォルトでは、Quorum/Witness パッケージをインストールすると、Quorum モードと Witness モードの両方のモードが有効になります。

これらのモードの動作は、/etc/default/LifeKeeper 設定ファイルでカスタマイズすることが可能です。Quorum モードと Witness モードを個別に調整することもできます。パッケージがインストールされると設定ファイルにはデフォルト設定が書き込まれ、 *majority* がデフォルトの Quorum モードに、*remote_verify* がデフォルトの Witness モードになります。以下はその例です。

```
QUORUM_MODE=majority
WITNESS_MODE=remote_verify
```

## 使用可能な Quorum モード

Quorum チェックのモードとして、次の 4 種類のモードが用意されています。これらは、/etc/default/LifeKeeper 設定ファイルの QUORUM_MODE で設定できます。

| QUORUM_MODE | 説明 |
|---|---|
| *majority*<br>（デフォルト） | コミュニケーションパスを通じて疎通確認を行い、Quorum チェックを行います。クラスター内の過半数のノードと疎通ができることで、Quorum チェック成功と判断します。 本モードは 3 ノード以上のクラスターで動作可能です。 2 ノード構成の場合は、 |

| | |
|---|---|
| | Witness 専用のノードを追加する必要があります。 詳細は「majority モード 」を参照してください。 |
| *tcp_remote* | コミュニケーションパスから独立したホストに対して、指定されたポート上の TCP/IP サービスに接続確認を行い、Quorum チェックを行います。指定されたホストの過半数に接続できることで、Quorum チェック成功と判断します。 別途、接続確認用のホストが必要です。 詳細は「tcp_remote モード 」を参照してください。 |
| *storage* | 共有ストレージを Witness デバイスとして用いる Quorum モードです。クラスター内のすべてのノードからアクセスできる共有ストレージを用いた合意システムで、共有ストレージを介してノードの情報交換を行います。共有ストレージにアクセスできることで、Quorum チェック成功と判断します。 本モードは 2 ノードのクラスターから動作可能で、最大 4 ノードまでサポートされます。別途、共有ストレージが必要です。<br>詳細は「storage モード 」を参照してください。<br>なお、Quorum モードに *storage* を選んだ場合、後述の Witness モードも *storage* を選択しなければなりません。 |
| *none/off* | Quorum チェックが無効になっています。この設定では、常に Quorum チェック成功と判断されます。 |

## 使用可能な Witness モード

Witness チェックのモードとして、次の 3 種類のモードが用意されています。これらは、`/etc/default/LifeKeeper` 設定ファイルの `WITNESS_MODE` で設定できます。

| WITNESS_MODE | 説明 |
|---|---|
| *remote_verify*<br>（デフォルト） | クラスター内の他のすべてのノードに対して障害が疑われるノードのステータスに関する意見を求めます。 1 つのノードでも障害なしと判断した場合は、Witness チェックの結果は障害なしと判断します。すべてのノードが障害と判断した場合は、Witness チェックの結果は障害が発生していると判断します。 |
| *storage* | 共有ストレージを Witness デバイスとして用いる Witness モードです。共有ストレージに書き込まれた情報を定期的に確認し、更新されていれば障害なし、更新が止まれば障害が発生していると判断します。 詳細は「storage モード 」を参照してください。<br>なお、Witness モードに *storage* を選んだ場合、先述の Quorum モードも *storage* を選択しなければなりません。 |
| *none/off* | Witness チェックが無効になっています。この設定では、常に障害が発生していると判断されます。 |

> **注記:** リソースを持たない Witness 専用ノードは Witness チェックを実行する必要がありません。そのため、そのようなノードでは Witness モードを *none/off* に設定する必要があります。

# 使用可能な Quorum モードと Witness モードの組み合わせ

以下の表の組み合わせが可能です。

| | | QUORUM_MODE | | | |
|---|---|---|---|---|---|
| | | ***majority*** | ***tcp_remote*** | ***storage*** | ***none/off*** |
| **WITNESS_MODE** | ***remote_verify*** | 利用可能<br>３ノード以上 | 利用可能<br>３ノード以上 | 利用不可 | 利用可能<br>３ノード以上 |
| | ***storage*** | 利用不可 | 利用不可 | 利用可能<br>２ノード以上<br>４ノード以下 | 利用不可 |
| | ***none/off*** | 利用可能<br>３ノード以上 | 利用可能<br>２ノード以上 | 利用不可 | 利用可能 |

# Quorum を喪失したときに利用可能なアクション

Quorum/Witness パッケージでは、Quorum を喪失したときにシステムがどのように応答すべきかについて３種類のオプションを提供しています。“*fastboot*”、“*fastkill*” および “*osu*” です。これらのオプションは、`/etc/default/LifeKeeper` 設定ファイルの `QUORUM_LOSS_ACTION` 設定によって選択できます。３つのオプションはすべてそのシステムのリソースをサービス休止状態にしますが、それぞれ異なる動作をします。

| オプション | 説明 |
|---|---|
| *fastboot* | （コミュニケーションパスの障害により）Quorum の喪失が検出されると、システムは **直ちに** リブートします。これは過激な方法ですが、確実にシステムを外部のリソースから素早く切り離すことができます。ストレージレベルのレプリケーションなど、多くの場合にリソースをこのように即座にリリースすることが望まれます。 このオプションには、以下の 2 つの重要な注意点があります。<br>1. システムは、最初にシャットダウン手順を実行することなく **直ちに** ハードリブートを実行します。（ディスクの同期などの）シャットダウンタスクは一切実行されません。<br>2. システムは、ストレージとのネゴシエーションやリソースへのアクセスなどを含む通常の起動ルーチンを実行しながら復帰します。 |
| *fastkill*（デフォル | *fastboot* オプションに非常に似ていますが、システムは Quorum を喪失したときにハードリブートするのではなく、 **直ちに** 停止します。 *fastboot* オプションと同様に（ディスクの同期などの）シャットダウンタスクは一切実行されません。システムは手動で起動する必要があります。 |

| | |
|---|---|
| ト) | その後システムは、ストレージとのネゴシエーションやリソースへのアクセスなどを含む通常の起動ルーチンを実行しながら復帰します。 |
| *osu* | 最も穏健なオプションです。Quorum を喪失したシステムはそのまま稼働しますが、システム上のリソースはサービス休止状態にされます。一部のクラスター構成ではこの方法で十分ですが、他のクラスター構成では保護能力不足だったり応答が遅すぎる場合があります。 |

# 5.4.2.10.3.1. majority モード

コミュニケーションパスを通じて疎通確認を行い、Quorum チェックを行います。クラスター内の過半数のノードと疎通ができることで、Quorum チェック成功と判断します。
本モードは 3 ノード以上のクラスターで動作可能です。 2 ノード構成の場合は、Witness 専用のノードを追加する必要があります。

> **注記:** 多数決ベースのため、クラスターを構成するノード数は奇数にすることを推奨します。

**quorum を使用してクラスターを停止する:**

1. ターゲットを停止
2. ソースを停止
3. witness を停止

**quorum を使用してクラスターを起動する:**

1. witness を起動
2. ターゲットを起動
3. ソースを起動

## majority モードの設定

`/etc/default/LifeKeeper` 設定ファイルで `QUORUM_MODE` の設定を *majority* に設定してください。
本モードでは、他の設定はありません。

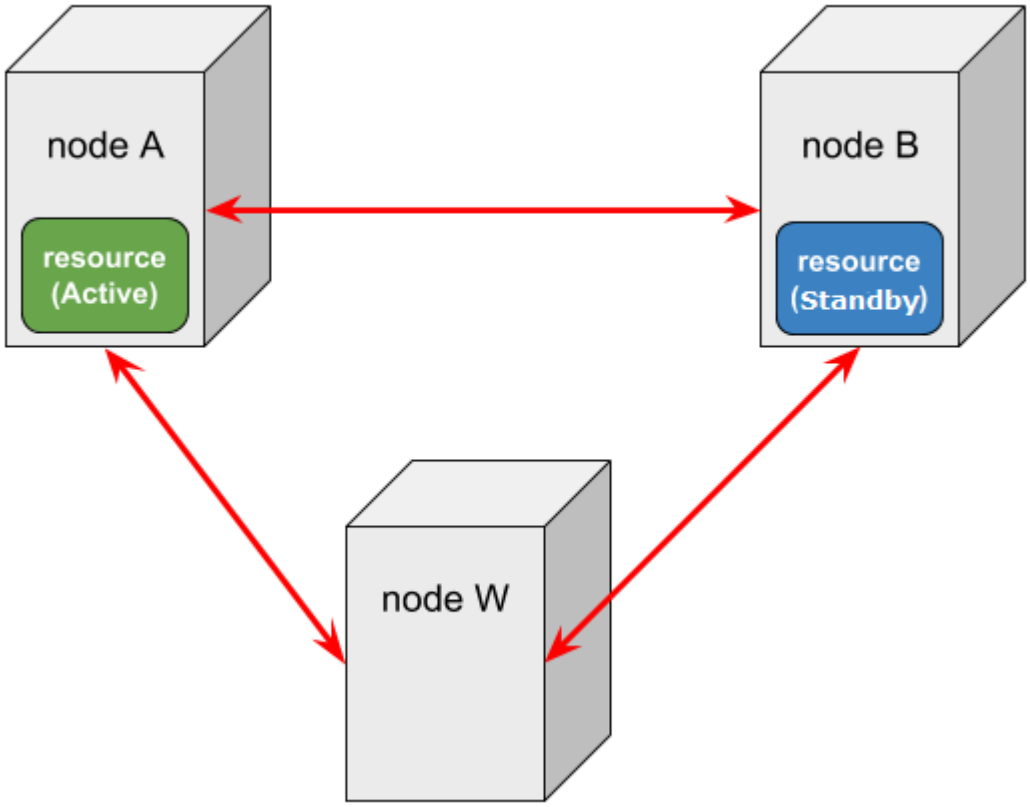
## majority モードで使用可能な Witness モード

本モードでは以下の Witness モードを使用できます。各モードの詳細は「使用可能な Witness モード 」を参照してください。

- *remote_verify*
- *none/off*

# majority モードの期待される動作（標準の設定を仮定）

ノード A（リソースはサービス状態）、ノード B（リソースは待機状態）、そしてノード W（保護対象のリソースを持っていない Witness 専用ノード）の 3 ノード構成のクラスターの振る舞いについて示します。




なお、ノード障害に関するリソースの状態を変更し得るイベントは以下の 3 つです。

- COMM_DOWN イベント
  ノード間のコミュニケーションパスが全て切断した時に呼び出されるイベント。

- COMM_UP イベント
  COMM_DOWN 状態からコミュニケーションパスが復旧した際に呼び出されるイベント。

- LCM_AVAIL イベント
  LCM の初期化が終わった時に呼び出されるイベントで、LifeKeeper 起動時に 1 度だけ呼び出される。起動時に他のノードとコミュニケーションパス通信ができる場合は、必ず COMM_UP イベントの処理より前に LCM_AVAIL イベントの処理が行われる。

## シナリオ 1

**ノード A とノード B の間のコミュニケーションパス通信に障害が発生**

この場合、以下のように動作します。

1. ノード A とノード B は、COMM_DOWN イベントの処理を開始します。ただし、全く同時とは限りません。

2. 両方のノードは Quorum チェックを実行し、両方共自身が Quorum を持っていると判断します（ノード A とノード B の両方からノード W が見えているため、既知の 3 ノードのうちの 2 ノードと疎通があるため多数派にいると判断します）。

3. 各ノードは、まだ通信可能な別のノードに対し、自ノードと通信できなくなったノードの状態について問い合わせ（Witness チェック）を行います。このシナリオでは、ノード A がノード B のステータスについてノード W に問い合わせ、ノード B がノード A のステータスについてノード W に問い合わせることになります。

4. ノード A とノード B は共に ノード W への問い合わせによって他方のノードがまだ生存していると判断し、フェイルオーバー処理は発生しません。リソースはノード A でサービス状態のままになります。

## シナリオ 2

**ノード A とノード W の間のコミュニケーションパス通信に障害が発生**

Quorum/Witness パッケージがインストールされていると、すべてのノードが Witness ノードとして動作するため、このシナリオは前のシナリオと同じになります。この場合、ノード A とノード W は共にノード B へ問い合わせを行い、それにより他方のノードがまだ生存していると判断します。

## シナリオ 3

**ノード A に障害が発生してノード A が停止**

この場合、ノード B は以下の動作をします。

1. ノード A との COMM_DOWN イベントの処理を開始します。

2. ノード W と通信が可能であるので、Quorum を持っていると判断します。

3. ノード A が見えないことをノード W に確認した後、通常のフェイルオーバー処理を開始します。

4. これにより保護対象のリソースはノード B でサービス中になります。

**リソースがノード B でサービス中に、他のノードと通信可能な状態でノード A のサーバーが電源 ON**

この場合、ノード A は LCM_AVAIL イベントを処理します。ノード A は、Quorum を持っていると判断し、ノード B でリソースがサービス中のためリソースを起動させません。

また、その直後に、ノード A とノード B、ノード A とノード W との COMM_UP イベントが各ノードで発生しします（ノード A では２回発生する）。各ノードの COMM_UP イベントの処理では、Quorum を持っていると判断し、またノード B でリソースがサービス中のためリソースは起動させません。

**リソースがノード B でサービス中に、他のノードと通信不能な状態でノード A のサーバーが電源ON**

この場合、ノード A は LCM_AVAIL イベントを処理し、ノード B とノード W はノード A と通信できないので何もしません。ノード A は、３ノードのうちの１ノード（自分自身）としか通信できないため Quorum を持っていないと判断します。Quorum を持っていないため、ノード A はリソースを起動させません。

## シナリオ４

**ノード A のネットワークに障害が発生（ノード A は稼働しているが、他のノードと通信不能）**

この場合、ノード A は以下の動作をします。

1. ノード B との COMM_DOWN イベントの処理を開始します（ほぼ同時に、ノード W との COMM_DOWN イベントの処理も開始されます）。
2. ノード B ともノード W とも通信ができないため、Quorum を持っていないと判断します。
3. LifeKeeper に設定されている QUORUM_LOSS_ACTION パラメータの設定に基づいてノードの強制再起動や強制停止などを行います。QUORUM_LOSS_ACTION のデフォルトは「fastkill」に設定されています。詳細は Quorum/Witness を参照してください。

また、ノード B は以下の動作をします。

1. ノード A との COMM_DOWN イベントの処理を開始します。
2. ノード W とまだ通信が可能であるため、Quorum を持っていると判断します。
3. ノード A が見えないことをノード W に確認（Witness チェック）した後、通常のフェイルオーバー動作を開始します。
4. これにより保護対象のリソースはノード B でサービス中になります。

**リソースがノード B でサービス中に、ノード A の通信が回復**

この場合、ノード B は、 COMM_UP イベントを処理し、Quorum を持っている（３ノードすべてが見える）ことと、サービス中のリソースを持っていることを判断します。ノード A は、COMM_UP イベントを処理し、Quorum を持っていることと、リソースが別のノードでサービス中であることを判断します。この時点では、ノード A はリソースを起動させません。

## シナリオ５

**3つのノードすべてが他のノードと通信不能**

この場合、ノード A は以下の動作をします。

1. ノード B との COMM_DOWN イベントの処理を開始します（ほぼ同時に、ノード W との COMM_DOWN イベントの処理も開始されます）。

2. ノード B ともノード W とも通信ができないため、Quorum を持っていないと判断します。

3. LifeKeeper に設定されている QUORUM_LOSS_ACTION パラメータの設定に基づいてノードの強制再起動や強制停止などを行います。QUORUM_LOSS_ACTION のデフォルトは「fastkill」に設定されています。詳細は Quorum/Witness を参照してください。

また、ノード B は以下の動作をします。

1. ノード A との COMM_DOWN イベントの処理を開始します。

2. ノード A ともノード W とも通信ができないため、Quorum を持っていないと判断します。

3. サービス状態のリソースがないので、強制停止はしません。

コミュニケーションパスが復旧した場合、ノード A 上でリソースが起動します。なお、本動作となるためには、以下の 2条件を満たす必要があります。

- ノード A のリソースに設定された初期化設定値が AUTORES_ISP であること。
- リソースにセットされた優先度がノード A で最も高い値であること。

# 5.4.2.10.3.2. tcp_remote モード

コミュニケーションパスから独立したホストに対して、指定されたポート上の TCP/IP サービスに接続確認を行い、Quorum チェックを行います。指定されたホストの過半数に接続できることで、Quorum チェック成功と判断します。
本モードは２ノードのクラスターでも動作可能です。ただし、Witness モードを *remote_verify* で使用する場合は３ノード必要になります。また、別途、接続確認用のホストが必要です。

**注記:** 多数決ベースのため、指定するホスト数は奇数にすることを推奨します。
**注記:** このモードは本質的に柔軟性と複雑さを備えるため、使用するには、LifeKeeper および関連する特定のネットワーク／クラスター設定の両方に対する十分な理解と注意が必要です。

## tcp_remote モードの設定

`/etc/default/LifeKeeper` 設定ファイルで QUORUM_MODE の設定を *tcp_remote* に設定してください。また、他に以下の設定があります。詳細は「Quorumパラメーター一覧」を参照してください。

- QUORUM_HOSTS
- QUORUM_TIMEOUT_SECS

## tcp_remote モードで使用可能な Witness モード

本モードでは以下の Witness モードを使用できます。各モードの詳細は「使用可能な Witness モード」を参照してください。

- *remote_verify*
- *none/off*

# 5.4.2.10.3.3. storage モード

クラスター内のすべてのノードからアクセスできる共有ストレージを用いた合意システムで、共有ストレージを介してノードの情報交換を行います。各ノードは自ノード情報を定期的に共有ストレージに書き込み、また定期的に他ノードが書き込んだノード情報を読み込みます。この共有ストレージに書き込まれたノード情報を QWK オブジェクトと呼びます。QWK オブジェクトはクラスターを構成するノード数だけ必要です。
Quorum チェックでは、共有ストレージにアクセスできることで Quorum チェック成功と判断します。また、Witness チェックでは、障害が疑われるノードの QWK オブジェクトを確認し、一定期間内に更新されていると障害なし、更新が止まっていれば障害が発生していると判断します。なお、実装上は、Quorum チェック時に Witness チェックも実施しています。
なお、Quorum モードに storage を選んだ場合、後述の Witness モードも storage を選択しなければなりません。
本モードは 2 ノードのクラスターから動作可能で、最大 4 ノードまでサポートされます。別途、クラスターを構成するノード数分の QWK オブジェクトを格納するための共有ストレージが必要です。なお共有ストレージにアクセスできなくなると、リソースの起動に影響します。すべてのノードから常時アクセス可能な共有ストレージを選定してください。

**注記:** 共有ストレージは各ノードからアクセス可能なストレージのことです。この共有ストレージは Quorum/Witness 機能専用であり、LifeKeeper がリソースとして保護することはできません。

**注記:** 本モードを利用するには、設定後に初期化が必要です（後述の「storage モードの使用手順」参照）。また、初期化後にクラスター内のノードの追加･削除や設定の変更を行う場合は、再初期化が必要です。

**注記:** クラスター内に”-”と”.”だけ異なるホスト名のノードが存在する場合（例：lifekeeper-siosと lifekeeper.sios）、本モードを利用することができません。本モードを利用する場合はホスト名を変更してください。

## 利用可能な共有ストレージ

Quorum/Witness 機能の目的であるスプリットブレイン回避を確実なものにするために、Quorum 分割（例：ノード A の QWK オブジェクトにアクセス可能であるが、ノード B の QWK オブジェクトにアクセス不能）が発生しない構成としなければなりません。そのため、クラスター内のすべての QWK オブジェクトが同一共有ストレージ上の同一種のブロックデバイス･レギュラーファイル･S3 オブジェクトに配置されていることを条件とします。

利用可能な共有ストレージは以下の通りです。/etc/default/LifeKeeper 設定ファイルの QWK_STORAGE_TYPE で指定します。

| QWK_STORAGE_TYPE | QWK オブジェクトの配置場所 |
|---|---|
| block | 共有ストレージに物理ストレージやRDM（物理互換）、iSCSI（VM 内イニシエーター）を使う場合、１つのQWK オブジェクトは以下のいずれかに配置してください。 |

| | |
|---|---|
| | (a) **1 QWK** オブジェクト **= 1** パーティション<br><br>(b) **1 QWK** オブジェクト **= 1 LU**<br><br>(a) の場合、複数のホストが１つの LU に書き込みを行いますので、パーティションの LU 内のオフセットは 4K（ストレージ装置のセクターサイズ）でアラインしてください。また、他用途のパーティションを混在させないでください。<br>(b) の場合、LU 内部にパーティションは作成しないでください。<br>また、(a) と (b) いずれの場合も、ファイルシステムを作成する必要はありません。 |
| | 共有ストレージにVMDKを使う場合、１つのQWK オブジェクトは以下に配置してください。<br><br>**1 QWK** オブジェクト **= 1 VMDK**<br><br>パーティションは作成しないでください。また、ファイルシステムを作成する必要はありません。<br>VMDKのプロビジョニングオプションは以下の通り設定してください。<br><br>thick (eager zeroed) |
| file | 共有ストレージにNFSを使う場合、１つのQWK オブジェクトは以下に配置してください。<br><br>**1 QWK** オブジェクト **= NFS** ファイルシステム内の **1** レギュラーファイル<br><br>NFSサーバーの export オプションは以下の通り設定してください。<br><br>rw,no_root_squash,sync,no_wdelay<br><br>NFSの mount オプションは以下の通り設定してください。<br><br>soft,timeo=20,retrans=1,noac<br><br>OS再起動後、自動でマウントするように `/etc/fstab` 等の設定を行ってください。 |
| aws_s3 | 共有ストレージに Amazon Simple Storage Service (S3) を使う場合、１つの QWK オブジェクトは以下に配置してください。 |

| | |
|---|---|
| | **1 QWK オブジェクト = 1 S3 オブジェクト**<br><br>LifeKeeper が動作しているインスタンスのリージョンと別のリージョンの S3 を利用してください。また、S3 のデータ整合性モデルから、QWK オブジェクト更新直後に読み込むと古いデータを読み込む可能性があります。そのため、S3を使用する場合だけ、１つのノードに異なるリージョンの２つのQWK オブジェクトを指定することが可能です。<br>クラスターを構成するすべてのノードで以下の要件を満たしている必要があります。<br><br>• AWS Command Line Interface (AWS CLI) がインストールされていて、rootユーザで使用可能であること。インストール方法は、「AWS Command Line Interface のインストール 」を参照してください。<br>• Amazon S3 におけるエンドポイント（AWS のリージョンとエンドポイント ）にプロトコル HTTP および HTTPS を使用してアクセスできること。<br>• EC2 の IAM ロール や AWS CLI の設定 を適切に行い、root ユーザから S3 オブジェクトにアクセスできること。<br><br>注記: 設定ファイル /etc/default/LifeKeeper のパラメーター PATH に AWS CLI 実行ファイルのパスが設定されていない場合、PATH に AWS CLI 実行ファイルのパスを追加してください。 |

1 QWK オブジェクトのサイズは4096バイトです。

自ノード QWK オブジェクトに対しては read / write を行います。他ノードの QWK オブジェクトに対しては read のみ行います。適切にアクセス権限を設定してください（Persistent Reservation などのアクセス権制限を伴う場合は要注意）。

**注記:** vSphere の4K ネイティブディスクは QWK オブジェクトの共有ストレージとして利用することはできません。

## storage モードの設定

`/etc/default/LifeKeeper` 設定ファイルで QUORUM_MODE と WITNESS_MODE の設定を storage に設定してください。また、他に以下の設定があります。詳細は「Quorumパラメーター一覧 」を参照してください。

- QWK_STORAGE_TYPE
- QWK_STORAGE_HBEATTIME
- QWK_STORAGE_NUMHBEATS

- QWK_STORAGE_OBJECT_<ホスト名>
- HTTP_PROXY, HTTPS_PROXY, NO_PROXY

## storage モードの使用手順

本モードで使用する場合は、初期化が必要です。以下の手順に従って初期化してください。

1. クラスターを構成するすべてのサーバーをセットアップして、他のサーバーとネットワーク通信ができることを確認します。
2. クラスターを構成するすべてのサーバーに LifeKeeper をインストールします。その際、setup コマンドで「Use Quorum / Witness functions」を有効にして、Quorum/Witness パッケージもインストールしてください。
3. すべてのノード間にコミュニケーションパスを作成します。
4. すべてのノードで /etc/default/LifeKeeper 設定ファイルを編集し、storage モードの設定を行います。

> ✻ Storage Quorum を使用するには、LCMNUMHBEATS を増やして、パスが失敗としてマークされるまでの時間を長くすることを 推奨します 。LCMNUMHBEATS を 9 にすることで、タイムアウト時間がデフォルトの15秒から45秒に変更されます。なお、LCMNUMHBEATS を変更した場合は、LifeKeeper を再起動してください。

> ! qwk_storage_init を実行する 前 に、すべてのコミュニケーションパスが稼働していてAliveであることを確認してください。

5. すべてのノードで qwk_storage_init コマンドを実行します。このコマンドは、すべてのノードで QWK オブジェクトの初期化が終わるまで待ち状態になります。このコマンドがすべてのノードで終了すると、storage モードとして Quorum/Witness 機能が利用できる状態となります。

初期化後にクラスターを構成するノードの追加･削除をする、または /etc/default/LifeKeeper 設定ファイルのパラメーターを変更する場合、再初期化が必要です。以下の手順に従って再初期化してください。

1. すべてのノードで qwk_storage_exit コマンドを実行します。
2. ノードを削除する場合、削除するノードと他すべてのノード間のコミュニケーションパスを削除します。
   ノードを追加する場合、追加するノードと他すべてのノード間にコミュニケーションパスを作成します。
3. すべてのノードで /etc/default/LifeKeeper 設定ファイルを再設定します。

4. すべてのノードで qwk_storage_init コマンドを実行します。

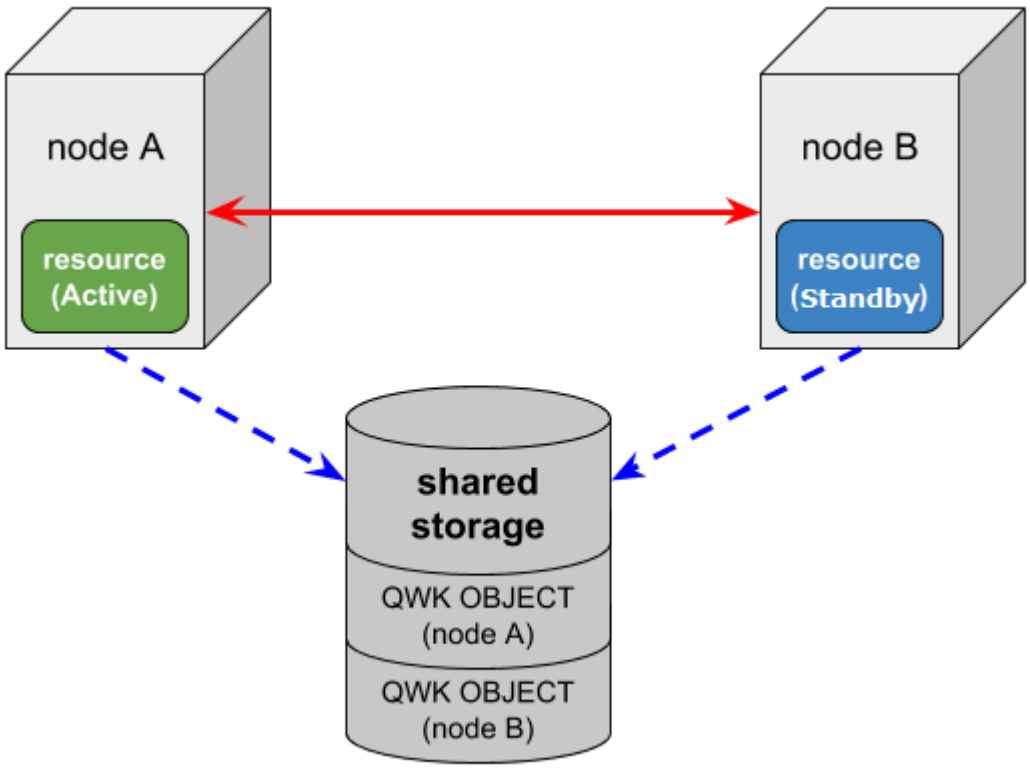
## トラブルシューティング

lifekeeper.log に 135802 メッセージが頻繁に出力されている場合、定期的に行っている QWK オブジェクトの読み書きの負荷が高すぎて、処理が間に合っていません。クラスター内のノード数が多い場合、共有ストレージにS3 を使用している場合（特に 2 リージョン）、QWK オブジェクトの読み書きの負荷が高くなります。
以下の対策を実施して、135802 メッセージが出力されないようにしてください。

- QWK_STORAGE_HBEATTIME の値を増やす（フェイルオーバーに要する時間が長くなります）。
- 共有ストレージが S3 の場合は、1 リージョンにして、ネットワーク的に近いリージョンの S3 を利用する。
- 処理能力を増やす（Amazon EC2 インスタンスの場合、インスタンスタイプを変更するなど）。

## storage モードの期待される動作（標準の設定を仮定）

ノード A（リソースはサービス状態）、ノード B（リソースは待機状態）の 2 ノード構成のクラスターの振る舞いについて示します。




なお、ノード障害に関するリソースの状態を変更し得るイベントは以下の 3 つです。

- COMM_DOWN イベント
  ノード間のコミュニケーションパスが全て切断した時に呼び出されるイベント。

- COMM_UP イベント
  COMM_DOWN 状態からコミュニケーションパスが復旧した際に呼び出されるイベント。

- LCM_AVAIL イベント
  LCM の初期化が終わった時に呼び出されるイベントで、LifeKeeper 起動時に 1 度だけ呼び出される。起動時に他のノードとコミュニケーションパス通信ができる場合は、必ず COMM_UP イベントの処理より前に LCM_AVAIL イベントの処理が行われる。

## シナリオ 1

**ノード A とノード B の間のコミュニケーションパス通信に障害が発生（ノード A とノード B とも共有ストレージにアクセス可能）**

この場合、以下のように動作します。

1. ノード A とノード B は、COMM_DOWN イベントの処理を開始します。ただし、全く同時とは限りません。

2. 両方のノードは Quorum チェックを実行し、両方共自身が Quorum を持っていると判断します（共有ストレージアクセスが可能であるため）。

3. 各ノードは、通信できなくなったノードの QWK オブジェクトの更新を確認します（Witness チェック）。ノード A とノード B は共に稼働しているため定期的に QWK オブジェクトを更新され、Witness チェックでその更新を確認します。

4. Witness チェックの結果、他方のノードがまだ生存しているためフェイルオーバー処理は発生しません。リソースはノード A でサービス状態のままになります。

## シナリオ 2

**ノード A に障害が発生してノード A が停止**

この場合、ノード B は以下の動作をします。

1. ノード A との COMM_DOWN イベントの処理を開始します。

2. 共有ストレージにアクセス可能であるので、Quorum を持っていると判断します。

3. ノード A の QWK オブジェクトの更新が停止していることを確認します（Witness チェック）。

4. Witness チェックの結果、ノード A で障害が発生していると判断したため通常のフェイルオーバー処理を開始します。これにより保護対象のリソースはノード B でサービス中になります。

**リソースがノード B でサービス中に、通信可能かつ共有ストレージにアクセス可能な状態でノード A のサーバーが電源 ON**

この場合、ノード A は LCM_AVAIL イベントを処理します。ノード A は、Quorum を持っていると判断し、ノード B でリソースがサービス中のためリソースを起動させません。
また、その直後に、COMM_UP イベントが各ノードで発生しします。各ノードの COMM_UP イベントの処理では、Quorum を持っていると判断し、またノード B でリソースがサービス中のためリソースは起動させません。

**リソースがノード B でサービス中に、通信不能かつ共有ストレージにアクセス可能な状態でノード A のサーバーが電源 ON**

この場合、ノード A は LCM_AVAIL イベントを処理します。ノード A は、Quorum を持っていると判断し、通信できないノード B に障害が発生しているか確認するためノード B の QWK オブジェクトの更新を確認します。ノード B は稼働しているため、QWK オブジェクトは定期的に更新されます。ノード A では QWK オブジェクトの更新を確認し、ノード B が生存しているが通信できないためにリソースは起動させません。
ノード B はノード A と通信できないので何もしません。

## シナリオ3

**ノード A のネットワークに障害が発生（ノード A は稼働しているが、他のノードや共有ストレージにアクセス不能）**

この場合、ノード A は以下の動作をします。

1. ノード B との COMM_DOWN イベントの処理を開始します。
2. 共有ストレージにアクセス不能であるため、Quorum を持っていないと判断します。
3. 直ちに 強制停止します（QUORUM_LOSS_ACTION のデフォルト「fastkill」の振る舞い）。

また、ノード B は以下の動作をします。

1. ノード A との COMM_DOWN イベントの処理を開始します。
2. 共有ストレージにアクセス可能であるため、Quorum を持っていると判断します。
3. ノード A の QWK オブジェクトの更新が停止していることを確認します（Witness チェック）。
4. Witness チェックの結果、ノード A で障害が発生していると判断したため通常のフェイルオーバー処理を開始します。これにより保護対象のリソースはノード B でサービス中になります。

**リソースがノード B でサービス中に、共有ストレージにアクセス可能な状態でノード A のサーバーが電源 ON**

シナリオ２と同じです。

**リソースがノード B でサービス中に、共有ストレージにアクセス不能な状態でノード A のサーバーが電源 ON**

この場合、ノード A は LCM_AVAIL イベントを処理します。ノード A は、Quorum を持っていないと判断し、リソースを起動させません。
ノード B とのコミュニケーションパス通信が可能であれば、その後に COMM_UP イベントが発生します。しかし、Quorum を持っていないので、ノード A ではリソースを起動させません。

# 5.4.2.10.4. STONITH

STONITH (Shoot the Other Node in the Head) は、クラスタ内のノードの電源をリモートから切断するフェンシング方式です。LifeKeeper の STONITH 機能は、外部電源スイッチコントロール、IPMI 対応マザーボード、およびハイパーバイザーが提供する電源機能を利用してクラスタ内の他のノードの電源を切断します。

## STONITH で IPMI を使用する

IPMI (Intelligent Platform Management Interface) は、コンピュータシステムにアクセスする共通インターフェースのセットを提供します。IPMI を使用すると、システムの状態を監視してシステムを管理できます。IPMI を STONITH で使用すると、故障と思われるクラスタノードの電源スイッチをクラスタソフトウェアが制御することにより、シリアル接続またはネットワーク接続を介してノードの電源切断やリブートができるため、故障ノードが共有データにアクセスしたりデータを破損するのを確実に防ぎます。

### パッケージの要件

- IPMI ツールのパッケージ (ipmitool-1.8.11-6.el6.x86_64.rpm など)

## VMware vSphere 環境での STONITH

vCLI (vSphere Command-Line Interface) は、ESXi ホストと仮想マシンを含む仮想インフラストラクチャを管理するための VMware でサポートされているコマンドラインインターフェースです。vCLI コマンドがお使いの環境のニーズに合致する場合は、VMware の仮想マシン間での LifeKeeper STONITH の実装に適用することができます。

### パッケージの要件

#### STONITH サーバ

- ◦ VMware vSphere SDK パッケージまたは VMware vSphere CLI (vSphere CLI は、vSphere SDK と同じインストールパッケージに含まれています)

#### 監視対象仮想マシン

- ◦ VMware Tools

## AWS 環境での STONITH

AWS 上での STONITH については、こちら を参照ください。

## インストールと設定

LifeKeeper をインストールし、クラスタ内の各ノードでコミュニケーションパスを設定した後、STONITH

をインストールおよび設定します。

1. 次のコマンドを実行して LifeKeeper STONITH スクリプトをインストールします。

   ```
   /opt/LifeKeeper/samples/STONITH/stonith-install
   ```

2. (*IPMI を利用する場合のみ) BIOS または ipmitool コマンドを使用して、以下の BMC (Baseboard Management Controller) 変数を設定します。

   - 静的 IP アドレスの使用
   - IP アドレス
   - サブネットマスク
   - ユーザ名
   - パスワード
   - ユーザに管理者権限を追加
   - ユーザのネットワークアクセスを有効化

   `ipmitool` コマンドの使用例を示します。

   (詳細については、 ipmitool のマニュアルページを参照してください。)

```
# ipmitool lan set 1 ipsrc static
# ipmitool lan set 1 ipaddr 192.168.0.1
# ipmitool lan set 1 netmask 255.0.0.0
# ipmitool user set name 1 root
# ipmitool user set password 1 secret
# ipmitool user priv 1 4
# ipmitool user enable 1
```

3. 設定ファイルを編集します。

   設定ファイルを編集し、STONITH を有効にして電源を切断するコマンドラインを追加します。 **注記:** フェンシングループ (2 台のマシン間で、通信は喪失しているもののお互いをまだ STONITH できる場合に、お互いに電源オフとリブートをし続ける状態) を回避するため、リブートではなく電源オフを推奨します。

   ```
   /opt/LifeKeeper/config/stonith.conf
   ```

```
# LifeKeeper STONITH configuration
#
```

```
# Each system in the cluster is listed below. To enable STONITH for a
# given system,
# remove the '#' on that line and insert the STONITH command line to power off
# that system.


# Example1: ipmi command


# node-1 ipmitool -I lanplus -H 10.0.0.1 -U root -P secret power off


# Example2: vCLI-esxcli command


# node-2 esxcli --server=10.0.0.1 --username=root --password=secret vms vm kill --type='hard' --
world-id=1234567


# Example3: vCLI-vmware_cmd command


# node-3 vmware-cmd -H 10.0.0.1 -U root -P secret <vm_id> stop hard


minute-maid ipmitool -I lanplus -H 192.168.0.1 -U root -P secret power off
kool-aid ipmitool -I lanplus -H 192.168.0.2 -U root -P secret power off


vm1 esxcli --server=10.0.0.1 --username=root --password=secret vms vm kill --type='hard' --world-
id=1234567
vm2 vmware-cmd -H 10.0.0.1 -U root -P secret <vm_id> stop hard
```

## <vm_id>

vSphere CLI コマンドは、vSphere SDK for Perl の上で実行されます。<vm_id > は、VM の識別子として使用されています。この変数によって、設定対象の VM 用の設定ファイルを指定します。

設定ファイルのパスを調べるには、以下の手順を実行してください。

1. 次のコマンドを実行します。

```
vmware-cmd -H <vmware host> -l
```

2. 上記のコマンドによって VMware ホストのリストが表示されます。

`vmware-cmd -l` の出力例を以下に示します (3 台の VM を表示)。

```
/vmfs/volumes/4e08c1b9-d741c09c-1d3e-0019b9cb28be/lampserver/
lampserver.vmx

/vmfs/volumes/4e1e1386-0b862fae-a859-0019b9cb28bc/oracle10/
oracle.vmx

/vmfs/volumes/4e08c1b9-d741c09c-1d3e-0019b9cb28be/lampserver02/
lampserver02.vmx
```

出力されたリストで設定中の VM を見つけます。

3. パス名を <vm_id> 変数にペーストします。上記の例では以下の通りになります。
   vmware-cmd -H 10.0.0.1 -U root -P secret /vmfs/volumes/
   4e08c1b9-d741c09c-1d3e-0019b9cb28be/lampserver/lampserver.vmx stop hard

注記: VMware コマンドの詳細については、引数なしで vmware-cmd を実行するとすべてのオプションに関するヘルプページが表示されます。

## 期待される動作

LifeKeeper がノードとの通信障害を検出すると、そのノードの電源が切断され、フェイルオーバが発生します。問題を修復した後に、手動でそのノードの電源を入れる必要があります。

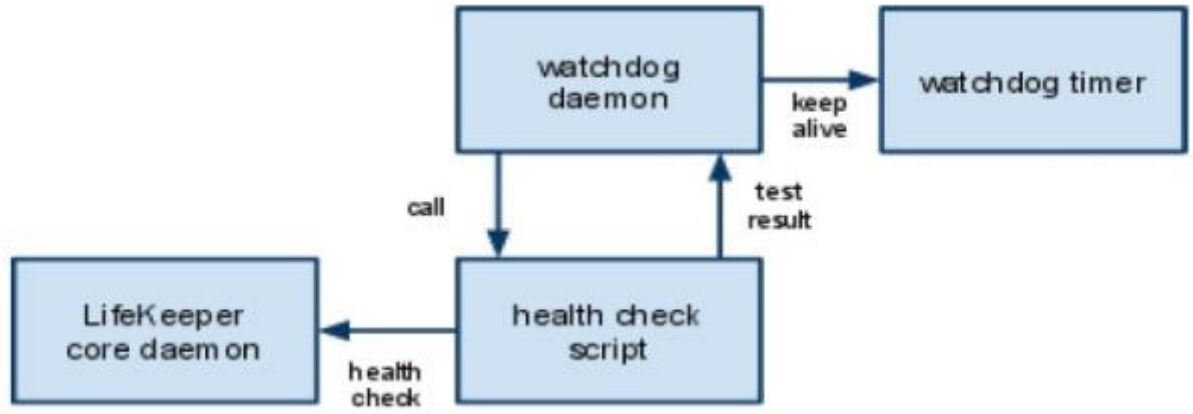
# 5.4.2.10.5. Watchdog

Watchdog は、サーバーが正常に動作しない場合に、問題の発生を予防する修正処置 (リブート) を確実に実行できるようにサーバーを監視する方法です。 Watchdog は、特別な Watchdog ハードウェアを使用して実装する場合と、ソフトウェアのみのオプションを使用して実装する場合があります。

> **注記:** この構成は、Red Hat Enterprise Linux Versions 7 でのみ検証されています。他のオペレーティングシステムでは検証されていないため、現時点ではサポートされません。

## コンポーネント

- Watchdog timer - ソフトウェアドライバまたは外部ハードウェアコンポーネント
- Watchdog daemon - 該当する Linux ディストリビューションを通じて rpm が入手可能
- LifeKeeper Core daemon - LifeKeeper のインストールに付随
- health cehck script - LifeKeeper core の動作状況を確認するスクリプト




**LifeKeeper と Watchdog の相互運用性**

次のセクションを注意深く読んでください。デーモンは、エラーからリカバリするように設計されており、注意深く設定しないとデーモンはシステムをリセットします。インストールおよび設定を行う前に慎重に計画してください。このセクションの目的は、 Watchdog についての説明や設定をすることではありません。ここでは、 Watchdog 構成での LifeKeeper との相互運用についての説明や設定のみ行います。

## 設定

以下の手順は、root ユーザ権限を持つ管理者が行う必要があります。管理者は、 Watchdog のリスクおよび問題についてすでに熟知しているものとします。

ヘルスチェックスクリプト (LifeKeeper 監視スクリプト) は、LifeKeeper の設定と Watchdog の設定 (`/opt/LifeKeeper/samples/watchdog/LifeKeeper-watchdog`) を関連付けるコンポーネントです。このスクリプトは、LifeKeeper コアコンポーネントの基本部分を監視できます。

1. 以前に Watchdog を設定していた場合は、次のコマンドを入力して Watchdog を停止します。そうでない場合は、手順 2 に進みます。
   `systemctl stop watchdog`

2. Watchdog ソフトウェアのインストールで供給される Watchdog 設定ファイル (*/etc/watchdog.conf*) を編集します。

   a. test-binary を次のように修正します。

   ```
   test-binary = /opt/LifeKeeper/samples/watchdog/LifeKeeper-watchdog
   ```

   b. test-timeout を次のように修正します。

   ```
   test-timeout = 5
   ```

   c. interval を次のように修正します。

   ```
   interval = 7
   ```

   interval は、LifeKeeper のコミュニケーションパスのタイムアウト (15 秒) よりも小さい値にしてください。約半分の値が妥当です。

3. LifeKeeper が起動していることを確認します。まだの場合は、LifeKeeper の起動 トピックを参照してください。

4. 次のコマンドを入力して Watchdog を起動します。
   `systemctl start watchdog`

5. 今後の再起動の際に Watchdog を自動的に起動させるには、次のコマンドを入力します。
   `systemctl enable watchdog`

✻ **注記:** Watchdog を設定すると、予想外のリブートがときどき発生する可能性があります。これは、 Watchdog の仕組みから来る一般的な性質です。正常に応答しないプロセスがあると、 Watchdog 機能は LifeKeeper (またはオペレーティングシステム) がハングしていると判断し、(警告なしに) システムをリブートします。

## アンインストール

LifeKeeper をアンインストールする場合は、慎重に行ってください。以下に列記の通り、上記の手順を逆の順で実行します。

! **警告:** LifeKeeper を構成する RPM パッケージを削除する方法で LifeKeeper をアンインストールする場合は、 **先に Watchdog を停止してください** 。上記の手順 2 では、

LifeKeeper の Watchdog スクリプトを呼び出すように Watchdog 設定ファイルを修正しています。したがって、先に Watchdog を停止しておかないと、存在しないスクリプトを呼び出すことになります。リブートを実行するこのスクリプトが見つからない場合は、エラーが発生します。この状態は Watchdog を停止するまで継続します。

1. 次のコマンドを入力して Watchdog を停止します。
   `systemctl stop watchdog`

2. Watchdog ソフトウェアのインストールで供給される Watchdog 設定ファイル (*/etc/watchdog.conf*) を編集します。

   a. test-binary および interval の両エントリをコメントアウトします (各行の先頭に # を追加します)。

   ```
   #test-binary =
   #interval =
   ```

   **注記:** interval が他の機能によって以前から使用されていた場合は、そのままにしておくこともできます。

3. LifeKeeper をアンインストールします。LifeKeeper の削除 トピックを参照してください。

4. これで Watchdog を起動し直すことができます。LifeKeeper のみが Watchdog を使用していた場合は、次のコマンドを入力すると Watchdog を永続的に無効にできます。
   `systemctl disable watchdog`

# 5.4.2.10.6. I/O フェンシング機能組み合わせ情報

LifeKeeper for Linux では様々なフェンシング機能を用意しておりますが、LifeKeeper がインストールされるサーバーの構成や、保護対象となるサービスで使用されるストレージの構成により、使用できる適切なフェンシング機能やこれらの組み合わせは多岐にわたります。

各サーバー構成ごとに利用可能なフェンシング機能と適切な組み合わせについては、以下の情報を参照してください。またストレージ構成については各サーバー構成のページに詳細を記載しています。

- 物理サーバー構成の場合は「フェンシング機能利用可否リスト（物理構成）」を参照してください。
- 仮想サーバー構成（VMware）の場合は「フェンシング機能利用可否リスト（仮想構成 VMware編）」を参照してください。

## 各フェンシング機能概要

上述の情報で示される各 I/O フェンシング機能の概要を以下に記載します。

- **SCSI-2 Reserve** リソースがアクティブであるノードからストレージに対し SCIS-2 のリザーブコマンドを発行することにより保護対象の LU をロックし、他ノードからの同時アクセスを抑止します。
- **SCSI-3 Reserve** リソースがアクティブであるノードからストレージに対し SCSI-3 のリザーブコマンドを発行することにより保護対象の LU をロックし、他ノードからの同時アクセスを抑止します。
- **SCSI Fencing** 自ノードのリソースがスタンバイである状態においてコミュニケーションパスの全断を検知した際、他ノードから行われた LU のロックを無効化し、再び自ノードから SCSI-2 または SCSI-3 のリザーブコマンドを発行することにより、保護対象の LU をロックし、他ノードからの同時アクセスを抑止します。
- **IPMI STONITH** コミュニケーションパスの全断を検知した際、通信が途絶した他ノードの IPMI インタフェースに対して、物理サーバーの停止命令を発行することにより、保護対象サービスが両ノードで起動することを防止します。こちらの機能は IPMI ツールを用いるため、IPMI インタフェースを搭載した物理サーバーに対してのみ使用できます。
- **VMware STONITH** コミュニケーションパスの全断を検知した際、通信が途絶した他ノードが稼働する仮想ホストまたは vCenter に対して、仮想マシンの停止命令を発行することにより、保護対象サービスが両ノードで起動することを防止します。こちらの機能は VMware が提供するツールを用いるため、VMware 上に構成した仮想マシンに対してのみ使用できます。
- **Quorum Witness** 事前に調停役を設け、コミュニケーションパスの切断または復旧を検知した際、自ノードへのフェイルオーバーがサービス継続に有効であるかを判断します。LifeKeeper では通常、前述のリザーブコマンドにより複数ノードから LU への同時アクセスを抑止しますが、これらのリザーブを利用しない環境では本機能の利用が必須となります。調停役には3種類のモードがあ

り、Witness 専用ノード、独立ホスト、共有ストレージのいずれかをご利用環境にあわせて選択いただけます。詳しくは Quorum/Witness のページを参照してください。

# 5.4.2.10.6.1. フェンシング機能利用可否リスト（物理構成）

本ページでは、2台の物理サーバーをクラスターノードとする構成を例に、使用できる適切なフェンシング機能の組み合わせをストレージ構成ごとに記載します。

## 共有ディスク構成（SCSI リザーブを用いる単一または複数パスの構成）

本構成は、サポートストレージ一覧 の表に認定済み共有ストレージとして記載されているストレージで SCSI リザーブによる排他制御機能を利用する場合が該当します。
SCSI-3 Reserve および複数パスの制御のためにマルチパスドライバを必要とするストレージを利用する場合、それぞれのマルチパスドライバに対応したマルチパスキットが必要となります。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | | | | |
|---|---|---|---|---|---|---|
| **SCSI-2 Reservations** | ● | ● | ● | ?[1] | ?[1] | ?[1] |
| **SCSI-3 Reservations** | ?[1] | ?[1] | ?[1] | ● | ● | ● |
| **IPMI STONITH** | ?[2] | ?[2] | ?[2] | ?[2] | ?[2] | ?[2] |
| **Quorum/Witness (tcp_remote)** | ◯ | ?[3] | ?[3] | ◯ | ?[3] | ?[3] |
| **Quorum/Witness (majority)** | ?[3] | ◯ | ?[3] | ?[3] | ◯ | ?[3] |
| **Quorum/Witness (storage)** | ?[3] | ?[3] | ◯ | ?[3] | ?[3] | ◯ |

[1] LifeKeeper の仕組み上、単一の共有ディスクに対して SCSI-2 および SCSI-3 のリザーブは共存することができません。

[2] 競合によるサービス停止の可能性がある為、SCSI Fencing と IPMI STONITH は共存することができません。

[3] LifeKeeper の仕組み上、単一のクラスターに対して Quorum Witness の各モードは共存することができ

ません。

### 構成上使えない機能について

- 仮想環境やクラウド環境向けに用意されたフェンシング機能は、本構成では使用できません。

## 共有ディスク構成（SCSIリザーブを用いない単一または複数パスの構成）

本構成は、SCSI / FC / iSCSI / SAS などの方式（NASを除く）により接続されるストレージで、SCSIリザーブによる排他制御機能を利用できない場合が該当します。
複数パスの制御のためにマルチパスドライバを必要とするストレージを利用する場合、それぞれのマルチパスドライバに対応したマルチパスキットが必要となります。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | |
|---|---|---|---|
| **IPMI STONITH** | ◯ | ◯ | ◯ |
| **Quorum/Witness (tcp_remote)** | ● | ?[1] | ?[1] |
| **Quorum/Witness (majority)** | ?[1] | ● | ?[1] |
| **Quorum/Witness (storage)** | ?[1] | ?[1] | ● |

[1] LifeKeeper の仕組み上、単一のクラスタに対して Quorum Witness の各モードの共存は不可

### 構成上使えない機能について

- SCSI-2 Reserve および SCSI-3 Reserve は、SCSIリザーブを用いない前提の本構成では使用できません。
- SCSI Fencing は、SCSI リザーブを用いない前提の本構成では使用できません。
- 仮想環境やクラウド環境向けに用意されたフェンシング機能は、本構成では使用できません。

## データレプリケーション構成

本構成は、DataKeeper を用いて各ノードに接続されたローカルストレージをノード間でレプリケーションする場合が該当します。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。本構成において必須となるフェンシング機能はありませんが、データ保護の観点からフェンシング機能を使用されることを推奨します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | |
|---|---|---|---|
| **IPMI STONITH** | ◯ | ◯ | ◯ |
| **Quorum/Witness (tcp_remote)** | ◯ | ?[1] | ?[1] |
| **Quorum/Witness (majority)** | ?[1] | ◯ | ?[1] |
| **Quorum/Witness (storage)** | ?[1] | ?[1] | ◯ |

[1] LifeKeeper の仕組み上、単一のクラスターに対して Quorum Witness の各モードは共存することができません。

### 構成上使えない機能について

- SCSI リザーブを用いるフェンシング機能は本構成では使用できません。
- 仮想環境やクラウド環境向けに用意されたフェンシング機能は、本構成では使用できません。

## Network Attached Storage (NAS) 構成

本構成は、Network File System (NFS) の方式により接続される NAS ストレージを利用する場合が該当します。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。本構成において必須となるフェンシング機能はありませんが、データ保護の観点からフェンシング機能を使用されることを推奨します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | |
|---|---|---|---|
| **IPMI STONITH** | ◯ | ◯ | ◯ |
| **Quorum/Witness (tcp_remote)** | ◯ | ?[1] | ?[1] |
| **Quorum/Witness (majority)** | ?[1] | ◯ | ?[1] |
| **Quorum/Witness (storage)** | ?[1] | ?[1] | ◯ |

[1] LifeKeeper の仕組み上、単一のクラスターに対してQuorum Witness の各モードは共存することができません。

### 構成上使えない機能について

- 共有ディスクの使用を前提としたフェンシング機能は、本構成では使用できません。
- 仮想環境やクラウド環境向けに用意されたフェンシング機能は、本構成では使用できません。

# 5.4.2.10.6.2. フェンシング機能利用可否リスト（仮想構成 VMware編）

本ページでは、VMware 上の仮想マシンとして作成された2台のサーバーをクラスターノードとする構成を例に、使用できる適切なフェンシング機能の組み合わせをストレージ構成ごとに記載します。

## 共有ディスク構成（SCSI リザーブを用いる単一または複数パスの構成）

本構成は、サポートストレージ一覧 の表に認定済み共有ストレージとして記載されているストレージで SCSI リザーブによる排他制御機能を利用する場合が該当します。
SCSI-3 Reserve および複数パスの制御のためにマルチパスドライバを必要とするストレージを利用する場合、それぞれのマルチパスドライバに対応したマルチパスキットが必要となります。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | | | | |
|---|---|---|---|---|---|---|
| **SCSI-2 Reservations** | ● | ● | ● | ?[1] | ?[1] | ?[1] |
| **SCSI-3 Reservations** | ?[1] | ?[1] | ?[1] | ● | ● | ● |
| **VMware STONITH** | ?[2] | ?[2] | ?[2] | ?[2] | ?[2] | ?[2] |
| **Quorum/Witness (tcp_remote)** | ◯ | ?[3] | ?[3] | ◯ | ?[3] | ?[3] |
| **Quorum/Witness (majority)** | ?[3] | ◯ | ?[3] | ?[3] | ◯ | ?[3] |
| **Quorum/Witness (storage)** | ?[3] | ?[3] | ◯ | ?[3] | ?[3] | ◯ |

[1] LifeKeeper の仕組み上、単一の共有ディスクに対して SCSI-2 および SCSI-3 のリザーブは共存することができません。

[2] 競合によるサービス停止の可能性がある為、SCSI Fencing と VMware STONITH は共存することができません。

[3] LifeKeeper の仕組み上、単一のクラスターに対して Quorum Witness の各モードは共存することができ

ません。

### 構成上使えない機能について

- VMware以外の仮想環境向けに用意されたフェンシング機能は、本構成では使用できません。
- クラウド環境や物理サーバー向けに用意されたフェンシング機能は、本構成では使用できません。

## 共有ディスク構成（SCSIリザーブを用いない単一または複数パスの構成）

本構成は、SCSI / FC / iSCSI / SAS などの方式（NASを除く）により接続されるストレージで、SCSIリザーブによる排他制御機能を利用できない場合が該当します。
複数パスの制御のためにマルチパスドライバを必要とするストレージを利用する場合、それぞれのマルチパスドライバに対応したマルチパスキットが必要となります。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | |
|---|---|---|---|
| **VMware STONITH** | ◯ | ◯ | ◯ |
| **Quorum/Witness (tcp_remote)** | ● | ?[1] | ?[1] |
| **Quorum/Witness (majority)** | ?[1] | ● | ?[1] |
| **Quorum/Witness (storage)** | ?[1] | ?[1] | ● |

[1] LifeKeeper の仕組み上、単一のクラスタに対して Quorum Witness の各モードの共存は不可

### 構成上使えない機能について

- SCSI-2 Reserve および SCSI-3 Reserve は、SCSIリザーブを用いない前提の本構成では使用できません。
- SCSI Fencing は、SCSI リザーブを用いない前提の本構成では使用できません。
- 仮想環境やクラウド環境向けに用意されたフェンシング機能は、本構成では使用できません。

## データレプリケーション構成

本構成は、DataKeeper を用いて各ノードに接続されたローカルストレージをノード間でレプリケーションする場合が該当します。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。本構成において必須となるフェンシング機能はありませんが、データ保護の観点からフェンシング機能を使用されることを推奨します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | |
|---|---|---|---|
| **VMware STONITH** | ◯ | ◯ | ◯ |
| **Quorum/Witness (tcp_remote)** | ◯ | ?[1] | ?[1] |
| **Quorum/Witness (majority)** | ?[1] | ◯ | ?[1] |
| **Quorum/Witness (storage)** | ?[1] | ?[1] | ◯ |

[1] LifeKeeper の仕組み上、単一のクラスターに対して Quorum Witness の各モードは共存することができません。

### 構成上使えない機能について

- SCSIリザーブを用いるフェンシング機能は本構成では使用できません。
- 仮想環境やクラウド環境向けに用意されたフェンシング機能は、本構成では使用できません。

## Network Attached Storage (NAS) 構成

本構成は、Network File System (NFS) の方式により接続される NAS ストレージを利用する場合が該当します。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。本構成において必須となるフェンシング機能はありませんが、データ保護の観点からフェンシング機能を使用されることを推奨します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | |
|---|---|---|---|
| **VMware STONITH** | ◯ | ◯ | ◯ |
| **Quorum/Witness (tcp_remote)** | ◯ | ?[1] | ?[1] |
| **Quorum/Witness (majority)** | ?[1] | ◯ | ?[1] |
| **Quorum/Witness (storage)** | ?[1] | ?[1] | ◯ |

[1] LifeKeeper の仕組み上、単一のクラスターに対してQuorum Witness の各モードは共存することができません。

### 構成上使えない機能について

- 共有ディスクの使用を前提としたフェンシング機能は、本構成では使用できません。
- 仮想環境やクラウド環境向けに用意されたフェンシング機能は、本構成では使用できません。

## VMDK as Shared Storage 構成

本構成は、VMDK as Shared Storage の方式により構成される VMware仮想ハードディスクを利用する場合が該当します。

### 使用できる機能と適切な組み合わせ

以下の表では、本構成で使用できるフェンシング機能と適切な組み合わせパターンを示します。

### 各記号の意味

● – 利用必須
◯ – オプションとして利用可能
? – 利用不可

表中で「オプションとして利用可能」としている機能については、組み合わせの中で当該機能の利用は必須ではありません。表中で「利用不可」としている機能については、表下の注記を参照してください。

| | 組み合わせパターン | | |
|---|---|---|---|
| **VMware STONITH** | ?[1] | ?[1] | ?[1] |
| **Quorum/Witness (tcp_remote)** | ◯ | ?[2] | ?[2] |
| **Quorum/Witness (majority)** | ?[2] | ◯ | ?[2] |
| **Quorum/Witness (storage)** | ?[2] | ?[2] | ◯ |

[1] 競合によるサービス停止の可能性がある為、VMDK as Shared Storage と VMware STONITH は共存することができません。

[2] LifeKeeper の仕組み上、単一のクラスターに対してQuorum Witness の各モードは共存することができません。

### 構成上使えない機能について

- 共有ディスクの使用を前提としたフェンシング機能は、本構成では使用できません。

# 5.4.2.11. リソースポリシー管理

## 概要

SIOS Protection Suite for Linux のリソースポリシー管理では、リソースのローカルリカバリとフェイルオーバの動作管理機能が提供されます。リソースポリシーは、 **lkpolicy** コマンドラインツール (CLI) を使用して管理できます。

## SIOS Protection Suite

SIOS Protection Suiteには、個々のアプリケーションおよび関連し合うアプリケーションのグループを監視する機能があり、定期的にローカルリカバリを実行したり、保護下のアプリケーションに障害が発生したときに通知することができます。関連し合うアプリケーションの例としては、主アプリケーションが下位のストレージまたはネットワークリソースに依存する階層などがあります。アプリケーションまたはリソースに障害が発生した場合のデフォルトの動作は以下の通りです。

1. **ローカルリカバリ**：最初に、リソースまたはアプリケーションの **ローカル** でリカバリを試みます。このときは、外部の介入なしにローカルサーバ上でリソースまたはアプリケーションをリストアしようとします。ローカルリカバリが成功した場合、SIOS Protection Suite は追加のアクションを実行しません。

2. **フェイルオーバ**：次に、ローカルリカバリでリソースまたはアプリケーションのリストアに失敗した (またはリソースを監視するリカバリキットがローカルリカバリをサポートしていない) 場合、 **フェイルオーバ** は開始されます。フェイルオーバ処理では、クラスタ内の別のサーバ上で該当アプリケーション (および依存するすべてのリソース) を起動しようと試みます。

リカバリ動作の詳細については、SIOS Protection Suite 障害検出とリカバリのシナリオ を参照してください。

## ポリシーによるカスタム動作およびメンテナンスモード動作

SIOS Protection Suite Version 7.5 以降では、デフォルトのリカバリ動作を変更する追加ポリシーを設定する機能をサポートします。リソース単位またはサーバ単位で、4 つのポリシーが設定可能です (リソース単位のポリシーに関する注意については下のセクションを参照してください)。 **サーバレベルでポリシーを変更する方法を推奨します。**

利用可能なポリシーは以下の通りです。

### 標準ポリシー

- **Failover** - このポリシー設定を使用すると、リソースフェイルオーバを有効 / 無効にできます。(**S**: リザベーションが適切に処理されるには、 **フェイルオーバ** は、個々の SCSI リソースで無効にすることはできません。)

- **LocalRecovery** - SIOS Protection Suite は、デフォルトでは、フェイルオーバを実行する前に、個々のリソースまたは保護対象アプリケーション全体を再起動することにより、保護対象リソースの

リカバリを試みます。このポリシー設定を使用すると、ローカルリカバリを有効 / 無効にできます。

- **TemporalRecovery** - 通常、SIOS Protection Suite は、障害リソースのローカルリカバリを実行します。ローカルリカバリに失敗すると、SIOS Protection Suite は、リソース階層を別ノードにフェイルオーバします。ローカルリカバリに成功した場合は、フェイルオーバは実行されません。

  ローカルリカバリに成功した場合でも、サーバの何らかの異常によって短時間の間にローカルリカバリが再試行される場合があり、結果として何度も連続してローカルリカバリが試行されることになります。これが発生すると、問題のアプリケーションは可用性が悪化します。

  この反復的なローカルリカバリ / 障害サイクルを回避するために、時間的リカバリポリシーを設定できます。時間的リカバリポリシーを使用すると、管理者は指定した時間内に試行するローカルリカバリの回数を (成功かどうかにかかわらず) 制限することができます。

  > *例:* リソースが試行するローカルリカバリの回数を 30 分間で 3 回に限定するポリシー定義をユーザが設定した場合、30 分以内に 3 回目のローカルリカバリが試行されると、SIOS Protection Suite はフェイルオーバを実行します。

  定義した時間的リカバリポリシーは有効または無効にできます。時間的リカバリポリシーが無効の場合、時間的リカバリ処理は継続して実行され、ポリシーが適用されるはずの時間に通知がログに表示されますが、実際のアクションは実行されません。

**注記:** 時間的リカバリポリシーを設定した状態で、フェイルオーバとローカルリカバリの一方または両方を無効にすることは可能です。フェイルオーバまたはローカルリカバリを無効にした場合に、時間的リカバリポリシーは実行されることが **ない** ため、この状態は非論理的です。

### メタポリシー

「メタ」ポリシーは、他の複数のポリシーに影響を与える可能性があるポリシーです。通常、これらのポリシーは、標準ポリシーであれば複数個の設定が必要になるような特定のシステム動作を実現するためのショートカットとして使用します。

- **NotificationOnly** - このモードでは、管理者は SIOS Protection Suite を「監視専用」状態にすることができます。 **1 つのリソース (または、サーバ単位のポリシーの場合はすべてのリソース) のローカルリカバリおよびフェイルオーバの両方が影響を受けます。** 障害が検知されると、ユーザインターフェースには **Failure** 状態が表示されます。 ただし、リカバリもフェイルオーバも実行されません。 **注記:** 管理者は、障害の原因となった問題を手動で修正し、障害が起きたリソースを復帰させて通常の SIOS Protection Suite の運用を継続する必要があります。

## リソースレベルのポリシーに関する重要な考慮事項

リソースレベルのポリシーとは、リソース階層全体またはサーバレベルのポリシーとは異なり、特定のリソースにのみ適用されるポリシーです。

例：

アプリケーション

- IP

- file system

上記のリソース階層では、アプリケーションは IP とファイルシステムの両方に依存しています。ポリシーは、特定のリソースのローカルリカバリまたはフェイルオーバを無効にするように設定できます。これは、例えば、IP リソースのローカルリカバリが失敗し、IP リソースのフェイルオーバが無効に設定されていた場合、IP リソースはフェイルオーバを実行せず、他のリソースのフェイルオーバも発生させないことを意味します。ただし、ファイルシステムリソースのローカルリカバリが失敗し、ファイルシステムリソースのポリシーのフェイルオーバが無効化されていない場合、階層全体がフェイルオーバを実行します。

**注記:** 重要事項として、リソースレベルのポリシーは設定対象の特定のリソースにのみ適用されることに注意してください。

上記は単純な例です。複雑な階層を構成することもできるため、リソースレベルのポリシーを設定するときは注意してください。

既知の問題を参照してください。 *(“フェイルオーバ時に誤った順序でリソースが停止されます“の項目)*

# lkpolicy ツール

lkpolicy ツールは、SIOS Protection Suite for Linux が稼働するサーバのポリシーを管理 (参照、設定、削除) するためのコマンドラインツールです。`lkpolicy` は、ポリシーの設定および修正、ポリシーの削除、利用可能なポリシーと現在の設定値の表示をサポートします。さらに、設定したポリシーは、有効または無効に設定できるため、リカバリ動作に影響を与えながらリソース / サーバ設定を保持できます。

全体的な使用方法は次の通りです。

```
lkpolicy [--list-policies | --get-policies | --set-policy | --remove-policy]
<name value pair data…>
```

`<name value pair data…>`は、運用方法および対象のポリシーによって異なります (特にポリシーを設定する場合)。例： 有効 / 無効タイプのポリシーのほとんどでは、必要なのは `[--on]` または `[--off]` のスイッチのみですが、時間的ポリシーの場合は、しきい値を設定するための値も必要です。

## lkpolicy の使用方法の例

### ローカルおよびリモートサーバとの認証

**lkpolicy** ツールは、サーバが公開する API を通じて SIOS Protection Suite サーバと通信します。この API は、 lkpolicy ツールなどのクライアントに対して認証を要求します。 lkpolicy ツールで SIOS

Protection Suite サーバに最初にアクセスしようとしたときに、そのサーバに対する認証情報がまだ保存されていない場合、ユーザは認証情報を求められます。認証情報はユーザ名とパスワードの形式であり、さらに以下の条件があります。

1. クライアントには SIOS Protection Suite の管理者権限が必要です。したがって、そのユーザ名は、( *PAM* による) オペレーティングシステムの認証設定によって *lkadmin* グループに属する必要があります。必ずしも ***root*** で実行する ***必要はありません*** が、root ユーザはデフォルトで適切なグループに属しているため、root を使用することもできます。
2. 認証情報は認証情報ストアに保存されるため、ツールを使用してこのサーバにアクセスするたびに手動で認証情報を入力する必要はありません。

認証情報ストアと credstore ユーティリティによる管理の詳細については、SIOS Protection Suite の認証情報の設定 を参照してください。

lkpolicy によるセッションの例は以下のようになります。

```
[root@thor49 ~]# lkpolicy -l -d v6test4
Please enter your credentials for the system 'v6test4'.
Username: root
Password:
Confirm password:
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]# lkpolicy -l -d v6test4
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]#
```

## ポリシーのリスト表示

lkpolicy --list–policy–types

## 現在のポリシーの表示

lkpolicy --get–policies

lkpolicy --get–policies tag=*

lkpolicy --get–policies --verbose tag=mysql*  # all resources starting with mysql

lkpolicy --get–policies tag=mytagonly

## ポリシーの設定

lkpolicy --set–policy Failover --off

lkpolicy --set–policy Failover --on tag=myresource

lkpolicy --set–policy Failover --on tag=*

lkpolicy --set–policy LocalRecovery --off tag=myresource

lkpolicy --set–policy NotificationOnly --on

lkpolicy --set–policy TemporalRecovery --on recoverylimit=5 period=15

lkpolicy --set–policy TemporalRecovery --on --force recoverylimit=5 period=10

## ポリシーの削除

lkpolicy --remove–policy Failover tag=steve

注記: *NotificationOnly* はポリシーのエイリアスです。 *NotificationOnly* を有効にすることは、対応する *LocalRecovery* および *Failover* ポリシーを無効にすることと等価です。

# 5.4.2.12. 認証情報の設定

他のシステムと通信するための認証情報は、認証情報ストアを使用して管理されています。このストアは、必要に応じて /opt/LifeKeeper/bin/credstore ユーティリティで管理できます。このユーティリティを使用すると、サーバアクセスに必要な認証情報をサーバごとに設定、変更、削除することができます。

## 認証情報の追加または変更

認証情報の追加と変更は同じ方法で実行できます。代表的な例として、サーバー server.mydomain.com に対する資格情報を追加または変更する場合は次のようになります。

```
/opt/LifeKeeper/bin/credstore -k server.mydomain.com myuser
```

この例では、server.mydomain.com へのアクセスに使用するユーザ名として *myuser* を指定して しています。パスワードを入力 / 確認するプロンプト (*passwd* の様に) が表示されます。

> 注記: LifeKeeper サーバの 認証情報 を格納するために使用されるキー名は、lkpolicy などのコマンドで使用するホスト名と完全に一致する必要があります。コマンドで使用するホスト名が FQDN であれば、認証情報のキーも FQDN でなければなりません。コマンドで使用するホスト名がショートネームであれば、認証情報のキーもショートネームでなければなりません。

認証情報ストアに デフォルト キーを作成することもできます。サーバキーが存在しない場合に デフォルト の認証情報が認証に使用されます。 デフォルト キーを追加、変更するには以下のコマンドを実行してください。

```
/opt/LifeKeeper/bin/credstore -k default myuser
```

## ストア内の認証情報のリスト表示

現在格納されている認証情報をリスト表示するには、以下のコマンドを実行します。

```
/opt/LifeKeeper/bin/credstore -l
```

これにより、認証情報ストア内に格納されているキーが表示されます。この場合の「キー」は、認証情報を使用する対象のサーバを示しています (認証情報自体は秘密情報のため、このコマンドが表示するのは、実際の認証情報の内容ではなくキー名のみです)。

## サーバの認証情報の削除

特定のサーバに対する認証情報を削除するには、以下のコマンドを実行します。

```
/opt/LifeKeeper/bin/credstore -d -k myserver.mydomain.com
```

この例では、サーバ myserver.mydomain.com に対する認証情報がストアから削除されます。

# 追加情報

credstore ユーティリティの詳細については、以下のコマンドを実行してください。

```
/opt/LifeKeeper/bin/credstore --man
```

コマンドのマニュアルページがすべて表示されます。

# 5.4.2.13. Standby Node Health Check

## 機能の概要

LifeKeeper Core の既存の監視機能に Standby Node Health Check を組み合わせることにより、Standby ノードにおける CPU や Memory の使用率の監視や Out of service のリソースの健全性の監視を行い、Standbyノードの異常を検知できます。これにより、事前に問題を解消することができ、Activeノードで異常が発生した際のフェイルオーバーに失敗するリスクを軽減できます。この監視処理は、通常の LifeKeeper の監視実行間隔（ LKCHECKINTERVAL の設定値）と同じ間隔で実行されます。

Standby Node Health Check は、主に以下の二つの機能に分類されます。

### Node監視

ノード上ですべてのリソースが Out of service の場合、Standbyノードと判断して Node監視のスクリプトを呼び出します。Node監視のスクリプトが CPU や Memory の使用率の監視を行い、ノードが正常に切り替えが出来ない可能性が高い状態にあると判断すると、異常ありと判断され、予め設定されたメール通知機能、またはSNMPのイベント転送機能を通じて、検知した内容を通知します。詳細は「 Node監視 」を参照してください。

### OSUリソース監視

Out of service の各リソースについて、lkcheck が定期的にそのリソースタイプの OSUquickCheck スクリプトを呼び出します。OSUquickCheck スクリプトがリソースのクイック健全性チェックを実行し、リソースが正常に起動できない状態にあると判断すると、該当リソースのステータスを OSF に変更し、異常ありと判断され、予め設定されたメール通知機能、またはSNMPのイベント転送機能を通じて、検知した内容を通知します。詳細は「 OSUリソース監視 」を参照してください。

## インストールと設定

特別なインストールは必要ありません。
Standby Node Health Check 機能をセットアップする一般的なプロセスには以下の手順が含まれます。

1. イベントメール通知 や SNMPによるイベント転送 の設定を行います。
2. Standby Node Health Check の設定を行います。 （詳細は次項の「Standby Node Health Check の設定」を参照してください。）
3. 既に LifeKeeper が起動している場合は、設定を反映させるため lkcheck プロセスを再起動します。lkcheck プロセスの再起動には次のコマンドを実行してください。

```
killall lkcheck
```

上記の手順が完了すると、そのノードでは Standby Node Health Check 機能が動作するようになります。設定に応じてNode監視や Out of service の各リソースの監視が行われます。詳細は「 Node監視 」と「 OSUリソース監視 」のページを参照してください。

## Standby Node Health Check の設定

/etc/default/LifeKeeper 設定ファイルで SNHC の設定を有効に設定してください。詳細は「 Standby Node Health Check パラメーター一覧 」を参照してください。

# 5.4.2.13.1. Node監視

ノード上ですべてのリソースが Out of service の場合、Standbyノードと判断して Node監視のスクリプトを呼び出します。Node監視のスクリプトが CPU や Memory の使用率の監視を行い、ノードが正常に切り替えが出来ない可能性が高い状態にあると判断すると、異常ありと判断され、予め設定されたメール通知機能、またはSNMPのイベント転送機能を通じて、検知した内容を通知します。この監視処理は、通常のLifeKeeperの監視実行間隔（ LKCHECKINTERVAL の設定値）と同じ間隔で実行されます。

## 監視可能な項目

Node監視で監視できる項目は以下です。

| リソース名 | 監視内容 |
|---|---|
| CPU使用率 | /proc/stat ファイルから、監視実行時のCPU使用率を確認します。 |
| Memory使用率 | /proc/meminfo ファイルから、監視実行時のMemory使用率を確認します。 |

## Node監視の設定

/etc/default/LifeKeeper 設定ファイルで SNHC_CPUCHECK や SNHC_MEMCHECK の設定を有効に設定してください。また、他に以下の設定があります。詳細は「 Standby Node Health Check パラメータ一覧 」を参照してください。

- SNHC_CPUCHECK_THRESHOLD
- SNHC_CPUCHECK_TIME
- SNHC_MEMCHECK_THRESHOLD
- SNHC_MEMCHECK_TIME

# 5.4.2.13.2. OSU リソース監視

Out of service（ステータスが OSU） の各リソースについて、lkcheck が定期的にそのリソースタイプの OSUquickCheck スクリプトを呼び出します。OSUquickCheck スクリプトがリソースのクイック健全性チェックを実行し、リソースが正常に切り替えが出来ない可能性が高い状態にあると判断すると、該当リソースのステータスを OSF に変更し、異常ありと判断され、予め設定されたメール通知機能、または SNMP のイベント転送機能を通じて、検知した内容を通知します。この監視処理は、通常の LifeKeeper の監視実行間隔（LKCHECKINTERVAL の設定値）と同じ間隔で実行されます。

## 監視可能なリソース

現在は、以下の表に記載されているリソースが OSU リソース監視の対象となっています。

| リソース名 | 監視内容 |
|---|---|
| IPリソース | IPリソースの対象となる NIC リンクの状態を確認します（IP_NOLINKCHECK=1 の場合は確認しません）。<br>ping が可能かつ ping list が設定されている場合は、登録されているアドレスへの到達性を確認します。実 IP リソースは監視対象外です。 |
| Disk または DMMP リソース | 各リソースに対応したコマンドを用いて対象の Disk へのパスを確認します。 |
| NASリソース | NFSサーバーに対してNFS接続ができるかを確認します。NFS接続のタイムアウト値に関しては、 設定上の考慮事項 をご参照ください。 |

## OSUリソース監視の設定

/etc/default/LifeKeeper 設定ファイルで SNHC_IPCHECK や SNHC_DISKCHECK の設定を有効に設定してください。また、他に以下の設定があります。詳細は「 Standby Node Health Check パラメータ一覧 」を参照してください。

- SNHC_IPCHECK_SLEEPTIME

## 障害からの復帰

OSU リソース監視で異常が検知された場合、該当リソースのステータスが OSF（Out of service かつ障害あり）に変更されます。ステータスが変更されると、該当のリソースに対して OSU リソース監視は実行されなくなります。障害検知の通知の内容を確認して障害を取り除いた後、該当リソースのステータスを OSU に復帰させる必要があります。

以下のコマンドを実行してステータスを OSF から OSU に復帰できます。

```
/opt/LifeKeeper/lkadm/bin/retstate <resource tag>
```

# 5.4.3. LifeKeeper 管理

## 概要

LifeKeeper は操作時に管理を必要としません。LifeKeeper は、保護されたリソースを監視し、障害が発生した場合に指定されたリカバリアクションを実行するように、自動的に機能します。以下のケースでは LifeKeeper GUI を使用します。

- **リソースおよび階層の定義。** LifeKeeper は次のインターフェースオプションを提供します。
  - LifeKeeper GUI。
  - LifeKeeper コマンドラインインターフェース。
- **リソース監視。** LifeKeeper GUI は、リソースステータス情報および LifeKeeper ログへのアクセスを提供します。
- **手動での処理。** メンテナンスやその他の管理アクションのために、サーバまたは特定のリソースを停止することが必要になる場合があります。LifeKeeper GUI には、特定のリソースを稼動させたり停止させることができるメニュー機能が用意されています。アプリケーションが LifeKeeper の保護下に置かれると、これらの LifeKeeper のインターフェースを介してのみアプリケーションを起動および停止させることができます。LifeKeeper の起動および停止は、コマンドラインを介してのみ行われます。

LifeKeeper の管理、設定、およびメンテナンス操作を実行する詳細な手順については、GUI の作業 およびメンテナンス作業 を参照してください。

> 注記 : LifeKeeperが提供しているコマンド（実行可能なスクリプトやプログラム）を実行するには、スーパーユーザ権限が必要です。

su コマンドや sudo でスーパーユーザ権限を付与したユーザで、LifeKeeper のコマンドを実行することは可能ですが、SIOS Technology Corp では、root ユーザ以外で LifeKeeper のコマンドのテストはしていません。

---

エラーの検出および通知

N-way リカバリ

管理作業

サーバプロパティの編集

コミュニケーションパスの作成

コミュニケーションパスの削除

サーバのプロパティ - フェイルオーバ

リソース階層の作成

ファイルシステムリソース階層の作成

Generic Application リソース階層の作成

Raw デバイスリソース階層の作成

Quick Service Protection (QSP) Recovery Kit

リソースプロパティの編集

リソース優先順位の編集

リソース階層の拡張

ファイルシステムリソース階層の拡張

Generic Application リソース階層の拡張

Raw デバイスリソース階層の拡張

階層の拡張解除

リソース依存関係の作成

リソース依存関係の削除

すべてのサーバからの階層の削除

# 5.4.3.1. エラーの検出および通知

アプリケーション内の問題を検出して通知する機能は、最適な総合的耐障害性ソリューションを構築する上で非常に重要です。すべての個々のアプリケーションは、障害発生のメカニズムと形式によって異なるため、一般的なメカニズムを示すことはできません。ただし、一般的に、多くのアプリケーションの設定は、LifeKeeper に用意されている Core システムのエラー検出機能を利用することができます。リソースエラー回復シナリオ および サーバ障害回復シナリオ の各トピックでは、共通する 2 つの障害状況を使用して、LifeKeeper のコア機能を示しています。

LifeKeeper には、エラー、アラーム、およびリカバリ手順を起動するイベントを定義するための完全な環境も用意されています。このインターフェースは、通常、システムエラーログ用のパターンマッチ定義 (`/var/log/messages`)、またはカスタムビルドのアプリケーション固有の監視プロセスが必要になります。

# 5.4.3.2. N-Way リカバリ

N-Way リカバリを使用すると、異なる複数のリソースを、クラスタ内の異なるバックアップサーバにフェイルオーバすることができます。

保護されたリソース に戻る

# 5.4.3.3. 管理作業

サーバプロパティの編集

コミュニケーションパスの作成

コミュニケーションパスの削除

サーバのプロパティ - フェイルオーバ

リソース階層の作成

リソースプロパティの編集

リソース優先順位の編集

リソース階層の拡張

階層の拡張解除

リソース依存関係の作成

リソース依存関係の削除

すべてのサーバからの階層の削除

# 5.4.3.3.1. サーバプロパティの編集

1. サーバプロパティを編集するには、サーバプロパティを表示する 場合と同様に、[Server Properties] ダイアログを表示してください。

2. 該当のサーバに適切な権限でログインした場合は、次の項目が編集可能になります。
   - シャットダウン方法
   - フェイルオーバ確認

3. 変更が加えられると、 **[Apply]** ボタンが有効になります。このボタンをクリックすると変更が適用されます。ウィンドウは閉じられません。

4. 完了したら、 **[OK]** をクリックし、変更内容を保存してウィンドウを閉じるか、または **[Cancel]** をクリックして、変更内容を適用せずにウィンドウを閉じます。

# 5.4.3.3.2. コミュニケーションパスの作成

サーバ間の LifeKeeper コミュニケーションパスを設定する前に、ハードウェアおよびソフトウェアのセットアップを検証します。詳細については、SPS for Linuxリリースノート を参照してください。

サーバのペア間にコミュニケーションパスを作成するには、両方のサーバに個別にパスを定義する必要があります。LifeKeeper では、サーバのペア間に TCP (TCP/IP) と TTY の両方のコミュニケーションパスを作成することができます。TTY パスは、所定のペア間に 1 つだけ作成できます。これに対し、TCP コミュニケーションパスは、パスの終点となるローカルおよびリモートのアドレスを指定することで、サーバのペア間に複数作成することができます。所定のリモートサーバへの TCP パスを使用する順序を LifeKeeper に設定するには、優先値を使用します。

**重要:** 単一のコミュニケーションパスを使用した場合、互いに通信するクラスタ内のサーバの機能に支障をきたす可能性があります。単一のコミュニケーションパスを使用しているときに、そのコミュニケーションパスで障害が発生した場合、複数のサーバ上で同時に LifeKeeper の階層が使用可能になることがあります。これは、「偽のフェイルオーバ」と呼ばれます。また、TCP コミュニケーションパス上のネットワークトラフィックが大きくなると、偽のフェイルオーバや LifeKeeper の初期化の問題など、予期せぬ動作が生じる可能性があります。

1. 開始するには、次の 4 つの方法があります。
   - サーバアイコンを右クリックし、サーバコンテキストメニュー が表示されたら **[Create Comm Path]** をクリックしてください。
   - グローバルツールバー で、 **[Create Comm Path]** ボタンをクリックしてください。
   - サーバコンテキストツールバー で (表示されている場合)、 **[Create Comm Path]** ボタンををクリックしてください。
   - [Edit] メニュー で、 **[Server]** 、 **[Create Comm Path]** の順に選択してください。

2. **[Create Comm Path]** というタイトルのダイアログが表示されます。表示されるオプションのそれぞれについて、 **[Help]** をクリックすると、選択した項目の説明が表示されます。

3. リストボックスから **[Local Server]** を選択し、 **[Next]** をクリックしてください。

4. リストボックスで 1 つ以上の **[Remote Servers]** を選択してください。リストボックスにリモートサーバが表示されていない (つまり、クラスタにまだ接続されていない) 場合は、 **[Add]** を使用して入力してください。ローカルとリモートの両方のサーバに対するネットワークアドレスが解決可能であることを確認する必要があります (たとえば、DNS で解決するか、 */etc/hosts* ファイルに追加します)。 **[Next]** をクリックしてください。

5. **[Device Type]** に対して **[TCP]** または **[TTY]** を選択して、 **[Next]** をクリックしてください。

6. **[Device Type]** が [TCP] に設定されている場合は、1 つ以上の **[Local IP Addresses]** を選択してください。 **[Device Type]** が [TTY] に設定されている場合は、 **[Local TTY Device]** を選択してください。 **[Next]** をクリックしてください。

7. **[Device Type]** が **[TCP]** に設定されている場合は、 **[Remote IP Address]** を選択してください。

**[Device Type]** が **[TTY]** に設定されている場合は、 **[Remote TTY Device]** を選択してください。 **[Next]** をクリックしてください。

8. **[Device Type]** が **[TCP]** に設定されている場合は、このコミュニケーションパスに対して **[Priority]** を入力または選択してください。 **[Device Type]** が **[TTY]** に設定されいる場合は、このコミュニケーションパスに対して **[Baud Rate]** を入力または選択してください。 **[Next]** をクリックしてください。

9. **[Create]** をクリックしてください。ネットワーク接続が正常に作成されたことを示すメッセージが表示されます。 **[Next]** をクリックしてください。

10. 複数のローカル IP アドレスまたは複数のリモートサーバを選択したときに、 **[Device Type]** が [TCP] に設定されている場合は、手順 6 に戻り、次のコミュニケーションパスの設定を続けます。複数のリモートサーバを選択したときに、 **[Device Type]** が [TTY] に設定されている場合は、手順 5 に戻り、次のコミュニケーションパスの設定を続けます。

11. 終了メッセージが表示されたら、 **[Done]** をクリックしてください。

[Server Properties] ダイアログ を表示するか、またはコマンド **lcdstatus -q** を入力することで、コミュニケーションパスを確認することができます。 **lcdstatus** の使用方法については、LCD マニュアルページを参照してください。 **[ALIVE]** ステータスが表示されます。

さらに、GUI の右ペインのサーバアイコンをチェックしてください。これが、作成済みの 1 つ目のコミュニケーションパスである場合は、1 つのコミュニケーションパスが **[ALIVE]** であるが、冗長コミュニケーションパスがないことを示す黄色のハートビートがサーバアイコンに表示されます。

**[ALIVE]** のコミュニケーションパスが 2 つ以上ある場合は、緑色のハートビートがサーバアイコンに表示されます。

> **重要:** IPv6 アドレスを使用してコミュニケーションパスを作成する場合は、自動設定/ステートレスアドレスではなく、静的に割り当てられたアドレスを使用してください。自動設定/ステートレスアドレスは、時間がたつと変更され、コミュニケーションパスが使用できなくなる可能性があります。

数分たってもコミュニケーションパスが使用可能にならない場合は、ペアのサーバのコンピュータ名が正しいことを確認してください。TTY コミュニケーションパスを使用している場合は、2 つのサーバ間のケーブル接続が正しく、緩んでいないことを確認してください。必要に応じて `portio(1M)` コマンドを使用して、TTY 接続の動作を確認してください。

# 5.4.3.3.3. コミュニケーションパスの削除

1. 開始するには、次の 4 つの方法があります。
   - サーバアイコンを右クリックして、サーバコンテキストメニュー が表示されたら **[Delete Comm Path]** をクリックしてください。
   - グローバルツールバー で、 **[Delete Comm Path]** ボタンをクリックしてください。
   - サーバコンテキストツールバー で (表示された場合)、 **[Delete Comm Path]** ボタンをクリックしてください。
   - [Edit] メニュー で、[Server]、 **[Delete Comm Path]** の順に選択してください。

2. [Delete Comm Path] というタイトルのダイアログが表示されます。表示されるオプションのそれぞれについて、 **[Help]** をクリックすると、選択した項目の説明が表示されます。

3. リストから **[Local Server]** を選択し、 **[Next]** をクリックしてください。このダイアログが表示されるのは、グローバルツールバー の **[Delete Comm Path]** ボタンを使用するか、[Edit] メニュー の [ **Server** ] を選択して、削除を選択した場合のみです。

4. 削除するコミュニケーションパスを選択して、 **[Next]** をクリックしてください。

5. **[Delete Comm Path(s)]** をクリックしてください。出力パネルが有効の場合、ダイアログが閉じ、コミュニケーションパスを削除するコマンドの結果が出力パネル に表示されます。有効でない場合は、ダイアログが表示されたままこれらの結果が表示されます。 すべての結果が表示されたら、 **[Done]** をクリックして終了してください。ネットワーク接続が正常に削除されたことを示すメッセージが表示されます。

6. **[Done]** をクリックして、ダイアログを閉じ、GUI ステータス表示に戻ってください。

# 5.4.3.3.4. サーバのプロパティ - フェイルオーバ

プライマリサーバがローカルリカバリを試行して失敗した場合、または完全に機能停止した場合、ほとんどのサーバ管理者は、LifeKeeper で保護されたリソースをバックアップサーバに復元することが必要になります。これがデフォルトの LifeKeeper の動作になります。ただし管理者によっては、保護されたリソースをリカバリサイトで自動的に稼動するようにしたくない場合もあります。例えば、ディザスタリカバリ状況においてサーバ間のネットワーク接続の信頼性が低くなるような WAN 環境に LifeKeeper がインストールされている場合です。

デフォルトでは、保護されたすべてのリソースに対して自動フェイルオーバが有効になっています。保護されたリソースに対する自動フェイルオーバを無効にしたり、バックアップサーバへの自動フェイルオーバを行わないようにするには、[Server Properties] の **[General]** タブにある **[Failover]** セクションを使用して、以下の通り設定してください。

クラスタ内の各サーバで次のようにします。

1. サーバのプロパティを表示する 場合と同様に、 **[Server Properties]** ダイアログを表示してください。

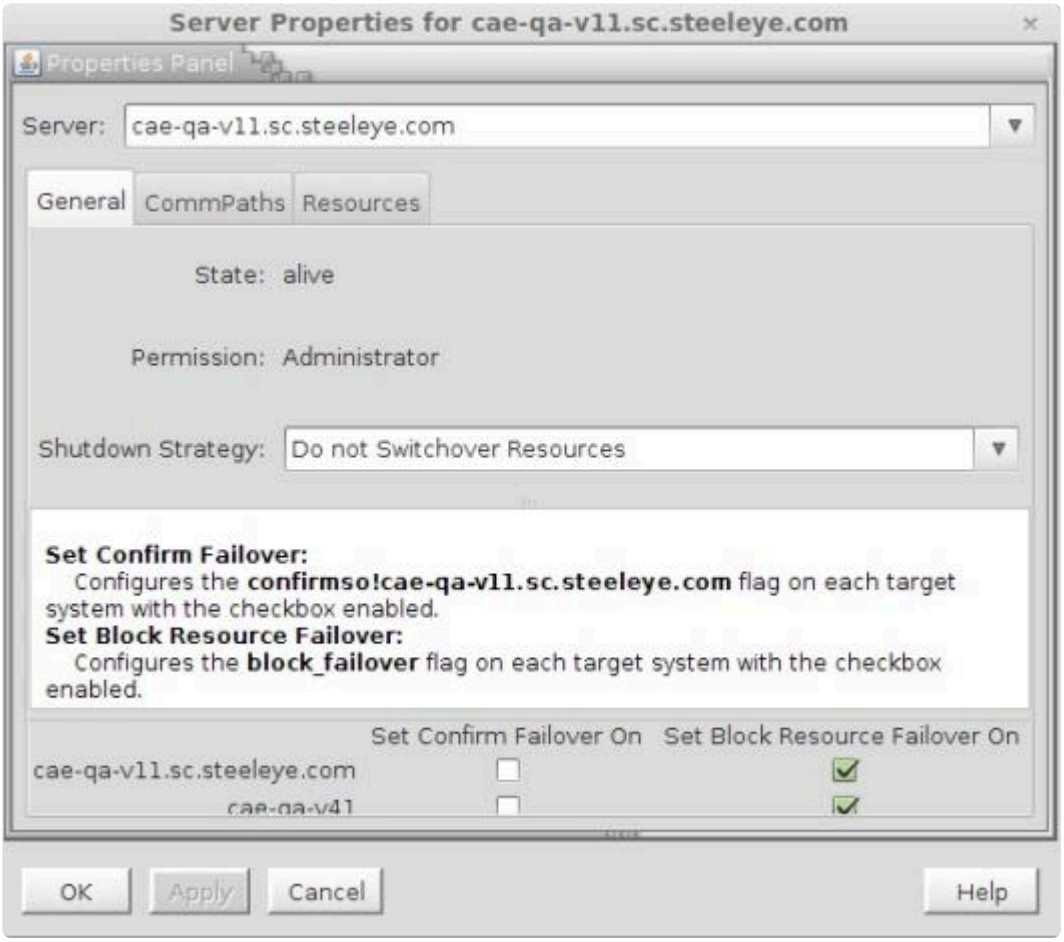
2. **[General]** タブを選択してください。[Server Properties] ダイアログの **[Failover]** セクションで、システムおよびリソースのフェイルオーバ機能を無効にするサーバをチェックしてください。デフォルトでは、LifeKeeper のすべてのフェイルオーバ機能が有効になっています。

   **[Set Confirm Failover On]** 列で、ローカルサーバの完全な機能停止に対応するバックアップサーバとしての資格を喪失させるサーバを選択してください。

   **[Set Block Resource Failover On]** 列で、このローカルサーバ上でリソース階層に障害が発生した場合に対応するバックアップサーバとしての資格を喪失させるサーバを選択してください。最初にシステムのフェイルオーバ機能を無効にしなければ、リソースのフェイルオーバを無効にすることはできません。

   選択内容を適用するには、 **[Apply]** ボタンを押してください。

設定の詳細につきましては、 **[Confirm Failover] と [Block Resource Failover] の設定** のページを参照してください。


Server Properties for cae-qa-v11.sc.steeleye.com
Properties Panel
Server: cae-qa-v11.sc.steeleye.com
General
CommPaths
Resources
State: alive
Permission: Administrator
Shutdown Strategy: Do not Switchover Resources
Set Confirm Failover:
Configures the confirmso!cae-qa-v11.sc.steeleye.com flag on each target system with the checkbox enabled.
Set Block Resource Failover:
Configures the block_failover flag on each target system with the checkbox enabled.
Set Confirm Failover On
Set Block Resource Failover On
cae-qa-v11.sc.steeleye.com
cae-qa-v41
OK
Apply
Cancel
Help

# 5.4.3.3.5. リソース階層の作成

1. リソース階層の作成を開始するには、次の 4 つの方法があります。

> 各種 ARK のライセンスを持たない Quorum ホストでリソース階層を作成することはできません。

- サーバアイコン を右クリックして、サーバコンテキストメニュー が表示されたら、 **[Create Resource Hierarchy]** をクリックしてください。
- グローバルツールバー で、 **[Create Resource Hierarchy]** ボタンをクリックしてください。
- サーバコンテキストツールバー で (表示された場合)、 **[Create Resource Hierarchy]** ボタンをクリックしてください。
- [Edit]メニュー で、 **[Server]** を選択して、 **[Create Resource Hierarchy]** をクリックしてください。

2. [Create Resource Wizard] というタイトルのダイアログが表示され、クラスタ内にインストールされているすべての認識されたリカバリキットのリストが示されます。 アプリケーションを保護するためにリソース階層を構築する Recovery Kit を選択して、[Next] をクリックしてください。
3. [Switchback Type] を選択して、[Next] をクリックしてください。
4. [Server] を選択して、[Next] をクリックしてください。注記: サーバコンテキストメニューから開始した場合、クリックしたサーバアイコンから自動的にサーバが決定されるので、この手順はスキップされます。
5. 続いて表示されるダイアログを使用して、作成しているリソース階層の種類に必要なデータを入力してください。

## LifeKeeper アプリケーションリソース階層

LifeKeeper をリカバリキット無しでインストールした場合、デフォルトでは [Select Recovery Kit] リストに、ファイルシステムまたは Generic Application 用のオプションが含まれています。Generic Application のオプションは、関連付けられたリカバリキットがないアプリケーションに使用できます。

Raw I/O Recovery Kit または IP Recovery Kit (これらは両方とも、別個にパッケージ化され、LifeKeeper Core メディアに含まれている Core Recovery Kit です) をインストールした場合、[Select Recovery Kit] リストはこれらの Recovery Kit に対する追加のオプションを提供します。

これらの利用可能なオプションについては、以下のトピックを参照してください。

- ファイルシステムリソース階層の作成
- Generic Application リソース階層の作成

- Raw デバイスリソース階層の作成

IP Recovery Kit については、IP Recovery Kit テクニカルドキュメンテーション を参照してください。

# Recovery Kit のオプション

インストールしたオプションの各リカバリキットは、[Select Recovery Kit] リストにエントリを追加します。たとえば、Oracle、Apache、および NFS の Recovery Kit がリストに表示されます。必要なリソース階層を作成する手順については、各リカバリキットに付属の管理ガイドを参照してください。

> 注記: ファイルシステムまたは論理ボリューム上に構築されるその他のアプリケーションリソース階層を作成する場合、最初にLogical Volume Manager (LVM) Recovery Kit をインストールする必要があります。

# 5.4.3.3.5.1. ファイルシステムリソース階層の作成

このオプションは、ファイルシステムのみを保護する場合 (たとえば、保護を必要とする共有ファイルがある場合) に使用します。

1. ファイルシステムリソース階層の作成を開始するには、次の 4 つの方法があります。

   • サーバアイコンを右クリックして、サーバコンテキストメニュー が表示されたら、 **[Create Resource Hierarchy]** をクリックしてください。

   • グローバルツールバー で 、 **[Create Resource Hierarchy]** ボタンをクリックしてください。

   • サーバコンテキストツールバー で (表示された場合)、 **[Create Resource Hierarchy]** ボタンをクリックしてください。

   • [Edit] メニュー で、 **[Server]** を選択して、 **[Create Resource Hierarchy]** をクリックしてください。

2. *[Create Resource Wizard] というタイトルのダイアログが表示され、* ***[Recovery Kit]*** *リストが示されます。 [File System Resource] を選択して、* ***[Next]*** *をクリックしてください。*

3. **[Switchback Type]** を選択して、[Next] をクリックしてください。

4. **[Server]** を選択して、 **[Next]** をクリックしてください。注意:サーバコンテキストメニューから開始した場合、クリックしたサーバアイコンから自動的にサーバが決定されるので、この手順はスキップされます。

5. *[Create gen/filesys Resource] ダイアログが表示されます。*ファイルシステムリソース階層に対して **[Mount Point]** を選択し、 **[Next]** をクリックしてください。選択したマウントポイントが、クラスタ内の別のサーバと共有されていることを確認するには、各ストレージキットをチェックして、マウントされたデバイスを共有として認識しているかどうかを確認します。ストレージキットがマウントされたデバイスを認識していない場合は、次のエラーダイアログが表示されます。

   < ***file system*** > is not a shared file system

   **[OK]** を選択すると、 *[Create gen/filsys Resource]* ダイアログに戻ります。

   注記:

   • マウントポイントを選択リストに表示するには、そのマウントポイントがマウントされている必要があります。マウントポイントに対するエントリが `/etc/fstab` ファイルに存在する場合、階層の作成および拡張時にこのエントリが削除されます。NAS Recovery Kit を使用する前に (特にマウント設定が複雑な場合)、`/etc/fstab` のバックアップを作成することをお勧めします。`/etc/default/LifeKeeper` で調整可能な `REPLACEFSTAB=true|TRUE` を設定することにより、`/etc/fstab` のエントリが削除されても、元に戻すように指定することができま

す。

- これらのリソースの多く (SIOS DataKeeper、LVM、Device Mapper Multipath など) は、ファイルシステムリソースを作成するために、クラスタ内の各サーバで LifeKeeper リカバリキットを必要とします。 これらのキットが適切にインストールされていない場合、クラスタ内で共有されるファイルシステムが表示されません。

6. ファイルシステムリソース階層に対するデフォルトの **[Root Tag]** が作成されます。(これは、ステータス表示でこのリソースに使用されるラベルです)。このルートタグを選択するか、独自のルートタグを作成して、 **[Next]** をクリックしてください。

7. **[Create Instance]** をクリックしてください。インスタンス作成のステータスを示すメッセージが、ウィンドウに表示されます。

8. **[Next]** をクリックしてください。ファイルシステム階層が正常に作成されたというメッセージが、ウィンドウに表示されます。

9. この時点で、ファイルシステムリソース階層の拡張 に移動するには **[Continue]** をクリックし、GUI に戻るには **[Cancel]** をクリックします。 **[Cancel]** をクリックすると、階層が 1 つのサーバにしか存在しないという警告メッセージが表示され、この時点で保護されなくなります。

# 5.4.3.3.5.2. Generic Application リソース階層の作成

このオプションは、関連付けられたリカバリキットがないユーザ定義のアプリケーションを保護する場合に使用します。下記のユーザ指定スクリプトに対するテンプレートは、`$LKROOT/lkadm/subsys/gen/app/templates` に用意されています。これらのテンプレートは、保護するアプリケーション用にカスタマイズしてテストする前に、別のディレクトリにコピーしてください。

> **注記:** ファイルシステム、ディスクパーティション、IP アドレスなどの他のリソースに依存するアプリケーションについては、これらの各リソースを個別に作成し、Create Dependency を使用して適切な依存関係を作成してください。

1. Generic Application リソース階層の作成を開始するには、次の 4 つの方法があります。
   - サーバアイコンを右クリックして、サーバコンテキストメニュー が表示されたら、 **[Create Resource Hierarchy]** をクリックしてください。
   - グローバルツールバー で、[Create Resource Hierarchy] ボタンをクリックしてください。
   - サーバコンテキストツールバー で (表示された場合)、 **[Create Resource Hierarchy]** ボタンをクリックしてください。
   - [Edit] メニュー で、 **[Server]** を選択して、 **[Create Resource Hierarchy]** をクリックしてください。

2. [Create Resource Wizard] というタイトルのダイアログが表示され、 **[Recovery Kit]** リストが示されます。Generic Application を選択して、 **[Next]** をクリックしてください。

3. **[Switchback Type]** を選択して、 **[Next]** をクリックしてください。

4. **[Server]** を選択して、 **[Next]** をクリックしてください。 **注記:** サーバコンテキストメニューから開始した場合、クリックしたサーバアイコンから自動的にサーバが決定されるので、この手順はスキップされます。

5. 次のダイアログで、アプリケーションに対する **[Restore Script]** へのパスを入力し、 **[Next]** をクリックしてください。これは、アプリケーションを起動するコマンドです。テンプレートディレクトリに、テンプレート起動スクリプト restore.template が用意されています。この restore スクリプトが、既に起動されているアプリケーションに影響を与えてはなりません。

6. アプリケーションに対する **[Remove Script]** へのパスを入力し、 **[Next]** をクリックしてください。これは、アプリケーションを停止するコマンドです。テンプレートディレクトリに、テンプレート停止スクリプト remove.template が用意されています。

7. アプリケーションに対する **[quickCheck Script]** へのパスを入力し、 **[Next]** をクリックしてください。これは、アプリケーションを監視するコマンドです。テンプレートディレクトリに、テンプレート監視スクリプト quickCheck.template が用意されています。

8. アプリケーションに対する **[Local Recovery Script]** へのパスを入力し、 **[Next]** をクリックしてください。これは、ローカルサーバ上の障害が発生したアプリケーションの復元を試みるコマンドで

す。テンプレートディレクトリに、テンプレート回復スクリプト recover.template が用意されています。

9. **[Application Information]** を入力し、 **[Next]** をクリックしてください。これは、起動、停止、回復、監視の各スクリプトで必要になる可能性のあるアプリケーションに関するオプションの情報です。

10. **[Bring Resource In Service]** に対して [Yes] または [No] を選択し、 **[Next]** をクリックしてください。[No] を選択すると、リソース状態が作成後に OSU に設定されます。[Yes] を選択すると、あらかじめ用意された restore スクリプトが実行されます。 ファイルシステム、ディスクパーティション、IP アドレスなどの他のリソースに依存するアプリケーションについては、適切な依存リソースをまだ作成していない場合には、[No] を選択してください。

11. **[Root Tag]** を入力してください。これは、リソースインスタンスに対する一意の名前です。(これは、ステータス表示でこのリソースに対して表示されるラベルです。)

12. **[Create Instance]** をクリックして、作成プロセスを起動してください。インスタンス作成のステータスを示すメッセージが、ウィンドウに表示されます。

13. **[Next]** をクリックしてください。階層が正常に作成されたというメッセージが、ウィンドウに表示されます。

14. この時点で、Generic Application リソース階層の拡張 に移動するには **[Continue]** をクリックし、GUI に戻るには **[Cancel]** をクリックします。[Cancel] をクリックすると、階層が 1 つのサーバにしか存在しないという警告が表示され、この時点で保護されなくなります。

**注記:** リソース階層作成時に指定したrestore、remove、quickCheckなどの各種スクリプトは、LKROOT/subsys/gen/resource/app/以下の各ディレクトリに配置されています。

- **restore - actions/!restore/<** タグ名 **>**
- **remove - actions/!remove/<** タグ名 **>**
- **quickCheck - actions/!quickCheck/<** タグ名 **>**
- **recover - recovery/!recover/<** タグ名 **>**

# 5.4.3.3.5.3. Raw デバイスリソース階層の作成

このオプションは、Raw デバイスリソースを保護する場合に使用します。たとえば、既存のデータベース階層に追加する必要がある Raw デバイスに対して追加のテーブルスペースを作成する場合、このオプションを使用して Raw デバイスリソースを作成します。

**注記:** LifeKeeper では、ディスク論理ユニット (または LUN) レベルのディスクパーティションリソースを、一度に 1 つのクラスタの 1 つのシステムにロックします。

1. Raw デバイスリソース階層の作成を開始するには、次の 4 つの方法があります。

- º サーバアイコンを右クリックして、

サーバコンテキストメニュー が表示されたら、 **[Create Resource Hierarchy]** をクリックしてください。
- º

グローバルツールバー で、 **[Create Resource Hierarchy]** ボタンをクリックしてください。
- º

サーバコンテキストツールバー で (表示された場合)、 **[Create Resource Hierarchy]** ボタンをクリックしてください。
- º

[Edit] メニュー で、 **[Server]** を選択して、 **[Create Resource Hierarchy]** をクリックしてください。

2. [Create Resource Wizard] というタイトルのダイアログが表示され、 **[Recovery Kit]** リストが示されます。Raw デバイスを選択して、 **[Next]** をクリックしてください。

3. **[Switchback Type]** を選択して、 **[Next]** をクリックしてください。

4. **[Server]** を選択して、 **[Next]** をクリックしてください。

   **注記:** サーバコンテキストメニューから開始した場合、クリックしたサーバアイコンから自動的にサーバが決定されるので、この手順はスキップされます。

5. このリソースが配置される共有ストレージデバイスで **[Raw Partition]** を選択して、 **[Next]** をクリックしてください。

6. **[Root Tag]** を入力してください。これは、リソースインスタンスに対する一意の名前です。(これは、ステータス表示でこのリソースに対して表示されるラベルです。)

7. **[Create Instance]** をクリックして、作成プロセスを起動してください。作成中に何が発生したかを示すテキストが、[Creating scsi/raw resource] というタイトルのウィンドウに表示されます。

8. **[Next]** をクリックしてください。階層が正常に作成されたというメッセージが、ウィンドウに表示されます。

9. この時点で、Raw リソース階層の拡張 に移動するには **[Continue]** をクリックし、GUI に戻るには

**[Cancel]** をクリックします。[Cancel] をクリックすると、階層が 1 つのサーバにしか存在しないということを警告するメッセージが表示され、この時点で保護されなくなります。

# 5.4.3.3.5.4. Quick Service Protection (QSP) Recovery Kit

## はじめに

QSP Recovery Kit は、OS のサービスを簡易的に保護する仕組みを提供します。OS の service コマンドまたは systemctl コマンドで起動 (start) ・停止 (stop) ができるサービスであれば、容易にリソースが作成できます。Generic Application を使っても同じことが実現できますが、QSP を使えばコードの開発が不要です。また、依存関係を作成することで、他のリソースで保護しているアプリケーションと連携してサービスを起動･停止させることができます。

ただし、QSP の quickCheck では簡易なチェックしか行っていません（service コマンドの status を利用しています）。実際にそのサービスが提供できているか、プロセスが動作しているかは保証されません。複雑な起動や停止処理、堅牢なチェックが必要な場合は、Generic Application の利用を検討してください。

## 要件

QSPで保護するサービスは、以下の要件を満たしている必要があります。

- OS の service コマンドまたは systemctl コマンドで保護したいサービスの起動 (start) と停止 (stop) ができること。また、コマンド成功時は0を返すこと。
- quickCheck 機能を有効にする場合は、OS の service コマンドまたは systemctl コマンドで保護したいサービスの状態取得 (status) ができること。また、コマンド成功時は 0 を返すこと。
- 保護したいサービス名が 256 文字以内で、英数記号以外の文字が含まれていないこと。

> 補足 : service コマンドの互換性を利用して、実装上は systemd 環境でも service コマンドを実行する場合があります。

リソース作成時は、プライマリサーバで保護したいサービスを起動しておく必要があります。

なお、専用の Recovery Kit があるサービスやアプリケーションではないサービス等については（以下、これらのサービスを「QSP 保護対象外サービス」と呼ぶ）、QSP 保護対象外となり QSP Recovery Kit では保護できません。

## QSP リソース階層の作成

このオプションは、OS のサービスを保護する場合に使用します。

1. QSPリソース階層の作成を開始するには、次の 4 つの方法があります。

- サーバアイコンを右クリックして、サーバコンテキストメニュー が表示されたら、[Create Resource Hierarchy] をクリックしてください。

- グローバルツールバー で、[Create Resource Hierarchy] ボタンをクリックしてください。
- サーバコンテキストツールバー で (表示された場合)、[Create Resource Hierarchy] ボタンをクリックしてください。
- [Edit] メニュー で、[Server] を選択して、[Create Resource Hierarchy] をクリックしてください。

2. [Create Resource Wizard] というタイトルのダイアログが表示され、[Recovery Kit] リストが示されます。Quick Service Protection を選択して、[Next] をクリックしてください。

3. [Switchback Type] を選択して、[Next] をクリックしてください。

4. [Server] を選択して、[Next] をクリックしてください。

   注記： サーバコンテキストメニューから開始した場合、クリックしたサーバアイコンから自動的にサーバが決定されるので、この手順はスキップされます。

5. 次のダイアログで、[Service Name] にサービスの一覧が示されます。保護したいサービスを選択して [Next] をクリックしてください。

   注記： サービスが起動していない場合は、リストに表示されないことがあります。その場合は、[Cancel] をクリックして処理を中断し、サービス起動後に最初からやり直してください。また、QSP 保護対象外サービスは表示されません。

6. quickCheck 機能を有効にする場合は [enable] を選択、無効にする場合は [disable] を選択し、[Next] をクリックしてください。この設定は後から変更することも可能です。

   注記： service コマンドで状態取得 (status) がサポートされていないサービスでは、QSP の監視機能を使用できませんので必ず “disable” を選択してください。

7. [Resource Tag] を入力してください。これは、リソースインスタンスに対する一意の名前です。（これは、ステータス表示でこのリソースに対して表示されるラベルです。）

8. [Create Instance] をクリックして、作成プロセスを起動してください。インスタンス作成のステータスを示すメッセージがウィンドウに表示されます。

9. [Next] をクリックしてください。続けて拡張を行うダイアログが表示されます。GUI に戻るには [Cancel] をクリックしてください。[Cancel] をクリックすると、階層が１つのサーバにしか存在しないという警告が表示され、この時点では LifeKeeper による保護は行われません。

# QSP リソース階層の拡張

この操作は、リソース階層の拡張 に関するセクションで説明されているように、QSP リソース階層の作成を終了した後に自動的に開始したり、既存の QSP リソースから開始することができます。それが済んだら、次に以下の手順を完了します。

1. LifeKeeper が提供する [Resource Tag] を選択するか、またはターゲットサーバ上のリソース階層に対する独自のタグを入力してください。

2. [Extend] をクリックして、拡張プロセスを起動してください。ダイアログに拡張操作のステータス

が表示され、階層が正常に拡張されたことを示すメッセージが表示されて終了します。同じリソース階層を別のサーバに拡張する場合は、[Next Server] をクリックしてください。その場合は、拡張の操作が繰り返されます。または、[Finish] をクリックして、この操作を完了してください。

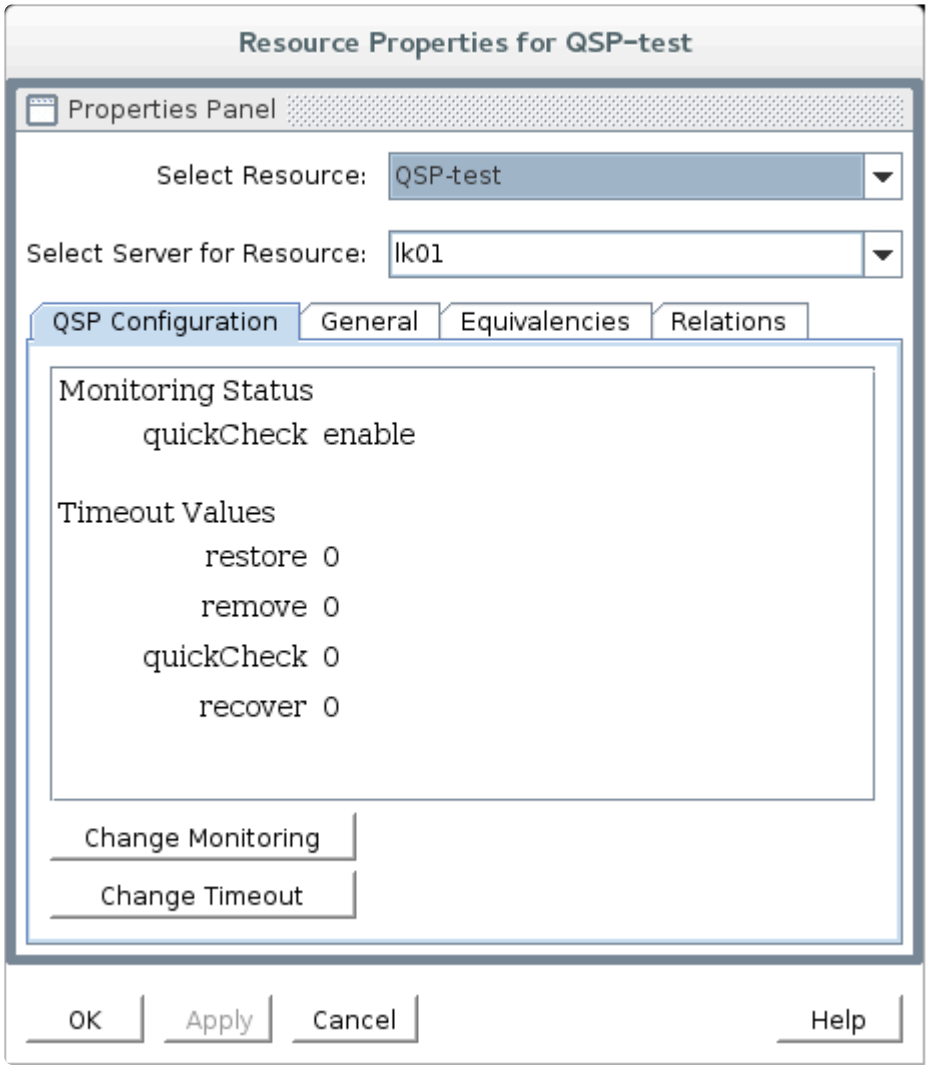
3. 拡張された階層が確認されると、確認情報がダイアログに表示されます。これが終了すると、[Done] ボタンが有効になります。[Done] をクリックして終了してください。

## QSPリソースのコンフィグレーション

QSPリソース固有の設定として以下の項目があります。

| 設定項目 | | デフォルト値 | 意味 |
|---|---|---|---|
| 監視機能 | quickCheck | リソース作成時に指定 | 監視機能の有効･無効の設定。 |
| タイムアウト | restore | 0 | restore のタイムアウト（単位：秒）を指定する。0が指定された場合はタイムアウトしない。 |
| | remove | 0 | remove のタイムアウト（単位：秒）を指定する。0が指定された場合はタイムアウトしない。 |
| | quickCheck | 0 | quickCheck のタイムアウト（単位：秒）を指定する。0が指定された場合はタイムアウトしない。 |
| | recover | 0 | recover のタイムアウト（単位：秒）を指定する。0が指定された場合はタイムアウトしない。 |

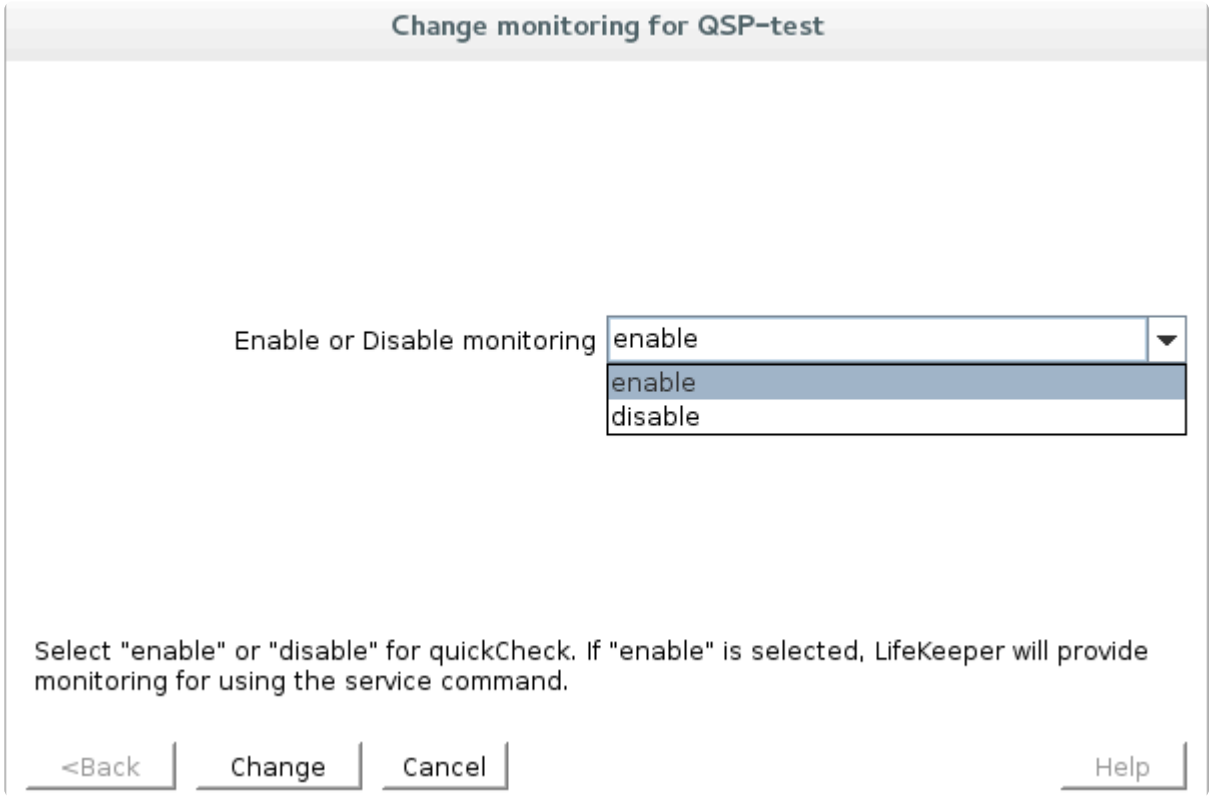
設定値の確認･変更はリソースのプロパティを表示 させ [QSP Configuration] タブから行うことができます。また、設定はノード毎に行う必要があります。 なお、quickCheck 機能を無効にした場合、タイムアウトの quickCheck と recover は表示されません。また変更もできません。




## 監視機能の変更方法

1. リソースプロパティの [QSP Configuration] タブを表示して、[Change quickCheck] をクリックしてください。
2. quickCheck を有効にする場合は [enable] を選択、無効にする場合は [disable] を選択してください。
3. [Change] をクリックすると変更処理が起動して、変更処理のメッセージが表示されます。
4. [Done] をクリックして終了してください。

   注記： 全ノードの設定を変更することはできません。変更した場合は、他のノードでも変更してください。



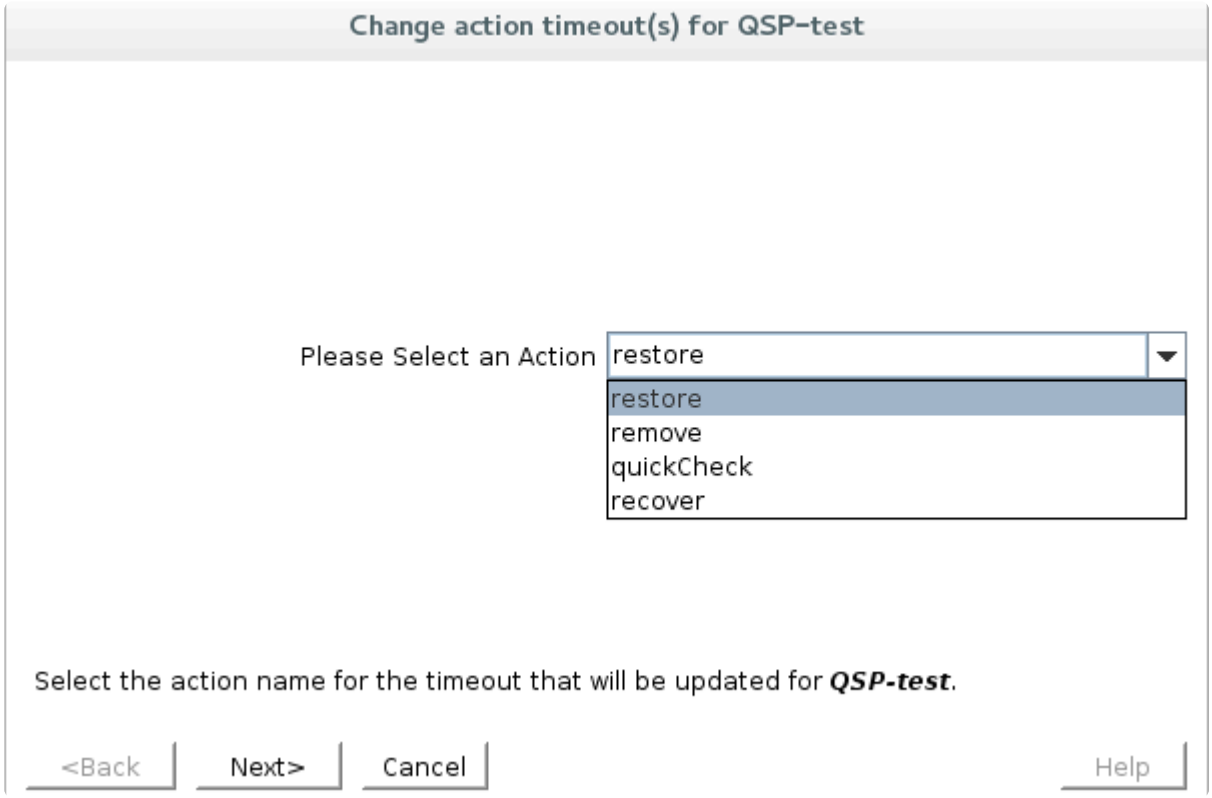

## タイムアウト値の変更方法

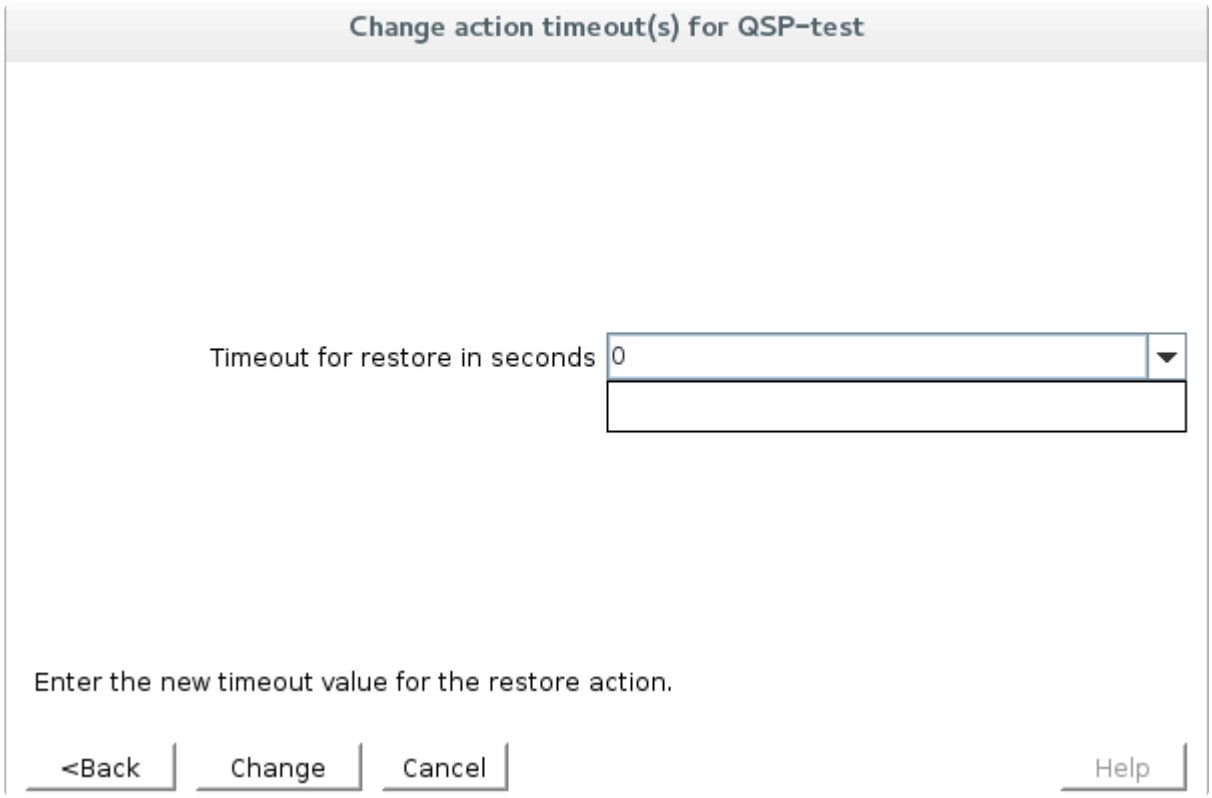
1. リソースプロパティの [QSP Configuration] タブを表示して、[Change Timeout] をクリックしてください。
2. 変更したいタイムアウトのアクションを選択して、[Next] をクリックしてください。

   注記： 監視機能が無効の場合は、[quickCheck] と [recover] は選択肢に表示されません。
3. 秒単位でタイムアウト値を入力してください。

   注記： 10進数で入力してください。数字以外の文字は無効です。
4. [Change] をクリックすると変更処理が起動して、変更処理のメッセージが表示されます。
5. [Done] をクリックして終了してください。

   注記： 全ノードの設定を変更することはできません。変更した場合は、他のノードでも変更してください。


Change action timeout(s) for QSP-test
Please Select an Action
restore
restore
remove
quickCheck
recover
Select the action name for the timeout that will be updated for QSP-test.
<Back
Next>
Cancel
Help


Change action timeout(s) for QSP-test
Timeout for restore in seconds
0
Enter the new timeout value for the restore action.
<Back
Change
Cancel
Help

# 5.4.3.3.6. リソースのプロパティの編集

1. リソースのプロパティを編集するには、リソースのプロパティを表示する 場合と同様に、[Resource Properties] ダイアログを表示してください。

2. 該当のサーバに適切な権限でログインした場合は、次の項目が編集可能になります。

- º スイッチバック
- º リソース設定 (特殊な設定を持つリソースの場合のみ)
- º

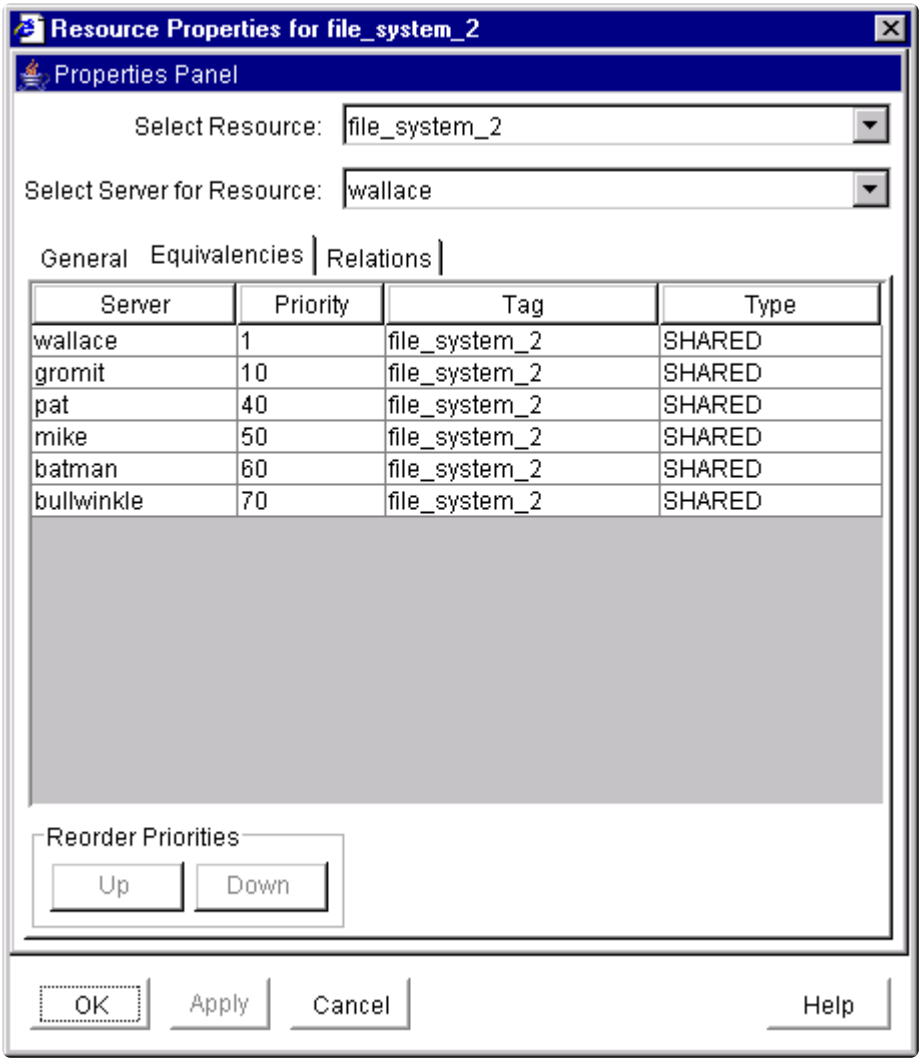
リソースの優先順位

3. 変更が加えられると、 **[Apply]** ボタンが有効になります。このボタンをクリックすると変更が適用されます。ウィンドウは閉じられません。

4. 完了したら、 **[OK]** をクリックし、変更内容を保存してウィンドウを閉じるか、または **[Cancel]** をクリックして、変更内容を適用せずにウィンドウを閉じます。

# 5.4.3.3.7. リソースの優先順位の編集

リソース階層が定義されているサーバの優先順位は、編集または変更することができます。最初に、リソースのプロパティを表示する 場合と同様に、Resource Properties ダイアログを表示してください。下に示すように Resource Properties ダイアログの [Equivalencies] タブに、サーバ上の特定のリソースに対する優先順位が表示されます。




優先順位を変更するには、次の 2 つの方法があります。

- **[Up]** / **[Down]** ボタンを使用してイクイバレンシを移動することにより、優先順位を変更してください。または
- 優先順位の値を直接編集してください。

## [Up] および [Down] ボタンの使用

1. Equivalencies 表で行をクリックして、イクイバレンシを選択してください。選択したイクイバレンシに応じて、**[Up]** または **[Down]** ボタンが有効になります。最も優先順位が高いサーバを選択した場合以外は、**[Up]** ボタンが有効になります。最も優先順位が低いサーバを選択した場合以外は、**[Down]** ボタンが有効になります。

2. **[Up]** または **[Down]** をクリックして、優先順位リストのイクイバレンシを移動してください。

数値の優先順位の列は変更されませんが、イクイバレンシがリスト内で上下に移動します。

## 優先順位の値の編集

1. Equivalencies 表の Priority 列で優先順位の値をクリックすることにより、優先順位を選択してください。優先順位の値の周りにボックスが表示され、値が強調表示されます。

2. 必要な優先順位を入力して、 **Enter** を押してください。

   **注記:** 有効なサーバの優先順位は、1 ～ 999 です。

優先順位を編集した後、Equivalencies 表が再ソートされます。

## 変更の適用

Equivalencies 表で必要な優先順位を設定したら、 **[Apply]** (または **[OK]** ) をクリックして変更を適用します。 **[Apply]** ボタンをクリックすると、加えられた変更内容が適用されます。 **[OK]** ボタンをクリックすると、加えられた変更内容が適用され、ウィンドウが閉じられます。 **[Cancel]** ボタンをクリックすると、 **[Apply]** が直前にクリックされているので、加えられた変更内容を保存せずにウィンドウが閉じられます。

# 5.4.3.3.8. リソース階層の拡張

LifeKeeper の **[Extend Resource Hierarchy]** オプションは、あるサーバから既存の階層をコピーして、別の LifeKeeper サーバ上に同様の階層を作成します。階層が他のサーバに拡張されると、そのリソースに対してカスケーディングフェイルオーバが使用可能になります。既存の階層が現在存在するサーバは、テンプレートサーバと呼ばれます。新たに拡張された階層が配置されるサーバは、ターゲットサーバと呼ばれます。

ターゲットサーバは、拡張された階層をサポートすることができ、他のリモートサーバ上の同等の階層と (アクティブな LifeKeeper コミュニケーションパスを介して) 通信できなければなりません。つまり、既存の階層内のリソースに関連付けられているすべてのリカバリキットが、ターゲットサーバだけでなく、階層が現在存在する他のすべてのサーバに既にインストールされている必要があります。

1. GUI を介してリソース階層を拡張するには、次の 5 つの方法があります。

- º 新しいリソース階層を

作成 してください。階層が作成されたことがダイアログに表示されたら、 **[Continue]** ボタンをクリックして、Pre-Extend Wizard を介して新しい階層の拡張を開始してください。
- º グローバルまたはサーバ固有のリソースアイコンを右クリックして、

リソースコンテキストメニュー を表示し、次に [Extend Resource Hierarchy] をクリックして、Pre-Extend Wizard を介して選択したリソースを拡張してください。
- º

グローバルツールバー で、 **[Extend Resource Hierarchy]** ボタンをクリックしてください。[Pre-Extend Wizard] ダイアログが表示されたら、 **[Template Server]** および **[Tag to Extend]** を選択し、それぞれ選択した後に **[Next]** をクリックしてください。
- º

リソースコンテキストツールバー で (表示された場合)、 **[Extend Resource Hierarchy]** ボタンをクリックして、Pre-Extend Wizard を表示してください。
- º

[Edit] メニュー で、 **[Resource]** を選択して、 **[Extend Resource Hierarchy]** をクリックしてください。[Pre-Extend Wizard] ダイアログが表示されたら、 **[Template Server]** および **[Tag to Extend]** を選択し、それぞれ選択した後に **[Next]** をクリックしてください。

2. デフォルトの **[Target Server]** を選択するか、または選択リストの中から 1 つ入力して、 **[Next]** をクリックしてください。

3. **[Switchback Type]** を選択して、 **[Next]** をクリックしてください。

4. デフォルト値を選択するか、または独自の **[Template Priority]** を入力して、 **[Next]** をクリックしてください。

5. 独自の **[Target Priority]** を選択するか入力して、 **[Next]** をクリックしてください。

6. ダイアログに、次に実行される拡張前のチェックが表示されます。これらのテストが成功した場合、LifeKeeper は、拡張している特定の種類のリソースに必要な手順の実行を開始します。

**[Extend Resource Hierarchy]** オプションの **[Accept Defaults]** ボタンは、LifeKeeper の **[Extend Resource Hierarchy]** のデフォルト値を熟知して、値の入力や確認をしないで素早く LifeKeeper リソース階層を拡張したいユーザ向けです。GUI ダイアログを使用して対話的に段階を追って LifeKeeper リソース階層を拡張する場合は、 **[Next]** ボタンを選択します。

✻ **注記:** マルチルート階層のすべてのルートは、まとめて拡張する必要があります。つまり、単一ルート階層として拡張することはできません。

✻ **注記:** コマンドラインによる手順については、SAP ドキュメントのコマンドラインからの SAP リソースの拡張 を参照してください。

# 5.4.3.3.8.1. ファイルシステムリソース階層の拡張

この操作は、リソース階層の拡張 に関するセクションで説明されているように、ファイルシステムリソース階層の作成 を完了した後に自動的に開始したり、既存のファイルシステムリソースから開始することができます。それが済んだら、次に以下の手順を完了します。これらの手順は、ファイルシステムリソースに固有のものです。

1. *[Extend gen/filesys Resource Hierarchy]* ダイアログボックスが表示されます。ファイルシステム階層に対して **[Mount Point]** を選択し、 **[Next]** をクリックしてください。

2. LifeKeeper が提供する **[Root Tag]** を選択するか、またはターゲットサーバ上のリソース階層に対する独自のタグを入力して、 **[Next]** をクリックしてください。

3. ダイアログに拡張操作のステータスが表示され、階層が正常に拡張されたことを示すメッセージが表示されて終了します。同じリソース階層を別のサーバに拡張する場合は、 **[Next Server]** をクリックしてください。その場合は、拡張の操作が繰り返されます。または、 **[Finish]** をクリックして、この操作を完了してください。

4. 拡張された階層が検証されると、確認情報がダイアログに表示されます。これが終了すると、 **[Done]** ボタンが有効になります。 **[Done]** をクリックして終了してください。

# 5.4.3.3.8.2. Generic Application リソース階層の拡張

この操作は、リソース階層の拡張 に関するセクションで説明されているように、Generic Application リソース階層の作成 を終了した後に自動的に開始したり、既存の Generic Application リソースから開始することができます。それが済んだら、次に以下の手順を完了します。これらの手順は、Generic Application リソースに固有のものです。

1. LifeKeeper が提供する **[Root Tag]** を選択するか、またはターゲットサーバ上のリソース階層に対する独自のタグを入力して、 **[Next]** をクリックしてください。

2. 次に **[Application Information]** (オプション) を入力し、 **[Next]** をクリックしてください。

3. ダイアログに拡張操作のステータスが表示され、階層が正常に拡張されたことを示すメッセージが表示されて終了します。同じリソース階層を別のサーバに拡張する場合は、 **[Next Server]** をクリックしてください。その場合は、拡張の操作が繰り返されます。または、 **[Finish]** をクリックして、この操作を完了してください。

4. 拡張された階層が確認されると、確認情報がダイアログに表示されます。これが終了すると、 **[Done]** ボタンが有効になります。 **[Done]** をクリックして終了してください。

# 5.4.3.3.8.3. Raw デバイスリソース階層の拡張

この操作は、リソース階層の拡張 に関するセクションで説明されているように、Raw デバイスリソース階層の作成 を終了した後に自動的に開始したり、既存の Raw デバイスリソースから開始することができます。それが済んだら、次に以下の手順を完了します。これらの手順は、Raw デバイスリソースに固有のものです。

1. LifeKeeper が提供する **[Root Tag]** を選択するか、またはターゲットサーバ上のリソース階層に対する独自のタグを入力して、 **[Next]** をクリックしてください。

2. ダイアログに拡張操作のステータスが表示され、階層が正常に拡張されたことを示すメッセージが表示されて終了します。同じリソース階層を別のサーバに拡張する場合は、 **[Next Server]** をクリックしてください。その場合は、拡張の操作が繰り返されます。または、 **[Finish]** をクリックして、この操作を完了してください。

3. 拡張された階層が検証されると、確認情報がダイアログに表示されます。これが終了すると、 **[Done]** ボタンが有効になります。 **[Done]** をクリックして終了してください。

# 5.4.3.3.9. 階層の拡張解除

LifeKeeper の **[Unextend Resource Hierarchy]** オプションは、単一サーバから階層全体 (すべてのリソースを含む) を削除します。これは、すべてのサーバから 1 つの階層を削除する **[Delete Resource Hierarchy]** 選択項目とは異なります。

**[Unextend Resource Hierarchy]** を使用する場合、既存の階層を削除するサーバは、ターゲットサーバと呼ばれます。

**[Unextend Resource Hierarchy]** 選択項目は、ターゲットへのアクティブな LifeKeeper コミュニケーションパスを持つ LifeKeeper サーバから使用することができます。

1. 開始するには、次の 5 つの可能な方法があります。

   - º 拡張解除したいリソース階層/サーバの組み合わせに対するアイコンを右クリックしてください。

リソースコンテキストメニュー が表示されたら、 **[Unextend Resource Hierarchy]** をクリックしてください。

   - º 拡張解除したいグローバルリソース階層のアイコンを右クリックしてください。

リソースコンテキストメニュー が表示されたら、 **[Unextend Resource Hierarchy]** をクリックしてください。ダイアログが表示されたら、リソース階層の拡張を解除するサーバを **[Target Server]** リストで選択し、**[Next]** をクリックしてください。

   - º

グローバルツールバー で、 **[Unextend Resource Hierarchy]** ボタンをクリックしてください。ダイアログが表示されたら、リソース階層の拡張を解除するサーバを **[Target Server]** リストで選択し、 **[Next]** をクリックしてください。次のダイアログで、 **[Hierarchy to Unextend]** リストから拡張解除したいリソース階層を選択し、再度 **[Next]** をクリックしてください。

   - º

リソースコンテキストツールバー で (表示された場合)、 **[Unextend Resource Hierarchy]** ボタンをクリックしてください。

   - º

[Edit] メニュー で、 **[Resource]** をポイントして、 **[Unextend Resource Hierarchy]** をクリックしてください。ダイアログが表示されたら、リソース階層の拡張を解除するサーバを **[Target Server]** リストで選択し、**[Next]** をクリックしてください。次のダイアログで、 **[Hierarchy to Unextend]** リストから拡張解除したいリソース階層を選択し、再度 **[Next]** をクリックしてください。

2. 拡張解除するように指定したサーバおよびリソース階層を確認するメッセージが、ダイアログに表示されます。 **[Unextend]** をクリックしてアクションを実行してください。

3. 出力パネル が有効な場合、ダイアログが閉じて、リソース階層の拡張を解除するコマンドの結果が出力パネルに表示されます。有効でない場合は、ダイアログが表示されたままこれらの結果が表示されます。 すべての結果が表示されたら、 **[Done]** をクリックして終了します。

# 5.4.3.3.10. リソース依存関係の作成

ほとんどの Recovery Kits では、元のリソース階層作成タスク中にそれらの依存関係が作成されますが、特定の条件下では、新規または追加のリソース依存関係を作成したり、既存のリソース依存関係を削除することが必要になる場合があります。一例として、既存の IP 依存関係を別の IP アドレスに変更する場合が挙げられます。リソース階層全体を削除して、新しいリソース階層を作成する代わりに、既存の IP 依存関係を削除して、異なる IP アドレスを持つ新しい依存関係を作成することができます。

1. 開始するには、次の 4 つの可能な方法があります。

   - º 親子依存関係を追加したい、サーバの下の親サーバ固有のリソース、または親グローバルリソースに対するアイコンを右クリックしてください。

リソースコンテキストメニュー が表示されたら、 **[Create Dependency]** をクリックしてください。 **注記:** 右ペインでサーバ固有のリソースを右クリックした場合、 **[Server]** の値はそのサーバになります。左ペインでグローバルリソースを右クリックした場合、 **[Server]** の値は、リソースが最も高い優先度を持つサーバになります。

- º

**グローバルツールバー** で、 **[Create Dependency]** ボタンをクリックしてください。ダイアログが表示されたら、リソース依存関係の作成を開始するサーバを **[Server]** リストで選択し、 **[Next]** をクリックしてください。次のダイアログで、 **[Parent Resource Tag]** リストから親リソースを選択し、再度 **[Next]** をクリックしてください。

- º

リソースコンテキストツールバー で (表示された場合)、 **[Create Dependency]** ボタンをクリックしてください。

- º

[Edit] メニュー で、 **[Resource]** をポイントして、 **[Create Dependency]** をクリックしてください。ダイアログが表示されたら、リソース依存関係の作成を開始するサーバを **[Server]** リストで選択し、 **[Next]** をクリックしてください。次のダイアログで、 **[Parent Resource Tag]** リストから親リソースを選択し、再度 **[Next]** をクリックしてください。

2. サーバ上の既存の有効なリソースのドロップダウンボックスから、 **[Child Resource Tag]** を選択してください。以下の例外を持つサーバ上で利用可能なすべてのリソースが、ダイアログに表示されます。

   - º 親リソース、その先祖、およびその子。
   - º 親リソースと同じサーバに拡張されていないリソース。
   - º 親リソースと同じ相対優先度を持たないリソース。
   - º 親リソースが稼働中の場合に、親と同じサーバ上で稼働していないリソース。

   **[Next]** をクリックして、次のダイアログに進んでください。

3. このダイアログで、依存関係の作成に対して適切な親および子のリソースタグが選択されていることを確認できます。 **[Create Dependency]** をクリックして、親を拡張したクラスタ内のすべてのサーバで依存関係を作成してください。

4. 出力パネル が有効な場合、ダイアログが閉じて、依存関係を作成するコマンドの結果が出力パネルに表示されます。有効でない場合は、ダイアログが表示されたままこれらの結果が表示されます。すべての結果が表示されたら、 **[Done]** をクリックして終了します。

# 5.4.3.3.11. リソース依存関係の削除

1. 開始するには、次の 4 つの可能な方法があります。

   - º 親子依存関係を削除したい、サーバの下の親サーバ固有のリソース、または親グローバルリソースに対するアイコンを右クリックしてください。

リソースコンテキストメニュー が表示されたら、 **[Delete Dependency]** をクリックしてください。
- º

グローバルツールバー で、 **[Delete Dependency]** ボタンをクリックしてください。ダイアログが表示されたら、リソース依存関係の削除を開始するサーバを **[Server]** リストで選択し、 **[Next]** をクリックしてください。次のダイアログで、 **[Parent Resource Tag]** リストから親リソースを選択し、再度 **[Next]** をクリックしてください。
- º

リソースコンテキストツールバー で (表示された場合)、 **[Delete Dependency]** ボタンをクリックしてください。
- º

[Edit] メニュー で、 **[Resource]** をポイントして、 **[Delete Dependency]** をクリックしてください。ダイアログが表示されたら、リソース依存関係の削除を開始するサーバを **[Server]** リストで選択し、 **[Next]** をクリックしてください。次のダイアログで、 **[Parent Resource Tag]** リストから親リソースを選択し、再度 **[Next]** をクリックしてください。

2. ドロップダウンボックスから **[Child Resource Tag]** を選択してください。これは、削除したい依存関係における子のタグ名である必要があります。 **[Next]** をクリックして、次のダイアログボックスに進んでください。

3. このダイアログで、依存関係の削除に対して適切な親および子のリソースタグが選択されていることを確認できます。 **[Delete Dependency]** をクリックして、クラスタ内のすべてのサーバ上の依存関係を削除してください。

4. 出力パネル が有効な場合、ダイアログが閉じて、依存関係を削除するコマンドの結果が出力パネルに表示されます。有効でない場合は、ダイアログが表示されたままこれらの結果が表示されます。すべての結果が表示されたら、[Done] をクリックして終了します。

# 5.4.3.3.12. すべてのサーバからの階層の削除

1. 開始するには、次の 5 つの可能な方法があります。

- 削除を開始するサーバの下の、削除したい階層にあるリソースのアイコンを右クリックしてください。リソースコンテキストメニュー が表示されたら、 **[Delete Resource Hierarchy]** をクリックしてください。

> **注記:** リソースをすべてのノードで out of service にせず先にリソース階層を削除してしまうといくつかのシステムリソースが使用できなくなる場合があります。このような状況をできるだけ避けるため、リソース階層を削除する前にすべてのノードでリソースを out of service にしておくことを推奨します。

- 削除したい階層にあるグローバルリソースのアイコンを右クリックしてください。リソースコンテキストメニュー が表示されたら、 **[Delete Resource Hierarchy]** をクリックしてください。ダイアログが表示されたら、リソース階層の削除を開始するサーバを **[Server]** リストで選択し、 **[Next]** をクリックしてください。

- グローバルツールバー で、 **[Delete Resource Hierarchy]** ボタンをクリックしてください。ダイアログが表示されたら、リソース階層の削除を開始するサーバを **[Target Server]** リストで選択し、 **[Next]** をクリックしてください。次のダイアログで、 **[Hierarchy to Delete]** リストから削除したい階層内のリソースを選択し、再度 **[Next]** をクリックしてください。

- プロパティーパネル のリソースコンテキストツールバー で (表示された場合)、 **[Delete Resource Hierarchy]** ボタンをクリックしてください。

- メニュー で、 **[Resource]** をポイントして、 **[Delete Resource Hierarchy]** をクリックしてください。ダイアログが表示されたら、リソース階層の削除を開始するサーバを **[Target Server]** リストで選択し、 **[Next]** をクリックしてください。次のダイアログで、 **[Hierarchy to Delete]** リストから削除したい階層内のリソースを選択し、再度 **[Next]** をクリックしてください。

2. 削除するために指定した階層を確認するメッセージが、ダイアログに表示されます。 **[Delete]** をクリックしてアクションを実行してください。

3. 出力パネル が有効な場合、ダイアログが閉じて、階層を削除するコマンドの結果が出力パネルに表示されます。有効でない場合は、ダイアログが表示されたままこれらの結果が表示されます。 すべての結果が表示されたら、 **[Done]** をクリックして終了します。

# 5.4.4. LifeKeeper User Guide

User Guide は、検索可能な総合リソースで、LifeKeeper の GUI で実行できる多くの作業の詳細情報があります。このドキュメンテーションにアクセスするには、User Guide をクリックしてください。

GUI から実行できる作業は 3 つの分野に分類できます。

共通の作業 - これらはどのユーザでも実行できる基本的な作業で、クラスタへの接続、サーバやリソースのプロパティの表示、ログファイルの表示、GUI の設定の変更などがあります。

オペレータの作業 - これらは Operator (オペレータ) の権限を必要とする高度な作業で、リソースを in service および out of service にする操作などがあります。

管理者の作業 - これらは、Administrator (管理者) の権限を必要とする作業です。サーバのプロパティの編集、リソースの作成、コミュニケーションパスの作成や削除などのサーバレベルの作業、およびリソースの編集、拡張、削除などのリソースレベルの作業があります。

以下の表に、それぞれのユーザ権限で使用できるデフォルトの作業を示します。特定のリソースタイプについてその他の作業が使用できることがあります。これらの作業については、関連するリソースキットのドキュメントで説明しています。

| 作業 | 権限 | | |
|---|---|---|---|
| | ゲスト | オペレータ | 管理者 |
| サーバとリソースの表示 | X | X | X |
| サーバへの接続と切断 | X | X | X |
| サーバのプロパティとログの表示 | X | X | X |
| サーバのプロパティの変更 | | | X |
| リソース階層の作成 | | | X |
| コミュニケーションパスの作成と削除 | | | X |
| リソースのプロパティの表示 | X | X | X |
| リソースのプロパティの変更 | | | X |
| リソースのサービス中とサービス休止の切り替え | | X | X |
| リソース階層の拡張と拡張解除 | | | X |
| リソースの依存関係の作成と削除 | | | X |
| リソース階層の削除 | | | X |

# 5.4.4.1. LifeKeeper for Linux の使用

以下のトピックでは、LifeKeeper のグラフィカルユーザインターフェース (GUI)、および LifeKeeper の GUI から実行できる多数の作業について詳しく説明しています。

---

GUI

ステータスの表

プロパティパネル

出力パネル

メッセージバー

GUI の終了

共通の作業

オペレータの作業

高度な作業

メンテナンス作業

テクニカルノート

# 5.4.4.1.1. GUI

GUI のコンポーネントは、LifeKeeper Core のインストールの一部として、すでにインストールされています。

LifeKeeper の GUI は、Java テクノロジを使用して、LifeKeeper およびその設定データ用にグラフィカルユーザインターフェースを提供します。LifeKeeper の GUI はクライアント / サーバアプリケーションなので、ユーザはクライアントシステムでグラフィカルユーザインターフェースを実行して、LifeKeeper が動作中のサーバシステムの監視や管理を行います。クライアントとサーバのコンポーネントは、同一システム上にある場合も、ない場合もあります。

---


- GUI の概要 - 全般
  - LifeKeeper GUI ソフトウェアパッケージ
- メニュー
  - リソースのコンテキストメニュー
  - サーバのコンテキストメニュー
  - [File] メニュー
  - [Edit] メニュー - [Resource]
  - [Edit] メニュー - [Server]
  - [View] メニュー
  - [Help] メニュー
- ツールバー
  - GUI のツールバー
  - リソースのコンテキストツールバー
  - サーバのコンテキストツールバー
- GUI の実行準備
  - LifeKeeper の GUI - 概要
  - LifeKeeper の GUI の設定

GUI サーバの開始 / 停止

Java のセキュリティポリシー

LifeKeeper サーバでの GUI の実行

# 5.4.4.1.1.1. GUI の概要 – 全般

クラスタマシンの必要なグループメンバシップをユーザが持っている限り、GUI を使用することで、任意のマシンから、任意のクラスタ内のサーバとリソースの管理、操作、または監視ができます (詳細については、GUI のユーザの設定 を参照)。GUI のサーバとクライアントのコンポーネントについて説明します。

## GUI サーバ

GUI サーバは、ハイパーテキスト転送プロトコル (HTTP) とリモートメソッド呼び出し (RMI) を使用して GUI クライアントと通信します。デフォルトでは、GUI サーバは LifeKeeper の起動時に初期化されますが、任意に設定することもできます。GUI サーバの開始 / 停止 を参照してください。

## GUI クライアント

GUI クライアントは、任意の LifeKeeper サーバ上のアプリケーション として実行できます。

クライアントには、以下のコンポーネントがあります。

- 左上のステータスの表 には、接続しているサーバとそのリソースの上位のステータスが表示されます。
- 右上のプロパティパネル には、ステータスの表で直前に選択したオブジェクトの詳細情報が表示されます。
- 下部の出力パネル には、コマンドの出力が表示されます。
- ウィンドウの最下部にあるメッセージバー には、処理のステータスメッセージが表示されます。
- コンテキストツールバー (プロパティパネル内) とグローバルツールバー を使用すると、頻繁に使用する作業に即座にアクセスできます。
- コンテキストメニュー (ポップアップ) とグローバルメニュー から、すべての作業にアクセスできます。

## GUI クライアントの終了

[File] メニュー から **[Exit]** を選択すると、すべてのサーバから切断され、クライアントが終了します。

# 5.4.4.1.1.1.1. LifeKeeper GUI ソフトウェアパッケージ

LifeKeeper GUI は、LifeKeeper Core パッケージクラスタにバンドルされている **steeleye-lkGUI** ソフトウェアパッケージに含まれています。 **steeleye-lkGUI** パッケージは、以下の動作を実行します。

- Java アーカイブフォーマットの LifeKeeper GUI クライアントをインストールする。
- LifeKeeper GUI サーバをインストールする。
- LifeKeeper 管理 Web サーバをインストールする。

  **注記:** LifeKeeper 管理 Web サーバは、パブリック Web サーバとは異なるポート 81 を使用するように設定されます。

- ディレクトリ */opt/LifeKeeper/htdoc/* に Java ポリシーファイルをインストールします。このファイルには、LifeKeeper GUI の実行に必要な最小限の権限があります。LifeKeeper GUI アプリケーションは、この場所にある java.policy ファイルを使用して、アクセスを制御します。
- GUI 管理用に LifeKeeper を準備する。

続行する前に、LifeKeeper サーバに LifeKeeper GUI パッケージがインストール済みであることを確認する必要があります。コマンド **`rpm -qi steeleye-lkGUI`** を入力して、このパッケージがインストール済みであるかどうかを確認できます。GUI パッケージがインストール済みである場合、出力にパッケージ名 **`steeleye-lkGUI`** が表示されます。

# 5.4.4.1.1.2. メニュー

## SIOS LifeKeeper for Linux のメニュー

---

リソースのコンテキストメニュー

サーバのコンテキストメニュー

[File] メニュー

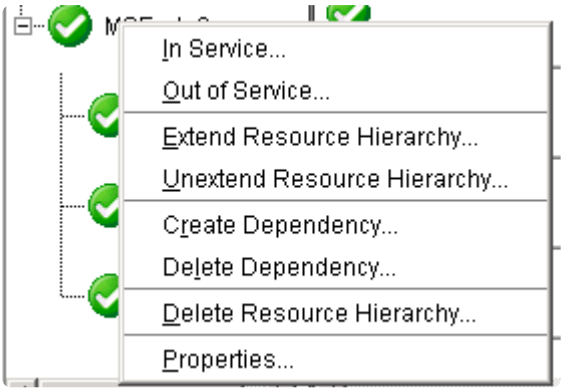
[Edit] メニュー - [Resource]

[Edit] メニュー - [Server]

[View] メニュー

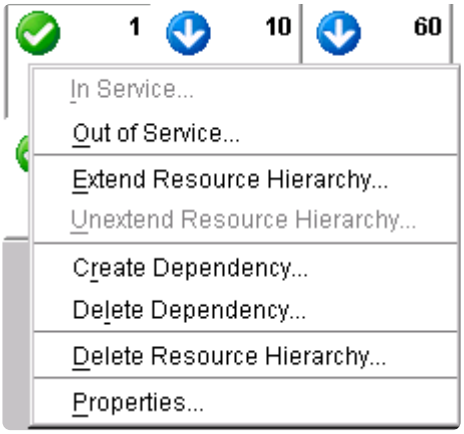
[Help] メニュー

# 5.4.4.1.1.2.1. リソースのコンテキストメニュー




リソースのコンテキストメニューは、ステータスの表 内にあるグローバル (クラスタ全体の) リソース (上図)、またはサーバ固有のリソースインスタンス (下図) を右クリックしたときに表示されます。デフォルトのリソースコンテキストメニューについて、ここで説明しますが、このメニューは特定のリソースタイプについてカスタマイズされていることがあります。この場合、メニューは該当するリソースキットのドキュメンテーションで説明されています。

選択したリソースについて、動作が呼び出されます。特定のサーバ上にあるリソースインスタンスを選択した場合、そのサーバについて動作が呼び出されます。一方、グローバル (クラスタ全体の) リソースを選択した場合は、サーバを選択する必要があります。




In Service -リソース階層を in service にします。

Out of Service -リソース階層を out of service にします。

Extend Resource Hierarchy -フェイルオーバをサポートするために、リソース階層を別のサーバに拡張します。

Unextend Resource Hierarchy -1 台のサーバから拡張リソース階層を削除します。

Create Dependency -2 つのリソース間に親 / 子の関係を作成します。

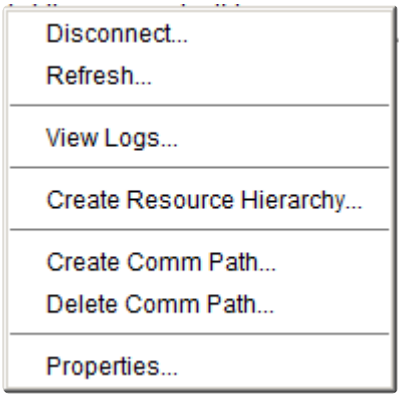
Delete Dependency -2 つのリソース間にある親 / 子の関係を削除します。

Delete Resource Hierarchy -リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。

Properties -[Resource Properties] ダイアログ を表示します。

# 5.4.4.1.1.2.2. サーバのコンテキストメニュー

サーバのコンテキストメニューは、ステータスの表 内にあるサーバアイコンを右クリックしたときに表示されます。このメニューは [Edit] メニューの [Server] サブメニューと同じですが、動作は常に、最初に選択したサーバ上で呼び出される点が異なります。




Disconnect -クラスタから切断します。

Refresh -GUI を最新情報に更新します。

View Logs -接続しているサーバについて、LifeKeeper のログメッセージを表示します。

Create Resource Hierarchy -リソース階層を作成します。

Create Comm Path -サーバ間にコミュニケーションパスを作成します。

Delete Comm Path -サーバからコミュニケーションパスを削除します。

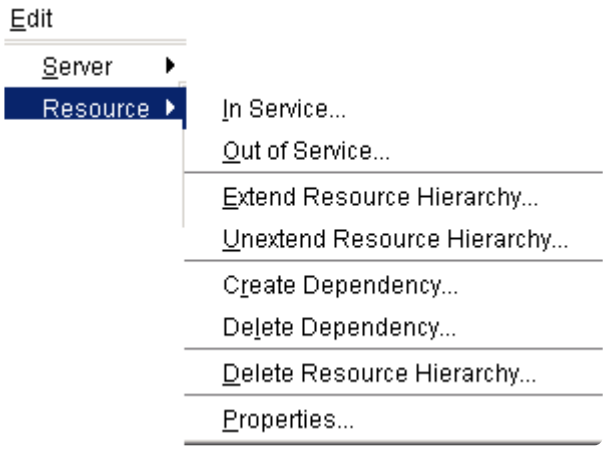
Properties -[Server Properties] ダイアログ を表示します。

# 5.4.4.1.1.2.3. [File] メニュー

**Connect** - LifeKeeper クラスタに接続します。LifeKeeper クラスタ内の各サーバに接続するには、そのサーバでログイン認証が必要です。

**Exit** - すべてのサーバから切断し、GUI のウィンドウを閉じます。

# 5.4.4.1.1.2.4. [Edit] メニュー - [Resource]




In Service - リソース階層を in service にします。

Out of Service - リソース階層を out of service にします。

Extend Resource Hierarchy - フェイルオーバをサポートするために、リソース階層を別のサーバに拡張します。

Unextend Resource Hierarchy - 1 台のサーバから拡張リソース階層を除去します。

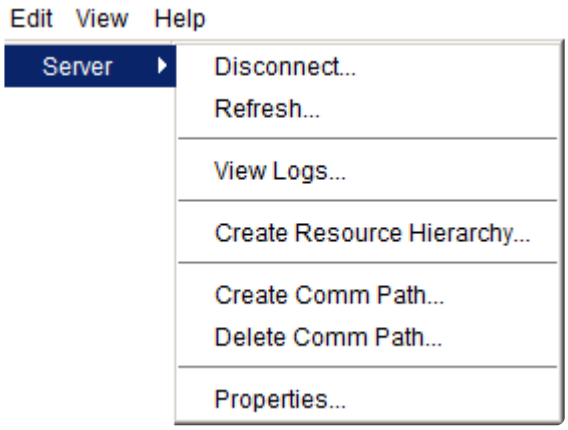
Create Dependency - 2 つのリソース間に親 / 子の関係を作成します。

Delete Dependency - 2 つのリソース間にある親 / 子の関係を削除します。

Delete Resource Hierarchy - リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。

Properties - [Resource Properties] ダイアログ を表示します。

# 5.4.4.1.1.2.5. [Edit] メニュー - [Server]




Disconnect - クラスタから切断します。

Refresh - GUI を最新情報に更新します。

View Logs - 接続しているサーバについて、LifeKeeper のログメッセージを表示します。

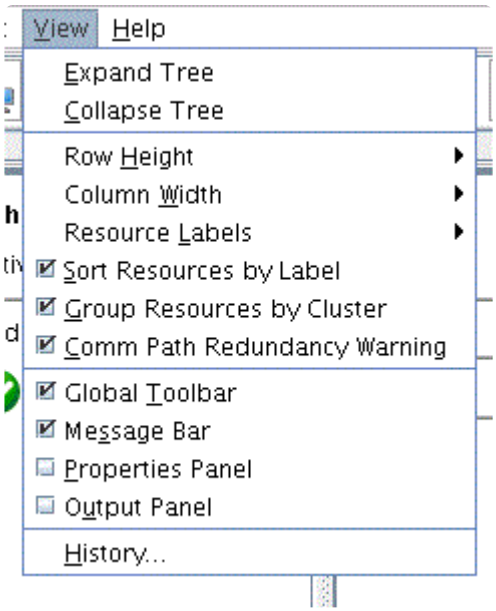
Create Resource Hierarchy - リソース階層を作成します。

Create Comm Path - サーバ間にコミュニケーションパスを作成します。

Delete Comm Path - サーバからコミュニケーションパスを削除します。

Properties -[Server Properties] ダイアログ を表示します。

# 5.4.4.1.1.2.6. [View] メニュー




Expand Tree -リソース階層ツリー全体を展開します。

Collapse Tree -リソース階層ツリー全体を折り畳みます。

**Row Height** -リソース階層ツリーおよびリソーステーブルの行の高さを修正します。表示されたリソースの数により大、中、小を選択してください。

**Column Width** -リソース階層ツリーおよびリソーステーブルの列幅を修正します。表示されたリソースにより 自動 、大、中、小を選択してください。

Resource Labels -このオプショングループを使用すると、リソース階層ツリー内のリソースを、タグ名別と ID 別のいずれで表示するかを指定できます。

**Sort Resources by Label** -リソースラベルのみでリソースを分類します。

**Group Resources by Cluster** -クラスタサーバおよびリソースラベルによって分類します。同じクラスタ内に属するリソースが同じグループになります。

**Comm Path Redundancy Warning** -サーバステータスグラフィックでコミュニケーションパスの状態についての説明を明記します。

- 選択した場合は、一組のサーバ間のコミュニケーションパスが冗長化されていない場合、サーバ警告が表示されます。
- 選択しない場合は、一組のサーバ間のコミュニケーションパスが冗長化されていない場合は無視しますが、コミュニケーションパスが切れた場合には、サーバ警告が表示されます。

Global Toolbar -チェックボックスがオンの場合、このコンポーネントを表示します。

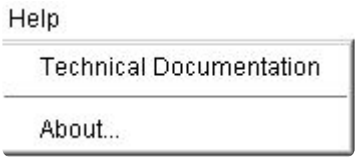
Message Bar -チェックボックスがオンの場合、このコンポーネントを表示します。

Properties Panel -チェックボックスがオンの場合、このコンポーネントを表示します。

Output Panel -チェックボックスがオンの場合、このコンポーネントを表示します。

History -メッセージバーに表示された最新メッセージを、LifeKeeper の GUI の [Message History] ダイアログボックスに表示します (最大 1000 行)。

# 5.4.4.1.1.2.7. [Help] メニュー




**Technical Documentation -** SIOS Technology Corp. のテクニカルドキュメンテーションの開始ページを表示します。

**About...** - LifeKeeper GUI のバージョン情報を表示します。

# 5.4.4.1.1.3. ツールバー

## SIOS LifeKeeper for Linux のツールバー

---

GUI のツールバー

リソースのコンテキストツールバー

サーバのコンテキストツールバー

# 5.4.4.1.1.3.1. GUI のツールバー

このツールバーは、プロパティパネル に表示されるデフォルトのサーバ とリソース のコンテキストツールバーを組み合わせたものですが、このツールバーから動作を実行するときには、サーバとリソースを選択する必要があります。

| | |
|---|---|
| | Connect -LifeKeeper クラスタに接続します。 |
| | Disconnect -LifeKeeper クラスタから切断します。 |
| | Refresh -GUI を最新情報に更新します。 |
| | View Logs -接続しているサーバについて、LifeKeeper のログメッセージを表示します。 |
| | Create Resource Hierarchy -リソース階層を作成します。 |
| | Delete Resource Hierarchy -リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。 |
| | Create Comm Path -サーバ間にコミュニケーションパスを作成します。 |
| | Delete Comm Path -サーバからコミュニケーションパスを削除します。 |
| | In Service -リソース階層を in service にします。 |
| | Out of Service -リソース階層をサービス停止にします。 |
| | Extend Resource Hierarchy -フェイルオーバをサポートするために、リソース階層を別のサーバに拡張します。 |
| | Unextend Resource Hierarchy -1 台のサーバから拡張リソース階層を削除します。 |

| | |
|---|---|
| | Create Dependency -2 つのリソース間に親 / 子の関係を作成します。 |
| | Delete Dependency -2 つのリソース間にある親 / 子の関係を削除します。 |

# 5.4.4.1.1.3.2. リソースのコンテキストツールバー

ステータスの表 からサーバ固有のリソースインスタンスを選択すると、プロパティパネル にリソースのコンテキストツールバーが表示されます。

選択したサーバとリソースについて、動作が呼び出されます。灰色表示のリソースについて、動作を選択することはできません。

| | |
|---|---|
| | In Service -リソース階層を in service にします。 |
| | Out of Service -リソース階層を out of service にします。 |
| | Extend Resource Hierarchy -フェイルオーバをサポートするために、リソース階層を別のサーバに拡張します。 |
| | Unextend Resource Hierarchy -1 台のサーバから拡張リソース階層を削除します。 |
| | Add Dependency - 2 つのリソース間に親 / 子の関係を作成します。 |
| | Remove Dependency - 2 つのリソース間にある親 / 子の関係を削除します。 |
| | Delete Resource Hierarchy -リソース階層をすべてのサーバから削除します。 |

# 5.4.4.1.1.3.3. サーバのコンテキストツールバー

ステータスの表 からサーバを選択すると、プロパティパネル にサーバのコンテキストツールバーが表示されます。選択したサーバについて、動作が呼び出されます。

| | |
|---|---|
| | Disconnect -LifeKeeper クラスタから切断します。 |
| | Refresh -GUI を最新情報に更新します。 |
| | View Logs -接続しているサーバについて、LifeKeeper のログメッセージを表示します。 |
| | Create Resource Hierarchy -リソース階層を作成します。 |
| | Delete Resource Hierarchy -リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。 |
| | Create Comm Path -サーバ間にコミュニケーションパスを作成します。 |
| | Delete Comm Path -サーバからコミュニケーションパスを削除します。 |

# 5.4.4.1.1.4. GUI の実行の準備

LifeKeeper の GUI - 概要

LifeKeeper の GUI の設定

GUI サーバの開始 / 停止

Java のセキュリティポリシー

LifeKeeper サーバでの GUI の実行

# 5.4.4.1.1.4.1. LifeKeeper の GUI の設定

## GUI 管理用の LifeKeeper サーバーの設定

各 LifeKeeper サーバーについて、以下の手順を実行してください。各手順には、詳細手順の参照先またはリンクがあります。

1. 各サーバーに、Java 実行時環境 (JRE) または Java ソフトウェア開発キット (JDK) をインストールする必要があります。必要な Java のバージョンについては、SPS for Linux リリースノート を参照してください。

2. 各サーバーで、LifeKeeper GUI サーバーを開始してください (GUI サーバーの開始/停止 を参照)。
   **注記:** GUI サーバーが後続の初期インストールを開始した後、LifeKeeper の開始と停止は、GUI サーバーを含む LifeKeeper のすべてのデーモンプロセスの開始と停止を行います。

3. root 以外のユーザに GUI の使用を許可するように計画している場合は、GUI ユーザの設定 が必要です。

## GUI の実行

LifeKeeper の GUI は、クラスター内の LifeKeeper サーバーで実行できます。

LifeKeeper クラスター内のサーバーで GUI の設定と実行を行う方法については、LifeKeeper サーバーでの GUI の実行 を参照してください。

## GUI の設定

| 項目 | 説明 |
|---|---|
| **GUI** のクライアントとサーバーの通信 | LifeKeeper GUI のクライアントとサーバーは、通信に Java のリモートメソッド呼び出し (RMI) を使用します。RMI が正しく動作するためには、クライアントとサーバーは解決可能なホスト名または IP アドレスを使用する必要があります。DNS が実装されていない場合 (または、他の名前のルックアップメカニズムを使用して名前が解決できない場合) は、クライアントとサーバーのそれぞれについて、*/etc/hosts* ファイルを編集し、他のすべての LifeKeeper サーバーの名前とアドレスを含めてください。 |
| **GUI** サーバーの **Java** プラットフォーム | LifeKeeper GUI サーバーには、Java 実行環境 (JRE) - Java 仮想マシン、Java プラットフォームのコアクラス、およびサポートするファイル - をインストールする必要があります。LifeKeeper のGUI は、OpenJDK をサポートしています。LifeKeeper インストール時に実行する setup スクリプトが OS 同梱の OpenJDK をインストールします。ただし、以下の環境では Linux ディストリビューターがサポートしている OpenJDK を提供していないため、LifeKeeper のインストールイメージに同梱している OpenJDK をインス |

| | |
|---|---|
| | トールします。サポートしている OpenJDK のバージョンは、リリースノートを参照してください。<br><br>• RedHat Enterprise Linux/CentOS/Oracle Linux 7.0<br><br>**注記:** LifeKeeper インストール時に、GUI が使用する JRE のパスを LifeKeeper のデフォルトファイル `/etc/default/LifeKeeper` の PATH に設定します。使用する JRE を変更したい場合はこの PATH を編集してください。このファイルの編集時に LifeKeeper が実行中である場合は、変更内容を認識させるために LifeKeeper GUI サーバーと GUI クライアントを停止し、再起動する必要があります。<br><br>**注記:** 以下の LifeKeeper と OS のバージョンの組み合わせでは、LifeKeeper と OS 同梱の OpenJDK の間でメモリ管理の不整合があります。それを回避するため、LifeKeeper のインストールイメージに同梱している OpenJDK を `/opt/LifeKeeper/lib64/java` 以下にインストールして使用しています。メモリ管理の不整合は既に修正済みですが、LifeKeeper をアップグレードした場合は `/opt/LifeKeeper/lib64/java` にインストールされた JRE をそのまま使用し続けます。<br>OS 同梱の JRE を使用したい場合は、JRE をインストールして、上記の手順に従って /etc/default/LifeKeeper の PATH を変更してください。`/opt/LifeKeeper/lib64/java` は不要ですので削除しても問題ありません。<br><br>• LifeKeeper<br>◦ v9.4.1 ～ v9.5.1<br>• OS<br>◦ RedHat Enterprise Linux/CentOS/Oracle Linux 8.0 ～ 8.2<br>◦ SUSE Linux Enterprise Server 15.0 ～ SP2 |
| **Java Runtime Environment** のアンインストール | • LifeKeeperのインストールイメージに同梱しているOpenJDKパッケージをインストールした環境<br>LifeKeeper のアンインストール時にOpenJDKパッケージも同時にアンインストールされます。<br>• Linuxディストリビューターが提供しているOpenJDKをインストールした環境<br>LifeKeeper のアンインストール時にOpenJDKパッケージはアンインストールされません。必要であれば手動でアンインストールを行ってください。 |
| **Java** リモートオブジェクトレジストリーのサーバーポート | LifeKeeper GUI サーバーは、各 LifeKeeper サーバー上の Java リモートオブジェクトレジストリー用にポート 82 を使用します。これにより、サーバーは、典型的なファイアウォールの後にあるクライアントからの RMI 呼び出しをサポートできます。 |
| **LifeKeeper** の管理 **Web** サーバー | LifeKeeper GUI サーバーには、クライアントのブラウザーの通信用に管理 Web サーバーが必要です。現在、LifeKeeper GUI サーバーは、管理 Web サーバーとして lighttpd Web サーバーのプライベートコピーを使用しています。この Web サーバーは、steeleye-lighttpd パッケージによりインストールと設定が実行され、他の Web サーバーとの競合を避けるためにポート 81 を使用します。 |
| **GUI** クライア | LifeKeeper GUI クライアントには、LifeKeeper クラスター内のすべてのホストへのネット |

| | |
|---|---|
| ントのネットワークアクセス | ワークアクセスが必要です。LifeKeeper GUI クライアントをブラウザー内で実行する場合、アプレットへのネットワークアクセスを可能にするためにセキュリティーレベルを低下させる必要があります。セキュリティーを低い値に設定した状態で、他のサイトを閲覧しないように注意してください (つまり、イントラネットまたは信頼できるサイトについてのみセキュリティー設定を変更する)。 |

## GUI の制限

| 項目 | 説明 |
|---|---|
| **GUI の相互運用性の制限** | LifeKeeper for Linux クライアントは、Linux サーバー上の LifeKeeper の管理にのみ使用できます。LifeKeeper for Linux の GUI と LifeKeeper for Windows は、同時には使用できません。 |

# 5.4.4.1.1.4.2. GUI サーバの開始 / 停止

## LifeKeeper GUI サーバを開始するには

LifeKeeper GUI サーバが動作していない場合は、 *root* として以下のコマンドを入力してください。

```
/opt/LifeKeeper/bin/lkGUIserver start
```

このコマンドは、管理しているサーバで LifeKeeper GUI サーバのデーモンプロセスが現在動作していない場合、それらのデーモンプロセスをすべて開始します。以下のようなメッセージが表示されます。

```
# Installing GUI Log
# LK GUI Server Startup at:
# Mon May 8 14:14:46 EDT 2006
# LifeKeeper GUI Server Startup completed at:
# Mon May 8 14:14:46 EDT 2006
```

LifeKeeper GUI サーバが開始した後、以降の LifeKeeper の開始操作はすべて、LifeKeeper GUI サーバのプロセスを自動的に開始します。

### トラブルシューティング

LifeKeeper GUI は、各サーバのポート 81 を管理 Web サーバ用に、ポート 82 を Java リモートオブジェクトレジストリに使用します。他のアプリケーションがそれらのポートを使用している場合、LifeKeeper GUI は正しく機能しません。これらの値は、ファイル ***/etc/default/LifeKeeper*** の以下のエントリを編集することにより変更できます。

```
GUI_WEB_PORT=81 GUI_RMI_PORT=82
```

**注記：**これらのポートの値は、起動時に GUI サーバで初期化されます。ポートの値を変更した場合、steeleye-lighttpdプロセスを停止し、再起動する必要があります。これらの値は、接続するすべてのクラスタ全体で同一である必要があります。

## LifeKeeper GUI サーバを停止するには

LifeKeeper GUI サーバが動作している場合は、 *root* として以下のコマンドを入力してください。

```
/opt/LifeKeeper/bin/lkGUIserver stop
```

このコマンドは、管理しているサーバで LifeKeeper GUI サーバのデーモンプロセスが現在動作している場合、それらのデーモンプロセスをすべて停止します。以下のメッセージが表示されます。

```
# LifeKeeper GUI Server Shutdown at:
# Fri May 19 15:37:27 EDT 2006
# LifeKeeper GUI Server Shutdown Completed at:
```

```
# Fri May 19 15:37:28 EDT 2006
```

## LifeKeeper GUI サーバのプロセス

LifeKeeper GUI サーバが動作していることを確認するには、以下のコマンドを入力してください。

```
ps -ef | grep runGuiSer
```

以下のような出力が表示されます。

```
root 2805 1 0 08:24 ? 00:00:00 sh/opt/LifeKeeper/bin/runGuiSer
```

現在動作している他の GUI サーバのデーモンプロセスのリストを表示するには、以下のコマンドを入力してください。

```
ps -ef | grep S_LK
```

以下のような出力が表示されます。

```
root 30228 30145 0 11:20 ? 00:00:00 java -Xint -Xss3M
-DS_LK=true -Djava.rmi.server.hostname=thor48 ...
```

# 5.4.4.1.1.4.3. Java のセキュリティポリシー

LifeKeeper の GUI は、ポリシーベースのアクセス制御を使用します。GUI クライアントのロード時に、現在有効なセキュリティポリシーに基づいて権限がGUI クライアントに割り当てられます。ポリシーはさまざまな署名者 / 場所からのコードに提供される権限を指定し、外部から設定可能なポリシーファイルから初期化されます。

デフォルトでは、システム全体のポリシーファイルとオプションのユーザポリシーファイルが 1 つずつあります。システム全体でコードに権限を付与するシステムポリシーファイルが先にロードされ、次にユーザポリシーファイルが追加されます。LifeKeeper GUI がアプリケーションとして起動される場合は、これらのポリシーファイルに加えて、LifeKeeper GUI のポリシーファイルもロードされることがあります。

## ポリシーファイルの場所

デフォルトでは、システムポリシーファイルは以下の場所にあります。

*<JAVA.HOME>/lib/security/java.policy (Linux)*

*<JAVA.HOME>\lib\security\java.policy (Windows)*

**注記:** JAVA.HOME は、システムのプロパティ「JAVA.HOME」の値を指し、JRE または JDK がインストールされたディレクトリの場所を指定します。

ユーザポリシーファイルは「.」の文字で始まり、デフォルトでは以下の場所にあります。

*<USER.HOME>\.java.policy*

**注記:** USER.HOME は、システムのプロパティ「user.home」の値を指し、ユーザのホームディレクトリを指定します。例えば、Windows NT ワークステーション上にあるユーザ Paul のホームディレクトリは、「paul.000」です。

Windows システムの場合、user.home のプロパティ値のデフォルト値は以下のとおりです。

*C:\WINNT\Profiles\<USER> (マルチユーザ **Windows NT** システム)*

*C:\WINDOWS\Profiles\<USER> (マルチユーザ **Windows 95/98** システム)*

*C:\WINDOWS (シングル – ユーザ **Windows 95/98** システム)*

デフォルトでは、LifeKeeper GUI のポリシーファイルは以下の場所にあります。

*/opt/LifeKeeper/htdoc/java.policy **(Linux)***

## ポリシーファイルの作成と管理

デフォルトでは、LifeKeeper GUI がアプリケーションとして起動される場合に LifeKeeper GUI のポリシ

ーファイルが使用されます。LifeKeeper GUI をアプレットとして実行する場合、ホームディレクトリにユーザポリシーファイルを作成する必要があります (存在しない場合)。ユーザポリシーファイルは、LifeKeeper GUI を実行するために必要な最低限の権限を指定する必要があります。このトピックの「ポリシーファイルの例」セクションで後述します。

ポリシーファイルの作成と管理は、単純なテキストエディタ、または Java 実行時環境 (JRE) や Java 開発キット (JDK) に含まれるグラフィカルな **Policy Tool** ユーティリティから行うことができます。Policy Tool を使用すると、入力が簡略化され、ポリシーファイルに必要な構文の知識が不要になります。Policy Tool の使用方法の詳細については、http://docs.oracle.com/javase/8/docs/technotes/tools/ にある Policy Tool のドキュメンテーションを参照してください。

LifeKeeper GUI を実行するために必要な最低限の権限を持つ **ユーザポリシーファイルを作成する最も簡単な方法** は、 ***/opt/LifeKeeper/htdoc/java.policy*** にある LifeKeeper GUI のポリシーファイルをホームディレクトリにコピーし、ファイル名を *.java.policy* に変更することです (ファイル名の前にあるドットは必須)。Windows システムでは、ファイル *http://<server name>:81/java.policy* (<server name> は LifeKeeper サーバのホスト名) を開いてホームディレクトリに *.java.policy* の名前を付けて保存することで、LifeKeeper GUI のポリシーファイルをコピーできます。ユーザポリシーファイルの正しい場所を特定する必要がある場合は、Java のコントロールパネルを使用して Java コンソールを有効にし、LifeKeeper GUI をアプレットとして起動します。ユーザポリシーファイルのホームディレクトリのパスが、Java コンソールに表示されます。

## ポリシーファイルでの権限の付与

権限は、システムリソースへのアクセスを表します。アプレットにリソースへのアクセスを許可するには、対応する権限を、アクセスを試行するコードに明示的に付与する必要があります。権限は通常、名前を持ち (「ターゲット名」として参照される)、場合によっては、1 つ以上の動作を含むカンマ区切りリストを持ちます。例えば、以下のコードは、 */tmp* ディレクトリのファイル *abc* に対する読み取りアクセスを表す FilePermission オブジェクトを作成します。

```
perm = new java.io.FilePermission("/tmp/abc","read");
```

この例では、ターゲット名は「 */tmp/abc* 」、動作文字列は「 *read* 」です。

ポリシーファイルは、指定したコードソースからのコードに許可する権限を指定します。この例で、*/home/sysadmin* ディレクトリのコードにファイル */tmp/abc* への読み取りアクセスを付与するポリシーファイルのエントリは以下のとおりです。

```
grant codeBase "file:/home/sysadmin/" { permissionjava.io.FilePermission
"/tmp/abc", "read"; };
```

## ポリシーファイルの例

このポリシーファイルの例には、LifeKeeper GUI の実行に必要な最小限の権限があります。このポリシーファイルは、LifeKeeper GUI パッケージにより ***/opt/LifeKeeper/htdoc/java.policy*** にインストールされます。

```
/*
```

```
* LifeKeeper GUI に必要な権限。コードベースで
* これを制限することもできます。ただし、そのようにする場合は、リカバリキットが
* 任意のコードベース付きの任意の jar コンポーネントを持つことができることに
* 注意してください。したがって、これらも含めるには
* grant 文を変更する必要があります。
*/
grant {

/*
* LifeKeeper クラスタ内のすべてのマシンに対して
* これを行うことができなければなりません。それに合わせてネットワーク仕様を
* 制限することもできます。
*/
permission java.net.SocketPermission"*", "accept,connect,resolve";
/*
* リモートプロパティファイルおよび
* jar ファイルを取得するには、URLClassLoaders を使用してください。
*/
permission java.lang.RuntimePermission"createClassLoader";
/*
* GUI をアプリケーションとして実行する場合のみ以下が必要です
* (デフォルトの RMI セキュリティマネージャは、
* ブラウザがアプレット用にインストールするセキュリティマネージャよりも
* 制限あり)。
*/
permission java.util.PropertyPermission "*","read";
permission java.awt.AWTPermission "*";
permission java.io.FilePermission "<<ALL FILES>>","read,execute";
};
```

# 5.4.4.1.1.4.4. LifeKeeper サーバでの GUI の実行

LifeKeeper GUI を実行する最も簡単な方法は、LifeKeeper サーバでアプリケーションとして実行することです。これは、実際には、同一システム上で GUI のクライアントとサーバを実行することです。

1. GUI 管理用の LifeKeeper サーバを設定した後、root として以下のコマンドを入力することにより、サーバ上で GUI をアプリケーションとして実行できます。

   ```
   /opt/LifeKeeper/bin/lkGUIapp
   ```

2. lkGUIapp スクリプトが適切な環境変数を設定して、アプリケーションを開始します。アプリケーションのロード時に、LifeKeeper のアプリケーション指定ダイアログまたはスプラッシュ画面が表示されます。

3. アプリケーションのロード後、LifeKeeper の GUI が表示され、[Cluster Connect] ダイアログが自動的に表示されます。接続先のサーバ名、およびログインとパスワードを入力してください。

4. クラスタへの接続が確立した後、GUI のウィンドウに、接続しているサーバにより保護されているリソースとステータスがグラフィックで表示されます。GUI のメニューとツールバーのボタンから、管理機能を使用できます。

# 5.4.4.1.1.4.5. LifeKeeper の GUI – 概要

LifeKeeper の GUI は、Java テクノロジを使用して、LifeKeeper およびその設定データとのグラフィカルなステータスのインターフェースを提供します。LifeKeeper の GUI はクライアント / サーバアプリケーションなので、ユーザはクライアントシステムでグラフィカルユーザインターフェースを実行して、LifeKeeper が動作中のサーバシステムの監視や管理を行います。クライアントとサーバは、同一システム上にある場合も、ない場合もあります。クラスタマシンの必要なグループメンバシップをユーザが持っている限り、LifeKeeper の GUI を使用することで、任意のマシンから、任意のクラスタ内のサーバとリソースの管理、操作、または監視ができます (詳細については、GUI のユーザの設定 を参照)。LifeKeeper GUI のサーバとクライアントのコンポーネントについて説明します。

## GUI サーバ

システムの起動時に、LifeKeeper クラスタ内の各サーバで、LifeKeeper GUI サーバが初期化されます。LifeKeeper GUI サーバは、Java ネイティブインターフェース (JNI) 経由で LifeKeeper Core ソフトウェアと、リモートメソッド呼び出し (RMI) を使用して LifeKeeper GUI と通信します。

## GUI クライアント

LifeKeeper GUI クライアントは、Linux システム上のアプリケーションとして、または Windows や Unix システム上の Web ブラウザから呼び出し可能なアプレットとして動作するように設計されています。

LifeKeeper GUI クライアントには、以下のグラフィカルコンポーネントがあります。

- 左上のステータスの表 には、接続しているサーバとそのリソースの上位のステータスが表示されます。
- 右上のプロパティパネル には、ステータスの表で直前に選択したオブジェクトの詳細情報が表示されます。
- 下部の出力パネル には、コマンドの出力が表示されます。
- ウィンドウの最下部にあるメッセージバーには、処理のステータスメッセージが表示されます。
- サーバのコンテキスト ツールバーと リソースのコンテキスト ツールバー (プロパティパネル内)、およびグローバルツールバー からは、頻繁に使用する作業に即座にアクセスできます。
- サーバのコンテキスト メニューと リソースのコンテキスト メニュー (ポップアップ) およびグローバルメニュー ([File] 、[Edit Server] 、[Edit Resource] 、[View] 、および [Help] ) からは、すべての作業にアクセスできます。

グラフィックのリソース、サーバ、または表のセルを右クリックすると、コンテキストメニューが表示されます。また、多くの作業はこれらのコンテキストメニューから開始できます。この場合、リソースとサーバは自動的に指定されます。

# アプリケーションクライアントの開始

ある LifeKeeper サーバで管理者権限を持つユーザは、そのサーバからアプリケーションクライアントを実行できます。LifeKeeper GUI アプリケーションを開始するには、グラフィカルウィンドウから /opt/LifeKeeper/bin/lkGUIapp を実行してください。

この操作を実行してもクライアントがロードされない場合は、GUI のトラブルシューティング を参照してください。

# GUI クライアントの終了

[File] メニュー から [Exit] を選択すると、すべてのサーバから切断され、クライアントが終了します。

# 5.4.4.1.2. ステータスの表

ステータスの表には、接続しているサーバとそのリソースのステータスがグラフィック表示されます。以下の項目が表示されます。

- 最も上の行に、各サーバの状態。
- 左端の列に、各リソースのグローバル (サーバ全体での) 状態と親 / 子の関係。
- 残りのセルに、各サーバの各リソースの状態。

サーバとリソースの状態は、グラフィックス、テキスト、および色を使用して表示されます。サーバのテーブルの空白セルは、特定のリソースがそのサーバで定義されていないことを示します。

ステータスの表でサーバまたはリソースを選択した場合、その項目の詳細な状態の情報とコンテキスト依存ツールバーがプロパティパネル に表示されます。また、任意の項目のセルを右クリックすることで、該当するサーバのコンテキストメニュー またはリソースのコンテキストメニュー をポップアップ表示できます。

ステータスの表は 2 つのセクションに分かれています。左右のセクションの境界を移動して、それらのセクションの相対サイズを変更できます。また、ステータスの表を折り畳んで、階層ツリーの上位項目のみを表示できます。ツリーのリソース項目の折り畳み / 展開 を実行すると、表内にリストされる階層に対しても折り畳み / 展開が適用されます。

# 5.4.4.1.3. プロパティパネル

プロパティパネルには、ステータスの表から選択されたサーバまたはリソースのプロパティが表示されます。プロパティパネルは、[Server Properties] ダイアログ または [Resource Properties] ダイアログ と同じ機能を持ち、さらに一般的に使用するコマンドに即座にアクセスできるコンテキスト依存ツールバーがあります。このパネルの上部には、サーバを選択した場合は ***server_name*** 、リソースを選択した場合は ***server_name: resource_name*** がキャプションとして表示されます。

プロパティパネルに表示されるコンテキスト依存ツールバーは、サーバのコンテキストツールバー とリソースのコンテキストツールバー です。サーバまたはリソースのツールバーもカスタマイズできます。ツールバーのカスタマイズの詳細については、該当するApplication Recovery Kit のドキュメンテーション を参照してください。

プロパティパネル下部にあるボタンは、以下の機能を持ちます。

- **[Apply]** ボタンは、パネルの編集可能なプロパティに対する変更内容を適用します。このボタンが有効になるのは、編集可能なプロパティを変更した場合のみです。
- **[Reset]** ボタンは、サーバにすべてのプロパティの現在の値を照会し、これまで変更した内容を消去します。このボタンは常に有効です。
- **[Help]** ボタンは、プロパティパネルのコンテキスト依存ヘルプを表示します。このボタンは常に有効です。

プロパティパネルのサイズを増減するには、パネルの左端にある境界を左右にスライドしてください。このパネルを開閉するには、[View] メニュー の **[Properties Panel]** チェックボックス を使用してください。

# 5.4.4.1.4. 出力パネル

出力パネルは、LifeKeeper GUI クライアントが送出したコマンドの出力を収集します。コマンドの実行開始時に、タイムスタンプ付きのラベルが出力パネルに追加され、そのラベルの下に、そのコマンドの出力がすべて追加されます。複数のコマンドを同時に実行する場合 (通常は異なるサーバ上)、各コマンドの出力が対応するセクションに送られ、各コマンドの結果が見やすくなります。

出力パネルのサイズを増減するには、パネル上部にある境界を上下にスライドしてください。このパネルを開閉するには、[View] メニュー の **[Output Panel]** チェックボックスを使用してください。出力パネルを閉じているときには、各コマンドを開始するダイアログが表示されたままになり、このダイアログを閉じるまで出力がこのダイアログに表示されます。そして、このダイアログを閉じた後はコマンドの出力を確認できなくなります。出力パネルを再び開いた後は、LifeKeeper の GUI はデフォルトの動作に戻ります。

# 5.4.4.1.5. メッセージバー

メッセージバーは、[Status] ウィンドウの下に表示されます。メッセージが 1 行のテキストで表示されます。「Connecting to Server X」や「Failure to connect to Server X」などのメッセージが表示されます。

メッセージバーを非表示にするには、[View] メニュー の **[Message Bar]** チェックボックスをオフにします。

メッセージバーを表示するには、[View] メニューの **[Message Bar]** チェックボックスをオンにします。

メッセージバーに表示されたメッセージの履歴を表示する方法については、メッセージ履歴の表示 を参照してください。

# 5.4.4.1.6. GUI の終了

[File] メニュー から **[Exit]** を選択すると、すべてのサーバから切断され、GUI のウィンドウが閉じます。

# 5.4.4.1.7. 共通の作業

以下に、すべてのユーザが実行できる基本作業を示します。

---


LifeKeeper の起動

LifeKeeper の停止

LifeKeeper プロセスの表示

LifeKeeper GUI サーバプロセスの表示

LifeKeeper の制御プロセスの表示

サーバのクラスタへの接続

クラスタからの切断

接続サーバの表示

サーバのステータスの表示

サーバプロパティの表示

サーバログファイルの表示

リソースのタグと ID の表示

リソースのステータスの表示

リソースのプロパティの表示

Resource Labels

メッセージ履歴の表示

リソース階層ツリーの展開および折り畳み

[Cluster Connect] ダイアログ

[Cluster Disconnect] ダイアログ

[Resource Properties] ダイアログ

[Server Properties] ダイアログ

# 5.4.4.1.7.1. LifeKeeper の起動

SPS ソフトウェアはすべて、 */opt/LifeKeeper* ディレクトリにインストールされます。

すべての確認作業 が完了すると、両方のサーバで LifeKeeper を起動する準備が整います。このセクションでは、LifeKeeper サーバデーモンプロセスの起動について説明します。LifeKeeper GUI アプリケーションは、別個のコマンドを使用して起動され、LifeKeeper GUI の設定 に説明されています。LifeKeeper には、LifeKeeper デーモンプロセスの起動と停止を行うコマンドラインインターフェース が用意されています。これらのデーモンプロセスは、LifeKeeper GUI を起動する前に実行する必要があります。

## LifeKeeper サーバプロセスの起動

LifeKeeper がシステムで現在実行されていない場合は、すべてのサーバで root ユーザとして次のコマンドを入力してください。

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

このコマンドを実行すると LifeKeeper サービスが開始され、LifeKeeper はシステムの起動時に自動的に開始するように設定されます。
数秒の遅延の後、情報メッセージが表示されます。

> 注記: LifeKeeper を起動するときに **LifeKeeper Distribution Enabling Package** を参照するエラーメッセージが表示された場合は、LifeKeeper インストールイメージファイル をインストール/再インストールする必要があります。

lkstartコマンドの詳細については、コマンドラインに man LCD と入力して、LCD ヘルプページを参照してください。

また、自動起動を有効にせずにLifeKeeperプロセスのみを起動させる場合には、以下のコマンドを実行してください。

```
service lifekeeper start (または, systemctl start lifekeeper)
```

## LifeKeeper の自動起動の有効化

上述のコマンドで LifeKeeper が起動しますが、システムを再起動するたびに毎回同じコマンドを実行する必要があります。サーバの起動時に LifeKeeper が自動的に起動するようにするには、次のコマンドを入力してください。

```
chkconfig lifekeeper on (または, systemctl enable lifekeeper)
```

詳細については、chkconfig マニュアルページを参照してください。

# 5.4.4.1.7.2. LifeKeeper の停止

LifeKeeper を停止する必要がある場合は、root ユーザーで次のコマンドを入力して停止してください。

> 注記: リソースの停止が失敗した場合でも `lkstop` の **プロセスは続行** され、lkstopが失敗することは **ありません** 。

- **`/opt/LifeKeeper/bin/lkstop`**
  このコマンドを実行すると、現在実行されている場合ローカルシステムの LifeKeeper はシャットダウンされます。まず保護されているすべてのリソースをローカルシステムのサービスから停止し、次に LifeKeeper デーモンをシャットダウンします。

- **`/opt/LifeKeeper/bin/lkstop -f`**
  このコマンドは、サービスからリソースを停止するセクションをスキップします。リソースは引き続きローカルシステムで実行されますが、LifeKeeper には保護されなくなります。

## LifeKeeper の自動起動の無効化

システムの再起動時に LifeKeeper が自動で再起動しないようにするには、次のコマンドを入力してください。

```
chkconfig lifekeeper off
```

*または*

```
systemctl disable lifekeeper
```

詳細については、chkconfig （または systemctl）マニュアルページを参照してください。

# 5.4.4.1.7.3. LifeKeeper プロセスの表示

現在実行されているすべての LifeKeeper Core デーモンプロセスのリストを表示するには、次のコマンドを実行してください。

```
ps -ef | grep LifeKeeper | grep -w bin | grep -v lklogmsg
```

出力の例を以下に示します。

```
root 11663 11662 0 14:03 pts/0 00:00:00 /bin/bash /etc/redhat-lsb/
lsb_start_daemon /opt/LifeKeeper/sbin/runsvdir -P /opt/LifeKeeper/etc/
service log: runit just
started.....

root 11666 11663 0 14:03 pts/0 00:00:00 /bin/bash -c ulimit -S -c 0
>/dev/null 2> &1 ; /opt/LifeKeeper/sbin/runsvdir -P /opt/LifeKeeper/etc/
service log: runit just
started.....

root 11880 11873 0 14:03 ? 00:00:00 /opt/LifeKeeper/bin/lk_logmgr -l/opt/
LifeKeeper/out -d/etc/default/LifeKeeper

root 12240 11877 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lcm

root 12247 11879 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/ttymonlcm

root 12250 11876 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lcd

root 12307 11874 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lkcheck

root 12311 11875 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lkscsid

root 12325 11871 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/lkvmhad

root 12335 12330 0 14:04 ? 00:00:00 /opt/LifeKeeper/bin/perl /opt/
LifeKeeper/htdoc/cgi-bin/DoRequest.fcgi
```

LifeKeeper の実行状態は次のコマンドで確認できます。

```
/opt/LifeKeeper/bin/lktest
```

LifeKeeper が実行中の場合は次のように出力されます。

```
F S UID PID PPID C CLS PRI NI SZ STIME TIME CMD

4 S root 12240 11877 0 TS 39 -20 6209 14:04 00:00:00 lcm
```

```
4 S root 12247 11879 0 TS 39 -20 30643 14:04 00:00:00 ttymonlcm

4 S root 12250 11876 0 TS 29 -10 9575 14:04 00:00:00 lcd
```

LifeKeeper が実行中でない場合は何も出力されず、コマンドは 1 で終了します。

注記：グラフィカルユーザインターフェース (GUI) に必要なプロセスとともに、LifeKeeper Core デーモンを開始、停止、および監視する別の LifeKeeper プロセスも実行されています。プロセスのリストについては、LifeKeeper 制御プロセスの表示 および LifeKeeper GUI サーバプロセスの表示 を参照してください。また、ほとんどの LifeKeeper プロセスには子プロセス lklogmsg があり、予期しない出力をキャプチャして記録します。

# 5.4.4.1.7.4. LifeKeeper GUI サーバプロセスの表示

LifeKeeper GUI サーバが動作していることを確認するには、以下のコマンドを入力してください。

```
ps -ef | grep runGuiSer
```

以下のような出力が表示されます。

```
root 2805 1 0 8:24 ?00:00:00 sh /opt/LifeKeeper/bin/runGuiServer
```

現在動作している他の GUI サーバのデーモンプロセスのリストを表示するには、以下のコマンドを入力してください。

```
ps -efw | grep S_LK
```

以下のような出力が表示されます。

```
root 819 764 0 Oct16 ? 00:00:00 java -Xint -Xss3M -DS_LK=true
-Djava.rmi.server.hostname=wake -Dcom.steeleye.LifeKeeper.rmiPort=82
-Dcom.steeleye.LifeKeeper.LKROOT=/opt/LifeKeeper
-DGUI_RMI_REGISTRY=internal -DGUI_WEB_PORT=81
com.steeleye.LifeKeeper.beans.S_LK
```

LifeKeeper GUI サーバ管理 Web サーバが動作していることを確認するには、以下のコマンドを入力してください。

```
ps -ef|grep steeleye-light | egrep -v "lklogmsg|runsv"
```

以下のような出力が表示されます。

```
root 12330 11872 0 14:04 ? 00:00:00 /opt/LifeKeeper/sbin/steeleye-
lighttpd -D -f/opt/LifeKeeper/etc/lighttpd/lighttpd.conf
```

# 5.4.4.1.7.5. LifeKeeper の制御プロセスの表示

LifeKeeper の制御プロセスが動作していることを確認するには、以下のコマンドを入力してください。

```
ps -ef | grep runsv
```

以下のような出力が表示されます。

```
root 29093 1 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsvdir -P /opt/
LifeKeeper/etc/service

root 29097 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lcd

root 29098 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lcm

root 29099 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lk_logmgr

root 29100 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lkcheck

root 29101 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lkscsid

root 29102 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv lkvmhad

root 29103 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv steeleye-
lighttpd

root 29104 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv ttymonlcm

root 29105 29093 0 11:35 ? 00:00:00 /opt/LifeKeeper/sbin/runsv
lkguiserver

root 29465 2894 0 11:36 pts/0 00:00:00 grep --color=auto runsv
```

これらのプロセスは LifeKeeper Core のデーモンプロセスの開始、停止、および監視を行います。LifeKeeper を開始するにはこれらのプロセスが動作している必要があります。これらのプロセスは、デフォルトでシステムの起動時に開始するように設定されています。この動作は変更しないでください。

# 5.4.4.1.7.6. サーバのクラスタへの接続

1. 開始するには、以下の 2 つの方法があります。

- ◦ グローバルツールバー の **[Connect]** ボタンをクリックする。
- ◦ [File] メニュー の **[Connect]** をクリックする。

2. [Cluster Connect] ダイアログ の **[Server Name]** フィールドに、接続するクラスタ内のサーバ名を入力してください。

> 注記: **IPv6** アドレスを使用する場合は、このアドレスを大かっこ [ ] で囲む必要があります。これにより、マシンの IPv6 アドレス経由で接続を確立できます。別の方法として、名前をアドレスに割り当てることができ、その名前を使用して接続できます。

3. **[Login]** と **[Password]** のフィールドに、指定のサーバ上で LifeKeeper が認証に使用するユーザのログイン名とパスワードを入力してください。

4. **[OK]** をクリックしてください。

GUI が正常に指定サーバに接続した場合、GUI は、新しいサーバが検出されなくなるまで、クラスタ内にあるすべての既知のサーバへの接続 (およびステータス表示への追加) を継続します。

> 注記 : 最初のログイン名とパスワードが、クラスタ内のサーバ上にあるクライアントで認証に失敗した場合、そのサーバでの別のログイン名とパスワードを入力するように要求されます。[Password] ダイアログ で **[Cancel]** を選択した場合、サーバへの接続は中止され、GUI はクラスタ内の残りのコンポーネントへの接続を継続します。

# 5.4.4.1.7.7. クラスタからの切断

この作業は、選択したサーバ経由で、GUI クライアントをクラスタ内のすべてのサーバから切断します。

1. 開始するには、以下の 3 つの方法があります。
   - ◦ グローバルツールバー の **[ Disconnect ]** ボタンをクリックする。
   - ◦ [Edit] メニュー の **[Server]** を選択し、 **[Disconnect]** をクリックする。
   - ◦ サーバのコンテキストツールバー が表示される場合は、そこにある **[Disconnect]** ボタンをクリックする。
2. [Cluster Disconnect] ダイアログ の **[Select Server in Cluster]** リストから、切断するクラスタ内のサーバ名を選択してください。
3. **[OK]** をクリックしてください。クラスタ内の全サーバのリストを持つ **[Confirmation]** ダイアログが表示されます。
4. **[Confirmation]** ダイアログの **[OK]** をクリックして、クラスタ内の全サーバからの切断を確定してください。

クラスタからの切断後、そのクラスタ内にあるすべてのサーバが、GUI のステータス表示から消去されます。

# 5.4.4.1.7.8. 接続サーバの表示

サーバの状態は、下図に示すように、表内のサーバのグラフィック表示で表されます。サーバアイコンが視覚的に示すサーバの状態の詳細については、サーバの状態の表示 を参照してください。

# 5.4.4.1.7.9. サーバのステータスの表示

サーバの状態は、下図に示すように、表内のサーバのグラフィック表示で表されます。

| サーバの状態 | 状態のシンボル | 意味 |
|---|---|---|
| ALIVE | | クライアントはサーバに有効な接続を行うことができます。<br><br>このサーバから ALIVE のリモートサーバへのコミュニケーションパスが ALIVE です。<br><br>DEAD とマークされたコミュニケーションパス、および DEAD のサーバをターゲットとするコミュニケーションパスは無視されます。これは、DEAD のサーバには DEAD のグラフィックで表されるからです。 |
| ALIVE | | クライアントはサーバに有効な接続を行うことができます。<br><br>このサーバから指定リモートサーバへの 1 つ以上のコミュニケーションパスが DEAD です。<br><br>このサーバから指定リモートサーバへの間には、冗長コミュニケーションパスが存在しません。 |
| DEAD | | クラスタ内の他のサーバから DEAD として報告されました。 |
| UNKNOWN | | ネットワーク接続が失われました。最後に分かっている LifeKeeper の状態が ALIVE です。 |

# 5.4.4.1.7.10. サーバのプロパティの表示

1. 開始するには、以下の 2 つの方法があります。
   - プロパティを表示するサーバのアイコンを右クリックします。サーバのコンテキストメニューが表示されたら、 **[Properties]** をクリックします。サーバのプロパティは、サーバをクリックすると プロパティパネル (有効になっている場合) にも表示されます。
   - [Edit] メニュー の **[Server]** をポイントし、 **[Properties]** をクリックします。ダイアログが表示されたら、表示するサーバを **[Server]** リストから選択します。

2. 別のサーバのプロパティを表示する場合は、サーバを **[Server]** リストから選択してください。

3. 確認が完了したら、 **[OK]** をクリックしてウィンドウを閉じてください。

# 5.4.4.1.7.11. サーバのログファイルの表示

1. 開始するには、以下の 4 つの方法があります。

   - ◦ サーバのアイコンを右クリックしてサーバのコンテキストメニュー を表示し、次に **[View Log]** をクリックして [LifeKeeper Log Viewer] ダイアログを表示する。
   - ◦ グローバルツールバー の **[View Log]** ボタンをクリックし、[LifeKeeper Log Viewer] ダイアログの [Server] リストから、表示するサーバを選択する。
   - ◦ サーバのコンテキストツールバー が表示される場合は、その **[View Log]** ボタンをクリックする。
   - ◦ [Edit] メニュー の **[Server]** をポイントし、 **[View Log]** をクリックする。次に、**[LifeKeeper Log Viewer]** ダイアログの [Server] リストから、表示するサーバを選択する。

2. グローバルツールバー または **[Edit]** メニュー から捜査を開始して、別のサーバのログを表示する場合は、[LifeKeeper Log Viewer] ダイアログの **[Server]** リストから、そのサーバを選択します。 **サーバのコンテキストメニュー** または **サーバのコンテキストツールバー** から **[View Log]** を選択した場合は、この機能は使用できません。

3. 確認が完了したら、 **[OK]** をクリックして **[Log Viewer]** ダイアログを閉じてください。

# 5.4.4.1.7.12. リソースのタグと ID の表示

リソースのタグと ID を即座に表示するには、[Status] ウィンドウのリソースアイコンにカーソルを合わせて、マウスの左ボタンを 1 回押します (シングルクリック)。優先順位が最も低いサーバのリソースのタグと ID がメッセージバーに表示されます。特定サーバ上にあるリソースのタグと ID を表示する場合は、表内のリソースインスタンスセルを左クリックしてください。

メッセージバーに表示されるメッセージは、以下のようになります。

```
Resource Tag = ipdnet0-153.98.87.73, Resource ID = IP-153.98.87.73
```

特定の状況では、GUI がリソース ID を特定できないことがあります。この場合は、リソースタグのみがメッセージバーに表示されます。

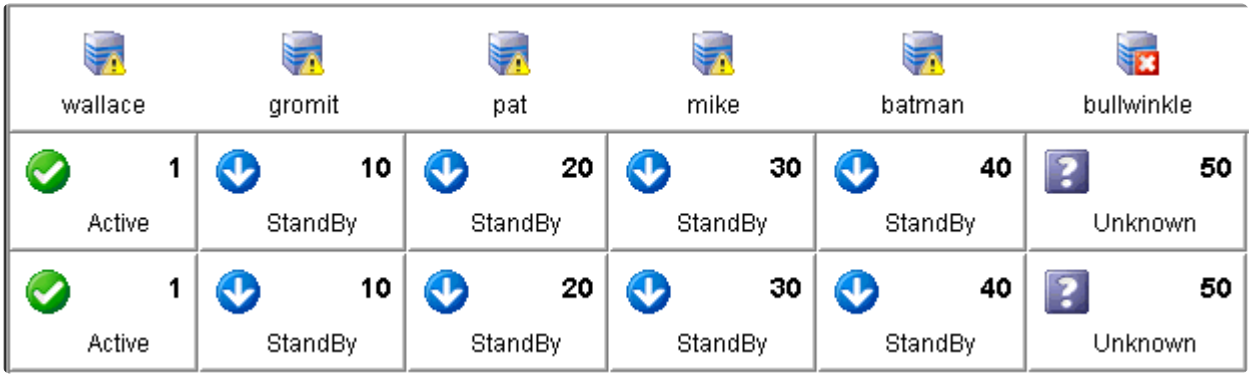
# 5.4.4.1.7.13. リソースのステータスの表示

リソースのステータスつまり状態は、 **グローバルリソースのステータス** (すべてのサーバについて) と **サーバリソースのステータス** (1 台のサーバ上) の 2 つの形式で表示されます。グローバルリソースのステータスは、 **[Status]** ウィンドウの左ペインにある **リソース階層ツリー** に表示されます。サーバリソースのステータスは、リソースの列とサーバ行の交点にある表のセルにあります。

## サーバリソースのステータス

下図に、アクティブ、スタンバイ、および不明のリソースステータスを持つサーバを示します。

- 「wallace」上のリソースはすべて、アクティブです。
- 「gromit」、「pat」、「mike」、および「batman」上のリソースはすべてスタンバイです。
- 「bullwinkle」上のリソースはすべて、不明です。




| サーバリソースの状態 | 状態のシンボル | 意味 |
|---|---|---|
| アクティブ | | このサーバ上でリソースは動作可能であり、保護されています。 (ISP) |
| 可用性の低下 | | このサーバ上でリソースは動作可能ですが、バックアップリソースによる保護はされていません。 (ISU) |
| スタンバイ | | サーバは、リソースの動作を引き継ぐことができます。(OSU) |
| 障害 | | このサーバ上のリソースに問題が検出されました。例えば、リソースを In Service にする試行が失敗しました。(OSF) |
| 不明 | | リソースが初期化されていないか (ILLSTATE)、このサーバで LifeKeeper が動作中でありません。 |
| | 空のパネル | サーバのリソースが定義されていません。 |

# グローバルリソースのステータス

| 状態のシンボル | 説明 | 意味 / 原因 |
|---|---|---|
| | 正常 | リソースがアクティブ (ISP) で、バックアップがアクティブです。 |
| | 警告 | リソースがアクティブ (ISP) です。1 つ以上のバックアップが、不明または障害 (OSF) としてマークされています。 |
| | 障害。リソースが、いずれのサーバでもアクティブでありません (OSF)。 | リソースが、通常の原因により Out of Service になりました。<br><br>リソースが、通常ではない方法により動作が停止しました。<br><br>リカバリは完了していないか、失敗しました。 |
| | 不明。利用可能な情報からは、状態を特定できませんでした。 | 複数のサーバがアクティブであることを告げています。<br><br>サーバへの接続が遮断されました。<br><br>サーバのリソースインスタンスがすべて、不明の状態です。 |

# 5.4.4.1.7.14. リソースのプロパティの表示

1. 開始するには、以下の 3 つの方法があります。

   - º プロパティを表示するリソース / サーバの組み合わせのアイコンを右クリックします。

リソースのコンテキストメニュー が表示されたら、[ **Properties** ]をクリックします。リソースのプロパティは、プロパティパネル (有効になっている場合) にも表示されます。

- º プロパティを表示するグローバルリソースのアイコンを右クリックします。

リソースのコンテキストメニュー が表示されたら、 **[Properties]** をクリックします。ダイアログが表示されたら、表示するリソースが存在するサーバを **[Server]** リストから選択します。

- º

[Edit] メニュー の **[ Resource ]** をポイントし、**[ Properties ]** をクリックします。ダイアログが表示されたら、プロパティを表示するリソースを **[Resource]** リストから選択し、表示するリソースが存在するサーバを **[Server]** リストから選択します。

2. 別のリソースのプロパティを表示する場合は、リソースを **[Resource]** リストから選択してください。

3. 別のサーバ上にあるリソースのプロパティを表示する場合は、サーバを **[Server]** リストから選択してください。

4. 確認を終了したら、 **[OK]** をクリックしてウィンドウを閉じてください。

# 5.4.4.1.7.15. Resource Labels

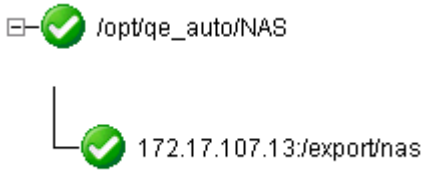
## Resource Labels

このオプショングループを使用すると、リソース階層ツリー内のリソースを、タグ名別 と ID 別のいずれで表示するかを指定できます。

注記：リソース階層ツリーに表示されるリソースタグ /ID は、優先順位が最も低い番号を持つサーバに属します。特定サーバ上にあるリソースのタグ /ID を表示する場合は、表内のリソースインスタンスセルを左クリックしてください。メッセージバーにそのタグ /ID が表示されます。

**By tag name:**

**By ID:**



# 5.4.4.1.7.16. メッセージ履歴の表示

1. [View] メニュー の [ **History** ] をクリックしてください。LifeKeeper GUI の [Message History] ダイアログが表示されます。

2. 履歴のメッセージをすべて消去する場合は、 **[Clear]** をクリックしてください。

3. ダイアログを閉じるには、 **[OK]** をクリックしてください。

**[Message History]** ダイアログには、メッセージバーからの最新のメッセージが表示されます。履歴リストには、最大 1000 行を表示できます。最大行数を超えた場合、新しいメッセージにより最も古いメッセージが「押し出され」ます。

これらのメッセージは、クライアントとサーバとの間の動作のみを表し、時系列で表示されます。最新のメッセージがリストの上部に表示されます。

## メッセージ履歴の解釈

**<—** は、メッセージがサーバから受信したことを示し、通常は以下の形式をとります。

```
<--"server name":"action"

<--"server name":"app res": "action"

<--"server name":"res instance":"action"
```

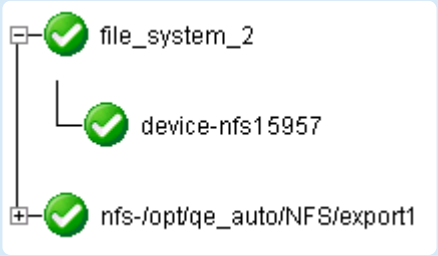
**—>** はメッセージがクライアントから送信されたことを示し、通常は以下の形式をとります。

```
-->"server name":"action"

-->"server name":"app res": "action"

-->"server name":"res instance":"action"
```

**[Clear]** ボタンをクリックすると、履歴が消去されますが、ダイアログは閉じません。

**[OK]** ボタンをクリックすると、履歴を消去せずにダイアログが閉じます。

# 5.4.4.1.7.17. リソース階層ツリーの展開と折り畳み

| | |
|---|---|
|  | このツリーのセグメントでは、リソース *file_system_2* が展開されており、リソース *nfs-/opt/qe_auto/NFS/export1* が折り畳まれています。<br><br>展開されているリソースアイコンの左には、⊟ が表示されます。<br><br>折り畳まれているリソースアイコンの左には、⊞ が表示されます。 |

リソース階層ツリーを **展開** するには、

- ◦ ⊞ をクリックするか、
- ◦ ⊞ の右側にあるリソースアイコンをダブルクリックしてください。

リソース階層ツリーを **すべて展開** するには、

- ◦ **[ View ]** メニュー の **[ Expand Tree ]** をクリックするか、
- ◦ **[Status]** ウィンドウの左ペインにある列ヘッダの **[Resource Hierarchy Tree]** ボタンをダブルクリックしてください。

> **注記:** リソース階層ツリーに表示されるリソースタグ /ID は、優先順位が最も低い番号を持つサーバに属します。特定サーバ上にあるリソースのタグ /ID を表示する場合は、表内のリソースインスタンスセルを左クリックしてください。メッセージバーにそのタグ /ID が表示されます。

リソース階層ツリーを折り畳むには、

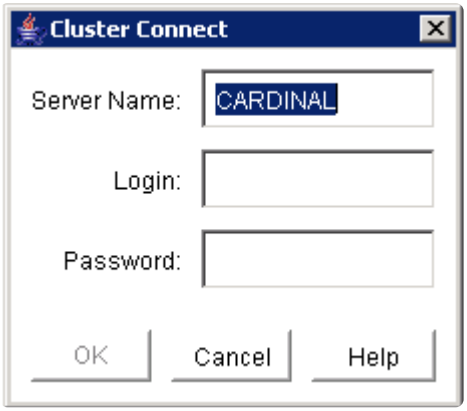
- ◦ ⊟ をクリックするか、
- ◦ ⊟ の右側にあるリソースアイコンをダブルクリックしてください。

リソース階層ツリーをすべて折り畳むには、

- ◦ **[View]** メニュー の **[Collapse Tree]** をクリックするか、
- ◦ [Status] ウィンドウの左ペインにある列ヘッダの **[Resource Hierarchy Tree]** ボタンをダブルクリックしてください。

**注記:**「9」と「0」のキーは、すべてのリソース階層ツリーに対して即座に展開 / 折り畳みを実行するホットキー / アクセラレータキーとして指定されています。

# 5.4.4.1.7.18. [Cluster Connect] ダイアログ

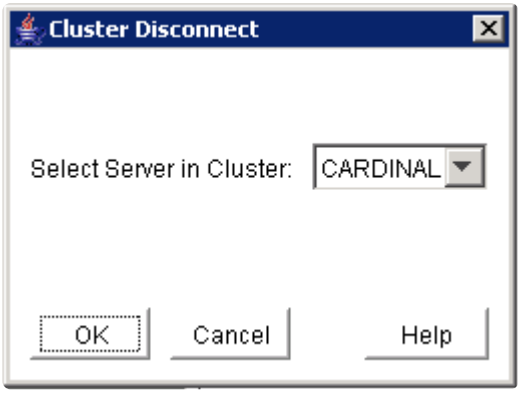



**Server Name** - 接続先のサーバ名。

**Login** - 接続先のサーバに LifeKeeper 認証情報を持つユーザのログイン名。

**Password** - 接続先のサーバで指定ログインを認証するパスワード。

# 5.4.4.1.7.19. [Cluster Disconnect] ダイアログ




**Select Server in Cluster** -

接続しているサーバの名前がドロップダウンリストボックスに表示されます。リストから、切断するクラスタのサーバを選択してください。切断されるクラスタ内のすべてのサーバが、確認ダイアログに表示されます。

# 5.4.4.1.7.20. [Resource Properties] ダイアログ

[Resource Properties] ダイアログは、[Edit] メニュー やリソースコンテキストメニュー から使用できます。このダイアログには、サーバ上にある特定のリソースのプロパティが表示されます。[Edit] メニューからアクセスした場合は、リソースとサーバを選択できます。リソースコンテキストメニューからアクセスした場合、はサーバを選択できます。

## [General] タブ

- ◦ Tag - リソースインスタンスの名前。システムに対して一意で、管理者にリソースを示します。
- ◦ ID - リソースインスタンスに関連する文字列であり、リソースタイプのすべてのインスタンス間で一意です。関連するアプリケーションソフトウェアに対して、リソースインスタンスの内部特性のいくつかを示します。
- ◦ Switchback (管理者権限を持つユーザは編集可能) - サービス起動中のリソースが存在するサーバに障害が発生した場合に、サーバのリカバリ動作を管理する設定。この設定が [Intelligent] の場合、指定リソースの可能なバックアップとしてサーバが動作します。この設定が [Automatic] の場合、サーバはアクティブにリソースの再取得を試行します (以下の条件が満たされる場合)。
  - º サーバがクラスタから離れるときには、リソース階層のサービスが既に起動している必要があります。
  - º リソース階層がすべてサービス起動している場合は、低プライオリティのサーバでサービスを起動している必要があります。

  **注記:** 自動スイッチバックのチェックは、LifeKeeper を起動したとき、またはクラスタに新しいサーバを追加したときにのみ実行されます。通常のクラスタ動作中には実行されません。
- ◦ State -リソースインスタンスの現在の状態。
  - º

*Active* - ローカルで in service であり、保護されています。
- º

*Warning* - ローカルで in service ですが、ローカルリカバリは試行されません。
- º

*Failed* - out of service、障害。
- º

*Standby* - サービス停止、障害なし。

- ○

*LLSTATE* - LifeKeeper の起動シーケンスの一部として実行されるリソース初期化プロセスにより適切に初期化されていません。この状態のリソースは、LifeKeeper で保護されていません。

- ○

*UNKNOWN* - リソースの状態を特定できませんでした。GUI サーバが使用できない可能性があります。

- ○ Reason -存在する場合、リソースが現在の状態にある原因 (つまり、最後の状態変化の原因) を示します。例えば、galahad 上にあるアプリケーションの状態が OSU である原因は、tristan 上にある共有プライマリリソース *ordbfsaa-on-tristan* の状態が ISP か ISU であることです。共有リソースは、グループ内の 1 つのシステムでのみ同時にアクティブにできます。

- ○ Initialization - 起動時のリソースの初期化動作を決定する設定であり、AUTORES_ISP、INIT_ISP、INIT_OSU などがあります。

## [Relations] タブ

- ○ Parent - このリソースに直接依存するリソースのタグ名を示します。

- ○ Child - このリソースが依存するすべてのリソースのタグ名を示します。

- ○ Root - このリソース階層で、親を持たないリソースのタグ名。

## [Equivalencies] タブ

- ○ Server - リソースが定義済みの同等性を持つサーバ名。

- ○ Priority (管理者権限を持つユーザは編集可能) - このリソースについて、ターゲットサーバのフェイルオーバの優先順位の値。

- ○ Tag -同等のサーバ上にあるこのリソースのタグ名。

- ○ Type -同等性のタイプ (SHARED、COMMON、COMPOSITE)。

- ○ Reorder Priorities (管理者権限を持つユーザは編集可能) - [Up]/[Down] ボタンを使用して、選択した同等リソースの優先順位を並べ替えることができます。

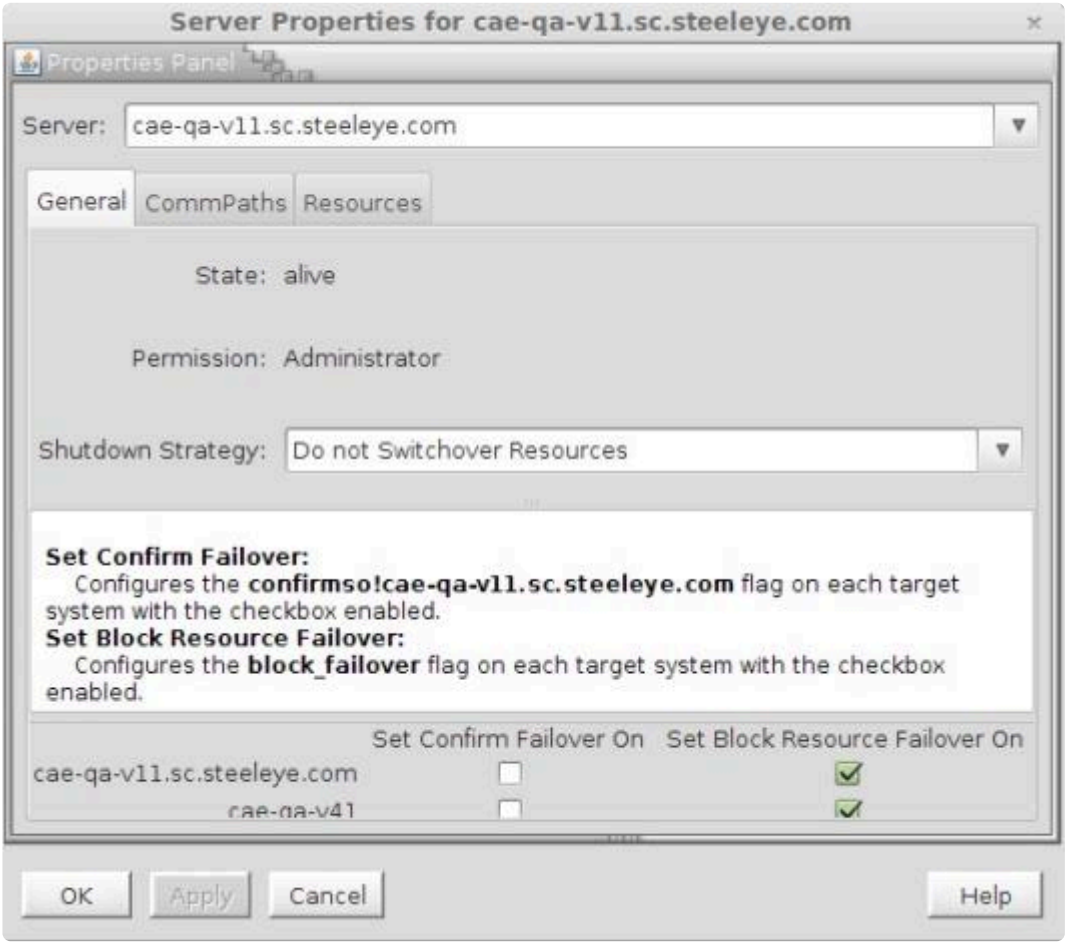
[OK] ボタンをクリックすると、変更内容が適用されてウィンドウが閉じます。[Apply] ボタンをクリックすると、変更内容が適用されます。[Cancel] ボタンをクリックすると、最後に [Apply] をクリックして以降の変更内容を保存せずに、ウィンドウが閉じます。

# 5.4.4.1.7.21. [Server Properties] ダイアログ

[Server Properties] ダイアログは、サーバのコンテキストメニューや[Edit] メニュー から使用できます。このダイアログには、特定のサーバのプロパティが表示されます。サーバのプロパティは、プロパティパネル (有効になっている場合) にも表示されます。

このダイアログの 3 つのタブについて説明します。[OK] ボタンをクリックすると、変更内容が適用されてウィンドウが閉じます。[Apply] ボタンをクリックすると、変更内容が適用されます。[Cancel] ボタンをクリックすると、最後に [Apply] をクリックして以降の変更内容を保存せずに、ウィンドウが閉じます。

## [General] タブ




- **Name** - 選択したサーバの名前。
- **State** -サーバの現在の状態。 サーバの状態は以下の値をとります。
  - *ALIVE* - サーバが使用可能。
  - *DEAD* - サーバが使用不可。
  - *UNKNOWN* - リソースの状態を特定できませんでした。GUI サーバが使用できない可能性があります。
- **Permission** - そのサーバに現在ログインしているユーザの権限レベル。権限は以下の値をとります。
  - *Administrator* - ユーザは LifeKeeper のすべての作業を実行できます。
  - *Operator* - ユーザは、LifeKeeperのリソースとサーバのステータスを監視でき、リソースのサービス起動やサービス停止ができます。

    - *Guest* - ユーザは LifeKeeper のリソースとサーバのステータスを監視できます。
- **Shutdown Strategy** (管理者権限を持つユーザは編集可能) - サーバがシャットダウンしたときに、リソースがクラスタ内のバックアップサーバにスイッチオーバするかどうかを制御する設定。設定「*Switchover Resources*」は、リソースがクラスタ内のバックアップサーバでサービスが起動することを示します。設定「*Do not Switchover Resources*」は、リソースがクラスタ内にある別のサーバでサービスが起動しないことを示します。
- **Failover Strategy** - この設定を使用して、LifeKeeper のクラスタ内にある特定システムからのフェイルオーバをユーザに確定するよう要求できます。この設定は、LifeKeeper の管理者のみが使用できます。オペレータとゲストには、この設定は表示されません。デフォルトでは、フェイルオーバはすべて、ユーザの操作を必要とせず自動実行されます。ただし、confirm failover フラグが設定されると、指定システムからフェイルオーバするには、以下のコマンドを実行して確定することが必要です。`lk_confirmso -y system` 。以下のコマンドを実行して、フェイルオーバをブロックできます。`lk_confirmso -n system` 。指定期間内にこれらのコマンドのいずれかが実行されない限り、システムは事前プログラミングされたデフォルト動作を実行します。 `/etc/default/LifeKeeper` ファイル内にある 2 つのフラグが、この自動動作を制御します。

        - º

CONFIRMSODEF

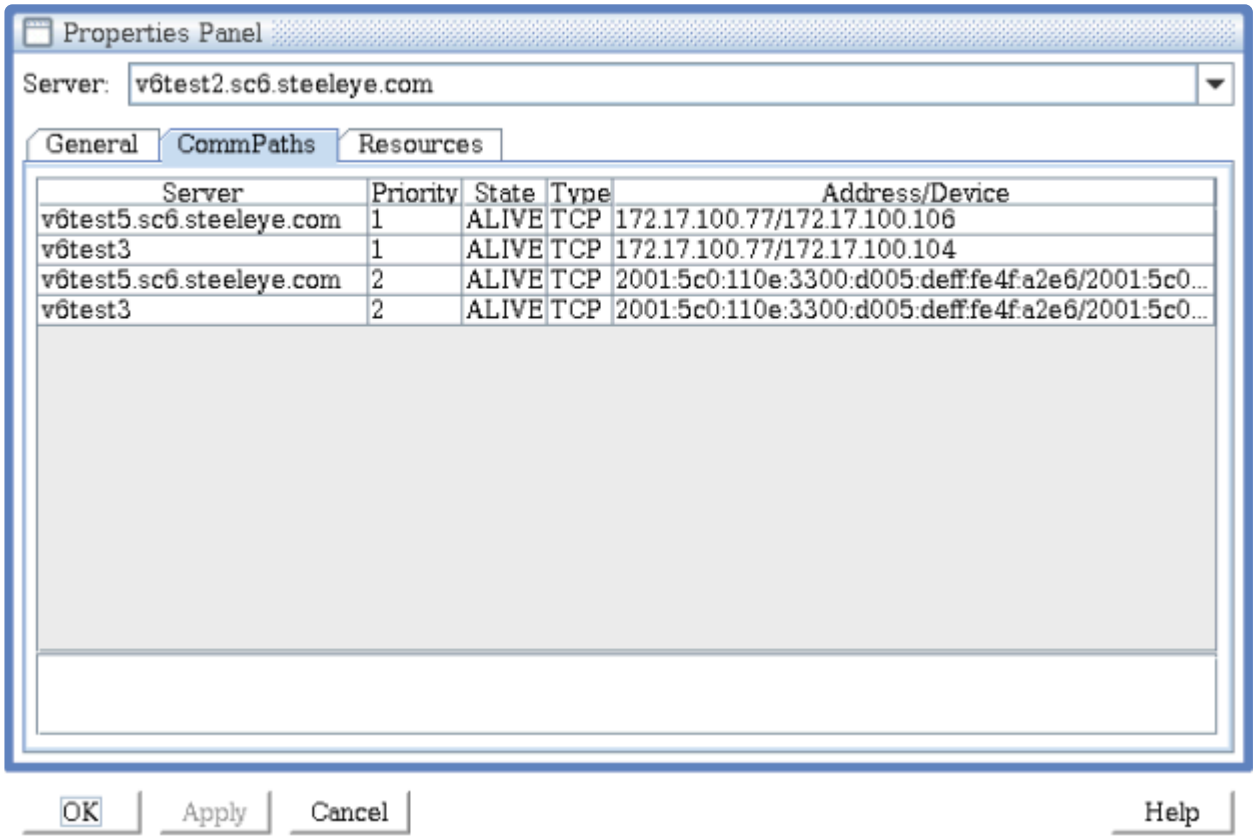
            これは、デフォルト動作を指定します。「0」に設定されている場合、デフォルト動作はフェイルオーバを実行します。「*1*」に設定されている場合、デフォルト動作はフェイルオーバをブロックします。

        - º

CONFIRMSOTO

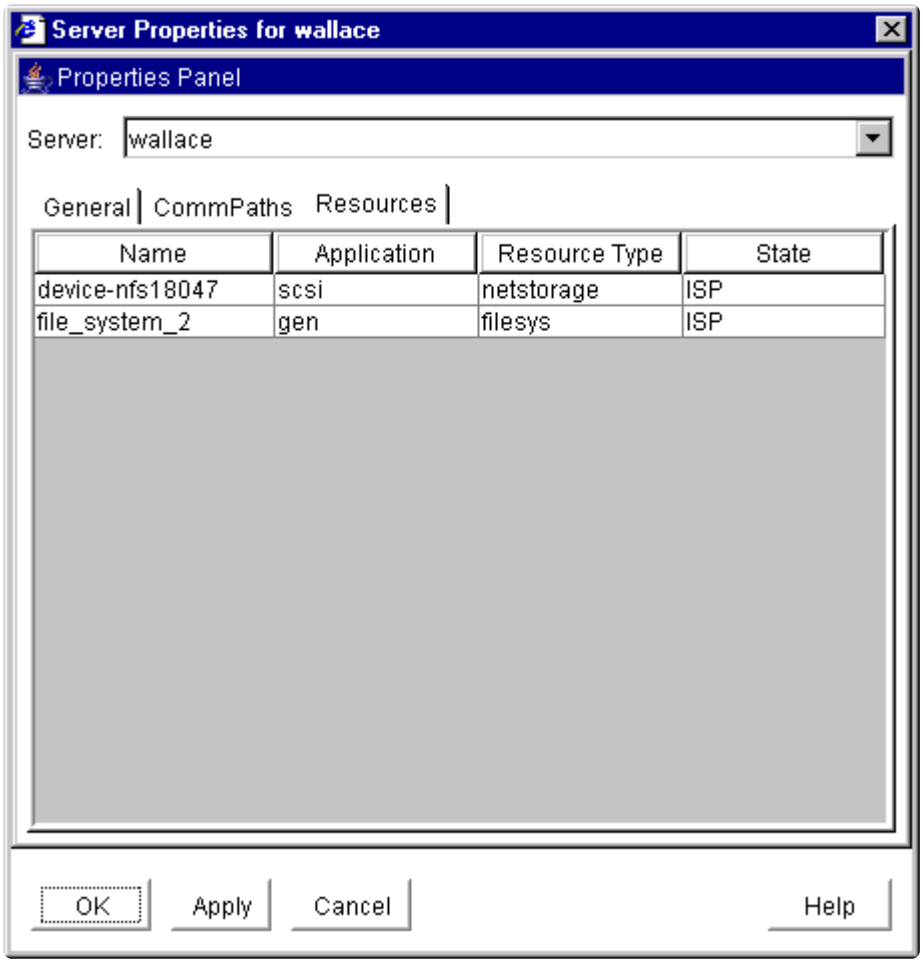
            これは秒単位で設定され、デフォルト動作を実行する前に LifeKeeper が待機する時間を示します。

# [CommPaths] タブ




- **Server** -LifeKeeper のクラスタ内で、コミュニケーションパスが接続している他のサーバのサーバ名。
- **Priority** - 2 台のサーバ間でコミュニケーションパスを使用する順序を定義する優先順位。1 が最高の優先順位で、99 が最低の優先順位です。
- **State** -LifeKeeper の設定データベース (LCD) のコミュニケーションパスの状態。コミュニケーションパスの状態は以下の値をとります。
  - *ALIVE* - 通常の動作をしています。
  - *DEAD* - 通常の動作をしていません。
  - *UNKNOWN* - 状態を特定できませんでした。GUI サーバが使用できない可能性があります。
- **Type -** リスト内のサーバと、[Server] フィールドに指定されたサーバとの間のコミュニケーションパスの種類。TCP (TCP/IP)、または TTY。
- **Address/Device** - コミュニケーションパスが使用する IP アドレスまたはデバイス名。
- **Comm Path Status** - LifeKeeper の設定データベース (LCD ) 内のコミュニケーションパスの状態に基づいて、GUI が判定したコミュニケーションパスのステータスの概要。以下に、コミュニケーションパスのステータスの値を示します。これらの値は、下のパネルの詳細テキストの下に表示されます。
  - *NORMAL* - すべてのコミュニケーションパスが通常の動作をしています。
  - *FAILED* - 指定サーバに対するすべてのコミュニケーションパスが動作していません。
  - *UNKNOWN* - コミュニケーションパスのステータスを特定できませんでした。GUI サーバが使用できない可能性があります。
  - *WARNING* - 指定サーバに対する 1 つ以上のコミュニケーションパスが動作していません。
  - *DEGRADED* - 指定サーバに対する 1 つ以上の冗長コミュニケーションパスが動作していません。
  - *NONE DEFINED* - コミュニケーションパスが定義されていません。

# [Resources] タブ




- **Name** -選択したサーバ上にあるリソースインスタンスのタグ名。
- **Application** - リソースタイプのアプリケーション名 (gen、scsi など)。
- **Resource Type** - サービスを提供するリソースタイプ、ハードウェアのクラス、ソフトウェアのクラス、またはシステムのエンティティのクラス (app、filesys、nfs、device、disk など)。
- **State** -リソースインスタンスの現在の状態。
  - *ISP* — ローカルでサービス起動中であり、保護されています。
  - *ISU* - ローカルでサービス起動中ですが、ローカルリカバリは試行されません。
  - OSF - サービス停止、障害。
  - *OSU* - サービス停止、障害なし。
  - *LLSTATE* - LifeKeeper の起動シーケンスの一部として実行されるリソース初期化プロセスにより、リソースの状態が適切に初期化されていません。この状態のリソースは、LifeKeeper で保護されていません。
  - *UNKNOWN* - リソースの状態を特定できませんでした。GUI サーバが使用できない可能性があります。

# 5.4.4.1.8. オペレータの作業

以下のトピックは、オペレータの権限を必要とする高度な作業です。

---

リソースを In Service にする

リソースを Out of Service にする

# 5.4.4.1.8.1. リソースを In Service にする

注記: LifeKeeper は、リソースを階層の最下位から最上位へ順に In Service にします。1 つの階層内のすべてのリソースを In Service にする場合は、階層内の最上位（親）リソースを選択してください。

1. 開始するには、以下の 5 つの方法があります。

- ◦ in service にするリソース / サーバの組み合わせのアイコンを右クリックします。リソースのコンテキストメニュー が表示されたら、 **[In Service]** をクリックします。

- ◦ in service にするグローバルリソースのアイコンを右クリックします。 **リソースのコンテキストメニュー** が表示されたら、 **[In Service]** をクリックします。ダイアログが表示されたら、in service にするリソースが存在するサーバを **[Server]** リストから選択し、 **[Next]** をクリックします。

- ◦ グローバルツールバー の **[In Service]** ボタンをクリックします。ダイアログが表示されたら、in service にするリソースが存在するサーバを **[Server]** リストから選択し、 **[Next]** をクリックします。次のダイアログで、in service にするリソースを 1 つ以上 **[Resouce(s)]** リストから選択し、 **[Next]** をもう一度クリックします。

- ◦ リソースのコンテキストツールバー が 表示される場合は、その **[In Service]** ボタンをクリックします。

- ◦ [Edit] メニュー の **[Resource]** をポイントし、 **[In Service]** をクリックします。ダイアログが表示されたら、in service にするリソースがあるサーバを **[Server]** リストから選択し、 **[Next]** をクリックします。次のダイアログで、in service にするリソースを 1 つ以上 [Resouce(s)] リストから選択し、 **[Next]** をもう一度クリックします。

2. 選択したサーバとリソースを In Service にすることを示すダイアログボックスが表示されます。親リソースの in service にせずに依存する子リソースを in service にしようとする場合、このダイアログには警告も表示されます。 **[In Service]** をクリックして、依存する子リソースと共にリソースを in service にしてください。

3. 出力パネル が有効の場合は、ダイアログが閉じ、リソースを In Service にするコマンドの結果が **出力パネル** に表示されます。出力パネルが無効の場合は、これらの結果を表示するダイアログが表示されたままになり、結果がすべて表示されたら **[Done]** をクリックします。in service になった追加の依存 (子) リソースが、ダイアログまたは **出力パネル** に表示されます。

4. リソースを in service にする動作で発生したエラーは、in service にするリソースがあるサーバの LifeKeeper ログに記録されます。

# 5.4.4.1.8.2. リソースを Out of Service にする

注記: LifeKeeper は、リソースを階層の最上位のリソースから下位のリソースへ、順に Out of Service にします。階層全体（子リソースを持つ親）を Out of Service にしたい場合は、最下位レベルのリソースを Out of Service にしてください。

1. 開始するには、以下の 4 つの方法があります。

- ◦ Out of Service にするグローバルリソース、またはリソース / サーバの組み合わせのアイコンを右クリックします。リソースのコンテキストメニュー が表示されたら、 **[Out of Service]** をクリックします。

- ◦ グローバルツールバー の [Out of Service] ボタンをクリックします。[Out of Service] ダイアログが表示されたら、 Out of Service にするリソースを 1 つ以上 [Resouce(s)] リストから選択し、 **[Next]** をクリックします。

- ◦ リソースのコンテキストツールバー が表示される場合は、 **[Out of Service]** ボタンをクリックします。

- ◦ [Edit] メニュー の **[ Resource]** をポイントし、 **[Out of Service]** をクリックします。 **[Out of Service]** ダイアログが表示されたら、 Out of Service にするリソースを 1 つ以上 [Resouce(s)] リストから選択し、 **[Next]** をクリックします。

2. 選択したリソースが Out of Service になることを示す **[Out of Service]** ダイアログボックスが表示されます。親リソースを Out of Service にせずに依存する子リソースを Out of Service にしようとする場合、このダイアログには警告も表示されます。 **[Out of Service]** をクリックして、次のダイアログボックスに進みます。

3. 出力パネル が有効の場合は、ダイアログが閉じ、リソースを Out of Service にするコマンドの結果が出力パネルに表示されます。出力パネルが無効の場合は、これらの結果を表示するダイアログが表示されたままになり、結果がすべて表示されたら **[Done]** をクリックします。

4. リソースを Out of Service にする動作で発生したエラーは、 Out of Service にするリソースが存在するサーバの LifeKeeper ログに記録されます。

# 5.4.4.1.9. 高度な作業

LCD

LCM

LifeKeeper API for Monitoring

# 5.4.4.1.9.1. LCD

## LifeKeeper 設定データベース

LifeKeeper 設定データベース (LCD) は、LifeKeeper が既知のすべてのリソースタイプについて、オブジェクト指向のリソース階層情報を管理し、リカバリ方向の情報を保存します。データは共有メモリにキャッシュされ、ファイルに保存されるので、システムの再起動後もデータが保持されます。LCD には、リカバリが必要なリソースインスタンスについての状態の情報、および特定の詳細情報もあります。

LCD のディレクトリ構造、保存されるデータタイプ、使用できるリソースタイプ、およびアプリケーションスクリプトの使用の詳細については、以下の関連トピックを参照してください。

### 関連トピック

- LCDI のコマンド
- LCD の設定データ
- LCD のディレクトリ構造
- LCD のリソースタイプ
- LifeKeeper のフラグ
- リソースのサブディレクトリ
- /opt/LifeKeeper の LCD のディレクトリ構造

# 5.4.4.1.9.1.1. LCDI のコマンド

**注：以下の文章では、独自のリカバリキットを定義してリソースを作成する方法を案内しています。既存のリカバリキットを使用する場合は、GUI を使用してリソースを作成してください。**

LifeKeeper には、アプリケーションのリソース階層を定義するためのメカニズムが 2 つ用意されています。

- LifeKeeper の GUI
- LifeKeeper 設定データベースのインターフェース (LCDI) コマンド

LCDI は LifeKeeper が提供するインターフェースコマンドのセットで、使用するアプリケーションのニーズに合わせてリソース階層の設定の作成とカスタマイズができます。アプリケーションが複数のリソース (例: 2 つ以上のファイルシステム) に依存する場合、コマンドインターフェースを使用します。

コマンドの詳細については、LCDI のマニュアルページを参照してください。このトピックでは、開発シナリオを示し、GUI とコマンドの両方の機能を使用してリソース階層を作成できる方法を説明します。

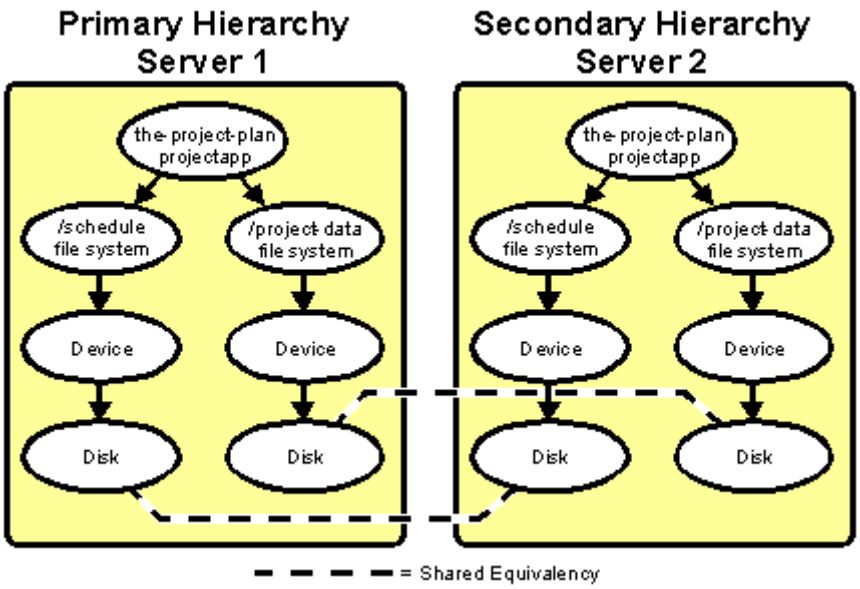
## シナリオの状況

アプリケーションの例である ProjectPlan は、サーバ 1 とサーバ 2 が共有する SCSI ファイルシステムにデータを保存しています。サーバ 1 が、アプリケーションのプライマリ階層にあります。アプリケーションには、 */project-data* と */schedule* の 2 つのファイルシステムがあります。階層定義の最初の手順では、依存関係を指定します。

このアプリケーションの例には、以下の依存関係があります。

- **共有ファイルシステム。** アプリケーションは、 */project-data* と */schedule* のファイルシステムに依存します。
- **SCSI ディスクサブシステム。** .次に、ファイルシステムは、SCSI ディスクサブシステム (デバイス、ディスク、およびホストアダプタのリソースを含む) に依存します。

結果として、階層を作成する作業は以下の図のようになります。




# 階層の定義

この例のアプリケーション階層を作成するために必要な作業を示します。

1. **ファイルシステムリソースの作成。** LifeKeeper の GUI には、ファイルシステムリソースを作成するメニューがあります。[ファイルシステムリソース階層の作成] を参照してください。

   この定義作業の最後で、LCD では 2 つのファイルシステムリソースが以下のように定義されます。

| ID | タグ | サーバ |
|---|---|---|
| /project-data | project-data-on-Server1 | Server1 |
| /project-data | project-data-from-Server1 | Server2 |
| /schedule | schedule-on-Server1 | Server1 |
| /schedule | schedule-from-Server1 | Server2 |

   **注記:** LifeKeeper で使用されるタグ名には意味はありません。単なるラベルです。表内のタグ名は LifeKeeper のデフォルト値です。

2. **リソースの定義。** この例では、以下の項目を定義する必要があります。

| | |
|---|---|
| アプリケーション: | projectapp |
| リソースタイプ: | plan |
| インスタンス ID: | 1yrplan |

| タグ: | the-project-plan |
|---|---|

**注記:** LifeKeeper の GUI を使用して定義の大部分を作成できますが、この例の以降ではコマンドインターフェースの操作を説明します。

3. **ディレクトリの作成。** 各システムで以下のコマンドを使用して、ディレクトリ */opt/LifeKeeper/subsys* の下に必要なアプリケーションリカバリディレクトリを作成します。

```
mkdir -p /opt/LifeKeeper/subsys/projectapp/Resources/plan/actions
```

4. **アプリケーションの定義。** 以下のコマンドで、アプリケーション *projectapp* を作成します。

```
app_create -d Server1 -a projectapp

app_create -d Server2 -a projectapp
```

5. **リソースタイプの定義。** 以下のコマンドで、リソースタイプ *plan* を作成します。

```
typ_create -d Server1 -a projectapp -r plan

typ_create -d Server2 -a projectapp -r plan
```

6. **リカバリスクリプトのインストール。** restore と remove のスクリプトを各サーバの以下のディレクトリにコピーします。

   */opt/LifeKeeper/subsys/projectapp/Resources/plan/actions*

7. **インスタンスの定義。** 以下のコマンドで、リソースのインスタンスタイプが plan、ID が *1yrplan* のリソースを定義します。

```
ins_create -d Server1 -a projectapp -r plan -I\

AUTORES_ISP -t the-project-plan -i 1yrplan

ins_create -d Server2 -a projectapp -r plan -I\

SEC_ISP -t the-project-plan -i 1yrplan
```

   Server1 に作成したインスタンスの -I AUTORES_ISP 命令は、LifeKeeper の再起動時にそのリソースを自動的に in service にするように LifeKeeper に指示します。この例では、リソースの restore スクリプトが実行され、正常に実行された場合はリソースが ISP 状態になります。この動作は、ペアのリソースがすでにサービス起動している場合は実行されません。

   Server2 に作成したインスタンスの -I SEC_ISP 命令は、LifeKeeper の再起動時にそのリソースを in service にしないように LifeKeeper に指示します。その代わり、Server2 は Server1 上にあるリソースのバックアップとして機能し、プライマリのリソースまたはサーバに障害が発生したときにローカルリソースを in service にします。

8. **依存関係の定義。** 以下のコマンドは、アプリケーションとファイルシステムの依存関係を定義します。

```
dep_create -d Server1 -p the-project-plan -c project-data-on-System1

dep_create -d Server2 -p the-project-plan -c project-data-from-
Server1

dep_create -d Server1 -p the-project-plan -c schedule-on-Server1

dep_create -d Server2 -p the-project-plan -cschedule-from-Server1
```

9. **lcdsync の実行。** 以下の `lcdsync` コマンドを実行して、設定のコピーを更新するように LifeKeeper に通知します。

```
lcdsync -d Server1

lcdsync -d Server2
```

10. **リソースを In Service にする。** プライマリサーバで LifeKeeper の GUI にアクセスし、**[Edit] > [Resource] > [In-Service]** をクリックしてリソースを In Service にします。もしくはプライマリサーバー上で以下のコマンドを実行し In-Service にします。

```
perform_action -t the-project-plan -a restore
```

11. **共有関係の作成。** 以下のコマンドで各ノードで登録したリソースの共有関係を作成し、リソースを切り替えることができるようにします。

```
eqv_create -d Server1 -t the-project-plan -p 1 -S Server2 -o the-
project-plan -r 10 -e SHARED

eqv_create -d Server2 -t the-project-plan -p 10 -S Server1 -o the-
project-plan -r 1 -e SHARED
```

# 5.4.4.1.9.1.2. LCD の設定データ

LCD には、以下の関連するデータタイプが保存されます。

- ◦ 依存関係の情報
- ◦ リソースのステータス情報
- ◦ サーバ間のイクイバレンシ情報

## 依存関係の情報

定義した各リソースについて、LifeKeeper は依存関係のリスト、および依存物 (あるリソースに依存するリソース) のリストを保持します。詳細については、`LCDI_relationship (1M)` と `LCDI_instances (1M)` のマニュアルページを参照してください。

## リソースのステータス情報

LifeKeeper は、各リソースインスタンスのステータス情報をメモリに保持します。LCD が認識するリソースの状態 は、 **ISP** 、 **ISU** 、 **OSF** 、 **OSU** 、および **ILLSTATE** です。システムイベントが発生した場合、または管理者が特定の操作を行った場合に、リソースがある状態から別の状態に変化することがあります。リソースの状態が変化した場合、ステータスの変化が、ローカルサーバの LCD、およびそのリソースのダイアログサーバ上にあるデータベースに反映されます。

## サーバ間のイクイバレンシ情報

さまざまなサーバ上にある複数のリソース間に関係が存在することがあります。イクイバレンシ情報 とは、別のサーバ上にある 2 つのリソースが同一の物理エンティティであることを示す関係です。2 台のサーバが、イクイバレンシ情報の関係にある 1 つのリソースを持つ場合、LifeKeeper はその動作により、2 台のサーバ上にあるリソースの 1 つのみが同時に In Service、保護 (ISP) になるようにします。両方のサーバでそのリソースインスタンスを Out of Service ( **OSU** または **OSF** ) にすることができますが、データの整合性の理由から、同時に In Service にできるリソースは 1 つのみです。

SCSI バス上にある複数のディスクが、同等なリソースの一例です。SCSI のロック (または予約) メカニズムにより、任意の時点でディスクデバイスのロックを所有できるのは 1 台のサーバのみです。このロック所有機能により、同時に複数のサーバによる同一ディスクリソースへのアクセスが防止されます。

さらに、階層内の依存関係により、ファイルシステムのようにディスクに依存するリソースはすべて、同時に 1 台のサーバでのみ In Service になります。

# 5.4.4.1.9.1.3. LCD のディレクトリ構造

*/opt/LifeKeeper* の下にある主なサブディレクトリを示します。

**config** - LifeKeeper の設定ファイル。イクイバレンシ情報を含みます。 **bin** - LifeKeeper の実行可能プログラム。is_recoverable などがあります。詳細については、障害検出とリカバリのシナリオ を参照してください。 **subsys** - リソースとタイプ。LifeKeeper は、共有 SCSI ディスクサブシステムのリソースとタイプの定義を scsi で、汎用アプリケーションのメニュー機能を gen で提供します。アプリケーションのインターフェースを定義する場合は、subsys の下にディレクトリを作成してください。

- 

**events** - 警報イベント。詳細については、LifeKeeper の警報とリカバリ を参照してください。

*/opt/LifeKeeper* 内の LCD ディレクトリの構造については、*/opt/LifeKeeper* 内の　LCD ディレクトリの構造 のトピックを参照してください。

# 5.4.4.1.9.1.4. LCD のリソースタイプ

LCD は共有メモリ、および */opt/LifeKeeper* ディレクトリの両方に保持されます。ディレクトリ構造の図に示すように、subsys には、アプリケーションインターフェースの指定に使用できるアプリケーションリソースセットが 2 つあります。

- gen - 汎用アプリケーションとファイルシステムの情報
- scsi - SCSI に固有のリカバリ情報

これらのサブディレクトリについてはリソースのサブディレクトリ を参照してください。

# 5.4.4.1.9.1.5. LifeKeeper のフラグ

ステータスの詳細表示 の後部近くに、システムのフラグセットがあります。共通タイプは、プロセスのロックが動作を完了するまで他のプロセスを確実に待機させるために使用する LCD のロックフラグです。LCD のロックの標準フォーマットは以下のとおりです。

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

一般的な LCD のロックフラグの例を示します。

- ◦ **!action!02833!701236710!<servername>:filesys** .ファイルシステム階層を作成すると、このフォーマットでステータス表示にフラグが生成されます。 *filesys* の指定は、他のアプリケーションリソース階層では別のリソースタイプである場合も、一般的なアプリケーションやユーザ定義アプリケーションでは *app* である場合もあります。

- ◦ 他の代表的なフラグとして、 **!nofailover!machine** and **shutdown_switchover** があります。 **!nofailover!machine** フラグは、LifeKeeper が作成と削除を行う内部の一時フラグで、サーバのフェイルオーバを制御します。 **shutdown_switchover** フラグは、このサーバのシャットダウン方針がスイッチオーバに設定されたことを示し、サーバのシャットダウンによりスイッチオーバが発生します。使用可能なフラグの詳細については、`LCDI-flag(1M)` を参照してください。

# 5.4.4.1.9.1.6. リソースのサブディレクトリ

***scsi*** と ***gen*** のディレクトリはそれぞれ、リソースのサブディレクトリを持ちます。これらのディレクトリの内容は、LifeKeeper が提供するリソースタイプのリストです。

**scsi のリソースタイプ。** これらのリソースタイプは、*/opt/LifeKeeper/subsys/scsi/resources* ディレクトリにあります。実際の設定によっては、その他のディレクトリが存在する場合があります。

- **device** - ディスクパーティション、または仮想ディスクデバイス
- **disk** - 物理ディスク、または LUN
- **hostadp** - ホストアダプタ

**gen のリソースタイプ。** これらのリソースタイプは、*/opt/LifeKeeper/subsys/gen/resources* ディレクトリにあります。

- **filesys** - ファイルシステム
- **app** - 汎用またはユーザ定義のアプリケーションであり、他のリソースに依存することがある

各リソースタイプのディレクトリには、以下のものが 1 つ以上あります。

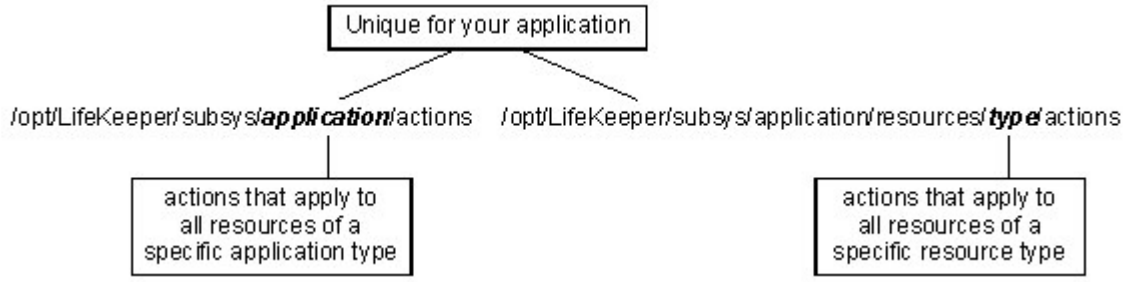
- **インスタンス。** このファイルは、LCD に保存されている、リソースインスタンスに関する恒久的な情報を反映します。このリソースタイプに関連付けられたリソースインスタンスの記述的な情報があります。

**警告:** インスタンスファイル (または LCD ファイル) を直接変更しないでください。リソースインスタンスの作成や操作を行うには、LifeKeeper の GUI の機能、または ins_create、ins_remove、ins_gettag、ins_setas、ins_setinfo、ins_setinit、ins_setstate、および ins_list の LifeKeeper の LCDI_instances コマンドのみを使用してください。これらのコマンドの詳細については、LCDI_instances (1M) のマニュアルページを参照してください。

- ***recovery*** 。このオプションのディレクトリには、障害が検出されたリソースのローカルリカバリの試行に使用されるプログラムがあります。recovery ディレクトリには、sendevent に渡されるイベントクラスに対応するディレクトリがあります。ディレクトリの名前は、sendevent プログラムに渡されるクラスパラメータ (-C) と一致する必要があります。 (LifeKeeper の警告とリカバリ を参照)。

各サブディレクトリに、アプリケーションは対応するイベントタイプを処理するリカバリプログラムを入れることができます。これらのプログラムの名前は、sendevent の -E パラメータで渡される文字列と一致する必要があります。このオプションのディレクトリは、複数のアプリケーションに使用されるように存在することはできません。

- ***actions*** 。このディレクトリには、特定のリソースタイプのリソースインスタンスについてのみ動作するリカバリ実行プログラムのセットがあります。使用するアプリケーションについて、アプリケーション内のすべてのリソースに適用する動作がある場合は、その動作を、***resource type*** ディレクトリではなく、アプリケーションディレクトリの ***actions*** サブディレクトリに入れてください。




リカバリ指示ソフトウェアが、リソースインスタンスの変更や復旧に使用されます。各リソースタイプの ***actions*** ディレクトリに、 **remove** と **restore** の 2 つの動作が必要です。

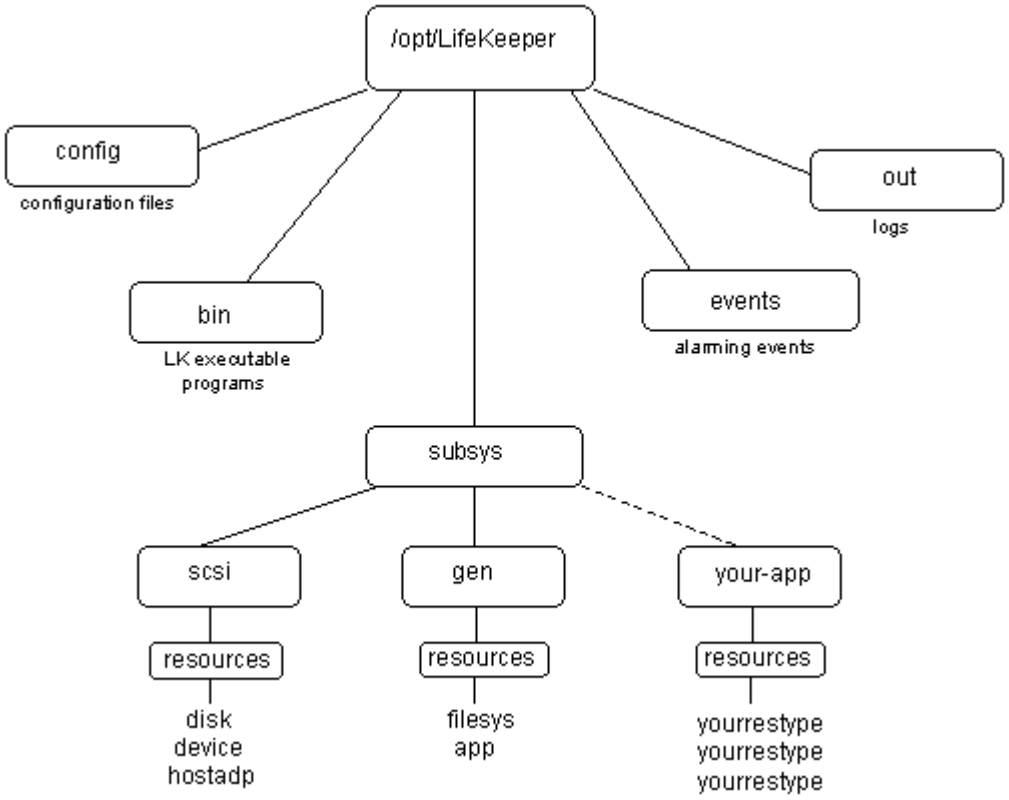
## リソースの動作

リソースタイプの ***actions*** ディレクトリには、特定のアプリケーションの動作を記述するプログラム (多くの場合は shell スクリプト) があります。各リソースタイプについて、restore と remove の 2 つの動作が必要です。

remove と restore のプログラムは、正反対の機能を実行する必要があります。つまり、相互の動作を元に戻す必要があります。これらのスクリプトは、絶対に手動で実行しないでください。これらのスクリプトは、LifeKeeper のリカバリ動作と制御のインターフェース (LRACI) の `perform_action` shell プログラムのみが実行する必要があります (`LRACI-perform_action （1M）` マニュアルページを参照)。

# 5.4.4.1.9.1.7. /opt/LifeKeeper の LCD のディレクトリ構造

以下の図に、 ***/opt/LifeKeeper*** のディレクトリ構造を示します。



# 5.4.4.1.9.2. LCM

The LifeKeeper Communications Manager (LCM) は、1 台以上の LifeKeeper サーバ上にあるプロセス間に信頼性の高い通信を提供します。このプロセスは、システム間の冗長コミュニケーションパスを使用できるので、1 つのコミュニケーションパスに障害が発生しても、LifeKeeperやそれが保護するリソースには障害が発生しません。LCM は、RS-232 (TTY) と TCP/IP の接続を含む多様な通信方法をサポートしています。

LCM は以下の機能を提供します。

- ◦ **LifeKeeper のハートビート。** 接続している他の LifeKeeper システムと定期的に通信して、他のシステムが動作を継続しているかどうかを判断します。LifeKeeper はハートビート信号がないことを認識することにより、他の方法では検出されないシステム全体の障害を検出できます。

- ◦ **管理サービス。** LifeKeeper の管理機能は、LCM の機能を使用してリモート管理を実行します。この機能は、シングルポイントの管理、設定の検証、および管理動作の正常性チェックに使用されます。

- ◦ **設定とステータスの通信。** LifeKeeper 設定データベース (LCD) は、リソースのステータス、可用性、および設定を LCM 機能経由で記録します。LCM の機能により、LCD はプライマリとセカンダリのシステム間で整合性のあるリソース情報を保持できます。

- ◦ **フェイルオーバリカバリ。** あるシステム上のリソースに障害が発生すると、LCM は LifeKeeper に、バックアップシステム上にリソースを復旧するように通知します。

LCM が提供する LifeKeeper のサービスに加えて、shell コマンドセットによりアプリケーションによる信頼性の高いシステム間通信が可能です。これらのコマンドとして、`snd_msg`, `rcv_msg`、`can_talk` などがあります。これらのコマンドの詳細については、`LCMI_mailboxes (1M)` のマニュアルページを参照してください。LCM はシステム上でリアルタイムプロセスとして動作し、システムのハートビートが送信されるなどの重要な通信が確実に実行されるようにします。

## 関連トピック

通信ステータスの情報

LifeKeeper の警報とリカバリ

# 5.4.4.1.9.2.1. 通信ステータスの情報

ステータス表示の通信ステータスの情報のセクションには、LifeKeeper が認識しているサーバとその現在の状態、および各コミュニケーションパスの情報がリストされます。

以下の例は、ステータスの簡略表示の通信ステータスのセクションのものです。

```
MACHINE NETWORK ADDRESSES/DEVICE STATE PRIO

tristan TCP 100.10.100.100/100.10.100.200 ALIVE 1

tristan TTY /dev/ttyS0 ALIVE --
```

詳細については、ステータスの詳細表示 とステータスの簡略表示 のトピックの「通信ステータスの情報」セクションを参照してください。

# 5.4.4.1.9.2.2. LifeKeeper の警報とリカバリ

LifeKeeper のエラー検出と通知は、イベント警報メカニズム sendevent をベースにしています。**sendevent** メカニズムの重要な概念は、独立したアプリケーションが重要なコンポーネントについて警報を受信できるように登録できることです。警報を開始する側のコンポーネントと受信する側のアプリケーションのいずれも、他のアプリケーションの存在を知るように変更する必要はありません。アプリケーションに固有のエラーが、 **sendevent** 機能経由で LifeKeeper のリカバリメカニズムをトリガできます。

このセクションでは、警報クラス、警報の処理、および警報ディレクトリのレイアウトを含む警報に関連するトピックを説明し、次に警報の概念を示す処理シナリオを示します。

## 警報クラス

*/opt/LifeKeeper/events* ディレクトリには、アラームクラスのセットがリストされます。これらのクラスは、イベントを生成するシステムの特定サブコンポーネントに対応します (例: *filesys* )。各警報クラスのサブディレクトリには、可能性のある警報のセットがあります (例: *badmount* 、 *diskfull* )。shell スクリプトまたはプログラムを適切なディレクトリに入れることで、これらの警報を受信するようにアプリケーションを登録できます。

LifeKeeper は基本的な警報通知機能を使用しています。この警報機能により、イベントについて登録されたすべてのアプリケーションで、該当する警報の発生時に sendevent により処理プログラムが非同期で実行されます。LifeKeeper が存在する場合、 **sendevent** プロセスははじめに、LifeKeeper のリソースオブジェクトがクラスとイベントを処理できるかどうかを判断します。LifeKeeper がクラス/イベントの一致を検出した場合、適切な復旧シナリオが実行されます。

**sendevent** 警報機能の追加スクリプトを定義することは任意です。LifeKeeper リソースを定義すると、LifeKeeper が基本的な警報機能を提供します。その詳細は、この章の処理シナリオで後述します。

注記: リソースインスタンスのローカルリカバリは、LifeKeeper の制御下にあるアプリケーションが、中断されたリソースサービスを、イベントが発生したシステムのエンドユーザに返そうとする試行です。サーバ間リカバリでは、アプリケーションはバックアップシステムに移行できます。この種のリカバリは、ローカルリカバリが失敗したか、ローカルリカバリが不可能である場合に試行されます。

## 警報の処理

LifeKeeper の注意が必要な可能性のあるイベントを検出するアプリケーションまたはプロセスは、**sendevent** プログラムを実行し、各エラークラス、エラー名、および障害のあるインスタンスの引数を渡すことにより、イベントを報告できます。必須の詳細、オプションのパラメータ、および構文については、sendevent (5) のマニュアルページを参照してください。

## 警報ディレクトリのレイアウト

*/opt/LifeKeeper/events* ディレクトリには、2 種類の内容があります。

- ◦ **LifeKeeper の指定クラス** 。LifeKeeper は、 *events* ディレクトリの下に *lifekeeper* と *filesys*

の 2 つの警報クラスを用意しています。警報イベントの例としては、 *diskfull* などがあります。警報クラスは、 **sendevent** コマンドの -C オプションで渡される文字列に対応し、警報イベントは -E オプションで渡される文字列に対応します。Lifekeeper の警報クラスは、LifeKeeper のサブシステム内のイベント報告用に内部的に使用されます。

- ◦ アプリケーションに固有のクラス 。特定のアプリケーションで警報クラスの定義が必要な場合、 *events* ディレクトリに他のサブディレクトリが追加されます。アプリケーションは shell スクリプトまたはバイナリプログラムをそのサブディレクトリに入れることで、これらの警報を受信するように登録します。これらのプログラムの名前は、属するアプリケーションパッケージの名前に由来します。

# 5.4.4.1.9.3. LifeKeeper API for Monitoring

## はじめに

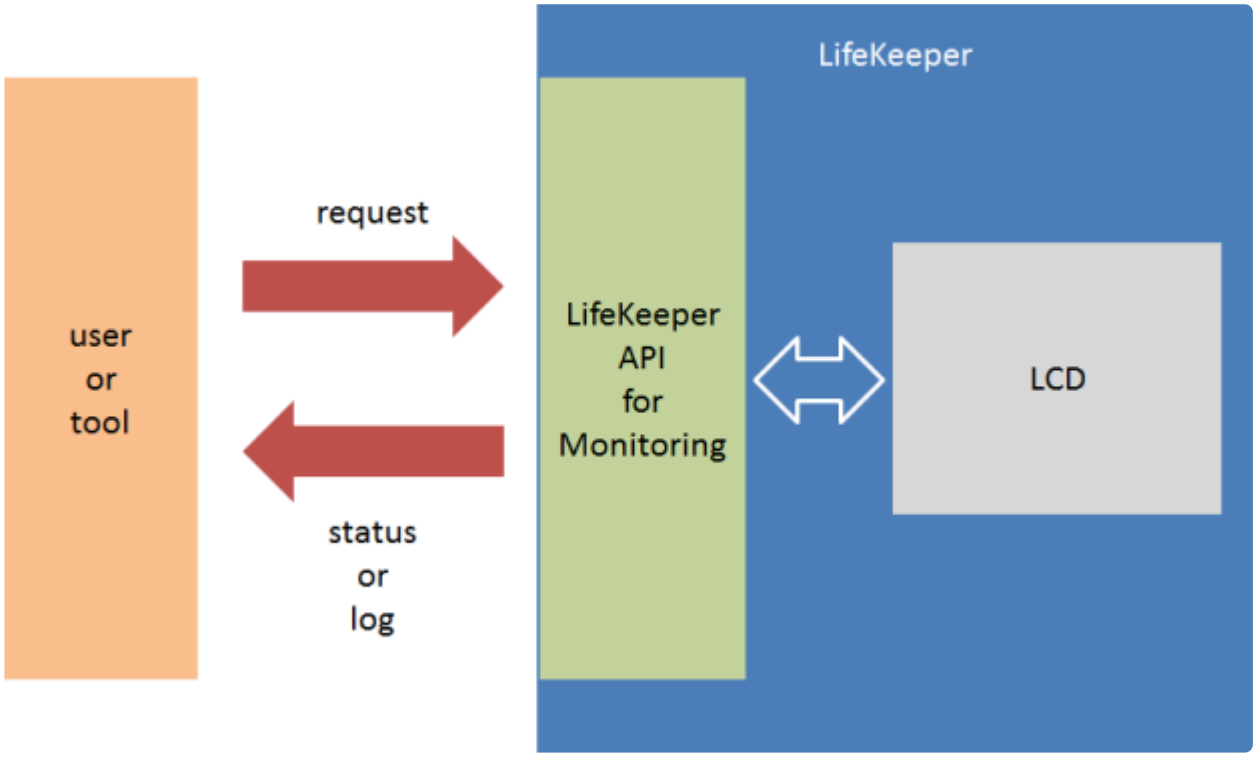
LifeKeeper API for Monitoring は、稼働中の LifeKeeper に対して問い合わせを行うことで、LifeKeeper ノードの稼働状況や保護しているリソースの状況を取得できます。

## 概要

本APIはlighttpd上でCGIとして動作し、lcdstatusコマンドで取得できる情報を、API経由で取得できるようにしたものです。本APIで取得できる情報の概要は以下のとおりです。

- ◦ LifeKeeper ノードの死活
- ◦ コミュニケーションパスの死活
- ◦ リソースの保護状況

これにより、ユーザは LifeKeeper サーバにログインせずとも異常の発生を知ることができます。しかし、具体的にどのような異常が発生しているのかを確認するためには、LifeKeeperGUI にログインしたり、LifeKeeper のログを確認したりする必要があります。




## 取得できる情報

LifeKeeper が稼働している１台のノードに対して、ユーザが問合せを行った場合、以下の情報が取得できます。なお、LifeKeeper は通常複数のサーバで構成されることが多いですが、取得できる情報は問合せを

行ったサーバのもののみです。

- ◦ ステータス
  - º 各サーバの稼働状況
    - ▪ノード名
    - ▪ 稼働状況（ALIVE/DEAD)
  - º コミュニケーションパスの稼働状況
    - ▪ ノード名
    - ▪ 稼働状況（ALIVE/DEAD）
    - ▪ アドレス/デバイス名
  - º リソースの保護状況
    - ▪ ノード名
    - ▪ タグ
    - ▪ 稼働状況（ALIVE/DEAD）
    - ▪ ステータス（ISP、OSU、OSF、…）
    - ▪ 依存関係
    - ▪ DRリソースのミラー情報(ステータスがISPの場合のみ取得可能)
      - ▪ タグ
      - ▪ Mirror status configuration (Sync, Pause, …)
      - ▪ Replication status (75%, 100%, …)
- ◦ ログ
  - º /var/log/lifekeeper.log ※ログファイルのパスが変更されている場合はサポートしない
    - ▪ 最大1000行分（データ出力フォーマットがHTMLの場合）
    - ▪ 全て（データ出力フォーマットがplain textの場合）
  - º /var/log/lifekeeper.err ※ログファイルのパスが変更されている場合はサポートしない
    - ▪ 最大1000行分（データ出力フォーマットがHTMLの場合）

- ▪ 全て（データ出力フォーマットがplain textの場合）

# 通信形式

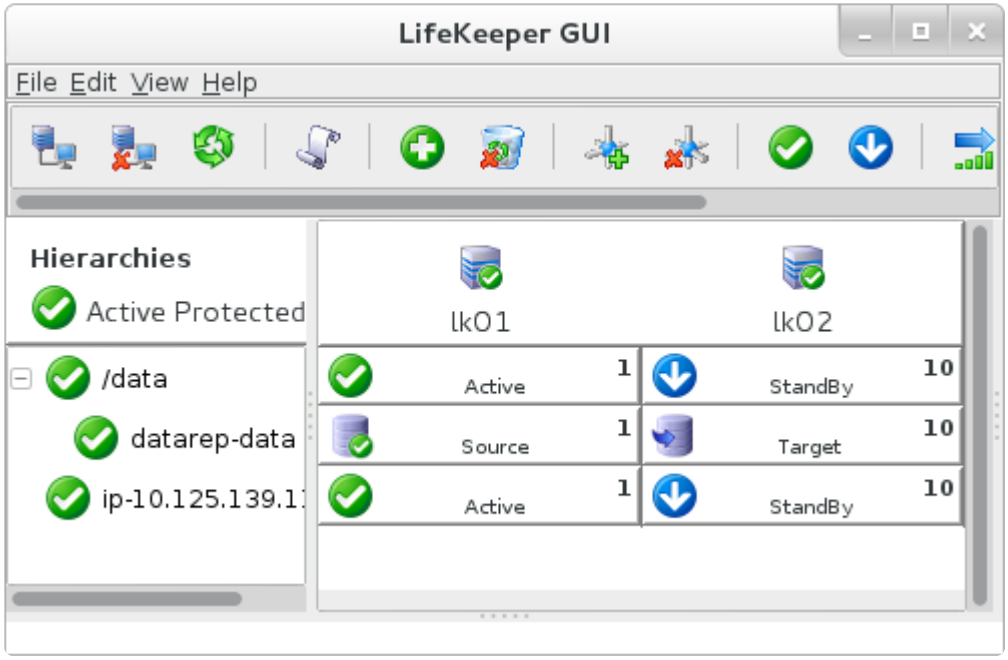
本 API で情報を取得するための通信は、HTTP で行います。
ユーザが対象サーバ上で動作する lighttpd 上で動作する CGI に対してHTTP GET リクエストを送信すると、前述の情報を取得することができます。

# データフォーマット

データフォーマットは以下の３つがあります。

- ◦ JSON
  - º 外部ツールにてステータスを解析するために利用されることを想定
  - º ステータスの確認が可能
  - º ログの出力はできない
- ◦ HTML
  - º ブラウザでの目視確認などに利用されることを想定
  - º ステータスの確認が可能
  - º ログは 1000 行分のみ取得可能
- ◦ plain text
  - º ステータスの確認はできない
  - º ログの取得ができる
  - º /var/log/lifekeeper.log や /var/log/lifekeeper.err の全ての内容を取得可能

LifeKeeper のクラスタ設定が以下の図の状態であるとき、取得できる JSON フォーマットと HTML フォーマットを示します。





```
{
        "resource": [
                {
                        "replication": {},
                        "child": [
                                {
                                        "tag": "datarep-data"
                                }
                        ],
                        "server": {
                                "status": "ISP",
                                "name": "lk01"
                                },
                        "tag": "/data"
                },
                {
                        "replication": {
                                "percent": "100%",
                                "mirror": "Fully Operational"
                        },
                        "child": [],
                        "server": {
                                "status": "ISP",
                                "name": "lk01"
                        },
                        "tag": "datarep-data"
                },
                {
                        "replication": {},
                        "child": [],
```

```
                        "server": {
                                "status": "ISP",
                                "name": "lk01"
                        },
                        "tag": "ip-10.125.139.118"
                }
        ],
        "compath": [
                {
                        "status": "ALIVE",
                        "server": [
                                {
                                        "name": "lk01",
                                        "term": "192.168.139.18"
                                },
                                {
                                        "name": "lk02",
                                        "term": "192.168.139.19"
                                }
                        ]
                },
                {
                        "status": "ALIVE",
                        "server": [
                                {
                                        "name": "lk01",
                                        "term": "172.20.139.18"
                                },
                                {
                                        "name": "lk02",
                                        "term": "172.20.139.19"
                                }
                        ]
                }
        ],
        "server": [
                {
                        "status": "ALIVE",
                        "name": "lk01"
                },
                {
                        "status": "ALIVE",
                        "name": "lk02"
                }
        ]
}
```

RESOURCEs

| tag | lk01 |
|---|---|
| /data | ISP |
| datarep-data | ISP |
| ip-10.125.139.118 | ISP |

DATA REPLICATIONs

| tag | nodes | mirror status | replication status |
|---|---|---|---|
| datarep-data | lk01 -> lk02 | Fully Operational | 100% |

COMMUNICATION PATHs

| communication path | status |
|---|---|
| 192.168.139.18/192.168.139.19 | ALIVE |
| 172.20.139.18/172.20.139.19 | ALIVE |

# 使用方法

## 本機能の有効化

本機能はデフォルトでは無効になっています。有効にする場合、/etc/default/LifeKeeper に以下の設定を行う必要があります。

```
LKAPI_MONITORING=true
```

なお、本機能の有効･無効を切り替える場合、LifeKeeper の再起動は不要です。

クラスタを構成している場合、設定の有効化は、問い合わせを行うLifeKeeperノードにのみ必要です。問い合わせを行わないノードは本機能の設定が無効であっても問題ありません。

## ポート番号

本機能は、デフォルトでは 779 番ポートを使用します。ポート番号をデフォルトから変更する場合、/etc/default/LifeKeeper に以下の設定を行う必要があります。

```
LKAPI_WEB_PORT=<ポート番号>
```

## 使用例

LifeKeeper が稼働しているサーバに以下のように問い合わせることで情報を取得できます。ここでは、curl を用いた例を示します。

```
curl http://<IPADDR>:779/Monitoring.cgi
```

このように引数を与えない場合、デフォルトでは JSON 形式にて現在のステータスが取得できます。HTML 形式でログを取得する場合、以下のように引数を指定します。

```
curl http://<IPADDR>:779/Monitoring.cgi?format=html&show=log
```

使用できる引数一覧は以下の通りです。

| 名前 | 説明 | 値 | 備考 |
|---|---|---|---|
| show | 取得したい情報を指定する | status, log, log-err | デフォルトは status |
| format | データフォーマットを指定する | json, html, plain | デフォルトでは json<br>show=log,log-err を指定した時に、format=json を指定した場合はエラーとなる |

# セキュリティ

本機能を用いると、情報取得のリクエストを行った全てのユーザが LifeKeeper のステータスを取得することができます。
そのため、ユーザのセキュリティ設定に基づき、ステータスを取得できるユーザの制限や、SSL の設定、情報の暗号化を行うことができます。

## Basic 認証

本APIで情報を取得するために、Basic認証を使用することができます。

lighttpd 設定ファイルの修正例は以下の通りです。赤字で示す箇所を追加、変更してください。修正後、設定を反映させるため、コマンド「/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd」を実行し、lighttpd を再起動してください。

なお htpasswd コマンドが存在しない場合、ご使用のディストリビューション提供の httpd-tools をインストールしてください。

/opt/LifeKeeper/etc/lighttpd/lighttpd.conf

```
server.modules = ( : "mod_auth", # uncommenting
```

/opt/LifeKeeper/lib64/steeleye-lighttpd/include_server_bind.pl

```
print qq/\$SERVER["socket"] == "$addr:$port" {\n/;
print qq/    server.document-root = "\/opt\/LifeKeeper\/api"\n/;
print qq/    auth.backend = "htpasswd"\n/;
print qq/    auth.backend.htpasswd.userfile = "\/opt\/LifeKeeper\/etc\/lighttp
d\/lighttpd.user.htpasswd"\n/;
print qq/    auth.require = ( "\/" =>\n/;
print qq/        (\n/;
print qq/                    “method”  => “basic”,\n/;
```

```
print qq/                    “realm”    => “LifeKeeperAPI”,\n/;
print qq/                    “require” => “valid-user”\n/;
print qq/         )\n/;
print qq/    )\n/;
print qq/ }\n/;
```

htpasswdファイル作成手順

```
htpasswd -c /opt/LifeKeeper/etc/lighttpd/lighttpd.user.htpasswd
<USERNAME>
```

## SSL/TLS 対応

本 API での通信において、SSL/TLS を用いることができます。lighttpd 設定ファイルの修正例は以下の通りです。修正後、設定を反映させるため、コマンド「/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd」を実行し、lighttpd を再起動してください。

また、この対応により、本 API が使用するデフォルトポートの設定が不要となった場合は、次の「779 番ポートの無効化」を参照し無効化をおこなってください。

/opt/LifeKeeper/etc/lighttpd/include_ssl_port.pl

```
configAPI("0.0.0.0", 443);
if(socket($sock, AF_INET6, SOCK_STREAM, 0)) {
    configAPI("[::]", 443);
}
sub configAPI {
    my $addr = shift;
    my $port = shift;

    print qq/\$SERVER["socket"] == "$addr:$port" {\n/;
    print qq/    server.document-root = "\/opt\/LifeKeeper\/api"\n/;
    print qq/    ssl.engine  = "enable"\n/;
    print qq/    ssl.pemfile = "\/opt\/LifeKeeper\/etc\/certs\/LK4LinuxValidNo
de.pem"\n/;
    print qq/    ssl.use-sslv2 = "disable"\n/;
    print qq/    ssl.use-sslv3 = "disable"\n/;
    print qq/ }\n/;
}
```

### 779 番ポートの無効化

SSL/TLS の対応を行った場合、お客様の環境によっては本 API が従来使用していたポート（デフォルトでは 779 番ポート）の設定が不要となる場合があります。その場合は、以下の設定を削除、あるいはコメントアウトを行い、本 API が使用するデフォルトポートの設定を無効化してください。修正後、設定を反映させるため、コマンド「/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd」を実行し、lighttpd を再起動してください。

/opt/LifeKeeper/lib64/steeleye-lighttpd/include_server_bind.pl

```
#lkapi_config("0.0.0.0", $lkapi_port);
#if(socket($sock, AF_INET6, SOCK_STREAM, 0)) {
#     lkapi_config("[::]", $lkapi_port);
#}
```

## IP アドレスアクセス制限

アクセス可能な IP アドレスを制限することができます。

IP アドレス 192.168.10.1 以外の接続を拒否する場合の lighttpd 設定ファイルの修正例は以下の通りです。修正後、設定を反映させるため、コマンド「/opt/LifeKeeper/sbin/sv restart steeleye-lighttpd」を実行し、lighttpd を再起動してください。

/opt/LifeKeeper/etc/lighttpd/conf.d/lkapi_user.conf

```
$HTTP["remoteip"] != "192.168.10.1" {
url.access-deny = ( "" )
}
```

# エラー

本 API 実行時にエラーが発生することがあります。その際、エラーの概要が出力されます。以下に JSON フォーマットでのエラー例を示します。 なお、ここでは lighttpd が返す HTTP ステータスコードについては触れません。

```
{
"error" : {
        id : -1,
        message : "Failed to get LCD status"
```

```
            }
}
```

出力フォーマットが HTML 版の場合も、同様の内容のメッセージを出力します。

# 5.4.4.1.10. メンテナンス作業

以下に、LifeKeeper のメンテナンス作業を示します。

---

LifeKeeper の設定値の変更

ファイルシステムの健全性の監視

LifeKeeper が保護するシステムのメンテナンス

リソース階層のメンテナンス

フェイルオーバ後の復旧

LifeKeeper の削除

ファイアウォールを使用した状態での LifeKeeper の実行

ファイアウォール経由での LifeKeeper GUI の実行

リソース階層の転送

# 5.4.4.1.10.1. LifeKeeper の設定値の変更

LifeKeeper には、設定と設定を行った後に変更を要する場合がある値が多数あります。変更を要する場合がある値の例として、LifeKeeper サーバーの uname、コミュニケーションパスの IP アドレス、IP リソースのアドレス、タグ名などがあります。これらの値を変更するには、注意して以下の手順に従ってください。

1. 以下のコマンドを使用して、すべてのサーバーで LifeKeeper を停止してください。
   `$LKROOT/bin/lkcli stop`
   コミュニケーションパスを削除したり、サーバーからリソース階層を拡張解除したりする必要はありません。

2. LifeKeeper サーバーの uname を変更する場合は、`hostname(1)` コマンドを使用してサーバーのホスト名を変更してください。

3. 先に進む前に、新しいホスト名がクラスター内のすべてのサーバーで解決可能であることを確認してください。コミュニケーションパスのアドレスを変更する場合は、新しいアドレスが設定され、動作していることを確認してください (**ping** と **telnet** のユーティリティーをこの確認に使用可能)。

4. LifeKeeper の複数の値を変更する必要がある場合は、クラスター内の各サーバー上のファイルで、古い値と新しい値を以下のフォーマットで指定する必要があります。

   ```
   old_value1=new_value1
   ...
   old_value9=new_value9
   ```

5. クラスター内の **すべて** のサーバーで **lk_chg_value** コマンドを実行し、出力を確認して、予測しなかった変更内容による副作用が発生していないことを確認してください。

   a. 変更する値が複数ある場合は、以下のコマンドを実行してください。
      `$LKROOT/bin/lk_chg_value -Mvf file_name`

      注記: ***file_name*** は、手順 4 で作成したファイルの名前です。

   b. 変更する値が 1 つのみの場合は、以下のコマンドを実行してください。
      `$LKROOT/bin/lk_chg_value -Mvo old_value -n new_value`

      **-M** オプションは、LifeKeeper のすべてのファイルに対して変更を行わないことを指定します。

6. クラスター内のすべてのサーバーで、 **-M** オプションを指定せずに `lk_chg_value` コマンドを実行して、LifeKeeper のファイルを変更してください。

   a. 変更する値が複数ある場合は、以下のコマンドを実行してください。
      `$LKROOT/bin/lk_chg_value -vf file_name`

      注記: ***file_name*** は、手順 4 で作成したファイルの名前です。

b. 変更する値が 1 つのみの場合は、以下のコマンドを実行してください。
```
$LKROOT/bin/lk_chg_value -vo old_value -n new_value
```

7. 以下のコマンドを使用して、LifeKeeper を再起動してください。
```
$LKROOT/bin/lkcli start
```
LifeKeeper の GUI を使用してクラスターを表示する場合は、GUI を閉じてから再起動しなければならないことがあります。

## 例

*Server1* と *Server2* は、2 ノードクラスター内にある LifeKeeper サーバーの uname です。 *Server1* は、アドレス 172.17.100.48 のコミュニケーションパスを持ちます。 *Server2* はアドレス 172.17.100.220 の IP リソースを持ち、この IP リソースは *Server1* に拡張されています 。 *Server1* について以下の値を変更します。

| 値 | 旧 | 新 |
|---|---|---|
| uname | Server1 | Newserver1 |
| コミュニケーションパスのアドレス | 172.17.100.48 | 172.17.105.49 |
| IP リソースのアドレス | 172.17.100.220 | 172.17.100.221 |

これらの変更を行うには、以下の手順を実行する必要があります。

1. 以下のコマンドを使用して、 *Server1* と *Server2* の両方で LifeKeeperを停止してください。
```
$LKROOT/bin/lkcli stop
```

2. 以下のコマンドを使用して、 *Server1* の uname を *Newserver1* に変更してください。
```
hostname Newserver1
```

3. *Newserver1* と *Server2* の両方に、以下の内容を持つファイル */tmp/subs* を作成してください。

```
Server1=Newserver1
172.17.100.48=172.17.105.49
172.17.100.220=172.17.100.221
```

4. 両方のサーバーで以下のコマンドを実行し、出力を確認して、予測しなかった変更内容による副作用が発生していないことを確認してください。
```
$LKROOT/bin/lk_chg_value -Mvf /tmp/subs
```

5. 両方のサーバーで、 **-M** オプションを指定せずに **lk_chg_value** コマンドを実行して、LifeKeeper のファイルを変更してください。
```
$LKROOT/bin/lk_chg_value -vf /tmp/subs
```

6. 以下のコマンドを使用して、両方のサーバーで LifeKeeper を再起動してください。
```
$LKROOT/bin/lkcli start
```

注記: LifeKeeper のファイルを変更せずに **lk_chg_value** による変更内容を表示するには、 **-M** オプションを使用してください。 **lk_chg_value** が調べるファイルを表示するには、 **-v** を使用してください。タグ名を変更しない場合は、 **-T** オプションを使用してください。リソース ID を変更しない場合は、 **-I** オプションを使用してください。

# 5.4.4.1.10.2. ファイルシステムの健全性の監視

ファイルシステムの健全性の監視機能は、LifeKeeper が保護する、ファイルシステム依存のアプリケーションで障害が発生する原因となる条件を検出します。監視は、アクティブ / サービス中のリソース (つまりファイルシステム) でのみ実行されます。監視する条件は、以下の 2 つです。

- ファイルシステムがフル (またはほぼフル) の状態になる。
- ファイルシステムが不適切にマウント (またはアンマウント) された。

これら 2 つの条件のいずれかが検出されると、いくつかの動作のいずれかが実行されることがあります。

- 警告メッセージがログ記録され、システム管理者に電子メールを送信できる。
- リソースインスタンスのローカルリカバリを試行できる。
- リソースをバックアップサーバにフェイルオーバできる。

## 条件の定義

### フル (もしくは空き容量が少なくなっている) のファイルシステム

ディスクフルの条件は検出できますが、ローカルリカバリまたはフェイルオーバの実行で解決することはできません。管理者の操作が必要です。デフォルトでは、メッセージがログ記録されます。さらに、通知スクリプトを使用して警告メッセージを管理者あてにメール送信するなど、任意の追加処理を行うことが可能です。このスクリプトはLifeKeeperインストールディレクトリ内の "events/filesys/diskfull/notify" にあります。

ディスクフルの条件に加えて、ディスクの使用率が指定のしきい値を超えている場合、警告メッセージを LifeKeeper のログに記録できます。

ディスクフルのしきい値は以下のとおりです。

```
FILESYSFULLERROR=95
```

警告を行うしきい値は以下のとおりです。

```
FILESYSFULLWARN=90
```

デフォルト値は上記のとおりそれぞれ 90% と 95% ですが、 */etc/default/LifeKeeper* ファイルの調整可能なパラメータを使用して設定できます。これら 2 つのしきい値の意味は以下のとおりです。

`FILESYSFULLWARN` – ファイルシステムがこの割合まで使用されると、メッセージが LifeKeeper のログに表示されます。

`FILESYSFULLERROR` – ファイルシステムがこの割合まで使用されると、メッセージが LifeKeeper のログ、およびシステムログに表示されます。ファイルシステムの通知スクリプトも呼び出されます。

## アンマウントされた、または不適切にマウントされたファイルシステム

LifeKeeper は */etc/mtab* ファイルをチェックして、LifeKeeper が保護するサービス中のファイルシステムが実際にマウントされているかどうかを調べます。さらに、 *filesys* のリソース情報フィールドに保存されているマウントオプションに対してマウントオプションをチェックし、階層の作成時に使用されていた元のマウントポジションと一致するかどうかを確認します。

ファイルシステムがアンマウントされているか、不適切にマウントされていることを検出した場合、ローカルリカバリが起動され、正しいマウントオプションを使用してファイルシステムの再マウントが試行されます。

再マウントに失敗した場合、条件を解消するためにフェイルオーバが試行されます。以下のリストに、フェイルオーバに進行する場合がある再マウントの障害の一般的な原因を示します。

- ファイルシステムが破損している (fsck の障害)
- マウントポイントディレクトリの作成失敗
- マウントポイントがビジー
- マウントの失敗
- LifeKeeper の内部エラー

# 5.4.4.1.10.3. LifeKeeper が保護するシステムのメンテナンス

LifeKeeper が保護するサーバをシャットダウンしてメンテナンスを行うときには、メンテナンスの前に、バックアップサーバでシステムのリソース階層を in service にする必要があります。このプロセスにより、メンテナンスが必要なシステム上にある共有ディスクの動作がすべて停止します。

記載の順序で、以下の操作を実行してください。 *Server A* はメンテナンスが必要なプライマリシステム、 *Server B* はバックアップサーバです。

1. ***Server B* で階層を in service にしてください。** バックアップの *Server B* で、LifeKeeper の GUI を使用して、現在 *Server A* でサービス中のリソース階層を in service にします。これにより、LifeKeeper の保護下にある共有ディスクに存在している *Server A* のファイルシステムがアンマウントされます。詳細については、リソースを In Service にする を参照してください。

2. ***Server A* で LifeKeeper を停止してください。** LifeKeeper のコマンド `/opt/LifeKeeper/bin/lkstop -f`を使用して、LifeKeeper を停止します。リソースが保護されていない状態になります。

3. **Linux をシャットダウンし、 *Server A* の電源をオフにしてください。** *Server A* のLinux オペレーティングシステムをシャットダウンし、サーバの電源をオフにします。

4. **メンテナンスを実行してください 。** *Server A* で必要なメンテナンスを実行します。

5. ***Server A* の電源をオンにし、Linux を再起動してください 。** *Server A* の電源をオンにし、次に Linux オペレーティングシステムを再起動します。

6. ***Server A* で LifeKeeper を開始してください。** LifeKeeper のコマンド`/opt/LifeKeeper/bin/lkstart` を使用して、LifeKeeper を開始します。リソースが保護されている状態になります。

7. **必要に応じて、 *Server A* で階層を in-service にしてください。** *Server A* で LifeKeeper の GUI を使用して、 *Server B* にスイッチオーバしていたすべてのリソース階層を in service にしてください。

# 5.4.4.1.10.4. リソース階層のメンテナンス

システム上のその他すべての階層を LifeKeeper で保護した状態で、あるリソース階層のメンテナンスを実行できます。このためには、メンテナンスが必要な階層を Out of Service にし、メンテナンス作業の完了後にその階層を In Service にします。

リソース階層のメンテナンスを実行するには、以下の手順に従ってください。

1. **階層を Out of Service にしてください。** LifeKeeper の GUI を使用して、メンテナンスを実行する必要があるリソース階層をすべて Out of Service にします。詳細については、リソースを Out of Service にする を参照してください。

2. **メンテナンスを実行してください。** リソース階層で必要なメンテナンスを実行します。

3. **階層をリストアしてください。** LifeKeeper の GUI を使用して、リソース階層を In Service にします。詳細については、リソースを In Service にする を参照してください。

# 5.4.4.1.10.5. フェイルオーバ後の復旧

LifeKeeper がプライマリサーバー (*Server A*) からバックアップサーバー (*Server B*) にフェイルオーバリカバリを実行した後、以下の手順を実行してください。

1. **ログを確認してください。** *Server B* の LifeKeeper が *Server A* からフェイルオーバリカバリを実行すると、フェイルオーバ中にステータスメッセージが表示されます。

   実際の出力は、設定によって異なります。マウントやアンマウントの失敗に関するいくつかのメッセージが表示されることが予測されますが、これらのメッセージはリカバリの失敗を示唆しません。これらのメッセージ、および *Server B* でリソースを in service にするときに発生したエラーは、LifeKeeper のログに記録されます。

2. **メンテナンスを実行してください。** *Server A* の障害の原因を特定し、解決します。メンテナンスを実行するために、 *Server A* の電源をオフにすることが必要な場合があります。

3. **必要に応じて、 *Server A* を再起動してください。** メンテナンスが完了したら、必要に応じて *Server A* を再起動します。

4. **必要に応じて、LifeKeeper を開始してください。** *Server A* で LifeKeeper が動作していない場合は、コマンド `$LKROOT/bin/lkcli start` を使用して、LifeKeeper を開始してください。

5. **アプリケーションを *Server A* に戻してください。**都合のよい時点で、LifeKeeper の GUI を使用して、 *Server A* でアプリケーションを in service にします。詳細については、リソースを In Service にする を参照してください。 *Server A* でアプリケーションが **[Automatic Switchback]** に設定されている場合は、この手順は不要なことがあります。

# 5.4.4.1.10.6. LifeKeeper の削除

Linux 環境での LifeKeeper パッケージのアンインストールはコマンドラインから行います。以下のコマンドを実行してください。

```
/opt/LifeKeeper/bin/rmlk
```

これによりすへ゛ての LifeKeeper パッケージがアンインストールされ、システムから /opt/LifeKeeper ディレクトリが削除されます。このコマンドは、システム上で LifeKeeper が実行されているかどうかにかかわらず実行できます。
クラスタ内のすべてのノードで LifeKeeper を実行している状態でコマンドを実行すると、階層の拡張が解除され、すべてのコミュニケーションパスが削除されます。これにより、コマンドが実行されたノードに残っているものがすべて削除され、ノードを効果的にクラスタから削除できます。
LifeKeeper がコマンドの実行時に実行されていない場合、別のノードには削除されずに残るものが発生し、実行中のシステムに影響を与える可能性があります。クラスタ内の他のノードで次のコマンドを実行して（LifeKeeperを実行せずに）これらを削除すると、アンインストールが完了します。

```
/opt/LifeKeeper/bin/rmlk -l
```

このコマンドでは、次のオプションを使用できます。

- -l ：すべての LifeKeeper ライセンスを削除します。

**このコマンドは慎重に使用してくた゛さい。**

**注記:** `lkbackup` コマンドによる LifeKeeper 構成の定期的なバックアップは、結果を自動的に /opt/LifeKeeper/config/ にアーカイブします。 `rmlk` コマンドは /opt/LifeKeeper ディレクトリを削除するため、コマンドを実行する前にアーカイブをバックアップすることを推奨します。

# 5.4.4.1.10.7. ファイアウォールを使用した状態での LifeKeeper の実行

以下のネットワークアクセス要件を満たす場合、LifeKeeper for Linux は、同一サーバー上にファイアウォールを設定した状態で実行できます。

> **注記:** ファイアウォールを単に無効にする場合は、後述のファイアウォールの無効化 を参照してください。

## LifeKeeper のコミュニケーションパス

コミュニケーションパスは、特定の IP アドレスを使用して、LifeKeeper クラスタ内にあるサーバーペアの間に設定されます。TCP ポート 7365 は、作成時にデフォルトで各通信のリモート側により使用されますが、通信の開始側の TCP ポートは任意です。推奨方法は、そのシステムが既知のコミュニケーションパスでローカルとリモートの IP アドレスの各指定ペアについて、受信と送信の両方のトラフィックを許可するように、各 LifeKeeper サーバーにファイアウォールを設定することです。

## LifeKeeper GUI の接続

LifeKeeper GUI は、デフォルトの初期接続ポートであるポート 81 と 82 を含めて、特定の TCP ポートを多数使用します。また、LifeKeeper GUI は、ポート 1024 以降をオブジェクトの送受信に使用するリモートメソッド呼び出し (RMI)　も使用します。これらすべてのポートが、各 LifeKeeper サーバーのファイアウォールで、少なくとも GUI クライアントが動作する外部システムに対して開いている必要があります。

## LifeKeeper の IP アドレスリソース

IP アドレスに関連するアプリケーションにアクセスする必要があるクライアントシステムから、LifeKeeper の階層にある IP アドレスリソースにアクセスできるように、ファイアウォールを設定する必要があります。IP アドレスリソースは LifeKeeper クラスタ内のあるサーバーから別のサーバーに移動できるので、すべての LifeKeeper サーバー上のファイアウォールを適切に設定する必要があります。

また、LifeKeeper は、ブロードキャスト ping のテストを使用して、IP アドレスリソースの健全性を定期的にチェックします。このテストでは、仮想 IP アドレスからブロードキャスト ping パケットを送信し、ローカルサブネット上の他のいずれかのシステムが最初に応答するまで待ちます。このテストが失敗しないようにするには、各 LifeKeeper サーバー上のファイアウォールが以下のタイプのネットワーク動作を許可するように設定する必要があります。

- 仮想 IP アドレスからのインターネット制御メッセージプロトコル (ICMP) パケットの送信 (アクティブな LifeKeeper サーバーがブロードキャスト ping を送信できる)
- 仮想 IP アドレスからの ICMP パケットの受信 (他の LifeKeeper サーバーがブロードキャスト ping を受信できる)

- 任意のローカルアドレスからの ICMP 応答パケットの送信 (他の LifeKeeper サーバーがブロードキャスト ping に応答できる)
- 仮想 IP アドレスでの ICMP 応答パケットの受信 (アクティブな LifeKeeper サーバーがブロードキャスト ping への応答を受信できる)

# LifeKeeper Data Replication

LifeKeeper Data Replication を使用する場合は、複製に ndb を使用する任意のポートへのアクセスを許可するように、ファイアウォールを設定する必要があります。nbd が使用するポートは、以下の式で計算できます。

```
10001 + <mirror number> + <256 * i>
```

**i** は 0 から始まり、使用されていないポート番号が計算されるまで加算されます。 */etc/services* に定義されているポート、 `netstat -an --inet --inet6` の出力に含まれるポート、または LifeKeeper Data Replication の他のリソースが使用中としてすでに定義されているポートは、使用中です。

**例:** LifeKeeper Data Replication リソースのミラー番号が 0 である場合、式は当初、使用するポートを 10001 として計算しますが、この番号は、一部の Linux ディストリビューションでは SCP 設定ポートとして */etc/services* に定義されています。この場合、 ***i*** が 1 だけ増分されてポート番号 10257 が得られます。この番号は、これらの Linux ディストリビューションの */etc/services* には定義されていません。

## その他ノード間通信

LifeKeeper の各サーバーは、ポート 778 の SSL 接続を使用して通信を行います。このポートは、/etc/default/LifeKeeper 内の設定変数 API_SSL_PORT を使用して変更できます。

## ファイアウォールの無効化

ファイアウォールを無効化する場合は、ご使用のディストリビューションのマニュアルに従い設定を行ってください。

# 5.4.4.1.10.8. ファイアウォール経由での LifeKeeper GUI の実行

場合によっては、LifeKeeper クラスタが会社のファイアウォール内に配置され、管理者はファイアウォールの外側にあるリモートシステムから LifeKeeper GUI を実行します。

LifeKeeper は、GUI のサーバとクライアントとの通信にリモートメソッド呼び出し (RMI) を使用します。RMI クライアントは、それぞれの方向に通信を確立できる必要があります。RMI クライアントは動的ポートを使用するので、クライアントには推奨ポートを使用できません。

解決法としては、以下のように ssh を使用して、ファイアウォールを通過する方法があります。

1. 社内の IT 部門が、ファイアウォール内にアクセスするために十分にセキュリティの高い shell ポートを社内ファイアウォールに開けていることを確認します。多くの場合、IT 部門がアクセスを許可するマシンは、実際にはクラスタ内のマシンではなく、そこからクラスタ内にアクセスできる中間マシンです。このマシンは、Unix または Linux が動作するマシンである必要があります。
2. 中間マシンと LifeKeeper サーバの両方が、sshd (Secure Shell デーモン) を実行していること、および X11 ポート転送が有効になっていること (これは通常、 *etc/ssh/sshd_config* の `X11Forwarding yes' 行にある) を確認してください。不明の場合は、IT 部門に依頼してください。
3. X の Unix クライアントから以下のコマンドを使用して、中間マシンにトンネルを作成します。

   ```
   ssh -X -C <intermediate machine>
   ```

   **-C** は「トラフィックの圧縮」を意味し、低速のインターネットリンクから受信する場合に役立つことが多々あります。

4. 中間マシンから以下のコマンドを使用して、LifeKeeper サーバにトンネルを作成します。

   ```
   ssh -X <LifeKeeper server>
   ```

   中間マシンは LifeKeeper サーバとの間にかなり高い帯域幅の接続をもつはずなので、このコマンドには圧縮は不要です。

5. すべての操作が良好に完了した場合、以下のコマンドを実行してください。

   ```
   echo $DISPLAY
   ```

   「*localhost:10.0*」のような値に設定されます。値が設定されない場合、X11 の転送がいずれかの sshd 設定ファイルで無効になっています。

6. 以下のコマンドを実行して、LifeKeeper サーバから単純な *xterm* をポップアップ表示できることを確認してください。

   ```
   /usr/X11R6/bin/xterm
   ```

7. *xterm* が表示された場合、以下のコマンドを使用して、LifeKeeper で **lkGUIapp** を実行できます。

```
/opt/LifeKeeper/bin/lkGUIapp
```

8. GUI コンソールが表示されるまで待ってください。Java は多くのグラフィックス動作を使用し、低速リンクで伝播するには時間がかかります (圧縮している場合でも)。しかし、最終的には GUI コンソールが表示されます。

# 5.4.4.1.10.9. リソース階層の転送

LifeKeeper サーバで定期的なメンテナンスやその他の作業を実行する必要がある場合、LifeKeeper の GUI を使用して In Service のリソースを別のサーバに移動できます。 サーバ *A* から サーバ *B* に In Service のリソース階層を転送するには、GUI を使用して サーバ *B* でリソース階層を in service にします。 サーバ *A* のリソースがすべて、対応するバックアップサービスで In Service になるまで、操作を繰り返します。手順については、リソースを In Service にする を参照してください。

サーバ *A* のリソースがすべて、バックアップサーバでアクティブになった後、アプリケーションの処理に影響を与えることなく、 サーバ *A* をシャットダウンできます。ただし、メンテナンスの期間中、クラスタ内にあるサーバ数によっては、リソースが LifeKeeper で保護されないことがあります。

# 5.4.4.1.11. テクニカルノート

お使いの LifeKeeper 環境に関する設定と動作上の問題に関する以下のテクニカルノートをお読みになることを強く推奨します。

## LifeKeeper の機能

| 項目 | 説明 |
|---|---|
| ライセンス | LifeKeeper を使用するには、各サーバーに一意の実行時ライセンスキーが必要です。これは物理サーバーおよび仮想サーバーの両方に適用されます。ライセンスキーは、LifeKeeper Core ソフトウェア、および LifeKeeper リカバリキットの各パッケージにそれぞれ必要です。インストールスクリプトで、サーバーの Host ID を取得して表示するラインセンスユーティリティパッケージをインストールします。ライセンスがインストールされた後、ユーティリティが、使用可能な Entitlement ID、または Host ID (Entitlement ID が使用できない場合) を返します。ホスト ID およびソフトウェアに付属のアクティベーション ID を使用して **SIOS Technology Corp.** の **Web サイト** からライセンスキーを取得してください。 |
| 大型クラスタのサポート | LifeKeeper は、最大 32 台のサーバーを持つ大型クラスタの設定をサポートします。ただし、LifeKeeper 以外の多くの要因が、クラスタ内でサポートされるサーバーの台数に影響することがあります。この要因として、ストレージの相互接続、オペレーティングシステム、ストレージソフトウェアの制限などがあります。サポートされる最大クラスタサイズを調べるには、ベンダ固有のハードウェアとソフトウェアの設定情報を参照してください。 |
| 国際化とローカライズ | LifeKeeper for Linux v5.2 以降は、リソース名とタグ名でのワイド / マルチバイト文字の使用をサポートしていますが、ネイティブの言語メッセージサポートは含まれていません。Java のプロパティファイルのロケール固有バージョンを作成することにより、LifeKeeper の GUI をローカライズできますが、現在フルにローカライズされているのは英語バージョンのみです。ただし、LifeKeeper の GUI に表示される多くのメッセージは LifeKeeper Core から来ているので、GUI のローカライズは、ユーザーにとって、Core ソフトウェアがフルにローカライズされるまでの単なる部分的な解決法です。<br>追加情報については、制限または既知の問題 の「**言語環境の影響**」も参照してください。 |
| **LifeKeeper の MIB ファイル** | LifeKeeper は、LifeKeeper クラスタ内で発生するイベントを記述する SNMP トラップを送出するように設定できます。この機能の設定に関する詳細については、lk_configsnmp(8) のマニュアルページを参照してください。LifeKeeper のトラップを記述する MIB ファイルは、*/opt/LifeKeeper/include/LifeKeeper-MIB.txt* に記載されています。 |
| **Watchdog** | LifeKeeper は、Watchdog 機能をサポートしています。この機能は、SIOS Technology Corp. により Red Hat EL 6 + softdog 、および Red Hat EL 7 + softdog でテスト済みです。 |
| **STONITH** | LifeKeeper は、STONITH 機能をサポートしています。この機能は、SIOS Technology Corp. により IBM x3550 x86_64 アーキテクチャ上の SLES 11、および RHEL5.5 の 64-ビットでテスト済みです。 |
| **XFS ファイルシステム** | XFS ファイルシステムは、ファイルシステムのチェックと修正に fsck ユーティリティを使用しません。その代わりに、ログの再生をマウントに依存します。整合性の問題についての懸念がある場合は、システム管理者がファイルシステムを out of service にしてシステムをアン |

| | |
|---|---|
| | マウントし、`xfs_check(8)` と `xfs_repair(8)` を実行して問題を解決する必要があります。 |
| **IPv6** | SIOS は、ip コマンドの使用に移行し、`ifconfig` コマンドを使用しなくなりました (詳細については IPv6 の既知の問題 を参照)。 |

## チューニング

| 項目 | 説明 |
|---|---|
| **IPC セマフォと IPC 共有メモリ** | LifeKeeper には、プロセス間通信 (IPC) セマフォと IPC 共有メモリが必要です。以下の Linux カーネルオプションの Red Hat のデフォルト値は、 */usr/include/linux/sem.h* にあり、LifeKeeper の多数の設定をサポートするのに十分な値です。<br><br>必要なオプション **Red Hat 7.0** のデフォルト値<br>SEMOPM 14 32<br>SEMUME 20 32<br>SEMMNU 60 32000<br>SEMMAP 25 32000<br>SEMMNI 25 128 |
| **システムファイルテーブル** | LifeKeeper がバックアップシステムに正常にフェイルオーバするためには、システムリソースが使用可能である必要があります。例えば、システムファイルテーブルがフルの場合、LifeKeeper が新しいプロセスを開始してリカバリを実行することができない可能性があります。エンタプライズパッチを持つカーネル (LifeKeeper がサポートするものを含む) では、file-max、つまりシステムで開いているファイルの最大数は、デフォルトでシステムメモリサイズの 1/10 に設定されます。これは、LifeKeeper の多数の設定をサポートするのに十分な値です。file-max 値をデフォルト値よりも低く設定すると、予期しない LifeKeeper の障害が発生することがあります。<br><br>file-max 値は、以下のコマンドで取得できます。<br><br>`cat /proc/sys/fs/file-nr`<br><br>このコマンドは、3 つの値を返します。1 番目の値はファイルテーブルのエントリのこれまでの最大値 (システムがこれまでに検出した最大値)、2 番目の値は現在のファイルテーブルのエントリ数、3 番目の値は file-max 値です。<br><br>file-max を調整するには、/etc/sysctl.conf の「fs,file-max」値を追加 (または変更) し (フォーマットについては sysctl.conf(5) を参照)、<br><br>`sysctl -p`<br><br>次にこのファイルを実行して、システムを更新します。/etc/sysctl.conf の値は、再起動後も保持されます。 |

# LifeKeeper の動作

| 項目 | 説明 |
|---|---|
| カーネルデバッガ (kdb) | LifeKeeper が保護するサーバーでカーネルデバッガ ( **kdb** ) を使用する前に、そのサーバーで LifeKeeper をシャットダウンするか、LifeKeeper が保護するリソースをバックアップサーバーにスイッチオーバする必要があります。LifeKeeper の SCSI 予約デーモン ( **lkscsid** ) を有効にした状態で (デフォルトで有効)、 **kdb** を使用すると、予期しないパニックが発生することがあります。 |
| ロックしている共有デバイスでのシステムパニック | LifeKeeper はロックを使用して、共有 SCSI バス上にある他のサーバーがアクセスしないように共有データを保護します。他のサーバーがデバイスをロックしたことにより LifeKeeper がデバイスにアクセスできない場合、致命的なエラーが発生し、即座に対処する必要があります。対処しない場合、データが破損するおそれがあります。この条件が検出された場合、LifeKeeper はシステムにパニックを発生させる機能を有効にします。<br><br>共有デバイスが予約された状態で、LifeKeeper が「`lkcli stop -f`」の様にリソースを停止させずに LifeKeeper のデーモンのみを停止した場合、他のサーバーがリソースを復旧するときに LifeKeeper のロックメカニズムによりカーネルのパニックがトリガされることがあります。この方法で LifeKeeper を停止する前に、リソースをすべて out-of-service にする必要があります。 |
| **nolock** オプション | NFS 領域でロックを使用するアプリケーションを使用し、SPS の推奨するマウントオプションを使用する場合は、nolock を追加する必要があります。<br>`rw,nolock,bg,hard,tcp,nfsvers=3,timeo=600,rsize=32768,wsize=32768,actimeo=0` |
| **Out-of-Service** 階層の復旧 | LifeKeeper サーバーの障害発生後のリカバリの一部として、障害が発生したサーバーに設定されているリソース階層のうち、障害発生時にいずれかのサーバーで *in-service* ではないものは、その時点で優先順位が最高の *alive* のサーバーで復旧されます。これは、障害が発生したサーバー、復旧中のサーバー、クラスタ内の他のサーバーを含め、 *out of service* の階層が最後にどこで in service だったかには無関係です。 |
| **Linux** ファイアウォールとの共存 | ファイアウォールが OS インストール時に有効になります。インストールの完了後、ファイアウォールを変更する必要があります。<br><br>ホストのファイアウォールが有効の場合、LifeKeeper は機能します。ただし、絶対に必要な場合を除き、ファイアウォールを無効にし、LifeKeeper で保護するリソースは別の保護ファイアウォール内に配置することをお勧めします。<br><br>LifeKeeper がファイアウォールを有効にしたホストと共存させる必要がある場合、LifeKeeper はコミュニケーションパス、GUI、IP、およびデータレプリケーションに特定のポートを使用することに注意してください。Linux のファイアウォール機能を使用する場合、LifeKeeper が使用している特定のポートを開放する必要があります。詳細については、ファイアウォールを使用した状態での LifeKeeper の実行 を参照してください。 |
| **SELinux** との共存 | SELinux モードが「有効」の場合、LifeKeeper はインストールも機能もしません。 SELinux を無効にするには、お使いの OS ディストリビューションのマニュアルを参照してください。 |

| | |
|---|---|
| | SELinux の permissive モード。 SAP 環境で必要な場合を除いて、SIOS は SELinux を permissive モードで使用することは推奨しません。クラスターで実行中のアプリケーションが、permissive モードの SELinux をサポートしていることを確認してください。次のアプリケーションリカバリーキットはテスト済みです： SAP、SAP MaxDB、Sybase、Oracle、DB2、NFS、DataKeeper、NAS、EC2、IP、FileSystem、MQ<br><br>AppArmor （このセキュリティモデルを使用するディストリビューションの場合）は有効にすることができます。 |
| **Suid マウントオプション** | suid マウントオプションは、 *root* としてマウントするときのデフォルトであり、マウントコマンドにより */etc/mtab* に書き込まれることはありません。LifeKeeper 環境では、suid マウントオプションは不要です。 |

## サーバーの設定

| 項目 | 説明 |
|---|---|
| **BIOS のアップデート** | 使用可能な最新の BIOS を常にすべての LifeKeeper サーバーにインストールする必要があります。 |

# LifeKeeper 8.2.0 以降の GUI 要件

LifeKeeper GUI クライアントでユーザーを正常に認証するには、64 ビットバージョンの PAM 関連のパッケージがすべて必要です。

# [Confirm Failover] と [Block Resource Failover] の設定

以下の説明、例、および考慮事項をよく読んで理解してから、お使いの LifeKeeper 環境で **[Confirm Failover]** または **[Block Resource Failover]** を設定してください。これらの設定は、コマンドライン、または LifeKeeper の GUI の **[Properties]** パネルから使用できます。

## Confirm Failover On:

定義 – システム *A* から システム *B* へのフェイルオーバの手動確認を有効にします ( システム *A* はプロパティが [Properties] パネル に表示されるサーバーで、 システム *B* はチェックボックスの左にあるシステム)。あるシステムでこのオプションをオンに設定した場合、障害発生が検出されたシステムについて LifeKeeper がフェイルオーバリカバリを実行するには、システム管理者による手動確認が必要になります。

フェイルオーバを確認するには、**`lk_confirmso`** コマンドを使用してください。デフォルトでは、このコマンドを実行するまで管理者には 10 分の猶予時間があります。この時間は、 */etc/default/LifeKeeper* の **CONFIRMSOTO** 設定で変更できます。管理者が 10 分以内に **`lk_confirmso`** コマンドを実行しない場合、フェイルオーバは続行されるか、ブロックされます。デフォルトでは、フェイルオーバが続行されます。この動作は、 */etc/default/LifeKeeper* の **COMFIRMSODEF** 設定で変更できます。

例：自動フェイルオーバをすべてブロックする場合は、 **[Properties]** パネルの **[Confirm Failover On]** オプションを設定し、さらに **CONFIRMSODEF** を **1** (フェイルオーバをブロック)、 **CONFIRMSOTO** を **0** (フェイルオーバ動作が決定されるまで待機しない) に設定してください。

この設定を選択するタイミング:

この設定は、設定に冗長ハートビートコミュニケーションパスを含まない多くのディザスタリカバリ、その他の WAN 設定で使用されます。

あるサーバーで **[Properties]** ページを開き、 **[Confirm Failover]** フラグ をオンに設定するサーバーを選択してください。

## Block Resource Failover On:

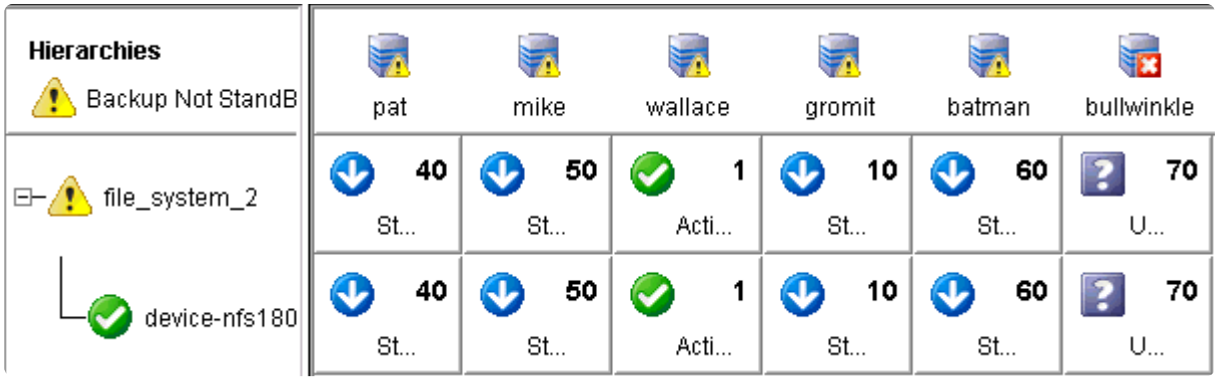
定義 - デフォルトでは、リソースのすべての障害について復旧イベントが発生し、ローカルシステムの障害リソースの復旧が試行されます。ローカルリカバリが失敗した場合、または有効になっていない場合は、リソースが定義されている、優先順位が次に最も高いシステムに、LifeKeeper がローカル履歴を転送します。ただし、宛先として指定したシステムでこの設定を選択している場合、リソース障害に起因するリソースの転送はすべてブロックされます。

この設定が有効の場合、以下のメッセージがログに記録されます。

Local recovery failure, failover blocked, MANUAL INTERVENTION REQUIRED

# 5.4.4.2. クラスタの例

## 拡張したマルチクラスタの例



# 5.4.4.3. ダイアログ

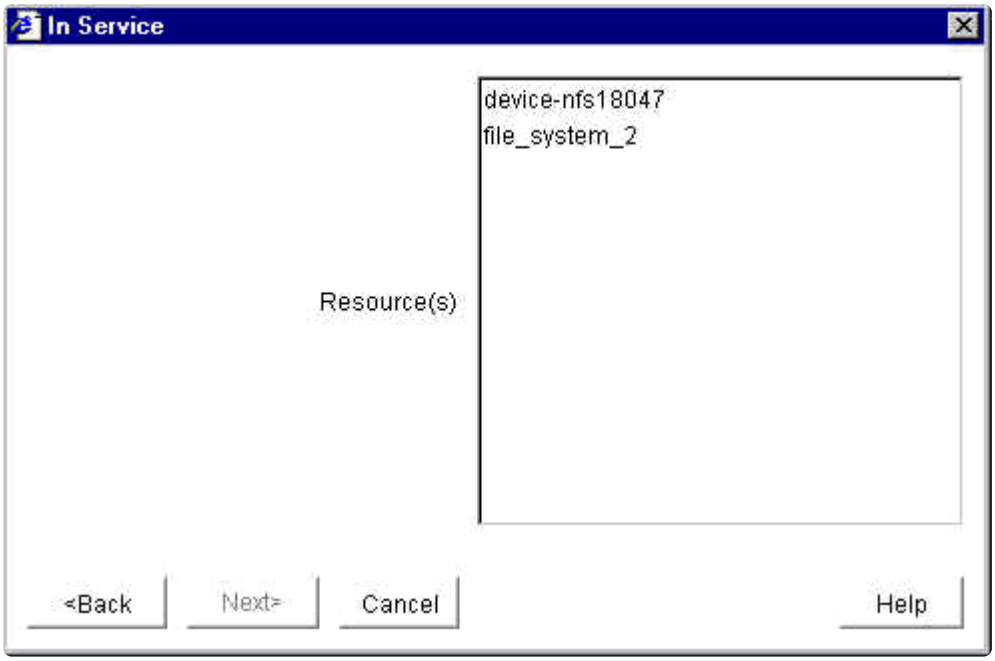
- [In Service] ダイアログ
- [Out-of-Service Resource Properties] - [EquivalenciesResource] ダイアログ
- [Password] ダイアログ
- [Resource Properties] - [Equivalencies]
- [Resource Properties] - [General]
- [Resource Properties] - [Relations]
- [Server Properties] - [Commpath]
- [Server Properties] - [General]
- [Server Properties] - [Resource]

# [In Service] ダイアログ

**Select a Server** - 最初のダイアログのドロップダウンリストボックスには、LifeKeeper クラスタ内にあるサーバ名があります。 in service にするリソースインスタンスが存在する サーバ を選択してください。 **[Next]** ボタンをクリックして、次のダイアログに進んでください。

注記: 右側のペインにあるサーバ固有のリソースを右クリックして [In Service] の作業を開始した場合、このダイアログと次のダイアログは表示されません。これは、 in service にするサーバとリソースを指定したからです。

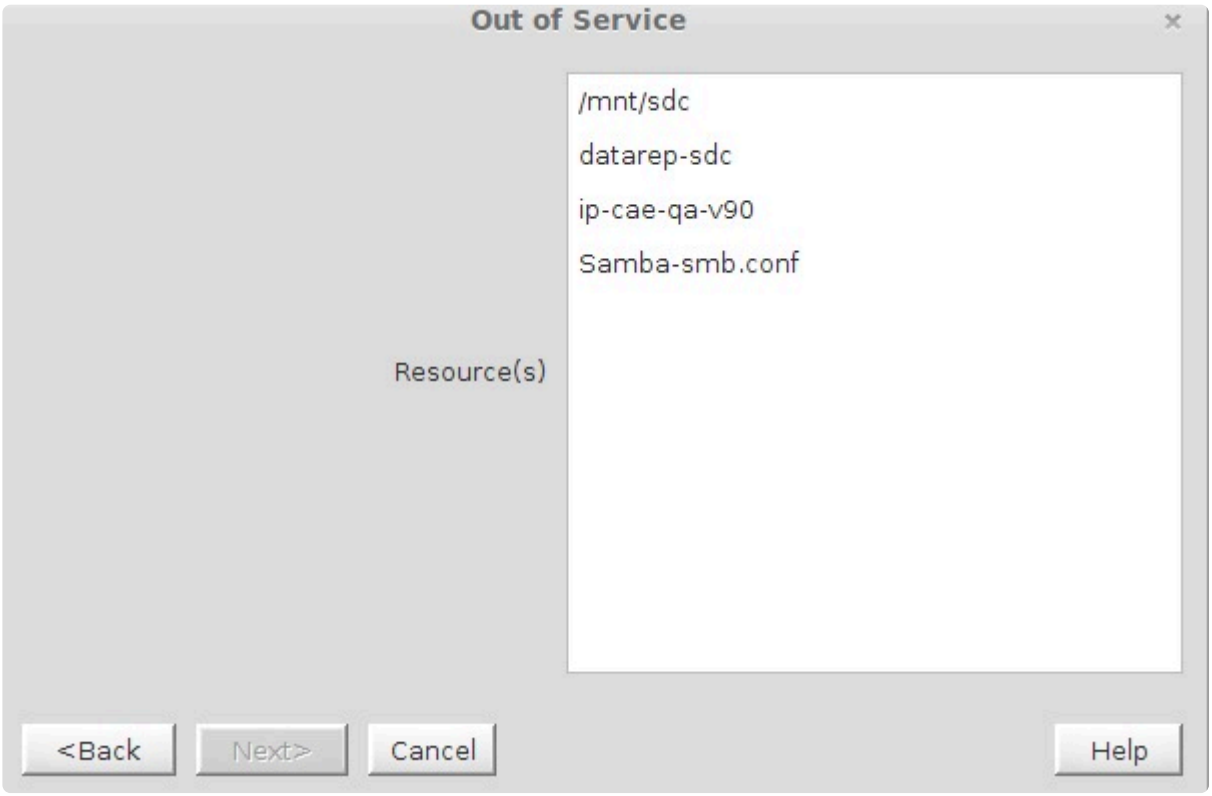
**Select a Resource** - 2 番目のダイアログのドロップダウンリストボックスには、前のダイアログで選択したサーバ上で使用できるすべてのリソースの名前があります。 in service にするリソースを選択してください。




注記: 左側のペインにあるグローバルリソースを右クリックして [In Service] の作業を開始した場合、このダイアログは表示されません。これは、 in service にするリソースを指定したからです。

# [Out-of-Service Resource] ダイアログ

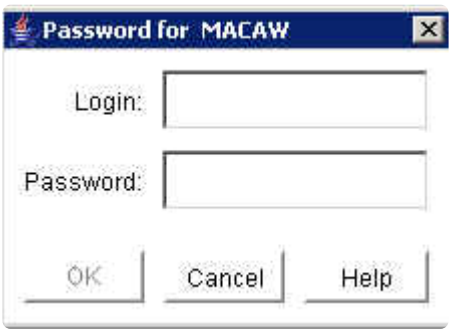
**Select a Resource** - このダイアログのリストには、LifeKeeper クラスタ内で in service になっているすべてのリソースの名前があります。out of service にするリソースを選択してください。




注記 : 左側のペインにある in service のグローバルリソース、または右側のペインにある in service のサーバに固有のリソースインスタンスを右クリックして [Out-of-Service] の作業を開始した場合、このダイアログは表示されません。これは、out of service にするリソースを指定したからです。

# [Password] ダイアログ

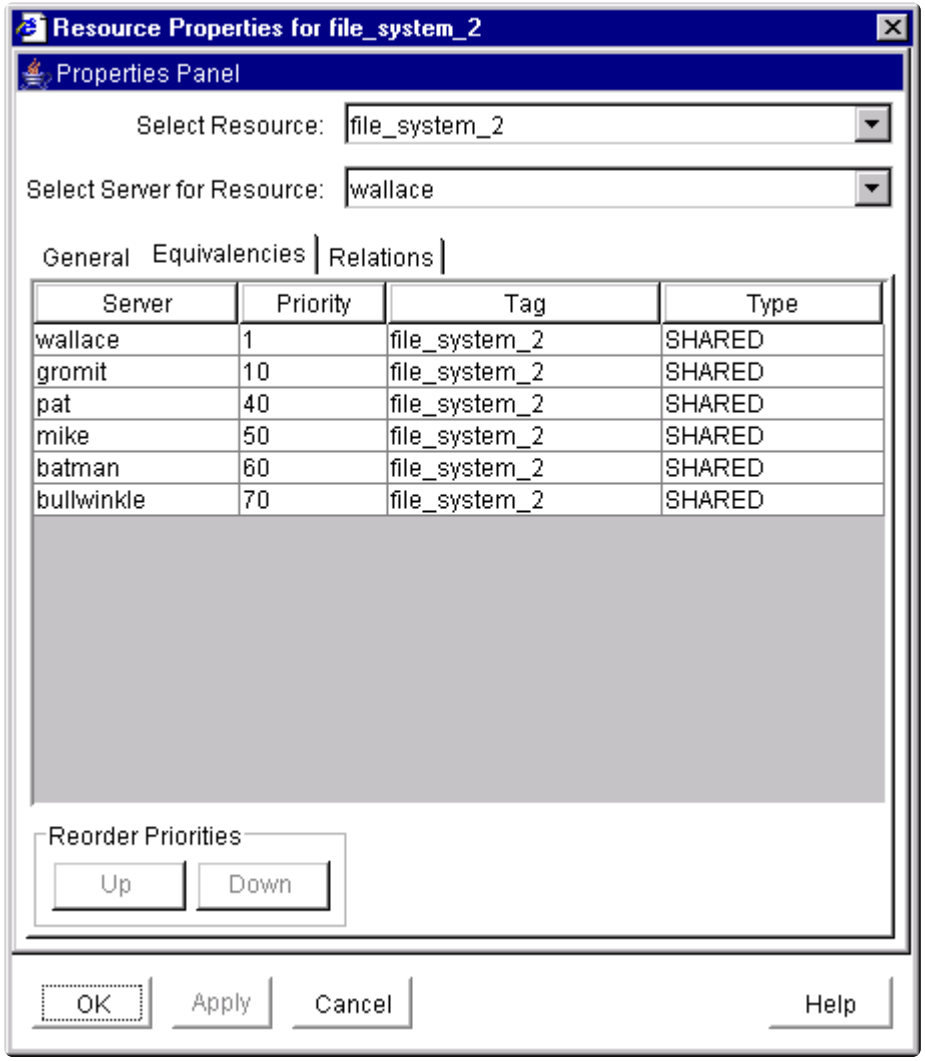
**注記:** このダイアログは、[Cluster Connect] ダイアログ に最初に入力したログイン名またはパスワードが正しくない場合に表示されます。



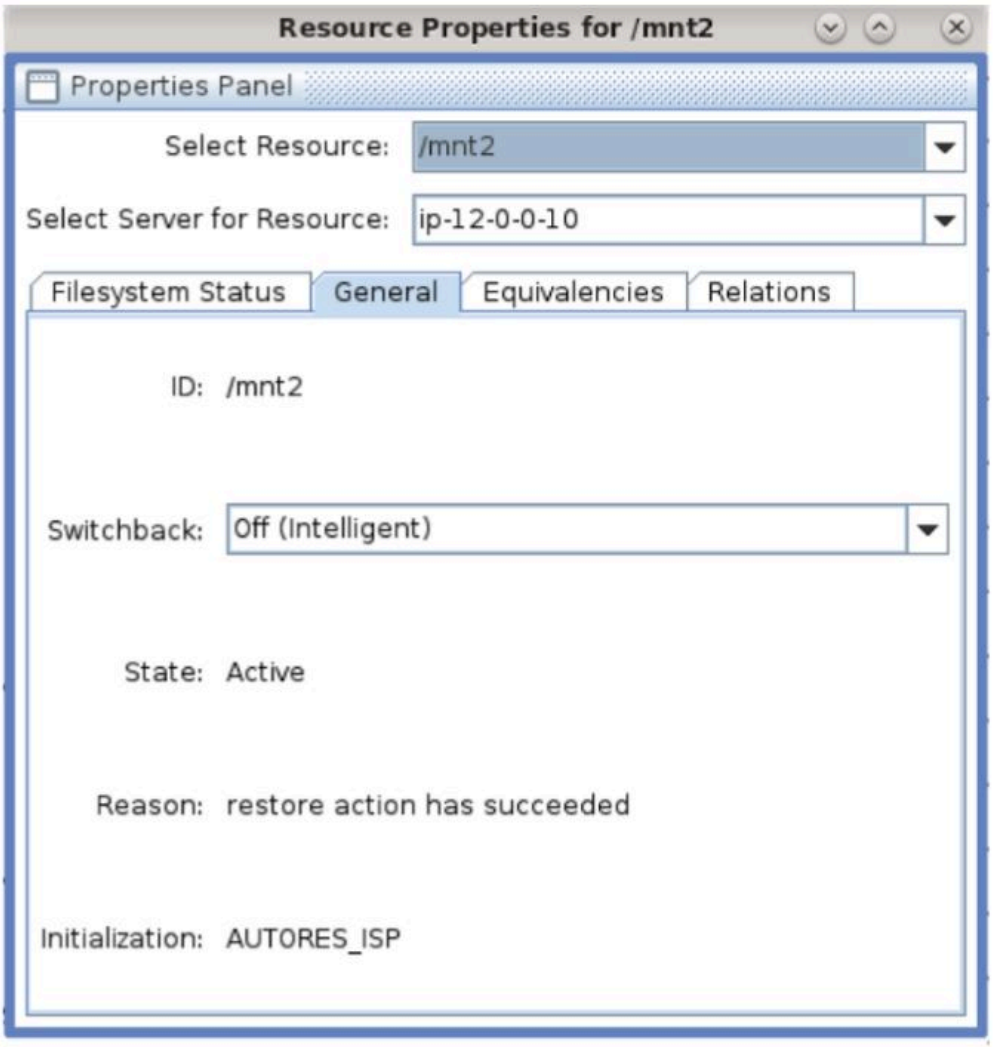

**Login** - 指定サーバに LifeKeeper 認証情報を持つユーザのログイン名。

**Password** - サーバで指定ログインを認証するパスワード。

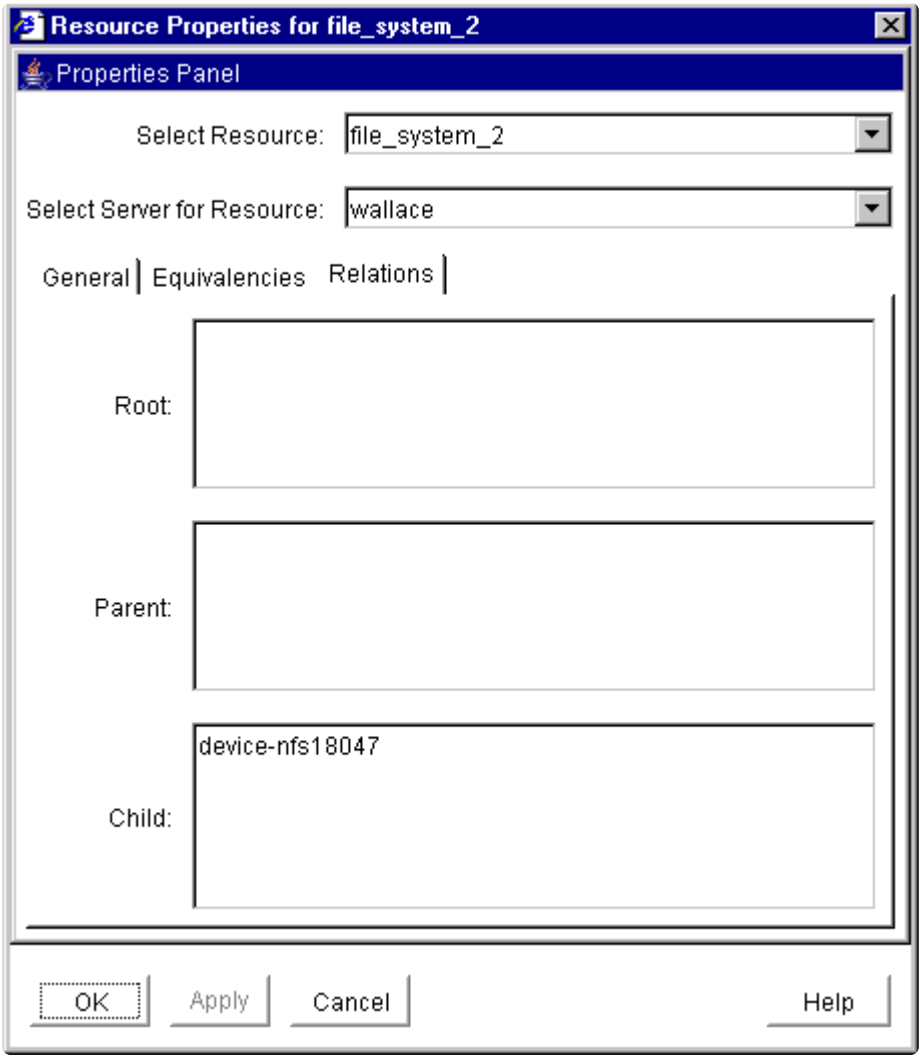
# [Resource Properties] - [Equivalencies]



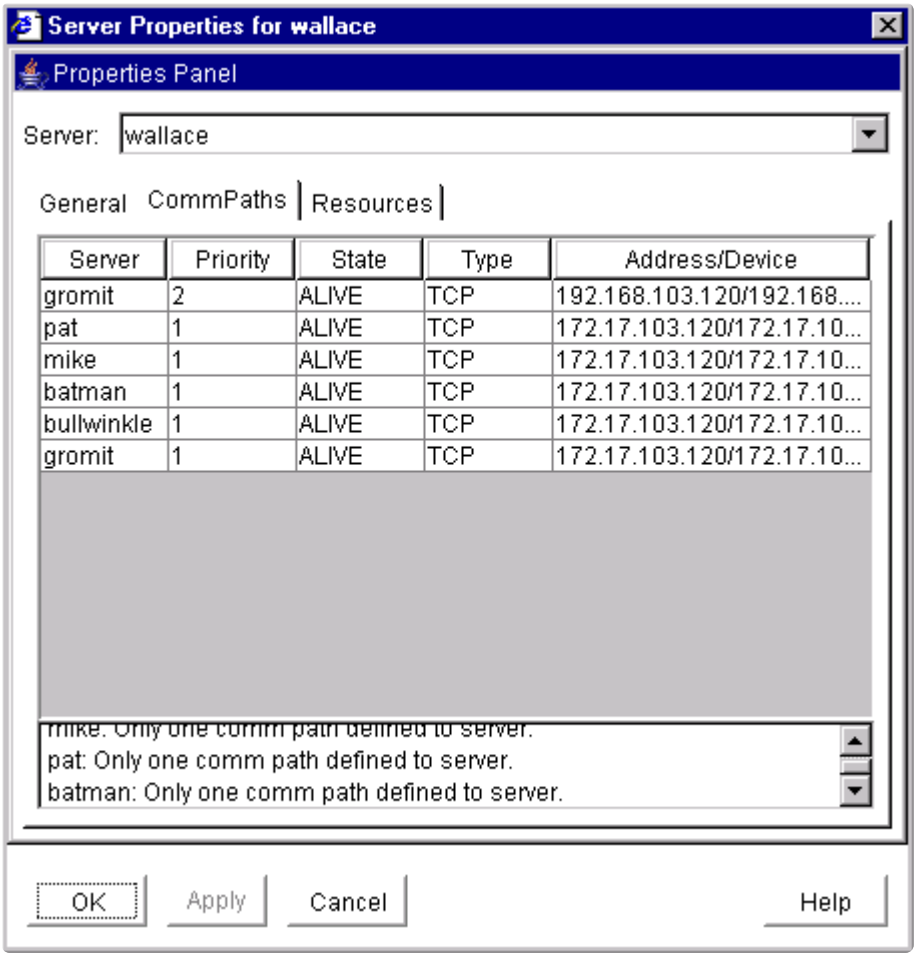
# [Resource Properties] - [General]



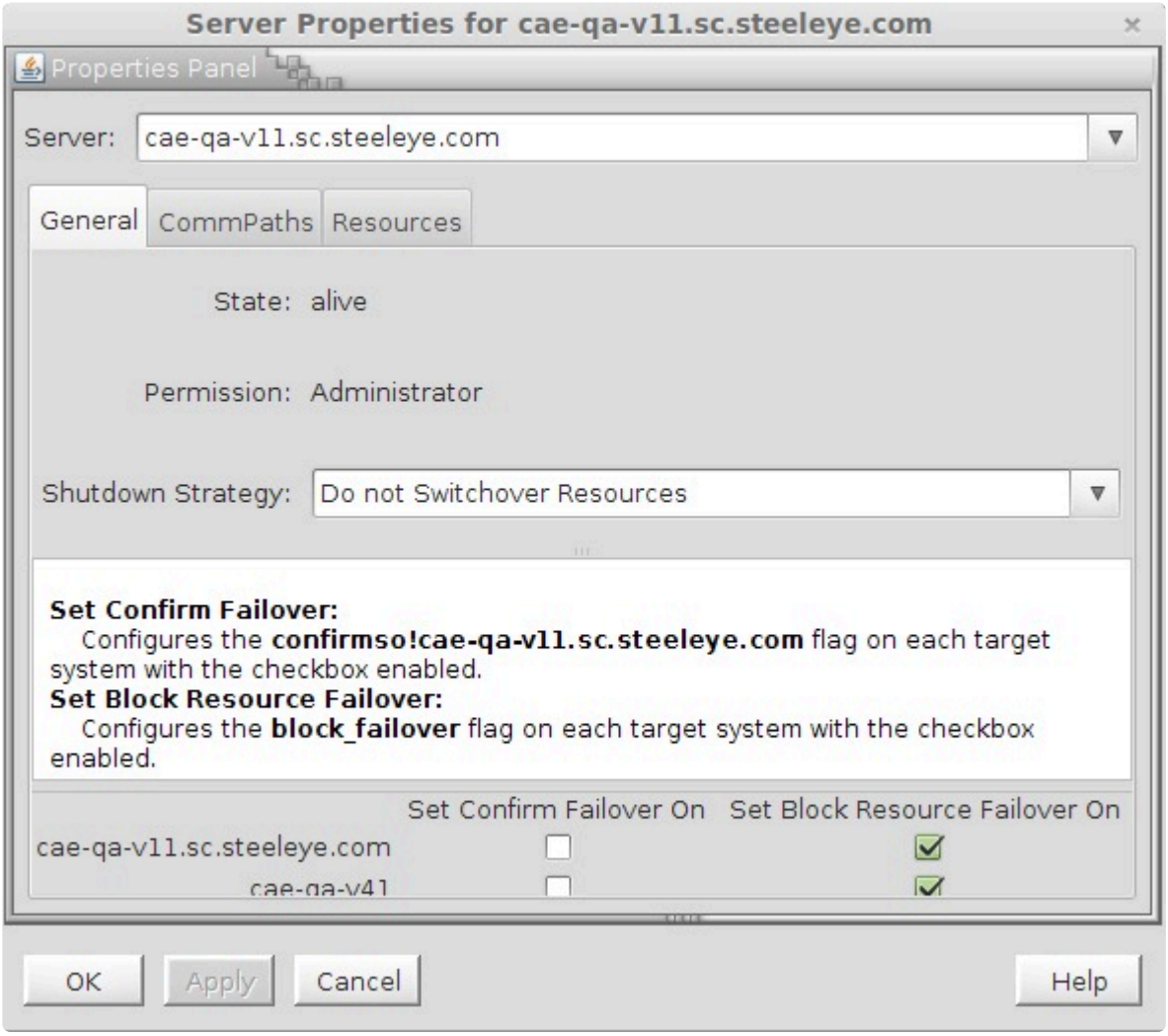
# [Resource Properties] - [Relations]



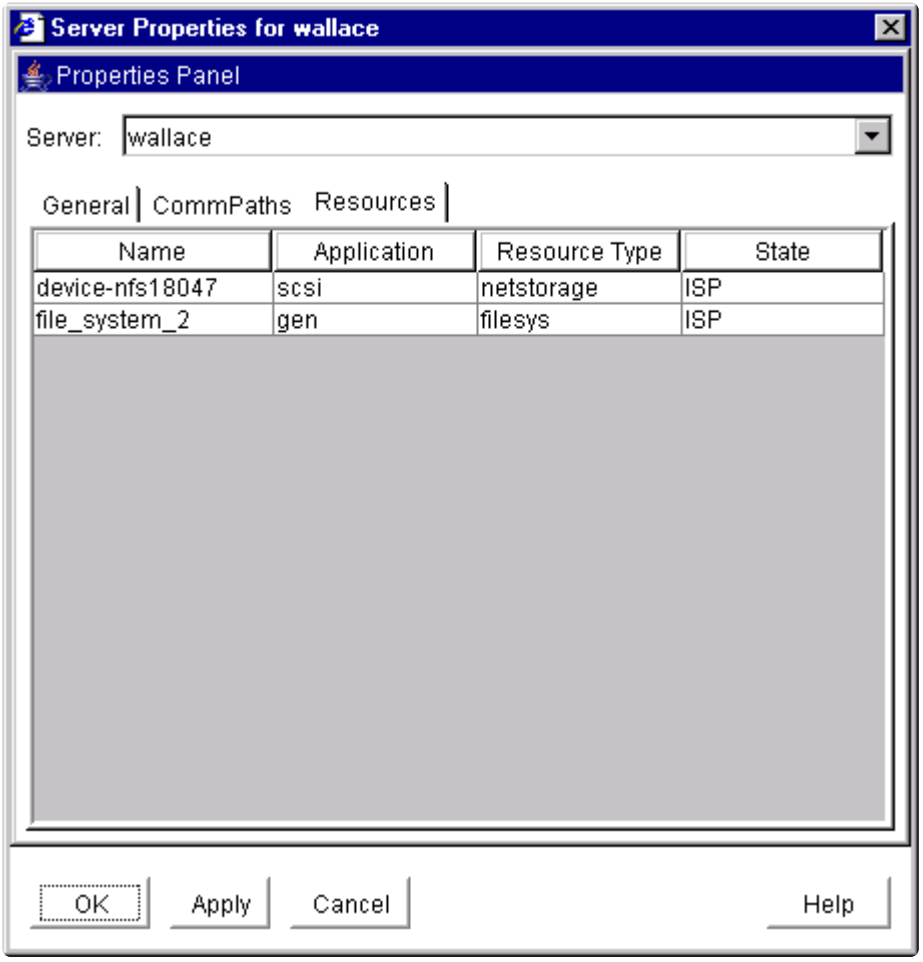
# [Server Properties] - [Commpath]



# [Server Properties] - [General]



# [Server Properties] - [Resource]



# 5.4.5. トラブルシューティング

メッセージカタログ には、操作、管理、GUI など、SIOS Protection Suite for Linux の使用中に遭遇する可能性のあるすべてのエラーコードが列挙されています。また、エラーコードの原因に関する追加の説明や、問題解決のために必要な処置についても、必要に応じて記載されています。この完全なリストを検索すると、受信したエラーコードを見つけることができます。また、以下の個別のメッセージカタログに直接アクセスすることもできます。

- Core メッセージカタログ
- DB2 メッセージカタログ
- DMMP Kit メッセージカタログ
- EC2 メッセージカタログ
- ファイルシステムキットメッセージカタログ
- Gen/App Kit メッセージカタログ
- IP Kit メッセージカタログ
- Oracle Listener Kit メッセージカタログ
- Oracle Kit メッセージカタログ
- SCSI Kit メッセージカタログ
- DataKeeper Kit メッセージカタログ
- QSP Kit メッセージカタログ
- GUI メッセージカタログ

上記のメッセージカタログに加え、以下のトピックでも、直面する可能性がある問題や制限事項のトラブルシューティングについて詳細を説明します。

---

- SPS が開始するフェイルオーバの一般的な原因
- 既知の問題と制限
- GUI トラブルシューティング
- コミュニケーションパスの稼働と停止
- 不完全なリソースの作成

不完全なリソースの優先順位の変更

階層の設定中に共有ストレージが見つからない

LifeKeeper サーバ障害からの復旧

停止できないプロセスからの復旧

手動リカバリ時のパニックからの復旧

Out-of-Service 階層の復旧

リソースタグ名の制限

シリアル (TTY) コンソールの警告

システムが init 状態 S に遷移しているという警告

共有ストレージでスレッドがハングしているというメッセージ

# 5.4.5.1. ソリューション

## - Storage Quorum -

**SPS for Linux: Storage Quorumを使用する際のハートビートに関する推奨事項**

ソリューションの詳細

問題:
Storage Quorumを使用する際のハートビートに関する推奨事項について

ソリューション:

**Storage Quorum** を使用する 場合、 **SIOS** では **LCMNUMHBEATS** を増やして 、パスが障害と判定されるまでの時間を長くすることを推奨しています。これにより、タイムアウト期間がデフォルトの15秒から45秒に変更されます。

**LCMNUMHBEATS** を **9** に変更します

```
LCMNUMHBEATS=9
```

コアパラメータを変更することになるので、LifeKeeperを再起動する必要があります。

**サービスの停止時間を最小限に抑える** には,“`lkstop -f`“ を使用するとリソースを実行したままにすることができます。
LifeKeeperが停止している間は、保護されたリソースの障害は検出または対処されません。

```
# lkstop -f

# lkstart
```

> 注意: これらの変更は、クラスター内の各ノードで行う必要があります。

**SPS for Linux:** ノード間の通信が失われたときに、**Storage Quorum**はフェイルオーバーを防止できない

ソリューションの詳細

問題:
Storage Quorumの構成が不完全なため、通信処理が失われたときに障害が発生する

ソリューション:

`qwk_storage_init` をクラスター内の各ノードで実行する前に、クラスターノード間のすべてのコミュニケーションパスを作成し、 **“ALIVE”** にする必要があります。

これがうまくいかない場合 は、 すべてのコミュニケーションパスが **ALIVE** になってから 、 以下の

コマンドを実行して Storage Quorum 構成を再初期化してください。

```
# /opt/LifeKeeper/bin/qwk_storage_exit

# /opt/LifeKeeper/bin/qwk_storage_init
```

> 注意: これにより、Storage Quorum が再初期化されます。

**SPS for Linux: Amazon S3 ストレージを使用する場合の storage quorum パラメーターの推奨値**

ソリューションの詳細

問題:
Amazon S3 ストレージを使用する場合の storage quorum パラメーターの推奨値について

ソリューション:

ドキュメントに従って storage quorum を設定したら、ハートビートのパラメーターを設定します。
詳細はこちら

**いくつかの方法** で、あなたの環境においてデフォルトの設定が十分なものになっているか確認することができます。

2つの主要なパラメーターがあります。

- `QWK_STORAGE_HBEATTIME` (デフォルト値: 6) – QWK オブジェクトを読み書きする間隔を **秒単位** で指定し ます。
- `QWK_STORAGE_NUMHBEATS` (デフォルト値 :4) – Witness チェックで対象ノードに障害が発生していると判断するための値を指定します。QWK オブジェクトの読み込みにおいて、このパラメーターに指定した **回数以上更新が停止** していると、対象ノードに障害が発生していると判断します。

Amazon S3 bucketを使用してQWK オブジェクトを保存する場合 (QWK_STORAGE_TYPE=aws_s3)、ご利用の環境において以下のコマンドを実行して接続に **問題がないこと** を確認することを **推奨**します 。

1. **ping s3.amazonaws.com** を実行し、応答時間が1秒以下であることを確認してください。この操作により、EC2 ノードからグローバル AWS ドメインへの正常な **接続性が確保** されます。

> 注意: S3 はグローバルサービスであっても、S3 バケットはリージョンにあります。

2. **ping <bucketname>.s3.amazonaws.com** を実行することで、ホストするS3 サービスの IP ア

ドレスを解決します。こちらも1秒以下である必要があります。

その他に検討する事は、このノードでのS3のアクティビティ全体に対して渡される **データの量** です。ファイル転送で確認することもできます。ping を使用して応答時間の測定をすることも可能です。(上記の ping フォーマット) を参照してください。

> **注意:** AWS においては、S3 はグローバルサービスですが、EC2 ノードはリージョンサービスです。S3 URI (Amazon S3 storage Uniform Resource Identifier またはそのウェブアドレス) は、 **特定のリージョン** にあることがあります。

トラフィックがある場合とない場合を比較し、上記のハートビートの回数 (QWK_STORAGE_NUMHBEATS) および ハートビートの時間 (QWK_STORAGE_HBEATTIME) を調整することが可能です。

上記 パラメーター は **qwk_storage_init** を実行する前に指定する必要があります。

**ほとんどの場合** また ping 応答が1秒以下の場合はデフォルト値で十分ですが、S3 が遅延するもしくはデグレードしているように見える場合は、 `QWK_STORAGE_HBEATTIME` の値を増やすことをお勧めします。デフォルトの 6 から 7 へ変更してください。 これによりタイムアウト値が 24 秒 (6 × 4) から 28 秒 (7 × 4) に増加します。 タイムアウト値を 30 秒以上にすることは推奨しません。

デフォルト値を変更する場合は、 **クラスター内の各ノード** で変更を行い、すべてのコミュニケーションパスが **ALIVE** になっている状態で、クラスター内の各ノード で以下のコマンドを実行して Storage Quorum 構成を再初期化してください。

```
# /opt/LifeKeeper/bin/qwk_storage_exit

# /opt/LifeKeeper/bin/qwk_storage_init
```

# 5.4.5.2. SPS が開始するフェイルオーバの一般的な原因

障害が発生した場合、SPS には2つのリカバリ方式があります。ローカルリカバリとサーバ間リカバリです。ローカルリカバリが失敗した場合、「フェイルオーバ」が実行されます。フェイルオーバは、アクティブだったアプリケーション、サーバ、システム、ハードウェアコンポーネント、ネットワークの障害や異常終了が発生したときのバックアップサーバへの自動的な切り替えと定義されます。フェイルオーバとスイッチオーバは基本的には同じ動作ですが、スイッチオーバが人の介入を必要とするのに対し、フェイルオーバは自動で、通常は警告なしで実行されます 。この自動的なフェイルオーバは、さまざまな理由で発生します。以下は、SPS が開始するフェイルオーバの最も一般的な例の一覧です。

## サーバレベルでの原因

### サーバの障害

SPS には、設定内の各サーバに、ペアのサーバが動作していることを定期的に通知する組み込みのハートビート信号があります。サーバがハートビートメッセージを受信しなかった場合に、障害として検出されます。

- プライマリサーバが電源を喪失するか、電源がオフになる。
- 過負荷による CPU 使用率 -- 非常に高負荷の I/O の下では、これらの遅延と低メモリ状態によってシステムが無応答になり、SPS がこれをサーバのダウンとして検知し、フェイルオーバを開始することがあります。
- Quorum/Witness - プライマリサーバが Quorum を失った場合、Quorum/Witness の I/O フェンシングメカニズムの一環として、( 設定に基づいて )「 *fastboot* 」、「 *fastkill* 」、または「 *osu* 」が実行され、フェイルオーバが開始されます。フェイルオーバ時を決定する際、プライマリサーバで障害が発生しクラスタを構成できなくなったことが確認できた場合にのみ、witness サーバによりバックアップサーバ上でリソースを in service にすることができます。全体のアクセスや、パフォーマンス、in-service のノードが影響を受けない場合は、これによってノード間で発生する単純な通信障害から発生するフェイルオーバを回避します。

| 関連トピック |
| --- |
| サポートストレージ一覧 |
| サーバ障害リカバリのシナリオ |
| LifeKeeper ハートビートの調整 |
| Quorum/Witness |

## 通信障害 / ネットワーク障害

SPS は、サーバ間で 5 秒ごとにハートビートを送信します。通信障害によって 2 回のハートビートが途絶し、3 回目のハートビートで再開した場合、SPS は一切処置を行いません。コミュニケーションパスの切断継続時間がハートビート 3 回分になった場合は、SPS はそのコミュニケーションパスを切断と判定します。ただし、他方の冗長的なコミュニケーションパスも切断と判定されるまではフェイルオーバを開始しません。

- プライマリサーバへのネットワーク接続の喪失。
- ネットワークの遅延。
- TCP コミュニケーションパス上のネットワークトラフィックが大きくなると、偽のフェイルオーバや LifeKeeper の初期化の問題など、予期せぬ動作が生じる可能性があります。
- STONITH を使用しているときに SPS がノードとの通信障害を検出すると、そのノードの電源が切断され、フェイルオーバが発生します。
- NIC の障害。
- ネットワークスイッチの障害。
- 手動によるネットワーク接続の解除。

| 関連トピック |
| --- |
| コミュニケーションパスの作成 |
| LifeKeeper ハートビートの調整 |
| ネットワーク設定 |
| ネットワーク設定の確認 |
| SNMP 経由の LifeKeeper イベントの転送 |
| ネットワーク関連のトラブルシューティング |
| ファイアウォールを使用した状態での LifeKeeper の実行 |
| STONITH |

## スプリットブレイン

単一のコミュニケーションパスを使用しているときに、そのコミュニケーションパスで障害が発生した場合、複数のシステム上で同時に SPS の階層が in service になることがあります。これは、偽のフェイルオーバまたは「 **スプリットブレイン** 」シナリオと呼ばれます。「スプリットブレイン」シナリオでは、各サーバが、アプリケーションを制御できると認識しているため、データにアクセスしようとしたり共有ストレージデバイスにデータを書き込もうとする場合があります。スプリットブレインシナリオを解決するため

に、SPS では、サーバの電源をオフにしたり、再起動したり、階層を out-of-service にすることで、すべての共有データに対するデータの整合性を保証することができます。また、TCP コミュニケーションパス上のネットワークトラフィックが大きくなると、偽のフェイルオーバや LifeKeeper が適切に初期化できなくなるなど、予期しない動作が生じる可能性があります。

以下に、スプリットブレインが発生する可能性のあるシナリオを示します。

- 上記のいずれかの通信障害
- LifeKeeper の不正なシャットダウン
- サーバリソースの枯渇
- すべてのネットワークパスの喪失
- DNS またはその他のネットワークの問題
- システムのロックアップ / 解除

# リソースレベルでの原因

SPS には、個々のアプリケーションおよび関連アプリケーションのグループを監視する機能があり、定期的にローカルリカバリを実行したり、保護下のアプリケーションに障害が発生したときに通知することができます。関連し合うアプリケーションの例としては、主アプリケーションが下位のストレージまたはネットワークリソースに依存する階層などがあります。SPS は、これらの保護下のリソースのステータスと健全性を監視します。SPS は、リソースが障害状態にあると判断すると、外部の介入なしに現在のシステム (in-service のノード) でリソースまたはアプリケーションをリストアしようとします。このローカルリカバリが失敗した場合、リソースのフェイルオーバが開始されます。

## アプリケーションの障害

- アプリケーションの障害が検出されたが、ローカルリカバリプロセスが失敗。
- 削除の失敗 - リソースのフェイルオーバプロセスでは、該当する重要なアプリケーションのすべての機能を提供するために、プライマリサーバ上でのサービスから特定のリソースを削除し、選択したバックアップサーバでそのリソースを in service にする必要があります。 **この削除プロセスが失敗した場合は、プライマリサーバの再起動が実行され、** その結果、完全なサーバフェイルオーバに移行します。

  削除の失敗の例

  - ファイルシステムをアンマウントできない
  - 保護対象のアプリケーション (Oracle、mySQL、Postgres など) をシャットダウンできない

| 関連トピック |
| --- |
| ファイルシステムの健全性監視 |
| リソースエラーリカバリのシナリオ |

## ファイルシステム

- ディスクフル - SPS のファイルシステムの健全性監視は、ファイルシステムリソースのフェイルオーバになる可能性のある、ファイルシステムのディスクフル状態を検出できます。
- アンマウントされた、または不適切にマウントされたファイルシステム - in-service 状態で LK 保護下のファイルシステムをユーザが手動でアンマウントまたはオプションを変更。
- 再マウントの障害 - 以下のリストに、フェイルオーバに進行する場合がある再マウントの障害の一般的な原因を示します。
- ◦ ファイルシステムが破損している (fsck の障害)
- ◦ マウントポイントディレクトリの作成失敗
- ◦ マウントポイントがビジー
- ◦ マウントの失敗
- ◦ SPS の内部エラー

| 関連トピック |
| --- |
| ファイルシステムの健全性監視 |

## IP アドレスの障害

IP Recovery Kit によって IP アドレスの障害が検出されると、結果として生じる障害によって IP ローカルリカバリスクリプトが起動されます。SPS は最初に、その IP アドレスを現在のネットワークインターフェース上で in service に戻そうとします。ローカルリカバリの試みが失敗すると、SPS はその IP アドレスと依存関係を持つすべてのリソースのバックアップサーバへのフェイルオーバを実行します。フェイルオーバ中に、削除プロセスによって現在のサーバ上の該当する IP アドレスの設定が解除され、バックアップサーバ上でそのアドレスを設定できるようになります。 **この削除プロセスに失敗すると、システムは再起動します。**

- IP 競合
- IP コリジョン

- DNS の名前解決の障害
- NIC やスイッチの障害

| 関連トピック |
| --- |
| 切り替え可能な IP アドレスの作成 |
| IP ローカルリカバリ |

### リザベーションコンフリクト

- 保護されたデバイスへのリザベーションの喪失または盗難
- 保護されたリソースデバイスへのリザベーションまたは制御の回復が不可能 (手動のユーザ介入、HBA またはスイッチの障害が原因)

| 関連トピック |
| --- |
| SCSI リザベーション |
| リザベーションの無効化 |

## SCSI デバイス

- 保護された SCSI デバイスを開くことができない。デバイスに障害が発生しているか、システムからデバイスが削除されている可能性があります。

# 5.4.5.3. 既知の問題と制限

下記に、LifeKeeper for Linux で明らかになっている制限または既知の問題を機能領域ごとに示します。


インストール - 既知の問題と制限

LifeKeeper Core - 既知の問題と制限

インターネット / IP ライセンス - 既知の問題と制限

GUI - 既知の問題と制限

データレプリケーション - 既知の問題と制限

IPv6 - 既知の問題と制限

Apache Recovery Kit - 既知の問題と制限

Oracle Recovery Kit - 既知の問題と制限

MySQL Recovery Kit - 既知の問題と制限

NFS Server Recovery Kit - 既知の問題と制限

SAP Recovery Kit - 既知の問題と制限

LVM Recovery Kit - 既知の問題と制限

Multipath Recovery Kits (DMMP / HDLM / PPATH / NECSPS) - 既知の問題と制限

DMMP Recovery Kit - 既知の問題と制限

DB2 Recovery Kit - 既知の問題と制限

Sybase ASE Recovery Kit - 既知の問題と制限

WebSphere MQ Recovery Kit - 既知の問題と制限

# 5.4.5.3.1. インストール - 既知の問題と制限

| 説明 |
| --- |
| **LifeKeeper を正しくインストールするには、yum または zypper の機能構成が必要です**<br><br>設定が正しくない場合や非機能構成の yum または zypper では、LifeKeeper のインストールスクリプトが失敗する可能性があります。次のような出力が表示されます。<br><br>`Install LifeKeeper and dependent packages done.`<br>`Setup high availability data replication features.. done.`<br>`Setup NFS high availability features…`<br>`Configure LifeKeeper management group`<br><br>`Setup failed.`<br>`Fix the problem and try again.`<br>`sed: can't read /etc/default/LifeKeeper: No such file or directory`<br>`/tmp/mnt/setuplibs/install.sh: line 118: /config/system_information: No such file or directory`<br>`/tmp/mnt/setuplibs/install.sh: line 122: /config/system_information: No such file or directory`<br>`/tmp/mnt/setuplibs/install.sh: line 122: /config/system_information: No such file or directory`<br>`/tmp/mnt/setuplibs/install.sh: line 122: /config/system_information: No such file or directory`<br><br>さらに、/var/log/LK_install.log ファイルに次のように表示されることがあります。<br><br>`Abort, retry, ignore? [a/r/i/? shows all options] (a): a`<br>`done.` |
| **リリース 7.4 以降では、SIOS 製品 RPM パッケージの再配置はサポートされません** |
| **Linux の依存関係**<br><br>オプションの Recovery Kit を含めて SIOS Protection Suite for Linux のインストールには、いくつかの依存関係を持つパッケージが必要になります。パッケージマネージャが適切に設定されていれば、必要なパッケージはパッケージマネージャによって自動的にインストールされます。<br>自動でインストールが行えなかった場合には setup スクリプトが中断されるので、手動で依存関係のあるパッケージ (詳細は Linux の依存関係 を参照してください) をインストールした後に、setup スクリプトを再実行してください。<br><br>**注記**：依存関係にあるこれらのパッケージのインストールが正常に完了していない場合、SIOS |

Protection Suite for Linux を起動する機能、および SIOS Protection Suite for Linux の GUI をロードする機能にも影響します。

**新しいデバイスがスキャンされているときに nbd ドライバがロードされると、multipathd デーモンはエラーログにエラーを記録します**

**解決方法**：これらのエラーをログに記録しないようにするには、*/etc/multipath.conf* の blacklist に **devnode "^nbd"** を追加します。

**LK のインストール前に mksh が導入されていると、setup 実行時にパッケージ競合を検出して失敗する**

`mksh` パッケージがインストール済みの場合、 SIOS Protection Suite for Linux のセットアップスクリプトはパッケージの競合を検出して失敗します。また、システムの要件で `KornShell(ksh)` が必要になる場合は `KornShell(ksh)` パッケージを導入してください。

**対応策:** RHEL、CentOS、または Oracle Linux では、`mksh` パッケージをアンインストールし、必要に応じて `KornShell(ksh)` パッケージをインストールしてください。その後、SIOS Protection Suite for Linux のセットアップスクリプトを再実行してください。

例:

1. mksh パッケージをアンインストールしてください。
   ```
   yum remove mksh
   ```
2. ksh パッケージをインストールしてください。 (※KornShell(ksh) が必要な場合)
   ```
   yum install ksh
   ```
3. セットアップスクリプトを再実行してください。

**デーモンの予期しない終了**

Red Hat Enterprise Linux 7.2 および Red Hat 7.2 の派生システムへのアップデートを行った環境で、IPC を使用するデーモンが予期せず終了することがあります。 Red Hat Enterprise Linux 7.2 へ導入された新機能である systemd が、最後のユーザセッション終了時に全てのプロセス間通信 (IPC) に割り当てられた資源の回収を行います。 このセッションとは管理用の cron ジョブもしくは対話式のものです。このふるまいにより、同一ユーザでかつ同一資源を使用しているデーモンが意図せず終了することがあります。

この問題を回避するためには、/etc/systemd/logind.conf を編集して次の行を追加してください。
`RemoveIPC=no`

その後、次のコマンドを実行し、変更を反映させてください。

```
systemctl restart systemd-logind.service
```

これらの手順により、デーモンは上記のような状況でも予期せず終了することはなくなります。共有メモリとセマフォを使用しているアプリケーション（MQ、Oracle、SAP など）はこの問題の影響を受ける可能性があるため、この手順が必要となります。

**Kernel のアップデート後、LifeKeeper の "setup" の再実行が必要になるケースがあります**

Red Hat Enterprise Linux 7.x / CentOS 7.x / Oracle Linux 7.x 環境において、カーネルを7.3 相当以降にアップデートした場合に、DataKeeper が正常に動作しないという不具合が生じます。

**回避策:**
アップデート後のシステムで、LifeKeeper をインストールする際に実行した "setup" スクリプトを再実行することで問題は解消します。

**説明**
OSのカーネルモジュールの互換性に問題があり、アップデート前に組み込まれていたカーネルモジュールが、アップデート後のカーネルでは利用出来ません。
DataKeeper は、 nbd.ko という、ネットワーク経由でディスクアクセスを行うカーネルモジュールを使用しています。このモジュールは、LifeKeeper インストール時に行う setup スクリプトにより、適切なものがインストールされます。
カーネルのアップデート後に、setup スクリプトを再度実行することで、新しいカーネル用の nbd.ko が導入されます。

**setup スクリプトの不要な警告**

LifeKeeperパッケージのインストール状況によって、setup スクリプトを実行した際に以下のような警告が表示されます。
Found changes in following files.
These files are overwritten in install process.
If you want to keep changes, please backup these files.

/opt/LifeKeeper/lkadm/subsys/scsi/DEVNAMEが見つかりません。
/opt/LifeKeeper/lkadm/subsys/scsi/DEVNAME/binが見つかりません。
/opt/Lifekeeper/subsys/scsi/resources/DEVNAMEが見つかりません。

これはパッケージ管理上の問題により発生する警告であり、setup スクリプト及び LifeKeeper の動作には影響しません。

将来のバージョンで修正予定です。

# 5.4.5.3.2. LifeKeeper Core – 既知の問題と制限

| 説明 |
| --- |
| **Quorum 構成において、コミュニケーションパス復旧時にリソースに障害が発生するとスプリットブレインが起こる可能性があります**<br><br>クラスターに Quorum が構成されていて、コミュニケーションパスの復旧中にクイックチェックおよびローカルリカバリーに失敗すると、階層のフェールオーバーと Quorum 処理の間で競合が発生し、両ノード間でスプリットブレインが発生する場合があります。 |
| **Quorum が構成されていて、アクティブノードに障害が発生した場合、または [Shutdown Strategy] を [Switchover Resources] に設定して再起動した場合、リソースはセカンダリーノードにスイッチオーバーします。ただし、再起動後に元のアクティブノードが復帰すると、階層を元のアクティブノードに戻そうとするため、階層は障害/サービス停止状態のままになります。**<br><br>大規模な階層では、元のアクティブノードでの LifeKeeper 再起動時に競合状態が発生し、元のプライマリーノード上の階層を復元しようとします。これにより、新しいアクティブノードのリソース階層が停止され、どのノードでも階層が In Service ではなくなります。この問題に対するパッチについては、サポートまでお問い合わせください。 |
| **ネットワークの接続などに問題が生じた場合、ネットワークを自動構成するサービスは停止してください**<br><br>LifeKeeper を用いて IP アドレスを保護している環境では、avahi-daemon などのネットワークを自動構成するデーモンやサービスと IP リソースが 競合する場合があります。コミュニケーションパスの復旧や IP リソースの起動などに問題が生じた場合は、ネットワークを自動構成するサービスを停止してください。 |
| **ifconfig down や ip link down コマンドによるネットワークの切断はしないでください**<br><br>仮想 IP リソースの保護対象となっているネットワークインタフェースがコミュニケーションパスでも使用されている場合、ネットワークインタフェースを ifconfig down や ip link down コマンドを実行して切断すると、再接続後もコミュニケーションパスが復帰しないことがあります。そのような操作は行わないでください。 |
| **マルチユーザーに設定された systemd ターゲットでは LifeKeeper は起動しません** |

LifeKeeper を適切に機能させるためには、「systemctl set-default」または「systemctl isolate」を実行する時に、「lifekeeper-graphical.target」（グラフィカルモードの場合）または「lifekeeper-multi-user.target」（コンソールモードの場合）を使用する必要があります。通常の「graphical.target」および「multi-user.target」systemd ターゲットを使用しないでください。

**ディスクのUUIDに関するDataKeeperの制限事項**

バージョン9.5.0以降、DataKeeperはオペレーティングシステムにUUIDを提示しないディスクをミラーリングできなくなりました。このようなディスクをミラーリングする一番良い方法は、GPT（GUID Partition Table）でパーティションに分割することです。これは「parted」ツールを使用して実施できます。

**注記:** ディスクをパーティションに分割すると、ディスクにすでに保存されているデータはすべて破壊されます。

**回避策:** DataKeeper for Linux トラブルシューティング を参照してください。

**SLES 15では、ログのローテーション後に LifeKeeper のログが LifeKeeper ログファイルに表示されない場合があります**

コマンドラインで logrotate を実行した場合、またはログのサイズが原因でバックグラウンドでログのローテーションが発生した場合、LifeKeeper はロギングを停止します。

**対応策**: systemctl reload rsyslog を実行して、LifeKeeper のロギングを再開してください。

**lkbackup を実行すると、LifeKeeperログにエラーが表示される場合があります**

```
lkbackup[30809]: ERROR:lkbackup:::010064:Possible Configuration error: More than one LifeKeeper version is installed on this host
```

このエラーメッセージは無視しても問題ありません。

**ファイルシステムラベルは、大規模な設定で使用しないことを推奨します**

ファイルシステムラベルを使用すると、大きなクラスタの場合、起動時にパフォーマンスが低下する可能性があります。ラベルを使用するには、システムに接続されるすべてのデバイスをスキャンする必要があり、通常はその結果として問題が生じます。SAN に接続されているシステム、特にデバイスへのアクセスがブロックされている LifeKeeper が導入されているシステムの場合、このスキャニングは非常に遅くなる可能性があります。

Red Hat システムでこのパフォーマンスの問題を防ぐには、/etc/fstab を編集し、ラベルをパス名に置き換えます。

**`lkscsid` は、一部環境においてディスク障害時にシステムを停止します**

`lkscsid` がディスク障害を検出すると、デフォルトでは `sendevent` を LifeKeeper に発行し、障害から復旧しようとします。`sendevent` はまず、ローカルで障害から復旧しようとします。それに失敗すると、ディスクの階層を別のサーバーに切り替えて障害から復旧しようとします。一部のバージョンの Linux (SLES11) では、`lkscsid` は `sendevent` を発行できないため、代わりにすぐにシステムを停止します。これは、`/dev/sda` などの SCSI デバイスノードを使用している階層にのみ影響します。

**DataKeeper の Create Resource が失敗します**

特定の環境 (IDE ディスクエミュレーションを使用する仮想環境、HP CCISS ストレージを装備したサーバーなど) で DataKeeper を使用している場合、ミラーの作成時にエラーが発生することがあります。

```
ERROR 104052: Cannot get the hardware ID of the device “dev/hda3”
```

この原因は、LifeKeeper が該当ディスクを認識せず、デバイスに関連付けられている一意の ID を取得できないからです。

**対応策**: GUIDパーティションを使用することで該当ディスクを認識できるようになります。

**API アクセスに対するホスト名の指定**

LifeKeeper サーバー認証情報の格納に使用するキー名は他の LifeKeeper サーバーのホスト名と *完全に* 一致する必要があります (そのサーバーに対する hostname コマンドで表示されます)。ホスト名が FQDN の場合、認証キーは FQDN である必要があります。ホスト名が短縮名の場合、キーも短縮名にする必要があります。

**対応策**: credstore によって格納されたホスト名がホスト名と完全に一致していることを確認します。

8.2.0 より前のバージョンの LifeKeeper の `lkbackup` を使用する場合は、それ以降のバージョンでリストアするときに `/etc/default/LifeKeeper` を手動で更新する必要があります

LifeKeeper/SPS の現在のバージョンでは、ロギングなどの主要なコアコンポーネントに対して大幅な機能強化が加えられています。これらの機能強化は `/etc/default/LifeKeeper` ファイルの設定に影響します。旧バージョンの LifeKeeper/SPS で作成した lkbackup を新しいバージョンの LifeKeeper/SPS でリストアすると、これらの設定値が正しくなくなり、矛盾が発生します。

**解決方法**: lkbackup をリストアする前に、/etc/default/LifeKeeper を保存してください。lkbackup からリストアした後、以下のチューニングパラメータにマージします。

**8.0.0 より前のバージョンから取得した lkbackup を使用して、8.0.0 以降でリストアする場合:**

```
LKSYSLOGTAG=LifeKeeper

LKSYSLOGSELECTOR=local6
```

9.0.1より前のバージョンから取得した lkbackup を使用して、9.0.1以降でリストアする場合:

```
PATH=/opt/LifeKeeper/bin:/usr/java/jre1.8.0_51/bin:/usr/java/bin:/usr/java/jdk1.8.0_51/bin:/bin:/usr/bin:/usr/sbin:/sbin
```

詳細については、syslog でのロギング を参照してください。

**リソースの作成後に lkbackup をリストアすると、破損したイクイバレンシが残されます**

作成したリソースの設定ファイルは lkbackup 中に保存されます。lkbackup でバックアップした後に初めて作成したリソースは、このバックアップからリストアする際に適切に把握されない可能性があります。

**解決方法**: 新しいリソースを初めて追加する前に lkbackup からリストアしてください。新しいリソースが lkbackupの後で追加された場合、リストアの前に削除するか、リソースの階層のインスタンスを削除し、リストアの後で階層を再拡張してください。 **注記**：特定のリソースを初めて作成する際に lkbackup を実行することを推奨します。

**フェイルオーバ時に誤った順序でリソースが停止されます**

階層が共通リソースインスタンスを別のルート階層と共有している場合、カスケーディングフェイルオーバまたはリソースフェイルオーバの間、リソースは誤った順序で停止されることがあります。

**解決方法**: 共通ルートを作成すると、階層のリソースの停止がトップダウンで実行されます。

1. restore と remove を常に進める gen/app を作成します。
2. 現在のルートをすべて、この新しい gen/app の子にします。

**注記**：restore および remove スクリプトに /bin/true を使用すると、これが可能になります。

**ネストされたファイルシステム階層を削除すると、メッセージ「Object does not exist」が生成されます**

**解決方法**: これは問題ではないので、メッセージを無視してかまいません。

**ネストされたマウントの作成時に `filesyshier` が正しくないタグを返します**

ネストされたファイルシステムのリソースがデータベースに存在する場合、ファイルシステムキットは、親とネストされた子の両方にファイルシステムを作成します。ただし、`filesyshier` は子のタグのみを返します。これにより、アプリケーションは子の依存関係を作成しますが、親の依存関係は作成しません。

**解決方法**: 1 つのマウントポイント内で複数のファイルシステムがネストされている場合、`dep_create` または UI の [Create Dependency] を使用して、親アプリケーションタグの追加の依存関係を手動で作成しなければならないことがあります。

**DataKeeper: ネストされているファイルシステムの作成は DataKeeper で失敗します**

既存のファイルシステムを複製するために DataKeeper ミラーを作成するときに、ファイルシステムがこの構造内でネストされている場合、ファイルシステムをアンマウントしてからファイルシステムリソースを作成する必要があります。

**対応策**: ネストされているファイルシステムを手動でアンマウントしてから、個々のネストされているマウントの再マウント / 作成を行ってください。

**ファイルシステムリソースに保護されたデバイスのマウントポイントの変更は、データ破損を引き起こす可能性があります**

ファイルシステムリソースに保護されたデバイスのマウントポイントは変更しないでください。変更した場合は、ファイルシステムリソースが仕様通りにデバイスを処理することができず、デバイスが 2 つのノードにマウントされて、データ破損を引き起こす可能性があります。

**XFSファイルシステム使用時に、quickCheckに失敗することがあります**

Red Hat Enterprise Linux 7 / Oracle Linux 7 / CentOS 7 にインストールしたLifeKeeper で CHECK_FS_QUOTAS 設定を有効としたとき、 保護する XFS ファイルシステムリソースに uquota、gquota オプションを設定すると、quickCheck に失敗します。

**解決方法**: XFSファイルシステムのマウントオプションに uquota、gquota の代わりにusrquota、grpquota をご使用ください。 もしくは、CHECK_FS_QUOTAS 設定を無効にしてください。

**Btrfs はサポートしません**

LifeKeeper をインストールするファイルシステム（/opt/LifeKeeper）に Btrfs （またはサポートされていないその他の SPS for Linux ファイルシステム）を使用することはできません。またビットマップファイルやその他 LifeKeeper に関連するファイルを置くファイルシステムに Btrfs （またはサポートされていないその他の SPS for Linux ファイルシステム）を使用することはできません。 Btrfs ファイルシステムをリソース保護することもできません。

**対応策:** Btrfs ファイルシステムに /opt/LifeKeeper を配置するための簡単な回避策は、インスタンスに小さなディスクを追加し、ext4 または xfs でそのディスクをフォーマットし、このファイルシステムを /opt/LifeKeeper としてマウントすることです。

1. /opt/LifeKeeper に使用する小さなディスクを作成します
   • ソフトウェアのインストールには最低110MB が必要です
   • 注記：Azure では、最低 1GB のデータディスクを作成できます。
   • 注記：追加の ARK とミラーの数により、必要な合計スペースが増加する場合があります。

2. ディスクがノードに追加されて表示されたら、ディスクをパーティションに分割します（または lvm を使用します）。
   • 例：`gdisk /dev/sdb`
   • 注記：システムにディスクを追加する方法は、このKBA の対象範囲外です（お使いの環境については、自社のシステム管理者にお問い合わせください）

3. サポートされているファイルシステムでパーティションをフォーマットします（詳細については、リリースノート を参照してください） 。
   • 例：`mkfs.ext4 /dev/sdb1`（sdb1 は手順2 で作成）

4. 新しく作成され、フォーマットされたパーティションを /etc/fstab に追加し、システム起動時に自動的にマウントされるように設定します。

5. 新しいパーティションを /opt/LifeKeeper としてマウントします
   • 例：`mount /dev/sdb1 /opt/LifeKeeper`
   • ファイルシステムがマウントされていることを確認します

6. SPS-L をインストールします

7. インストール後、/etc/fstab を編集してエントリを追加し、再起動時にディスクをマウントできるようにします
   • 例: /dev/sdb　　/opt/LifeKeeper　　ext4

**AWS 上で SLES12SP1 以降を用いた場合に発生する問題**

・静的ルーティング設定ができない

/etc/sysconfig/network/routes に静的ルーティングが設定されている場合、DHCPにより自動で割り当

てられたIPアドレスを用いた通信に失敗することがあります。

解決方法: /etc/sysconfig/network/ifroute-ethX の ROUTE を正しく設定し、ルーティング情報を更新してください。

・"Change Hostname via DHCP"が無効になっているにもかかわらず、ホスト名が変更される

LifeKeeper はホスト名が変更された場合正しく動作しません。AWS上で動作するSLES12SP1以降において、"Change Hostname via DHCP"が無効になっているにもかかわらず、ホスト名が変更されることがあります。

解決方法:

- /etc/cloud/cloud.cfg にて update_hostname をコメントアウトしてください
- /etc/cloud/cloud.cfg にて preserve_hostname に true を設定してください
- /etc/sysconfig/network/dhcp にて DHCLIENT_SET_HOSTNAME を no に設定してください

**Quorum/Witness Kit を Witness モードで利用している環境において、Shutdown Strategy を ”Switchover Resources” に設定した場合**

Quorum/Witness Kit を Witness モードで利用している環境において、Shutdown Strategy を ”Switchover Resources” に設定した場合に、正しくスイッチオーバーが行われない場合があります。

**解決方法**: シャットダウン前に手動での切り替えを行ってください。

**/etc/service ファイルの編集**

/etc/service にある以下の行を削除すると LifeKeeper が起動できなくなります。

```
lcm_server 7365/tcp
```

このファイルを編集する場合は、この行を削除しないように注意してください。

**SCSI ID にスペースを含む文字列を返すストレージユニットは LifeKeeper で保護できません**

**bind マウントはサポートしません**

LifeKeeper で保護するファイルシステムには、bind マウント (mount --bind) を使用できません。

AWSやAzure上で動作するバージョン12以降のSLES環境において、クラウド ネットワーク プラグインによって仮想 IP アドレスが削除されるのを防ぐために、ネットワーク インターフェイスの構成ファイルを変更する必要があります。

詳細は こちら を参照してください。

**log ID 4739 –** ミラー再同期の回復イベントによって、スイッチオーバーのコア階層リザベーションロックがリセットされると、スイッチオーバー要求が失敗します。

スイッチオーバーが正常に完了し、ロックを解除しようとしたときに別のプロセスによって所有されていると検知し、正常に復元されたにもかかわらず、切り替えに失敗してルートリソースに失敗とマークされます。

**解決策/回避策**: RESRVRECTIMEOUT の値を調整します。RESRVRECTIMEOUT が 600 よりも高くなった場合、RESRVTIMEOUT の値もそれに合わせて高くする必要があります。

注記: RESRVRECTIMEOUT および RESRVTIMEOUTのタイムアウトを設定する場合、値はリソースの数と階層内の各リソースのタイプによって大きく異なります。この値には、現在のソースノードで各リソースを停止する時間と、新しいソースノードで各リソースを復元する時間が含まれている必要があります。リソースのリストアと削除の時間はクラスタや時間によって異なるため、これらの調整値の推定値を得るには、次の手順に従います。

調整の手順:

- 複数のスイッチオーバーを実行します（少なくとも3回をお勧めします）。
- 各スイッチオーバーが完了するまでにかかる時間を記録し、最長の合計スイッチオーバー時間を記録します。
    - DataKeeperリソースの含まれる階層で計測された、「最長合計スイッチオーバー」よりも長い時間（秒単位）を RESRVRECTIMEOUTに設定します。

例: **RESRVRECTIMEOUTに設定された値を x、RESRVTIMEOUTを y とした場合、** RESRVRECTIMEOUTの「x」値 > 600であれば、RESRVTIMEOUTを x に設定する。 **[もし x >600なら、x = y]** 。

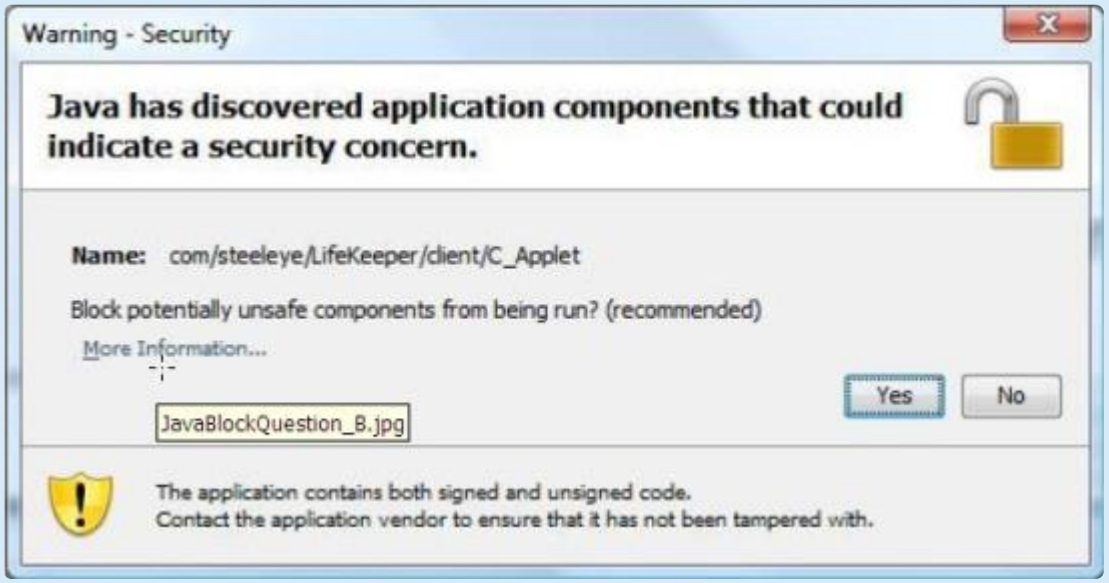
# 5.4.5.3.3. インターネット / IP ライセンス - 既知の問題と制限

| 説明 |
| --- |
| ***/etc/hosts*** **の設定の依存関係**<br><br>**/etc/hosts 設定:**<br>インターネットベースのライセンス (IPv4 アドレス) を使用している場合、 */etc/hosts* の設定はライセンスの検証に悪影響を与える可能性があります。LifeKeeper の起動に失敗した場合は、以下のようなメッセージが出力されます。<br><br>`Error in obtaining LifeKeeper license key:`<br>`Invalid host.`<br>`The hostid of this system does not match the hostid specified in the license file.`<br><br>リストされているインターネットホスト ID が正しい場合、 */etc/hosts* の設定が原因の可能性があります。 */etc/hosts* エントリを正しく一致させるには、IPv6 エントリの前に IPv4 エントリを記載する必要があります。 */etc/hosts* 設定が原因かどうかを確認するには、次のコマンドを実行します。<br><br>`/opt/LifeKeeper/bin/lmutil lmhostid -internet -n`<br><br>記載された IPv4 アドレスが、インストールされたライセンスファイルの IPv4 アドレスと一致しない場合、正しいアドレスを返すために、 */etc/hosts* を変更し、IPv4 エントリを IPv6 エントリの前に配置する必要があります。 |

# 5.4.5.3.4. GUI – 既知の問題と制限

| 説明 |
| --- |
| **LKCLI のエクスポート/インポート機能を用いて、DataKeeper リソースのミラーリング方式がノード毎に異なるようなリソースを作成することはできません**<br><br>GUI を用いて DataKeeper リソースを作成した場合、プライマリからセカンダリ間を非同期転送、セカンダリからプライマリ間を同期転送する設定が出来ますが、その構成をLKCLIでエクスポートした場合に別のクラスタ構成にインポートすることはできません。<br><br>**注意:** 上記構成はサポートされません。 |
| **GUI を終了した後で Web ブラウザを介して再接続すると、GUI ログインプロンプトが再表示されない場合があります**<br><br>GUI アプレットを終了するか切断してから同じ Web ブラウザセッションから再接続しようとすると、ログインプロンプトが表示されない場合があります。<br><br>**対応策:** Web ブラウザを閉じ、開き直してからサーバに接続してください。Firefox を使用している際は、すべての Firefox を閉じ、再び開きます。 |
| **ネットワークが切断され、再接続された後で、GUI は IP リソースの状態をすぐに更新しません**<br><br>クラスタ内のサーバ間のプライマリネットワークが切断され、再接続されると、RMI/TCP レイヤーの問題のため、リモート GUI クライアントの IP リソースの状態が更新されるまで 1 分 25 秒かかる場合があります。 |
| **Java 署名/未署名混合コードの警告 – LifeKeeper Java GUI クライアントアプレットをリモートシステムからロードすると、以下のセキュリティ警告が表示されることがあります** |

[Run] をクリックすると、以下のダイアログが表示されます。




ブロックするかどうかを確認するメッセージが表示されます。[No] をクリックすると、LifeKeeper GUI の動作が可能になります。

**推奨される処置:** セキュリティ警告の数を減らすには、2 つのオプションがあります。

1. **[Always trust content from this publisher]** ボックスをチェックし、 **[Run]** をクリックします。次に LifeKeeper GUI Java クライアントを読み込むときには、警告メッセージが表示されません。

または

2. ブロックに関する 2 番目のダイアログが表示されないようにするには、以下のエントリを Java の「 **deployment.properties** 」ファイルに追加します。Java クライアントをロードしたときにセキュリティ警告は表示されますが、アプレットはブロックされず、ブロックするかどうかを **[Yes]** または **[No]** で確認する上記のダイアログは表示されなくなります。この設定はすべての Java アプレットに適用されることに注意してください。

```
deployment.security.mixcode=HIDE_RUN
```

上記のメッセージが両方とも表示されないようにするには、1 と 2 を実行してください。

**ポート 778･779 が使用中の場合、steeleye-lighttpd プロセスの開始に失敗します**

steeleye-lighttpd の起動時にプロセスがポート 778 もしくは 779 を使用している場合、steeleye-lighttpd の起動に失敗し、GUIへの接続障害が発生する。

**解決方法**: クラスタ内のすべてのノードで以下の設定を行い、LifeKeeper をすべてのノードで再起動します。

以下の行を */etc/default/LifeKeeper* に追加します。

```
API_SSL_PORT=port_number
LKAPI_WEB_PORT=port_number
```

port_number は使用する新しいポートです。

# 5.4.5.3.5. データレプリケーション – 既知の問題と制限

| 説明 |
| --- |
| **非同期モードで作成し同期モードで拡張した DataKeeper リソースの構成はサポートされません**<br><br>非同期モードで作成し同期モードで拡張した DataKeeper リソースでは、ミラーを読み書きしているプロセスがカーネル内でハングアップすることがあります。<br><br>DataKeeper リソースの同期モードの判定方法は、以下のコマンドを各ノードで実行してください。0 は同期モードで、非 0 は非同期モードです。全ノードで、全て同期モードもしくは全て非同期モードであれば問題ありません。いずれかのノードで、同期モードと非同期モードが混在している場合は、本問題が発生し得る構成です。<br>`perl -nle 'my @x = split(/\x01/, $_); print "$x[0]:$x[3]";' /opt/LifeKeeper/subsys/scsi/resources/netraid/mirrorinfo_<md num>`<br><br>**回避策**：直接回避する方法はありません。DataKeeper リソース階層を再作成して、作成･拡張時にどちらも同期モードを選択してください。 |
| **カーネル 4.12 以降を実行している場合、総セクタ数が奇数になるパーティションはサポートされません**<br><br>総セクタ数が奇数になるパーティションの使用は、カーネル 4.12 以降を実行している環境の DataKeeper ミラーではサポートされていません。これは、ディスクの終端を超えて書き込みをしようとした場合に再同期が失敗するという問題に起因します。 |
| **LVM 構成における DataKeeper for Linux 非同期モードに関する注意事項**<br><br>複数の非同期ミラー上に LVM を構成すると、データの整合性が保てず、さらには kernel panic が発生する場合があります。<br><br>DataKeeper 上に LVM を構成する場合には、DataKeeper ミラー 1 つだけ、または複数の同期ミラー、いずれかの構成でなければなりません。 |
| **両方のサーバに重要な I/O トラフィックを持つシンメトリックなアクティブ SDR 設定で、netraid デバイス (ミラー) にマウントされたファイルシステムが応答を停止し、結果的にシステム全体がハングします**<br><br>Linux バッファキャッシュの単一スレッドの特性により、バッファキャッシュフラッシングデーモンは、リモートでコミットする必要があるバッファをフラッシュアウトしようとしてハングする可能性があります。フラッシングデーモンがハングすると、クリアされていないバッファの数がシステムで許容され |

ている上限 (*/proc/sys/kernel/vm/bdflush* で設定) を超えると、クリアされていないバッファを持つ Linux システムのすべてのアクティビティが停止します。

これは、リモートシステムがリモートバッファを消去できなくなるような事態でないかぎり、通常は深刻な問題ではありません。LifeKeeper はネットワークの障害を検出し、そのときにレプリケーションを停止するので、ハング状態は消去されます。ただし、リモートシステムがローカルシステムにもレプリケートされた場合 (つまり、相互がシンメトリカルにレプリケートされた場合)、両方のシステムでこのフラッシングデーモンのハング状態が発生すると、永久にデッドロックする可能性があります。

デッドロックを解除するには、両方のシステムの nbd-client デーモンを手動で停止します (これにより、ミラーが切断されます)。ただし、このデッドロックを完全に防止する場合は、シンメトリックアクティブレプリケーションを推奨しません。

**大きなミラーサイズを備えた md_raid1 プロセスではトップで高い CPU 使用率がレポートされます**

`mdX_raid1` プロセス *(X はミラー番号)* では、非常に大きなミラー (500GB 以上) を操作している際に、一部の OS ディストリビューションで高い CPU 使用率がトップで報告されることがあります。

**解決方法**: CPU の使用率を減らすには、LifeKeeper 設定 `LKDR_CHUNK_SIZE` でチャンクサイズを 1024 に変更し、この新しい設定を使用するためにミラーを削除して再作成します。

**DataKeeper リソースで `lkbackup` を使用する場合は、全同期が必要です**

`lkbackup` では *`instance`* と *`mirror_info`* ファイルを保存しますが、リソースが存在しないときにソースおよびターゲットの状態は保証されないため、`lkbackup` から restore した後で、DataKeeper ミラーの全同期を実行することがベストプラクティスです。

**SLES12 SP1 以降でデータ圧縮機能を使用することはサポートしていません**

パフォーマンスの問題があるため、SLES12 SP1 以降でデータ圧縮機能を使用することはサポートしていません。

**特定バージョンのカーネルでは DataKeeper の非同期モードはサポートしていません。**

LifeKeepe for Linux において DataKeeper リソースの非同期モード設定を使用すると、特定バージョンのカーネルでカーネルパニックが発生することが確認されています。本事象はカーネルに依存した問題であるため、LifeKeeper による根本的な解決方法はありません。DataKeeper の非同期モード設定をご利用いただくには、カーネルのアップデート、またはダウングレードが必要となります。

DataKeeper の非同期モードを利用できないカーネルバージョンは次の通りです。

Red Hat Enterprise Linux / Oracle Linux / CentOS において、以下のバージョンのカーネルを使用している場合
3.10.0-693.24.1.el7.x86_64 以降の 3.10.0-693. 系カーネル
3.10.0-862.el7.x86_64 から 3.10.0-862.26.x.el7.x86_64 まで
3.10.0-957.el7.x86_64 から 3.10.0-957.3.x.el7.x86_64 まで

上記に該当する環境をご利用で、DataKeeper リソースを非同期モードでご利用の場合には以下のカーネル バージョンへのアップデート（またはダウングレード）を実施してください。

Red Hat Enterprise Linux / Oracle Linux / CentOS において、以下のバージョンのカーネルを使用している場合
3.10.0-693.24.1.el7.x86_64 未満 の3.10.0-693. 系カーネル
3.10.0-862.29.1.el7.x86_64 以降
3.10.0-957.4.1.el7.x86_64 以降

カーネルのアップデート（またはダウングレード）が不可能である場合には、DataKeeper では非同期モードを利用しないようにしてください。

一部のカーネルバージョンでは **Secure Boot** 機能をサポートしていません

RHEL7 以降、CentOS7 以降、Oracle Linux7(非 UEK カーネルのみ) 以降で Secure Boot を有効にしていると、nbd モジュールのロードに失敗します。また、SUSE Linux Enterprise Server、Oracle Linux UEK kernel においても、一部のカーネルバージョンでは、Secure Boot を有効にしていると、md/raid1 カーネルモジュールのロードに失敗します。

**解決方法:** 以下のいずれかの対応を行ってください。(1) を推奨します。また、いずれの方法も OS 再起動が必要です。
(1) Secure Boot を無効にする。
UEFI の設定で、Secure Boot を無効にしてください。
(2) 署名の検証を無効にする。
“mokutil --disable-validation” コマンドを実行して、署名の検証を無効にしてください。詳細は mokutil のマニュアルを参照ください。

# 5.4.5.3.6. IPv6 - 既知の問題と制限

| 説明 |
| --- |
| SIOS は、 ifconfig コマンドから ip コマンドの使用に移行しました。この変更のため、外部スクリプトを使用するお客様も、同様の変更を行うことを推奨します。 ifconfig コマンドを発行し、結果を解析して特定のインターフェースを探す代わりに、スクリプトは「ip -o addr show」を使用し、結果を解析して、「inet」および「secondary」という語を含む行を検索します。<br><br>`# ip -o addr show`<br>`1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN`<br>`\ link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00`<br>`1: lo inet 127.0.0.1/8 scope host lo`<br>`1: lo inet6 ::1/128 scope host`<br>`\ valid_lft forever preferred_lft forever`<br>`2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state U`<br>`P qlen 1000`<br>`\ link/ether d2:05:de:4f:a2:e6 brd ff:ff:ff:ff:ff:ff`<br>`2: eth0 inet 172.17.100.77/22 brd 172.17.103.255 scope global eth0`<br>`2: eth0 inet 172.17.100.79/22 scope global secondary eth0`<br>`2: eth0 inet 172.17.100.80/22 scope global secondary eth0`<br>`2: eth0 inet6 2001:5c0:110e:3364::1:2/64 scope global`<br>`\ valid_lft forever preferred_lft forever`<br>`2: eth0 inet6 2001:5c0:110e:3300:d005:deff:fe4f:a2e6/64 scope global dynam`<br>`ic`<br>`\ valid_lft 86393sec preferred_lft 14393sec`<br>`2: eth0 inet6 fe80::d005:deff:fe4f:a2e6/64 scope link`<br>`\ valid_lft forever preferred_lft forever`<br><br>ip コマンドの上記の結果では、以下の行に eth0 インターフェースの仮想 IP アドレスが含まれます。<br><br>`2: eth0 inet 172.17.100.79/22 scope global secondary eth0`<br>`2: eth0 inet 172.17.100.80/22 scope global secondary eth0` |
| ***/etc/sysconfig/network-scripts/ifcfg-<nicName>*** **の「IPV6_AUTOCONF = No」が再起動または起動の際に考慮されません**<br><br>起動時に、自動設定されたステートレスな IPv6 アドレスがネットワークインターフェースに割り当てられます。IPV6_AUTOCONF=No が設定されているインターフェースのステートレス IPv6 アドレスでコミュニケーションパスが作成された場合、任意のシステムリソースが ifdown <nicName>;ifup <nicName> などのインターフェースを管理する際にアドレスが削除されます。<br><br>IPV6_AUTOCONF が No に設定されていたため、プライマリサーバを再起動した後で、自動設定された |

IPv6 アドレスを使用しているコミュニケーションパスは復旧せず、無効のままでした。

**解決方法**: スタティックな IPv6 アドレスのみを使用してください。自動設定された IPv6 アドレスを使用すると、再起動後に通信が失われたり、NIC が変更されたりする可能性があります。

自動設定された IPv6 アドレスをコミュニケーションパスの作成に使用できますが、システム管理者は以下の条件を認識する責任があります。

- 自動設定されたステートレス IPv6 アドレスがネットワークインターフェース (NIC) の MAC アドレスに準拠していること。コミュニケーションパスが作成され、関連 NIC が後で置き換えられた場合、自動設定された IPv6 アドレスは異なるものになり、LifeKeeper はコミュニケーションパスが無効になっていることを正しく表示します。コミュニケーションパスを再作成する必要があります。
- Red Hat Enterprise Linux では、ホスト操作のあらゆる側面で一貫した IPv6 自動設定を確保するための動作を実行するには、sysctl.conf、net.ipv6.* 命令 (‘if/ip’ ユーティリティで参照される ifcfg-<nic> の明示的な IPV6_AUTOCONF 設定、およびシステムが起動して init レベルを切り替える際に NIC 制御に影響する /etc/sysctl.conf) に加え、個々のインターフェース設定ファイルを正確に設定するために、詳細かつ具体的なドメインの知識が必要になります。

**IP:IPv6 のソースアドレス変更設定ではソースアドレスが設定されません**

IPv6 IP リソースのソースアドレスを設定しようとすると、何も変更されていない場合でも成功となります。

**対応策**: 現在のところ、対応策はありません。今後のリリースで対応する予定です。

**IP:無効な IPv6 アドレス設定が IP リソース作成で許可されます**

オクテットに 4 文字を超える文字が含まれている場合、2001:5c0:110e:3368:000000:000000001:61:14 という形式の IPv6 アドレスが許容されます。

**対応策**: 正しい形式の IPv6 アドレスを入力してください。

**bonding NIC に割り当てられているものの、「暫定的な」状態のアドレスでは、IPv6 リソースが ISP としてレポートされます**

LifeKeeper の IPv6 保護リソースは、IPv6 リソースが bonding インターフェース上にある SLES システムでは「In Service Protected」(ISP: in service の保護) と不正に識別されます。これは、‘active-backup’ (1) および 2.6.21 以前の Linux カーネルとは別のモードです。IPv6 の bonding リンクは、解決できないアドレスを持つ「暫定的な」状態のままになります。

**対応策:** bondingインターフェースモードを ‘active-backup’ (1) に設定します。または、‘active-backup’ (1) 以外のモードの場合、リンク状態を「tentative (暫定的)」から「valid (有効)」に設定する更新したカーネルで操作します。

# 5.4.5.3.7. Apache - 既知の問題と制限

| 説明 |
| --- |
| **Apache** キットは **IPv6** をサポートしていません。 ***httpd.conf*** で **IPv6** を識別しません<br><br>*httpd.conf* ファイルの「Listen」命令エントリに IPv6 アドレスを割り当てると、問題が発生します。<br><br>解決方法: Apache Recovery Kit で IPv6 がサポートされるまで、リソース作成後に IPv6 アドレスを *httpd.conf* ファイルに指定できません。 |

# 5.4.5.3.8. Oracle Recovery Kit - 既知の問題と制限

| 説明 |
| --- |
| **Oracle Recovery Kit** には **Connection Manager** および **Oracle Names** 機能のサポートが含まれていません<br><br>LifeKeeper Oracle Recovery Kit には、Oracle Connection Manager と Oracle Names という Oracle Net 機能のサポートが含まれていません。Oracle Connection Manager は、同じサービスにアクセスする必要がある多数の接続を管理するルーティングプロセスです。Oracle Names はサービスアドレスの一括格納を管理する Oracle 固有の命名サービスです。<br><br>LifeKeeper Oracle Recovery Kit は、送信されてきたクライアント接続要求をリスニングし、サーバへのトラフィックを管理する Oracle Net Listener プロセスを保護します。Oracle Listener に固有の LifeKeeper 設定情報については、『 *LifeKeeper for Linux Oracle Recovery Kit 管理ガイド* 』を参照してください。 |
| **Oracle Recovery Kit** は **ASM** またはグリッドコンポーネント機能をサポートしていません<br><br>Oracle で提供されている Oracle Automatic Storage Manager (ASM) 機能は現在、LifeKeeper ではサポートされていません。 また、グリッドコンポーネントは LifeKeeper Oracle Recovery Kit によって保護されません。Raw デバイス、ファイルシステム、論理ボリュームに対するサポートは、現在の LifeKeeper for Linux Oracle Recovery Kit に含まれています。グリッドコンポーネントに対するサポートは、gen/app リカバリキットを使用して LifeKeeper の保護機能に追加できます。 |
| **Oracle Recovery Kit** は **NFS** バージョン **4** をサポートしていません<br><br>Oracle Recovery Kit は共有データベースストレージ用に NFS バージョン 3 をサポートしています。NFSv4 ファイルロッキングメカニズムのため、NFS バージョン 4 は、現時点ではサポートされていません。 |
| フェイルオーバ後にプライマリサーバで **Oracle Listener** が **in service** のままになります<br><br>ネットワーク障害により、アプリケーションがバックアップサーバにフェイルオーバした後、プライマリサーバで Listener プロセスがアクティブなままになります。正しいデータベースへの接続は影響を受けませんが、その Listener プロセスを終了したいと考えるかもしれません。 |

# 5.4.5.3.9. MySQL - 既知の問題と制限

| 説明 |
| --- |
| **include 文をサポートしません**<br><br>MySQL Recovery Kit は、my.cnf 設定ファイルにおいて include 文をサポートしておりません。 |
| クラッシュリカバリについて<br><br>MySQL リカバリキットは、MySQL がクラッシュリカバリ処理を行っている間、それを異常と判断して回復処理を起動します。 そのため、両者の処理が競合してしまい、MySQL のクラッシュリカバリが正常に行われない場合があります。<br>この現象は、リカバリ中にクライアントからの接続を受け付けなくなるために、LifeKeeper の監視処理で行われる接続確認が失敗するために発生します。 |

# 5.4.5.3.10. NAS Recovery Kit – 既知の問題と制限

| 説明 |
| --- |
| **RHEL8 など UDP が無効になっているサーバーがエクスポートした NFS を保護する場合、/etc/default/LifeKeeper に “NFS_RPC_PROTOCOL=tcp” を設定する必要があります**<br><br>NFS サーバーとの接続確認に UDP を使用しています。UDP が無効になっているサーバーではリソース作成に失敗します。TCP で接続確認をするために変更が必要です。 |

# 5.4.5.3.11. NFS Server Recovery Kit – 既知の問題と制限

| 説明 |
| --- |
| **/etc/mtab** が **/proc/self/mounts** へのシンボリックリンクであるシステムでは、**NFS v4** とバインドマウントは一緒に使用できない<br><br>RHEL 7.0 以降、CentOS 7.0 以降、OL 7.0 以降、および SLES 12 SP1 以降のシステムでは、以前 /etc/mtab にあったバインド情報が /proc/self/mounts にないため、NFS v4 共有を使用してバインドマウントを使用することはできません。 |
| 最上位の **NFS** リソース階層は **hanfs** リソースのスイッチバックタイプを使用します<br><br>障害からサービス状態に復旧する際に NFS リソース階層がプライマリサーバにスイッチバックするかどうかを制御するスイッチバックタイプは、hanfs リソースで定義されます。 |
| 一部のクライアントが **nfs** ファイルロックを再取得できません<br>~~NFS クライアントとして動作しているとき、一部の Linux カーネルは、「NFS ロックが解除されているので、再取得する必要がある」という、NFS サーバからの通知に正常に応答しません。そのため、これらのシステムが、LifeKeeper に保護されている NFS ファイル共有のクライアントである場合、これらのクライアントで保持されている NFS ロックは、フェイルオーバまたはスイッチオーバの際に失われます。~~<br>~~NFS マウントオプションの推奨値で NFS ロックを使用する場合、以下のオプションを設定することを推奨します。~~<br>~~rw,nolock,bg,hard,nointr,tcp,nfsvers=3,timeo=600,rsize=32768,wsize=32768,actimeo=0~~<br><br>（2021年10月訂正）<br>リソースのスイッチオーバーまたはフェイルオーバーの際、ファイルロックのスイッチオーバーが機能しません。<br>クライアントのアプリケーションで、ファイルロックを使用しないでください。 |
| **IPv6** では **NFS v4** を設定できません<br><br>IPv6 仮想 IP は NFSv4 階層にまとめられます。<br><br>解決方法: NFS v4 リソースの作成時に IPv6 仮想 IP リソースを使用しないでください。 |
| **NFS v4:**拡張解除後に階層を再拡張できません |

エクスポートポイントがすでにターゲットサーバでエクスポート済みなので、拡張に失敗します。NFS v4 階層がサーバ A で作成され、サーバ B に拡張され、サーバ B で in service になった後にサーバ A から拡張解除された場合、NFS v4 階層のサーバ A への再拡張は失敗します。

**解決方法**: サーバ A で `exportfs -ra` というコマンドを実行し、残された追加エクスポート情報をクリーンアップします。

**NFS v3 でのファイルロックのスイッチオーバーが、特定の OS で失敗します**
~~リソースのスイッチオーバーまたはフェイルオーバーの際、下記に記載の OS 上では NFS v3 のファイルロックのフェイルオーバーが機能しません。NFS v3 でのロックフェイルオーバーはこれらのバージョンではサポートされていません。~~

- ~~RHEL8, CentOS8, OL8~~
- ~~RHEL7, CentOS7, OL7~~
- ~~SLES15~~
- ~~SLES12~~

~~**解決方法**: NFS v4 で利用可能なロックフェイルオーバー機能を使用します。~~

（2021年10月訂正）
リソースのスイッチオーバーまたはフェイルオーバーの際、ファイルロックのスイッチオーバーが機能しません。これは、OS によらず、NFS v4 でも同じです。
クライアントのアプリケーションで、ファイルロックを使用しないでください。

**Oracle Recovery Kit は NFS v4 をサポートしていません**

Oracle Recovery Kit は共有データベースストレージ用に NFS v3 をサポートしています。NFS v4 ファイルロッキングメカニズムのため、NFS v4 は、現時点ではサポートされていません。

**NFS サブシステムの停止と起動が、SIOS Protection Suite で保護される NFS エクスポートに悪影響を及ぼします**

SIOS Protection Suite for Linux が NFS エクスポートを保護しているときに、NFS サブシステム (`nfs-server.service`) が停止された場合、NFS の停止動作ですべてのディレクトリのエクスポート解除が実行されるため、SIOS Protection Suite for Linux が保護しているすべてのエクスポート済みディレクトリが影響を受けます。SIOS Protection Suite for Linux の NFS quickCheck スクリプトが停止した NFS のプロセスを検出し、ローカルリカバリを実行してプロセスを再起動してから、ディレクトリを再エクスポートします。ただし、SIOS Protection Suite の NFS ARK がすべてを復旧するために、保護されたエクスポートのすべてについて quickCheck が実行されます。例えば、SIOS Protection Suite for Linux の NFS ARK により 5 個のエクスポー

トが保護されている場合、そのキットが保護するすべてのエクスポート済みディレクトリを復旧するために、quickCheck が 5 回実行されます。デフォルトの quickCheck 時間は 2 分なので、すべてのエクスポート済みディレクトリを復旧するには、10 分かかります。

**対応策**: SIOS Protection Suite の NFS ARK がシステム上のエクスポート済みディレクトリをアクティブに保護しているときには、NFS サブシステムを停止しないでください。NFS サブシステムを停止する必要がある場合は、すべての NFS リソースをスタンバイノードに切り替えてから NFS サブシステムを停止してください。`exportfs` コマンドの使用も検討する必要があります。このコマンドラインユーティリティは 1 つのディレクトリのエクスポート / エクスポート解除の機能を装備しているので、NFS サブシステム全体を停止する必要がなくなります。

**RHEL8 など UDP が無効になっているサーバーでは、/etc/default/LifeKeeper に “NFS_RPC_PROTOCOL=tcp” を設定する必要があります**

nfsd 起動確認に UDP を使用しています。UDP が無効になっているサーバーではリソース作成に失敗します。TCP で起動確認をするために変更が必要です。

# 5.4.5.3.12. SAP Recovery Kit – 既知の問題と制限

## SAP Recovery Kit

| 説明 |
| --- |
| **LifeKeeper GUI を使用する際に複数の SAP 階層に対して SAP メンテナンスモードを有効にできません。**<br><br>SPS-L 9.4.0 以降で有効な 再設計された ERS リソースタイプ を使用した構成において、ERS リソースは、対応するセントラルサービスのリソースとは別の階層で動作します。そのため、予定されたメンテナンス実行時にいずれの階層もメンテナンスモードにする必要があります。LifeKeeper GUI の問題により、一つの SAP 階層に対してSAP メンテナンスモードを有効にすると、別の SAP 階層に対してメンテナンスモードを有効にすることができません。このシナリオに対する回避策として、 SAP メンテナンスモード のページの「コマンドラインから SAP メンテナンスモードを有効または無効にする。」に記載されている lk_maintenance_mode スクリプトを使用してコマンド経由で SAP メンテナンスモードを有効および無効にしてください。 |
| **コミュニケーションパス復旧時に ERS v2 リソースに対してスプリットブレインが発生する可能性があります**<br><br>SAP クラスターで Quorum が構成されていて、ASCS と ERS v2 リソースの両方が同じノードに存在する場合、コミュニケーションパスが復旧する際に ERS v2 リソースの切り替えが発生するとスプリットブレインになる場合があります。本件に関するパッチについては製品サポートへお問い合わせください。 |
| **RHEL 8 の csh バグのため、SAP リソースが In Service になりません**<br><br>RHEL 8 で利用可能な tcsh パッケージのバグにより、特定の状況では、SAP管理ユーザーの .cshrc および .login ファイルを正しく読み込めません。このため、SAP Recovery Kit が依存する重要な環境変数が適切にエクスポートされず、SAP リソースがRHEL 8 で In Service にならない可能性があります。詳細とに解決方法ついては、RedHat Bug 1714267 を参照してください。 |
| **SAP デュアルスタック環境の制限**<br><br>SPS-L 9.4.0（対応するセントラルサービスインスタンスとは別の階層で動作）で実装された新しい設計の ERSリソースタイプ は、同じSIDでセントラルサービスとエンキューレプリケーションサーバーインスタンスのペアが2組インストールされている（例：ASCS00/ERS10 および SCS01/ERS11SAPなど）SAPデュアルスタック環境（ABAP + Java）をサポートしません。<br><br>同じSIDでインストールされたSAPデュアルスタック（ABAP + Java）環境を利用しているユーザーは、9.4.0より前のERSリソース設計（対応するASCS/SCSリソースに依存するSAP階層の最上位に位置）を引き続き使用する必要があります。 |
| **SAP 階層の削除または拡張解除に失敗します**<br><br>同じ IP リソースを階層内の複数の場所に格納している SAP 階層を削除または拡張解除すると、リソース |

が削除されず、コアダンプが発生することもあります。

この問題を修正するには、拡張解除または削除操作に失敗した後で、残ったリソースを LifeKeeper GUI から手動で削除します。サーバからコアファイルを削除するという方法もあります。

**[Handle Warnings] により -e 行 1 で構文エラーが発生します**

**[Handle Warnings]** のデフォルトの動作 **[No]** を **[Yes]** に変更すると、エラーを受信します。

**解決方法**: このオプションをデフォルト設定の **[No]** のままにしてください。 **注記**: 黄は過渡的な状態であり、ほとんどの場合は障害を表すものではないため、この設定はデフォルト選択の **[No]** のままにすることを強くお勧めします。

**同じ設定を選択すると、Update Wizard のボタンが非表示になります**

現在の設定を変更せずに **[Handle Warnings]** を更新しようとすると、戻る必要があることを示す次の画面で **[Done]** ボタンが表示されません。

**res_state に変更を加えると、監視が無効になります**

**[Protection Level]** を **[BASIC]** に設定し、SAP を手動でダウンさせた場合 (メンテナンスなどの目的で)、「FAILED」とマークされ、監視が停止します。

**解決方法**: 監視を再開する場合、手動で起動するのではなく LifeKeeper がリソースを起動する必要があります。

**ERS が Core/CI の親ではない場合、サービス中の ERS がリモートホストで失敗します**

**注記**: これは、9.4.0 より前の ERS リソース設計（ERS リソースが、対応する ASCS/SCS リソースに依存する SAP階層の最上位に存在）にのみ適用されます。詳細については、 LifeKeeperのERSリソースタイプ を参照してください。

追加 SAP リソースの依存関係なしで ERS リソースを作成すると、スイッチオーバ時に初期のサービス中状態が失敗します。

**解決方法**: CI/Core インスタンス (SCS または ASCS) の親として ERS を作成してから、in-service を再試行します。

**SAPインスタンスプロセスの状態が一致しなくなる**

SAPリソースの移行中に並行して管理コマンドを発行すると、SAPインスタンスプロセスの状態が一致しなくなる可能性があります。これを解決するには、手動による介入が必要になる場合があります。

# 5.4.5.3.13. LVM Recovery Kit – 既知の問題と制限

| 説明 |
| --- |
| **LVM 構成における DataKeeper for Linux 非同期モードに関する注意事項**<br><br>複数の非同期ミラー上に LVM を構成すると、データの整合性が保てず、さらには kernel panic が発生する場合があります。<br>DataKeeper 上に LVM を構成する場合には、DataKeeper ミラー 1 つだけ、または複数の同期ミラー、いずれかの構成でなければなりません。 |
| **lkID の使用はディスク全体で LVM を上書きした状態と互換性がありません**<br><br>lkID を使用して、LVM 物理ボリュームとして設定されているディスクで一意のディスク ID を生成すると、lkID および LVM 情報が格納されているディスク上の場所で競合が発生します。これにより、lkID および pvcreate が使用された順番に従って、lkID または LVM 情報のどちらかが上書きされます。<br><br>**対応策**: lkID を LVM と組み合わせて使用する必要がある場合は、ディスクをパーティション化し、ディスク全体ではなくディスクパーティションを LVM 物理ボリュームとして使用します。 |

# 5.4.5.3.14. Multipath Recovery Kits (DMMP / HDLM / PPATH /NECSPS) - 既知の問題と制限

| 説明 |
| --- |
| Multipath Recovery Kit（DMMP / HDLM / PPATH /NECSPS）では、マルチパスリソースの状態がOSFの場合、LifeKeeperをスタンバイ（OSU）ノードで停止した時に、アクティブ（ISP）ノードでシステム停止が発生することがあります。<br><br>回避策：<br><br>a）LifeKeeperを停止する前に階層をスタンバイノードに切り替える<br><br>または<br><br>b）LifeKeeperを停止する前にスタンバイノードで `ins_setstate`を実行し、マルチパスリソースの状態をOSUに設定する |

# 5.4.5.3.15. DMMP Recovery Kit – 既知の問題と制限

| 説明 |
| --- |
| **DMMP:スタンバイサーバーで発行された write がハングすることがあります**<br><br>別のサーバーでリザーブされている DMMP デバイスに write が発行されると、IO が永久に (またはデバイスがもう一方のサーバーでリザーブされなくなるまで) ハングすることがあります。 もう一方のサーバーでデバイスが解放され、write が発行されると、データが破損することがあります。<br>この問題の原因は、DMMP での IO 再試行に従ってパス確認が実行される方法にあります。<br>「no_path_retry」が 0 (失敗) に設定されている場合、このハングは発生しません。 別のサーバーでパスがリザーブされているときにデバイスの path_checker が失敗しても (MSA1000)、この問題は発生しません。<br><br>**対応策**:「no_path_retry」を 0 (失敗) に設定します。 しかしこれにより、一時的なパスの障害が原因で、IO の障害が発生する可能性もあります。 |
| **DMMP:複数のイニシエータが ATP_C をサポートする SAS アレイで正しく登録されていません**<br><br>LifeKeeper は、複数の SAS イニシエータが SAS アレイに接続される設定をネイティブにサポートしていません。 こうした設定では、各イニシエータが正常に登録されないので、1 つのイニシエータのみが IO を発行できるようになります。 マルチパスドライバ (DMMP など) が未登録のイニシエータに IO を発行すると、エラーが発生します。<br><br>**解決方法**: SAS ストレージ情報に基づいてパス ID が設定されるように、`/etc/default/LifeKeeper` の設定値を以下のように設定してください。<br><br>`MULTIPATH_SAS=TRUE` |
| **DMMP リカバリーキットのパラメータ設定が必要な場合、複数の異なるストレージを同時に使用することはできません** |
| **末尾が "p<数字>" になっている LV と併用できません**<br><br>LVM を併用する場合、末尾が "p<数字>"になっている LV が存在すると DMMP RK が正常に動作しません。 このような名前を持つ LV は作成しないでください。 |

# 5.4.5.3.16. DB2 Recovery Kit – 既知の問題と制限

| 説明 |
| --- |
| **DB2 Recovery Kit** が不要なエラーをレポートします<br><br>DB2 が共有ディスクにインストールされている場合、DB2 リソースを拡張する際に以下のメッセージが表示されることがあります。<br><br>`LifeKeeper was unable to add instance "%s" and/or its variables to the DB2 registry.`<br><br>このメッセージは、DB2 リソースの拡張動作には悪影響を及ぼしません。 |

# 5.4.5.3.17. Sybase ASE Recovery Kit – 既知の問題と制限

| 説明 |
| --- |
| **ユーザー名 / パスワードの問題**:<br><br>• **デフォルトのユーザー名がパスワードで保護されている場合、すべての検証が完了するまで create UI はこれを検出しません**<br><br>Sybase リソースを作成する際に、ユーザー名の入力が要求されます。フロントのヘルプには、ユーザーが指定されない場合はデフォルトの「sa」が使用されるというメッセージが表示されます。ただし、この時点ではデフォルトのアカウントに対するパスワード検証は実行されません。SIOS Protection Suite が Sybase リソースを作成しようとする際に、パスワードの検証 / 入力が行われていないためにリソース作成が失敗します。パスワード検証は [user/password] ダイアログで発生しますが、ユーザープロンプトに有効なユーザーが実際に入力された場合のみです。デフォルトのユーザー名を使用する場合でも、 **create** 操作時にそのユーザー名を指定する必要があります。<br><br>• **ユーザー名が指定されない場合、パスワードの入力要求が省略されます**<br><br>ユーザー名を入力しない場合、[user/password] ダイアログはパスワードの入力要求を省略します。UI オプションを使用してユーザー / パスワードを更新する際に Sybase のユーザー名を入力しない場合、デフォルトの「sa」が使用され、そのアカウントのパスワード検証は実行されません。これにより、無効な認証情報のエラーが発生して、データベースの監視が失敗します。デフォルトのユーザー名を使用する場合でも、update 操作時にそのユーザー名を指定する必要があります。この問題を解決するには、以下の手順を実行してください。<br><br>1. 必要な Sybase データファイルが、目的のサーバーから現在アクセス可能であることを確認します。多くの場合、プライマリ上のローカルリカバリの障害を監視するため、これはバックアップサーバーです。<br><br>2. このサーバーのコマンドラインから Sybase データベースインスタンスを開始してください (データベースを手動で開始する方法の詳細は、Sybase 製品のドキュメンテーションを参照)。<br><br>3. コマンドラインを使用して、LKROOT/bin ディレクトリ (ほとんどのインストールでは /opt/LifeKeeper/bin) に移動 (cd) してください。<br><br>4. bin ディレクトリで次のコマンドを実行してください。<br>./ins_setstate -t <SYBASE_TAG> -S ISP<br><SYBASE_TAG>は Sybase リソースのタグ名です。<br><br>5. コマンドの実行が完了した後、ただちに UI から **Update User/Password Wizard** を実行 |

し、有効なユーザー名を入力してください (Sybase のデフォルトの「sa」を使用する場合も同様)。
注記 : **Update User/Password Wizard** にアクセスするには、Sybase リソースインスタンスを右クリックし、 **[Change Username/Password]** を選択してください。

6. ローカルサーバー上で階層が更新された後、リソースがすべてのノードで in service にすることができることを確認してください。

- **Sybase のローカルユーザー名が 8 文字以上の場合、バックアップサーバーの保護が失敗します**

  Sybase のユーザー名は 8 文字未満にする必要があります。Sybase のローカルユーザー名が 8 文字以上の場合、リソース作成に使用されるプロセスとユーザー ID のチェック、および監視が失敗します。また、これにより、有効な Sybase Backup Server のインスタンスを保護の対象として選択できなくなります。この問題は、オペレーティングシステムが 8 文字以上のユーザー名を各種コマンド (ps など) の UID に変換することに起因します。7 文字以下のユーザー名を使用する必要があります。

**リソース作成問題:**

- Sybase のデフォルトのインストールプロンプトは、16.0 SP02(/opt/sybase) に基づきます。SIOS Protection Suite でリソースを作成する際に、Sybase のインストール場所のデフォルトプロンプトが、Sybase バージョン 16.0 SP02(/opt/sybase) からの相対パスとして表示されます。リソース作成のプロンプトには、Sybase の正しいインストール場所を手動で入力するか、参照する必要があります。

**拡張の問題:**

- **拡張時に Sybase のタグプロンプトが編集可能ですが変更しないでください** 拡張時に Sybase のタグプロンプトが編集可能ですが、これは推奨されません。サーバーごとに異なるタグを使用すると、コマンドラインによるリモート管理で問題が発生するおそれがあります。

**[Properties] ページの問題:**

- **[Properties] ペインに [update user/password] の画像が表示されません** 正しい画像の代わりに小さい四角がツールバーに表示されます。この四角を選択すると、 **User/Password Update Wizard** が起動されます。

**Sybase Monitor Server** は、Sybase 15.7 と SIOS Protection Suite の組み合わせではサポートされていません。Sybase 15.7 で Sybase Monitor Server プロセスを設定する場合、Generic Application (gen/app) リソースを使用してこのサーバープロセスを保護する必要があります。

**Sybase RK を使用してリソースを作成できない、または SLES 15 を使用している場合に LifeKeeper のアップグレードで問題が発生する**

**症状:**
SLES 15（SLES 15 SP1 または SP2 を含む）を設定する場合、LifeKeeper のアップグレードまたは新規

インストール時に Sybase RK が動作しない場合があります。

LifeKeeperの新規インストール中は、Sybaseリソースを作成できません。また、LifeKeeperのアップグレード中に、LifeKeeperからのリソース制御で問題が発生する可能性があります。この問題の元になっているエラーは、「An error occurred when attempting to allocate localization-related structures（ローカリゼーション関連の構造を割り当てようとした際にエラーが発生しました）」です。

**解決方法:**
解決方法は2つあります。

1. ファイル */sybase/<SID>/locales/locales.dat* の [Linux] セクションに、次の行を追加します。
   `locale = POSIX, us_english, utf8`

   または、

2. 次の行を、 *syb<SID>* ユーザープロファイルの Sybase プロファイルに追加します。このプロファイルは通常、 */sybase//SYBASE.sh* にあります。
   `export LANG=en_US.UTF-8`

   <SID> は 3文字のシステム ID です。

   これらの変更により強制的に正しい言語が使用されるようになり、米国英語のロケールに適したものになります。日本や中国で別のロケールを使用している場合は、現地の国やシステムのロケールを確認してください。

   また、 syb<SID> ユーザーに su して `echo $LANG` コマンドを実行することもできます。cshell パラメーターは、適切なロケールの正しい言語を指します。

   一方の方法ではシェルスクリプトを変更し、他方ではロケールに POSIX サポートを追加しています。Sybase (SAP ASE) データベース管理者のやりやすいほうで変更してください。

**Sybase ARK　の稼働を検出できません**

**症状** ：Sybase ARK の稼働を検出できません

**原因** ：Sybase ARK はデフォルトの sql interface tool (isql) を使用します。一部の 64 ビットシステム上では、この isql toolは、isql ではなくisql64 としてインストールされています。

**解決方法** ： isql64 ツールを同一パスにおいて isql としてコピーしてください。もしくは isql64 に対して isql という名前でリンクを作成してください。

# 5.4.5.3.18. WebSphere MQ Recovery Kit – 既知の問題と制限

| 説明 |
| --- |
| **lksupport**コマンド実行時のエラー<br><br>MQ RK の保護下にある MQ キューマネージャーが NFS 共有されているディスク上に配置されている場合、 lksupportコマンドを実行すると、以下のエラーが出力されることがあります。<br><br>`cat: <PATH>/mqm/qmgrs/tkqmgr/qm.ini: Operation not permitted`<br><br>これは、NFS 領域への root アクセスが禁止されていることにより発生するものです。 このエラーが出力されることにより発生する問題はありません。 |
| **quickCheck** がモニタしているキューに長いメッセージ**(101**文字以上**)**を書き込まれた場合、異常と判定してしまいます |
| **MQ** 標準のディレクトリ以外にインストールされている場合、リカバリキットのインストールに失敗します |
| **MQ** のパッケージ名を変更した場合、リカバリキットのインストールに失敗します |
| **MQ** が標準のディレクトリ以外にインストールされている場合、サンプルのビルドに失敗します |
| 複数のリスナーを一つのインスタンスに定義し、自動起動と手動起動が混在している場合、リソース作成および **quickCheck** に失敗することがあります |

# 5.4.5.3.19. SAP HANA – 既知の問題と制限

| 説明 |
| --- |
| **SAP HANA 管理ユーザーのシェル制限**<br><br>SAP HANA 管理ユーザー (つまり SAP HANA インストールの <sid>adm ユーザー) は、 デフォルトのログインシェルである BASH を使用するように設定する必要があります。 (SAP HANA インストール時のデフォルトの選択になっています。) <sid>adm ユーザーに対してデフォルト以外のログインシェル (例えば tcsh/csh) を使用すると、SAP HANA リソースのスイッチオーバーもしくはフェイルオーバー時に予期しない障害が発生する可能性があります。 |

# 5.4.5.4. GUI トラブルシューティング

LifeKeeper GUI をリモートシステムから設定する際に問題が発生した場合は、以下のトピックを参照してください。

ネットワーク関連トラブルシューティング（GUI）

# 5.4.5.4.1. ネットワーク関連トラブルシューティング (GUI)

LifeKeeper は GUI クライアントとサーバの通信に Java RMI (Remote Method Invocation) を使用します。問題となりうる要素の一部は RMI に関連し、それ以外は一般的なネットワークの設定に関する問題です。

## Windows プラットフォームでの論理接続の遅延

### Sun FAQ から:

最も蓋然性が高いのは、ホストのネットワーク設定が誤りというものです。RMI は Java API ネットワーククラス、特に ava.net.InetAddress を使用します。これは、アドレスマッピングおよびホスト名へのアドレスに対して両方のホストに TCP/IP ホスト名のルックアップを実行させます。Windows では、ルックアップ機能はネイティブ Windows ソケットライブラリで実行されるので、遅延は RMI ではなく、Windows ライブラリで発生するものです。ホストが DNS を使用するように設定されている場合、これは、通信に関連するホストについて認識しないという DNS サーバの問題となる可能性があります。その場合、DNS ルックアップのタイムアウトが発生します。このケースに当てはまる場合は、ファイル *\windows\system32\drivers\etc\hosts* で関連ホスト名/アドレスをすべて指定してください。通常のホストファイルのフォーマットを次に示します。

IP アドレス サーバ 名称

例:

*192.168.100.1 homer.example.com homer*

これで、最初のルックアップにかかる時間を短縮できるはずです。

また、サブネットマスクとゲートウェイアドレスの設定が誤っていると、接続の遅延や障害を引き起こす可能性があります。これらの設定が正しいことをネットワーク管理者に確認してください。

## モデムからの実行:

サーバが存在するネットワークにモデムで (PPP または SLIP を使用して) 接続する場合、コンピュータは一時的な IP 番号を操作用に取得します。この一時的な番号は、ホスト名がマップしたものではない可能性があります (ホスト名が何かにマップしている場合)。そのため、この場合は、IP のみで通信するようにサーバに指示する必要があります。これには、モデム接続ウィンドウを開いて一時的な IP 番号を取得します。この番号を使用して、GUI クライアントのホスト名プロパティを設定します。

プラグインでブラウザのホスト名を設定するには、 **[Java Plug-In Control Panel]** を開き、 **[Java Run Time Parameters]** に以下の値を追加してクライアントのホスト名を設定します。

```
-Djava.rmi.server.hostname=<MY_HOST>
```

HotJava ブラウザのホスト名を設定するには、hotjava コマンドラインに以下の値を追加します。

```
-Djava.rmi.server.hostname=<MY_HOST>
```

たとえば、以下のようになります。

```
-Djava.rmi.server.hostname=192.168.100.2
```

## プライマリネットワークインターフェースのダウン:

LifeKeeper GUI は、GUI クライアントと GUI サーバの通信を維持するために Remote Method Invocation (RMI) を使用します。ほぼどのような場合でも、プライマリネットワーク・インターフェースを介してサーバへの接続が確立されます。つまり、サーバのプライマリイーサネットインターフェースがダウンした場合、接続は失われ、GUI クライアントに [Unknown] というサーバの状態が表示されます。

この問題の唯一の解決策は、サーバのプライマリイーサネットインターフェースを再び有効にすることです。また、RMI の制限のため、マルチホームサーバ (複数のネットワークインターフェースを備えたサーバ) でこの問題を解決することはできません。

## ホストへのルートが存在しない例外:

ホストに接続できなかったため、ソケットをリモートホストに接続できませんでした。これは通常、ネットワークのローカルサーバとリモートホストの間のリンクの一部がダウンしたか、ホストがファイアウォールの後ろにあることを意味します。

## 不明なホストの例外:

LifeKeeper GUI クライアントとサーバは、通信に Java RMI (Remote Method Invocation) 技術を使用します。RMI が正常に動作するために、クライアントとサーバは解決可能なホスト名または IP アドレスを使用する必要があります。解決不可能な名前、WINS 名、修飾されていない DHCP 名を使用した場合、Java は UnknownHostException を送出します。

このエラーメッセージは、以下の条件でも発生する可能性があります。

- サーバ名が存在しない場合。サーバ名の誤記がないか確認してください。
- 設定された DHCP サーバが、RMI サーバが実際に存在するドメインではなく、リゾルバドメインのドメイン名になるように RMI サーバの完全修飾ドメイン名を設定している場合。この場合、サーバの DHCP ドメインの外側の RMI クライアントは、不正なドメイン名のためにサーバにアクセスできません。
- サーバが、Windows Internet Naming Service (WINS) を使用するように設定されたネットワーク上にある場合。DNS にのみ依存しているホストは、WINS の下に登録されたホストにアクセスできない場合があります。
- RMI クライアントとサーバがファイアウォールをはさんだ反対側にある場合。ファイアウォールの

外側に RMI クライアント、内側にサーバがある場合、クライアントはサーバに対してリモート呼び出しを実行できません。

LifeKeeper GUI を使用している場合、クライアントによって提供されたホスト名はサーバから解決できるものであり、サーバからのホスト名はクライアントによって解決できるものである必要があります。LifeKeeper GUI はこの例外を捕捉し、ユーザに警告します。クライアントがサーバのホスト名を解決できない場合、この例外が捕捉され、メッセージ 115 が表示されます。サーバがクライアントのホスト名を解決できない場合、この例外が捕捉され、メッセージ 116 が表示されます。どちらのメッセージにも、実行が試された未修飾ホスト名を指定する Java 例外の一部が含まれています。

下記に、ホスト名の解決が正常に機能していることをテストまたは検証するために使用できる手順をいくつか示します。

## Windows から:

1. Linux サーバとの通信の確認

   DOS プロンプトから、ホスト名を使用してターゲットを ping します。

   ```
   ping <TARGET_NAME>
   ```

   たとえば、以下のようになります。

   ```
   ping homer
   ```

   ターゲットの修飾されたホスト名と IP アドレスをリストする応答が表示されるはずです。

2. 正しい設定の確認

- ◦ DNS の設定を確認するか、ネットワークに DNS サーバをインストールします。
- ◦ *ControlPanel->Network->Protocols->TCP/IP* の設定を確認します。これらの設定が正しいことをネットワーク管理者に確認してください。

  [DNS] タブのホスト名は、ローカルネームサーバで使用されているものと一致している必要があります。これは、GUI エラーメッセージで指定したホスト名とも一致している必要があります。

- ローカルホストおよびその接続先となる LifeKeeper サーバのエントリを含める形で hosts ファイルを編集してください。

Windows 95/98 システムでは、hosts ファイルは以下のようになります。

```
%windir%\HOSTS (for example, C:\WINDOWS\HOSTS).
```

**注記**: Windows 95/98 では、hosts ファイルの最後のエントリがキャリッジリターン (CR) またはラインフィード (LF) で終わっていない場合、hosts ファイルはまったく読み取られません。

Windows NT システムでは、hosts ファイルは以下のようになります。

```
%windir%\System32\DRIVERS\ETC\HOSTS
(for example, C:\WINNT\System32\DRIVERS\ETC\HOSTS).
```

たとえば、システムが *HOSTCLIENT.EXAMPLE.COM* と呼ばれ、IP アドレスとして *192.168.200.10* を使用している場合、hosts ファイルに次のエントリを追加します。

```
192.168.200.10 HOSTCLIENT.EXAMPLE.COM HOSTCLIENT
```

3. GUI クライアントで使用するホスト名プロパティを設定してください。プラグインでブラウザからホスト名を設定するには、 **[Java Plug-In Control Panel]** を開き、 **[Java Run Time Parameters]** に以下の値を追加してクライアントのホスト名を設定します。

   ```
   Djava.rmi.server.hostname=<MY_HOST>
   ```

4. Microsoft のネットワーク関連のパッチを www.microsoft.com で確認してください。

## Linux から:

1. ホスト名または IP アドレスを使用して Linux からターゲットサーバを ping し、他のサーバとの通信を確認します。

   ```
   ping <TARGET_NAME>
   ```

   たとえば、以下のようになります。

   ```
   ping homer
   ```

   ターゲットの修飾されたホスト名をリストする応答が表示されるはずです。

2. ホスト名または IP アドレスで **ping** を実行し、クラスタ内の各サーバで *ローカルホスト* が解決可能であることを確認します。DNS が実装されていない場合、 */etc/hosts* ファイルを編集し、 *ローカルホスト* 名のエントリを追加します。このエントリで、ローカルサーバの IP アドレスまたはデフォルトエントリ (127.0.0.1) をリストできます。

3. DNS が NIS の前に指定されていることを確認します。 */etc/nsswitch.conf* の hosts 行で DNS を NIS の前に置く必要があります。また、 */etc/resolv.conf* は正しく設定された DNS サーバを指す必要があります。

4. DNS を実装しない場合、または他の方法がうまくいかない場合は、 */etc/hosts* ファイルを編集し、ホスト名のエントリを追加します。

5. GUI クライアントで使用するホスト名プロパティを設定してください。これは、管理者ごとに変更する必要があります。

   プラグインでブラウザからホスト名を設定するには、 **[Java Plug-In Control Panel]** を開き、 **[Java Run Time Parameters]** に以下の値を追加してクライアントのホスト名を設定します。

```
-Djava.rmi.server.hostname=<MY_HOST>
```

HotJava ブラウザからホスト名を設定するには、hotjava コマンドラインに以下の値を追加します。

```
-Djava.rmi.server.hostname=<MY_HOST>
```

たとえば、以下のようになります。

```
-Djava.rmi.server.hostname=192.168.200.10

-Djava.rmi.server.hostname= homer.example.com
```

## X Window Server に接続できない:

LifeKeeper GUI アプリケーションを telnet セッションから実行している場合、GUI クライアントが LifeKeeper サーバで X Window Server にアクセスできることを確認する必要があります。LifeKeeper サーバは GUI クライアントのホスト名またはネットワークアドレスを解決できる必要があります。

LifeKeeper サーバに対して telnet を実行して LifeKeeper GUI アプリケーションを実行した場合、*DISPLAY* 環境変数にはクライアントのホスト名と表示番号を含める必要があります。たとえば、 *Server1* というサーバに *Client1* というクライアントから telnet を実行した場合、 *DISPLAY* 環境変数は *Client1:0* に設定される必要があります。LifeKeeper GUI アプリケーションを実行した場合、 *Client1* の *DISPLAY* 名に出力が送信されます。 *Client1* が *X Window Server* にアクセスできない場合、例外が発生して LifeKeeper GUI アプリケーションは失敗します。

LifeKeeper GUI をアプリケーションとして起動したときに、 *X Window Server* に接続できない、またはクライアント *DISPLAY* 名を開くことができないというエラーが発生した場合は、以下の手順を実行してください。

1. ホスト名または IP アドレスを使用して表示変数を設定します。たとえば、以下のようになります。

   ```
   DISPLAY=Client1.somecompany.com:0

   DISPLAY=172.17.5.74:0
   ```

2. xhost または xauth コマンドを使用し、クライアントが LifeKeeper サーバで *X Window Server* に接続できることを確認します。

3. クライアント用の DNS エントリを追加するか、クライアント用のエントリを LifeKeeper サーバのローカルホストファイルに追加します。LifeKeeper サーバからクライアントに対して、ホスト名または IP アドレスを使用して ping を実行し、クライアントとの通信を確認します。

# 5.4.5.5. コミュニケーションパスの稼働と停止

コミュニケーションパスの停止と稼働が繰り返される場合（LifeKeeper GUI でAlive、Dead、Aliveというように表示される場合）、ハートビートの設定がクラスタ内のすべてのサーバで同じ値に設定されていない可能性があります。

この状態は、いずれか一方のサーバにある LifeKeeper デフォルトファイル **/etc/default/LifeKeeper** で設定名に誤記がある場合にも発生する可能性があります。

## 推奨される対策

1. クラスタ内のすべてのサーバで LifeKeeper を停止します。

2. クラスタ内の各サーバで、 **/etc/default/LifeKeeper** にある **LCMHBEATTIME** 設定と **LCMNUMHBEATS** 設定の値とスペルを確認します。設定値、スペルミスの無いことを各ノードで確認します。

3. クラスタ内のすべてのサーバで LifeKeeper を再起動します。

# 5.4.5.6. 不完全なリソースの作成

インスタンスの一部のみが作成された状態で、リソース設定プロセスが中断された場合、階層を再設定する前に、手動でクリーンアップする必要があります。LifeKeeper GUI を使用し、一部が作成されたリソースを削除してください。手順については、すべてのサーバからの階層の削除 を参照してください。階層リストにこれらのリソースが含まれていない場合、ins_remove（LCDI-instances(1M) を参照）および dep_remove（LCDI-relationship(1M)）を使用し、部分的な階層をクリーンアップしなければならない可能性もあります。

# 5.4.5.7. 不完全なリソースの優先順位の変更

LifeKeeper の階層は、親子の関係によって関連付けられたすべてのリソースとして定義されています。複数の親を持つリソースの場合、GUI と階層のすべてのリソースを区別することは一概に簡単とも言えなくなります。階層の整合性を保持するには、サーバごとに階層内のすべてのリソースに対して優先順位を変更する必要があります。[OK] または [Apply] ボタンを押した後で選択される階層のすべてのルートリソースを表示することで、GUI はこの要件を強制します。この時点で、すべてのルートを受け付けるか、操作をキャンセルするかを選択できます。ルートのリストを受け付けた場合、新しい優先順位の値が階層内のすべてのリソースに割り当てられます。

その階層の [Resource Properties] ダイアログが表示されている間、他の変更を階層に加えていることを確認する必要があります。[Resource Properties] ダイアログの優先順位を編集する前に、LifeKeeper に加えられた変更が動的にダイアログで更新されます。ただし、変更を加えると、基本的な変更が LifeKeeper で加えられた場合でも、ダイアログの値は凍結されます。[Apply] または [OK] ボタンをクリックした後でのみ、変更が加えられたことが通知されるので、優先順位の変更操作は要求どおりに進みません。

複数の優先順位の変更を伴う優先順位の変更操作時に、復旧できないエラーの可能性を最低限に抑えるには、プログラムは、一度に 1 つのサーバに対して個別に行われる一連の変更として、複数の優先順位の変更操作を実行します。また、操作時に優先順位の競合を防ぐために、必要に応じて一時的な値がプロパティに割り当てられます。この一時的な値は、最大許容値 999 を超えるもので、優先順位の変更中に一時的に GUI に表示されることもあります。操作が完了すると、一時的な値はすべて、要求された値に置き換えられます。エラーが発生し、優先順位の値をロールバックできない場合、一時的な優先順位の値の一部がそのまま残る可能性もあります。この場合は、下記の推奨手順に従って階層を修復してください。

## 一貫した状態への階層のリストア

優先順位の変更操作の間にエラーが発生し、操作を完了できない場合、優先順位は不整合の状態のまま残る可能性があります。エラーは、システムやコミュニケーションパスの障害を含め、さまざまな理由で発生します。操作が開始された後や完了する前にエラーが発生し、プログラムが前の優先順位にロールバックできなかった場合、操作中にエラーがあったこと、および前の優先順位を restore できなかったことを示すメッセージが表示されます。この場合、以下の処置を実行し、階層を一貫性のある状態に restore する必要があります。

1. 可能であれば、問題の原因を特定します。システムまたはコミュニケーションパスの障害を確認します。優先順位管理プログラムの実行中に、その他の操作が行われていないことを確認します。

2. 可能であれば、問題の原因を修正してから先に進みます。たとえば、階層を修復する前に、障害が発生したシステムまたはコミュニケーションパスを restore する必要があります。

3. [Resource Properties] ダイアログから操作を再試行します。

4. [Resource Properties] ダイアログから変更できない場合は、コマンドライン `hry_setpri` を使用して階層を修復するとより簡単かもしれません。このスクリプトを使用すると、一度に 1 つのサーバに対して優先順位を変更できます。このスクリプトは、GUI からは実行できません。

5. 修復を実行したら、階層が存在するすべてのサーバに対して `eqv_list` コマンドを実行し、階層の

すべてのリソースに対して返された優先順位の値を調べ、LifeKeeper データベースがすべてのサーバで一貫していることを確認します。

6. 最終的に、階層を修復できない場合は、階層を削除して再作成する必要がある可能性もあります。

# 5.4.5.8. 階層の設定中に共有ストレージが見つからない

リソースの階層を設定中に、LifeKeeper が「No shared storage」(共有ストレージがありません) というメッセージをレポートする状況がいくつかあります。

**考えられる原因**：ストレージを共有するサーバー間でコミュニケーションパスが定義されていません。共有ストレージデバイスで階層が設定されている場合、LifeKeeper は、クラスタ内の別のサーバを少なくとも 1 つ検証し、その共有ストレージにアクセスできることを確認します。

**推奨される対策:** LifeKeeper GUI または `lcdstatus (1M)` を使用し、コミュニケーションパスが設定されており、アクティブになっていることを確認します。

**考えられる原因:** ストレージを共有するサーバー間でコミュニケーションパスが機能していません。

**推奨される対策:** LifeKeeper GUI または `lcdstatus (1M)` を使用し、コミュニケーションパスが設定されており、アクティブになっていることを確認します。

**考えられる原因:** Linux が共有ストレージにアクセスできない。この原因としては、ドライバがロードされていないことや、ドライバがロードされたときにストレージの電源が入っていないこと、あるいはストレージデバイスが正しく設定されていないことなどが考えられます。

**推奨される対策:** `/proc/scsi/scsi` でデバイスが正しく定義されていることを確認します。

**考えられる原因:** LifeKeeper を起動する前にストレージが Linux で設定されていない。LifeKeeper の起動時に、すべての SCSI デバイスがスキャンされ、デバイスのマッピングが判別されます。LifeKeeper の起動後にデバイスが設定された (電源がオンにされた、接続された、またはドライバがロードされた) 場合、デバイスを設定して使用できるようにするには、LifeKeeper を停止してから再起動する必要があります。

**推奨される対策:** `$LKROOT/subsys/scsi/resources/hostadp/device_info` にデバイスがリストされていることを確認してください。`$LKROOT` は、デフォルトでは /opt/LifeKeeper. です。デバイスがこのファイルにリストされていない場合、LifeKeeper はそのデバイスを使用しません。

**考えられる原因:** ストレージがサポートされていない。サポートストレージ一覧には、LifeKeeper で動作がテストされ、サポートされている具体的な SCSI デバイスが列挙されています。ただし、このリストに含まれているのは、すでに知られているデバイスなので注意してください。LifeKeeper の要件を満たしているものの、SIOS Technology Corp. がテストしていないデバイスが存在する可能性もあります。

**推奨される対策:** `$LKROOT/subsys/scsi/resources/hostadp/device_info` にデバイスがリストされていることを確認してください。`$LKROOT` は、デフォルトでは `/opt/LifeKeeper` です。デバイスがこのファイルにリストされているものの、デバイス名の後に来る ID が「NU-」で始まる場合、LifeKeeper はデバイスから一意の ID を取得できなかったことを示します。一意の ID がない場合、LifeKeeper はデバイスが共有されているかどうかを判別できません。

**考えられる原因:** ストレージでは、デバイスを LifeKeeper で使用できるようにする前に、特定の LifeKeeper ソフトウェアをインストールする必要があります。たとえば、Raw I/O サポートを有効にするための **steeleye-lkRAW** キット、データレプリケーションを有効にするための **steeleye-lkDR** ソフトウェアなどです。

**推奨される対策:** 必要な LifeKeeper パッケージが各サーバにインストールされていることを確認します。ソフトウェアの要件については、 SPS for Linux リリースノート を参照してください。

補足のヒント：

test_lk(1M)ツールを使用すると、ストレージおよび通信の問題のデバッグに役立ちます。

# 5.4.5.9. LifeKeeper サーバ障害からの復旧

LifeKeeper クラスタ内のサーバに、オペレーティングシステムの再インストールを (したがって LifeKeeper の再インストールも) 必要とする障害が発生した場合、クラスタの各サーバからリソース階層を再拡張する必要があります。ただし、再インストールしたサーバとの共有イクイバレンシ関係がクラスタのサーバにある場合、LifeKeeper は、再インストールしたサーバへ既存のリソース階層を拡張することを許可しません。また、再インストールされたサーバには階層が実際には存在していないため、再インストールしたサーバから階層を拡張解除することもできません。

## 推奨される対策:

1. リソース階層が設定されている各サーバで、eqv_list コマンドを使用してすべての共有イクイバレンシのリストを取得します (詳細については、`LCDI-relationship` を参照してください).

   下記の例では、server1 および server2 に対する IP リソースの iptag のコマンドおよび結果の出力を示します。ここでは、server2 が再インストールされたサーバ、server1 が設定された階層です。

   ```
   eqv_list -f:

   server1:iptag:server2:iptag:SHARED:1:10
   ```

2. リソース階層が設定された各サーバで、eqv_remove を使用して、階層の各リソースのイクイバレンシ関係を手動で削除します (詳細については、`LCDI-relationship` を参照してください)。

   たとえば、上記の手順 1 の例を基に、server1 に対して以下のコマンドを実行します。

   ```
   eqv_remove -t iptag -S server2 -e SHARED
   ```

3. 2つ以上のサーバがあるクラスタでは、これらのリソース階層のイクイバレンシ関係が定義されているクラスタ内の各サーバに対して手順 1 と 2 を繰り返します。

4. 最後に、GUI を使用し、リソース階層が in-service になっているサーバから再インストールされたサーバに各リソース階層を拡張します。

# 5.4.5.10. 停止できないプロセスからの復旧

プロセスが停止不可の場合、LifeKeeper は共有ディスクパーティションをアンマウントできない可能性があります。そのため、リソースを別のシステムで In Service にすることができません。停止できないプロセスから復旧する唯一の方法は、システムを再起動することです。

# 5.4.5.11. 手動リカバリ時のパニックからの復旧

手動スイッチオーバ時に PANIC になると、リカバリが不完全に終わる可能性があります。PANIC またはその他の大きなシステム障害が手動スイッチオーバ時に発生した場合、バックアップシステムへの完全自動リカバリは保証できなくなります。In Service になる必要があるすべてのリソースが In Service であることをバックアップシステムで確認してください。In Service ではないリソースがあった場合は、LifeKeeper GUI を使用して、そのリソースを手動で In Service にします。手順については、リソースを In-Service する を参照してください。

# 5.4.5.12. Out-of-Service 階層の復旧

LifeKeeper サーバの障害からの復旧の一環として、障害が発生したサーバで設定されているものの、サーバの障害時にどのサーバでも In Service ではなかったリソース階層が、障害時に最優先で Alive になったサーバで復旧されます。これは、障害が発生したサーバ、復旧中のサーバ、階層内の他のサーバを含め、Out of Service の階層が最後にどこで In Service だったかには無関係です。

# 5.4.5.13. リソースタグ名の制限

## タグ名の長さ

LifeKeeper 内のすべてのタグは、256 文字以内にする必要があります。

## 有効な特殊文字

| |
|---|
| - _ . / |

タグの最初の文字に「.」および「/」を使用することはできません。

## 無効な文字

| |
|---|
| + ; : ! @ # $ * = 「スペース」 |

# 5.4.5.14. シリアル (TTY) コンソールの警告

シリアルコンソールデータパスの一部が信頼できない場合、または Out of Service になった場合、シリアル (RS-232 TTY) コンソールを使用するユーザは LifeKeeper サービスで深刻な問題に直面する可能性があります。操作中に、LifeKeeper はコンソールメッセージを生成します。設定に (標準的な VGA コンソールではなく) シリアルコンソールがある場合、これらのコンソールメッセージが確実に配信されるようにするために、LifeKeeper からエンドユーザターミナルへのデータパス全体が機能している必要があります。

ターミナルの電源オフ、モデムの未接続、ケーブルのゆるみなど、データパスがつながっていない場合、Linux STREAMS ファシリティは、コンソールメッセージをキューに入れます。STREAMS キューがいっぱいになった場合、Unix カーネルは、STREAMS バッファキューにメッセージを入れる余地ができるまで LifeKeeper を保留にします。これにより、LifeKeeper がハングすることもあります。

**注記:** LifeKeeper 環境のシリアルコンソールは可能なかぎり避け、VGA コンソールを使用することを推奨します。シリアルコンソールを使用する必要がある場合、シリアルコンソールがオンになっていること、ケーブルとオプションのモデムが正しく接続されていること、メッセージが表示されていることを必ず確認してください。

# 5.4.5.15. システムが init 状態 S に遷移しているという警告

LifeKeeper が動作している場合、システムを直接、init 状態 S に切り替えないでください。Linux の init システムの操作が原因で、こうした遷移が全 LifeKeeper プロセスの即時停止につながり、突発的な障害を発生させる可能性があります。この場合は、代わりに LifeKeeper を (`lkstop` で) 手動停止するか、システムを最初に init 状態 1 にしてから init 状態 S にしてください。

# 5.4.5.16. 共有ストレージでスレッドがハングしているというメッセージ

デバイス確認スレッドがそれほど迅速に処理を完了していない場合、スレッドがハングしているというメッセージが LifeKeeper ログに記録されることがあります。これにより、リソースがあるサーバから別のサーバに移動し、さらに悪いケースでは、サーバが停止する可能性があります。

## 説明

(/etc/default/LifeKeeperの) FAILFASTTIMER は、各デバイスが正常に動作していること、および特定のシステムによって所有されているすべてのリソースがそのシステムからアクセス可能で、そのシステムに所有されていることを確認するための秒数を定義します。FAILFASTTIMER は、この所有権を確定し、データの信頼性を最大限に確保するために、可能なかぎり小さくする必要があります。ただし、デバイスがビジー状態で、負荷がピークの場合、指定した時間内で応答できない可能性もあります。デバイスの操作が FAILFASTTIMER よりも長くかかっている場合、LifeKeeper はデバイスがハングしている可能性を検討します。FAILFASTTIMER の時間を 3 回繰り返してもデバイスが応答しない場合、LifeKeeper は、デバイスに障害が発生したものとみなして、リカバリを実行します。リカバリプロセスは、SCSIERROR の設定で定義します。SCSIERROR の設定によっては、ローカルリカバリを実行し、失敗した場合はスイッチオーバを実行するために sendevent が発行されることもあります。この操作がない場合、システムが停止するおそれもあります。

## 推奨される対策:

ハングメッセージがまれにエラーログに出力され、もうハングしていないというメッセージがそれに続く場合、さらに括弧の数が常に 1 つの場合、それほど警戒する理由はありません。ただし、このメッセージが頻繁にログに記録され、数が 2 または 3 の場合、以下の 2 つの処置が必要になる可能性があります。

- ◦ ストレージの負荷を減らすことを試みる。ストレージの処理に FAILFASTTIMER (デフォルトでは、5 秒または 15 秒を 3 回) の 3 倍の時間がかかっている場合、ストレージに対する負荷を考慮し、I/O の長い遅延を避けるために負荷を分散する必要があります。これにより、LifeKeeper は、デバイスを頻繁に確認できるようになり、さらにそのデバイスを使用しているアプリケーションのパフォーマンスも向上します。

- ◦ 負荷を減らすことができない場合、FAILFASTTIMER をデフォルトの 5 秒から増やすことができます。この値は、できる限り低く抑える必要があります。そのため、メッセージがまったく表示されなくなるか、まれにしか表示されなくなるまで、少しずつ値を増やしてください。

注記: FAILFASTTIMER の値が変更された場合、新しい値を有効にするために、LifeKeeper を終了し、再起動する必要があります。

# 5.5. SIOS DataKeeper for Linux

## はじめに

SIOS DataKeeper for Linux は、LifeKeeper 環境に統合データミラーリング機能を提供します。この機能により、LifeKeeper リソースが共有 / 非共有ストレージ環境で動作可能になります。

SIOS DataKeeper for Linux によるミラーリング

SIOS DataKeeper の仕組み

# 5.5.1. SIOS DataKeeper for Linux によるミラーリング

SIOS DataKeeper for Linux は、共有ストレージを使用せずに可用性の高いクラスタ（SIOS LifeKeeper を使用）を構築したいお客様や、ビジネスに不可欠なデータをサーバ間でリアルタイムに複製したいお客様に別の方法を提供します。

SIOS DataKeeper は、同期または非同期のボリュームレベルのミラーリングを使用して、プライマリサーバ（ミラーソース）から 1 台以上のバックアップサーバ（ミラーターゲット）にデータを複製します。

## DataKeeper の特長

SIOS DataKeeper には、以下の特長があります。

- ◦ TCP/IP ベースのローカルエリアネットワーク（LAN）またはワイドエリアネットワーク（WAN）経由で、リモートの場所に高い信頼性、効率、整合性でデータをミラーリングできます。
- ◦ 同期と非同期のミラーリングをサポートします。
- ◦ 複製はファイルシステムの下のブロックレベルで実行されるので、関与するアプリケーションに対して透過的です。
- ◦ LifeKeeper と共に使用した場合、複数ターゲットへのカスケーディングフェイルオーバも含めて、複数ターゲットへの同時ミラーリングをサポートします。
- ◦ 内蔵のネットワーク圧縮ににより、ワイドエリアネットワークでの最大スループットが向上します。
- ◦ 主要なファイルシステムをすべてサポートします（ファイルシステムのジャーナリングサポートの詳細については、 SPS for Linux リリースノート の製品説明を参照してください）。
- ◦ ミラーリングしたデータにフェイルオーバの保護を提供します。
- ◦ LifeKeeper のグラフィカルユーザインターフェースに統合されています。
- ◦ 他の LifeKeeper Application Recovery Kit をフルにサポートします。
- ◦ システムリカバリ時に、プライマリサーバとバックアップサーバとの間でデータを自動的に再同期します。
- ◦ 障害発生時には、仮想のシステムコンポーネントの健全性を監視し、ローカルリカバリを実行します。
- ◦ I/O フェンス用の STONITH デバイスをサポートします。詳細については、STONITH のトピックを参照してください。

# 同期ミラーリングと非同期ミラーリングの違い

同期ミラーリングと非同期ミラーリングの違いを理解すると、アプリケーション環境に適切なミラーリング方法を選択することができます。

## 同期ミラーリング

SIOS DataKeeper は、プライマリサーバとバックアップサーバに同時にデータを書き込む同期ミラーリング方法を使用して、リアルタイムミラーを実現します。書き込み動作のたびに、DataKeeper は書き込みをターゲットデバイスに転送し、リモート確認を受信してから I/O 完了を通知します。同期ミラーリングの長所は、データの保護レベルが高いことです。これは、常にデータのすべてのコピーを確実に同一にしているからです。ただし、リモート確認を待つために、パフォーマンスが低下することがあり、これは特に WAN 環境で発生します。

## 非同期ミラーリング

非同期ミラーリングでは、それぞれの書き込みがソースデバイスに対して行われ、次に、コピーがターゲットデバイスに送信されるキューに入れられます。これはつまり、任意の時点で、ソースからターゲットデバイスへの送信を待っている多数の書き込みトランザクションが存在する可能性があります。非同期ミラーリングの長所は、書き込みがプライマリディスクに到達した時点で確認されるため、パフォーマンスが高いことです。ただし、プライマリディスクに障害が発生した場合、非同期書き込みキュー内にある書き込みはターゲットに送信されないため、信頼性が低くなります。この問題を緩和するために、SIOS DataKeeper はプライマリディスクに対する個々の書き込みについてインテントログファイルにエントリを作成します。また、多量の書き込みが継続的に行われた場合、キューに書き込まれたデータの送出を優先するため一時的にI/Oパフォーマンスが低下する場合があります。

インテントログとは、プライマリとターゲットのミラー間で同期していないデータブロックを示すビットマップです。サーバの障害発生時にインテントログを使用すると、データ全体の再同期を回避できます。

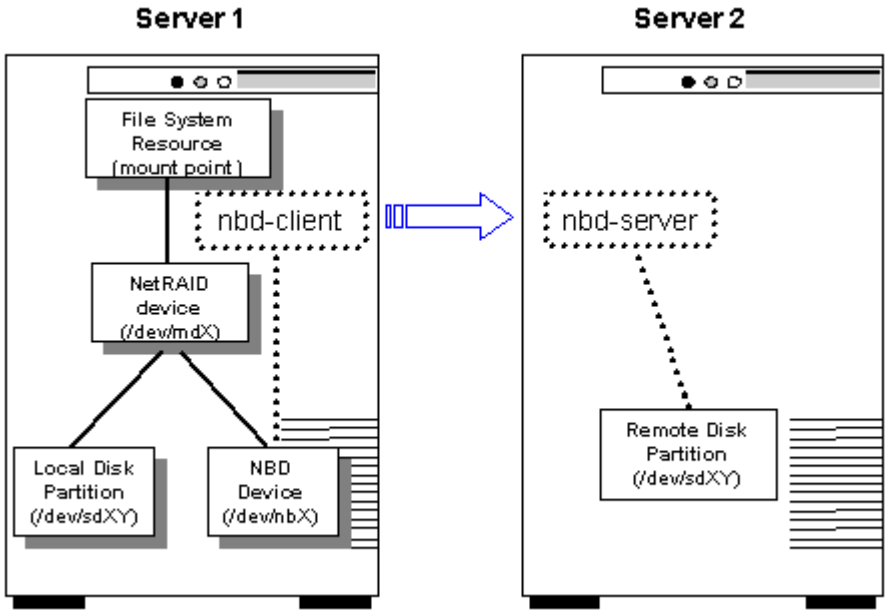
補足: 非同期ミラーの場合、DataKeeperでは、最大4096個の未処理のターゲット書き込みをキューに入れることができます。/etc/default/LifeKeeper の `LKDR_ASYNC_LIMIT` で、さらに多くの書き込みをキューに入れられるように設定してください。

ターゲットへの未処理の書き込みの数が上限に達すると、その数が設定値を下回るまで、ミラーは同期モードに戻ります。

ブロックサイズが4Kであると想定すると、値がデフォルト（書き込みの非同期制限が4096）のままになっている場合、最大16MBのデータが転送されます。

# 5.5.2. SIOS DataKeeper の仕組み

SIOS DataKeeper は、NetRAID デバイスを作成して保護します。NetRAID デバイスは RAID1 のデバイスであり、下図に示すようにローカルのディスクまたはパーティション、およびネットワークブロックデバイス（NBD）で構成されます。




LifeKeeper がサポートするファイルシステムは、その他すべてのストレージデバイスと同様に、NetRAID デバイスにマウントできます。この場合、ファイルシステムは複製されたファイルシステムと呼ばれます。LifeKeeper は、NetRAID デバイスと複製されたファイルシステムの両方を保護します。

ファイルシステムは、DataKeeper [リソース階層を作成することにより作成されます。NetRAID デバイスを別のサーバに拡張すると NBD デバイスが作成され、2 台のサーバ間にネットワーク接続が確立されます。NBD 接続が確立されるとただちに、SIOS DataKeeper がデータの複製を開始します。

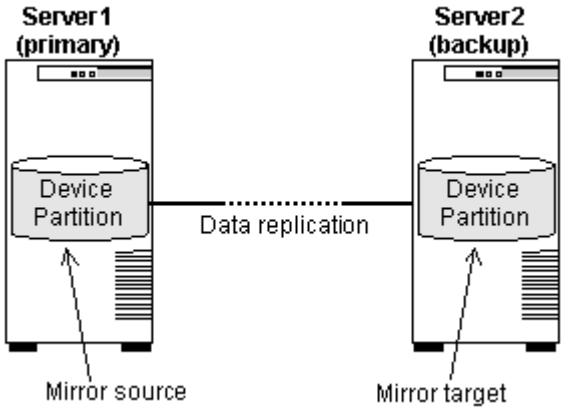
nbd-client プロセスがプライマリサーバで実行され、バックアップサーバで動作している nbd-server プロセスと接続します。

## 同期（および再同期）

DataKeeper リソース階層は作成されてから拡張されるまでの間、デグレードモードです。つまり、データはローカルのディスクまたはパーティションにのみ書き込まれます。階層をバックアップ（ターゲット）システムに拡張すると、SIOS DataKeeper が 2 つのシステム間でデータを同期し、以降の書き込みはすべてターゲットに複製されます。どの時点でもデータが「非同期」になった場合（システムまたはネットワークの障害が発生した場合）、SIOS DataKeeper はソースとターゲットのシステムでデータを自動的に再同期します。インテントログ（ビットマップファイル）を使用するようにミラーが設定されている場合、SIOS DataKeeper はインテントログを使用して非同期のデータを特定するので、全体の再同期は不要です。インテントログ（ビットマップファイル）を使用するようにミラーが設定されていない場合は、データ複製の中断後に全体の再同期が実行されます。

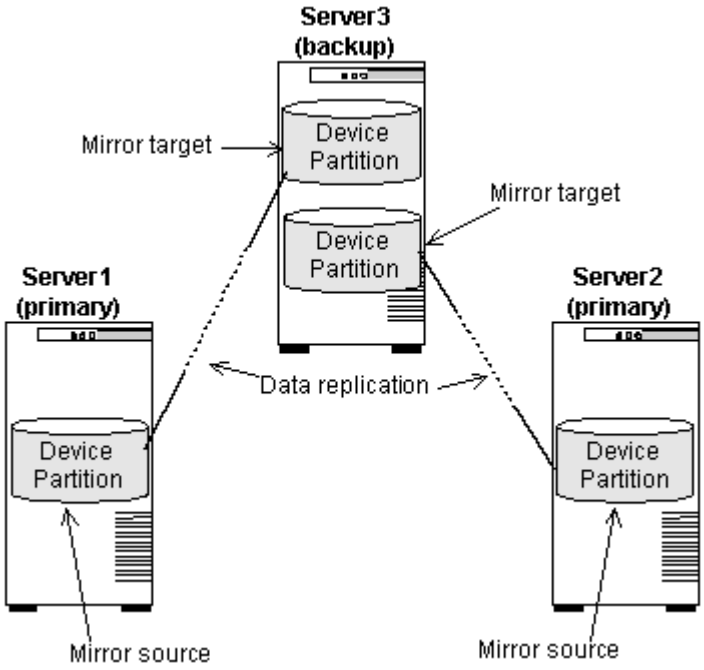
# 標準ミラーの構成

最も一般的なミラーの構成では、下図に示すように 2 台のサーバがあり、各サーバのローカルのディスクまたはパーティションとの間にミラーが確立されます。サーバ 1 は、ミラーソースを持つプライマリサーバです。サーバ 2 は、ミラーターゲットを持つバックアップサーバです。



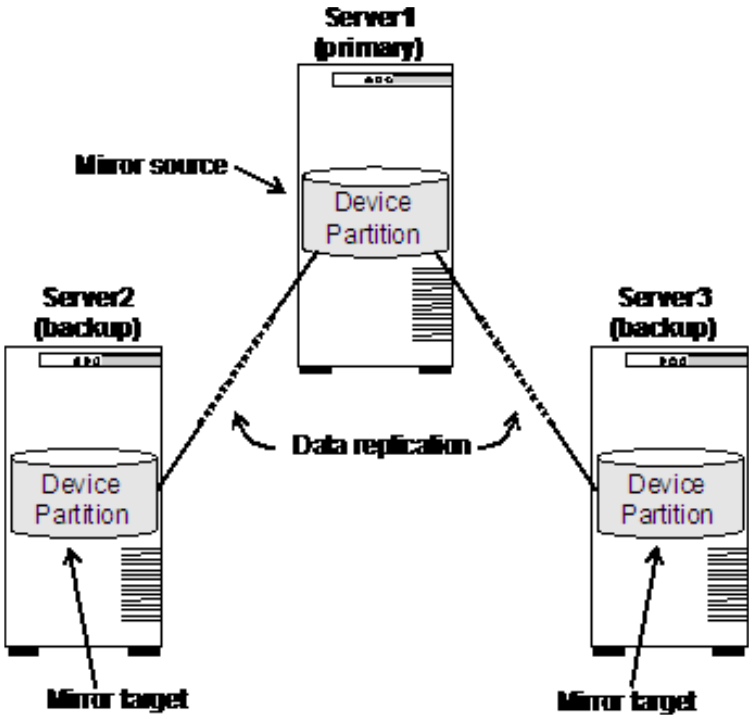

# N+1 Configuration

前述した標準ミラーの構成の変形として一般的に使用される構成では、クラスタ内にある 2 台以上のサーバが共通のバックアップサーバにデータを複製します。この場合は、下図に示すように、各ミラーソースがバックアップサーバの個別のディスクまたはパーティションに複製する必要があります。



# 複数ターゲットの構成

適切な Linux のディストリビューションとバージョン 2.6.7 以降のカーネルと共に使用した場合、下図に示すように、SIOS DataKeeper は、プライマリサーバの 1 つのディスクまたはパーティションから複数のバックアップシステムにデータを複製することもできます。



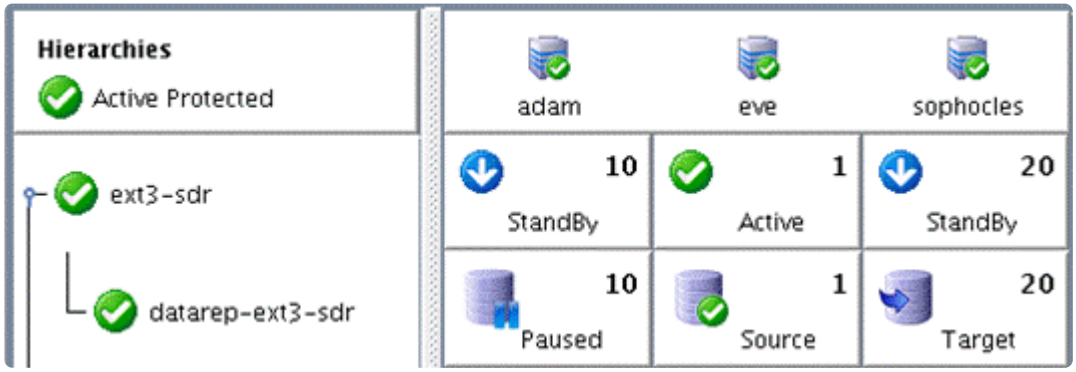

特定のソースのディスクまたはパーティションを最大 7 つのミラーターゲットに複製でき、各ミラーターゲットは別のシステムに存在する必要があります。つまり、ソースのディスクまたはパーティションを、同一ターゲットシステム上にある複数のディスクまたはパーティションにミラーリングすることはできません。

このタイプの構成では、LifeKeeper のカスケーディングフェイルオーバ機能を使用でき、保護するアプリケーションとそのデータに対して複数のバックアップシステムを提供できます。

ミラーリングの開始時にすべてのターゲットに完全再同期されるのを回避するには、クラスター内の残りのターゲットを再接続する前に、まず前のソースのビットマップをマージする必要があります。 9.3.2 以前のバージョンでは、ミラーリングの開始時に前のソースが利用できなかった場合は、各ターゲットに対して完全再同期が自動的に行われます。 9.3.2 からは、ミラーがシステム上で起動されると、ターゲットに接続する前に、前のソースがクラスターに参加するのを待ちます。前のソースがクラスターに参加すると、そのビットマップがマージされ、すべてのターゲットが部分的な再同期で参加できるようになります。ミラーが停止し、ターゲットが同期中の場合、ミラーを起動してターゲットに複製するために前のソースは必要ありません。前のソースがクラスターへの再参加に利用できない場合は、「mirror_action fullresync」コマンドを使用して、ターゲットを完全再同期で手動で再同期することができます。 /etc/default/LifeKeeper の変数 LKDR_WAIT_FOR_PREVIOUS_SOURCE_TIMEOUT によって、再同期の動作が決まります。（詳細は「DataKeeper パラメーターリスト 」を参照）。

# SIOS DataKeeper リソース階層

以下の例に、LifeKeeper の GUI に表示される典型的な DataKeeper リソース階層を示します。




リソース *datarep-ext3-sdr* は NetRAID リソースであり、親リソース *ext3-sdr* はファイルシステムリソースです。本書の以降の部分では、「DataKeeper リソース」は両方のリソースを合わせたものを指すことに注意してください。ファイルシステムリソースは NetRAID リソースに依存するので、NetRAID リソースに対する動作はその上にあるファイルシステムにも影響します。

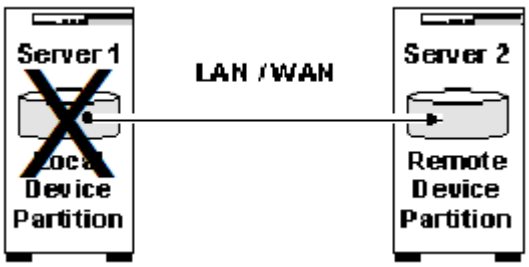
# フェイルオーバのシナリオ

## フェイルオーバのシナリオ – 2 ノード

以下の 4 つの例で、SIOS DataKeeper を使用するフェイルオーバで何が起きるかを説明します。これらの例では、LifeKeeper for Linux クラスタは、サーバ 1（プライマリサーバ）とサーバ 2（バックアップサーバ）の 2 台のサーバで構成されます。

### シナリオ 1

サーバ 1 からサーバ 2 へのミラーが正常に完了し、その後サーバ1が動作不能に陥る。




**結果:** フェイルオーバが発生します。サーバ 2 がプライマリサーバの役割を担当し、サーバ 1 が再び動作可能になるまでデグレードモード（バックアップなし）で動作します。サーバ 1 が再び動作可能になると、SIOS DataKeeper がサーバ 2 からサーバ 1 への再同期を開始します。2.6.18 以前のカーネルでは、全体の再同期が実行されます。2.6.19 以降のカーネル、または Red Hat Enterprise Linux 5.4 の 2.6.18-164 以降のカーネル（または Red Hat 5.4 以降のサポートする派生カーネル）では、部分的な再同

期が実行されます。つまり、ソースとターゲットにあるビットマップファイルに記録された変更部分についてのみ同期が必要です。

**注記:** SIOS DataKeeper は、現在ミラーソースとして動作しているサーバに以下のフラグをセットします。

```
$LKROOT/subsys/scsi/resources/netraid/$TAG_last_owner
```

サーバ 1 がサーバ 2 にフェイルオーバすると、このフラグがサーバ 2 にセットされます。このため、サーバ 1 が動作を再開すると、SIOS DataKeeper はこの最終オーナフラグをサーバ 1 から削除します。その後、サーバ 2 からサーバ 1 にデータの再同期を開始します。

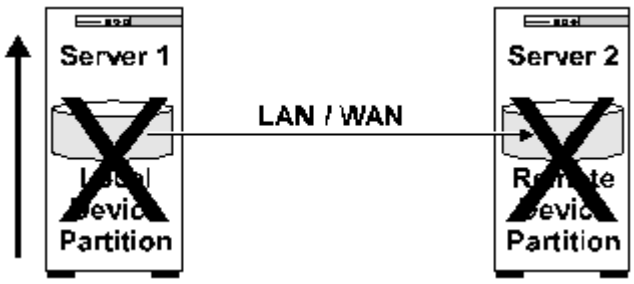
### シナリオ 2

シナリオ 1 で、サーバ 2（プライマリサーバである状態）が、サーバ 1（この時点ではバックアップサーバ）との再同期中に動作不能になる。

**結果:** 再同期プロセスが正常に完了しなかったので、データが破損している可能性があります。この結果、LifeKeeper は DataKeeper リソースをサーバ 1 にフェイルオーバしません。サーバ 2 が動作可能になった場合にのみ、LifeKeeper はサーバ 2 で DataKeeper リソースをサービス中（ISP）にします。

### シナリオ 3

サーバ 1（プライマリ）とサーバ 2（ターゲット）の両方が動作不能になる。サーバ 1（プライマリ）が最初に動作可能になる。




**結果:** サーバ 1 は、DataKeeper リソースを in service にしません。この理由は、停止してからオンラインに戻ったソースサーバは、ターゲットと通信できないからです。ソースサーバは以下のタグをセットします。

```
$LKROOT/subsys/scsi/resources/netraid/$TAG_data_corrupt
```

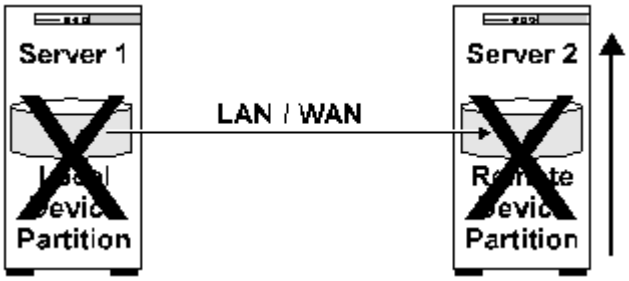
これは、正しくない方向へのデータ同期を防止する安全策です。この場合、サーバ 1 でミラーを強制的にオンラインにする必要があります。つまり、サーバ 1 の *data_corrupt* フラグを削除し、リソースを in service にします。ミラーを強制的にオンラインにする を参照してください。

**注記:** *$TAG_data_corrupt* フラグを削除する前に、サーバ 1 が最終のプライマリサーバであることを確認する必要があります。サーバ 1 が最終のプライマリサーバでない場合、データが破損する可能性があります

す。これは、 *last_owner* フラグの有無で確認できます。

#### シナリオ 4

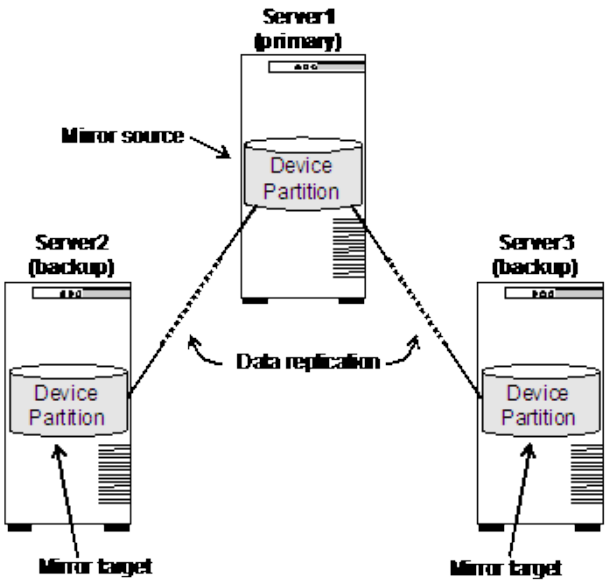
サーバ 1（プライマリ）とサーバ 2（ターゲット）の両方が動作不能になる。サーバ 2（ターゲット）が最初に動作可能になる。



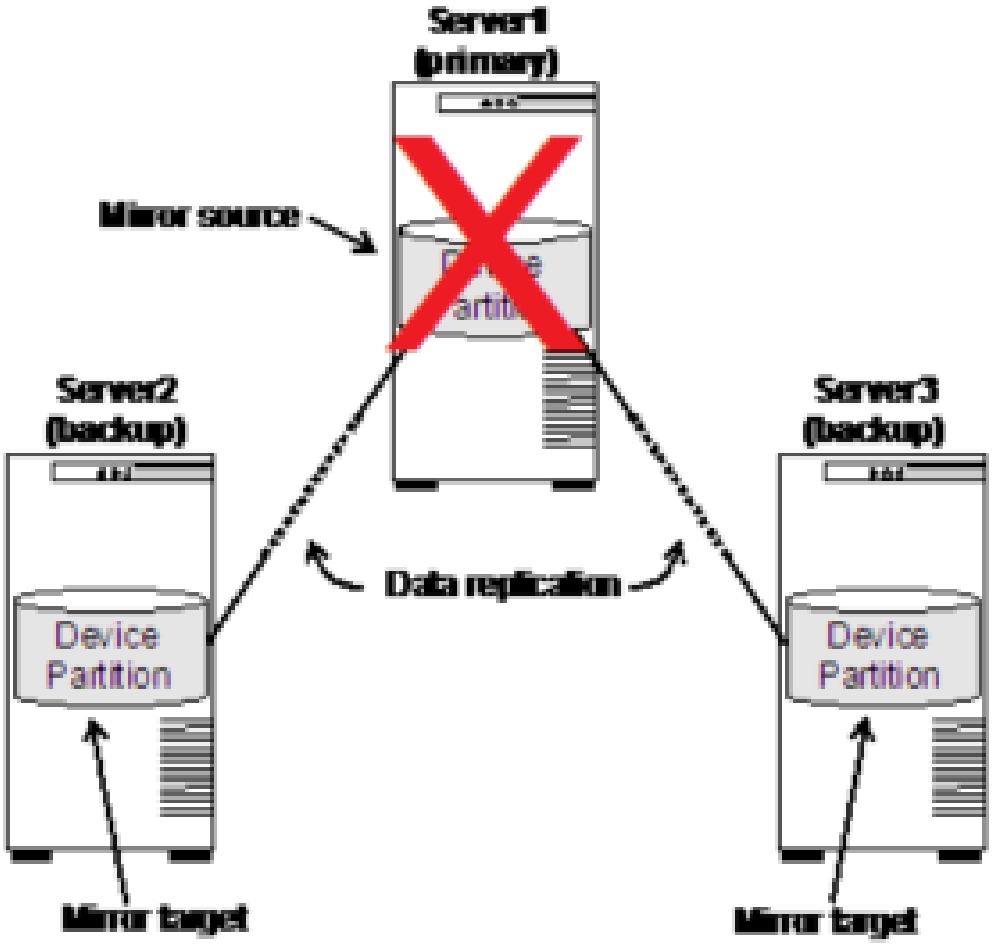

**結果:** LifeKeeper は、サーバ 2 の DataKeeper リソースを ISP にしません。サーバ 1 が動作可能になると、LifeKeeper はサーバ 1 の DataKeeper リソースを ISP にします。

## フェイルオーバのシナリオ – 3 ノード

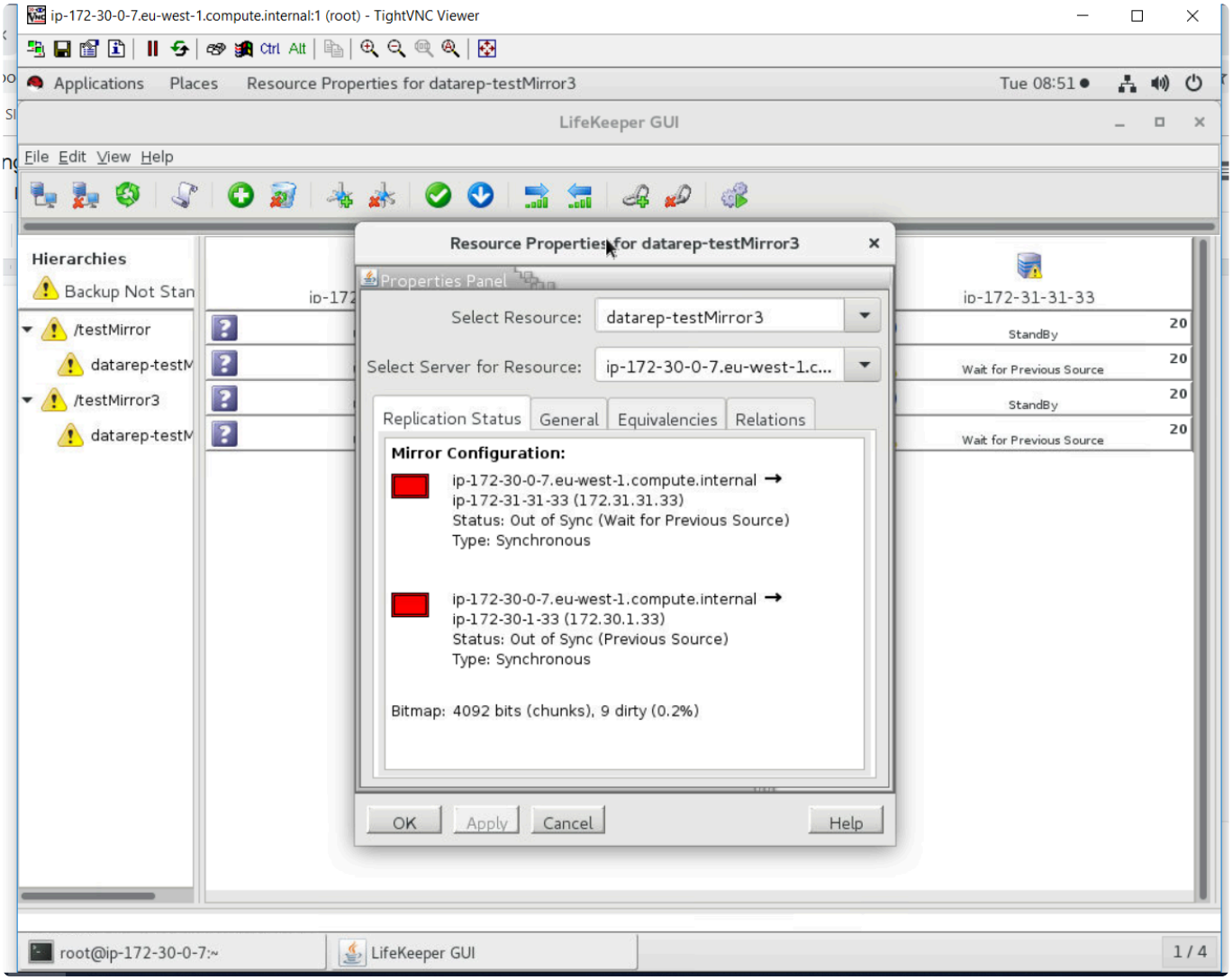
次の例は、SIOS DataKeeper を使用した場合にフェイルオーバ中に何が起こるかを示しています。この例では、LifeKeeper for Linuxクラスターは、サーバ1（プライマリサーバ）、サーバ2（バックアップサーバ）、およびサーバ3（バックアップサーバ）の3つのサーバで構成されています。

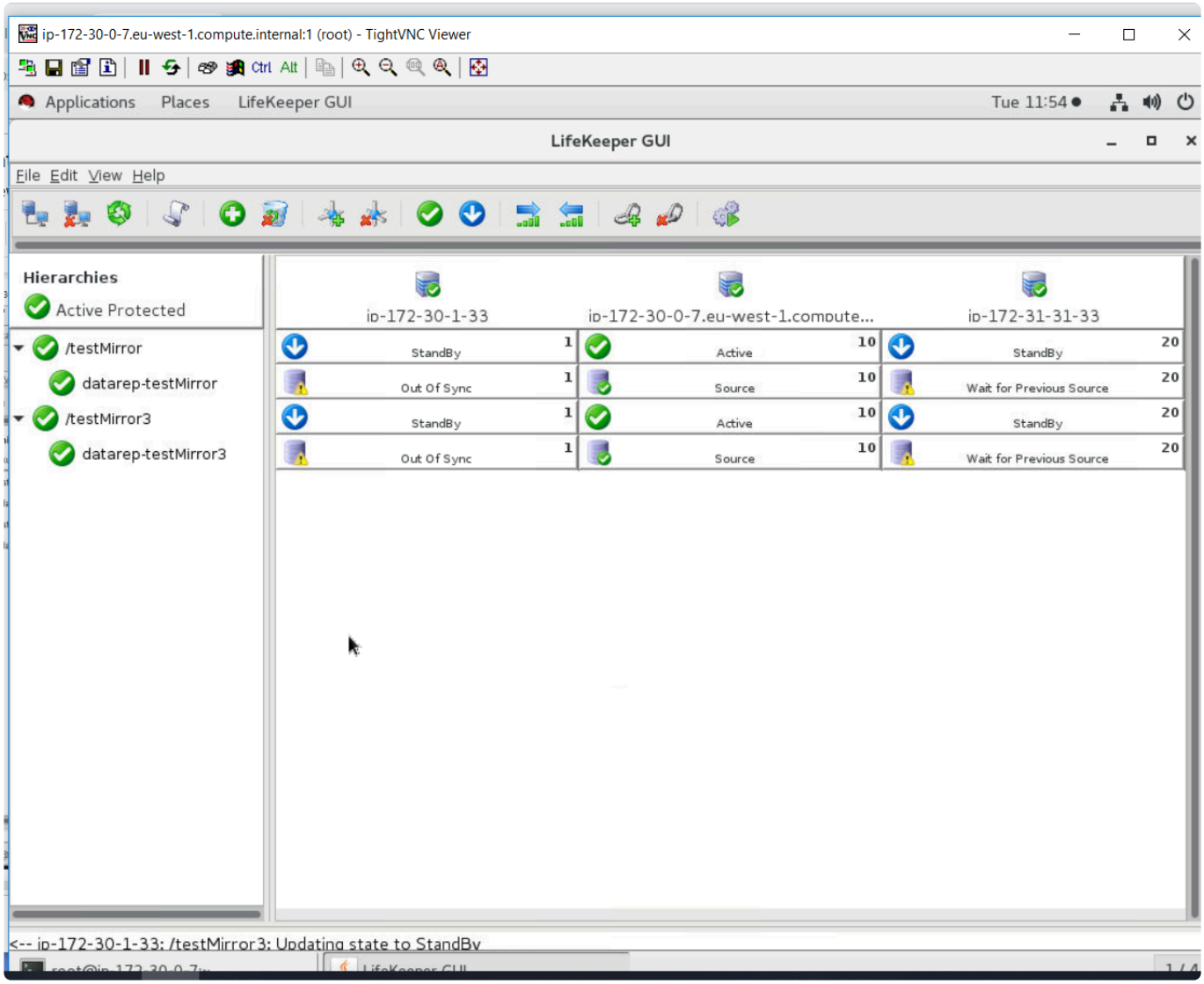



サーバ1（優先順位1）がサーバ2（優先順位10）およびサーバ3（優先順位20）への複製を正常に完了した後、サーバ1 は動作不能になります。

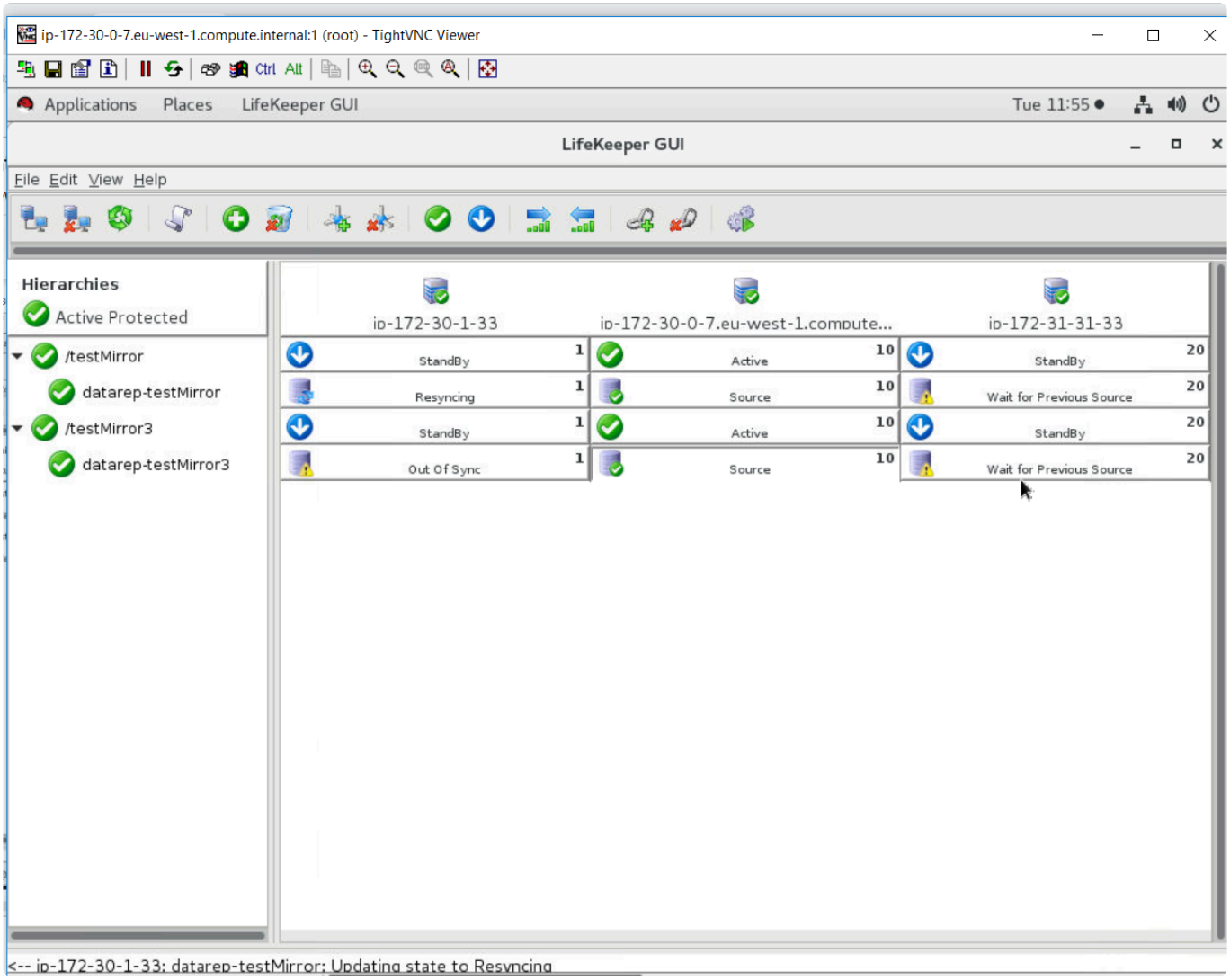



**結果：** フェイルオーバは、2番目に優先順位の高いサーバ2 で行われます。 サーバ2 は、プライマリサーバの役割を引き継ぎます。 v9.3.2 以前のバージョンでは、サーバ3 は完全再同期化とともにミラーに追加されます。 v9.3.2 では、サーバ2はサーバ1（前のサーバ）がクラスターに戻るのを待ってからサーバ3 への複製を再開します。これにより、サーバ1 のビットマップとサーバ2 のビットマップをマージして、サーバ1 とサーバ3 の両方に部分的に再同期することができます。 サーバ1 がクラスターに再接続するのを待っている間、LifeKeeper GUI はサーバ3 のステータスを「Out of Sync (Wait for Previous Source)」として表示します。 サーバが接続されていない間、サーバ1 のステータスは「Unknown」になり、最初に接続されたときに GUI のステータスに「Out of Sync」と表示されます。 ミラーのプロパティページは、それを「Out of Sync (Previous Source)」として識別します。


ip-172-30-0-7.eu-west-1.compute.internal:1 (root) - TightVNC Viewer
Applications Places Resource Properties for datarep-testMirror3 Tue 08:51
LifeKeeper GUI
File Edit View Help
Hierarchies
Backup Not Stan
/testMirror
datarep-testM
/testMirror3
datarep-testM
ip-172-31-31-33
StandBy 20
Wait for Previous Source 20
StandBy 20
Wait for Previous Source 20
Resource Properties for datarep-testMirror3
Properties Panel
Select Resource: datarep-testMirror3
Select Server for Resource: ip-172-30-0-7.eu-west-1.c...
Replication Status General Equivalencies Relations
Mirror Configuration:
ip-172-30-0-7.eu-west-1.compute.internal →
ip-172-31-31-33 (172.31.31.33)
Status: Out of Sync (Wait for Previous Source)
Type: Synchronous
ip-172-30-0-7.eu-west-1.compute.internal →
ip-172-30-1-33 (172.30.1.33)
Status: Out of Sync (Previous Source)
Type: Synchronous
Bitmap: 4092 bits (chunks), 9 dirty (0.2%)
OK Apply Cancel Help
root@ip-172-30-0-7:~
LifeKeeper GUI
1 / 4




サーバ1 が再接続すると、そのビットマップがマージされて再同期が開始され、その時点でステータスは「Resyncing」と表示されます。




再同期が完了するとステータスは「Target」に更新され、サーバ3 はそのステータスを「Resyncing」に変更して再同期を開始します。

注記： SIOS DataKeeper はミラーソースを追跡するために以下のフラグを設定します。

*$LKROOT/subsys/scsi/resources/netraid/$TAG_last_owner*

*$LKROOT/subsys/scsi/resources/netraid/$TAG_source*

*$TAG_last_owner* フラグは、現在ミラーソースとして機能しているシステム上にあり、 *$TAG_source* フラグには、ローカルノードがミラーの一部であった最後の時点でソースだったシステムの名前が含まれています。

サーバ1 がサーバ2 にフェイルオーバすると、サーバ2 に *$TAG_last_owner* フラグが設定されます。サーバ2 の *$TAG_source* フラグは、サーバ1 を前のソースとして識別します（これには、サーバ1 とサーバ3 への部分的な再同期を実行するために必要なビットマップが含まれています）。サーバ1 が復旧すると、SIOS DataKeeper はサーバ1から *$TAG_last_owner* フラグを削除します。次にサーバ2 はサーバ1 からのビットマップをマージし、サーバ2 からサーバ1 へのデータの再同期を開始します。サーバ1 が復旧すると、SIOS DataKeeper はサーバ1 から *$TAG_last_owner* フラグを削除します。次にサーバ2 はサーバ1 のビットマップをマージし、サーバ2 からサーバ1 へのデータの再同期を開始します。サーバ1 への再同期が完了すると、サーバ1 の *$TAG_source* フラグがサーバ2 の名前で更新されます。サーバ1 が同期された

後、サーバ2 はサーバ3 へ同じ再同期を実行します。サーバ3 の *$TAG_source* フラグは、サーバ2 の名前で更新されます。

# 5.5.3. インストールと設定

## SIOS DataKeeper for Linux のインストールと設定

ハードウェア / ソフトウェア要件

## DataKeeper リソースを設定する前に

以下のトピックには、DataKeeper リソースの作成と管理を行う前に考慮が必要な情報があります。また、3 種類の DataKeeper リソースについても説明しています。LifeKeeper Core のリソース階層を設定する手順については、LifeKeeper の設定 セクションを参照してください。

---

ハードウェアとソフトウェアの要件

全般的な設定

ネットワーク設定

データレプリケーションパスの変更

ネットワーク帯域幅の要件の特定

Linux システム（物理または仮想）での変化率の測定

WAN の設定

SIOS DataKeeper for Linux のリソースタイプ

DataKeeper 構成での I/O フェンシング

リソースの設定作業

DataKeeper リソース階層の作成

リソース階層の拡張

リソース階層の拡張解除

リソース階層の削除

DataKeeper リソースを Out of Service にする

DataKeeper リソースを In Service にする

リソース階層のテスト

# 5.5.3.1. ハードウェアとソフトウェアの要件

SIOS DataKeeper を インストールするには 、LifeKeeper の構成が次の要件を満たしている必要があります。

## ハードウェアの要件

- サーバ - LifeKeeper for Linux をサポートする 2 台以上のサーバ。
- IP ネットワークインターフェースカード - 各サーバにネットワークインターフェースカードが 1 つ以上必要です。ただし、LifeKeeper クラスタには 2 つのコミュニケーションパスが必要です。独立した 2 つのサブネットを使用する 2 つの分離した LAN ベースのコミュニケーションパスが推奨され、これらの 1 つ以上をプライベートネットワークとして構成する必要があります。ただし、TCP と TTY を組み合わせて使用することもできます。

> 注記: ソフトウェアミラーリングの特性により、サーバ間のネットワークトラフィックが多くなる可能性があります。このため、SIOS DataKeeper のデバイス用に個別のプライベートネットワークを実装することが推奨されます。この実装には、各サーバに追加のネットワークインターフェースカードが必要になることがあります。

- ディスクまたはパーティション - ソースとターゲットのディスクまたはパーティションとして動作する、プライマリサーバとバックアップサーバのディスクまたはパーティション。ターゲットのディスクまたはパーティションは、ソースのディスクまたはパーティション以上のサイズである必要があります。

> 注記: SIOS Data Replication 7.1.1 のリリースから、パーティションが作成されていないディスク全体（ */dev/sdd* ）の複製が可能になりました。旧バージョンの SIOS Data Replication では、ディスクを複製するには、パーティションを作成する必要がありました（ */dev/sdd1* のような 1 つの大きいパーティションの場合でも）。SIOS Data Replication 7.1.1 からこの制限が取り除かれました。

> 注記: SPS for Linux v9.5.0 のリリースに伴い、すべてのディスクは一意に識別できる必要があります。DataKeeper は、デバイス名（ */dev/sdd* ）を使用してデバイスを識別することを許可していましたが、状況によっては、デバイス名が変更され、データが破損する可能性があります。GPT パーティションテーブルを使用すると、一意の識別子を提供できます。

## ソフトウェアの要件

- オペレーティングシステム – SIOS DataKeeper は、Linux カーネル 2.6 をベースにする主要な Linux のディストリビューションと共に使用できます。サポートするディストリビューションのリストについては、SPS for Linuxリリースノート を参照してください。非同期ミラーリングとインテントログは、2.6.16 以降の Linux カーネルを使用するディストリビューションでのみサポートされます。複数のターゲットのサポート（複数のミラーターゲットのサポート）には、2.6.7 以降の Linux

カーネルが必要です。

- **LifeKeeper Installation** スクリプト - 多くの場合、以下のパッケージをインストールする必要があります (特定の SIOS DataKeeper の要件については、SPS for Linux リリースノート の「製品要件」セクションを参照してください)。

  **HADR-generic-2.6**

  SIOS DataKeeper を インストールする前 に、LifeKeeper クラスタの各サーバにこのパッケージをインストールする必要があります。HADR パッケージはSPS のインストールイメージファイル内にあり、Installation の **setup** スクリプトにより自動的に適切なパッケージがインストールされます。

- **LifeKeeper** ソフトウェア - 各サーバに同じバージョンの LifeKeeper Core をインストールする必要があります。また、使用を計画している同じバージョンの Recovery Kit も各サーバにインストールする必要があります。特定の SPS の要件については、SPS for Linuxリリースノート を参照してください。

- **SIOS DataKeeper** ソフトウェア – SPS クラスタの各サーバには SIOS DataKeeper ソフトウェアが必要です。SIOS DataKeeper のインストールとアンインストールの手順については、SPS for Linux インストールガイド を参照してください。

# 5.5.3.2. 全般的な設定

- ターゲットのディスクまたはパーティションのサイズ（バックアップサーバ上）は、ソースのディスクまたはパーティションのサイズ（プライマリサーバ上）以上である必要があります。

- DataKeeper リソースを作成して拡張すると、同期プロセスにより ターゲットのディスクまたはパーティション に存在するデータが削除され、ソースのパーティションにあるデータに置き換えられます。

# 5.5.3.3. ネットワーク設定

- 各ペアのサーバ間でデータのレプリケーション用に選択するパスは、あらかじめそれらのサーバ間の LifeKeeper コミュニケーションパスとしても設定されている必要があります。ネットワークパスを変更する方法については、データレプリケーションパスの変更を参照してください。

- DataKeeperがデータリプリケーションに使用するネットワークインタフェースへIPリカバリキットで仮想IPアドレスを付加する構成は行わないでください。
  IPリカバリキットがネットワークインタフェースを操作する時に一時的に回線断が発生するため、予期しないタイミングでデータリプリケーションが停止して不要な再同期が発生する可能性があります。

- このリリースの SIOS DataKeeper は、DataKeeper リソースの自動スイッチバックをサポートしていません。さらに、自動スイッチバックの制限は、DataKeeper リソースの上に存在する他の LifeKeeper リソースにも適用されます。

- Fusion-io を使用する場合のネットワーク設定情報については、Fusion-io を使用するクラスタ化 の「ネットワーク」セクションを参照してください。

# 5.5.3.4. データレプリケーションパスの変更

LK 7.1 から、lk_chg_value を使用して、ミラーのレプリケーションパスのIPアドレスを変更できるようになりました。例えば、ミラーのレプリケーションパスを IP アドレスの 192.168.0.1 から 192.168.1.1 に変更するには、以下の操作を行ってください。

```
# lkstop (lk_chg_value は、LifeKeeper の動作中は実行できません)。

# lk_chg_value -o 192.168.0.1 -n 192.168.1.1

# lkstart
```

この IP アドレスを使用するミラーに含まれるすべてのサーバで、これらのコマンドを実行してください。

注記: このコマンドは、該当アドレスを使用するコミュニケーションパスも変更します。

# 5.5.3.5. ネットワーク帯域幅の要件の特定

SIOS DataKeeper をインストールする前に、現在の構成の複製に仮想マシンを使用するか、物理的な Linux サーバを使用するかによりネットワーク帯域幅の要件を特定する必要があります。仮想マシン（VM）を使用する場合は、Linux システム（物理または仮想）の更新頻度の測定 方法を使用して、複製を計画している仮想マシンの変化率を測定してください。この値は、仮想マシンの複製に必要となるネットワーク帯域幅を表します。

ネットワーク帯域幅の要件を特定した後、ネットワークが最大のパフォーマンスを発揮するように構成してください。ネットワーク帯域幅の要件が、現在使用できるネットワーク能力を超えている場合は、以下のオプションを 1 つ以上検討しなければならない可能性があります。

- SIOS DataKeeper（または可能な場合はネットワークハードウェア）の圧縮を有効にする。
- ネットワーク能力を増強する。
- 複製するデータ量を低減する。
- 一時データおよびスワップファイル用に、複製しないローカルのストレージリポジトリを作成する。
- 毎日、ピーク時以外に複製を手動でスケジュールする。

# 5.5.3.5.1. Linux システム（物理または仮想）での変化率の測定

DataKeeper for Linux は、使用できるネットワーク内でデータを複製できます。マルチサイト、すなわち広域ネットワーク（WAN）構成では、「ソースパーティションが 1 日中更新されるときに、パーティションを正常に複製してミラーをミラーリング状態に維持するために十分な帯域幅があるか」という質問に対して特別な検討が必要です。

ミラーがミラーリング状態でない場合にはパーティションのスイッチオーバは許可されないので、ミラーをミラーリング状態に維持することが重要です。

SIOS DataKeeper はデータを非同期キューに追加することにより、短期間に急増した書き込み動作を処理します。ただし、長期間にわたって複製されるすべてのボリュームのディスク書き込み動作の合計が、平均して DataKeeper とネットワークが送信できる変化量を下回ることを確認してください。

ネットワーク能力が不十分なためにディスクの変化率に対処できず、非同期キューがいっぱいになった場合、ミラーは同期動作に戻ります。これによりソースサーバのパフォーマンスに悪影響を及ぼすことがあります。

## 基本変化率の測定

以下のコマンドを使用して、ミラーリングするファイルまたはパーティションを特定してください。例えば /dev/sda3 を使用して、1 日に書き込まれたデータ量を測定します。

```
MB_START=`awk '/sda3 / { print $10 / 2 / 1024 }' /proc/diskstats`
```

1 日後

```
MB_END=`awk '/sda3 / { print $10 / 2 / 1024 }' /proc/diskstats`
```

1 日の変化率 (単位: MB) は `MB_END - MB_START` で得られます。

SIOS DataKeeper が 1 日にミラーリングできるおよその量は以下のとおりです。

T1 (1.5Mbps) – 14,000 MB/日(14 GB)

T3（1.5 Mbps） – 410,000 MB/日 (410 GB)

ギガビット（1 Gbps） - 5,000,000 MB/日（5 TB）

## 詳細変化率の測定

変化率を収集する最良の方法は、一定期間（例: 1 日）ディスクの書き込み動作をログに記録して、ディスクの書き込みのピーク期間を特定することです。

ディスクの書き込み動作を追跡するには、システムのタイムスタンプをログに記録して */proc/diskstats* のダンプを行う cron ジョブを作成してください。例えば、2 分間隔でディスクの統計値を収集するには、*/etc/crontab* に以下のリンクを追加します。:

```
*/2 * * * * root ( date ; cat /proc/diskstats ) >> /path_to/
filename.txt
```

1 日、1 週間などの期間が経過した後、cron ジョブを無効にし、得られたデータファイルを安全な場所に保存します。

# 収集した詳細変化率データの解析

roc-calc-diskstats ユーティリティは、前述の手順で収集したデータを解析します。このユーティリティは、長期間ログに記録された出力を持つ */proc/diskstats* 出力ファイルから、データセットに含まれるディスクの変化率を計算します。

roc-calc-diskstats をダウンロードするには[ここをクリック]

**使用法:**

```
# ./roc-calc-diskstats <interval> <start_time> <diskstats-data-file> [dev-
list]
```

**使用例（概要のみ）:**

```
# ./roc-calc-diskstats 2m “Jul 22 16:04:01” /root/diskstats.txt
sdb1,sdb2,sdc1 > results.txt
```

この例は、概要（およびディスク別のピーク I/O 情報）を *results.txt* にダンプします。

**使用例（概要とグラフデータ）:**

```
# export OUTPUT_CSV=1

# ./roc-calc-diskstats 2m “Jul 22 16:04:01” /root/diskstats.txt
sdb1,sdb2,sdc1 2> results.csv > results.txt
```

この例は、グラフデータを *results.csv* に、概要（およびディスク別のピーク I/O 情報）を *results.txt* にダンプします。

**結果の例（results.txt）**

```
Sample start time: Tue Jul 12 23:44:01 2011

Sample end time: Wed Jul 13 23:58:01 2011

Sample interval: 120s #Samples: 727 Sample length: 87240s
```

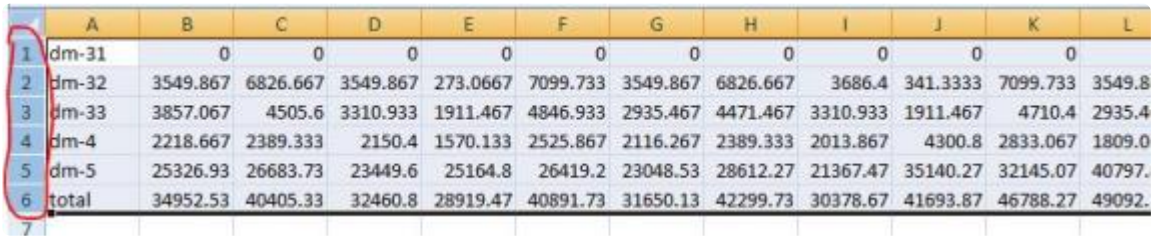
```
(Raw times from file: Tue Jul 12 23:44:01 EST 2011, Wed Jul 13 23:58:01
EST 2011)

Rate of change for devices dm-31, dm-32, dm-33, dm-4, dm-5, total

dm-31 peak:0.0 B/s (0.0 b/s) (@ Tue Jul 12 23:44:01 2011) average:0.0 B/s
(0.0 b/s)

dm-32 peak:398.7 KB/s (3.1 Mb/s) (@ Wed Jul 13 19:28:01 2011) average:19.5
KB/s (156.2 Kb/s)

dm-33 peak:814.9 KB/s (6.4 Mb/s) (@ Wed Jul 13 23:58:01 2011) average:11.6
KB/s (92.9 Kb/s)

dm-4 peak:185.6 KB/s (1.4 Mb/s) (@ Wed Jul 13 15:18:01 2011) average:25.7
KB/s (205.3 Kb/s)

dm-5 peak:2.7 MB/s (21.8 Mb/s) (@ Wed Jul 13 10:18:01 2011) average:293.0
KB/s (2.3 Mb/s)

total peak:2.8 MB/s (22.5 Mb/s) (@ Wed Jul 13 10:18:01 2011) average:349.8
KB/s (2.7 Mb/s)
```

## 詳細変化率データのグラフ作成

お客様に固有の経時的な帯域幅のニーズを分かりやすくするために、テンプレートスプレッドシート diskstats-template.xlsx が用意されています。このスプレッドシートにはサンプルデータがあり、`roc-calc-diskstats`で収集したデータで上書きできます。

diskstats-template.xslx をダウンロードするには[ここをクリック]

1. results.csv を開き、total 列を含めて **すべての行** を選択してください。



| | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | dm-31 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 2 | dm-32 | 3549.867 | 6826.667 | 3549.867 | 273.0667 | 7099.733 | 3549.867 | 6826.667 | 3686.4 | 341.3333 | 7099.733 | 3549.8 |
| 3 | dm-33 | 3857.067 | 4505.6 | 3310.933 | 1911.467 | 4846.933 | 2935.467 | 4471.467 | 3310.933 | 1911.467 | 4710.4 | 2935.4 |
| 4 | dm-4 | 2218.667 | 2389.333 | 2150.4 | 1570.133 | 2525.867 | 2116.267 | 2389.333 | 2013.867 | 4300.8 | 2833.067 | 1809.0 |
| 5 | dm-5 | 25326.93 | 26683.73 | 23449.6 | 25164.8 | 26419.2 | 23048.53 | 28612.27 | 21367.47 | 35140.27 | 32145.07 | 40797. |
| 6 | total | 34952.53 | 40405.33 | 32460.8 | 28919.47 | 40891.73 | 31650.13 | 42299.73 | 30378.67 | 41693.87 | 46788.27 | 49092. |
| 7 | | | | | | | | | | | | |

2. **diskstats-template.xlsx** を開き、 **diskstats.csv** ワークシートを選択してください。

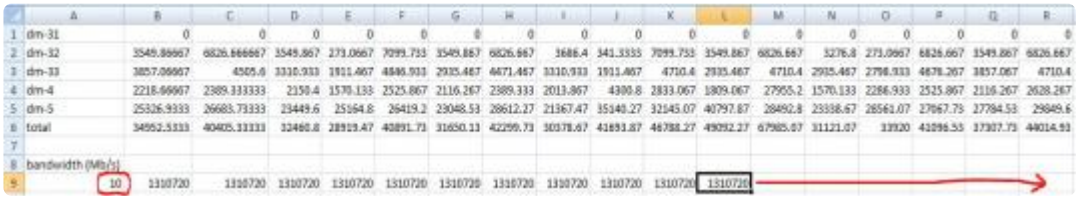
3. セル 1-A を右クリックし、 **[Insert Copied Cells]** を選択してください。

4. レプリケーション用に割り当てた帯域幅の量を反映するように、ワークシートの左下にあるセルの

**bandwidth** 値を調整してください。

単位: メガビット/秒（Mb/sec）

**注記:** その右側にあるセルの値は、収集した生データに合わせて自動的にバイト / 秒単位に変換されます。

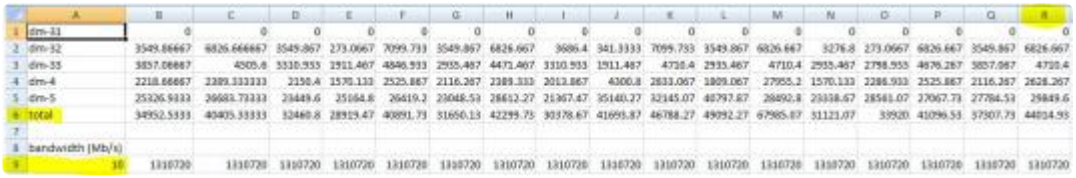


5. 以下の行/列番号を記録してください。

   a. Total(下のスクリーンショットでは行 6)

   b. Bandwidth(下のスクリーンショットでは行 9)

   c. 最終データポイント(下のスクリーンショットでは列 R)



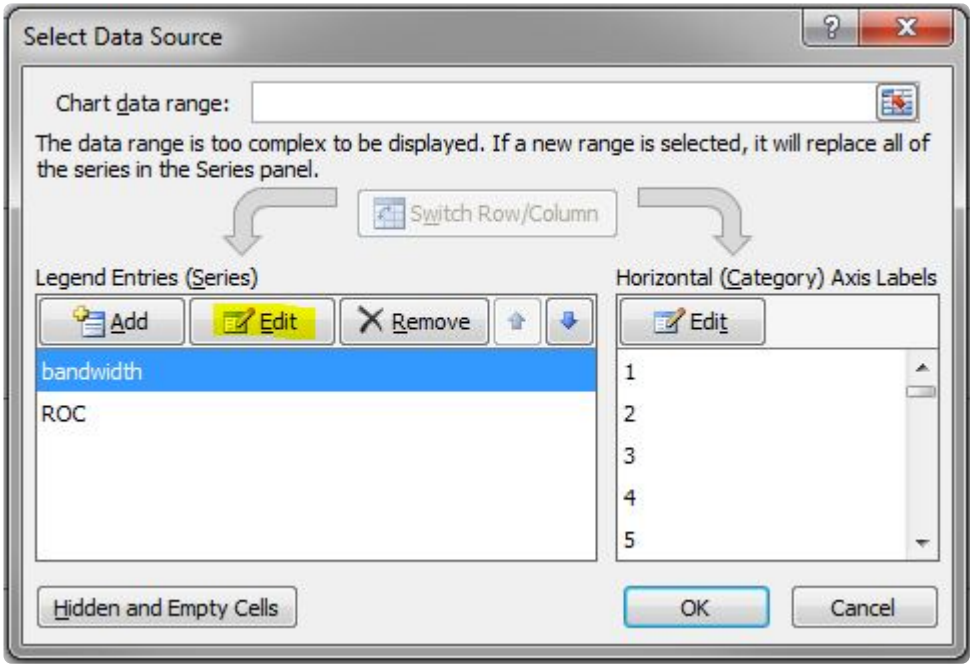
6. **bandwidth vs ROC** ワークシートを選択してください。

7. グラフを右クリックし、 **[Select Data...]** を選択してください。

   a. **Bandwidth** 系列 を調整してください。

      i. 左の **[Series]** リストから **bandwidth** を選択してください。

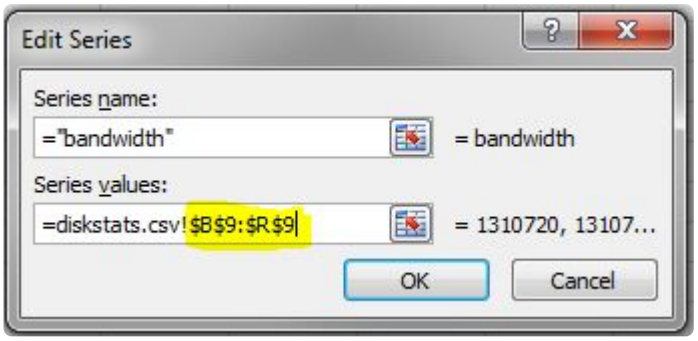
      ii. **[Edit]** をクリックしてください。




iii. 以下の構文を使用して、 **[Series Values]** フィールドを調整してください。

"=diskstats.csv!$B$<row>:$<final_column>$<row>"

例: "=diskstats.csv!$B$9:$R:$9"

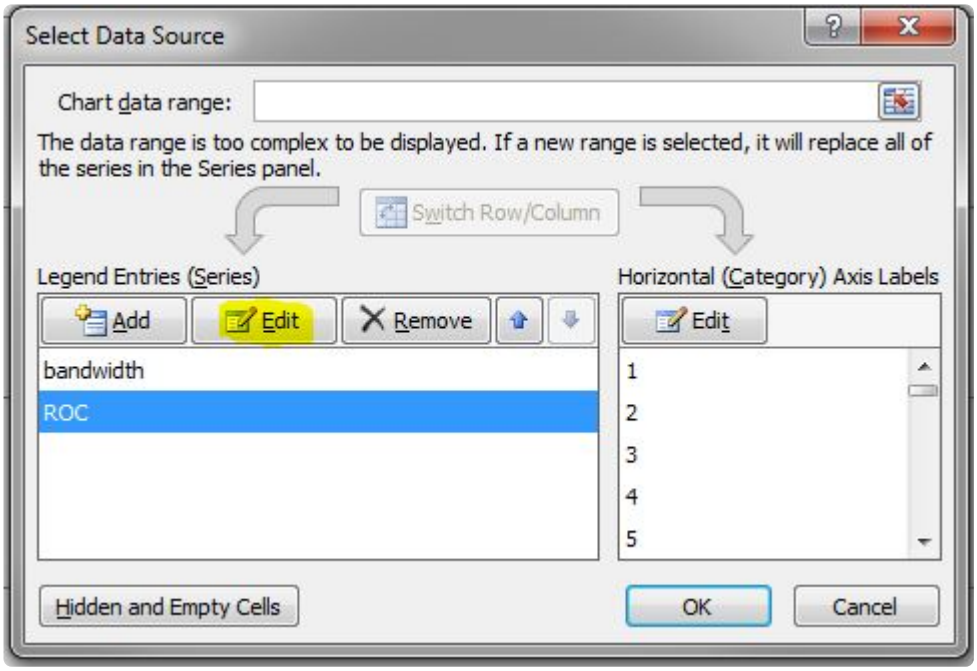



iv. **[OK]** をクリックしてください。

b. **ROC 系列** を調整してください。

i. 左の **[Series]** リストから **ROC** を選択してください。

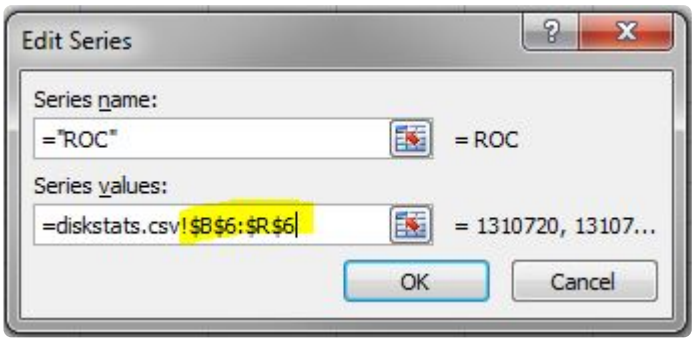
ii. **[Edit]** をクリックしてください。




iii. 以下の構文を使用して、 **[Series Values]** フィールドを調整してください。

```
"=diskstats.csv!$B$<row>:$<final_column>$<row>"
```

例:"=diskstats.csv!$B$6:$R:$6"




iv. **[OK]** をクリックしてください。

c. **[OK]** をクリックしてウィザードを終了してください。

8. Bandwidth vs ROC のグラフが更新されます。結果を解析して、データのレプリケーションをサポートするために十分な帯域幅があるかどうかを判断してください。

# roc-calc-diskstats

```
#!/usr/bin/perl
# Copyright (c) 2011, SIOS Technology, Corp.
# Author: Paul Clements
use strict;
sub msg {
        printf STDERR _;
}
sub dbg {
        return if (! $ENV{'ROC_DEBUG'});
        msg _;
}
$0 =~ s@^.*/@@; # basename
sub usage {
        msg "Usage: $0 <interval> <start-time> <iostat-data-file> [dev-lis
t]\n";
        msg "\n";
        msg "This utility takes a /proc/diskstats output file that contains\
n";
        msg "output, logged over time, and calculates the rate of change of\
n";
        msg "the disks in the dataset\n";
        msg "OUTPUT_CSV=1 set in env. dumps the full stats to a CSV file on ST
DERR\n";
        msg "\n";
        msg "Example: $0 1hour \"jun 23 12pm\" steeleye-iostat.txt sdg,sdh\n";
        msg "\n";
        msg "interval - interval between samples\n";
        msg "start time - the time when the sampling starts\n";
        msg "iostat-data-file - collect this with a cron job like:\n";
        msg "\t0 * * * * (date ; cat /proc/diskstats) >> /root/diskstats.txt\
n";
        msg "dev-list - list of disks you want ROC for (leave blank for al
l)\n";
        exit 1;
}
usage if (ARGV < 3);
my $interval = TimeHuman($ARGV[0]);
my $starttime = epoch($ARGV[1]);
my $file = $ARGV[2];
my $blksize = 512; # /proc/diskstats is in sectors
my %devs = map { $_ => 1 } split /,/, $ARGV[3];
my %stat;
my $firsttime;
my $lasttime;
```

```
# datestamp divides output
my %days = ( 'Sun' => 1, 'Mon' => 1, 'Tue' => 1, 'Wed' => 1,
                'Thu' => 1, 'Fri' => 1, 'Sat' => 1);
my %fields = ( 'major'  => 0,
                'minor' => 1,
                'dev' => 2,
                'reads' => 3,
                'reads_merged' => 4,
                'sectors_read' => 5,
                'ms_time_reading' => 6,
                'writes' => 7,
                'writes_merged' => 8,
                'sectors_written' => 9,
                'ms_time_writing' => 10,
                'ios_pending' => 11,
                'ms_time_total' => 12,
                'weighted_ms_time_total' => 13 );
my $devfield = $fields{'dev'};
my $calcfield = $ENV{'ROC_CALC_FIELD'} || $fields{'sectors_written'};
dbg "using field $calcfield\n";
open(FD, "$file") or die "Cannot open $file: $!\n";
foreach (<FD>) {
        chomp;
        _ = split;
        if (exists($days{$_[0]})) { # skip datestamp divider
                if ($firsttime eq '') {
                        $firsttime = join ' ', _[0..5];
                }
                $lasttime = join ' ', _[0..5];
                next;
        }
        next if ($_[0] !~ /[0-9]/); # ignore
        if (!%devs || exists $devs{$_[$devfield]}) {
                push {$stat{$_[$devfield]}}, $_[$calcfield];
        }
}
{$stat{'total'}} = totals(\%stat);
printf "Sample start time: %s\n", scalar(localtime($starttime));
printf "Sample end time: %s\n", scalar(localtime($starttime + (({$stat{'tota
l'}} - 1) * $interval)));
printf "Sample interval: %ss #Samples: %s Sample length: %ss\n", $interval,
({$stat{'total'}} - 1), ({$stat{'total'}} - 1) * $interval;
print "(Raw times from file: $firsttime, $lasttime)\n";
print "Rate of change for devices " . (join ', ', sort keys %stat) . "\n";
foreach (sort keys %stat) {
        my vals = {$stat{$_}};
        my ($max, $maxindex, $roc) = roc($_, $blksize, $interval, vals);
```

```
        printf "$_ peak:%sB/s (%sb/s) ( %s) average:%sB/s (%sb/s)\n", HumanSiz
e($max), HumanSize($max * 8), scalar localtime($starttime + ($maxindex * $inte
rval)), HumanSize($roc), HumanSize($roc * 8);
}
# functions
sub roc {
        my $dev = shift;
        my $blksize = shift;
        my $interval = shift;
        my ($max, $maxindex, $i, $first, $last, $total);
        my $prev = -1;
        my $first = $_[0];
        if ($ENV{'OUTPUT_CSV'}) { print STDERR "$dev," }
        foreach (_) {
                if ($prev != -1) {
                        if ($_ < $prev) {
                                dbg "wrap detected at $i ($_ < $prev)\n";
                                $prev = 0;
                        }
                        my $this = ($_ - $prev) * $blksize / $interval;
                        if ($this > $max) {
                                $max = $this;
                                $maxindex = $i;
                        }
                        if ($ENV{'OUTPUT_CSV'}) { print STDERR "$this," }
                }
                $prev = $_; # store current val for next time around
                $last = $_;
                $i++;
        }
        if ($ENV{'OUTPUT_CSV'}) { print STDERR "\n" }
        return ($max, $maxindex, ($last - $first) * $blksize / ($interval *
($i - 1)));
}
sub totals { # params: stat_hash
        my $stat = shift;
        my totalvals;
        foreach (keys %$stat) {
                next if (!defined($stat{$_}));
                my vals = {$stat{$_}};
                my $i;
                foreach (vals) {
                        $totalvals[$i++] += $_;
                }
        }
        return totalvals;
}
```

```
# converts to KB, MB, etc. and outputs size in readable form
sub HumanSize { # params: bytes/bits
        my $bytes = shift;
        my suffixes = ( '', 'K', 'M', 'G', 'T', 'P' );
        my $i = 0;
        while ($bytes / 1024.0 >= 1) {
                $bytes /= 1024.0;
                $i++;
        }
        return sprintf("%.1f %s", $bytes, $suffixes[$i]);
}
# convert human-readable time interval to number of seconds
sub TimeHuman { # params: human_time
        my $time = shift;
        my %suffixes = ('s' => 1, 'm' => 60, 'h' => 60 * 60, 'd' => 60 * 60 *
24);
        $time =~ /^([0-9]*)(.*?)$/;
        $time = $1;
        my $suffix = (split //, $2)[0]; # first letter from suffix
        if (exists $suffixes{$suffix}) {
                $time *= $suffixes{$suffix};
        }
        return $time;
}
sub epoch { # params: date
        my $date = shift;
        my $seconds = `date +'%s' --date "$date" 2>&1`;
        if ($? != 0) {
                die "Failed to recognize time stamp: $date\n";
        }
        return $seconds;
}
```

# 5.5.3.6. WAN の設定

WAN ネットワークの特性により、SIOS DataKeeper を WAN 環境で使用するには特殊な設定が必要です。以下のヒントに従うことが推奨されます。

- 意図しないフェイルオーバを防止するために manual failover confirmation を有効にする必要があります。多くの WAN は LAN よりも信頼性が低い上、WAN のミラー設定はコミュニケーションパスが 1 本だけになるため、通常得策といえます。このオプションを有効にすると、LifeKeeper のフェイルオーバはユーザが `lk_confirmso` コマンドを使用してフェイルオーバを確認した場合にのみ実行されます。詳細については、 `lk_confirmso` のマニュアルページを参照してください。

- WAN の待機時間とスループットに基づいて、 `LKDR_ASYNC_LIMIT` の適切な値を決定してください。 `LKDR_ASYNC_LIMIT` パラメータ ( */etc/default/LifeKeeper* に設定) が、SIOS DataKeeper が許可する未処理の書き込み動作 (ミラーあたり) の数を指定します。このパラメータのデフォルト値は 4096 ですが、大きい値を指定するとミラーの書き込みパフォーマンスを向上できます。この値を増加することの短所は、どの時点でもステータスとセカンダリとの間で非同期になっているデータ量が多くなることです。 `LKDR_ASYNC_LIMIT` パラメータについての詳細は **SIOS DataKeeper for Linux** によるミラーリング の非同期ミラーリング を参照してください。.

- 低速の WAN リンク経由で大量のデータをミラーリングする場合、最初のデータ全体の再同期を行わずに、ソースのディスクまたはパーティションのコピーをリモート（ディザスタリカバリ）サイトに輸送するのが望ましいことがあります。最初の再同期を回避するには、全体の再同期の回避 の手順に従ってください。

> **重要 :** LifeKeeper の GUI にある [New Replicated Filesystem] オプションを使用して階層を作成した場合、この手順は不要です。[New Replicated Filesystem] オプションは、最初の全体の再同期を回避するように最適化されています。

- WAN リンクに 15 秒以上のダウンタイム期間が定期的に発生する場合、LifeKeeper のハートビートパラメータを調整することが推奨されます。詳細については、LifeKeeper のハートビートの調整 を参照してください。

# 5.5.3.7. SIOS DataKeeper for Linux のリソースタイプ

DataKeeper リソース階層を作成するときに、リソースタイプを選択するように LifeKeeper から要求されます。DataKeeper リソースには、いくつかのタイプがあります。お使いの環境に最適なタイプを選択するときに、以下の情報が役立ちます。

## Replicate New File System

Replicate New File System を選択すると、NetRAID デバイスが作成/拡張され、NetRAID デバイスに指定のマウントポイントがマウントされます。また、LifeKeeper がサポートするファイルシステムと NetRAID デバイスの両方が、LifeKeeper で保護されます。ローカルのディスクまたはパーティションがフォーマットされます。

このリソースが2番目のノードに拡張されると、完全再同期は行われず、ファイルシステムのメタデータデータのみ再同期が行われます。ただし、3番目以上のノードに拡張されると、そのノードに対して完全再同期が行われます。完全再同期を回避するには、全同期の回避 に記載されている手順に従ってください。

注意：データがすべて削除されます。

## Replicate Existing File System

Replicate Existing File System を選択すると、現在マウントされているディスクまたはパーティションが使用され、ディスクまたはパーティションのデータが削除されることなく NetRAID デバイスが作成されます。SIOS DataKeeper はローカルのディスクまたはパーティションをアンマウントし、ローカルのディスクまたはパーティションを使用して NetRAID デバイスを作成します。そして、NetRAID デバイスにマウントポイントをマウントします。次に、NetRAID デバイスと LifeKeeper がサポートするファイルシステムの両方を LifeKeeper で保護します。

**重要:** SIOS Protection Suite for Linux のマルチサイトクラスタ階層を作成する場合、作成プロセス中にアプリケーションが停止します。階層の作成と拡張が完了した後、アプリケーションを再起動する必要があります。

## DataKeeper Resource

DataKeeper リソース を選択すると、NetRAID デバイスが作成/拡張され、ファイルシステムは含めずに LifeKeeper で保護されます。RAW I/O デバイスを使用できるデータベースを使用している場合は、この複製タイプを選択できます。

ユーザがデータアクセスを続行できるように、SIOS DataKeeper は、現在マウントされている NetRAID デバイスのアンマウントと削除は実行しません。ユーザは、手動スイッチオーバの前に NetRAID デバイスを手動でアンマウントし、手動スイッチオーバの後に他のサーバにマウントする必要があります。

**注記:** DataKeeper リソースの作成後に、手動マウントしたファイルシステムを LifeKeeper で保護する場合は、以下の操作を行います。

1. LifeKeeper がサポートするファイルシステムで、NetRAID デバイスをフォーマットしてください。

2. NetRAID デバイスをマウントしてください。

3. NetRAID デバイスを使用して、共有ストレージのディスクまたはパーティションにあるかのようにファイルシステムの階層を作成し、拡張してください。

これで、LifeKeeper のファイルシステムリカバリキットが、フェイルオーバ時のファイルシステムのマウント / アンマウントを実行します。

# 5.5.3.8. DataKeeper 構成での I/O フェンシング

DataKeeper 構成では、原理上ストレージのリザベーションを使用した I/O フェンシングを使用することが出来ないため、単体ではスプリットブレイン状態を回避することが出来ません。
そのため、以下のような方法で DataKeeper リソースがスプリットブレイン状態に陥ることを防止する必要があります。

## IP リソースを使用した排他制御

IP リソースには重複チェックを利用した排他制御機能があるため、複数ノードで同一の IP リソースが起動することはありません。この性質を利用して DataKeeper リソースのスプリットブレインを防止することが出来ます。
DataKeeper の子リソースに適当な仮想 IP アドレスを割り当てたIPリソースを配置することで、親になる DataKeeper リソースが複数ノードで同時に起動することを防止します。
なお、IP リソースの重複チェックはすべてのノードが同一サブネットに所属する必要があるため、この手法はローカルネットワークだけで構成された環境のみ適用できます。

## Quorum/Witness 機能による排他制御

LifeKeeper が持つ Quorum/Witness 機能を使用することで、複数ノードが同時にアクティブ状態になることを防止します。
Quorum/Witness 機能の詳細はテクニカルドキュメンテーションの Quorum/Witness を参照してください。

# 5.5.3.9. リソースの設定作業

SIOS DataKeeper の設定作業はすべて、LifeKeeper のグラフィカルユーザインターフェース（GUI）から実行できます。LifeKeeper の GUI では、SIOS DataKeeper のリソースの設定、管理、監視の作業をガイド付きで行うことができます。

## 概要

SIOS DataKeeper の設定に関して、以下の作業を行うことができます。

- **Create a Resource Hierarchy** - DataKeeper リソース階層を作成します。
- **Delete a Resource Hierarchy** - DataKeeper リソース階層を削除します。
- **Extend a Resource Hierarchy** - DataKeeper リソース階層をプライマリサーバからバックアップサーバに拡張します。
- **Unextend a Resource Hierarchy** - LifeKeeper クラスタ内にある 1 台のサーバの DataKeeper リソース階層を拡張解除（削除）します。
- **Create Dependency** - 既存のリソース階層と別のリソースインスタンスとの間に子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を伝播します。
- **Delete Dependency** - リソースの依存関係を削除して、クラスタ内にあるすべての対象サーバに依存関係の変更を伝播します。
- **In Service** - リソース階層をアクティブにします。
- **Out of Service** - リソース階層を非アクティブにします。
- **View/Edit Properties** - リソース階層のプロパティの表示または編集を行います。

# 5.5.3.9.1. DataKeeper リソース階層の作成

プライマリサーバで以下の操作を行ってください。

1. **[Edit] > [Server] > [Create Resource Hierarchy]** を選択してください。

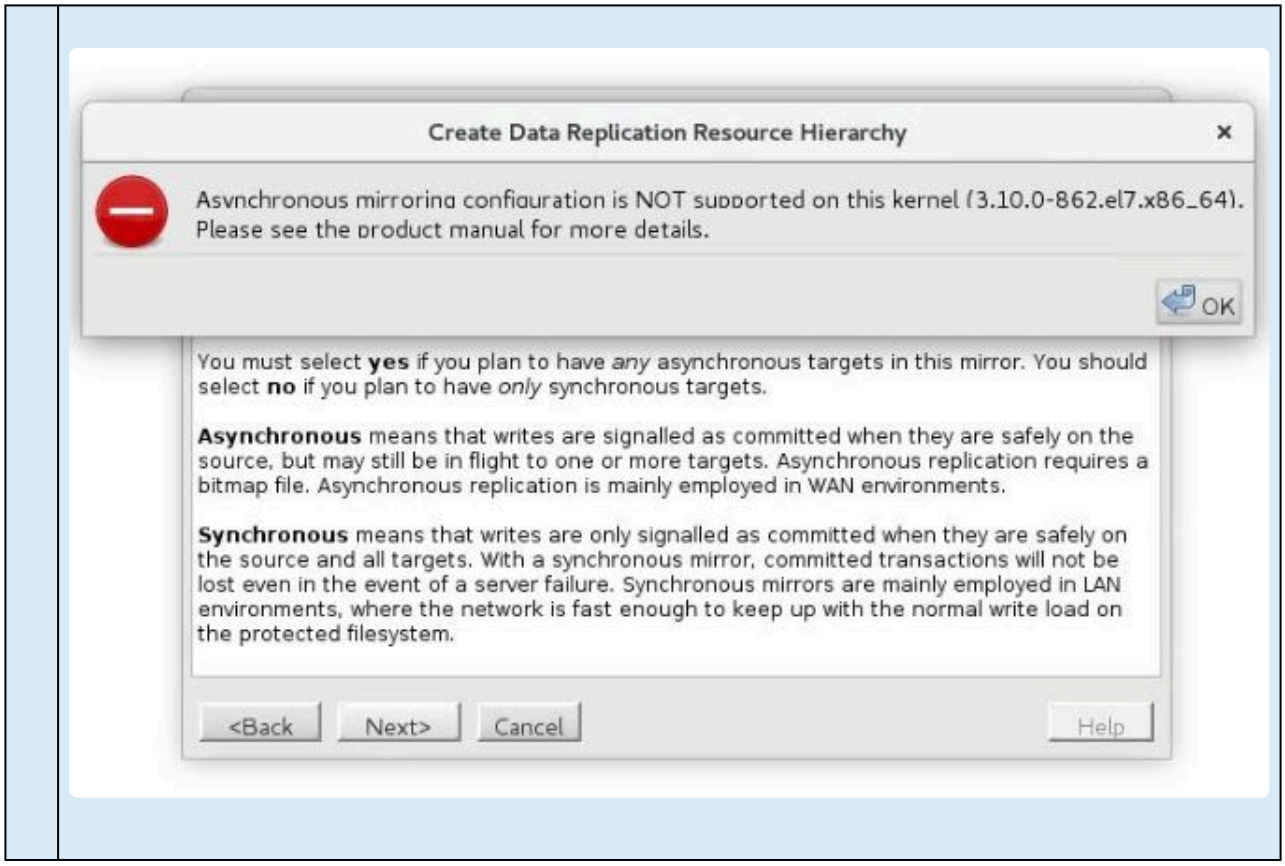
   ***[Create Resource Wizard]*** ダイアログボックスが表示されます。

2. ドロップダウンリストから **[Data Replication]** オプションを選択し、 **[Next]** をクリックして続行してください。

3. 以下の情報を入力するように要求されます。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも **[Cancel]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| Switchback Type | **[intelligent switchback]** を指定する必要があります。これは、バックアップサーバにフェイルオーバした後、管理者が手動で DataKeeper リソースをプライマリサーバにスイッチバックする必要があることを意味します。<br><br>**注意:** このリリースの SIOS DataKeeper は、DataKeeper リソースの自動スイッチバックをサポートしていません。さらに、自動スイッチバックの制限は、DataKeeper リソースの上に存在する他の LifeKeeper リソースにも適用されます。 |
| Server | 作成する NetRAID デバイスが存在するサーバ（通常はプライマリサーバ）の名前を選択してください。ドロップダウンリストボックスには、クラスタ内のすべてのサーバが表示されます。 |
| Hierarchy Type | 以下のいずれかを選択して、作成するデータレプリケーションのタイプを選択してください。<br>• Replicate New File System<br>• Replicate Existing File System<br>• DataKeeper Resource |
| Bi | インテントログの記録に使用するビットマップファイルの名前を入力してください。 **[none]** を |

| | |
|---|---|
| tmapFile | 入力すると、インテントログは使用されず、すべての再同期が部分的ではなく全体の再同期になります。<br><br>**重要:** ビットマップファイルは btrfs ファイルシステム（またはサポートされていないその他の SPS for Linux ファイルシステム）上に置いてはいけません。データレプリケーションのビットマップファイルが btrfs ファイルシステム（またはサポートされていないその他の SPS for Linux ファイルシステム）上に置かれると、LifeKeeper がミラーを構成しようとした時、"invalidargument" エラーの原因になります。ビットマップファイルのデフォルトの置き場所は、 /opt/LifeKeeper の下です。このデフォルトの置き場所は、/opt/LifeKeeper が btrfs ファイルシステム上にある場合変更されます。<br><br>**注記:** btrfs は、現在 SIOS Protection Suite for Linux でサポートされていません。 |
| EnableAsynchronousReplication? | このレプリケーションリソースによるターゲットシステムへの非同期レプリケーションのサポートを許可するには、 **[Yes]** を選択してください。すべてのターゲットについて同期レプリケーションを使用する場合は、 **[No]** を選択してください。後で、レプリケーションリソースが各ターゲットサーバに拡張されるときに、実際のレプリケーションタイプ（同期または非同期）を選択するように要求されます。(両方のレプリケーションタイプの詳細については、SIOS DataKeeper によるミラーリング を参照してください)。これらのターゲットのいずれかへのレプリケーションを非同期で実行する場合は、他のターゲットへのレプリケーションを同期実行する場合でもここでは **[Yes]** を選択する必要があります。<br><br>**注記:** 非同期ミラーリングをサポートしない環境で非同期ミラーリングを選択した場合は、以下のメッセージが表示されます。<br>Asynchronous mirroring configuration is NOT supported on this kernel (<linuxカーネルのバージョン>). Please see the product manual for more details. |




以降の一連のダイアログボックスは、**[Hierarchy Type]** で選択した項目によって異なります。一部のダイアログボックスはすべての階層タイプで同じですが、表示される順序と必要な情報が少し異なることがあります。以下の 3 つのトピックで、階層作成の残りのプロセスについて説明しています。

- ◦ DataKeeper Resource
  ◦ Replicate New File System
  ◦ Replicate Existing File System

# 5.5.3.9.1.1. Replicate New File System

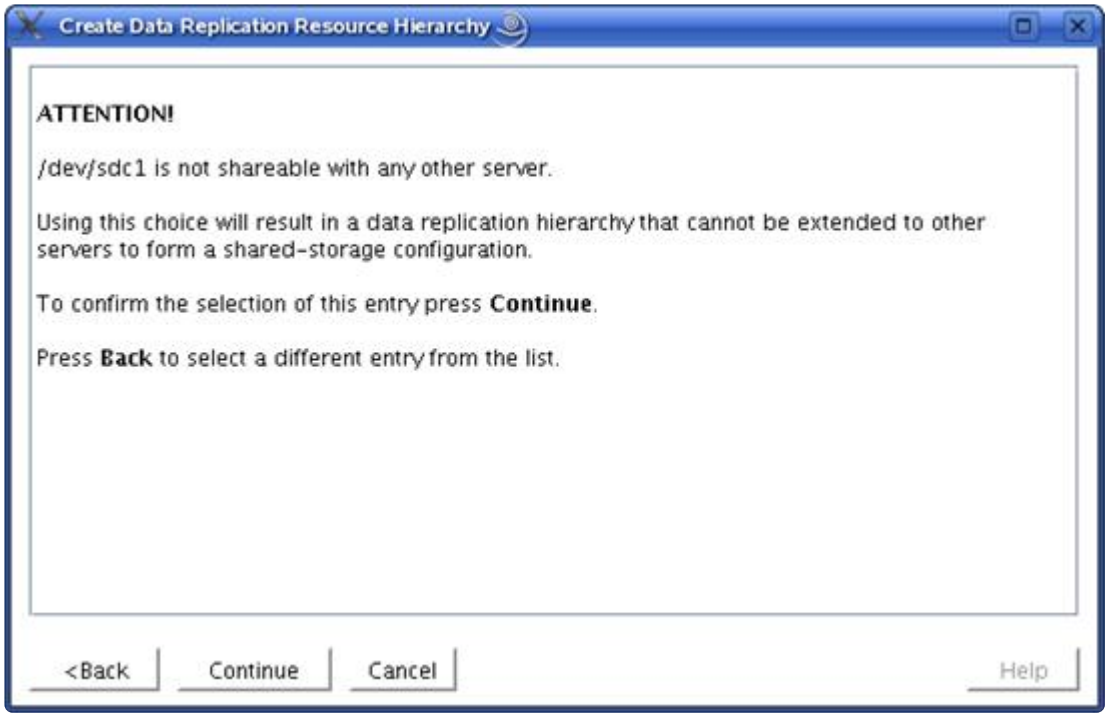
このオプションは、NetRAID デバイスを作成し、LifeKeeper がサポートするファイルシステムタイプでフォーマットします。ファイルシステムを NetRAID デバイスにマウントし、マウントしたファイルシステムと NetRAID デバイスの両方を LifeKeeper で保護します。NetRAID デバイスとローカルのディスクまたはパーティションがフォーマットされ、既存のデータが削除されます。新しいファイルシステムにミラーを作成し、LifeKeeper で保護する場合にこのオプションを選択してください。このリソースタイプには、1 つの空いているディスクまたはパーティションが必要です。

**注意:** このオプションを選択すると、ローカルのディスクまたはパーティションがフォーマットされ、既存のデータがすべて削除されます。

1. 要求されたら、以下の情報を入力してください。

| フィールド | ヒント |
|---|---|
| Source Disk or Partition | ドロップダウンリストには、 以下のものを除いて 、使用できるすべてのディスクが表示されます。<br>• 現在マウントされているもの<br>• スワップディスクまたはスワップパーティション<br>• LifeKeeper が保護するディスクまたはパーティション<<br><br>ドロップダウンリストには、root (*/*)、boot (*/boot*)、 */proc* 、floppy、cdrom などの特殊なディスクまたはパーティションも表示されません。<br><br>**注記:** ソースディスクまたはパーティションは一意に識別できる必要があります。v9.5.0 より、LifeKeeper ではデバイス名を使用してデバイスを識別できなくなります。 |

2. 非共有のソースのディスクまたはパーティションを選択した場合、以下の画面が表示されます。




3. 共有のソースのディスクまたはパーティションを選択するには、**[Back]** を選択してください。残りの情報を指定して、SIOS Protection Suite for Linux Multi-Site Cluster リソースの構成を完了してください。

| フィールド | ヒント |
|---|---|
| New Mount Point | 新しいファイルシステムの **新しいマウントポイント** を入力してください。これは、複製したディスクまたはパーティションが配置されるマウントポイントです。 |
| New File System Type | **ファイルシステムタイプを** 選択します。LifeKeeper がサポートするファイルシステムタイプのみを選択できます。 |
| DataKeeper Resource Tag | DataKeeper リソースインスタンスの一意の DataKeeper リソースタグ名を選択するか、入力してください。 |
| File System Resource Tag | ファイルシステムリソースインスタンスのファイルシステムリソースタグを選択するか、入力してください。 |
| Bitmap File | プルダウンリストからビットマップファイルの項目を選択してください。<br><br>表示されたリストには、ビットマップファイルの保持に使用できる共有ファイルシステムがあります。$LKROOT/bin ディレクトリを参照）。ビットマップファイルは、階層内のローカルノード間で切り替え可能な共有デバイスに配置する必要があります。<br><br>**重要:** ビットマップファイルは btrfs ファイルシステム（またはサポートされていないその他のSPS for Linuxファイルシステム）上に置いてはいけません。データレ |

| | |
|---|---|
| | プリケーションのビットマップファイルが btrfs ファイルシステム（またはサポートされていないその他のSPS for Linuxファイルシステム）上に置かれると、LifeKeeper がミラーを構成しようとした時、"invalid argument" エラーの原因になります。ビットマップファイルのデフォルトの置き場所は、 */opt/LifeKeeper* の下です。このデフォルトの置き場所は、 */opt/LifeKeeper* が btrfs ファイルシステム上にある場合変更されます。 |

4. **[Next]** をクリックして、 **確認** 画面に進んでください。

5. 確認画面に、新しいファイルシステムの作成場所、およびローカルのディスクまたはパーティションについて保留中の再フォーマットに関する警告が表示されます。 **[Create]** をクリックして、 **リソースの作成** を開始します。

6. リソースを新しいファイルシステムに作成するために有効なデータを指定したかどうかが、LifeKeeper により検証されます。LifeKeeper が問題を検知した場合は、情報ボックスにエラーが表示されます。検証が正常に完了すると、リソースが作成されます。ディスクまたはパーティションのサイズにより、ファイルシステムの作成には数分かかることがあります。

   **[Next]** をクリックして次に進んでください。

7. 新しい複製ファイルシステムのリソース階層が正常に作成されたことを示す情報ボックスが表示されます。複製を開始してリソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにリソース階層を **拡張** する必要があります。

   リソースを拡張する場合は **[Next]** 、後でリソースを拡張する場合は **[Cancel]** をクリックしてください。

   **[Continue]** をクリックすると、 ***Pre-extend Wizard*** が起動します。リソース階層を別のサーバに拡張する方法の詳細については、 リソース階層の拡張 の手順 2 を参照してください。

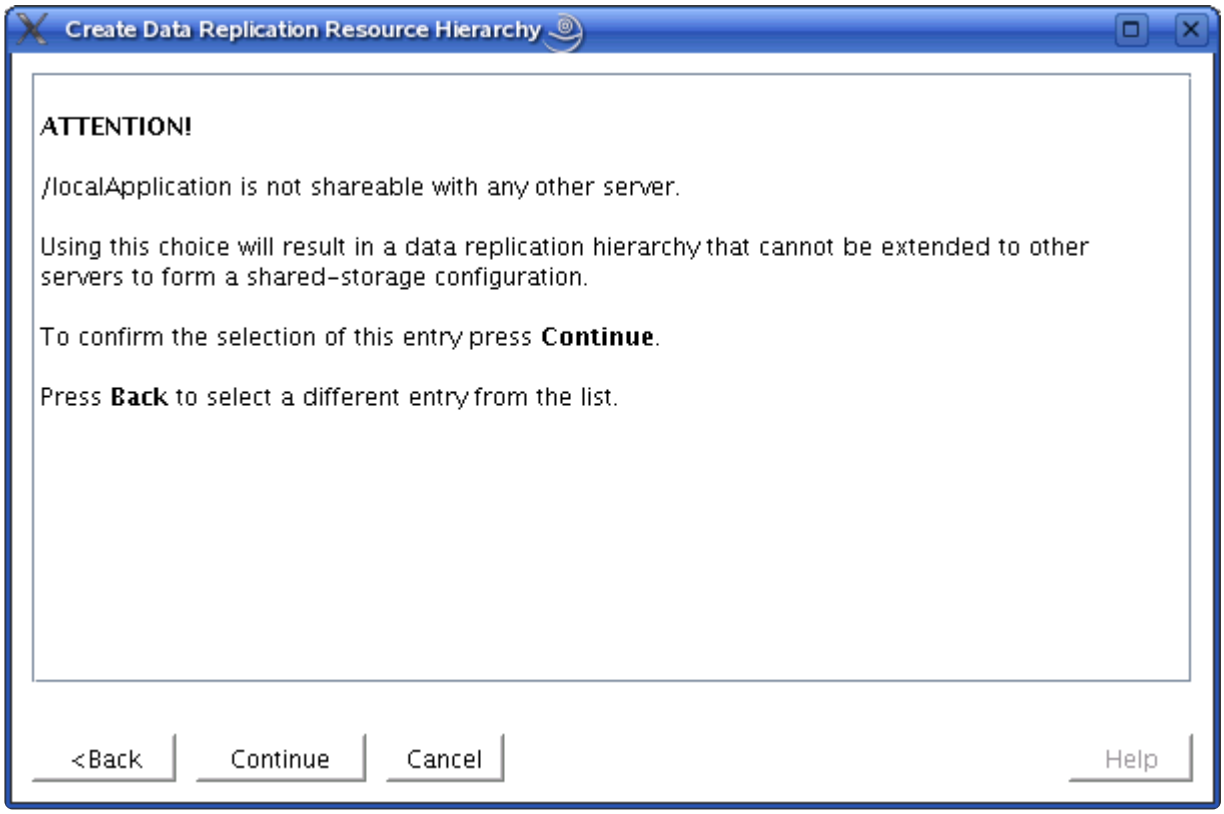
# 5.5.3.9.1.2. Replicate Existing File System

このオプションは、ローカルのディスクまたはパーティションに現在マウントされているファイルシステムをアンマウントし、NetRAID デバイスを作成して、ファイルシステムを NetRAID デバイスに再マウントします。NetRAID デバイスとマウントされたファイルシステムの両方が、LifeKeeper で保護されます。既存のファイルシステムにミラーを作成し、LifeKeeper で保護する場合にこのオプションを選択してください。

1. 要求されたら、以下の情報を入力してください。

| フィールド | ヒント |
|---|---|
| Existing Mount Point | これは、プライマリサーバの NetRAID デバイスにマウントするマウントポイントです。ローカルのディスクまたはパーティションがすでに、このマウントポイントにマウントされている必要があります。<br><br>**注記:** ソースディスクまたはパーティションは一意に識別できる必要があります。v9.5.0 より、LifeKeeper ではデバイス名を使用してデバイスを識別できなくなります。 |

2. 非共有のソースのマウントポイントを選択した場合、以下の画面が表示されます。




3. 共有のソースのディスクまたはパーティションを選択するには、**[Back]** を選択してください。残りの情報を指定して、SIOS Protection Suite for Linux Multi-Site Cluster リソースの構成を完了してください。

| フィールド | ヒント |
|---|---|
| DataKeeper Resource Tag | DataKeeper リソースインスタンスの一意の **DataKeeper リソースタグ名** を選択するか、入力してください。 |
| File System Resource Tag | **ファイルシステムリソースタグ** の名前を選択するか、入力してください。 |
| Bitmap File | プルダウンリストからビットマップファイルの項目を選択してください。<br><br>表示されたリストには、ビットマップファイルの保持に使用できる共有ファイルシステムがあります。$LKROOT/bin ディレクトリを参照）。ビットマップファイルは、階層内のローカルノード間で切り替え可能な共有デバイスに配置する必要があります。<br><br>**重要:** ビットマップファイルは btrfs ファイルシステム（またはサポートされていないその他のSPS for Linuxファイルシステム）上に置いてはいけません。データレプリケーションのビットマップファイルが btrfs ファイルシステム（またはサポートされていないその他のSPS for Linuxファイルシステム）上に置かれると、LifeKeeper がミラーを構成しようとした時、"invalid argument" エラーの原因になります。ビットマップファイルのデフォルトの置き場所は、*/opt/LifeKeeper* の下です。このデフォルトの置き場所は、*/opt/LifeKeeper* が btrfs ファイルシステム上にある場合変更されます。<br><br>**重要:** プルダウンリストには、レプリケーション対象となる共有ディスク領域も表示されますが、これをビットマップファイルの保存領域として選択しないでください。ビットマップファイルの保存先に、レプリケーション対象となる共有ファイルシステムを使用することはできません。必ずレプリケーション領域とは別の、 ビットマップ専用に追加した共有ファイルシステムを選択するようにしてください。 |

4. **[Next]** をクリックして、DataKeeper リソースをプライマリサーバに作成してください。

5. DataKeeper リソースのを作成するために有効なデータを指定したかどうかが、LifeKeeper により検証されます。LifeKeeper が問題を検知した場合は、情報ボックスにエラーが表示されます。検証が正常に完了すると、リソースが作成されます。

   **[Next]** をクリックしてください。

6. 既存のレプリケーションファイルシステムのリソース階層が正常に作成されたことを示す情報ボックスが表示されます。レプリケーションを開始してリソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにリソース階層を **拡張** する必要があります。

   リソースを拡張する場合は **[Next]** 、後でリソースを拡張する場合は **[Cancel]** をクリックしてください。

**[Continue]** をクリックすると、 *Pre-extend Wizard* が起動します。リソース階層を別のサーバに拡張する方法の詳細については、 リソース階層の拡張の手順 2 を参照してください。

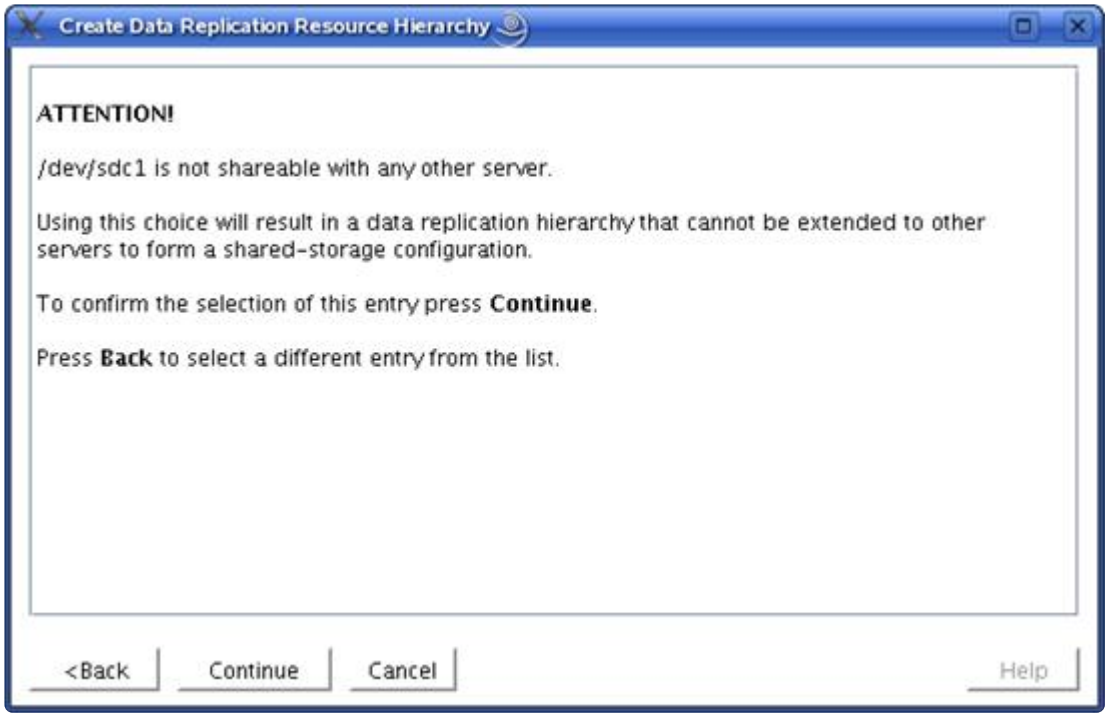
# 5.5.3.9.1.3. DataKeeper Resource

このオプションは、NetRAID デバイスのみを作成し（ファイルシステムは作成しない）、NetRAID デバイスを LifeKeeper で保護します。ディスクまたはパーティション上に DataKeeper デバイスのみを作成し、LifeKeeper で保護する場合にこのオプションを選択してください。読み取り可能なミラーを作成するには、このデバイス上にファイルシステムを作成し、マウントする操作を手動で行う必要があります。このリソースタイプには、1 つの空いているディスクまたはパーティションが必要です。

1. 要求されたら、以下の情報を入力してください。

| フィールド | ヒント |
| --- | --- |
| Source Disk or Partition | ドロップダウンボックスのソースディスクまたはパーティションのリストには、以下のものを除いて、使用できるすべてのディスクが表示されます。<br>• ○ 現在マウントされているもの<br>• ○ スワップディスクまたはスワップパーティション<br>• ○ LifeKeeper が保護するディスクまたはパーティション<br><br>ドロップダウンリストには、root (*/*)、boot (*/boot*)、 */proc* 、floppy、cdrom などの特殊なディスクまたはパーティションも表示されません。<br><br>**注記:** VMware を使用する場合は、「VMware の既知の問題」を参照してください。<br><br>**注記:** ソースディスクまたはパーティションは一意に識別できる必要があります。v9.5.0 より、LifeKeeper ではデバイス名を使用してデバイスを識別できなくなります。 |

2. 非共有のソースのディスクまたはパーティションを選択した場合、以下の画面が表示されます。




3. 共有のソースのディスクまたはパーティションを選択するには、 **[Back]** を選択してください。残りの情報を指定して、SIOS Protection Suite for Linux Multi-Site Cluster リソースの構成を完了してください。

| フィールド | ヒント |
|---|---|
| DataKeeper Resource Tag | DataKeeper リソースインスタンスの一意の **DataKeeper リソースタグ名** を選択するか、入力してください。 |
| Bitmap File | プルダウンリストからビットマップファイルの項目を選択してください。<br><br>表示されたリストには、ビットマップファイルの保持に使用できる共有ファイルシステムがあります。$LKROOT/bin ディレクトリを参照）。ビットマップファイルは、階層内のローカルノード間で切り替え可能な共有デバイスに配置する必要があります。<br><br>**重要:** ビットマップファイルは btrfs ファイルシステム（またはサポートされていないその他のSPS for Linuxファイルシステム）上に置いてはいけません。データレプリケーションのビットマップファイルが btrfs ファイルシステム（またはサポートされていないその他のSPS for Linuxファイルシステム）上に置かれると、LifeKeeper がミラーを構成しようとした時、“invalid argument” エラーの原因になります。ビットマップファイルのデフォルトの置き場所は、 */opt/LifeKeeper* の下です。このデフォルトの置き場所は、 */opt/LifeKeeper* が btrfs ファイルシステム上にある場合変更されます。 |

4. **[Next]** をクリックします。

5. 使用する前に、ファイルシステムを手動で作成し、NetRAID デバイス（*/dev/mdX*）にマウントする必要があることを示す情報ウィンドウが表示されます。

   **[Create]** をクリックして、DataKeeper デバイスをローカルのディスクまたはパーティションに作成してください。

6. 情報ボックスが表示され、DataKeeper リソースのを作成するために有効なデータを指定したかどうかが、LifeKeeper により検証されます。LifeKeeper が問題を検知した場合は、情報ボックスにエラーが表示されます。検証が正常に完了すると、リソースが作成されます。

   **[Next]** をクリックして次に進んでください。

7. DataKeeper リソースデバイスが正常に作成されたことを示す情報ボックスが表示されます。データの複製を開始し、バックアップ/ターゲットサーバを LifeKeeper で保護するには、クラスタ内の別のサーバに階層を **拡張** する必要があります。

   リソースを拡張する場合は **[Continue]** 、後でリソースを拡張する場合は **[Cancel]** をクリックしてください。

   **[Continue]** をクリックすると、 ***Pre-extend Wizard*** が起動します。リソース階層を別のサーバに拡張する方法の詳細については、 リソース階層の拡張 の手順 2 を参照してください。

# 5.5.3.9.2. リソース階層の拡張

この操作は **[Edit]** メニューから開始できます。または **[Create Resource Hierarchy]** オプションの動作が完了すると自動的に開始されます。その場合は、手順 2 を参照してください。

1. **[Edit]** メニューの **[Resource]** から **[Extend Resource Hierarchy]** を選択します。 ***Pre-Extend Wizard*** が表示されます。拡張操作に慣れていない場合は、 **[Next]** をクリックしてください。LifeKeeper の **[Extend Resource Hierarchy]** のデフォルト値が分かっていて、入力と確認を省略する場合は **[Accept Defaults]** をクリックしてください。

2. ***Pre-Extend Wizard*** に以下の情報を入力します。

   **注記:** 最初の 2 つのフィールドは ***[Edit]*** メニューから拡張を開始した場合にのみ表示されます。

| フィールド | ヒント |
|---|---|
| *Template Server* | 現在 In Service のDataKeeper リソース階層 が存在するテンプレートサーバを選択してください。ここで選択する **テンプレートサーバ** と次のダイアログボックスで選択する **拡張するタグ** によって、 *In Service* （アクティブ）のリソース階層が表されることを理解しておくことが重要です。<br><br>選択したテンプレートサーバで In Service でないリソースタグを選択した場合、エラーメッセージが表示されます。このダイアログのドロップダウンボックスには、クラスタ内にある全サーバの名前が表示されます。 |
| *Tag to Extend* | これは、テンプレートサーバからターゲットサーバに拡張する *DataKeeper* インスタンスの名前です。ドロップダウンボックスには、テンプレートサーバ上に作成したすべてのリソースが表示されます。 |
| *Target Server* | 拡張先のサーバを入力するか、選択してください。 |
| *Switchback Type* | *[intelligent switchback]* を指定する必要があります。これは、バックアップサーバにフェイルオーバした後、管理者が手動で *DataKeeper* リソースをプライマリサーバにスイッチバックする必要があることを意味します。<br><br>**注意**: このリリースの *SIOS DataKeeper* は、 *DataKeeper* リソースの自動スイッチバックをサポートしていません。さらに、自動スイッチバックの制限は、 *SIOS DataKeeper* リソースの上に存在する他の *LifeKeeper* リソースにも適用されます。 |
| *Template Priority* | テンプレートの優先順位を選択するか、入力してください。これはサーバで現在 *In Service* の *DataKeeper* 階層の優先順位です。 *1* ～ *999* の範囲で、まだ優先順位として使用されていない値が有効で、小さい数字ほど優先順位が高くなります(数値 *1* が最高の優先順位)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値をが推奨されます。<br><br>**注記**: このフィールドは、階層をはじめて拡張するときにだけ表示されます。 |
| *Target* | ターゲットの優先順位を選択するか、入力してください。これは、他のサーバにある |

| | |
|---|---|
| *Priority* | 同等の階層に対する、新しく拡張する *DataKeeper* 階層の優先順位です。 *1* ～ *999* の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位は高くなります（数値 *1* が最高の優先順位）。 *LifeKeeper* のデフォルトでは、階層が作成されたサーバに「 *1* 」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |

拡張前のチェックが正常に終了したというメッセージが表示されたら、 *[Next]* をクリックしてください。

拡張する階層に応じて、拡張されるリソースタグ（一部編集不可）を示す一連の情報ボックスが表示されます。

3. ***[Next]*** をクリックして、 ***[Extend Resource Hierarchy]*** の構成タスクを開始してください。

4. 次のセクションには、別のサーバに *DataKeeper* リソースを拡張するために必要な手順を示します。

## DataKeeper リソース階層の拡張

1. *pre-extend* スクリプトが正常に実行されたというメッセージが表示されたら、以下の情報を指定するように要求されます。

| フィールド | ヒント |
|---|---|
| *Mount Point* | ターゲットサーバ上にあるファイルシステムマウントポイントを入力してください（ *DataKeeper* リソースに関連する、 *LifeKeeper* が保護するファイルシステムがない場合は、このダイアログは表示されません）。 |
| *Root Tag* | ルートタグを選択するか、入力してください。これは、ターゲットサーバ上にあるファイルシステムリソースインスタンスの一意の名前です（ *DataKeeper* リソースに関連する、 *LifeKeeper* が保護するファイルシステムがない場合は、このダイアログは表示されません）。 |
| *Target Disk or Partition* | 複製するファイルシステムの配置先となる、ターゲットサーバ上のディスクまたはパーティションを選択してください。<br><br>ドロップダウンボックスのディスクまたはパーティションのリストには、以下のものを除いて、使用できるすべてのディスクが表示されます。<br>• すでにマウント済みのもの<br>• スワップディスクまたはスワップパーティション<br>• スワップディスクまたはスワップパーティション<br><br>ドロップダウンリストには、 *root (/)*、*boot (/boot)*、*/proc, floppy*、*cdrom* などの特殊なディスクまたはパーティションも表示されません。 |

| | |
|---|---|
| | **注記:** ターゲットのディスクまたはパーティションは、ソースのディスクまたはパーティション以上のサイズである必要があります。 |
| *DataKeeper Resource Tag* | DataKeeper リソースタグの名前を選択するか、入力してください。 |
| *Bitmap File* | インテントログの記録に使用するビットマップファイルの名前を選択するか入力してください。 *[None]* を選択すると、インテントログは使用されず、すべての再同期が部分的ではなく全体の再同期になります。 |
| *Replication Path* | ターゲットサーバとクラスタ内の他の指定サーバとの間で複製に使用する、ローカルとリモートの IP アドレスのペアを選択してください。有効なパスおよび対応する IP アドレスは、このサーバのペアに対して指定した *LifeKeeper* コミュニケーションパスのセットから得られます。 *DataKeeper* の特性により、プライベート（専用）ネットワークを使用することが強く推奨されます。<br><br>*DataKeeper* リソースをすでに 1 台以上のターゲットサーバに拡張している場合、追加のサーバに対する拡張を実行すると、新しいターゲットサーバと既存のサーバとの組み合わせのそれぞれについて、繰り返し複製パスを指定するように要求されます。 |
| *Replication Type* | 指定したサーバのペアについて使用する複製タイプとして、 *[synchronous]* または *[asynchronous]* を選択してください。<br><br>前述の *[Replication Path]* フィールドと同様に、 *DataKeeper* リソースをすでに 1 台以上のターゲットサーバに拡張している場合、追加のサーバに対する拡張を実行すると、新しいターゲットサーバと既存のサーバとの組み合わせのそれぞれについて、繰り返し複製タイプを指定するように要求されます。 |

2. ***[Extend]*** をクリックして次に進んでください。拡張が実行中であることを確認する情報ボックスが表示されます。

3. ***[Finish]*** をクリックして、 *DataKeeper* リソースインスタンスが正常に拡張されたことを確認してください。

4. ***[Done]*** をクリックして、 ***[Extend Resources Hierarchy]*** メニューを終了してください。

**注記:** 必ず すべての サーバでスイッチオーバを手動実行して、新しいインスタンスの機能をテストしてください。詳細については、リソース階層のテスト を参照してください。この時点で、 *SIOS DataKeeper* がソースからターゲットのディスクまたはパーティションにデータの再同期を開始しています。 *LifeKeeper* の *GUI* では、ターゲットサーバにあ

> る *DataKeeper* リソースのステータスは「 *Resyncing* 」になります。再同期が完了すると、ステータスは「 *Target* 」になります。これは通常のスタンバイ状態です。

再同期中、 *DataKeeper* リソース、およびそれに依存するリソースはフェイルオーバできません。これは、データの破損を防止するためです。

# 5.5.3.9.3. リソース階層の拡張解除

LifeKeeper クラスタ内にある 1 台のサーバからリソース階層を削除するには、次の手順を実行します。

1. **[Edit]** メニューの **[Resource]** から **[Unextend Resource Hierarchy]** を選択してください。
2. DataKeeper リソースを拡張解除する **ターゲットサーバ** を選択してください。DataKeeper リソースが現在 In Service （アクティブ）のサーバは選択できません。

   **注記:** 右側のペインから個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックしてください。
3. **拡張解除する DataKeeper 階層** を選択し、 **[Next]** をクリックしてください（このダイアログは、いずれかのペインでリソースインスタンスを右クリックして、 **[Unextend]** を選択した場合には表示されません）。
4. 選択したターゲットサーバと DataKeeper リソース階層の拡張解除を確認する情報ボックスが表示されます。 **[Unextend]** をクリックしてください。
5. DataKeeper リソースが正常に拡張解除されたことを確認する別の情報ボックスが表示されます。 **[Done]** をクリックして、 **[Unextend Resource Hierarchy]** メニューを終了してください。

   **注記:** これで、データがバックアップサーバにレプリケーションされなくなります。

# 5.5.3.9.4. リソース階層の削除

LifeKeeper 構成内の すべての サーバから DataKeeper リソースを削除するには、次の手順を実行してください。

> 注記: DataKeeper リソースは、削除する前に Out of Service にすることが推奨されます。 Out of Service にしない場合、 **md** と **NetRAID** のデバイスが削除されず、ファイルシステムを手動でアンマウントする必要があります。DataKeeper リソースを Out of Service にする を参照してください。

1. **[Edit]** メニューの **[Resource]** から **[Delete Resource Hierarchy]** を選択してください。
2. 削除する DataKeeper リソース階層が存在する **ターゲットサーバ** の名前を選択してください。
   **注記:** 左側ペインのグローバルリソースまたは右側ペインの個々のリソースインスタンスを右クリックして [Delete Resource] 作業を選択した場合、このダイアログボックスは表示されません。
3. **[Hierarchy to Delete]** を選択してください（左右のペインのリソースインスタンスを右クリックして **[Delete Resource]** 作業を選択した場合、このダイアログは表示されません） **[Next]** をクリックしてください。
4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[Delete]** をクリックしてください。
5. DataKeeper リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。
   **[Done]** をクリックして終了してください。

> 注記: リソースを削除する前にマウントされた状態の NetRAID デバイスは、マウントされたまま残ります。それ以外の NetRAID デバイスは削除されます。

# 5.5.3.9.5. DataKeeper リソースを Out of Service にする

DataKeeper リソースを Out of Service にすると、LifeKeeper によるリソースの保護が解除されます。ミラーが解除され、ファイルシステムがアンマウントされます（該当する場合）。 **md** デバイスが停止し、**nbd** サーバとクライアントが強制終了されます。

**警告:** データのミラーリングを停止して LifeKeeper の保護を解除する場合以外は、DataKeeper リソースを Out of Service にしないでください。 **一時停止** の操作を使用して、ミラーリングを一時停止してください。

1. LifeKeeper の GUI の右側ペインにある、 In Service の **DataKeeper リソース** を右クリックしてください。
2. リソースのポップアップメニューの **[Out of Service]** をクリックしてください。
3. 選択したリソースが Out of Service になることを示すダイアログボックスが表示されます。この操作に関連するリソースの依存関係がダイアログに表示されます。 **[Next]** をクリックしてください。
4. 情報ボックスに、 Out of Service にするリソースの結果が表示されます。 **[Done]** をクリックしてください。

# 5.5.3.9.6. DataKeeper リソースを In Service にする

DataKeeper リソースを In Service にする操作は、リソースの作成と似ています。LifeKeeper は **nbd** サーバとクライアントを起動し、ソースとターゲットのデバイス間でデータを同期する **md** デバイスを起動して、ファイルシステムをマウントします（該当する場合）。

1. 右側のペインにある **DataKeeper リソースインスタンス** を右クリックしてください。

2. ポップアップメニューの **[In Service]** をクリックしてください。選択したサーバとリソースを In Service にすることを確認するダイアログボックスが表示されます。 **[In Service]** をクリックしてリソースを In Service にしてください。

3. 情報ボックスに、 In Service にするリソースの結果が表示されます。この操作に関連するリソースの依存関係が確認ダイアログに表示されます。 **[Done]** をクリックしてください。

# 5.5.3.9.7. リソース階層のテスト

手動スイッチオーバを開始することによって、DataKeeper リソース階層をテストできます。このテストは、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

## LifeKeeper の GUI からの手動スイッチオーバの実行

手動スイッチオーバを開始するには、LifeKeeper の GUI で **[Edit] > [Resource] > [In Service]** を選択します。例えば、バックアップサーバで In Service リクエストが実行されると、DataKeeper リソース階層がバックアップサーバ側で In Service になり、プライマリサーバ側では Out of Service になります。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバになります。

新しいプライマリサーバーの DataKeeper リソースの状態は、LifeKeeper GUI で [Source] に設定されています。スイッチオーバー中は、各ターゲットの DataKeeper リソースの状態は最初は [Out of Sync] に設定され、データがまだそのターゲットに複製されていないことを示します。マルチターゲット構成の場合、前のソースには[Out of Sync]、その他のターゲットには [Out of Sync (Wait for Previous Source)] と表示されます。 再同期は、前のソースから各ターゲットの順に、自動的に開始されます（状態は [Resyncing] に移行します）。再同期が完了すると、状態は [Target] に変わります。これは通常の スタンバイ 状態です。これらの状態通常素早く変化するため、GUI には表示されない場合があります。

**注記:** 再同期中は、DataKeeper リソースの手動フェイルオーバはできません。

**[Out of Service]** 要求を実行した場合、リソース階層は他のサーバで In Service にならずに、Out of Service になります。リソースを同じサーバ上で In Service に戻すことができるのは、再同期中にリソースが Out of Service になった場合のみです。

# 5.5.4. 管理

## SIOS DataKeeper for Linux の管理

以下のトピックには、リソースを作成した後の SIOS DataKeeper for Linux の動作と問題を理解し、管理するのに役立つ情報があります。

---


- ミラーステータスの表示
- GUI からのミラーの管理
  - ミラーを強制的にオンラインにする
  - 一時停止と再開
  - 圧縮レベルの設定
- コマンドラインからのミラー管理
- コマンドラインからのミラーステータスの監視
- サーバの障害
- 再同期
- 全同期の回避

# 5.5.4.1. ミラーのステータスの表示

**[Replication Status]** ダイアログには、ミラーに関する以下の情報が表示されます。

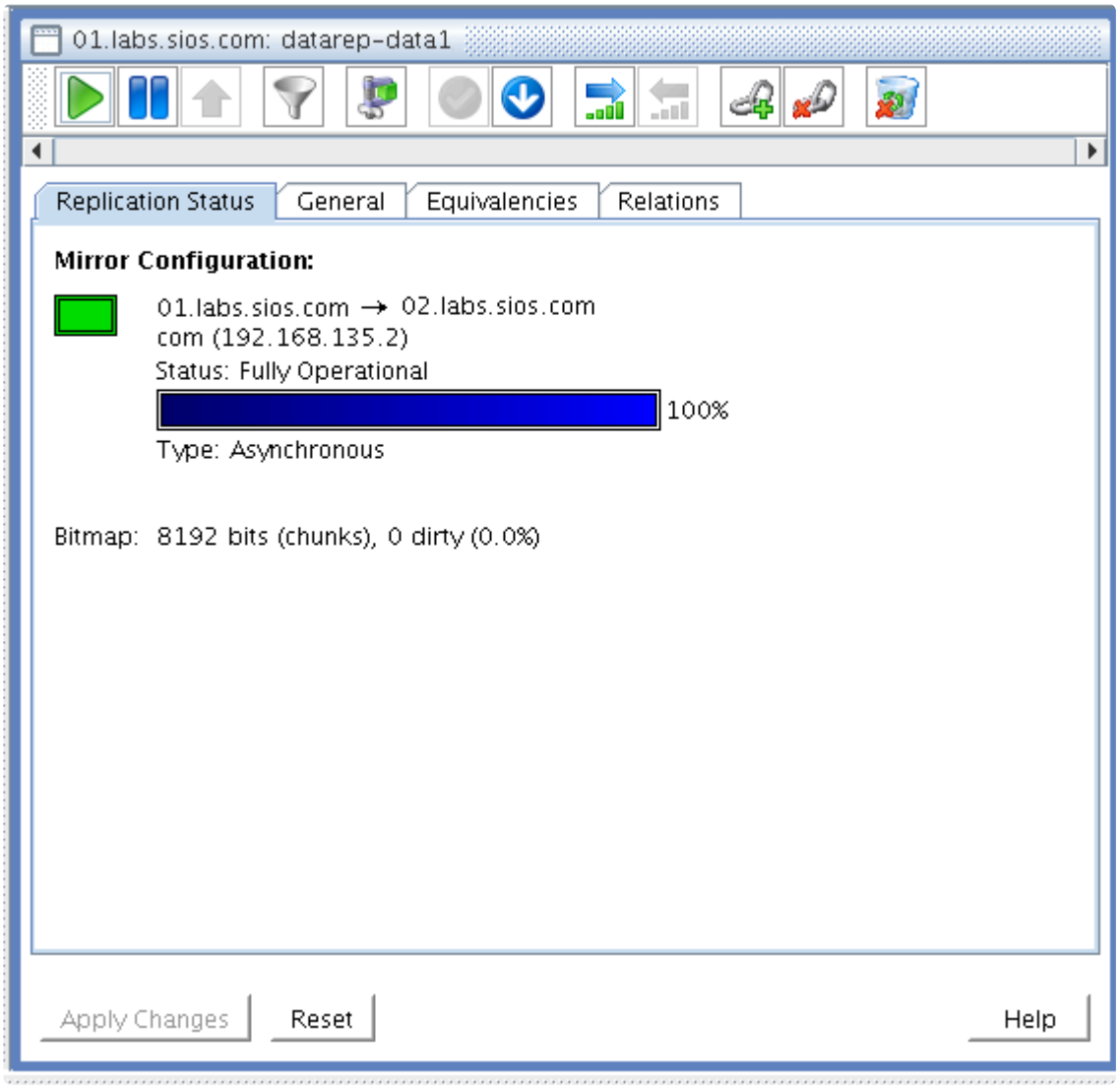
- ◦ **Mirror status**:Fully Operational（フルに動作可能）、Paused（一時停止）、Resyncing（再同期中）、またはOut Of Sync（同期停止）
- ◦ **Synchronization status**: 同期が完了した割合
- ◦ **Replication type**: synchronous（同期）または asynchronous（非同期）
- ◦ **Replication direction**: ソースサーバからターゲットサーバに
- ◦ **Bitmap**: ビットマップ / インテントログの状態
- ◦ **Network Compression Level**: 圧縮レベル（有効の場合）

**[Replication Status]** ダイアログを表示するには、次の手順に従います。

1. **[View]** メニューをクリックし、 **[Properties Panel]** を選択します。
2. **[LifeKeeper status]** 表示にある **DataKeeper リソース** をクリックします。

   または

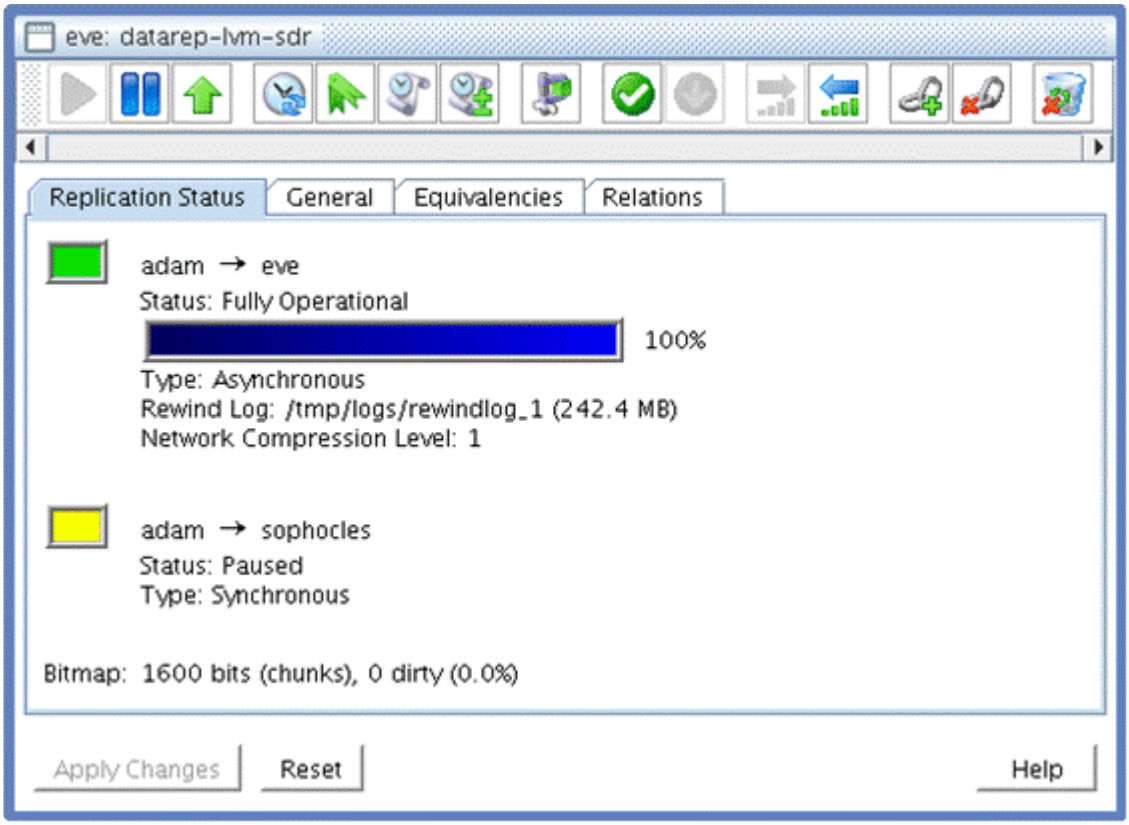
1. **[LifeKeeper status]** 表示にある **DataKeeper リソース** を右クリックします。
2. ポップアップメニューから **[Properties]** を選択します。


01.labs.sios.com: datarep-data1
Replication Status
General
Equivalencies
Relations
Mirror Configuration:
01.labs.sios.com → 02.labs.sios.com
com (192.168.135.2)
Status: Fully Operational
100%
Type: Asynchronous
Bitmap: 8192 bits (chunks), 0 dirty (0.0%)
Apply Changes
Reset
Help

# 5.5.4.2. GUI からのミラーの管理

SIOS DataKeeper のミラーは、LifeKeeper の GUI から以下の 2 とおりの方法で実行できます。

1. **[Properties Panel]** を有効にし、ツールバーのアイコン（スクリーンショットを参照）をクリックします。




説明を表示するには、それぞれのアイコンをクリックしてください。

または

2. **data replication** リソース を右クリックし、ポップアップメニューから動作を選択します。

# ミラーを強制的にオンラインにする

**[Force Mirror Online]** は、両方のサーバが動作不能になり、かつプライマリサーバの再起動後にリソースを In Service にできない場合にのみ使用してください。 **[Force Mirror Online]** を選択すると、*data_corrupt* フラグが削除され、DataKeeper リソースが In Service になります。詳細については、トラブルシューティング セクションの「プライマリサーバがリソースを ISP にできない」を参照してください。

**注記:** mirror_settings はターゲットシステム（ミラーのソースになるシステムには無関係に設定を有効にする場合はすべてのシステム）で実行する必要があります。設定の変更内容を有効にするには、ミラーを **一時停止** してから **再起動** する必要があります。

# 5.5.4.2.1. 一時停止と再開

## ミラーの一時停止

## ミラーの再開

ミラーを一時停止して、すべての書き込みをターゲットディスクに複製する動作を一時的に停止できます。例えば、ミラーを一時停止して、ターゲットディスクのスナップショットを収集することも、トラフィックのピーク時にソースシステムの I/O パフォーマンスを向上させることもできます。

ミラーを一時停止すると、ミラーは、ターゲットシステムの通常のファイルシステムのマウントポイントに読み取りアクセスするように（カーネル 2.6.19 以降は読み取り / 書き込みアクセス）マウントされます。ミラーの一時停止中にターゲットに書き込まれたデータはすべて、ミラーの再開時に上書きされます。

# 5.5.4.2.2. 圧縮レベルの設定

ネットワークの圧縮レベルは、0 ～ 9 の値に設定できます。値 0 は、圧縮を無効にします。レベル 1 は最も高速で圧縮率が最も低いレベルです。一方、レベル 9 は最も低速ですが圧縮率が最高です。ネットワーク圧縮は通常、WAN 環境で利用することによって効果を発揮します。

# 5.5.4.3. コマンドラインからのミラー管理

ミラーの管理は、LifeKeeper の GUI からの操作だけでなく、コマンドラインからも実行できます。DataKeeper のリソースの管理に使用できるコマンドがいくつかあります（*$LKROOT/bin* ディレクトリを参照）。

## ミラーの操作

```
mirror_action <tag> <action> [source] [target(s)]
```

**<tag>** DataKeeper リソースを表す LifeKeeper リソースタグ

**<action>** 以下のいずれか:pause, resume, force, fullresync

**[source]** ( オプション ) 現在のソースシステム (ソースが指定されない場合、コマンドが実行されたカレントシステムを使用)

**[target]** ( オプション ) 操作対象のターゲットシステム (またはシステムのリスト。ターゲットが指定されない場合、該当するすべてのターゲットを使用)

注記: force 操作を使用する場合、ソースノードを指定するために source 引数が必要ですが、target(s) 引数は必要ありません。pause, resume, fullresync のいずれかの操作を使用するときに target(s) 引数を指定する場合は、source 引数も必要です。

例:

*datarep-ext3* という名前のミラーを一時停止する。

```
mirror_action datarep-ext3 pause
```

*adam* から eve と *sophocles* の両方のシステムへの複製を再開する。

```
mirror_action datarep-ext3 resume adam eve sophocles
```

システム *eve* へのオンラインミラーリングを強制実行する。

```
mirror_action datarep-ext3 force eve
```

*adam* から *sophocles* への複製を再開し、これらのシステム間で全体の再同期を強制実行する。

```
mirror_action datarep-ext3 fullresync adam sophocles
```

## ミラーのサイズ変更

mirror_resize コマンドは、リソースの再作成をせずにミラーのサイズを変更することができま

す。ミラーのサイズを変更する前に、ソースシステムおよびターゲットシステムの両方でベースとなるデバイスのサイズを変更する必要があります。その後、ソースシステム上でこのコマンドを実行してください。ベースとなるデバイスのサイズが自動的に検出され、新しいミラーのサイズとして使用されます。また、オプションでサイズを指定することもできます。

LifeKeeper v9.5.0 からは、リソースが In Service の状態でもミラーサイズを変更できるようになりました。ただし、ミラーサイズを縮小する場合は、リソースの状態は Out of Service でなければなりません。なお、ファイルシステムによっては縮小に対応していないものもあります。

**注記:** エラー等でコマンドが途中で終了した場合は、”-f” オプションを付けて再度実行してください。正しく終了しないと、データ不整合が発生する可能性があります。

```
mirror_resize [-f] [-s <size>] <tag>
```

`<tag>` ミラーリソースのタグ。

`-f` ユーザに確認せずにサイズ変更を強制実行（推奨しません）。

`-s <size>` ミラーサイズ (KB 単位) を指定。

## ミラーのサイズコマンドの利用条件

- ベースとなるデバイスは論理ボリューム (LV) であること
- 単一のターゲットであること（複数ターゲット構成は非サポート）

## ミラーのサイズ変更の推奨手順

1. ベースとなるデバイスのサイズを変更します。ソースとターゲットの両方でこれを実行します(ターゲットのサイズはソースのサイズ以上でなければならないことに注意してください)。
2. ソースシステムで `mirror_resize` を実行します。これによってミラーの内部メタデータとビットマップが更新され、新しく拡張したデバイスサイズが反映されます。

   例： `mirror_resize -s <size in KB> <tag>`
3. リソースの状態が Out of Service の場合、ミラー ( *datarep* ) リソースのみを In Service にします。新しく拡張したデバイス、または パーティションの再同期が実行されます。
4. ミラーデバイス上でファイルシステムのサイズ変更を実行します (例: resize2fs /dev/mdX。X はサイズを変更するミラーの md デバイス番号で、/dev/md0 のようになります)。

**注記:** ファイルシステムのサイズを変更する前に、`fsck` が必要になる場合があります。

> **注記:** ファイルシステムによってはマウントしていなければサイズ変更ができないことがあります。その場合は、In Service にしてから実行してください。

5. リソースの状態が Out of Service の場合、ファイルシステムとアプリケーションリソースを In Service にします。

### XFS ファイルシステムを使用したミラーサイズ変更の推奨手順

1. ベースとなるデバイスのサイズを変更します。ソースとターゲットの両方でこれを実行します(ターゲットのサイズはソースのサイズ以上でなければならないことに注意してください)。

2. ソースシステムで `mirror_resize` を実行します。これによってミラーの内部メタデータとビットマップが更新され、新しく拡張したデバイスサイズが反映されます。

   **例:** `mirror_resize -s <size in KB> <tag>`

3. リソースの状態が Out of Service の場合、ミラー（datarep）リソースとファイルシステムリソースを In Service にします。新しく拡張されたデバイスまたはパーティションが再同期されます。

4. ファイルシステム上で、ファイルシステムのサイズ変更を実行します。

   **例:** `xfs_growfs -D size /path/to/file/system`

5. リソースの状態が Out of Service の場合、アプリケーションリソースを In Service にします。

## ビットマップの管理

```
bitmap -a <num>|-c|-d|-x <size_kb>|-X <bitmap_file>
```

`-a <num>`ビットマップファイルに非同期書き込みのパラメータを追加します。同期ミラーのアップグレードにより非同期ターゲットを含むようになった場合、これは必須です。<num> のデフォルト値は 256 です。この制限に最適な値を計算するには、 **SIOS DataKeeper for Linux** によるミラーリング の非同期ミラーリング情報 を参照してください。

`-c` ビットマップファイルをクリーニングします (*すべてのビットを 0 に設定*)。ソースディスクの余分な複製がターゲットに存在する場合、これを使用することで全体の再同期を回避できます。、 **このオプションは、特に注意して使用してください** 。

`-d` ビットマップファイルをダーティに設定します (*すべてのビットを 1 に設定*)。例えばスプリットブレインの状況が発生した後などに、このオプションを使用して全体の再同期を強制実行できます。

`-m` ビットマップを読み取り、マージストリームを作成します。

`-X <bitmap file>` ビットマップファイルを調べて、ビットマップとミラーに関する有用

な情報を表示します。

`-x <size_kb>` size_kb. のディスクが有効になるようにビットマップファイルを拡張します。
(注記：このオプションはミラーのサイズ変更のために内部的にのみ使用されます。)

さらに、`mdadm` コマンドを使用して、DataKeeper のリソースを管理できます。これは、DataKeeper のリソースが実際には md デバイスに存在するからです。詳細については、`mdadm(8)` のマニュアルページを参照してください。

注記：`mdadm` を使用するときには、OS 同梱の mdadm ではなく *$LKROOT/bin/mdadm* を必ず使用してください。

# 5.5.4.4. コマンドラインからのミラーステータスの監視

通常、ミラーステータスは、LifeKeeper の GUI から、**[Resource Properties]** ダイアログの **[Replication Status]** を使用して確認できます。ただし、以下の操作でもミラーのステータスを監視できます。

```
$LKROOT/bin/mirror_status <tag>
```

### 例:

```
# mirror_status datarep-ext3-sdr
[-]    eve -> adam
   Status: Paused
   Type: Asynchronous
[-]    eve -> sophocles
   Status: Resynchronizing
   [=>              ] 11%
   Resync Speed: 1573K/sec
   Type: Synchronous
Bitmap: 4895 bits (chunks), 4895 dirty (100.0%)
```

以下のコマンドも役に立つことがあります。

```
cat /proc/mdstat
```

サンプルの ***mdstat*** ファイルを示します。

```
eve:~ # cat /proc/mdstat
Personalities : [raid1]
md1 : active raid1 nbd10[1] nbd8[3](F) sdb1[0]
   313236 blocks super non-persistent [3/2] [UU_]
   bitmap: 3/3 pages [12KB], 64KB chunk, file: /opt/LifeKeeper/
```

```
        bitmap_ext3-sdr

    unused devices: <none/></tag>
```

# 5.5.4.5. サーバの障害

プライマリサーバとバックアップサーバの両方が動作不能になった場合、DataKeeper リソースは、 **両方**のサーバが再び動作可能になった場合にのみ In Service / アクティブになります。これは、間違った方向への再同期に起因するデータの破損を防ぐためです。動作可能なサーバが、リソースが「**In Service Protected**」( **ISP** ）である最後のサーバであることが確実に分かっている場合は、DataKeeper リソースを右クリックし、 **[Force Mirror Online]** を選択してそのリソースを強制的にオンラインにすることができます。

# 5.5.4.6. 再同期

DataKeeper リソースの再同期中、ターゲットサーバにあるこのリソースインスタンスのステータスは「**Resyncing**」（再同期中）になります。ただし、リソースインスタンスは、プライマリサーバの「**ソース**」（**ISP**）です。LifeKeeper の GUI は、ターゲットサーバにある DataKeeper のリソースのステータスを以下のアイコンで表示します。

プライマリサーバにある DataKeeper のリソースは、以下のアイコンで表示されます。

再同期が完了するとすぐに、ターゲットのリソースの状態が「**ターゲット**」になり、アイコンが以下のように変化します。

再同期プロセスについて、以下の点に注意してください。

- SIOS DataKeeper のリソースとその親リソースは、プライマリでの障害発生時に再同期プロセス中のターゲットにはフェイルオーバできません。
- ターゲットサーバの同期中に DataKeeper リソースが out of service / 非アクティブになった場合、そのリソースは、同じシステム、またはすでに同期済みの別のターゲット（複数のターゲットが存在する場合）でのみ In Service / アクティブにすることができ、再同期が続行されます。
- 再同期プロセス中にプライマリサーバが動作不能になった場合、同期プロセス中のターゲットサーバはすべて、DataKeeper リソースを In Service にすることができません。プライマリサーバが再び動作可能になった後に、ミラーの再同期が続行されます。

# 5.5.4.7. 全同期の回避

大量のデータを WAN リンク経由で複製する場合、膨大なネットワーク帯域幅と時間を消費する可能性がある全同期は避けることが望ましいです。新しいカーネルと共に使用する場合、SIOS DataKeeper はビットマップテクノロジを使用して全同期をほぼ防ぐことができます。ただし、既存のデータを複製する場合、ミラーの初期設定時に発生する最初の全同期を回避することはできません（新規データの場合には SIOS DataKeeper は全同期を実行しないので以降の手順は不要です）。

既存のデータを複製するときに、全同期を回避する方法がいくつかあります。ここでは推奨する2とおりの方法を説明します。

## 方法 1 – 2番目のノードへの複製

1 番目の方法では、RAW ディスクイメージを取得してターゲットサイトに輸送します。データがターゲットシステムに到着するまで、ソースシステムのミラーをアクティブにしておくことができるので、この方法ではダウンタイムが最小になります。

### 手順

1. ミラーを作成してください（[Replicate Existing Filesystem] を選択）。ただし、ターゲットシステムにミラーを拡張しないでください。

2. ミラーを out of service にしてください。

3. ソースディスクまたはパーティションのイメージを取得してください。この例では、選択したディスクまたはパーティションは /dev/sda1 です
   ```
   root@source# dd if=/dev/sda1 of=/tmp/sdr_disk.img bs=65536
   ```
   （ブロックサイズの引数65536は単に効率的にするためです。）

   ディスクまたはパーティションの RAW ディスクイメージを持つファイルが作成されます。
   ファイルの代わりに、ハードドライブやその他の記憶デバイスも使用できます。

4. オプション手順 - ソースディスクまたはパーティションのチェックサムを取得してください。
   ```
   root@source# md5sum /dev/sda1
   ```

5. オプション手順 - ディスクイメージファイルを圧縮してください。
   ```
   root@source# gzip /tmp/sdr_disk.img
   ```

6. ビットマップファイルをクリアしてください。
   （最後の引数はリソース作成時に指定したビットマップファイルのパスに置き換えてください。）
   ```
   root@source# /opt/LifeKeeper/bin/bitmap -c /opt/LifeKeeper/bitmap__dr
   ```

7. ミラーと依存ファイルシステム、およびアプリケーション（存在する場合）のサービスを開始してください。ビットマップファイルにより、データがターゲットシステムに転送される間に発生した変更内容が追跡されます。

8. 好みの転送方法を使用して、ターゲットシステムにディスクイメージを転送してください。

9. オプション手順 - ターゲットシステムでディスクイメージファイルを圧縮解除してください。
   ```
   root@target# gunzip /tmp/sdr_disk.img.gz
   ```

10. オプション手順 – イメージファイルのチェックサムが、手順 4 で取得した元のチェックサムと一致することを確認してください。
    ```
    root@target# md5sum /tmp/sdr_disk.img
    ```

11. イメージをターゲットシステム（例: /dev/sda2）に転送してください。
    ```
    root@target# dd if=/tmp/sdr_disk.img of=/dev/sda2 bs=65536
    ```

12. 両方のシステムで、/etc/default/LifeKeeper に LKDR_NO_FULL_SYNC=1 を設定してください。
    ```
    root@source# echo 'LKDR_NO_FULL_SYNC=1' >> /etc/default/LifeKeeper
    root@target# echo 'LKDR_NO_FULL_SYNC=1' >> /etc/default/LifeKeeper
    ```

13. ミラーをターゲットに拡張してください。部分的な再同期が実行されます。

14. /etc/default/LifeKeeper を編集してLKDR_NO_FULL_SYNC エントリを削除します。

## 全再同期を行わずに 3番目のノードまたは追加のノードに拡張する

### ソースからデータをコピーする方法

これらの手順では、ミラーがすでに作成され、2 番目のノード（ターゲット1）に拡張されていることを前提としています。

1. 各システムの /etc/default/LifeKeeper で、LKDR_NO_FULL_SYNC=1 を設定します。
   ```
   root@source# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
   root@target1# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
   root@target2# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
   ```

2. 新しいターゲット（ターゲット2）にミラーを拡張します。部分的な再同期が実行されます。

3. 新しいターゲット（ターゲット2）へのミラーを一時停止します。

4. ミラーをサービス停止にします。

5. ターゲット2で `umount <filesystem>` を実行して、ターゲット2 の一時停止したミラーのファイルシステムをアンマウントします。

6. ターゲット2で `mdadm --stop /dev/md#` を実行して、ターゲット2で実行されている md デバイスを停止します。 # は、 */proc/mdstat* で報告された値です。

7. ソースノードでソースディスクまたはパーティションのコピーを作成します。これは、dd またはディスクベンダーやクラウドベンダーのツールを使用して実行できます。コピーはファイルレベルのコピーではなく、 **ブロックごとに同一のコピーでなければなりません** 。

8. オプション手順 - チェックサムデータを収集してディスクイメージを検証します（md5sum、sha256sum など）。

9. オプション手順 - ディスクイメージを圧縮します。

10. ミラーとそれに依存するファイルシステムおよびアプリケーション（ある場合）を In-service にします。ビットマップファイルは、データがターゲットシステムに転送されている間に加えられた変更を追跡します。

11. ターゲット2 へのミラーがまだ一時停止していることを確認します。一時停止していない場合は、手順 4 で再起動します。

12. ファイルシステムと md デバイスがターゲット2 で実行されていないことを確認します。実行されている場合、ファイルシステムをアンマウントして md デバイスを停止します。

13. ターゲット2 のターゲットディスクにディスクイメージを転送します。

14. ディスクイメージが正しいことを確認します。ディスクの内容を検証するには、md5sum または sha256sum を使用できます。

15. 一時停止したミラーをターゲット2 に対して再開します。ソース上のビットマップは、ターゲット2 が一時停止してから実施された変更を追跡していましたが、ミラーが再開されると、これらの変更はターゲット2 に送信されます。

16. /etc/default/LifeKeeper を編集してLKDR_NO_FULL_SYNC エントリを削除します。

### 一時停止したターゲットからデータをコピーする手順

これによりダウンタイムは発生しなくなりますが、ターゲットが一時停止している間はデータの冗長性がなくなります。

1. この手順では、ミラーがすでに作成され、2 番目のノード（ターゲット1）に拡張されていることを前提としています。

2. 各システムの /etc/default/LifeKeeper で、LKDR_NO_FULL_SYNC=1 を設定します。
```
root@source# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
root@target1# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
root@target2# echo 'LKDR_NO_FULL_SYNC=1' >>/etc/default/LifeKeeper
```

3. 新しいターゲット（ターゲット2）にミラーを拡張します。部分的な再同期が実行されます。

4. 新しいターゲット（ターゲット2）へのミラーを一時停止します。

5. ターゲット1 へのミラーを一時停止します。

6. ターゲット2 で umount <filesystem> を実行して、ターゲット2 の一時停止したミラーのファイルシステムをアンマウントします。

7. ターゲット2 で `mdadm --stop /dev/md#` を実行して、ターゲット2 で実行されている md デバイスを停止します。 # は、 */proc/mdstat* で報告された値です。

8. ターゲット1 で `umount <filesystem>` を実行して、ターゲット1 の一時停止したミラーのファイルシステムをアンマウントします。

9. ターゲット1 で `mdadm --stop /dev/md#` を実行して、ターゲット1 で実行されている md デバイスを停止します。 # は、 */proc/mdstat* で報告された値です。

10. ターゲット1 のターゲットディスクまたはパーティションのコピーを作成します。これは、dd またはディスクベンダーやクラウドベンダーのツールを使用して実行できます。コピーはファイルレベルのコピーではなく、 **ブロックごとに同一のコピーでなければなりません** 。

11. オプション手順 - チェックサムデータを収集してディスクイメージを検証します（ md5sum、sha256sum など）。

12. オプション手順 - ディスクイメージを圧縮します。

13. ターゲット1 への複製を再開します。ビットマップファイルは、データがターゲット2 に転送されている間に加えられた変更を追跡します。

14. ターゲット2 へのミラーがまだ一時停止していることを確認します。一時停止していない場合は、手順 4 で再起動します。

15. ファイルシステムと md デバイスがターゲット2で実行されていないことを確認します。実行されている場合、ファイルシステムをアンマウントして md デバイスを停止します。

16. ターゲット2 のターゲットディスクにディスクイメージを転送します。

17. オプション手順 - ディスクイメージの圧縮を解除します。

18. オプション手順 - ディスクイメージが正しいことを確認します（md5sum、sha256sum など）。

19. 一時停止したミラーをターゲット2 に対して再開します。ソース上のビットマップは、ターゲット2 が一時停止してから実施された変更を追跡していましたが、ミラーが再開されると、これらの変更はターゲット2 に送信されます。

20. /etc/default/LifeKeeper を編集してLKDR_NO_FULL_SYNC エントリを削除します。

## 方法 2

ターゲットシステムを簡単に輸送できる場合、またはシステムの設定時にターゲットシステムがソースと同じ場所にある場合に、この方法を使用できます。この方法では、最初の全同期を高速なローカルネットワークで実行できるように、最終的な WAN ミラーを作成するネットワークルートを LAN ミラーに一時的に変更します。以下の例では、ソースサイトはサブネット 10.10.10.0/24 にあり、ターゲットサイトがサブネット10.10.20.0/24 にあると仮定しています。ソースとターゲットのシステムの間に一時的に静的ルートを設定することにより、ローカルのイーサネット接続またはループバックケーブルを使用して「WAN」ト

ラフィックをあるサーバーから別のサーバーに直接送信できます。

## 手順

1. ソースサイトでシステムをインストールし、設定してください。

2. 静的ルートを追加してください。
   ```
   root@source# ip route add 10.10.20.0/24 dev eth0
   root@target# ip route add 10.10.10.0/24 dev eth0
   ```
   この時点で、両方のシステムが LAN 上で相互に通信できる必要があります。

3. LifeKeeper でコミュニケーションパスを設定してください。

4. ミラーを作成し、ターゲットに拡張してください。全同期が実行されます。

5. ミラーを Pause にしてください。ミラーが再開されるまで、変更内容はビットマップファイルで追跡されます。

6. 静的ルートを削除してください。
   ```
   root@source# ip route del 10.10.20.0/24
   root@target# ip route del 10.10.10.0/24
   ```

7. ターゲットシステムをシャットダウンし、恒久的に配置する場所に輸送してください。

8. ターゲットシステムを起動し、ソースとのネットワーク接続を確立してください。

9. Resume Replication を実行してください。部分的な再同期が実行されます。

# 5.5.4.8. 再同期前のデータ検証 (Wait to Resync)

破損データや整合性のとれないデータがターゲットへ複製されるのを防ぐため、LifeKeeper は、データ複製の前に、リソースが In Serviceになるまで待機します。v9.5.2 からはデフォルトでLifeKeeper は、データの複製前にアプリケーションタイプ 'filesys' の親リソースが In Service になるのを待ちます。 これにより、データの複製前に、ファイルシステムの正確性が保証されます。ファイルシステムのマウントに失敗した場合は、適切なリカバリー対応として、データが複製される前にカレントシステムのファイルシステムを修復するか、別のシステムのデータを確認してください。

［Wait to Resync］機能は、 /etc/default/LifeKeeper の "LKDR_WAIT_TO_RESYNC" で設定可能です。
［Wait to Resync］ 機能を設定するにあたり 3 つのオプションがあります。:

1. **False** ［Wait to Resync］ 機能を無効にします。再同期前に親リソースは確認されません。再同期は、他に障害となる問題がなければ DataKeeper リソースの初期のリストア中に開始されます。

2. **<resource type>** 特定のリソースタイプを指定してください。デフォルトでは 'filesys' が指定されています。インストールされている任意のリソースタイプが選択可能です。 '/opt/LifeKeeper/bin/typ_list -f:' コマンドを実行することで、インストールされたリソースタイプの一覧を確認することができます。このコマンドのアウトプットは "アプリケーション:タイプ"の形式で表示されます。以下例です。

```
# typ_list -f:
gen:app
gen:filesys
gen:qsp
gen:hanfs
gen:nfs
scsi:DEVNAME
scsi:device
```

最初の項目はアプリケーションで、 ":" に続く2番目の項目はタイプになります。例えば、上記の [gen:app] の "gen" の部分がアプリケーションで "app" の部分がタイプになります。 ":" の右側はどのタイプでも指定が可能です。LifeKeeper は指定したタイプのDataKeeper リソースの親が In Service になるまで再同期を開始しません。例えば、/etc/default/LifeKeeper でリソースタイプ "LKDR_WAIT_TO_RESYNC=app"が定義されていて、リソース階層が下記のようになっていると仮定します。

```
# lcdstatus -q
LOCAL        TAG            ID                          STATE   PRIO  PRIMARY
ip-10-0-2-128  app1           app1                       OSU      1  ip-10-0-2-128
ip-10-0-2-128  SPSL1          SPSL1                      OSU      1  ip-10-0-2-128
ip-10-0-2-128  /maxdb1        /maxdb1                    OSU      1  ip-10-0-2-128
ip-10-0-2-128  datarep-maxdb1  536681bc-9259-4254       OSU      1  ip-10-0-2-128
ip-10-0-2-128  SPSL2          SPSL2                      OSU      1  ip-10-0-2-128
ip-10-0-2-128  /maxdb2        /maxdb2                    OSU      1  ip-10-0-2-128
ip-10-0-2-128  datarep-maxdb2  952e4ebd-507d-4b43       OSU      1  ip-10-0-2-128
ip-10-0-2-128  app3           app3                       OSU      1  ip-10-0-2-128
ip-10-0-2-128  app2           app2                       OSU      1  ip-10-0-2-128
ip-10-0-2-128  /maxdb1        /maxdb1                    OSU      1  ip-10-0-2-128
ip-10-0-2-128  datarep-maxdb1  536681bc-9259-4254       OSU      1  ip-10-0-2-128
```

app1、app2、app3 がリソースタイプ “app” をもっています。この場合、netraid リソース “datarep-maxdb1” は、3つ全ての gen:app となる親リソースが In Service になるまで同期を実施しません。netraid リソース “datarep-maxdb2”は、 “app1” のみがgen:app となる親なので、 “app1” が In Service になるまでは同期しません。

3. **Hierarchy** 同期前にDataKeeper リソースの の全ての親リソースが In Service になる必要があります。

> **注記:** 特に XFS ファイルシステムにおけるデータの不整合により In Service の処理に失敗することが頻繁にあるため ‘filesys’ タイプがデフォルトになっています。しかしながら、ファイルシステムはデータの不整合や破損があってもマウント可能です。アプリケーションを In Service にするか、階層を In Service にすることを要求すれば、データの高い正確性を保証するでしょう。

ユーザーはこの機能の情報を3つの方法で確認することができます。:

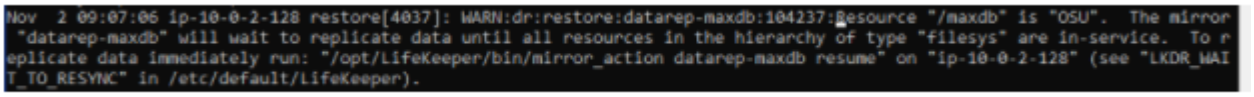
1. 要求された親リソースが Out of Service の場合は、GUI がターゲットを ［Wait to Resync］ と表示します。:

2. 要求された親リソースが Out of Service の場合は、In Service 処理の間に警告メッセージがログに記録されます。このメッセージは、確認中のタイプを指定し、“LKDR_WAIT_TO_RESYNC=hierarchy” となっている場合に、確認中の全ての階層を指定します。

   a. LKDR_WAIT_TO_RESYNC=filesys となっている場合の In Service 処理中に表示されるログメッセージの例です。:

b. LKDR_WAIT_TO_RESYNC=hierarchy となっている場合のログファイルの中のログメッセージです。:

WARN:dr:recover:datarep-maxdb:104237:Mirror “datarep-maxdb” will wait to reconnect targets until parent file system “/maxdb” is in-service. To reconnect targets immediately run: “/opt/LifeKeeper/bin/mirror_action datarep-maxdb resume” on “ip-10-0-2-128” (see “LKDR_WAIT_FOR_FILE_SYSTEM_TO_MOUNT” in /etc/default/LifeKeeper).



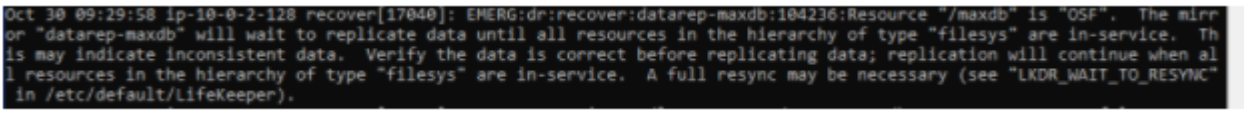

3. 要求された親リソースが 障害ステータス (OSF) の場合は、エマージェンシーメッセージが LifeKeeper ログファイルと全てのオープンしているターミナルに記録されます。

a. この場合、GUI は［Wait to Resync］ステータスを表示します。

b. 以下のメッセージが全てのターミナルと LifeKeeper ログファイルに記録されます。:

recover[26796]: EMERG:dr:recover:datarep-maxdb:104236:Resource “/maxdb” is “OSF”. The mirror “datarep-maxdb” will wait to reconnect targets until parent file system “/maxdb” is in-

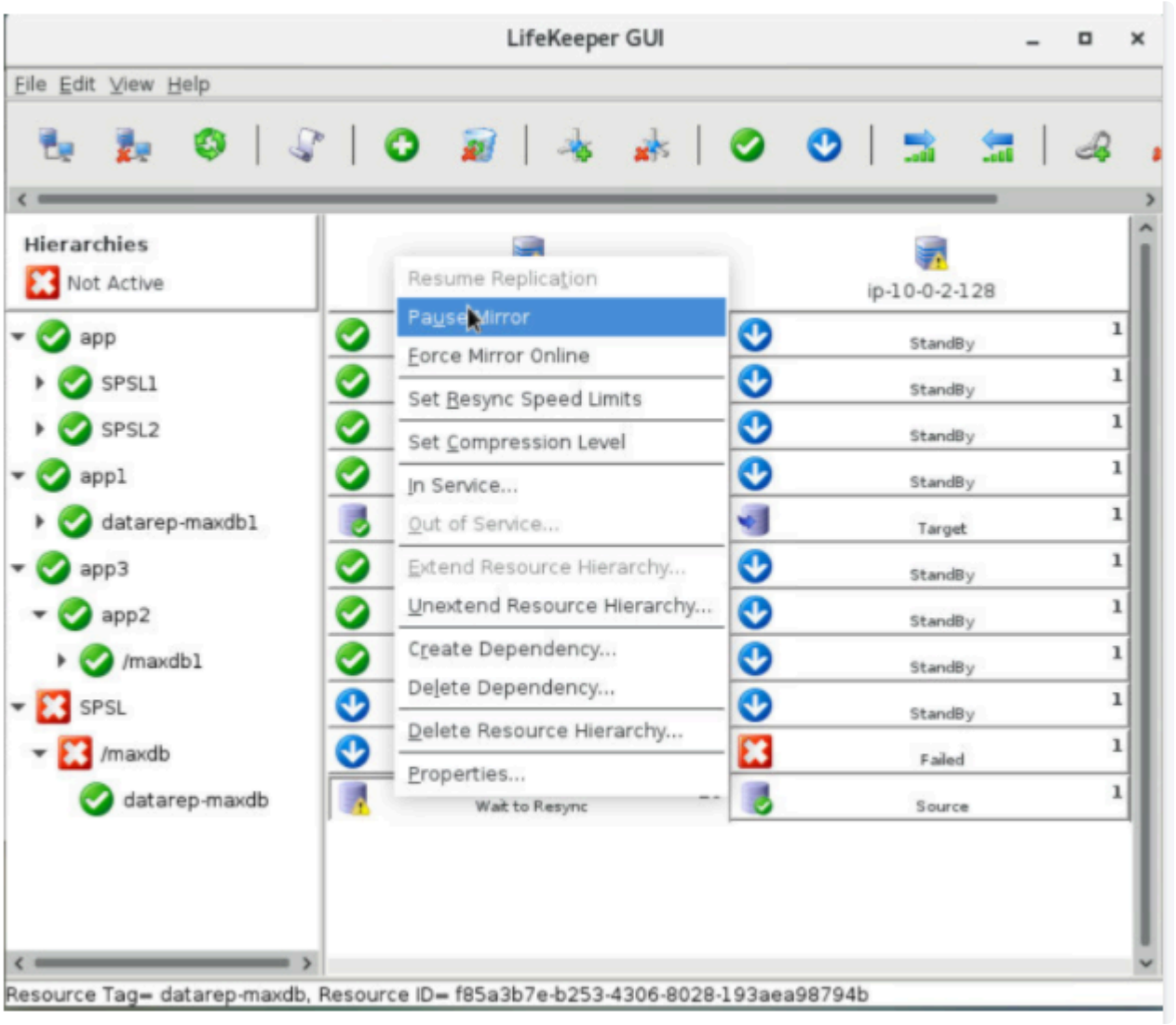
service. This may indicate inconsistent data. Do not bring the resource “/maxdb” in-service until the data has been verified; replication will continue when “/maxdb” is in-service. A full resync may be necessary (see “LKDR_WAIT_FOR_FILE_SYSTEM_TO_MOUNT” in /etc/default/LifeKeeper).



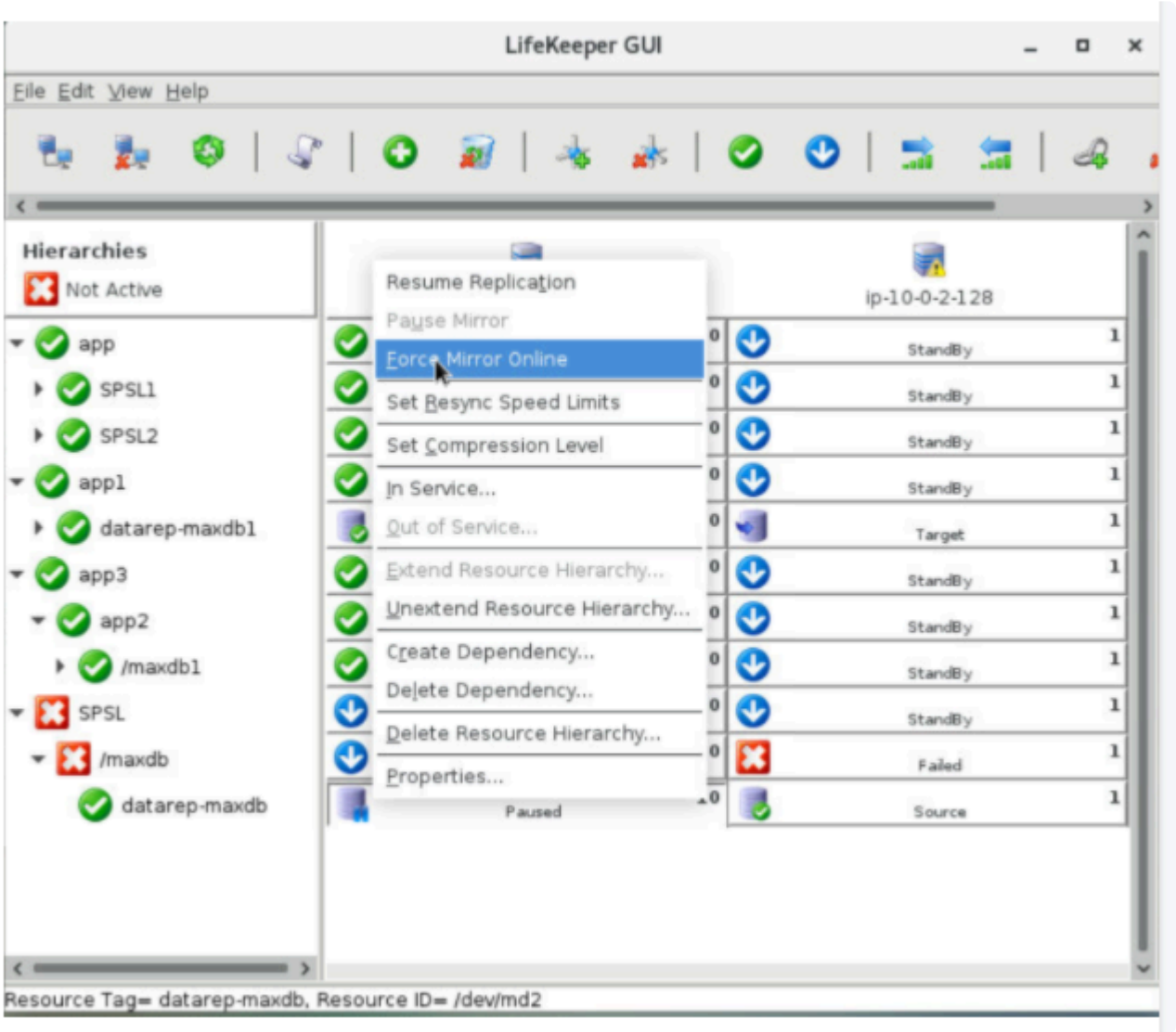

c. ファイルシステムのデータが検証されるまで、ファイルシステムリソースを In Service にしないでください。一旦ファイルシステムリソースが In Service になると、次の DataKeeper リソースに対する quickCheck のサイクル中に複製が再開されます。

d. 障害の発生したファイルシステムリソースは、マウント、ログの再生、fsck がファイルシステムを修復できないことを示します。これは、別のノードでファイルシステムを In Service にすることができた場合、ディスク上のデータ不整合を示している可能性があります。ファイルシステムを障害が発生したノードでリカバリーもしくは別のノードへの切り替えのどちらかの方法にて修復できたら、全てのノードのデータの一貫性を保つため、全同期が必要となります。

e. ファイルシステムがマウントできることおよびデータが検証されたことを確認するため、ミラーがターゲットにマウントされます。 これは [Pause Mirror] の機能を使用して実行可能です。

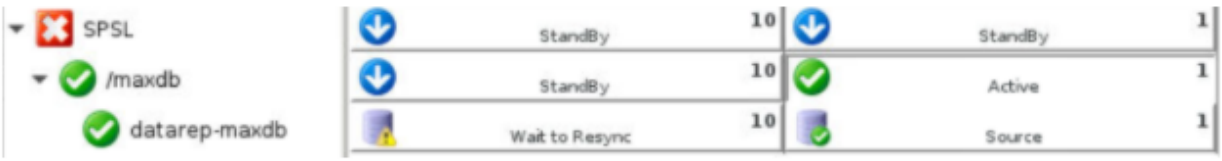



ミラーが一時停止されている際、LifeKeeper は自動的にファイルシステムをターゲットにマウントします。この処理に成功したら、データの整合性を確認してください。ターゲット側(一時停止されているサーバー)で [Force Mirror Online] を選択すると、ミラーを強制的にオンラインにすることができます。 **警告:** この処理は、親リソースが In Service になっていない場合であっても複製を再開します。データが検証済みの場合にのみミラーを強制的にオンラインにしてください。




データに全く問題ないことが確認できるまでは、ターゲット側の［Resume Replication］を選択 しないでください 。ミラーを一時停止した後に［Resume Replication］を選択すると、親リソースが In Service になっていない場合や、障害が発生した場合であっても、複製が開始されます。

f. データが検証され、親リソースの障害も修正され In Service になったら、DataKeeper リソースに対する次の quickCheck サイクルが、これ以上データの再同期を待つ必要がないことを検知します。親リソースが In Service であったとしても、［Wait to Resync］が表示される quickCheck サイクルまでに数分 (デフォルトで最大２分) 要します。




g. データの破損 (例：ファイルシステムを一つのノードにマウントできるが、別のノードにはマウントできないなど) が発見された場合や、破損が疑わしい場合は、全同期を実施することを提案します。全同期を強制的に実施するため、ミラーはまず最初に一時停止される必要があります。ミラーが一時停止されたら、ソースシステムで以下のコマンドを実行してください。:

```
/opt/LifeKeeper/bin/mirror_action <tag> fullresync
```

# 5.5.5. DataKeeper で LVM を使用する

SPS for Linux は現在 LVM 上での DataKeeper の使用と DataKeeper 上での LVM の使用の両方をサポートしています。標準的な DataKeeper の設定では LVM 上での DataKeeper の使用がサポートされているので、SPS LVM Recovery Kit をインストールしないでください。必要なリカバリキットは、DataKeeper だけです。ただし、DataKeeper 設定上で LVM を使用する場合は、LVM Recovery Kit が必要になります。

SIOS では LVM 上での DataKeeper 使用を推奨していますが、DataKeeper 設定上で LVM が使用される場合は、2 段階の階層作成プロセスが必要になります。DataKeeper デバイス (階層) では、プライマリサーバ上で LVM ボリュームグループと論理ボリュームを作成する前に、DataKeeper の「Data Replication Resource」オプションを使用した設定を行う必要があります。必要なボリュームグループと論理ボリュームが作成できたら、残りの階層を保護されるアプリケーションに関連したリカバリキットの設定手順に従って作成します。完成した階層は以下に示す図 3 のようになります。 **注記**: データの整合性を保つため、DataKeeper 上に LVM を構成する場合には、DataKeeper ミラー 1 つだけ、または複数の **同期** ミラー、いずれかの構成でなければなりません。

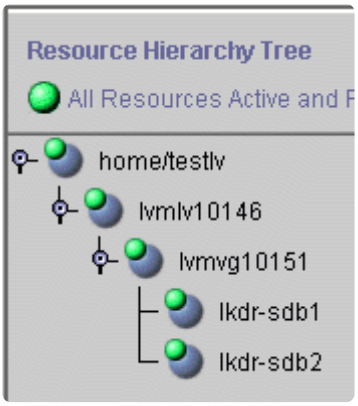
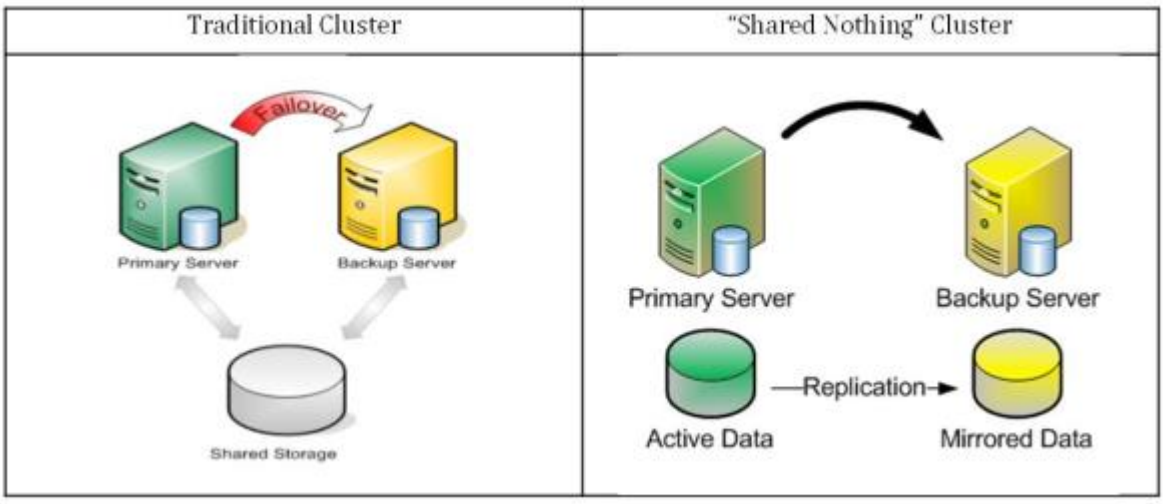



**図 3: DataKeeper 上の LVM を含む階層**

# 5.5.6. Fusion-io を使用するクラスタ化

## DataKeeper のパフォーマンスを最大に発揮させるための Fusion-io のベストプラクティス

SPS for Linux はブロックレベルのデータレプリケーション機能を備えており、共有ストレージを使用しないHAクラスタを非常に簡単に設定できます。Fusion-io を使用すると、SPS for Linux ではフェイルオーバ保護用として「shared nothing」クラスタを構成できます。




クラスタ設定の一部としてデータレプリケーションを活用する場合、ディスクへの書き込みと同等な速度でデータがネットワーク上で複製可能なだけの十分な帯域幅があることが重要です。以下のベストプラクティスを使用すると、高速ストレージを使用する場合に「shared nothing」SPS クラスタ設定を最大に活用できます。

### ネットワーク

- **10 Gbps の NIC を使用する。** Fusion-io のフラッシュベースのストレージデバイス (または OCZ、LSI、などの同様な製品) は、750 MB/ 秒を超えるデータ速度でデータを書き込むことができます。1 Gbps の NIC は理論的な最大値である約 125 MB/ 秒のみを送出できるので、ioDrive の性能を活用すると、1 Gbps のネットワーク接続での複製よりもはるかに高速で簡単にデータを書き込むことができます。リアルタイムのデータレプリケーションを実行するのに十分な帯域幅をサーバー間に確保するには、レプリケーショントラフィックの転送に必ず 10 Gbps の NIC を使用してください。

- **ジャンボフレームを有効にする。** 使用しているネットワークカードとスイッチでサポートされている場合、ジャンボフレームを有効にするとネットワークのスループットが増加するだけでなく、CPU サイクルが減少します。ジャンボフレームを有効にするには、以下の設定を実行してください (RedHat/CentOS/OEL Linux ディストリビューションの例)。

    - 次のコマンドを実行してください。
      ```
      ip link set <interface_name> mtu 9000
      ```

    - 再起動後も変更内容を保持するには、以下のファイルに「MTU=9000」を追加してください。

*/etc/sysconfig/network-scripts/ifcfg-<interface_name>*

- エンドツーエンドのフレーム動作を検証するには、次のコマンドを実行してください。
  `ping -s 8900 -M do <IP-of-other-server>`

- **NIC の転送キュー長を変更する。**
  - 次のコマンドを実行してください。
    `ip link set <interface_name> txqueuelen 10000`
  - 再起動後も変更内容を保持するには、 */etc/rc.local* に追加してください。

- **NIC の netdev_max_backlog を変更する。**
  - 次の設定を */etc/sysctl.conf* に追加してください。

```
net.core.netdev_max_backlog = 100000
```

### TCP/IP の調整

レプリケーションのパフォーマンスを向上することが確認された **TCP/IP の調整** を以下に示します。

- */etc/sysctl.conf* を編集して、以下のパラメータを追加してください ( **注記**: これらは例であり、使用している環境で異なる場合がある)。

```
net.core.rmem_default = 16777216
net.core.wmem_default = 16777216
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 65536 16777216
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_sack = 0
net.core.optmem_max = 16777216
net.ipv4.tcp_congestion_control=htcp
```

## 設定上の推奨項目

- Fusion-io ドライブ上に、ビットマップファイルを入れる小型の (約 100 MB) のディスクパーティションを割り当ててください。このパーティションにファイルシステムを作成し、マウントしてください (例: */bitmap* )。

```
# mount | grep /bitmap
/dev/fioa1 on /bitmap type ext3 (rw)
```

- ミラーを作成する前に、 */etc/default/LifeKeeper* 内の以下のパラメータを調整してください。

  ```
  LKDR_CHUNK_SIZE=4096
  ```

  デフォルト値は 256 です。

- 通常と同様に、ミラーを作成してクラスタを設定してください。

- ビットマップファイルは、上で作成したパーティションに作成するように設定してください。

- 再同期を高速化するための設定をおこないます。DataKeeper リソースの右メニューから、Set Resync Speed Limits を選択して、ウイザードに以下の値を設定してください。

  - `Minimum Resync Speed Limit: 200000`

- 同時に他の I/O が実行されているときに許可する再同期の速度を指定します。再同期の実行時に通常の I/O 動作が妨げられないようにするには、経験則として、この値をドライブの最大書き込みスループットの半分以下に設定する必要があります。

  - `Maximum Resync Speed Limit: 1500000`

- 再同期に使用する最大帯域幅を指定します。可能な最大速度で再同期が実行されるように、この値を十分高い値に設定する必要があります。

# 5.5.7. DataKeeper for Linux が保護している領域に対する、スナップショット機能の利用についての注意事項

DataKeeperで保護している領域に対して、ディスク領域のスナップショット機能を適用した場合、リストアしたデータを正とした完全同期が必要になります。

本稿で言及するスナップショット機能は、以下のようなものを含みます。

1. AWSなどのクラウド環境が提供しているもの
2. vSphereなどの仮想化ソフトウエアが提供しているもの
3. 物理環境において、ストレージが提供しているもの

これらのスナップショットのリストアは、OSを経由せずに行われます。その場合、DataKeeperはその書き込みをターゲット側に反映することができません。またbitmapにも書き込まれないため、差分同期で整合性を保つことも出来ません。

スナップショットを利用する場合の手順は以下のようになります。

1. Pause mirrorでミラーを静止状態とします
2. snapshotのリストアを行います
3. リストアした正のデータが格納されたディスクをソースとして、ミラーの完全同期を行います

ミラーの完全同期を実施するまでは、ソースとターゲットのデータは整合性のない状態となります。そのためスナップショットの適用後は完全同期の実行が必須となります。これはミラーのサイズによっては非常に長い時間を要する上に、完了するまではリソースを切り替えることができません。実施の際にはご注意ください。

なお、ソース、ターゲットの両方に同じスナップショットを戻し完全同期を回避するといった手法は、データの整合性が担保されないためサポートされません。

# 5.5.8. トラブルシューティング

このセクションでは、DataKeeper for Linux の使用時に遭遇する可能性がある問題に関する情報を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

DataKeeper for Linux に固有のメッセージについては、DataKeeper メッセージカタログ を参照してください。他の SPS コンポーネントからメッセージが送出されることもあります。その場合は、総合メッセージカタログ を参照してください。これらのメッセージカタログには、SIOS Protection Suite for Linux の使用中に遭遇される可能性のあるすべてのエラーコード (操作、管理、および GUI に関するものを含む) の一覧があります。また、エラーコードの原因に関する追加の説明や、問題解決のために必要な処置についても、必要に応じて記載されています。この一覧から、受信したエラーコードを検索できます。また、該当する SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

以下の表に、予測される問題と推奨される処置を示します。

| 症状 | 推奨される処置 |
|---|---|
| GUI に［Wait to Resync］が表示される。 | 再同期処理は親リソースが In Service になるまで待機します。LifeKeeper のログは、再同期処理の開始を妨げているリソースと、どのリソースが起動される必要があるかを示したメッセージを含みます。リソースが 単純に OSU になっているのであればリソースを In Service にするか ログメッセージに記載されている `mirror_action` コマンドを実行してください。リソースが OSF の場合は、リソースを In Service にする前に問題を解決することで、データの再同期が行われます。詳細については 再同期前のデータ検証 を参照してください。 |
| LKCLI のエクスポート/インポート機能を用いて、DataKeeper リソースのミラーリング方式がノード毎に異なるようなリソースを作成することはできません | GUI を用いて DataKeeper リソースを作成した場合、プライマリーからセカンダリー間を非同期転送、セカンダリーからプライマリー間を同期転送する設定が出来ますが、その構成をLKCLIでエクスポートした場合に別のクラスター構成にインポートすることはできません。<br><br>**注記:** この構成は、lkcliのエクスポート/インポート機能ではサポートされていません。lkcliのエクスポートはエラーを返しませんが、lkcliのインポートは失敗します。 |
| netraid ミラーに一意の識別子がないという警告メッセージ | 一意の識別子を使用するには、設定をできるだけ早く変更する必要があります。<br><br>修復するための推奨手順は次の通りです。<br>1. すべてのノードで LifeKeeper を起動します。<br><br>2. 安全でない netraid リソースと、各ノードの各リソースの元になるディスクを特定します。各ノードでマッピングが異なる場合があるため、これは各ノードで実施してください。 |

```
# ins_list -r netraid -f: | grep DEVNAME | grep -v mapper | cut -f4,5 -d:
datarep-test1:/dev/xvdb
datarep-test2:/dev/xvdc
datarep-filesys3:/dev/xvdd
```

**注記:** これは netraid「tag:ID」のリストです。以下の手順では、上記のデバイス名マッピングの一致を想定しています。

3. デバイスが GPT で設定されているかどうかを確認します。各デバイスについて、GPT getId を実行します。

```
#/opt/LifeKeeper/lkadm/subsys/scsi/gpt/bin/getId -i /dev/xvdb

4757cd62-e065-4013-8514-1031b446aa24
```

4. getId が一意の ID を返した場合は、インスタンスを更新します。

```
#ins_setid -t datarep-test1 -i “4757cd62-e065-4013-8514-1031b446aa24”

<GPT ではないデバイスがある場合は、ステップ5 に進んでください。>
```

5. 安全でない netraid リソースに依存するリソースを特定します。

```
# ins_list -r netraid -f: | grep DEVNAME | grep -v mapper | cut -f4 -d: | while read entry; do dep_list -p $entry -f: 2>/dev/null | cut -f1 -d:; done
/test1
/test2
filesys3
```

**注記:** これは、netraid デバイスに関連付けられたファイルシステムの **タグのリスト** です。

6. ファイルシステムのマウントポイントを特定します。通常、ファイルシステムリソースのタグはマウントポイントと同じです。同じでない場合は次のコマンドを使用して、ファイルシステムタグをファイルシステムのマウントポイントと一致させることができます。

```
# ins_list -t filesys3 -f: | cut -d: -f5
/test3
```

7. すべてのアクティビティーを停止して、ファイルシステムリソースのみを netraid デバイスで In Service にします。

a. すべてのリソースを Out of Service にします。
b. 安全でない netraid デバイス上のファイルシステムのみを In Service にします。

安全でないデバイスについてのみ、以下の手順に従ってください。

8. リソースが In Service で影響を受けるファイルシステム上のすべてのデータのバックアップを取得します。

9. すべてのクラスターノードですべてのリソースを Out of Service にします。

10. LifeKeeper 構成のバックアップを取得します（`lkbackup -c --cluster`）。

11. 安全でない各 netraid リソースに対して階層を削除します。

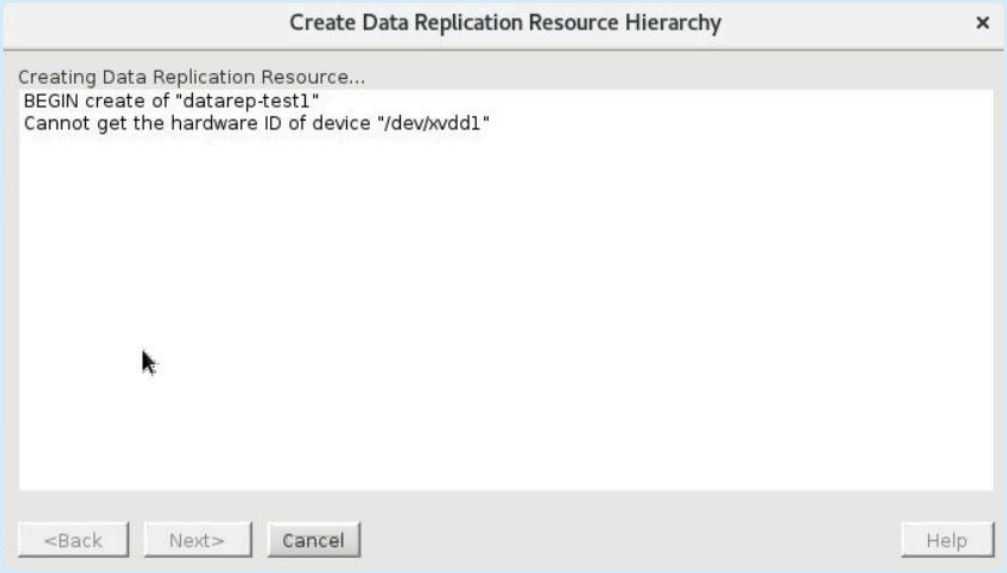
a. 階層に注意してください。影響を受けるアプリケーションは削除されます。
b. 複雑な構成では、階層が複数のブランチで構成されている複数の階層を削除する必要がある場合があります。

12. この時点で残っているのは、安全でない netraid リソースとの依存関係がないリソースのみです。ほとんどの場合、それは IP リソース、EC2 リソースなどです。
13. すべてのノードで `lkstop` を実行します。
14. 影響を受けるすべてのファイルシステムがアンマウントされていることを確認します。
15. 各ノードのデバイスを GPT パーティションテーブル（gdisk や parted などを使用）で再構成するか、LVM を使用します。
16. すべてのノードで LifeKeeper を起動します。
17. ファイルシステムごとにレプリケートされた新しいファイルシステムを作成し、各リソースをすべてのノードに拡張します。

- /dev/xvdb1 -> /test1
- /dev/xvdc1 -> /test2
- /dev/xvdd1 -> /test3

| | |
|---|---|
| | 18. バックアップから各マウントポイントにデータをリストアします。データは自動的にターゲットに再同期されます。<br>19. ステップ9で削除したアプリケーション階層を再作成します。<br><br>DataKeeper ストレージ構成オプションの詳細については、SIOS製品ドキュメント を参照してください。 |
| DataKeeper リソースを削除した後に NetRAID デバイスが削除されない。 | NetRAID デバイスがマウントされている場合、DataKeeper リソースを削除しても NetRAID デバイスは削除されません。手動でデバイスをアンマウントした後、以下のコマンドを使用してNetRAIDデバイスを削除することができます。<br><br>*mdadm -S <md_device> (<md_device> を調べるには cat /proc/mdstat)* |
| フェイルオーバー中のエラー | デバイスのステータスを確認してください。再同期が進行中の場合、フェイルオーバーは実行できません。 |
| プライマリーサーバーに障害が発生すると、セカンダリーサーバーの DataKeeper リソースが ISP になります。ただし、プライマリーサーバーが再起動すると、両方のサーバーで DataKeeper リソースが OSF になります。 | DataKeeper リソース階層の作成時に選択した「スイッチバックタイプ」を確認してください。このリリースでは、DataKeeper リソースの自動スイッチバックはサポートされていません。リソースプロパティーのウィンドウで、スイッチバックタイプを [Intelligent] に変更できます。 |
| 両方のサーバーが動作不能になってからプライマリーサーバーが再起動したときに、リソースを ISP にすることができない。 | セカンダリーサーバーよりも前にプライマリーサーバーが動作可能になった場合、DataKeeper リソースを強制的にオンラインにすることができます。このためには、リソースプロパティーのダイアログを開き、 **[Replication Status]** タブ、 **[Actions]** ボタンを順にクリックし、次に **[Force Mirror Online]** を選択してください。 **[Continue]** をクリックして確認してから、 **[Finish]** をクリックしてください。 |
| 現在マウントしている NFS ファイルシステムに DataKeeper 階層を作成するときのエラー | 現在 NFS がエクスポートしたファイルシステムに、DataKeeper 階層を作成しようとしています。エクスポートする前に、このファイルシステムをレプリケーションする必要があります。 |

| | |
|---|---|
| DataKeeper リソース作成時の GUI のウイザードに、新しく作成したパーティションがリストされない。 | Linux OS は、システムを次回再起動するまで、新しく作成したパーティションを認識しないことがあります。新しく作成したパーティションのエントリーを調べるには、/proc/partitions ファイルを表示してください。新しく作成したパーティションがこのファイルに表示されない場合、システムを再起動する必要があります。 |
| プライマリーとバックアップの両方のサーバーで、リソースが緑（ISP）で表示される。 | これは、「スプリットブレイン」のシナリオで、一時的な通信障害により発生することがあります。通信の再開後、両方のシステムがそれぞれ、それ自体をプライマリーと見なします。<br><br>いずれのシステムが最終のプライマリーシステムであったかが不明なので、DataKeeper はデータを再同期しません。手動操作が必要です。<br><br>**ビットマップを使用しない場合**:<br><br>最終のバックアップであったサーバーを特定し、そのサーバーのリソースをサービス休止にする必要があります。その後、DataKeeper が全体の再同期を実行します。<br><br>**ビットマップを使用する場合**:<br><br>最終のバックアップであったサーバーを特定し、そのサーバーのリソースをサービス休止にする必要があります。その後、DataKeeper が部分的な再同期を実行します。 |
| Core – 言語環境の影響 | LifeKeeper の一部のスクリプトは Linux のシステムユーティリティーの出力を解析し、情報を抽出するために特定のパターンに依存します。英語圏以外のロケールで一部のコマンドが実行されている場合、予測されたパターンは変更され、LifeKeeper スクリプトは必要な情報の取得に失敗します。このため、 */etc/default/LifeKeeper* では、言語環境変数 LC_MESSAGES が POSIX「C」のロケール (LC_MESSAGES=C) に設定されています。言語を英語にして Linux をインストールする必要はありません (インストールメディアで使用できる任意の言語を選択可能)。 */etc/default/LifeKeeper* の LC_MESSAGES の設定は LifeKeeper にのみ影響します。 */etc/default/LifeKeeper* の LC_MESSAGES の値を変更する場合は、LifeKeeper の動作に悪影響を及ぼす可能性があることに注意してください。悪影響は、メッセージカタログがさまざまな言語とユーティリティーに対応してインストールされているかどうか、および LifeKeeper が予期していないテキスト出力をそれらが生成するかどうかに左右されます。 |
| GUI – GUI の終了後に Web ブラウザから再接続したときに、GUI のログインプロンプトが再表示されないことがある。 | GUI アプレットを終了するか切断してから、同じ Web ブラウザのセッションから再接続しようとすると、ログインプロンプトが表示されないことがあります。<br><br>**回避策:** Web ブラウザを閉じ、Web ブラウザを開き直してからサーバーに接続します。Firefox ブラウザを使用している場合は、Firefox のウィンドウをすべて閉じてから、開き直します。 |
| DataKeeper リソースの作成 (およ | 特定の環境（IDE ディスクエミュレーションを使用した仮想化環境、HP CCISS ストレージを使用したサーバー、ソリッドステートデバイス(SSD)、Amazon EBS ストレージ |

| | |
|---|---|
| び拡張) に失敗する | など）で DataKeeper を使用している場合、ミラー作成時にエラーが発生することがあります。<br>  <br>これは、LifeKeeper が問題のディスクを認識せず、デバイスに関連付ける一意の ID を取得できないためです。<br><br>**回避策:** GUID パーティションを作成し、一意の ID をパーティションに割り当てるか、LVM を使用してください。 |
| アップグレード後、ターゲットの状態が「Out of Sync」になる | LifeKeeper 9.2.2 以降では NU デバイスの使用を推奨していないため、NU デバイスで構成された DataKeeper リソースが存在する環境では、このような症状が発生します。NU デバイスを使用した環境で LifeKeeper 9.2.2 以降にアップグレードする場合には、/etc/default/LifeKeeper に以下の設定を追加してください。<br><br>`LKDR_ALLOW_NU=TRUE`<br><br>**NUデバイスの確認方法:**<br><br>lcdstatus コマンドを実行し、ID に NU- で始まる文字列が含まれる場合は NU デバイスを使用した環境です。 |
| ターゲットに拡張しても、非同期または同期を設定するための「Replication Type」の入力画面が表示されない。 | ミラーが作成されると、[Enable Asynchronous Replication（非同期レプリケーションを有効にする）] で [no] が選択されました。ミラーを削除し、[Enable Asynchronous Replication（非同期レプリケーションを有効にする）] プロンプトが表示されたら [yes] を選択してください。 |
| レプリケーションタイプが同期ではなく非同期に | ミラーの拡張を解除して再度拡張し、Replication Type を要求されたら「synchronous」を選択してください。 |

| | |
|---|---|
| なっている。2つのシステム間のレプリケーションは、当初非同期レプリケーション用に構成されていたが、代わりに同期レプリケーションを要求される場合。 | |
| ターゲットが前のソースを待機中で、同期していない。 | 前のソースをクラスターに接続します。前のソースがタイムリーにクラスターに再参加できない場合は、現在のミラーソースで `$LKROOT/bin/mirror_action <tag> fullresync <source> <target>` コマンドを実行し、ターゲットを完全再同期で再接続してください。 |

# 5.6. コマンドラインインターフェース

## 導入

SIOS Protection Suite のコマンドラインインターフェースは、グラフィカルユーザインターフェースの代わりに使用できます。shell スクリプトに CLI の呼び出しを組み込んで、頻繁に使用する管理タスクを自動化することができます。

## ドキュメントの内容

このガイドでは次の事項について説明します。

- コマンド. CLI コマンドについて説明します。
- shell スクリプトの例. スクリプトで使用する CLI の例をいくつか示します。

## SIOS Protection Suite Documentation

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- SIOS Protection Suite インストールガイド
- オプションの Recovery Kit ドキュメンテーション

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

# shell スクリプトの例

## 階層を作成して拡張する shell スクリプトの例 (抜粋)

2 つの階層間の依存関係を作成する方法の例も示します。

### # システムパラメータが必要

```
LKROOT=/opt/LifeKeeper

OBJ_DIR=/opt/LifeKeeper/lkadm

LKBIN=/opt/LifeKeeper/bin

ExtendPath=/opt/LifeKeeper/lkadm/bin

PATH=$PATH:$LKBIN
```

### # 汎用の引数

```
LocalServer=unix121.ha.uk.sbphrd.com

TargetServer=unix122.ha.uk.sbphrd.com

Node2Priority=10
```

### # 上記の変数は、後続のコマンドで使用される

# 5.6.1. コマンド

## LifeKeeper の bin コマンドのグループ化と基本的な説明

コマンドは、以下のディレクトリにあります。 */opt/LifeKeeper/bin*

これを独自のパスに入れるには、以下のコマンドを実行します。

```
. /etc/default/LifeKeeper
```

### LifeKeeper、GUI などの開始 / 停止

`lkstart` : LifeKeeper Core を起動します。

オプション:

なし

`lkstop` : LifeKeeper Coreを停止します。オプションにより停止時の動作が変わります。

> **注意:** リソースの停止が失敗した場合でも `lkstop` の **プロセスは続行** され、lkstopが失敗することは **ありません** 。

> **注意:** `lkstop` の実行および/または複数のクラスターノードのシャットダウンを平行して行わないでください。

> **注意:** クラスター内の処理を完了させるため各 `lkstop` コマンドを発行する際には間隔を最低2分空けることを推奨します。

オプション:

なし – LifeKeeper Core及び保護対象サービスを停止します。シャットダウンステラトジーの設定に関わらず待機系へのスイッチオーバーは行われません。

-r – システム起動時の自動起動を有効のままにします。

-n – LifeKeeper Core及び保護対象サービスを停止します。 シャットダウンステラトジーで"Switch Over Resources"に設定されている場合は待機系にスイッチオーバーされます。また、/etc/default/LifeKeeper で LKSTOP_FAILOVER_N=1 が設定されている場合は、shutdown_switchover フラグの有無にかかわらずフェイルオーバーが発生します。

-f – LifeKeeper Coreのみ停止します。保護対象のサービスは停止しません。

-i – LifeKeeper Coreのみ停止します。保護対象のリソースは停止しません。続行するため (yes/no) の確認を行ってください。

`lkGUIserver` : LifeKeeper GUI デーモンプロセスを開始 / 停止します。

オプション:

`start`

`stop`

`restart`

`lkGUIapp` : LifeKeeper Java アプリケーションを開始します。

オプション:

なし

## LK とその他の LifeKeeper のコマンドの監視

`lcdstatus` : LifeKeeper のリソース、コミュニケーションパスなどの状態を表示します。

オプション:

`-d - <node to run command on>`

`-q : 短いレポート`

`lcdsync` : メモリにある LifeKeeper の設定情報をディスクに書き込みます。

オプション:

`-d - <other node to run it on>`

`lcdrcp` : コミュニケーションパス経由で、ある LifeKeeper ノードからファイルを別の LifeKeeper ノードに転送します。

オプション:

`lcdrcp <file names> {dest:ofile | dest:odir}`

`lcdremexec` : 指定の LifeKeeper ノードで、指定のコマンドを実行します。

オプション:

```
-d <node to run command on> <command>
```

`lcdrecover`：リソース階層インスタンス状態の確認と設定を行います。

オプション:

ドキュメンテーションを参照してください。

## 階層を in service または out of service にする

`perform_action`: 指定リソースで指定動作を実行します。

指定の階層から別のノードへの切り替えに使用できます。

オプション:

```
-a <action name>
```

```
-t <tag name>
```

例:

`perform_action -a restore -t $LKTag`：階層を in service にします。

`perform_action -a remove -t $LKTag`: 階層を out of service にします。

## LifeKeeperのコンフィグレーションをチェックする

lkchkconf : /etc/default/LifeKeeperに設定された値について、以下の内容をチェックします。

- LifeKeeperの起動時に読み込む設定が実際に反映されているか。 反映されていない場合には、その旨を通知するエラーメッセージを出力します。
- LifeKeeperの監視実行の間隔と、リソースの監視処理のタイムアウト値の設定値に矛盾がないか。 リソースの監視処理のタイムアウト値に、LifeKeeperの監視実行間隔（LKCHECKINTERVALの設定値）より大きい値が設定されている場合はその旨を通知するエラーメッセージが出力されます。

オプション:

なし

# 5.6.1.1. lklogmsg

このコマンドは、LifeKeeper ロギングインターフェースへのコマンドラインインターフェースを提供します。Recovery Kit の実装の一部であるスクリプトで使用できます。

## 名称

`lklogmsg` — LifeKeeper ロギングインターフェースへのコマンドラインインターフェース

## 書式

```
LKROOT/bin/lklogmsg [-o] [-e] [-h] [-p progname] -l level -s source -a action
-t tag -i messageid [-c command - cmd args ]|[-- message]
```

## 構文

```
lklogmsg [-o] [-e] [-h] [-p progname] -l level -s source -a action -t tag -i
messageid [-c command - cmd args ]|[-- message]
```

ログに記録されるメッセージは、特別な引数「--」を前に付けて最後の引数として渡されます。または、メッセージパラメーターが存在しない場合は、このコマンドは stdin から入力を読み取り、各行を個別のログメッセージに変換します。

また、 `-c` オプションでコマンドと引数が指定される場合は、追加の操作モードを使用できます。このモードでは、コマンドの `stdout` と `stderr` はパイプを介してリダイレクトされ、 `lklogmsg` によってキャプチャされます。コマンドの stdinは、 */dev/null* から取得されます。

オプション `-o` を指定すると、メッセージは `stdout` にエコーされます。

オプション `-e` を指定すると、メッセージは `stderr` にエコーされます。

オプション `-h` を指定すると、 `lklogmsg` は `nohup` と同じアクションを実行します。

`-p` オプションでは、 `lklogmsg` を呼び出しているスクリプトの名前を指定します。これは、ログに記録されるアプリケーション名として使用されます。

`-l` オプションでは、 `LK_NOTIFY` 、`LK_FATAL` 、`LK_ERROR` 、`LK_WARN` 、`LK_TRACE` 、`LK_INFO` または `LK_DEBUG` のいずれかを指定します。

`-s` オプションでは、 `lklogmsg` を呼び出しているApplication Recovery Kit 名 (mysql、lvm、oracleなど) を指定します。

`-a` オプションでは、 `lklogmsg` を呼び出している Recovery Kit のアクション (例：restore、remove、recover、create、extend) を指定します。

`-i` オプションでは、Application Recovery Kit に割り当てられた範囲の（6桁の）メッセージ ID を指定し

ます。

## 例

`/opt/LifeKeeper/bin/lklogmsg -e -l LK_ERROR -s gopher -a recover -t sample_tagname -i` 222016 Unable to restart gopher server sends “Unable to restart gopher server” to both the LifeKeeper log, and to `stderr`.

`/opt/LifeKeeper/bin/lklogmsg -l LK_INFO -s webserver -a create -t sample_sdr_resource -i` 100315

# 5.6.1.2. SYS – LifeKeeper クラスタ内のシステムに関する LifeKeeper のコマンド

sys_list：特定の LifeKeeper ノードが既知のシステムをリストします。

オプション:

```
-d <other node to run it on>
```

sys_create：LifeKeeper ノード上にある他のシステムの情報を作成します。

オプション:

```
-s <remote system name>

-d <node to run command on>
```

sys_remove: LifeKeeper ノード上にある他のシステムの情報を削除します。

オプション:

```
-d <dest>

-s <remote system name to be removed from list>
```

sys_getstate：指定の LifeKeeper ノードに、指定の LifeKeeper ノードの状態をリストします。

オプション:

```
-d <node to run command on>

-s <system concerning the state being checked>
```

sys_setstate：指定の LifeKeeper ノードに、指定の LifeKeeper ノードの状態を設定します。

オプション:

```
-d <node to run command on>

-s <system concerning the state being set>

-S <actual state> {DEAD|ALIVE|UNKNOWN}

-R <reason for state setting>
```

sys_getdescr：システムが現在の状態に変化した理由に関する一部の情報を出力します。

オプション:

```
-d <node to run command on>

-s <system to get data on>
```

# 5.6.1.3. NET – コマンドに関するコミュニケーションパス

net_create ：2 つの SPS ノード間にコミュニケーションパスを作成します。

オプション:

```
-d <node to run command on>

-s <other system>

-D <device path>

-n <TTY or TCP>

-b <baud rate>

-r <remote IP address>

-l <local IP address>

-p <priority>
```

net_remove ：2 つの SPS ノード間にあるコミュニケーションパスを削除します。

オプション:

```
-d <node to run command on>

-s <remote server name to be removed from>

-D <device path>

-r <remote IP address>
```

net_list ：指定の SPS ノードに関するコミュニケーションパスの情報をリストします。

オプション:

```
-d <node to run command on>

-f: <フィールド区切り文字「:」>

-s <system name>
```

net_change：指定のコミュニケーションパスに関する特定の情報を変更します。

オプション:

```
-d <node to run command on>

-s <server name for data to be modified>

-D <device>
```

`Crelcm` ：コミュニケーションパスを作成します。

```
/opt/LifeKeeper/lkadm/bin/crelcm <node 1> <node 2> <net type> <baud rate> <IP address 1> <IP address 2> <prio>
```

`portio` ：2 つの SPS ノード間のシリアル通信をテストします。

# 5.6.1.4. FLAG – LifeKeeper の内部フラグに関するコマンド

`flg_create` ：指定ノードに指定の LifeKeeper フラグを設定します。

オプション:

```
-d <node to run command on>

-f <flag name>
```

`flg_remove` ：指定ノードから指定の LifeKeeper フラグを削除します。

オプション:

```
-d <node to run command on>

-f <flag name>
```

`flg_list` ：指定ノードに設定された LifeKeeper のフラグをすべてリストします。

オプション:

```
-d <node to run command on>
```

# 5.6.1.5. TYP – リソース階層タイプに関する LifeKeeper のコマンド

typ_create ：指定ノードに指定のリソースタイプを作成します。

オプション:

-d <node to run command on>

-a <app type> (アプリケーションが最初に必要)

-r <resource type>

typ_remove ：指定ノードのリソースツリーから指定のリソースタイプを削除します。

オプション:

-d <node to run command on>

-a <application type>

-t <resource type>

typ_list ：指定ノード上にあるリソースタイプをすべてリストします。

オプション:

-d <node to run command on>

-f: <フィールド区切り文字「:」>

-a <app type>

# 5.6.1.6. APP – リソースアプリケーションに関する LifeKeeper のコマンド (関連タイプのグループ)

app_create ：指定ノードに指定のリソースアプリケーションを作成します。

オプション:

```
-d <node to run on>

-a <application name>
```

app_remove ：指定ノードのリソースツリーから指定のリソースアプリケーションを削除します。

オプション:

```
-d <dest>

-a <application type>
```

app_list ：指定ノード上にあるリソースアプリケーションをすべてリストします。

オプション:

```
-d <node to run on>
```

# 5.6.1.7. DEP – リソースアプリケーションを相互に関連付ける方法に関する LifeKeeper のコマンド

## コマンドの実行

- /opt/LifeKeeper/bin へ移動
- [コマンド] [オプション] の形式で実行 (例： (**./dep_list^-P**)

# コマンド

注意：コマンドの詳細については、LifeKeeperと共にインストールされているマニュアルページを参照してください。.

## dep_create – 2つのリソースインスタンスの間に依存関係を作成する

オプション/説明

**オプション:**

-p <parent tag>
-c <child tag>
-d <dest> （任意）

**説明:** 1つのシステム上で、 **<parent tag>** と **<child tag>** で指定したタグを持つリソースインスタンスの間に依存関係を作成します。

**追加情報:**

- -p および -c で指定したリソースはいずれも、引数 -d で指定したシステム上にある必要があります。これは、親リソースが適切な動作のために子リソースを必要とすることを意味します。
- -d で宛先が指定されていない場合、コマンドを実行したシステムが想定されます。
- **-p および -c で指定したリソースはいずれも、事前に作成しておく必要があります。**

注意: 依存関係は、コマンドが実行されるシステムまたは **引数 -d** 指定されたシステム上にのみ作成されます。依存関係の設定を確実に行うためには、このコマンドをすべてのクラスターノードに対して実行する必要があります。 これを行わないと、予期せぬ障害が発生する可能性があります。

## dep_remove – 2 つのリソースインスタンス間にある依存関係を削除する

オプション/説明

オプション：

-p <parent tag> (任意)
-c <child tag> (任意)
-d <dest> (任意)

**説明:** 1つのシステム上で、 <parent tag> と <child tag> で指定したタグを持つリソースインスタンス間の依存関係を削除します。

**追加情報:**

- -d で宛先が指定されていない場合、コマンドを実行したシステムが想定されます。
- -c で子リソースが指定されていない場合、-p で指定された親リソースとのすべての依存関係が削除されます。
- -p で親リソースが指定されていない場合、-c で指定された子リソースを持つすべての依存関係が削除されます。
- -p、-cのいずれか、または両方を指定する必要があります。 **どちらも指定されていない** 場合、コマンドはエラーとなります。

注意: 依存関係は、コマンドを実行したシステムまたは **引数 -d** で指定されたシステムでのみ削除されます。したがって、依存関係を完全に削除するには、すべてのクラスターノードに対してコマンドを実行する必要があります。 これを行わないと、予期せぬ障害が発生する可能性があります。

## dep_list – 2 つのインスタンス間にある依存関係をリストする

オプション/説明

オプション:

-p <ofchildtag> (任意)
-c <ofparenttag> (任意)
-C <allchild> (任意)
-P <allparent> (任意)
-r <typ> (任意)
-a <app> (任意)
-d <dest> (任意)
-f <field separator of ':'> (任意)

**説明:** リソースインスタンス間の依存関係に関する情報を、標準出力に出力します。 **注意:** -d で宛先が指定されていない場合、コマンドを実行した現在のシステムが想定されます。

このコマンドを実行すると、各文字列が次の形式で表示されます。
**Parent =** _____ **Child =** _____

```
varfs:part10
part10:ha5
usrfs:part20
part20:ha5
```

各文字列には、 **区切り文字** で区切られた2つのフィールドがあります。dep_list コマンドの後に -f <任意の文字> を使用して、文字列内の2つのリソースインスタンスを区切る区切り文字を設定します。上記の例では、区切り文字としてコロン (:) を使用しています。区切り文字より左側フィールドは依存関係のあるリソースのうち親リソースのタグ名、右側のフィールドは子リソースのタグ名を示しています。

例:

```
[ bin]# ./dep_list -f:
SAP-PDX_ERS10:ers-ip-12.1.2.0
ers-ip-12.1.2.0:ec2-12.1.2.0
SAP-PDX_ASCS00:ascs-ip-12.1.1.0
ascs-ip-12.1.1.0:ec2-12.1.1.0
```

**注意**: -f オプションを使用しない場合でも、出力には control->A の区切り文字が含まれます。 (*これは印刷不可能な文字で、文字列には表示されません。*)

**追加情報:**

- -p オプションが指定されている場合、<ofchildtag>で指定されたリソースの直接の親の依存関係のみを出力します。
- -c オプションが指定されている場合、<ofparenttag> で指定されたリソースの直接の子の依存関係のみを出力します。
- -C オプションが指定されている場合、 <allchild> で指定されたリソースのすべての直接および間接の子の依存関係を出力します。
- -P オプションが指定されている場合、<allparent> で指定されたリソースのすべての直接および間接の親の依存関係を出力します。
- -p、-c、-P、または -Cオプションがいずれも指定されていない場合、すべての依存関係が出力されます。
- -r オプションを指定すると、子の typ のすべての依存関係がリストされます。
- -a オプションを指定すると、アプリケーションの app のすべての依存関係がリストされます。

## eqv_create – 2 つのノード間に指定リソースのイクイバレンシーを作成する

オプション/説明

**オプション:**

```
-t <tag>
-o <tag on equivalent system>
-S <equivalent system>
-e <{COMMON|SHARED|COMPOSITE}>
-p <Priority> (任意)
```

```
-d <dest> (任意)
-r <Priority on equivalent system> (任意)
```

**説明:** ローカルシステム（-d で宛先が指定されていない場合）の <tag> で指定されたリソースと <equivalent system> 上の <tag on equivalent system> で指定した リソースとの間のイクイバレンシーを、構成データベース内に作成します。

**追加情報:**

- -d で宛先が指定されている場合、コマンドは <dest> システムで実行されます。指定されていない場合、コマンドはローカルシステムで実行されます。-d の使用がどのように影響するかについては、以下の例を参照してください。
- -p の引数 <Priority> および -r の引数 <Priority on equivalent system> は、イクイバレンシー情報を持つシステムのリソースの優先順位を表します。
- <Priority> および <Priority on equivalent system> オプションは、指定されていない場合、それぞれデフォルトで 1、2 になります。
- -S <equivalent system> で指定されたリモートシステムにイクイバレンシーを自動的に追加します。
- -e で指定するイクイバレンシータイプは、現在 **SHARED**のみ がサポートされます。

この例では、次のように仮定しています。:
NodeAのリソース： **TagOnNodeA**
NodeBのリソース： **TagonNodeB**

**例:**
**TagOnNodeA** と **TagOnNodeB** の間にイクイバレンシーを作成するには、以下を実行します（両方ともNodeAで実行します）。

- `eqv_create -t TagOnNodeA -p 1 -S NodeB -o TagOnNodeB -r 10 -e SHARED`

または

- `eqv_create -d NodeB -t TagOnNodeB -p 10 -S NodeA -o TagOnNodeA -r 1 -e SHARED`

## eqv_remove – 2 つのノード間にある指定リソースのイクイバレンシーを削除する

オプション/説明

```
オプション:

-S <equivalent system>
-t <tag>
-e <{COMMON|SHARED|COMPOSITE}>
-d <dest> (任意)
-o <tag on equivalent system> (任意)
```

**説明:** ローカルシステム（-d で宛先が指定されていない場合）の <tag> で指定されたリソースと <equivalent system> 上の <tag on equivalent system> で指定したリソースとの間のイクイバレンシー

を、構成データベースから削除します。

**追加情報:**

- -d で宛先が指定されている場合、コマンドは <dest> システムで実行されます。指定されていない場合、コマンドはローカルシステムで実行されます。-d の使用がどのように影響するかについては、以下の例を参照してください。
- -d で宛先が指定されていない場合、 <tag> はローカルノードのリソースタグを表します。指定されている場合、 <tag> は <dest> のリソースタグです。
- -e で指定するイクイバレンシータイプは、現在 **SHARED のみ** がサポートされます。
- -S <equivalent system> で指定されたリモートシステムのイクイバレンシーを自動的に削除します。

この例では、次のように仮定しています。:
NodeAのリソース： **TagOnNodeA**
NodeBのリソース： **TagonNodeB**

**例:**
**TagOnNodeA** と **TagOnNodeB** の間のイクイバレンシーを削除するには、以下を実行します（両方ともNodeAで実行します）。

- `eqv_remove -t TagOnNodeA -S NodeB -e SHARED`

または

- `eqv_remove -d NodeB -t TagOnNodeB -S NodeA -e SHARED`

## eqv_list - リソースインスタンス間にあるイクイバレンシー関係をリストする

オプション/説明

オプション:

-d <dest> (任意)
-S <system> (任意)
-t <tag name> (任意)
-f: <field separator of ':'> (任意)

**説明:**
リソースインスタンス間のイクイバレンシー関係に関する情報を、標準出力 に出力します。

**追加情報:**

- -dで宛先が指定されている場合、イクイバレンシーのリストはそのシステムからのリストになり、指定されていない場合は、ローカルシステムからのリストになります。
- -S で システムが指定されている場合、イクイバレンシーのリストは、ローカルシステム間に存在する任意のイクイバレンシーに対するものになります。

  **例:**

- iwstp:varfs:remote:varfs_backup:SHARED:1:2
- iwstp:usrfs:remote:usrfs_backup:SHARED:1:2

上記の例の各行には、 **区切り文字** で区切られた各文字列のフィールドが含まれています。 eqv_list コマンドの後に-f <任意の文字> を使用して、出力される文字列のフィールドを区切る区切り文字を設定します。上記の例では、区切り文字としてコロン (:) を使用しています。フィールドは以下のとおりです（フィールド1、2、6はローカルシステムの情報、フィールド3、4、7はリモートシステムに関する情報、フィールド5はイクイバレンシーのタイプです）。:

1. イクイバレンシーのリソースタグ1が存在するローカルシステム名: ***iwstp***
2. リソースタグ1のタグ名: ***varfs***
3. イクイバレンシーのリソースタグ2が存在するリモートシステム名: ***remote***
4. リソースタグ2のタグ名: ***varfs backup***
5. イクイバレンシーのタイプ: ***SHARED.***
6. ローカルシステム/リソースの優先順位: ***1***
7. リモートシステム/リソースの優先順位: ***2***

この関数の残りの引数は、以下のように、出力される情報を制限します。

8. -e SHARED プションは、すべての SHARED イクイバレンシー情報を出力します。
9. -t オプションは、引数で指定されたタグに関連するイクイバレンシーのみを含むように出力を制限します。

## hry_setpri - 指定のノードまたはそのノード上にある階層の優先順位を設定する

オプション/説明

**オプション:**

```
-t <tag(s)>
-p <priority>
-d <dest> (任意)
-q <Details in Description> (任意)
-l: <Details in Description> (任意)
```

**説明:** この関数は、指定された <tag(s)> リソース階層内のリソースインスタンスを <priority> に設定します。 -d で宛先が指定されていない場合、コマンドを実行した現在のシステムが想定されます。

**追加情報:**

- 関連するすべてのイクイバレンシーが更新されます。
- <priority> は、既存のイクイバレンシーによって使用中であってはなりません。
- 階層は、-t オプションで指定されたルートリソース <tag(s)> によって識別されます。共通のリソースを持つ複数のルート階層が存在する場合、すべてのルートは、「-t H1,H2」など、 カンマ(スペースは使用できません）で区切られたタグのリストで指定する必要があります。
- 新しい優先順位は、-p オプションを使用して指定します。

- -q オプションを指定した場合、標準出力は すべて抑制されます。
- -l オプションを指定すると、この階層が存在するシステムのリストと、それに関連する優先順位が表示されます。例：

```
teak 10
ash 20
plum 30
```

**注意:** 1つのシステム上にしか存在しないリソース階層の優先順位を設定することはできません。 `hry_setpri` コマンドは、プライマリーシステムを超えて拡張されていないリソースが見つかるたびに、警告メッセージを発行します。

# 5.6.1.8. INS – 個々の LifeKeeper 階層インスタンスに関するコマンド

ins_list ：指定のリソース階層インスタンスについて現在の情報をリストします。

オプション:

-d <dest>

-f:<フィールド区切り文字「:」>

-a / -r / -t / -i は、オプションのアプリケーション、タイプ、タグ、および ID の情報を指定します。

ins_setas ：指定階層の自動スイッチバックストラテジーを設定します。

オプション:

-d <dest>

-t <tag name>

-s <switchback typ> {INTELLIGENT|AUTOMATIC}

ins_setinit ：LifeKeeper の起動時に指定リソースを初期化する方法を定義します。

オプション:

-d <dest>

-t <tag name>

-I <init state> {AUTORES_ISP|INIT_ISP| INIT_OSU}

ins_setinfo：指定のリソース階層について、情報文字列を定義します。

オプション:

-d <dest>

-t <tag name>

-v <string of information>

ins_setstate ：指定ノード上にある指定リソース階層の状態を設定します。

オプション:

```
-d <dest>

-t <tag name>

-S <state to set instance> {ISP|ISU|OSU}

-R <reason for state setting>

-A <recursively set all resources that depend on this one>
```

ins_gettag：関連する ID のタグ名をリストします。

オプション:

```
-i <id>
```

# 5.6.1.8.1. 階層の拡張解除

```
/opt/LifeKeeper/lkadm/bin/unextmgr <Node_Name> <Tag_Name>
```

# 5.6.1.9. マニュアルページへのアクセス

SIOS Protection Suite for Linux のインストールには、マニュアルページが含まれています。これは、**lkMAN package** とともに提供されます（完全なパッケージ名については、以下を参照してください）。

```
steeleye-lkMAN-9.5.1-7154.noarch
```

## LifeKeeper コマンドのマニュアルページを表示するには、以下の手順に従ってください。

1. OS に SIOS Protection Suite for Linux がインストールされていることを **確認** します。(SIOS Protection Suite for Linux のインストールについては、 こちら をクリックしてください。)
2. コマンドプロンプトから、次のコマンドを **実行** します。

```
rpm -qa | grep steel
```

**注意:** これにより、インストールされているすべての LK rpm パッケージが表示されます（以下を参照）。

3. **lkMAN** パッケージがインストールされていることを確認したら 、使用している OS に manpage rpms がインストールされていることを確認します。

4. 以下のコマンドを **実行** して、MANPATH を設定します。

```
[ec2-user@use1pdxcs01 ~]$ MANPATH=/opt/Lifekeeper/man
[ec2-user@use1pdxcs01 ~]$ export MANPATH
[ec2-user@use1pdxcs01 ~]$ echo $MANPATH
/opt/Lifekeeper/man
```

> **注意:** MANPATH は、再起動した後毎回設定する必要がある場合があります。

これで、マニュアルページにアクセスできます。

## マニュアルのあるコマンド

特定のコマンドのマニュアルは、以下に示すように /opt/LifeKeeper/bin にあります。

```
[root@use1pdxcs01 bin]# ls
a2p                     lcdmachfail             openssl
app_create              lcdrcp                  pcregrep
app_list                lcdrecover              pcretest
app_remove              lcdremexec              perform_action
backupadm               lcdreserve              perl
c2ph                    lcdrmipc                perl5.8.8
can_talk                lcdstatus               perlbug
certtool                lcdsync                 perlcc
checkrklic              lcduname                perldoc
cpan                    lcdwait                 perlivp
create_terminal_leaf    lcm                     piconv
credstore               libnetcfg               pl2pm
c_rehash                lkbackup                pod2html
cre_mbox                lkcheck                 pod2latex
curl                    lk_chg_value            pod2man
delallsys               lkchkconf               pod2text
dep_create              lkcli                   pod2usage
dep_list                lk_configsnmp           podchecker
dep_remove              lk_configsnmp           podselect
dprofpp                 lk_confirmso            portio
empty_mbox              lkcore_parameter        prove
enc2xs                  lkdefault               psed
eqv_create              lkdiskmon               psktool
eqv_list                lkdisktest              pstruct
eqv_remove              lkexterrlog             qwk_storage_exit
eventslcm               lkGUIapp                qwk_storage_init
find2perl               lkGUIserver             rcv
flg_create              lkID                    rcv_msg
```

## マニュアルページにアクセスするためのコマンド

上記のコマンドのマニュアルページを表示するには、次のコマンドを実行します。

- man <command>

例:

```
# man dep_list
```

## トラブルシューティング

> ! 手順１～３を実施済で以下のエラーとなる場合、MANPATHの設定が不足している可能性があります。（手順４以降を実施してください）

```
[ec2-user@use1pdxcs01 ~]$ man dep_list
No manual entry for dep_list
```

# 5.6.2. LKCLI（LifeKeeper コマンドラインインターフェース）

LKCLI は、LifeKeeper の GUI で操作可能な機能をコマンドラインインターフェースで実現しています。また、GUI には無い機能として、コミュニケーションパスおよびリソース情報の export/import 機能があります。export/import 機能は、作成したシステムの量産や、テスト環境から本番環境のデプロイを容易に行える手段となります。

## サポート対象

次に示す環境での使用をサポート対象としています。

- ノード数： 2ノード構成のクラスタ環境のみサポートされます。
- OS：LifeKeeper 本体でサポートしている OS はすべてサポートされます。
- コミュニケーションパス：ノード間が TCP/IP で接続されている環境のみサポートされます。
- アプリケーションリカバリーキット： ARK 一覧 に記載されたアプリケーションリカバリーキットのみで構成された環境のみサポートされます。ノード内にサポート対象外の ARK リソースが作成されている場合、その環境はサポート対象外となります。

サポート対象外の環境でのコマンドライン操作をご検討されているお客様は、「 コマンドラインインターフェース 」の使用をご検討ください。

## root 以外のユーザーで実行する方法

LKCLI を実行するには、ユーザーは LifeKeeper の次の 3 つグループのいずれかのメンバーである必要があります：lkadmin、lkoper または lkguest。これらのグループについては こちら を参照ください。
root 以外のユーザーで LKCLI を実行する場合は、sudo コマンドを使用して実行してください。なお、lkcli コマンドは絶対パスで指定してください。コマンド実行時に sudo コマンドを実行しているユーザーの認証が求められます。以下、LifeKeeper 起動の例です。

```
$ sudo /opt/LifeKeeper/bin/lkcli start
```

LifeKeeper の sudo 設定は /etc/sudoers.d/lifekeeper で行っています。もし /etc/sudoers で LifeKeeper の設定がインクルードされていない場合は、 "#includedir /etc/sudoers.d" もしくは "#include /etc/sudoers.d/lifekeeper" を /etc/sudoers に追記してください。その他にも /etc/sudoers で secure_path オプションを設定することでパスを指定したり、認証の設定を変更することができます。詳細は man sudoers(5) を参照してください。

## 制限事項

- ホスト名を正しく設定できていない場合、 GUI の起動やログインができません。このとき一部の

lkcli コマンドは使用できますが、多くの lkcli コマンドの処理が停止します。ホスト名が正しく設定されていることを確認してください。

## コマンド一覧

| | |
|---|---|
| lkcli license | ライセンスキーのインストール |
| lkcli start | LifeKeeper の起動 |
| lkcli stop | LifeKeeper の停止 |
| lkcli import | コミュニケーションパス、リソース作成 |
| lkcli export | コミュニケーションパス、リソース情報の出力 |
| lkcli clean | コミュニケーションパスとリソースの削除 |
| lkcli commpath | コミュニケーションパスの操作 |
| lkcli dependency | 依存関係の操作 |
| lkcli resource | リソースの操作 |
| lkcli status | リソース状態の一覧表示 |
| lkcli log | LifeKeeper のログ表示 |
| lkcli server | サーバーの設定･操作 |
| lkcli mirror | DataKeeper のミラーリング操作 |
| lkcli esxi | VMDK RK で利用する ESXi ホストの設定 |

---

### すべてのコマンドで利用できる共通オプション

| オプション | デフォルト | 説明 |
|---|---|---|
| [--remote <str>] | | コマンドを実行したいマシンのホスト名。<br>このオプションを指定しない場合は、ローカルマシン上で実行されます。<br>リモートでコマンドを実行するには、事前に実行したいマシンとコミュニケーションパスが双方向に接続されている必要があります。<br>なお、start、stop、commpath および clean コマンドはサポートされません。 |

注記: オプション表の [ ] は、任意のオプションであることを表しています。

---

## lkcli license

LifeKeeper のライセンスを登録します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --file <str> | | ライセンスファイルのパス |

---

## lkcli start

LifeKeeper を起動します。

| 権限 | lkadmin |
|---|---|

---

## lkcli stop

LifeKeeper を停止します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| [-f] | | LifeKeeper daemon のみを停止し、保護しているサービスは停止しません。 |
| [-r] | | LifeKeeper daemon の自動起動の設定を変更せずに LifeKeeper を停止します。 |
| [-n] | | LifeKeeperの停止と共に、フェイルオーバーが実行されます。<br>このオプションは -f, -i, -r とは併用できません。 |
| [-i] | | LifeKeeper daemon のみを停止し、保護対象のサービスは停止しません。停止処理を続行するために「yes/no」の確認を行う必要があります。 |

---

## lkcli import

LifeKeeper の設定をファイルから読み込み、コミュニケーションパス、リソースを作成します。

### lkcli import commpath

LifeKeeper の設定をファイルから読み込み、コミュニケーションパスを作成します。
双方向にコミュニケーションパスを接続するには、ローカルマシン・リモートマシンの双方でこのコマンドを実行し、コミュニケーションパスを作成する必要があります。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --file <str> | | LifeKeeperの設定ファイルのパス。<br>`lkcli export` で出力を保存したファイル (YAML 形式) からコミュニケーションパスを作成します。<br>作成できるコミュニケーションパスのプロトコルは、TCP/IP (ソケット) です。 |

### lkcli import resource

LifeKeeper の設定をファイルから読み込み、リソースを作成します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --file <str> | | LifeKeeper の設定ファイルのパス。<br>`lkcli export` で出力を保存したファイル (YAML 形式) からリソースを作成します。<br>作成できるリソースは、 ARK 一覧 を参照してください。 |

> **注記:**
> ・ホスト名や IP アドレス、アプリケーションなどの環境が揃っていないと失敗します。
> ・失敗してもロールバックはされません。一部のみ作成される場合があります。
> ・既にリソースが存在する状態での実行は、サポート対象外です。

---

## lkcli export

LifeKeeper の設定を出力。
現在の LifeKeeper の設定が出力され、コマンドを実行したノードとコミュニケーションパスが接続されているすべてのノードが対象となります。
出力結果をファイルに YAML 形式で保存してください。

```
# lkcli export > lk_export.yml
```

> **注記:**
> ・出力できるコミュニケーションパスのプロトコルは、TCP/IP (ソケット) です。
> ・出力できるリソースは、 ARK 一覧 を参照してください。
> ・リソースの状態 (In Service、Out of Service など) は出力されません。
> ・サーバーのプロパティー値は出力されません。
> ・出力されるのは LifeKeeper 上の設定値のみで、保護するアプリケーション自体の設定は出力されません。
> ・出力した設定ファイルについては、ホスト名と IP アドレスのみ手動で編集してください。その他項目の手動での変更は、サポート対象外です。

| 権限 | lkadmin |
|---|---|

---

## lkcli clean

LifeKeeper の設定を削除。
コマンドを実行したノードのコミュニケーションパスとリソース設定を削除します。
全てのノードの LifeKeeper の設定を削除したい場合は、すべてのノードでこのコマンドを実行してください。その際、コミュニケーションパスが削除されると、その後リモートでコマンドを実行できないのでご注意ください。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --mode <str> | | **all** または **resource** を指定。<br>• **all** : 全てのコミュニケーションパスと全てのリソースを削除。<br>• **resource** : 全てのリソースを削除。 |

---

## lkcli commpath

コミュニケーションパスの操作

### lkcli commpath create

コマンドを実行したノードのコミュニケーションパスを作成。
作成されるコミュニケーションパスは、ローカルマシンからリモートマシンへのパスとなります。そのため、双方向にコミュニケーションパスを接続するには、ローカルマシン・リモートマシンの双方でこのコマンドを実行し、コミュニケーションパスを作成する必要があります。作成できるコミュニケーションパスのプロトコルは、TCP/IP (ソケット) です。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --laddr <str> | | コミュニケーションパスに設定するローカルマシン上の IP アドレス。 |
| --raddr <str> | | コミュニケーションパスに設定するリモートマシン上の IP アドレス。 |
| --dest <str> | | コミュニケーションパスに設定するリモートマシンのホスト名。 |
| [--priority <str>] | 既存のパスの最大値 +1 | コミュニケーションパスのプライオリティー。<br><br>**注記:** ローカルマシン・リモートマシンの双方で同一の値を指定してください。 |

### lkcli commpath delete

コマンドを実行したノードのコミュニケーションパスを削除。
削除されるコミュニケーションパスは、ローカルマシンからリモートマシンへのパスとなります。そのため、双方向にコミュニケーションパスを削除するには、ローカルマシン・リモートマシンの双方でこのコマンドを実行し、コミュニケーションパスを削除する必要があります。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --laddr <str> | | コミュニケーションパスに設定するローカルマシン上の IP アドレス。 |
| --raddr <str> | | コミュニケーションパスに設定するリモートマシン上の IP アドレス。 |
| --dest <str> | | コミュニケーションパスに設定するリモートマシンのホスト名。 |

---

## lkcli dependency

LifeKeeper のリソースの依存関係を作成・削除します。

### lkcli dependency create

2つのリソース間に新しく依存関係を作成します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --parent <str> | | 親となるリソースのタグ名。 |

| | | |
|---|---|---|
| --child <str> | | 子となるリソースのタグ名。 |

### lkcli dependency delete

2つのリソース間にある依存関係を削除します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --parent <str> | | 削除したい依存関係の親にあたるリソースのタグ名。 |
| --child <str> | | 削除したい依存関係の子にあたるリソースのタグ名。 |

---

## lkcli resource

LifeKeeper のリソースを操作します。

### lkcli resource create

リソースを作成します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | 作成するリソースのタグ名。<br>存在するタグ名は、作成できません。その他については、「リソースタグ名の制限」を参照してください。 |
| [--switchback <str>] | INTELLIGENT | リソースのスイッチバック設定。<br>「INTELLIGENT」か「AUTOMATIC」を指定してください。 |
| ARK 毎のオプション | | オプションは ARK によって異なります。<br>ARK 毎のオプションに関しては、「ARK 毎のサブコマンド」を参照してください。 |

### lkcli resource extend

リソースを拡張します。

> **注記:** 依存関係が設定されていても、対象のリソースのみ拡張されます。拡張はリソース毎に実施する必要があります。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | 拡張するリソースのタグ名。<br>拡張先に存在するタグ名は、拡張できません。その他については、「リソースタグ名の制限」を参照してください。 |
| --dest <str> | | リソース階層を拡張するターゲットサーバーのホスト名。 |
| [--switchback <str>] | INTELLIGENT | リソースのスイッチバック設定。<br>「INTELLIGENT」か「AUTOMATIC」を指定してください。 |
| [--template_priority <num> | 1 | 拡張元のリソース階層の優先順位。<br>優先順位は 1 ～ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります。 |
| [--target_priority <num> | 10 | 拡張先のリソース階層の優先順位。<br>優先順位は 1 ～ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります。 |
| ARK 毎のオプション | | オプションは ARK によって異なります。<br>ARK 毎のオプションに関しては、「ARK 毎のサブコマンド」を参照してください。 |

### lkcli resource config

リソースの設定を変更します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | 変更するリソースのタグ名。 |
| ARK 毎のオプション | | オプションは ARK によって異なります。<br>ARK 毎のオプションに関しては、「ARK 毎のサブコマンド」を参照してください。 |

### lkcli resource unextend

リソースの拡張を解除します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | 拡張を解除するリソースのタグ名。 |

| | | |
|---|---|---|
| --dest <str> | | リソース階層の拡張を解除するターゲットサーバーのホスト名。 |

### lkcli resource delete

リソースを削除します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | 削除するリソースのタグ名。 |

### lkcli resource restore

実行ノード上のリソース階層を In Service にします。

| 権限 | lkadmin, lkoper |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | In Service にするリソースのタグ名。 |

### lkcli resource remove

実行ノード上のリソース階層を Out of Service にします。

| 権限 | lkadmin, lkoper |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | Out of Service にするリソースのタグ名。 |

### lkcli resource info

リソースのプロパティー情報を出力します。出力内容は ARK 毎に異なります。

| 権限 | lkadmin, lkoper, lkguest |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | プロパティー情報を出力するリソースのタグ名。 |

### lkcli resource eqv

リソースのイクイバレンシ情報を出力します。

| 権限 | lkadmin, lkoper, lkguest |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | イクイバレンシ情報を出力するリソースのタグ名。 |

### lkcli resource reorder-priority

実行ノード上のリソースの優先順位を変更します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | 優先順位を変更するリソースのタグ名。 |
| --priority <num> | | 変更後の優先順位。<br>優先順位は 1 ～ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります。 |

### lkcli resource switchback

実行ノード上のリソースのスイッチバック設定を変更します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | 変更するリソースのタグ名。 |
| --switchback <str> | INTELLIGENT | リソースのスイッチバック設定。<br>「INTELLIGENT」か「AUTOMATIC」を指定してください。 |

---

## lkcli status

LifeKeeper のステータスを表示します。
出力されるステータス情報は「 ステータスの詳細表示 」をご参照ください。

| 権限 | lkadmin, lkoper, lkguest |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| [-q] | | ローカルシステムに存在する各リソースの情報を短いレポートで出力します。 |

| | | |
|---|---|---|
| [-e] | | ローカルシステムに存在する各リソースの情報を短いレポートで出力し、（次に優先度の高い）バックアップシステムをリストします。 |
| [-u] | | -q または -e オプションの出力の重複リソースエントリーを抑制します。 |
| [-r <str>] | | リソースルートタグの指定。<br>レポートを特定のリソースルートタグに制限します。 |

---

## lkcli log

LifeKeeper のログを表示します。

| 権限 | lkadmin, lkoper, lkguest |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --lines <num> | 10 | 表示するログの行数。 |

---

## lkcli server

LifeKeeper のサーバーに関する操作を行います。

### lkcli server info

サーバーのシャットダウン方法とフェイルオーバー機能を確認します。

| 権限 | lkadmin, lkoper, lkguest |
|---|---|

### lkcli server shutdown-strategy

サーバーのシャットダウン方法を設定します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --mode <str> | | **switchover** または **do_not_switchover** の値。<br>• **switchover** : LifeKeeper は、正常なシャットダウンでバックアップサーバーのリソースを起動します。<br>• **do_not_switchover** : LifeKeeper は、正常なシャットダウンではバックアップサーバーのリソースを起動しません。 |

### lkcli server confirmso

LifeKeeper クラスタのノード障害によるフェイルオーバーが発生した時、バックアップノードへの切り替えの実行に対してユーザー確認をするか設定します。
詳細については、「 [Confirm Failover] と [Block Resource Failover] の設定 」の [Confirm Failover] の項目を参照してください。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --server <str> | | フェイルオーバー先のサーバー名。 |
| --value <str> | | **enable** または **disable** の値。<br>• **enable** : フェイルオーバー時のユーザー確認を有効にします。<br>• **disable** : フェイルオーバー時にユーザー確認をしません。 |

### lkcli server block-failover

指定したシステムでのリソース障害に起因するフェイルオーバーをブロックする設定を行います。
詳細については、「 [Confirm Failover] と [Block Resource Failover] の設定 」の [Block Resource Failover] の項目を参照してください。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --server <str> | | フェイルオーバー先のサーバー名。 |
| --value <str> | | **enable** または **disable** の値。<br>• **enable** : 指定サーバーへのフェイルオーバーをブロックします。<br>• **disable** : 指定サーバーへのフェイルオーバーをブロックしません。 |

### lkcli server block-all-failovers

全サーバーでノード障害とリソース障害のいずれかの障害に起因するフェイルオーバーもブロックする設定を行います。ブロックの設定を解除する場合は、次の設定を見直してください。

- ノード障害に起因するフェイルオーバーのブロック設定
  - lkcli server confirmso
  - /etc/default/LifeKeeper の CONFIRMSODEF パラメーター
  - /etc/default/LifeKeeper の CONFIRMSOTO パラメーター
- リソース障害に起因するフェイルオーバーのブロック設定
  - lkcli server block-failover

---

## lkcli mirror

DataKeeper によるミラーリング操作を行います。

ミラーリングの詳細については、「 SIOS DataKeeper for Linux によるミラーリング 」を参照してください。

> 注記: `lkcli mirror` コマンドは DataKeeper Recovery Kit のインストールが必要になります。

| オプション | デフォルト | 説明 |
|---|---|---|
| --tag <str> | | DataKeeper リソースのタグ名。 |

### lkcli mirror status

ミラーのステータスを表示します。

| 権限 | lkadmin, lkoper, lkguest |
|---|---|

### lkcli mirror resume

ミラーを再開します。

| 権限 | lkadmin |
|---|---|

### lkcli mirror pause

ミラーを一時停止します。

| 権限 | lkadmin |
|---|---|

### lkcli mirror fullresync

全体を再同期します。

| 権限 | lkadmin |
|---|---|

### lkcli mirror force

ミラーリングを強制実行します。

> 注記: ソース – ターゲット間の同期が不完全な状態でも強制的にミラーが開始されるため、データの一貫性を保証できなくなります。不用意に実行した場合、リプリケーション領域のデータが破損する可能性があります。

| 権限 | lkadmin |
|---|---|

---

## lkcli esxi

VMDK RK を利用するときの ESXi ホストの設定を行います。
VMDK RK の詳細については「 VMDK as Shared Storage Recovery Kit 管理ガイド 」を、ESXi ホストの設定については「 ESXi ホストの登録 」や「 メンテナンス 」を参照してください。

### lkcli esxi add

ESXi ホストの情報を追加します。オプションでユーザー名とパスワードを指定しない場合は対話的にユーザー名･パスワードを入力することになります。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --host <host> | | 追加する ESXi ホスト名。 |
| [--user <user>] | | ESXi ホストにログインするときのユーザー名。 |
| [--password <password>] | | ESXi ホストにログインするときのパスワード。 |

### lkcli esxi list

登録されている ESXi ホストの一覧を表示します。

| 権限 | lkadmin, lkoper, lkguest |
|---|---|

### lkcli esxi update

登録されている ESXi ホストの情報を更新します。オプションでユーザー名とパスワードを指定しない場合は対話的にユーザー名･パスワードを入力することになります。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --host <host> | | 更新する ESXi ホスト名。 |
| [--user <user>] | | ESXi ホストにログインするときのユーザー名。 |
| [--password <password>] | | ESXi ホストにログインするときのパスワード。 |

### lkcli esxi delete

登録されている ESXi ホストの情報を削除します。

| 権限 | lkadmin |
|---|---|

| オプション | デフォルト | 説明 |
|---|---|---|
| --host <host> | | 削除する ESXi ホスト名。 |

# 5.6.2.1. ARK 毎のサブコマンド

## ARK 一覧


- Apache
- DataKeeper
- DB2
- EC2
- FileSystem
- Generic Application
- IP
- MySQL
- NFS
- Oracle
- OracleListener
- OraclePDB
- PostgreSQL
- Postfix
- QSP
- Raw
- Route53
- Samba
- SAP
- SAP HANA
- SAP MaxDB
- Sybase
- WebSphere MQ


---

## Apache

リソースの種類は `app: webserver, typ: apache` です。
このコマンドを利用する上での前提条件や詳細な動作については「Apache Recovery Kit 管理ガイド」の「LifeKeeper の設定作業」を参照してください。

### create apache

| オプション | デフォルト | 説明 |
|---|---|---|
| --root <str> | | Apache Web Server のルートディレクトリのフルパス<br>相対パスやシンボリックリンクを使用することはできません。 |
| --path <str> | | Apache Web Server デーモンのフルパス名 (ファイル名を含む) |

### extend apache

オプションなし

### config apache

コンフィグレーションする項目はありません

---

## DataKeeper

リソースの種類は `app: scsi, typ: netraid` です。
このコマンドを利用する上での前提条件や詳細な動作については「SIOS DataKeeper for Linux」の「リソースの設定作業」を参照してください。

### create dk

| オプション | デフォルト | 説明 |
|---|---|---|
| --mode <str> | | レプリケーションタイプ<br>「synchronous (同期)」または「asynchronous (非同期)」を指定します。<br>「synchronous」を指定すると、すべてのターゲットに対するレプリケーション設定が「synchronous」に限定されます。 |
| [--bitmap <str>] | /opt/LifeKeeper/bitmap_<Tag 名> | インテントログに使用するビットマップファイルのパス。ビットマップファイルは非同期レプリケーションでは必須であり、同期レプリケーションでは使用が推奨されます。詳細については、 DataKeeper リソース階層の作成 を参照してください。 |
| --hierarchy <str> | | 作成するデータレプリケーションのタイプ<br>タイプにより、オプションが異なります。<br>< new \| existing \| dronly > |

#### --hierarchy new

| オプション | デフォルト | 説明 |
|---|---|---|
| --device <str> | | ソースディスクまたはパーティション |
| --fstype <str> | | ファイルシステムタイプ<br>LifeKeeper がサポートするファイルシステムタイプのみ指定できます。 |
| --mount_point <str> | | 新しいファイルシステムの 新しいマウントポイント |
| --fstag <str> | | ファイルシステムリソースのタグ名 |

### --hierarchy existing

| オプション | デフォルト | 説明 |
|---|---|---|
| --mount_point <str> | | プライマリサーバの NetRAID デバイスにマウントするマウントポイント |
| --fstag <str> | | ファイルシステムリソースのタグ名 |

### --hierarchy dronly

| オプション | デフォルト | 説明 |
|---|---|---|
| --device <str> | | ソースディスクまたはパーティション |

## extend dk

| オプション | デフォルト | 説明 |
|---|---|---|
| --mode <str> | | レプリケーションタイプ<br>「synchronous」または「asynchronous」を指定します。 |
| --laddr | | ローカル IP アドレス |
| --raddr | | リモート IP アドレス |
| [--bitmap <str>] | /opt/LifeKeeper/bitmap_< Tag 名> | インテントログの記録に使用するビットマップファイルのパス |
| [--device <str>] | 拡張元と同一 | ソースディスクまたはパーティション |
| [--fstag <str>] | | ファイルシステムリソースのタグ名 |

## config dk

| オプション | デフォルト | 説明 |
|---|---|---|
| --resync_speed_min <num> | | 再同期速度制限 (KB/s) の最小値を設定します |
| --resync_speed_max <num> | | 再同期速度制限 (KB/s) の最大値を設定します |
| --compression_level <num> | | ネットワークの圧縮レベル (0～9) を設定します |

---

# DB2

このコマンドを利用する上での前提条件や詳細な動作については「DB2 Recovery Kit 管理ガイド」の「LifeKeeper の設定作業」を参照してください。

### create db2

| オプション | デフォルト | 説明 |
|---|---|---|
| --instance <str> | | DB2 インスタンス<br>保護する DB2 インスタンスを指定します。 |

### extend db2

オプションなし

### config db2

コンフィグレーションする項目はありません

---

## EC2

リソースの種類は app: comm, typ: ec2 です。
このコマンドを利用する上での前提条件や詳細な動作については「Recovery Kit for EC2 管理ガイド」の「設定」を参照してください。

### create ec2

| オプション | デフォルト | 説明 |
|---|---|---|
| --type <str> | | 作成する EC2 リソースの種類の指定します。<br>ルートテーブルシナリオを選択する場合は “RouteTable”、 Elastic IP シナリオを選択する場合は ”ElasticIP ”を指定します。 |

#### --type RouteTable

| オプション | デフォルト | 説明 |
|---|---|---|
| --ip_resource <str> | | 事前に作成した IP リソースのタグ名を指定します。 |

#### --type ElasticIP

| オプション | デフォルト | 説明 |
|---|---|---|
| --eip <str> | | 保護したい Elastic IP の IP アドレス。 |
| --dev <str> | | EIP がアタッチされたネットワークインターフェース名。 |

### extend ec2

オプションなし

### config ec2

コンフィグレーションする項目はありません

---

## FileSystem

リソースの種類は app: gen, typ: filesys です。
このコマンドを利用する上での前提条件や詳細な動作については「ファイルシステムリソース階層の作成」や「ファイルシステムリソース階層の拡張」を参照してください。

### create fs

| オプション | デフォルト | 説明 |
|---|---|---|
| --mountpoint <str> | | ファイルシステムのマウントポイントを指定。 |

### extend fs

| オプション | デフォルト | 説明 |
|---|---|---|
| [--mountpoint <str>] | 拡張元と同一 | ファイルシステムのマウントポイントを指定。 |

### config fs

| オプション | デフォルト | 説明 |
|---|---|---|
| --mountopt <str> | | ファイルシステムのマウントオプションを指定。 |

---

## Generic Application

リソースの種類は app: gen, typ: app です。
このコマンドを利用する上での前提条件や詳細な動作については「Generic Application リソース階層の作成」や「Generic Application リソース階層の拡張」を参照してください。

### create gen

| オプション | デフォルト | 説明 |
|---|---|---|
| --restore <str> | | restore スクリプトのパスを指定。 |
| --remove <str> | | remove スクリプトのパスを指定。 |
| [--quickCheck <str>] | | quickCheck スクリプトのパスを指定。 |
| [--recover <str>] | | recover スクリプトのパスを指定。 |

| | | |
|---|---|---|
| [--appinfo <str>] | | アプリケーションに関するオプション情報を指定。 |

### extend gen

| オプション | デフォルト | 説明 |
|---|---|---|
| [--appinfo <str>] | 拡張元と同一 | アプリケーションに関するオプション情報を指定。 |

### config gen

| オプション | デフォルト | 説明 |
|---|---|---|
| [--restore <str>] | | 更新する restore スクリプトのパスを指定。 |
| [--remove <str>] | | 更新する remove スクリプトのパスを指定。 |
| [--quickCheck <str>] | | 更新する quickCheck スクリプトのパスを指定。 |
| [--recover <str>] | | 更新する recover スクリプトのパスを指定。 |
| [--all <str>] | No | **Yes** または **No** を指定。<br>• **Yes**：全てのクラスターノードのスクリプトを更新。<br>• **No**：コマンド実行ノードのスクリプトを更新。 |

---

## IP

リソースの種類は `app: comm, typ: ip` です。
このコマンドを利用する上での前提条件や詳細な動作については「IP Recovery Kit 管理ガイド」の「設定」を参照してください。

### create ip

| オプション | デフォルト | 説明 |
|---|---|---|
| --ipaddr <str> | | 仮想 IP アドレス |
| [--netmask <str>] | ipaddr から適当と思われるものを補完 | 仮想 IP のネットマスク |
| [--device <str>] | ipaddr と netmask から適当と思われるものを補完 | 仮想 IP と紐づくネットワークインターフェース名 |

### extend ip

| オプション | デフォルト | 説明 |
|---|---|---|
| [--ipaddr <str>] | 拡張元の情報から補完 | 拡張先となるノード上の仮想 IP アドレス |

| | | |
|---|---|---|
| [--netmask <str>] | ipaddr から適当と思われるものを補完 | 拡張先となるノード上の仮想 IP のネットマスク |
| [--device <str>] | ipaddr と netmask から適当と思われるものを補完 | 拡張先となるノード上の仮想 IP と紐づくネットワークインターフェース名 |

### config ip

| オプション | デフォルト | 説明 |
|---|---|---|
| [--pinglist <str>] | | オプション用 Ping 先リスト (カンマ区切りで複数指定も可能) |
| [--srcaddr <str>] | | **0** または **1** を指定。<br>仮想 IP アドレスを同一サブネットへの外部通信 IP トラフィックのソースアドレスに使用するかを指定。<br>• **0** : 使用する。<br>• **1** : 使用しない。 |
| [--restoremode <str>] | | **Enabled** または **Disabled** を指定。<br>IP リソースに対するリストアおよびリカバリを有効･無効に設定。<br>• **Enabled** : リストアおよびリカバリを有効<br>• **Disabled** : リストアおよびリカバリを無効 |

---

## MySQL

リソースの種類は `app: database, typ: mysql` です。
このコマンドを利用する上での前提条件や詳細な動作については「MySQL Recovery Kit 管理ガイド」の「インストール」を参照してください。

### create mysql

| オプション | デフォルト | 説明 |
|---|---|---|
| --cnf <str> | | MySQL のコンフィグファイルの絶対パス |
| --bin <str> | | MySQL の実行バイナリのあるディレクトリの絶対パス |
| [--instance <str>] | なし | 保護したい MySQL のインスタンス番号。<br>もし MySQL をシングルインスタンスで使用している場合は、この番号を指定しないでください。 |

### extend mysql

| オプション | デフォルト | 説明 |
|---|---|---|

| | | |
|---|---|---|
| [--bin <str>] | | 拡張先となるノード上にある MySQL の実行バイナリのあるディレクトリの絶対パス |

### config mysql

コンフィグレーションする項目はありません

---

## NFS

リソースの種類は app: gen, typ: nfs です。
このコマンドを利用する上での前提条件や詳細な動作については「NFS Recovery Kit 管理ガイド」の「設定作業」を参照してください。

### create nfs

| オプション | デフォルト | 説明 |
|---|---|---|
| --export <str> | | NFS ファイルシステムのエクスポートポイント。 |
| --ip <str> | | NFS ファイルシステムにアクセスするためにクライアントが使用する仮想 IP アドレスに対応する IP リソースのタグ名。 |

### extend nfs

オプションなし

### config nfs

コンフィグレーションする項目はありません

---

## Oracle

リソースの種類は app: database, typ: oracle です。
このコマンドを利用する上での前提条件や詳細な動作については「Oracle Recovery Kit 管理ガイド」の「LifeKeeper の設定作業」を参照してください。

### create oracle

| オプション | デフォルト | 説明 |
|---|---|---|
| --sid <str> | | データベースの ORACLE_SID。 |

| | | |
|---|---|---|
| [--listener <str>] | なし | Oracle リソースに依存して含まれる Oracle Listener リソースのタグ名。 |
| [--user <str>] | なし | Oracle データベースのユーザ名。 |
| [--password <str>] | なし | Oracle データベースのユーザのパスワード。 |

### extend oracle

オプションなし

### config oracle

| オプション | デフォルト | 説明 |
|---|---|---|
| --user <str> | | Oracle データベースのユーザ名。 |
| --password <str> | | Oracle データベースのユーザのパスワード。 |
| --role <str> | | ユーザのロール。<br>sysdba または sysoper を指定してください。 |

---

## OracleListener

リソースの種類は `app: database, typ: listener` です。
このコマンドを利用する上での前提条件や詳細な動作については「Oracle Recovery Kit 管理ガイド」の「複数のリソースに対する共有 Oracle Listener の作成」を参照してください。

### create listener

| オプション | デフォルト | 説明 |
|---|---|---|
| --exe <str> | | Listener の実行パス。 |
| --config <str> | | Listener の実行設定ファイルのパス。 |
| --protection <str> | | Listenerの保護レベル。<br>• **Full** : 開始、停止、監視および回復<br>• **Intermediate** : 開始、監視および回復<br>• **Minimal** : 開始および監視のみ |
| --recovery <str> | | Listener のリカバリレベル。<br>• **Standard** : 標準の LifeKeeper リカバリを有効にします。すべての listener 障害がローカルで発生し、必要であれば、有効なバックアップサーバへのフェイルオーバを行います。<br>• **Optional** : オプションの LifeKeeper リカバリを有効にします。すべての |

| | | |
|---|---|---|
| | | listener 障害がローカルで発生しますが、有効なバックアップサーバへのフェイルオーバは行いません。 |
| [--user <str>] | なし | システムユーザー名。<br>Listener を開始、停止、監視、および回復する権限を持つシステムユーザーを指定してください。 |
| [--listener <str>] | LISTENER | 保護する Oracle Listener の名前。 |
| [--iptag <str>] | なし | このリソース階層の依存関係として保護される IP リソースのタグ名。 |

## extend listener

| オプション | デフォルト | 説明 |
|---|---|---|
| [--exe <str>] | 拡張元と同一 | Listener の実行パス。 |
| [--config <str>] | 拡張元と同一 | Listener の実行設定ファイルのパス。 |

## config listener

| オプション | デフォルト | 説明 |
|---|---|---|
| --type <str> | | 変更する項目名。項目によりオプションが異なります。<br>• **ProtectionLevel** : Listener の保護レベル。<br>• **RecoveryLevel** : Listener のリカバリレベル。<br>• **Listener** : 保護する Oracle Listener の名前。 |

### --type ProtectionLevel

| オプション | デフォルト | 説明 |
|---|---|---|
| --value <str> | | create の protection オプションと同様。 |

### --type RecoveryLevel

| オプション | デフォルト | 説明 |
|---|---|---|
| --value <str> | | create の recovery オプションと同様。 |

### --type Listener

| オプション | デフォルト | 説明 |
|---|---|---|
| --value <str> | | 保護する Oracle Listener の名前。 |
| [--iptag <str>] | なし | 依存関係として保護される IP リソースのタグ名。 |

# OraclePDB

このコマンドを利用する上での前提条件や詳細な動作については「Oracle Recovery Kit 管理ガイド」の「Oracle Multitenant 構成による Pluggable Database の設定」を参照してください。

## create pdb

| オプション | デフォルト | 説明 |
|---|---|---|
| --sid <str> | | データベースの ORACLE_SID。 |
| --pdb <str> | | 保護するPDBリスト（カンマ[,]区切りで複数指定できます。） |

## extend pdb

オプションなし

## config pdb

| オプション | デフォルト | 説明 |
|---|---|---|
| --pdb <str> | | 保護するPDBリスト（カンマ[,]区切りで複数指定できます。） |

---

# PostgreSQL

リソースの種類は `app: database, typ: pgsql` です。
このコマンドを利用する上での前提条件や詳細な動作については「PostgreSQL Recovery Kit 管理ガイド」の「インストール」を参照してください。

## create pgsql

| オプション | デフォルト | 説明 |
|---|---|---|
| --datadir <str> | | データベースのデータがあるディレクトリの絶対パス。 |
| --port <num> | | PostgreSQL が使用しているポート番号。 |
| --socket <str> | | PostgreSQL が使用しているソケットのパス。 |
| --dbuser <str> | | PostgreSQL が使用しているユーザー名。 |
| --logfile <str> | | ログを出力先となる絶対パス。 |
| [--exepath <str>] | /usr/bin | 実行ファイルがあるディレクトリの絶対パス。 |
| [--clientexe <str>] | <exepath>/psql | 実行ファイル “psql” の絶対パス。 |
| [--adminexe <str>] | <exepath>/pg_ctl | 実行ファイル “pg_ctl” の絶対パス。 |

### extend pgsql

| オプション | デフォルト | 説明 |
|---|---|---|
| [--exepath <str>] | 拡張元と同一 | 拡張先となるノード上で実行ファイルがあるディレクトリの絶対パス。指定しなかった場合は、拡張元の設定を引き継ぎます。 |

### config pgsql

| オプション | デフォルト | 説明 |
|---|---|---|
| [--dbuser <str>] | なし | PostgreSQL が使用しているユーザー名。 |

---

## Postfix

このコマンドを利用する上での前提条件や詳細な動作については「Postfix リソース階層の作成」や「Postfix リソース階層の拡張」を参照してください。

### create postfix

| オプション | デフォルト | 説明 |
|---|---|---|
| [--binary <binary location>] | /usr/sbin | postfix コマンドが置かれているディレクトリ。 |
| [--config <config file location>] | /etc/postfix | postfix の設定ファイル (main.cf) が置かれているディレクトリ。 |

### extend postfix

オプションなし

### config postfix

コンフィグレーションする項目はありません

---

## QSP

リソースの種類は `app: gen, typ: qsp` です。
このコマンドを利用する上での前提条件や詳細な動作については「Quick Service Protection (QSP) Recovery Kit」を参照してください。

### create qsp

| オプション | デフォルト | 説明 |
|---|---|---|
| --service <str> | | 保護したいサービス名。 |
| [--quickCheck <str>] | enable | 監視機能の有効･無効を設定します。<br>有効にする場合は “enable”、無効にする場合は “disable” を指定します。 |

### extend qsp

オプションなし

### config qsp

| オプション | デフォルト | 説明 |
|---|---|---|
| [--quickCheck <str>] | なし | 監視機能の有効･無効。<br>有効にする場合は “enable”、無効にする場合は “disable” を指定します。 |
| [--timeout_restore <num>] | なし | restore のタイムアウトの秒数。0 を指定した場合はタイムアウトしません。 |
| [--timeout_remove <num>] | なし | remove のタイムアウトの秒数。0 を指定した場合はタイムアウトしません。 |
| [--timeout_quickcheck <num>] | なし | quickCheck のタイムアウトの秒数。0 を指定した場合はタイムアウトしません。 |
| [--timeout_recover <num>] | なし | recover のタイムアウトの秒数。0 を指定した場合はタイムアウトしません。 |

---

## Raw

このコマンドを利用する上での前提条件や詳細な動作については「Rawデバイスリソース階層の作成」や「Rawデバイスリソース階層の拡張」を参照してください。

### create raw

| オプション | デフォルト | 説明 |
|---|---|---|
| --partition <str> | | Raw デバイスのパーティション。 |

### extend raw

オプションなし

### config raw

コンフィグレーションする項目はありません

---

## Route53

リソースの種類は app: comm, typ: route53 です。
このコマンドを利用する上での前提条件や詳細な動作については「Route53 Recovery Kit 管理ガイド」の「設定」を参照してください。

### create route53

| オプション | デフォルト | 説明 |
|---|---|---|
| --domain <str> | | 保護したい Route53 に存在するドメイン名。 |
| --hostname <str> | | 保護したいホスト名。 |
| --ip_resource <str> | | 事前に作成した IP リソースのタグ名。 |

### extend route53

オプションなし

### config route53

コンフィグレーションする項目はありません

---

## Samba

このコマンドを利用する上での前提条件や詳細な動作については「Samba リソース階層の作成」や「Samba リソース階層の拡張」を参照してください。

### create samba

| オプション | デフォルト | 説明 |
|---|---|---|
| --config <config file name> | /etc/samba/smb.conf | Samba 設定ファイルの絶対パス名。 |

### extend samba

オプションなし

### config samba

コンフィグレーションする項目はありません

---

## SAP

詳細については SAP Recovery Kit 管理ガイド の設定に関連する項目を参照してください。

### create sap

| オプション | デフォルト | 説明 |
|---|---|---|
| --sid <str> | | SAP インストール時の SID |
| --instance <str> | | SAP インスタンス(例：ASCS10) |
| [--protection_level <FULL\|STANDARD\|BASIC\|MINIMUM>] | FULL | 保護対象の SAP リソースの保護レベル |
| [--recovery_level <FULL\|REMOTE\|LOCAL\|OFF>] | FULL | 保護対象の SAP リソースのリカバリーレベル |
| [--ip <str>] | | 指定した SAP リソースの仮想 IP を保護する LifeKeeper IP リソースタグ。この IP リソースは、SAP HANA リソースが作成されているサーバーで In Service になっている必要があります。 このパラメーターには カンマ区切りされた IP リソースタグが許可されます。この引数により指定された IP リソースは、SAP リソースの子リソースになります。 |
| [--dependent_fs <str>] | | 指定の SAP インスタンスに関連するファイルシステムを保護するファイルシステムリソース (gen:filesys) もしくは NFS (gen:nfs) リソースに対する カンマ区切りされた LifeKeeper リソースタグ (例：/sapmnt,/usr/sap/SID/ASCS10)。全ての指定したリソースは、SAP HANA リソースが作成されているサーバーで In Service になっている必要があります。 この引数により指定されたファイルシステムおよび/もしくは NFS リソースは SAP リソースの子リソースになります。注記: このパラメーターは、 --automate_fs_creation とは併用できません。 |
| [--dependent_cs <str>] | | 作成されている SAP リソースの子リソースとして追加さ |

| | | |
|---|---|---|
| | | れるセントラルサービスリソース (例：ASCS10) に対する LifeKeeper SAP リソースタグ。指定したセントラルサービスリソースは、SAP HANA リソースが作成されているサーバーで In Service になっている必要があります。 注記: このパラメーターは、PAS インスタンスを保護するリソースを作成する時に適用可能である場合があります。詳細は SAP リソース階層の作成 の「プライマリーアプリケーションサーバーリソースの作成」の項目を参照してください。 |
| [--automate_fs_creation] | | SAP リソースの作成中に、LifeKeeper は既存の SAP ディレクトリ構造をスキャンし、検出した共有ファイルシステムマウントを保護するために LifeKeeper リソースの自動作成を試みます。 このプロセス中に作成されたリソースは SAP リソースの子リソースになります。プロセス完了後、階層の状態を確認してください。注記: 複製されたファイルシステムの階層はこのプロセスで自動的に作成されません。このパラメーターは --dependent_fs parameterと併用して使用することはできません。 |

## extend sap

オプションなし

## config sap

| オプション | デフォルト | 説明 |
|---|---|---|
| [--protection_level <FULL\|STANDARD\|BASIC\|MINIMUM>] | | 保護対象の SAP リソースの 保護レベル を更新します。このコマンドは SAP リソースが現在 In Service になっているサーバー上で実行する必要があります。 |
| [--recovery_level <FULL\|REMOTE\|LOCAL\|OFF>] | | 保護対象の SAP リソースの リカバリーレベル を更新します。このコマンドは SAP リソースが現在 In Service になっているサーバー上で実行する必要があります。 |
| [--sshcc_action <start\|stop\|migrate[:<node>]\|maintenance:<enable\|disable\|check>>] | | SAP SIOS HA Cluster Connector (SSHCC) アクションを実行します。このコマンドは SAP リソースが現在 In Service になっているサーバー |

| | | |
|---|---|---|
| | | 上で実行する必要があります。このパラメーターの設定可能な値は下記の通りです。:<br><br>start – HA cluster connector 経由で SAP リソースを In Service にします。<br><br>stop – HA cluster connector 経由で SAP リソースを Out of Service にします。<br><br>migrate[:<node>] – SAP リソースを別のサーバーへ移行します。 オプションとして、対象のマイグレーションサーバーは “:” をマイグレートパラメーターに追加することで指定することが可能です。 (例： lkcli resource config sap —tag SAP-SID_ASCS10 —sshcc_action migrate:node2)。対象サーバーが明示的に提供されない場合は、HA cluster connector が、現在のクラスターの状況をベースに最適な対象サーバーを決定します。<br><br>maintenance:<enable\|disable\|check> – 指定の SAP リソースを含む階層に対して SAP メンテナンスモード の有効化、無効化もしくはステータスの確認を行います。 |

# SAP HANA

詳細については SAP HANA Recovery Kit管理ガイド を参照してください。

## create hana

| オプション | デフォルト | 説明 |
|---|---|---|
| --sid <str> | | SAP HANA インストール時の SID |
| --instance | | SAP HANA インスタンス (例 : HDB00) |

| | | |
|---|---|---|
| <str> | | |
| [--ip <str>] | | SAP HANA データーベースの仮想 IP を保護する LifeKeeper IP リソースタグ。この IP リソースは、SAP HANA リソースが作成されているサーバーで In Service になっている必要があります。 |

### extend hana

オプションなし

### config hana

| オプション | デフォルト | 説明 |
|---|---|---|
| [--stop_all_dbs] | | 全てのクラスターノードで SAP HANA データベースを停止します。 (例：メンテナンスのため)。このコマンドは HANA リソースが現在 In Service になっているクラスターノード上で実行する必要があります。 |
| [--takeover_with_handshake <target server>] | Local server | 指定されたターゲットサーバーで SAP HANA データベースの “Takeover with handshake” を実行します。SAP HANA 2.0 SPS04 以降の機能で、新しいデータベースホスト上でSAP HANA システムレプリケーションの引き継ぎを行う前にプライマリーデータベースを (完全に停止するのではなく)一時停止することで、スイッチオーバー時のプライマリーデータベースのダウンタイムを抑えます。 |

## SAP MaxDB

詳細については SAP MaxDB Recovery Kit 管理ガイド の「リソース設定作業」を参照してください。

### create sapdb

| オプション | デフォルト | 説明 |
|---|---|---|
| [--prog <str>] | /etc/opt/sdb にあるパス | SAP MaxDB のプログラムパスです。別のディレクトリパスを入力することもできます。パス名に使用できる有効な文字は、文字、数字および特殊文字 – _ . / です。 |
| [--sid <str>] | まだ構成されていない1番目のインスタ | SAP MaxDB のインスタンス ID です。 |

| | | |
|---|---|---|
| | ンス | |
| [--user <str>] | /etc/opt/sdb にあるユーザー | SAP MaxDB コマンドを所有しているか、SAP MaxDB コマンドを実行する権限のあるシステムユーザーです。 このユーザーは、テンプレートサーバー上に存在している必要があります。 |
| [--xuser <str>] | 構成された1番目の XUSER key | XUSER key は、SAP MaxDB ツールで使用するデータベースユーザーのデータを保存するために使用されます。XUSER key は、システムユーザーとデータベースインスタンスの組み合わせに対して一致するサーバー上に構成される必要があります。 |

### extend sapdb

| オプション | デフォルト | 説明 |
|---|---|---|
| [--prog <str>] | ターゲット上の/etc/opt/sdb にあるパス | SAP MaxDB のプログラムパスです。別のディレクトリパスを入力することもできます。パス名に使用できる有効な文字は、文字、数字および特殊文字 – _ . / です。 |
| [--user <str>] | テンプレートサーバー上のこのインスタンスに対して定義されたユーザー | ターゲットサーバー上で SAP MaxDB コマンドを所有しているか、SAP MaxDB コマンドを実行する権限のあるシステムユーザーです。 このユーザーは、ターゲットサーバー上に存在している必要があります。 |
| [--xuser <str>] | テンプレートサーバー上のこのインスタンスに対する XUSER key | XUSER key は、SAP MaxDB ツールで使用するデータベースユーザーのデータを保存するために使用されます。XUSER key は、システムユーザーとデータベースインスタンスの組み合わせに対するターゲットサーバー上に構成される必要があります。 |

### config sapdb

オプションなし

---

## Sybase

このコマンドを利用する上での前提条件や詳細な動作については「Sybase Recovery Kit 管理ガイド」の「インストール」を参照してください。

### create sybase

| オプション | デフォルト | 説明 |
|---|---|---|
| --basedir <str> | | Sybaseのインストールディレクトリ |
| --datadir <str> | | Sybaseのインスタンスデータディレクトリ |

| | | |
|---|---|---|
| --instance <str> | | データベースインスタンス |
| --user <str> | | データベースのユーザー<br>※指定しない場合、saユーザーとしてデータベースへのアクセス、操作を試みます。 |
| --password <str> | | データベースのユーザーのパスワード |
| --backup <str> | none | Backupサーバー |

### extend sybase

オプションなし

### config sybase

| オプション | デフォルト | 説明 |
|---|---|---|
| --user <str> | | データベースのユーザー |
| --password <str> | | データベースのユーザーのパスワード |
| --backup <str> | 現在の値 | Backupサーバー |
| --monitor <str> | 現在の値 | Monitorサーバー |

## WebSphere MQ

このコマンドを利用する上での前提条件や詳細な動作については WebSphere MQ Recovery Kit 管理ガイド の「設定」セクションを参照してください。

### create mq

| オプション | デフォルト | 説明 |
|---|---|---|
| --qmgr <str> | | キュー・マネージャーの名前 |
| [--protect_listener <YES\|NO>] | YES | キュー・マネージャーに関連付けられたMQリスナーを保護するかどうかを指定します。 |
| [--channel <str>] | SYSTEM.DEF.SVRCONN | キュー・マネージャーの接続チャネル。 |
| [--ip <str>] | | リスナーのIPアドレスを保護するLifeKeeper IPリソースタグ。IPリソースは、MQリソースが作成されているサーバーでIn Serviceになっている必要があります。この引数で指定されたIPリソースは、MQリソースの子リソースになります。 |

## extend mq

オプションなし

## config mq

詳細については、 リソースプロパティ設定の編集 を参照してください。

| オプション | デフォルト | 説明 |
|---|---|---|
| [--protect_listener <YES\|NO>] | | キュー・マネージャーに関連付けられたMQリスナーを保護するかどうかを指定します。このパラメーターは、MQリソースがIn Service（ISP）になっているサーバーでのみ変更できます。 |
| [--test_queue <str>] | | PUT/GETヘルスチェックを実行する際に使用するLifeKeeper MQ Recovery Kitのテストキューの名前。値を空の文字列 ( `--test_queue ""` ) に設定して、PUT/GETテストを無効にすることができます。 |
| [--log_level <ERROR\|INFORMATIONAL\|DEBUG\|FINE>] | | MQリソースのログレベル。 |
| [--immediate_stop_timeout <str>] | | キュー・マネージャーの即時停止を試行する際に使用されるタイムアウト値（単位：秒）。 |
| [--preemptive_stop_timeout <str>] | | キュー・マネージャーの先制終了を試行するときに使用されるタイムアウト値（単位：秒）。 |
| [--channel <str>] | | キュー・マネージャーの接続チャネル。このパラメーターは、MQリソースがIn Service（ISP）になっているサーバーでのみ変更できます。 |
| [--cmd_server_protection <Full\|Minimal>] | | MQコマンド・サーバーの保護レベル。 |

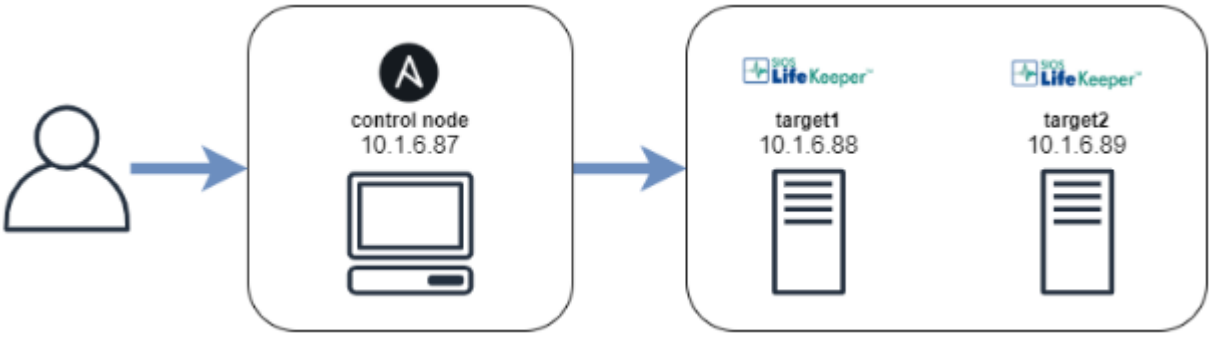
# 5.6.3. Ansible による LifeKeeper のセットアップ

## 概要

本ガイドは、設定ファイルを元に Ansible で、複数ノードに LifeKeeper をインストールする手順を記載しています。
主に、既に Ansible をご利用しているユーザーが、ご自身の環境に LifeKeeper のインストールを追加する際に参照して頂くことを想定しています。
そのため、本ガイドでは、実際の Ansible のタスクのサンプルを記載しています。
お客様自身でプレイブックをお持ちであれば、設定値などは環境に応じて変更してご活用ください。

## 環境

ここでは2ノード構成を作成します。




OS: CentOS 8.0
Python: 3.6.8
Ansible: 2.9.9
LifeKeeper: 9.5.1

## LifeKeeper をインストールするターゲットのセットアップ

### ホスト名と名前解決サービスの設定

LifeKeeper は、クラスター内のターゲットを名前解決できる必要があります。
その為、DNS サービスまたは、/etc/hosts を設定してクラスター間で名前解決できるように設定してください。
本ガイドでは、各 /etc/hosts を設定しています。以下で、設定を変更できます。

**hosts.yml**

```
---
# /etc/hostsにansibleを実行対象のホストをすべて登録する
- name: update /etc/hosts
  lineinfile:
    dest: /etc/hosts
    state: present
    insertafter: EOF
    regexp: "^{{ item.address }}"
    line: "{{ item.address }}\t{{ item.hostname }}"
  with_items:
    - address: "10.1.6.88"
      hostname: "target1"
    - address: "10.1.6.89"
      hostname: "target2"
    - address: "10.1.6.87"
      hostname: "control-node"
```

## Firewall の設定

Firewall が有効の場合、各ターゲットで Firewall の設定が必要です。
LifeKeeper では、ターゲット間で TCP の 7365 ポートを利用して通信します。
本ガイドでは、そのポートを、default の public ゾーンに設定しています。

**firewall.yml**

```
---
  - name: enable lifekeeper port
    firewalld:
      zone: public
      port: 7365/tcp
      permanent: yes
      state: enabled
    register: firewalld

  - name: reload
    command: firewall-cmd --reload
    when: firewalld.changed
```

## SELinux の設定

LifeKeeper を使用するには、SELinux を disabled にする必要があります。

以下で、設定を変更できます。

**selinux.yml**

```
---
- name: disable SELinux
  become: yes
  selinux: state=disabled
  register: selinux

- name: reboot a node
  become: yes
  reboot:
  when: selinux.reboot_required
```

---

## LifeKeeper のインストールイメージとライセンスの準備

1. LifeKeeper のインストールイメージとライセンスを取得してください。

   製品購入済みのお客様用
   LifeKeeper 製品のソフトウェアをダウンロード
   https://download.sios.com/

   評価目的のお客様用
   評価版ダウンロード
   https://sios.jp/products/lkdk/product/evaluation_lk.html

## LifeKeeper のインストール用のレスポンスファイルの準備

LifeKeeper をインストールするためには、インストール用のレスポンスファイルが必要です。既にレスポンスファイルをお持ちの方は、この項目をスキップしてください。本ガイドでは、control-nodeマシンで、レスポンスファイルを作成します。

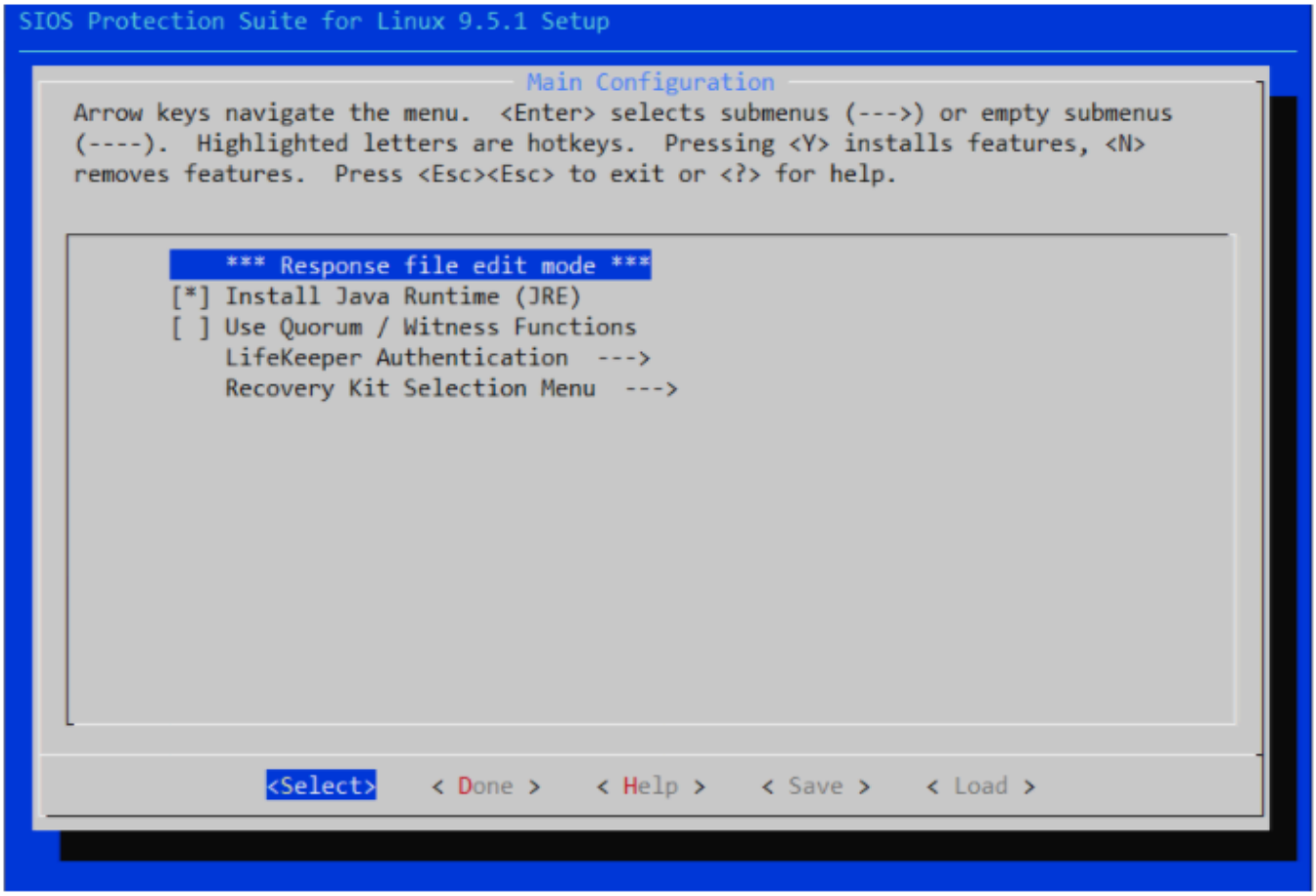
1. 以下のダウンロード先から create_response_file スクリプトを入手してください。

   自動インストール用設定 作成ツール

2. レスポンスファイルのファイル名を指定して、create_response_file スクリプトを実行します。

```
# ./create_response_file /root/lifekeeper/LKCONF
```

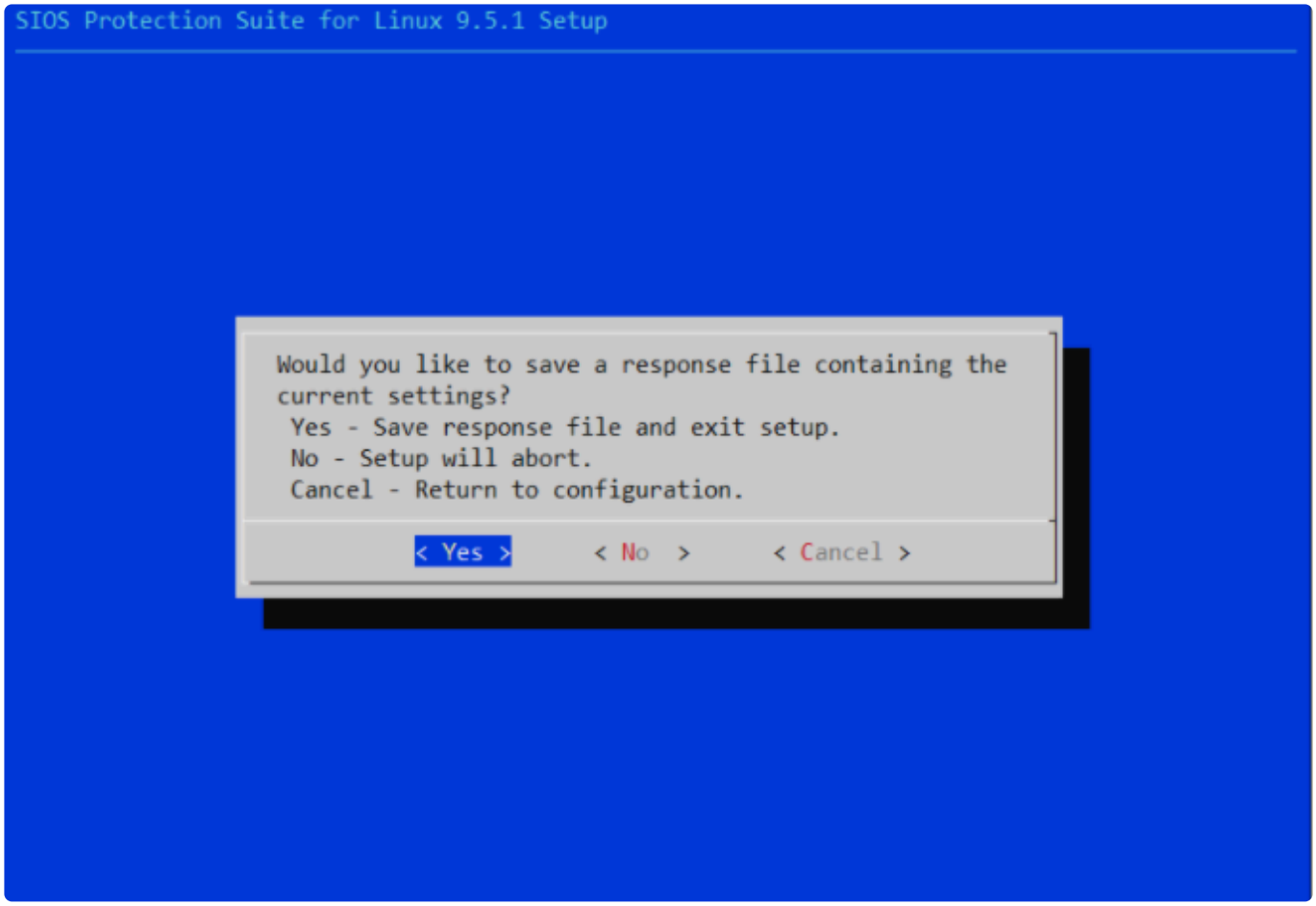
3. create_response_file スクリプトを起動すると、以下のメニュー画面が表示されます。




4. 必要な設定を選択してください。

- Install Java Runtime (JRE) の選択
- Recovery Kit Selection Menu でインストールしたい Recovery Kit の選択

5. 設定後、< Done > を選択してください。その後、< Yes > を選択し終了します。




6. 生成された LKCONF ファイルを確認し、設定が正しいことを確認してください。以下の例は、Oracle、PostgreSQL、MySQL、DataKeeper for Linux、Apache Recovery Kits を選択した場合のレスポンスファイルの例です。

```
# LifeKeeper setup response file
# DO NOT EDIT MANUALLY
LKCONF_INSTALL_JRE="y"
LKCONF_SELONLY="y"
LKCONF_AUTH="y"
LKCONF_LKUSER_lkadmin="root"
LKCONF_steeleye_lkORA="y"
LKCONF_steeleye_lkPGSQL="y"
LKCONF_steeleye_lkSQL="y"
LKCONF_steeleye_lkDR="y"
LKCONF_steeleye_lkAPA="y"
```

## 取得した各ファイルの配布とディレクトリーの指定

1. 取得したインストールイメージとすべてのクラスタノードのライセンスと事前に作成したレスポンスファイルを control-node マシンへ配布してください。以下は、取得した各ファイルのディレクトリ構成例です。

```
/root/lifekeeper/
        ├sps.img  ： LifeKeeper のインストールイメージ
        ├LKCONF   ： LifeKeeper のサイレントインストール用のレスポンスファイル
        └ licenses/
                ├target1/ ： target1 のライセンスファイル群
                └target2/ ： target2 のライセンスファイル群
```

**inventory**

```
[targets]
target1
target2
```

2. 使用するディレクトリーを指定してください。

**defaults/main.yml**

```
---
# control-node側でインストールに必要なファイルを保存しておくディレクトリ
control_node_dir: /root/lifekeeper
# ターゲット側でインストールに一時的に必要なファイルを保存しておくディレクトリ
target_dir: /tmp/lifekeeper
# ターゲット側でインストールイメージのマウントディレクトリ
target_mnt: /mnt
```

以下の処理は各ターゲットで実行されるようにしてください。

## LifeKeeper のインストールイメージとライセンスとレスポンスファイルの配布

**deploy.yml**

```
---
  - name: deploy to lifekeeper install files
    copy:
      src: "{{ control_node_dir }}/"
      dest: "{{ target_dir }}"
```

## LifeKeeperのインストールイメージのマウント

**mount.yml**

```
---
```

```
  - name: mount setup image
    mount:
      path: "{{ target_mnt }}"
      src: "{{ target_dir }}/sps.img"
      fstype: iso9660
      state: mounted
```

## LifeKeeper のインストール

### install.yml

```
---
  - name: install lifekeeper
    shell:
      # setupを直接実行すると失敗します。
      cmd: script -q -e -c "{{ target_mnt | quote }}/setup -f {{ target_dir |
quote }}/LKCONF -q y" {{ target_dir | quote }}/lifekeeper_install.log
```

## LifeKeeper の環境変更の追加

### bash_profile.yml

```
---
  - name: update lifekeeper PATH in bash_profile
    blockinfile:
      path: /root/.bash_profile
      marker: "# {mark} ANSIBLE MANAGED BLOCK: LifeKeeper"
      block: |
        PATH=$PATH:/opt/LifeKeeper/bin
        MANPATH=$MANPATH:/opt/LifeKeeper/man
        export PATH MANPATH
```

## LifeKeeper のライセンスのインストール

### license.yml

```
---
- name: install lifekeeper license
  shell:
```

```
    cmd: |
      for file in `ls`; do
        /opt/LifeKeeper/bin/lkcli license --file $file
      done
    chdir: "{{ target_dir }}/licenses/{{ inventory_hostname }}"
```

## LifeKeeper の起動

### lkstart.yml

```
---
- name: start lifekeeper
  command: /opt/LifeKeeper/bin/lkcli start

- name: lifekeeper status
  command: /opt/LifeKeeper/bin/lkcli status
```

## LifeKeeper のセットアップ後、不要なファイルの削除

### clean.yml

```
---
  - name: unmount a mounted lifekeeper image
    mount:
      path: "{{ target_mnt }}"
      state: absent

  - name: delete lifekeeper files
    file:
      path: "{{ target_dir }}"
      state: absent
```

## Ansible Playbook の例

前のセクションに記載された全てのタスクファイルを作成し、それらのタスクファイルを組み合わせることで以下の例に示すような Playbook を作成できます。

### deploy_lifekeeper.yml

```
---
```

```
- name: deploy and install lifekeeper on all targets
  hosts: targets
  tasks:
    - include_vars: main.yml
    - include_tasks: hosts.yml
    - include_tasks: firewall.yml
    - include_tasks: selinux.yml
    - include_tasks: deploy.yml
    - include_tasks: mount.yml
    - include_tasks: install.yml
    - include_tasks: bash_profile.yml
    - include_tasks: license.yml
    - include_tasks: lkstart.ym l
    - include_tasks: clean.yml
```

この Playbook の例は、下記のコマンドを root として実行するか、既存の Ansible Playbook に追加することができます。

```
# ansible-playbook deploy_lifekeeper.yml -i inventory
```

本ガイドに記載されている全ての Ansible タスクが実行されると、Ansible はインベントリファイルで指定された全てのクラスターノードに LifeKeeer をセットアップすることができます。

ここまでの手順で、LifeKeeper がインストールされたマシンが作成されます。

# 5.6.4. LKCLI ガイド

## 目的

本資料は、LKCLI (LifeKeeper コマンドラインインターフェース)を利用して、コマンドラインでの LifeKeeper 環境の構築に役立つことを目的としています。

なお、LKCLI の各コマンドの詳細については、「LKCLI（LifeKeeperコマンドラインインターフェース）」を参照してください。

# 5.6.4.1. コミュニケーションパスの作成･削除

## 概要

### コミュニケーションパスとは？

LifeKeeper では、複数台のノードをお互いに接続することで、クラスターを構成していきます。
もしあるノードで障害が発生しても、クラスター内の別のノード上のリソースに切り替えることで可用性を維持することができます。

そのため、各ノード間で通信を行うための経路を事前に設定する必要があります。
この通信経路のことを LifeKeeper では「コミュニケーションパス」と呼びます。

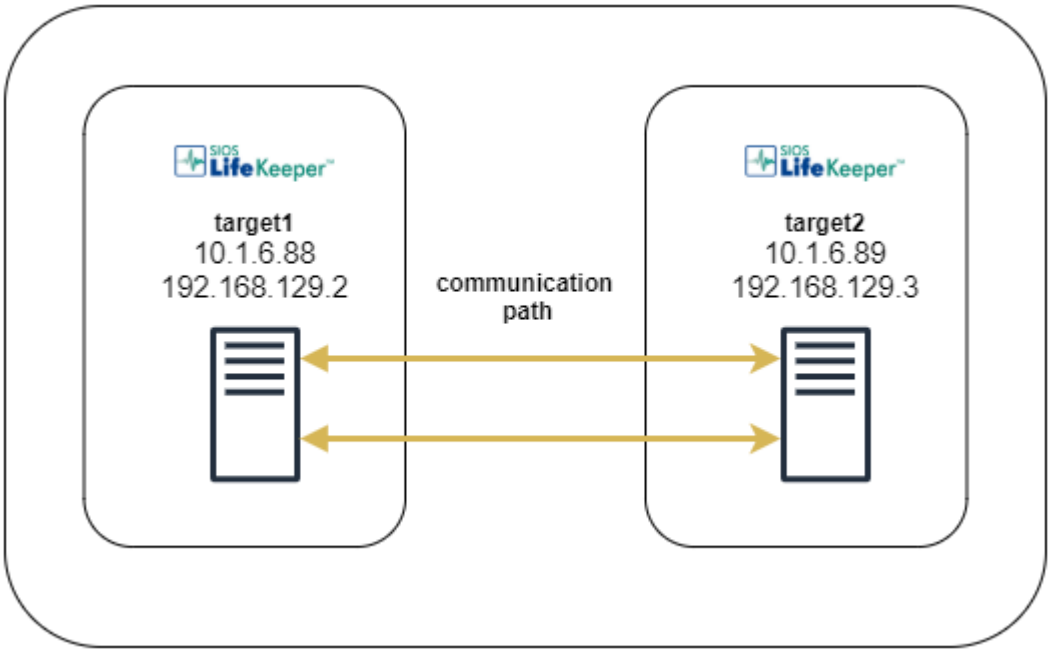
コミュニケーションパスを設定すると、各ノードはお互いに「ハートビート」と呼ばれる、LifeKeeper が正常に動作しているかを示す信号を送信します。
このハートビートが受信できなくなると、ノードに障害があったと判断されます。
もしネットワークに障害が発生した場合、お互い正常であるがハートビートを受け取れなくなるため、LifeKeeper ではすべての2つのノード間を、2本以上のコミュニケーションパスで接続することを推奨しています。

またコミュニケーションパスは、ハートビートのほかに「 LifeKeeper の内部通信」にも利用されます。そのため、コミュニケーションパスが接続されているノード間であれば、リモート先のノード上の LifeKeeper をネットワーク越しに操作することが可能です。

本ガイドでは、2つのノード間で1本のコミュニケーションパスを作成する手順と、そのコミュニケーションパスを削除する手順を説明します。

## 構成

本ガイドでは、LifeKeeper v9.5.0 の使用を想定しています。また、本ガイドに記載されているコマンドなどは以下の図を想定しています。




まずはじめに、コマンドを実行する環境で以下の項目をご確認ください。

- コミュニケーションパスで使用するポート( 7365/tcp ポート)が使用できる
  - Firewall で通信が許可されている、または Firewall が無効になっている
- LifeKeeper を起動しているマシンが2台存在する

次に、CLI コマンドの実行に必要な各マシンの情報を、必要であれば記録してください。

1. 各マシンで利用できる IP アドレス

```
# ip address | grep 'inet'
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
    inet 10.1.6.88/16 brd 10.1.255.255 scope global noprefixroute ens192
    inet6 fe80::a633:2758:4976:b42/64 scope link noprefixroute
    inet 192.168.129.2/24 brd 192.168.129.255 scope global noprefixroute ens22
4
    inet6 fe80::34a4:417f:b58b:dc15/64 scope link noprefixroute
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
```

2. 各マシンのホスト名

```
# hostname
target1
```

※ OS によってコマンドが異なる場合があります

# LifeKeeper のコミュニケーションパスの作成

コミュニケーションパスを作成するコマンドは、1度につき「コマンド実行元から指定先への片方向のみ」作成します。そのため、コミュニケーションパスを双方向に通信できるようにするには、2台のマシンそれぞれでコマンドを実行する必要があります。

## target1 で実行

1. コミュニケーションパスが接続されていないことを確認します。

```
[target1]# lkcli status -q
LOCAL    TAG    ID     STATE      PRIO   PRIMARY
```

2. target1 から target2 へのコミュニケーションパスを作成します。

```
[target1]# lkcli commpath create --laddr 10.1.6.88 --raddr 10.1.6.89 --dest ta
rget2
Performing commpath 'target2:10.1.6.88/10.1.6.89' create...
Commpath 'target2:10.1.6.88/10.1.6.89' created successful.
```

### コマンド引数

| 項目 | 入力値 |
|---|---|
| --laddr | 接続元となるローカルノード上の IP アドレス |
| --raddr | 接続先となるリモートノード上の IP アドレス |
| --dest | 接続先となるリモートノードのホスト名 |

3. target2 へのコミュニケーションパスが作成されていることを確認します。
   ※ この段階では片方向の接続のみであるため、STATE は DEAD となります

```
[target1]# lkcli status -q
LOCAL    TAG    ID     STATE      PRIO   PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE      STATE      PRIO
target2  TCP     10.1.6.88/10.1.6.89   DEAD          1
```

## target2 で実行

4. コミュニケーションパスが接続されていないことを確認します。

```
[target2]# lkcli status -q
LOCAL    TAG    ID     STATE      PRIO   PRIMARY
```

5. target2 から target1 へのコミュニケーションパスを作成します。

```
[target2]# lkcli commpath create --laddr 10.1.6.89 --raddr 10.1.6.88 --dest ta
rget1
Performing commpath 'target1:10.1.6.89/10.1.6.88' create...
Commpath 'target1:10.1.6.89/10.1.6.88' created successful.
```

### コマンド引数

| 項目 | 入力値 |
|---|---|
| --laddr | 接続元となるローカルノード上の IP アドレス |
| --raddr | 接続先となるリモートノード上の IP アドレス |
| --dest | 接続先となるリモートノードのホスト名 |

6. コミュニケーションパスが確立していることを確認します。
   STATE が ALIVE になることを確認します。
   ※ コミュニケーションパス作成後、ALIVE になるまで数秒かかります

```
[target2]# lkcli status -q
LOCAL    TAG    ID     STATE      PRIO   PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE      STATE      PRIO
target1  TCP     10.1.6.89/10.1.6.88   ALIVE         1
```

# LifeKeeperのコミュニケーションパスの削除

コミュニケーションパスを削除するコマンドは、1度につき「コマンド実行元から指定先への片方向のみ」削除します。そのため、コミュニケーションパスを完全に削除するには、2台のマシンそれぞれでコマンドを実行する必要があります。

## target2 で実行

1. 双方向に接続されたコミュニケーションパスが存在することを確認します。

```
[target2]# lkcli status -q
LOCAL    TAG    ID     STATE      PRIO   PRIMARY
```

```
MACHINE   NETWORK ADDRESSES/DEVICE      STATE      PRIO
target1   TCP     10.1.6.89/10.1.6.88   ALIVE         1
```

2. target2 から target1 へのコミュニケーションパスを削除します。

```
[target2]# lkcli commpath delete --laddr 10.1.6.89 --raddr 10.1.6.88 --dest ta
rget1
```

### コマンド引数

| 項目 | 入力値 |
|---|---|
| --laddr | 接続元となるローカルノード上の IP アドレス |
| --raddr | 接続先となるリモートノード上の IP アドレス |
| --dest | 接続先となるリモートノードのホスト名 |

3. target1 へのコミュニケーションパスが削除されたことを確認します。

```
[target2]# lkcli status -q
LOCAL     TAG    ID     STATE      PRIO   PRIMARY
```

## target1 で実行

4. target2へのコミュニケーションパスが残っていることを確認します。

```
[target1]# lkcli status -q
LOCAL     TAG    ID     STATE      PRIO   PRIMARY

MACHINE   NETWORK ADDRESSES/DEVICE      STATE      PRIO
target2   TCP     10.1.6.88/10.1.6.89   DEAD          1
```

5. target2 へのコミュニケーションパスを削除します。

```
[target1]# lkcli commpath delete --laddr 10.1.6.88 --raddr 10.1.6.89 --dest ta
rget2
```

6. target2 へのコミュニケーションパスが削除されたことを確認します。

```
[target1]# lkcli status -q
LOCAL     TAG    ID     STATE      PRIO   PRIMARY
```

以上で、コミュニケーションパスの作成･削除の手順は完了となります。

# 5.6.4.2. リソースの作成

## リソースの作成

保護するサービスやアプリケーションに対応するリソースを作成します。

> **注記:** コミュニケーションパスは「コミュニケーションパスの作成・削除」を参照して、事前に作成してください。

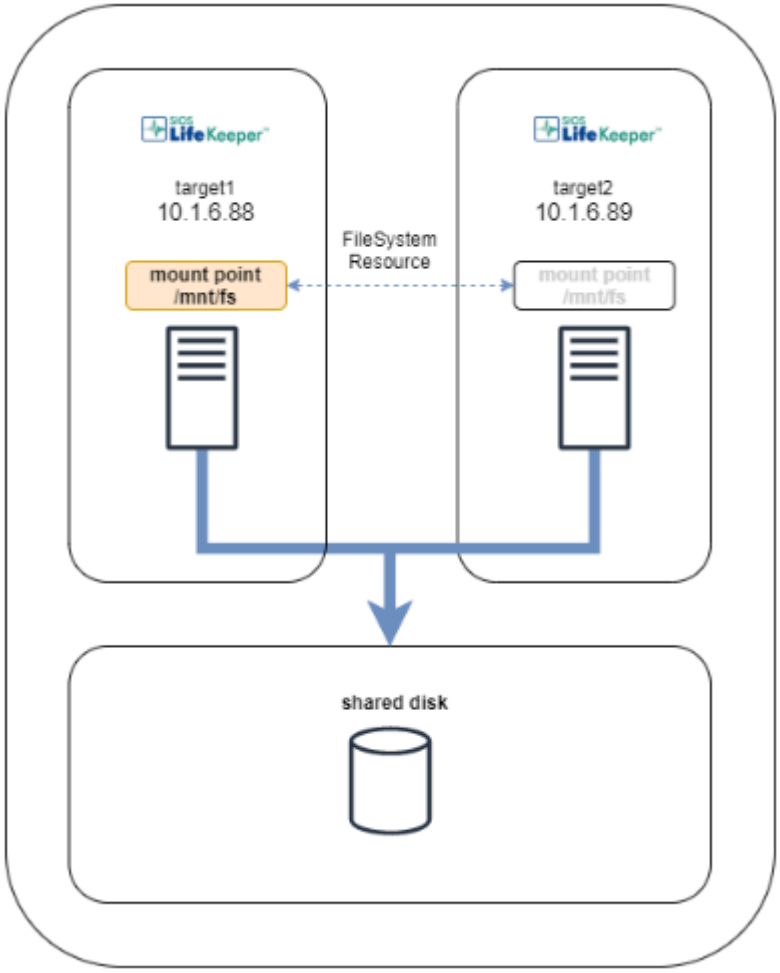
- ファイルシステムリソースの作成
- IP リソースの作成
- PostgreSQL リソースの作成
- DataKeeper リソースの作成

# 5.6.4.2.1. ファイルシステムリソースの作成

ファイルシステムリソースの作成手順を例示します。
ファイルシステムリソースは共有ストレージ上のファイルシステムをクラスターノード間で切り替えることを可能する機能を提供します。

## 構成図

本ガイドで作成する環境は、以下のような2ノード構成です。




LifeKeeper の制限事項にあたらないことを確認してください。

ファイルシステムリソースを作成するには、以下の条件を満たしている必要があります。

- 共有ストレージが物理的に接続され、各ノードから同じ状態で接続できること
- parted などのユーティリティを使い、 GUID パーティションテーブル (GPT) で共有ディスクのパーティションが切られていること
- mkfs などのユーティリティを使い、ファイルシステムが作成されていること
- 各 ノードでファイルシステムをマウント、アンマウントができること

### target1 で実行

1. ファイルシステムのマウント
   ファイルシステムリソース作成対象のファイルシステムを任意のディレクトリーにマウントします。
   例では /dev/sdb1 を /mnt/fs にマウントしています。/mnt/fs がマウントポイントになります。

```
[target1]# df
Filesystem              1K-blocks    Used Available Use% Mounted on
/dev/mapper/centos-root  14034944 6904924   7130020  50% /
devtmpfs                   929204       0    929204   0% /dev
tmpfs                      941312       0    941312   0% /dev/shm
tmpfs                      941312   25948    915364   3% /run
tmpfs                      941312       0    941312   0% /sys/fs/cgroup
/dev/sda1                 1038336  148528    889808  15% /boot
tmpfs                      188264       0    188264   0% /run/user/0
/dev/sdb1                 1044132   32992   1011140   4% /mnt/fs
```

2. リソースの作成
   以下のコマンドを実行します。

```
[target1]# lkcli resource create fs --tag fs-tag --mountpoint /mnt/fs
```

#### リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | タグ名 |
| --mountpoint | マウントポイント |

3. リソースの拡張
   さらに以下のコマンドを実行します。

```
[target1]# lkcli resource extend fs --tag fs-tag --dest target2
```

#### リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | 作成したリソースのタグ名 |
| --dest | バックアップノード名 |

4. リソースの確認
   リソースの作成および拡張が完了したら、次のコマンドを実行してください。
   リソースの情報が表示されます。

```
[target1]# lkcli status -q

LOCAL   TAG           ID                                      STATE      PRIO  PRI
MARY
target1 fs-tag        /mnt/fs                                 ISP           1  tar
get1
target1  device28856  36000c292eb0c693b2efb44ed56556636-1  ISP           1  tar
get1
target1   disk28786   36000c292eb0c693b2efb44ed56556636    ISP           1  tar
get1
```

ファイルシステムリソースを作成すると上記のように自動的に3つのリソースが依存関係を持った状態で作成されます。これらは、上位からマウントポイント、デバイス、ディスクを管理しており、それぞれ制御する対象が異なっています。これらは 3 つで 1 つの役割を担うため、依存関係の操作などで順序の変更などを行わないようにしてください。

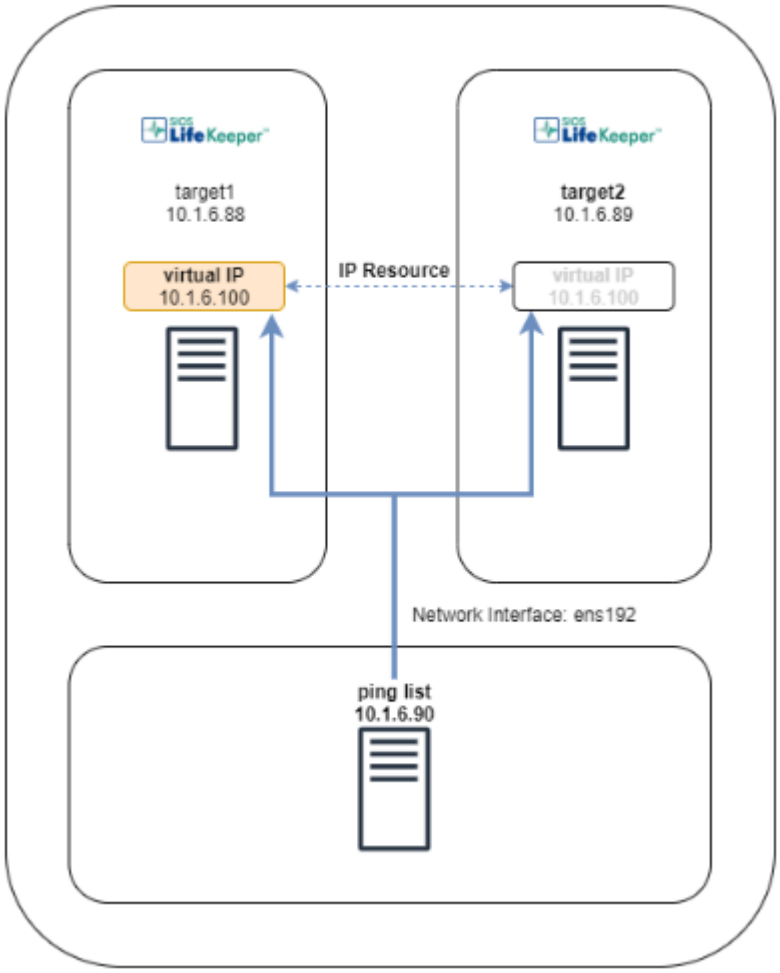
---

# 5.6.4.2.2. IP リソースの作成

IP リソースの作成手順を例示します。
IP リソースはクラスター間で切り替えることができる仮想 IP アドレスを作成し、それを保護します。

### 構成図

本ガイドで作成する環境は、以下のような2ノード構成です。




Ping で応答できる仮想 IP アドレス(上記では 10.1.6.100)を用意してください。
Ping 応答できるマシン(上記では 10.1.6.90)も用意してください。

Ping 応答は以下のようなコマンドで確認できます。

```
# ip -4 addr add 10.1.6.100/24 dev ens192

# ping 10.1.6.90 -b -c 1 -w 1 -I 10.1.6.100 10.1.6.90
```

結果に応答があること。

```
# ip -4 addr delete 10.1.6.100/24 dev ens192
```

### 制限事項

- 作成しようとしている仮想IPアドレスに重複は無いか確認してください。
- 仮想 IP アドレスと同じネットワークに、クラスターを構成するノード以外で Ping 応答できるマシンが存在するか確認してください。
  ※ IP リソースはブロードキャスト Ping、あるいはユニキャスト Ping を使用して応答確認を行います。そのため、クラスターを構成しているノード以外で Ping に応答できるマシンが必要です。

### target1 で実行

1. リソースの作成
   以下のコマンドを実行します。

```
[target1]# lkcli resource create ip --tag ip-tag --ipaddr 10.1.6.100
```

#### リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | タグ名 |
| --ipaddr | 仮想 IP アドレス |

2. Ping リストの設定
   以下のコマンドを実行して、Ping リストを設定します。

```
[target1]# lkcli resource config ip --tag ip-tag --pinglist 10.1.6.90
```

その後、リソースを In Service の状態にします。

```
[target1]# lkcli resource restore --tag ip-tag
```

3. リソースの拡張
   さらに以下のコマンドを実行します。

```
[target1]# lkcli resource extend ip --tag ip-tag --dest target2
```

#### リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | 作成したリソースのタグ名 |
| --dest | バックアップノード名 |

4. 拡張したリソースへ Ping リストの設定
   以下のコマンドを実行して、拡張したリソースに対しても、Ping リストを設定します。

```
[target1]# lkcli resource config ip --tag ip-tag --pinglist 10.1.6.90 --remot
e target2
```

5. リソースの確認
   リソースの作成および拡張が完了したら、次のコマンドを実行してください。
   リソースの情報が表示されます。

```
[target1]# lkcli status -q
LOCAL    TAG     ID             STATE     PRIO  PRIMARY
target1  ip-tag  IP-10.1.6.100  ISP          1  target1
```

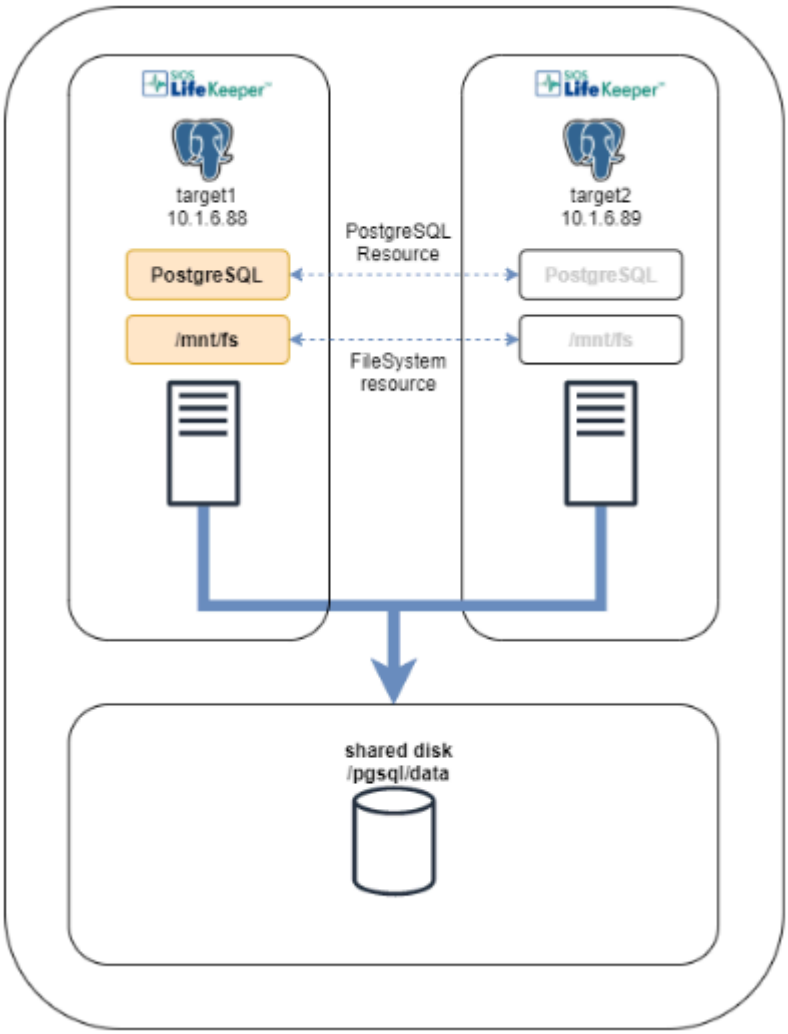
---

# 5.6.4.2.3. PostgreSQL リソースの作成

PostgreSQL リソースの作成手順を例示します。
PostgreSQL リソースは、LifeKeeper ノード上の PostgreSQL インスタンスを保護する機能を提供します。

## 構成図

本ガイドで作成する環境は、以下のような2ノード構成です。
事前に PostgreSQL ARK をインストールしてください。




本ガイドでの環境 (PostgreSQL)

- 本ガイドでは、PostgreSQL 9.2.24 を使用します。
- PostgreSQL のデータディレクトリーは、共有ストレージ上のファイルシステムに作成します。
- PostgreSQL データベース管理者ユーザーは、データベースの初期化時に作成される `postgres` を使用します。
- アクティブ / スタンバイ設定で、PostgreSQL リソースインスタンスを保護します。

※ ファイルシステムの制限事項にあたらないことを確認してください。詳細は こちら をご参照ください。

### target1、target2 で実行

1. PostgreSQL のインストール
   インストール後、PostgreSQL サービスの自動起動は、次のようにして無効にしてください。

```
# systemctl disable postgresql.service
```

### target1 で実行

2. ファイルシステムのマウント
   「ファイルシステムリソースの作成」の1. を参照して、データディレクトリーの作成先となるファイルシステムをマウントしてください。

3. データディレクトリーの作成
   PostgreSQL のデータディレクトリーを共有ディスク上に作成してください。

```
[target1]# mkdir -p /mnt/fs/pgsql/data
[target1]# chown -R postgres:postgres /mnt/fs/pgsql
```

   /usr/lib/systemd/system/postgresql.service ファイルに記載されている次のパスを、共有ディスク上のデータディレクトリーに変更してください。
   Environment=PGDATA=/mnt/fs/pgsql/data

4. データベースの初期化
   次のコマンドを実行し、データベースを初期化してください。
   データディレクトリーの下にデータベースが作成されます。

```
[target1]# postgresql-setup initdb
```

5. PostgreSQL サービスの起動
   以下のコマンドを実行し、PostgreSQL サービスを起動します。

```
[target1]# systemctl start postgresql.service
```

6. リソースの作成
   以下のコマンドを実行します。

```
[target1]# lkcli resource create pgsql --tag pgsql-tag --datadir /mnt/fs/pgsql/data --port 5432 --socket /tmp/.s.PGSQL.5432 --dbuser postgres --logfile /tmp/pgsql-5432.lk.log
```

### リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | タグ名 |

| | |
|---|---|
| --datadir | PostgreSQL データベースのデータがあるディレクトリーの絶対パス。 |
| --port | PostgreSQL が使用しているポート番号 |
| --socket | PostgreSQL が使用しているソケットのパス |
| --dbuser | PostgreSQL データベース管理者ユーザー名 |
| --logfile | PostgreSQL の起動や停止に使用する pg_ctl のログファイルの絶対パス。 |

7. リソースの拡張
   さらに以下のコマンドを実行します。

```
[target1]# lkcli resource extend pgsql --tag pgsql-tag --dest target2
```

リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | 作成したリソースのタグ名 |
| --dest | バックアップノード名 |

8. リソースの確認
   リソースの作成および拡張が完了したら、次のコマンドを実行してください。
   リソースの情報が表示されます。

```
[target1]# lkcli status -q
LOCAL    TAG            ID                                  STATE     PRIO  P
RIMARY
target1  pgsql-tag      target1.pgsql-5432                  ISP          1  t
arget1
target1   /mnt/fs       /mnt/fs                             ISP          1  t
arget1
target1    device17885  36000c292eb0c693b2efb44ed56556636-1  ISP          1  t
arget1
target1     disk17816   36000c292eb0c693b2efb44ed56556636    ISP          1  t
arget1
```

PostgreSQL リソースを作成すると、上記のように自動的に下位にファイルシステムリソースが依存関係を持った状態で作成されます。

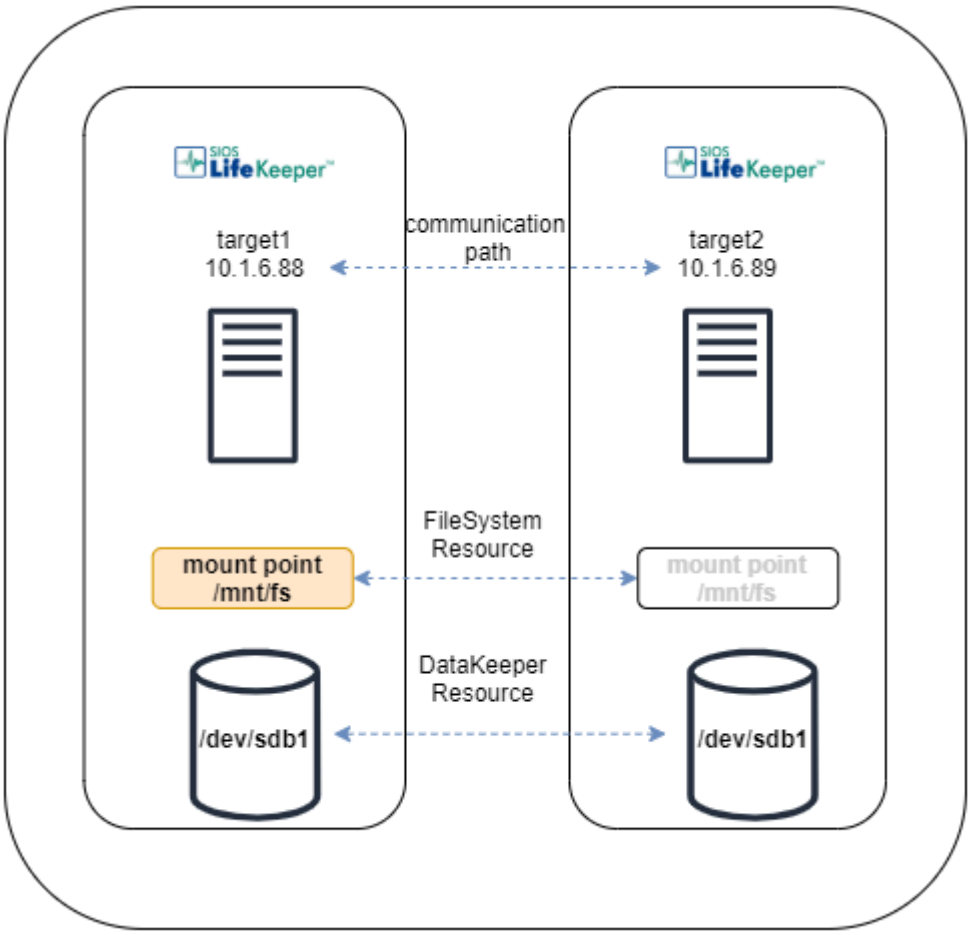
---

# 5.6.4.2.4. DataKeeper リソースの作成

DataKeeper リソースの作成手順を例示します。
DataKeeper リソースは、共有ストレージを使用せずに可用性の高いクラスタを構築する機能を提供します。

### 構成図

本ガイドで作成する環境は、以下のような 2ノード構成です。
新しいファイルシステムにミラーを作成し、LifeKeeper で保護します。




構成が以下の DataKeeper の要件を満たしていることを確認してください。
ハードウェアとソフトウェアの要件

### target1、target2 で実行

1. デバイスの確認

```
# parted /dev/sdb print
Model: VMware Virtual disk (scsi)
Disk /dev/sdb: 1074MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
```

```
Disk Flags:

Number  Start  End  Size  File system  Name  Flags
```

2. パーティションの作成
   デバイスの /dev/sdb にパーティションを作成します。

```
# parted -s /dev/sdb mklabel gpt
# parted /dev/sdb mkpart primary 0% 100%
# parted /dev/sdb print
Model: VMware Virtual disk (scsi)
Disk /dev/sdb: 1074MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name     Flags
 1      1049kB  1073MB  1072MB  ext4         primary
```

### target1 で実行

1. リソースの作成
   以下のコマンドを実行します。

```
[target1]# lkcli resource create dk --tag dk-tag --mode synchronous --hierarch
y new --device /dev/sdb1 --fstype xfs --mount_point /mnt/dk --fstag fs-tag
```

### リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | DataKeeper リソースのタグ名 |
| --mode | レプリケーションタイプ |
| --hierarchy | 作成するデータレプリケーションのタイプ |
| --device | ソースディスクまたはパーティション |
| --fstype | ファイルシステムタイプ |
| --mount_point | マウントポイント |
| --fstag | ファイルシステムリソースのタグ名 |

2. DataKeeper リソースの拡張
   さらに以下のコマンドを実行します。

```
[target1]# lkcli resource extend dk --tag dk-tag --dest target2 --mode synchro
nous --laddr 10.1.6.88 --raddr 10.1.6.89
```

リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | 作成した DataKeeper リソースのタグ名 |
| --dest | リソース階層を拡張するターゲットサーバのホスト名 |
| --mode | レプリケーションタイプ |
| --laddr | コミュニケーションパスに設定するローカルマシン上の IP アドレス |
| --raddr | コミュニケーションパスに設定するリモートマシン上の IP アドレス |

3. ファイルシステムリソースの拡張
   さらに以下のコマンドを実行します。

```
[target1]# lkcli resource extend fs --tag fs-tag --dest target2
```

リソースの設定値

| 項目 | 入力値 |
|---|---|
| --tag | 作成したファイルシステムリソースのタグ名 |
| --dest | リソース階層を拡張するターゲットサーバのホスト名 |

4. リソースの確認
   リソースの作成および拡張が完了したら、次のコマンドを実行してください。
   リソースの情報が表示されます。

```
[target1]# lkcli status -q

LOCAL   TAG           ID                                  STATE     PRIO  PRI
MARY
target1 fs-tag        /mnt/dk                             ISP          1  tar
get1
target1  dk-tag       36000c292eb0c693b2efb44ed56556636-1  ISP          1  tar
get1
```

   上記のように自動的に上位にファイルシステムリソースが依存関係を持った状態でDataKeeperリソースが作成されます。

5. ミラーリングの確認
   DataKeeper リソースを作成すると、全同期が実行されますので、全同期が完了するまで待ってください。以下のコマンドで状況が確認できます。Status が Fully Operational になっていれば全同期は

完了です。

```
[target1]# lkcli mirror status --tag dk-tag
```

### リソースの設定値

| 項目 | 入力値 |
| --- | --- |
| --tag | 作成した DataKeeper リソースのタグ名 |

### DataKeeper リソースの export/import

クラスターの設定を引き継いで複製する を参考にして DataKeeper とファイルシステムリソースを複製してください。
なお、import については、各ノードでファイルシステムをアンマウントした状態で実行してください。

# 5.6.4.3. ステータスを確認するには？

## ステータスを確認するには？

現在の LifeKeeper の状態を確認するには、 `lkcli status` を使用します。
表示されるメッセージには、実行したノード内のリソースやコミュニケーションパスのステータスが含まれています。

多くのケースでは、一部メッセージが省略された `lkcli status -q` でも、十分な情報を得られます。`q` オプションを指定した場合、上のブロックに現在のリソースの情報、下のブロックに現在のコミュニケーションパスの情報が表示されます。

```
# lkcli status -q
LOCAL    TAG             ID              STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

--remote オプションを利用することで、各ノードにログインすることなく、リソースのステータスを確認できます。その他、オプションや表示されるメッセージなどの詳細につきましては LifeKeeperCLI のマニュアル をご参照ください。

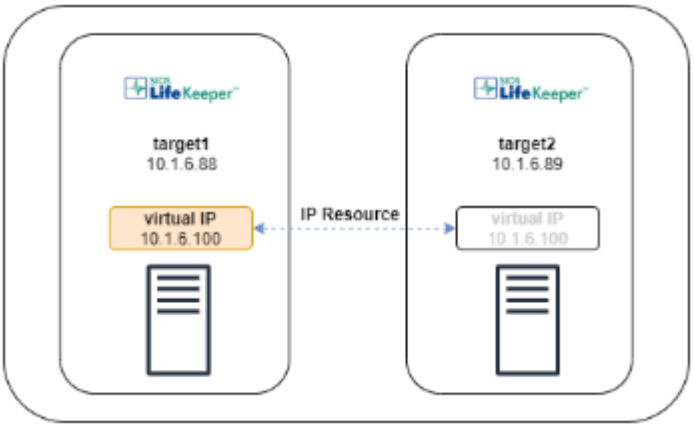
# 5.6.4.4. スイッチオーバーの動作確認

## スイッチオーバーの動作確認

ここでは target1 から target2 へスイッチオーバーを実施する手順を記載します。

### 構成

本ガイドでは LifeKeeper v9.5.0 を使用し、構成は以下の2ノードを想定しています。




コマンドは **target2** で実行してください。

1. target1 上でスイッチオーバーしたいリソースのステータスが ISP であることを確認します。

```
[target2]# lkcli status -q --remote target1
LOCAL    TAG             ID              STATE      PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. target2 上でスイッチオーバーしたいリソースのステータスが OSU であることを確認します。その際、そのリソースのタグ名をメモしておきます。

```
[target2]# lkcli status -q
LOCAL    TAG             ID              STATE      PRIO  PRIMARY
target2  ip-10.1.6.100  ip-10.1.6.100  OSU          10  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

3. target2 にリソースを切り替えます。

```
[target2]# lkcli resource restore --tag ip-10.1.6.100
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

4. target2 上でリソースが ISP であることを確認します。

```
[target2]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target2  ip-10.1.6.100  ip-10.1.6.100  ISP         10  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

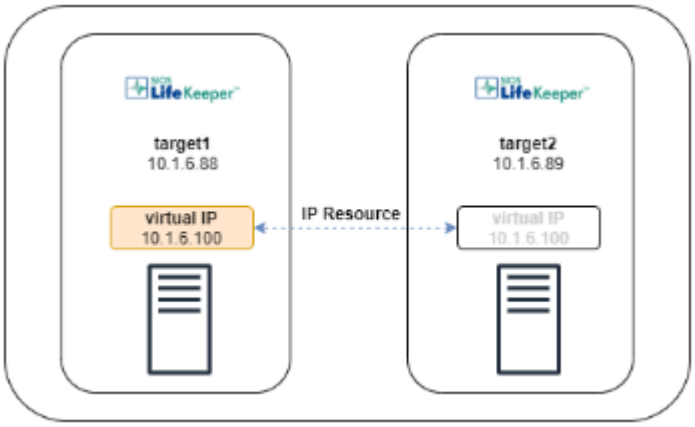
# 5.6.4.5. メンテナンス

## メンテナンス

ここでは、LifeKeeper で保護しているマシンやリソースでメンテナンスを実施する場合の手順を説明します。

### 構成

本ガイドでは LifeKeeper v9.5.0 を使用し、構成は以下の2ノードを想定しています。




## LifeKeeper で保護されているマシンのメンテナンス

ここでは、LifeKeeper が動作しているマシンのシャットダウンなど、LifeKeeper 本体・マシン内のすべてのリソースに影響を与えるメンテナンス作業を、target1 で実施する手順を記載します。

コマンドは **target1** で実行してください。

1. target1 でアクティブになっているリソースを、target2 に切り替えます。

    i. target1 ですべてのリソースのステータスを確認します。もし STATE が ISP になっているリソースがある場合は、そのリソースのタグ名をメモします。

```
[target1]# lkcli status -q
LOCAL    TAG             ID              STATE      PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP           1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE      PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE         1
```

    ii. target1 で ISP になっているリソースをひとつずつ target2 に切り替えます。なお、—remote オプションを使うことで、target1 上からコマンドを実行できます。

```
[target1]# lkcli resource restore --tag ip-10.1.6.100 --remote target2
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

iii. target1で、すべてのリソースのステータスを再確認します。
すべてのリソースで STATE が OSU になっていることを確認してください。

```
[target1]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  OSU          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. target1 の LifeKeeper を停止します。-fオプションをつけることで、LifeKeeper 停止時に行われる、リソースの停止処理をスキップします。

```
[target1]# lkcli stop -f
Removed /etc/systemd/system/lifekeeper-graphical.target.requires/lifekeeper.service.
Removed    /etc/systemd/system/lifekeeper-multi-user.target.requires/lifekeeper.service.
```

3. target1 でメンテナンスを実施してください。

4. target1 の LifeKeeper を起動します。

```
[target1]# lkcli start
Created symlink /etc/systemd/system/lifekeeper-graphical.target.requires/lifekeeper.service → /usr/lib/systemd/system/lifekeeper.service.
Created symlink /etc/systemd/system/lifekeeper-multi-user.target.requires/lifekeeper.service → /usr/lib/systemd/system/lifekeeper.service.
```

5. 必要に応じて target1 でリソースを ISP にします。

```
[target1]# lkcli resource restore --tag ip-10.1.6.100
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

参考資料： LifeKeeper が保護するシステムのメンテナンス

## LifeKeeper で保護されているリソースのメンテナンス

ここでは、マシン内の特定のリソースのみのメンテナンス作業を実施する場合の手順を記載します。

コマンドは **target1** で実行してください。

1. target1 でアクティブになっているリソースを、target2 に切り替えます。

   i. target1 ですべてのリソースのステータスを確認します。もし STATE が ISP になっているリソースがある場合は、ステップ ii 以降の手順でリソースをスイッチオーバーしてください。

```
[target1]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

   ii. target1 で ISP になっているリソースについて、target2 でのタグ名を確認します。

```
[target1]# lkcli status -q --remote target2
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target2  ip-10.1.6.100  ip-10.1.6.100  OSU         10  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target1  TCP     10.1.6.89/10.1.6.88  ALIVE        1
```

   iii. target1 で ISP になっているリソースをひとつずつ target2 に切り替えます。

```
[target1]# lkcli resource restore --tag ip-10.1.6.100 --remote target2
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

   iv. target1 で、すべてのリソースのステータスを再確認します。
   すべてのリソースで STATE が OSU になっていることを確認してください。

```
[target1]# lkcli status -q
LOCAL    TAG            ID             STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  OSU          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. OSU にしたリソースに対してメンテナンスを実施してください。

3. 必要に応じて target1 でリソースを ISP にします。

```
[target1]# lkcli resource restore --tag ip-10.1.6.100
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
```

参考資料： リソース階層のメンテナンス

## リソースの設定変更

ここでは、例としてIP リソースの設定を変更する手順を記載します。
IP リソース以外については「ARK 毎のサブコマンド 」を参考に、手順を適宜読み替えてください。

本ガイドでは LifeKeeper v9.5.0 を使用します。

1. 設定を変更したいリソースのタグ名を確認します。

```
# lkcli status -q
LOCAL    TAG             ID              STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2  TCP     10.1.6.88/10.1.6.89  ALIVE        1
```

2. タグ名をもとに、リソースの種類と現在の設定値を確認します。
   なお、リソースの種類は「ARK 毎のサブコマンド 」をご参照ください。

```
# lkcli resource info --tag ip-10.1.6.100
---
app: comm
priority: 1
properties:
  device: ens192
  ipaddr: 10.1.6.100
  netmask: 255.255.255.0
  pinglist: ''
  realip: 0
  restoremode: Enabled
  srcaddr: 0
switchback: INTELLIGENT
tag: ip-10.1.6.100
typ: ip
```

3. リソースの設定を変更します。ここでは `restoremode` を Disabled に変更します。

```
# lkcli resource config ip --tag ip-10.1.6.100 --restoremode Disabled
Performing restoremode change ...

restoremode change successful.
```

4. リソースの設定値が変化していることを確認します。

```
# lkcli resource info --tag ip-10.1.6.100
---
app: comm
priority: 1
properties:
  device: ens192
  ipaddr: 10.1.6.100
  netmask: 255.255.255.0
  pinglist: ''
  realip: 0
  restoremode: Disabled
  srcaddr: 0
switchback: INTELLIGENT
tag: ip-10.1.6.100
typ: ip
```

# 5.6.4.6. クラスターの設定を引き継いで複製する

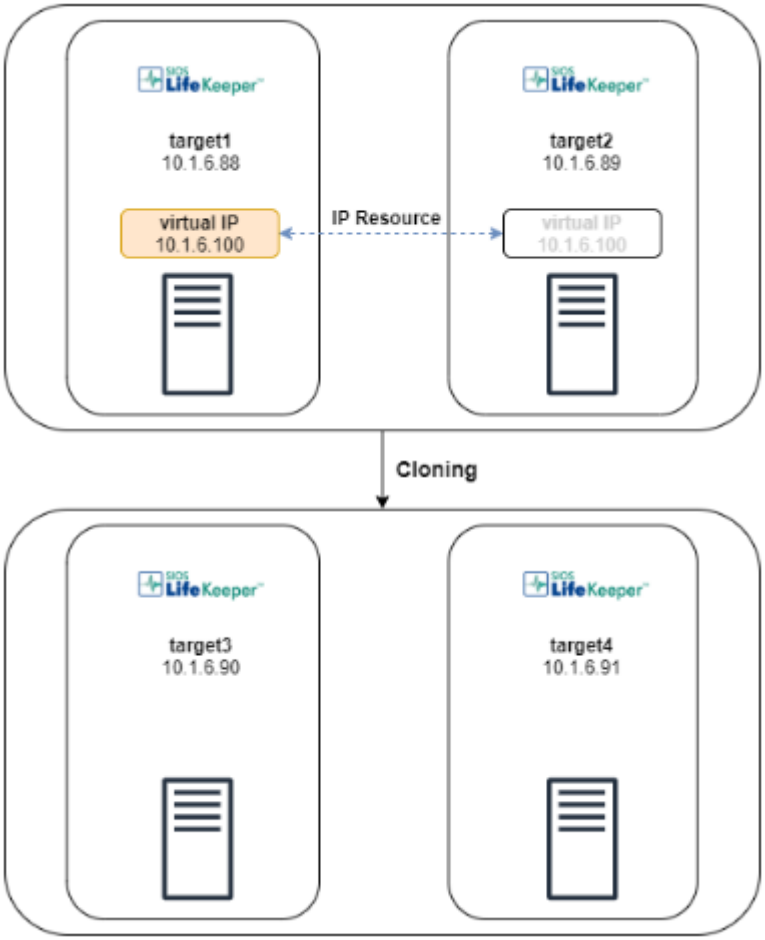
## クラスターの設定を引き継いで複製する

ここでは、コミュニケーションパスとリソースが設定されたクラスターをもとに、同じ設定を持つクラスターを複製する手順を記載します。

### 構成

本ガイドでは LifeKeeper v9.5.0 を使用します。

構成は以下の2ノードのクラスターを想定し、すでに LifeKeeper の設定が完了しているクラスターと、その複製先となる LifeKeeper がインストールされたクラスターを用意してください。なお、複製先の LifeKeeper には、コミュニケーションパス・リソースが設定されていない状態にしてください。
もし複数のクラスターを用意できない場合でも、インポート直前に LifeKeeper の設定を削除することで、動作を確認することができます。



## 注意点

export コマンドでは、出力できるリソースやコミュニケーションパスに一部制限があります。手順を実行する前に、サポートされた環境であるか以下の項目を確認してください。
制限事項の詳細につきましては、 LKCLI のマニュアル をご参照ください。

- エクスポートするクラスターのコミュニケーションパスのプロトコルが、すべてTCP/IPであることを確認してください。

```
# lkcli status -q
LOCAL    TAG             ID              STATE     PRIO  PRIMARY
target1  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target1

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target2    TCP   10.1.6.88/10.1.6.89  ALIVE        1
```

- エクスポートするクラスターのすべてのリソースが、LKCLI でサポートされている ARK であることを確認してください。なお、リソースの種類やサポートしている ARK であるかは ARK一覧 をご参照ください。

```
# lkcli resource info --tag ip-10.1.6.100
---
app: comm
priority: 1
properties:
  device: ens192
  ipaddr: 10.1.6.100
  netmask: 255.255.255.0
  pinglist: ''
  realip: 0
  restoremode: Disabled
  srcaddr: 0
switchback: INTELLIGENT
tag: ip-10.1.6.100
typ: ip
```

## 手順

**target1** でコマンドを実行してください。

1. LifeKeeper の設定が完了しているクラスターで、現在の設定をファイルに保存します。

```
[target1]# lkcli export > src_settings.yml
```

2. エクスポートされたファイルを複製先のクラスターにコピーします。

**target3** でコマンドを実行してください。

3. エクスポートされたファイルの IP アドレスとホスト名を複製先の環境にあわせて書き換えてください。

```
[target3]# sed -e "s/10\.1\.6\.88/10\.1\.6\.90/" -e "s/target1/target3/" -e
"s/10\.1\.6\.89/10\.1\.6\.91/" -e "s/target2/target4/" src_settings.yml > des
t_settings.yml
```

4. LifeKeeper の複製先のクラスターでコミュニケーションパス・リソースがまだつくられていないことを確認します。なおリソースの確認は、クラスター内のすべてのノードで確認してください。
もしコミュニケーションパス・リソースが存在する場合は `lkcli clean --mode all` ですべて削除することができます。なお、clean コマンドを実行すると、設定を元に戻すことはできませんのでご注意ください。

```
[target3]# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY
```

**target3・target4** でコマンドを実行してください。

5. LifeKeeper の複製先でコミュニケーションパスをインポートして、コミュニケーションパスが作成されることを確認します。なおこのコマンドは、クラスター内のすべてのノードで実行してください。

```
# lkcli import commpath --file dest_settings.yml
Performing commpath 'target3:10.1.6.90/10.1.6.91' create...
Commpath 'target3:10.1.6.90/10.1.6.91' created successful.
```

```
# lkcli status -q
LOCAL    TAG   ID    STATE     PRIO  PRIMARY

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target4    TCP   10.1.6.90/10.1.6.91  ALIVE        1
```

※ コミュニケーションパスのステータスは、2つのノード間で双方向に作成されたとき ALIVE となります。

6. LifeKeeper の複製先でリソースをインポートして、リソースが作成されることを確認します。なおこのコマンドは、クラスター内で1度だけ実行してください。

```
# lkcli import resource --file dest_settings.yml
BEGIN create of "ip-10.1.6.100"
```

```
LifeKeeper application=comm on target1.
LifeKeeper communications resource type= ip on target1.
Creating resource instance with id ip-10.1.6.100 on machine target1
Resource successfully created on target1
BEGIN restore of "ip-10.1.6.100"
END successful restore of "ip-10.1.6.100"
END successful create of "ip-10.1.6.100".
Removing ping list for subnet 172.31.0.0...
Performing restoremode change ...

restoremode change successful.
Building independent resource list
Checking existence of extend and canextend scripts
Checking extendability for ip-10.1.6.100
Pre Extend checks were successful
Extending resource instances for ip-10.1.6.100
Creating dependencies
Setting switchback type for hierarchy
Creating equivalencies
LifeKeeper Admin Lock (ip-10.1.6.100) Released
Hierarchy successfully extended
Removing ping list for subnet 172.31.0.0...
```

7. コミュニケーションパスとリソースが作成されていることを確認します。

```
# lkcli status -q
LOCAL    TAG              ID           STATE     PRIO  PRIMARY
target3  ip-10.1.6.100  ip-10.1.6.100  ISP          1  target3

MACHINE  NETWORK ADDRESSES/DEVICE     STATE     PRIO
target4  TCP     10.1.6.90/10.1.6.91  ALIVE        1
```

# 6. アプリケーションリカバリーキット

SIOS Protection Suite for Linux Application Recovery Kits (ARKs) は特定のアプリケーションをSPS上で管理、制御するためのツールとユーティリティを含んでいます。今バージョンのSPSで利用できるARKは以下の通りです。

Apache Recovery Kit 管理ガイド

DB2 Recovery Kit 管理ガイド

Recovery Kit for EC2 管理ガイド

Generic ARK for Load Balancer probe reply

LVM Recovery Kit 管理ガイド

IP Recovery Kit 管理ガイド

MySQL Recovery Kit 管理ガイド

WebSphere MQ Recovery Kit 管理ガイド

NAS Recovery Kit 管理ガイド

NFS Recovery Kit 管理ガイド

Oracle Recovery Kit 管理ガイド

PostgreSQL Recovery Kit 管理ガイド

Postfix Recovery Kit 管理ガイド

Recovery Kit for Route 53 管理ガイド

Samba Recovery Kit 管理ガイド

SAP Recovery Kit 管理ガイド

SAP HANA Recovery Kit管理ガイド

SAP MaxDB Recovery Kit 管理ガイド

Sybase Recovery Kit 管理ガイド

VMDK as Shared Storage Recovery Kit 管理ガイド

# 6.1. Apache Recovery Kit 管理ガイド

SIOS Protection Suite (SPS) for Linux Apache Web Server Recovery Kit は、SPS 環境で Apache Web Server　ソフトウェアの障害回復能力を向上します。

このガイドでは次の事項について説明します。

- SPS ドキュメンテーション：SPS for Linux のドキュメンテーションの全リスト、および情報が入手可能な場所。

- 要件：リカバリソフトウェアのインストールおよびセットアップを行うには、サーバが一定のハードウェア / ソフトウェア要件を満たしている必要があります。SPS Apache Recovery Kit のインストール / アンインストール方法については、SPS インストールガイドを参照してください。

- Recovery Kit　の設定：必要とする保護と柔軟性が得られるように SPS を設定するには、設定ルールを把握している必要があります。設定を適切にプランニングするには、ネットワーク設定、インターフェースの選択、ユーザシステムのセットアップ、階層オプション、Apache 設定作業について理解することが必要です。このセクションでは、設定のプランニングだけでなく、設定の例を示し、Recovery Kit を設定するために必要な特定の作業について説明します。

- トラブルシューティング：このセクションには、情報メッセージとエラーメッセージ、および推奨される対処方法の一覧を示します。

# 6.1.1. ドキュメンテーションとリファレンス

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション (LifeKeeper GUI 内の [Help] メニューから参照できます)。
- SPS for Linux インストールガイド

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下のサイトからオンラインで使用できます。

http://docs.us.sios.com

## 参考ドキュメント

以下に、Apache Web Server アプリケーションおよび SPS Apache Recovery Kit に関連する参考ドキュメントの一覧を示します。

- Apache のオンラインドキュメンテーション
- Apache: The Definitive Guide, 2nd Edition, Ben Laurie and Peter Laurie, O'Reilly & Associates, Inc. 1999

# 6.1.2. 要件

Apache Recovery Kit のインストールやアンインストールを試みる前に、このパッケージのハードウェア / ソフトウェア要件と、インストールとアンインストールの手順を理解しておく必要があります。

## キットのハードウェア / ソフトウェア要件

LifeKeeper Apache Recovery Kit をインストールして設定する前に、使用している設定が以下の要件を満たしていることを確認してください。

- サーバ。Recovery Kit を使用するには、SPS for Linux テクニカルドキュメンテーションとSPS リリースノートに記載されている LifeKeeperの要件に従って設定した、サポートされるコンピュータが 2 台以上必要です。これらのドキュメントは、SIOS のテクニカルドキュメンテーションサイト (docs.us.sios.com) にあります。

- LifeKeeper ソフトウェア。各サーバに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。特定の LifeKeeperの要件については、SPS リリースノートおよび SPS for Linux テクニカルドキュメンテーションを参照してください。

- LifeKeeper IP Recovery Kit。同じバージョンの Recovery Kit を各サーバにインストールする必要があります。

- IP ネットワークインターフェース。各サーバは、イーサーネット TCP/IP をサポートするネットワークインターフェースを 1 つ以上必要とします。IP スイッチオーバが正しく動作するには、ローカルネットワークに接続されているユーザシステムが標準の TCP/IP 仕様に準拠している必要があります。
  注記: 各サーバが必要とするネットワークインターフェースが 1 つだけであっても、異種メディア要件、スループット要件、単一障害点の排除、ネットワークのセグメンテーションといった多くの理由のために、複数のインターフェースを使用すべきです。

- TCP/IP ソフトウェア。各サーバは TCP/IP ソフトウェアも必要とします。

- Apache ソフトウェア。LifeKeeper および LifeKeeper Apache Web Server Recovery Kit を設定する前に、各サーバに Apache Web Server、および使用する DSO (動的共有モジュール) をインストールし、設定しておく必要があります。Web サーバソフトウェアのすべてのパッケージについて、各サーバに同じバージョンをインストールする必要があります。最新リリースの互換性と発注情報については、SPS リリースノートを参照するか、営業担当者に問い合わせてください。

LifeKeeper Apache Recovery Kit のインストール / アンインストール方法については、SPS インストールガイドを参照してください。

# 6.1.3. LifeKeeper における Apache Web Server の設定

本セクションでは、LifeKeeper における典型的な Apache Web Server の設定の定義および例と、Apache Web Server の設定を開始する前に考慮すべき情報について説明します。

LifeKeeper Core のリソース階層を設定する手順については、SPS for Linux テクニカルドキュメンテーションを参照してください。

# 6.1.3.1. 設定の定義と例

Apache Web Server は、httpd デーモンの複数インスタンスの同時実行をサポートしています。LifeKeeper の各 Apache Web Server 階層は、固有の「サーバルート」ディレクトリを持つ個別の Apache インスタンスに対応します。各インスタンスは、「仮想ホスト」を使用するように設定されているかどうかにより、1 つ以上の Web サイトをサポートできます。

主に、サーバルートディレクトリは 1 つの Apache Web Server インスタンスを定義します。これは、Web インスタンスの設定方法を指定する conf/httpd.conf 設定ファイルがこのディレクトリに入れられるからです。このファイル内にある Apache 設定ディレクティブにより、このインスタンスのログファイル、Web ドキュメント、その他の設定ファイルなどの場所、および使用する IP/ ドメイン名のアドレスが指定されます。

LifeKeeper ファイルシステム (共有ストレージを使用) を使用するかどうかに基づいて、Apache Web Server の設定を分類すると便利です。1 つの共有ファイルシステムを、サーバルートディレクトリ (設定ファイル conf/httpd.conf とともに使用) やドキュメントルートディレクトリ (さらにオプションで httpd 実行可能ファイル自体) に使用できます。特定の Apache インスタンスについて、ローカル設定と共有設定のいずれを使用するかは、2 つの主な要因によって決まります。その 1 つは、設定ファイルや Web サイトのドキュメントの複数の同一コピーを個別に管理することの困難さであり、もう 1 つは、2 台以上のサーバで共有 (またはミラー化) できるストレージの可用性とアクセス容易性です。ただし、1 台のサーバ上に、ローカルの Apache インスタンスと共有の Apache インスタンスの両方を設定できることに注意してください。

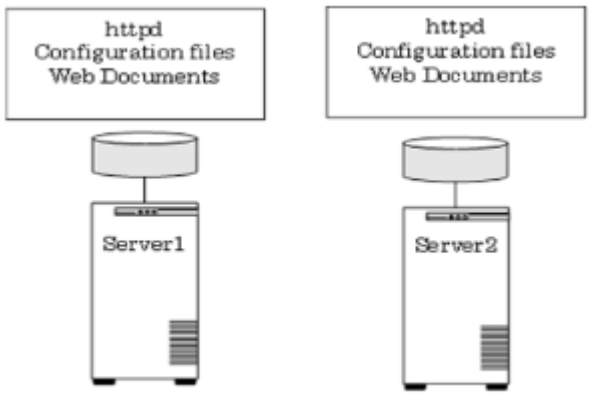
以降のセクションでは、LifeKeeper 環境でのローカル と 共有 の Apache Web Server 設定の例を示し、それぞれの主な特徴の概要を説明します。

## ローカル設定

一般的なローカル設定では、複数のサーバ間で共有されるものはありません。Apache Web Server の設定ファイル、Web ドキュメント、DSO モジュール (およびその設定ファイル)、および httpd 実行可能ファイルのそれぞれについて、同一コピーが各サーバの正確に同じ位置に存在します。各サーバに Apache コンポーネントの同一コピーを維持するのは、Apache 管理者の責任です。

設定ファイルにより、各 Web サイトに IP アドレス、または特定の IP アドレスをマップするドメインアドレスが割り当てられます。それぞれのアドレスについて LifeKeeper の IP アドレスが作成され、Apache リソース階層に追加されます。あるサーバから別のサーバに Apache 階層がスイッチオーバすると、元のサーバではこの特定の httpd インスタンスが停止し、IP アドレスが非アクティブになります。次に、スイッチオーバ先のサーバで、IP アドレスがアクティブになり、インスタンスが開始されます。その後、TCP/IP 経由でクライアントがスイッチオーバ先のサーバ上にある同一の Web サイトに自動的に接続されます。

図 1: ローカル設定




設定に関する注記:

- 図 1 は、共有ファイルシステムに何も配置しないローカル設定の例です。
- 各サーバの同じ位置 (通常は /usr/bin/httpd) に、同一バージョンの Apache Web Server 実行可能ファイルがあります。
- 各サーバは同一のサーバルートディレクトリを持ち、そこには各インスタンスの設定ファイルの同一コピーが配置されます。
- 各サーバは同一のドキュメントルートディレクトリを持ち、そこには各インスタンスの Web ドキュメントの同一コピーが配置されます。
- DSO モジュールを使用する場合は、各サーバの同じ場所にその同一コピーがあります。

Server 1 での Apache Web Server リソース階層の作成:

| | |
|---|---|
| サーバ: | Server1 |
| Web Server バイナリの位置: | /usr/sbin/httpd |
| Web Server のルートディレクトリ: | /home/www/examples/instance1/ |
| ルートタグ | apache-www.examples.instance1 |

Apache Web Server リソース階層の Server 2 への拡張:

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | apache-www.examples.instance1 |
| ターゲットサーバ | Server2 |

| ターゲットの優先順位: | 10 |
|---|---|

Apache リソース階層を 1 台以上の追加サーバに拡張するときには、ローカル設定と共有設定のいずれの場合でも、すべてのサーバで同一の Web Server バイナリの位置、および同一の Web Server のルートディレクトリを使用する必要があることに注意してください。詳細については、前述の説明、および[Apache Web Server の設定に関する考慮事項] のセクションを参照してください。また、階層の拡張時に、Apache リソース階層の一部を構成する依存リソースがすべて、LifeKeeper により拡張されます。

## 共有設定

一般的な共有設定では、サーバルートディレクトリとドキュメントルートディレクトリはすべて、同一の共有ファイルシステムに存在します。同一の設定ファイルと Web ドキュメントが複数のサーバで共有されるので、各サーバに同一コピーを維持する必要はありません。DSO モジュールを使用する場合は、必要となる設定ファイルやリソースとともに、同じ共有ファイルシステムに配置できます。

Web ドキュメントのみを共有ファイルシステムに配置できることに注意してください。この方法では、サーバルートディレクトリがローカルなので一般的なローカル設定に似ていますが、階層には共有ファイルも含まれます。

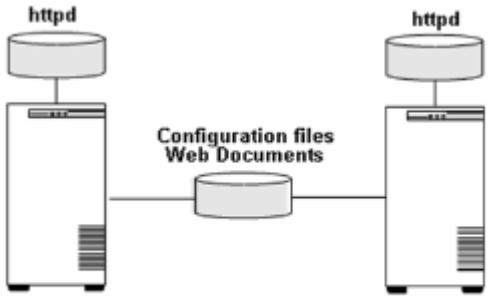
この Apache リソース階層で Apache 実行可能ファイルの特定バージョンまたは個別のコピーを使用する場合は、この実行可能ファイルも共有ファイルシステムに配置できます。この実行可能ファイルは、このインスタンスにのみ使用可能になります。この場合は、Web Server バイナリの位置の入力を要求されたときに、共有ファイルシステム上にある httpd 実行可能ファイルのフルパスを入力します。

使用できる共有ファイルシステムは 1 つのみであることに注意してください。これにより、確実に、共有ストレージにあるすべての必須コンポーネントが同時に利用可能になります。共有ファイルシステム上の Web Server バイナリの位置を使用する場合は、同じ共有ファイルシステム上にある Web Server のルートディレクトリも選択する必要があります。また、このサーバルート用に設定した DocumentRoot ディレクトリがすべて、同じ共有ファイルシステム上に存在する必要があります。同様に、共有ファイルシステム上の Web Server のルートディレクトリを選択するときには、DocumentRoot ディレクトリがすべて、同じ共有ファイルシステム上に存在する必要があります。バイナリとサーバルートが共有ファイルシステムに存在しないが、いずれかの DocumentRoot ディレクトリが共有される場合、すべての DocumentRoot ディレクトリを同じファイルシステム上で共有する必要があります。

これらの規則は、以下のように要約できます。

- Apache 実行可能ファイルを共有する場合は、サーバルートディレクトリを共有する必要があります。
- サーバルートディレクトリを共有する場合は、すべての DocumentRoot ディレクトリを共有する必要があります。
- DocumentRoot ディレクトリを 1 つでも共有する場合は、すべての DocumentRoot ディレクトリを共有する必要があります。
- 個々の Apache リソース階層について、共有ファイルシステムを 1 つのみ使用できます。

図 2:共有設定




設定に関する注記:

- 図 2 は、設定ファイルと Web ドキュメントを共有する共有設定の例です。
- Web ドキュメントのみを共有ファイルシステムに配置できます。この方法は一般的なローカル設定に似ていますが、階層には共有ファイルシステムも含まれる点が異なります。
- DSO モジュールを使用する場合は、必要な設定ファイルやリソースとともに、共有ファイルシステムに配置できます。

Server 1 での Apache Web Server リソース階層の作成:

| | |
|---|---|
| サーバ: | Server1 |
| Web Server バイナリの位置: | /usr/sbin/httpd<br>または<br>/shared/example/instance2/bin/httpd |
| Web Server のルートディレクトリ: | /shared/example/instance2 |
| ルートタグ | apache-shared.example.instance2 |

Apache Web Server リソース階層の Server 2 への拡張:

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | apache-shared.example.instance2 |

| | |
|---|---|
| ターゲットサーバ | Server2 |
| ターゲットの優先順位: | 10 |

# 6.1.3.1.1. アクティブ / スタンバイ設定とアクティブ / アクティブ設定

Apache Web Server は、LifeKeeper ではアクティブ / アクティブアプリケーションと呼ばれます。これは、いつでも複数の Apache インスタンスをサーバ上で実行できるということです。例えば、2 台のサーバが Apache インスタンスを 1 つずつ実行していて、1 台のサーバに障害が発生した場合、このサーバの Apache インスタンスがもう 1 台のサーバにフェイルオーバでき、さらにもともと実行していたインスタンスの実行も続行できます。一部のアプリケーションではこの機能をサポートしていないため、アプリケーションのインスタンスごとにサーバを使用可能にしておく必要があります。そのようなアプリケーションは、アクティブ / スタンバイアプリケーションと呼ばれます。いずれの設定も使用できるアプリケーションもあります。

Apache を「アクティブ / スタンバイ」モードで実行したい状況が発生することがあります。特に、例えばサーバの 1 台を Apache の実行専用にする場合です。この特殊な場合では、バックアップサーバで何も実行しないように、標準的な Apache のデフォルトインストールの自動起動を無効にする必要があります。

1 台以上の特定サーバで Apache インスタンスを手動で in service にすることで、任意に負荷を分散できます。さらに、各インスタンスのサーバ優先順位を調整することで、障害が発生したときに、Apache インスタンスが最終手段として特定サーバにフェイルオーバするように設定することも、複数のサーバにフェイルオーバして負荷を分散するように設定することもできます。

クラスタ内のすべてのサーバで Apache の自動起動を無効にした場合、このインスタンスで LifeKeeper の IP アドレスを使用するように設定することで、デフォルトサーバのルートディレクトリ「/etc/http」を LifeKeeper の Apache リソース履歴に使用できます。また、ドキュメントのルートディレクトリ用に共有ファイルシステムを使用できます。ただし、これはアクティブ / スタンバイ設定 (前述) なので、通常の方法ではデフォルトのインスタンスを起動できなくなります。さらに、サーバのルートは一意である必要があるため、デフォルトサーバのルートディレクトリを複数の履歴に使用することはできません。

# 6.1.3.2. Apache Web Server の設定に関する考慮事項

Apache リソース階層を作成する前に、Apache Web Server アプリケーションについて以下の設定を完了したことを確認する必要があります。

1. ディストリビューション付属の Apache パッケージをインストールした場合、通常はシステムの起動時に自動起動するよう設定されるため、LifeKeeper の保護と競合します。 競合を回避するためには、各ディストリビューションのマニュアルを参照して、自動起動を無効化してください。

   **注記: SuSE で Apache を使用する場合**: デフォルト設定を使用して Apache を SuSE にインストールする場合、httpd.conf 設定ファイルは ServerRoot のサブディレクトリ conf に配置されません。この場合は、設定ファイルを /etc/httpd/conf ディレクトリに再配置する必要があります。

2. LifeKeeper の個々の Apache Web Server 階層について、個別のルートディレクトリを作成する必要があります。この「サーバルート」ディレクトリは、Apache の「ServerRoot」設定とコマンドラインパラメータに対応します。LifeKeeper の各 Apache Web Server 階層は、一意の Apache インスタンスおよび関連付けられたサーバルートディレクトリに対応します。サーバルートディレクトリは、すべてのサーバ上で同一であり、特定の Apache 階層用に設定する必要があることに注意してください。Web サイトのすべての設定ファイルの情報は、LifeKeeper ソフトウェアから検出とアクセスができるように、サーバルートからの標準の場所 (conf/httpd.conf) に配置する必要があります。

3. BindAddress または Listen のディレクティブを使用して特定の LifeKeeper の IP アドレスをリスンするように、すべての Web サイト (仮想ホスト) を設定する必要があります。これらの LifeKeeper が保護する IP アドレスは、Apache 階層を作成する場所にあらかじめ作成し、in service 可能にしておく必要があります。LifeKeeper が保護する IP アドレスは、Apache リソース階層に自動的に追加されます。

   LifeKeeper の共有ファイルシステムを使用する場合は、Apache 階層を作成する前に、ファイルシステムの作成に必要な準備をすべて行う必要があります。特に、サーバ上のApache 階層を作成する場所に、ファイルシステムをマウントする必要があります。LifeKeeper のファイルシステム階層をあらかじめ作成していない場合、このファイルシステム階層は Apache 階層とともに自動作成され、その後 Apache リソース階層に結合されます。

   仮想ホストの設定の詳細については、Apache Web Server のドキュメンテーションを参照してください。前述したように、すべての Apache インスタンスが、LifeKeeper が保護する特定のアドレスをリスンするように設定する必要があります。例えば、IP ベースの仮想ホストとネームベースの仮想ホストを組み合わせたインスタンスの設定ファイルには、以下のようなディレクティブがあります。

```
User webuser
Group webgroup
ServerName localhost

Listen 172.17.100.55:8000
NameVirtualHost 172.17.100.55:8000
```

```
Listen 172.17.100.56:80

<Virtualhost site.name_one:8000>
ServerName site.name_one
DocumentRoot /shared/site/name_one
</VirtualHost>

<VirtualHost site.name_two:8000>
ServerName site.name_two
DocumentRoot /shared/site/name_two
</VirtualHost>

<VirtualHost 172.17.100.56:80>
ServerName site.ip
DocumentRoot /shared/site/ip
</VirtualHost>
```

4. Apache インスタンスに対する SSL サポートを有効にしている場合、LifeKeeper が保護する適切な IP アドレスを使用するように SSL Listen ディレクティブ (多くの場合、個別の ssl.conf ファイル内にある) を設定する必要があります。この設定を行わない場合、IP アドレス 0.0.0.0 が LifeKeeper で保護されていないことを示すエラーが発生して、Apache 階層の作成に失敗します。一部の Linux 配布の Apache 設定ファイルでは、デフォルトで SSL サポートが有効になっていることに注意してください。

   例えば、/etc/httpd/conf.d/ssl.conf にあるデフォルトの SSL 設定ファイルで、エントリ

   ```
   Listen 0.0.0.0:443
   ```

   を以下のように変更します。

   ```
   Listen 172.17.100.55:443
   ```

5. ローカル設定の場合、プライマリサーバとすべてのバックアップサーバの同じ場所に Apache をインストールして設定し、同一の (または同等の) 設定ファイルをすべてのサーバ上の同じサーバルートディレクトリに設定する必要があります。また、すべてのサーバ上にすべてのドキュメントルートディレクトリが存在し、同一のファイルを持つ必要があります (「設定と定義の例」のローカル設定参照)。

6. 共有設定では通常、LifeKeeper の共有ファイルシステム上にサーバルートディレクトリを設定します。使用できる共有ファイルシステムは 1 つのみであることに注意してください。これにより、確実に、共有ストレージにあるすべての必須コンポーネントが同時に利用可能になります。このため、すべてのドキュメントルートディレクトリが、同じ共有ファイルシステムのサブディレクトリである必要があります。ただし、サーバルートディレクトリ自体のサブディレクトリにする必要はありません。Apache 実行可能ファイルも同じ共有ファイルシステムに配置できますが、この実行可能ファイルは、この特定の Apache リソース階層のみが使用できます。

   **注記**: 共有ストレージを使用する目的で、サーバルートディレクトリを共有ファイルシステムに配置

する必要はありません。設定ファイルなどを入れるローカルのサーバルートディレクトリを選択でき、ドキュメントルートディレクトリのみを共有ファイルシステムに配置できます。ただし、すべてのサーバ上に (前述のローカル設定について)、同一のサーバルートディレクトリ、および同一の (または同等の) 設定ファイルを設定する必要があります。また、すべてのドキュメントルートディレクトリが同じ共有ファイルシステム上に存在する必要があります (「設定と定義の例」の共有設定参照)。

7. 一部の Web サイトの実装では、特定の機能に対する Apache のサポートを拡張するために、DSO (Dynamic Shared Object: 動的共有オブジェクト) モジュールを活用しています。例えば、PHP や Perl の機能を実装するモジュールがあります。実行時に Apache コアから、これらのモジュールのロードとアクセスができます。モジュールを使用する場合は、クラスタ内のすべてのサーバでモジュールを同一設定にする必要があります。Linux プラットフォームでモジュールを使用するように Apache を設定する方法については、モジュールパッケージのドキュメンテーション、およびベンダ提供のドキュメンテーションを参照してください。使用するモジュールおよびそのリソースによっては、フェイルオーバを適切に実行するために、一部のオブジェクトを共有ストレージに配置する必要があります。場合によっては、Generic Application Recovery Kit、またはカスタムリカバリキットを使用して、個別モジュールの保護が必要になることがあります。

8. SSL (Secure Sockets Layer: セキュアソケットレイヤ) モジュールを Apache とともに使用する場合は、サーバをパスワードで保護しないことが重要です。Web サーバがパスワードで保護されている場合、デーモンを起動するたびに管理者がパスワードをプロンプトに入力しなければなりません。この手動操作は、リカバリ時間が重要な HAクラスタ環境とは矛盾するため、LifeKeeper はパスワードで保護されたインスタンスをサポートしません。以下のコマンドを使用してパスワードを削除してください。

   ```
   openssl rsa -in server.key -out unprotected_server.key
   ```

   プロンプトが表示されたら、サーバキーパスワードを入力してください。サイトのセキュリティを維持するために、必ずファイルはルートでのみ読み取り可能にしてください。

   ```
   chmod 400 unprotected_server.key
   ```

   Apache インスタンスの階層作成時に、Recovery Kit により、リソースがパスワードで保護されていないことがチェックされます。リソースがパスワードで保護されている場合、階層の作成に失敗し、エラーメッセージが表示されます。ただし、インスタンスが別のサーバに拡張されている場合、Recovery Kit はバックアップサーバでパスワードの保護をチェックしません。拡張する階層がパスワードで保護されていないことを確認する必要があります。

   サーバキーファイル (Apache の設定ファイルの SSLCertificateKeyFile ディレクティブで指定) は、同じ名前を持ち、クラスタ内のすべてのサーバ上で同じ場所にある必要があります。

   **注記:** LifeKeeper が使用する httpd プロセスの PID ファイル名は、以下の形式を取ります。

   ```
   「/var/run/httpd.<TAG name>.pid」
   ```

   この PID ファイル名は、OS が使用するデフォルトの PID ファイル名とは異なります。この PID ファイルを参照する必要がある場合 (ログのローテーションなど)、LifeKeeper の PID ファイル名と形式に注意してください。

9. SuSE環境で、"/etc/sysconfig/apache2"に定義されている"APACHE_SERVER_FLAGS"を設定する場合は、フラグ名の前に"-D"を記述してください。

   **例:** `APACHE_SERVER_FLAGS=”-D SSL”`

   "-D"がない場合、リソースの作成や起動に失敗することがあります。

# 6.1.4. LifeKeeper の設定作業

以下の設定作業は、LifeKeeper GUI を使用して実行できます。以下の 4 つの作業は、Apache リソースインスタンスに特有のものであり、Recovery Kit ごとに異なるため、本書で説明しています。

- リソース階層の作成：アプリケーションリソース階層を LifeKeeper クラスタに作成します。
- リソース階層の拡張：リソース階層をプライマリサーバからバックアップサーバに拡張します。
- リソース階層の拡張解除：リソース階層を LifeKeeper クラスタ内の 1 つのサーバから拡張解除 (削除) します。
- リソース階層の削除：リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。

以下の作業は、すべての Recovery Kit で共通の作業であり、手順が同一なので、SPS テクニカルドキュメンテーションの「GUIの管理」セクションで説明しています。

- リソース依存関係の作成: 既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除: リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service: リソース階層を特定のサーバで In Service にします。
- Out of Service: リソース階層を特定のサーバで Out of Service にします。
- プロパティの表示 / 編集: 特定のサーバでリソース階層のプロパティを表示または編集します。

注記：本セクションの残りの部分では、LifeKeeper GUI の [Edit] メニューから特定の作業を選択して Recovery Kit を設定する方法について説明します。設定作業はツールバーから選択することもできます。状況表示ウィンドウのリソース階層ツリー (左側のペイン) のグローバルリソースを右クリックして、[Edit] メニューと同じドロップダウンメニューの選択項目を表示することもできます。言うまでもなく、これは階層がすでに存在している場合にだけ可能な方法です。

サーバおよび特定リソースの状況に応じて、状況表示ウィンドウのリソース階層表 (右側のペイン) のリソースインスタンスを右クリックして、リソース階層の作成を除くすべての設定作業を実行することもできます。

# 6.1.4.1. Apache Web Server リソース階層の作成

**重要:**

Web Server リソース階層を作成する前に、Apache の設定ファイルに LifeKeeper が保護する既存の IP リソースが含まれていることを確認する必要があります。

Web ドキュメントまたは設定ファイルが共有ディスク上にある共有環境では、共有ファイルシステムがマウントされていることを確認する必要があります。また、重要事項として、リソースをバックアップサーバに拡張するには、稼働しているコミュニケーションパス (すなわちハートビート) が必須であることを念頭においてください。|

プライマリサーバからリソースインスタンスを作成するには、以下の手順を実行します。

1. LifeKeeper GUI メニューから [Edit] を選択し、次に [Server] を選択してください。メニューから、[Create Resource Hierarchy] を選択してください。

   リソースを作成するときは、Apache Web Server が停止している必要があります。ただし、システム設定ファイルにリスン変数を設定している場合は、デフォルトのデーモンの動作を許可できます。

   [Create Resource Wizard] ダイアログボックスが表示され、クラスタ内にインストールされて認識済みの Recovery Kit がすべて、ドロップダウンリストボックスに表示されます。

2. [Apache Web Server] を選択し、 **[Next]** をクリックしてください。

3. 以下の情報を入力するようにプロンプトが表示されます。ダイアログボックスで [Back] ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも [Cancel] をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| Switchback Type | [intelligent] または [automatic] を選択してください。この設定は、バックアップサーバへのフェイルオーバー後にサーバが復帰したときに、このサーバに Apache インスタンスをスイッチバックする方法を指定します。スイッチバックタイプは、必要な場合、[Resource Properties] ダイアログボックスの [General] タブで後から変更できます。 |
| Server | Apache Web Server を配置するサーバ (通常これをプライマリサーバまたはテンプレートサーバと呼ぶ) を選択してください。クラスタ内のすべてのサーバがドロップダウンリストボックスに表示されます。 |
| Web Server Binary Location | httpd Apache Web Server デーモンのフルパス名 (ファイル名を含む) を選択するか、入力してください。デフォルトは、/usr/sbin/httpd です。 |

| | |
|---|---|
| Web Server Root Directory | Web Server のルートディレクトリのフルパスを指定する必要があります。相対パスやシンボリックリンクを使用することはできません。Apache Web Server の設定ファイルは、サーバルートの conf/httpd.conf にあります。<br><br>注記: この時点で、保護される IP リソースが利用可能かどうかが LifeKeeper によってチェックされます。また、Apache Web Server リソース階層の作成に使用する入力データが妥当かどうかもチェックされます。この 2 つのチェックのどちらかで問題が検出された場合、[ERROR] ボックスが表示されます。Web Server のルートディレクトリのパスは妥当であるが、Apache の設定自体にエラーがある場合は、そのエラーを修正してから、階層の作成を続けることができます。ここで必要な LifeKeeper IP リソースを作成することもできます。 |
| Root Tag | Web Server 階層に付けるタグ名を選択するか、入力してください。デフォルトの apache<root directory>を選択するか、独自のタグ名を入力することができます。 |

4. [Create] をクリックしてください。[Create Resource Wizard] によって、Apache リソース階層が作成されます。LifeKeeper によって入力データが検査されます。問題が検出された場合、情報ボックスにエラーメッセージが表示されます。

5. 情報ボックスが表示され、Apache リソース階層が正常に作成されたこと、およびフェイルオーバによる保護を実現するにはその階層をクラスタ内の別のサーバに拡張する必要があることが示されます。[Next] をクリックしてください。

   注記: 新しい Apache インスタンスの起動に失敗したというエラーメッセージが表示されることがあります。新しい Apache 階層を別のシステムに拡張するには、起動する(In Service にする) 必要があることに注意してください。この時点で、表示されたエラーメッセージに基づいて問題を修正し、新しい階層を In Service にしてから、階層の拡張を続行できます。

6. [Continue] をクリックしてください。[Pre-ExtendWizard] が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、「Apache リソース階層の拡張」(後述) の手順 2 を参照してください。

   [Cancel] をクリックすると、別のダイアログボックスが表示され、Apache リソース階層を LifeKeeper の保護下に置くには、いずれかの時点でここに戻って Apache リソース階層を別のサーバに拡張する必要があることが警告されます。

# 6.1.4.2. リソース階層の拡張

この操作は [Edit] メニューから開始できます。または [Create Resource Hierarchy] オプションの動作が完了すると自動的に開始されます。その場合は、手順 2 を参照してください。

1. [Edit] メニューの [Resource] から [Extend Resource Hierarchy] を選択してください。[Pre-Extend Wizard] が表示されます。拡張操作に慣れていない場合は、[Next] をクリックしてください。LifeKeeper のリソース階層の拡張作業のデフォルト値が分かっていて、入力と確認を省略する場合は [Accept Defaults] をクリックしてください。

2. [Pre-Extend Wizard] に以下の情報を入力してください。

   注記: 最初の 2 つのフィールドは、[Edit] メニューの [Extend] から操作を開始した場合にだけ表示されます。

| フィールド | ヒント |
|---|---|
| Template Server | Apache リソースが現在 in service のサーバを入力してください。ここで選択するテンプレートサーバと次のダイアログボックスで選択する [Tag to Extend] によって、in service のリソース階層が表されることを認識しておくことが重要です。選択したテンプレートサーバで out of service のリソースタグを選択すると、エラーメッセージが表示されます。このダイアログのドロップダウンボックスに、クラスタ内の全サーバの名前が表示されます。 |
| Tag to Extend | テンプレートサーバからターゲットサーバに拡張する Web Server インスタンスの名前を選択してください。ウィザードのドロップダウンリストボックスに、テンプレートサーバ上に作成した全リソースのうち、前のダイアログボックスで選択したリソースの一覧が表示されます。 |
| Target Server | Web Server リソース階層を拡張するターゲットサーバを選択してください。ドロップダウンボックスには、選択した階層に含まれないクラスタ内のサーバ名の一覧が表示されます。 |
| Switchback Type | [intelligent] または [automatic] を選択してください。ここでは、バックアップサーバへのフェイルオーバの後、Web Server インスタンスが in service に戻ったときに、このサーバにどのようにスイッチバックされるのかを設定します。[intelligent] の場合、インスタンスをプライマリ / 元のサーバにスイッチバックするときに管理者の操作が必要になります。[automatic] の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。 |
| Template Priority | テンプレート階層の優先順位を選択するか、入力してください。優先順位は、1 ~ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を使用することを推奨します。注記: このフィールドは、階層をはじめて拡張するときにだけ表示されます。 |
| Target Priority | 拡張する Web Server 階層のターゲットの優先順位を選択するか、入力してください。この優先順位は1 ~ 999 の数値で表され、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。階層の優先順位は数字でソートされ、小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当)。LifeKeeper では、 |

| | |
|---|---|
| | 階層が作成されたサーバに「1」が自動的に割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |
| | 拡張前のチェックが正常に終了したというメッセージが表示された後、[Next] をクリックしてください。<br><br>拡張する階層に応じて、拡張されるリソースタグを示す一連の情報ボックス (編集不可) が表示されます。[Extend] をクリックしてください。 |
| Network Interface | ネットワークインターフェースを選択するか、入力してください。これは、IP リソースがターゲットサーバで使用するネットワークインターフェース (つまりイーサーネットカード) の名前です。 |
| Backup Interface | IP リソースを拡張するサーバで IP ローカルリカバリ機能を使用する場合は、バックアップインターフェースを選択してください。デフォルト値は [none] です。ただし、このサーバ上で他のネットワークインターフェースカードを設定している場合は、そのネットワークインターフェースカードがドロップダウンリストに表示されます。 |
| IP Resource Tag | IP リソースタグを選択するか、入力してください。これは、ターゲットサーバに拡張される IP リソースが使用するリソースタグ名です。 |
| Root Tag | ルートタグを選択するか、入力してください。これは、Web Server 階層に付けたタグ名です。デフォルトでは、テンプレートサーバとターゲットサーバの両方でルートタグ名は同一です。 |
| Mount Point | Web Server のルートディレクトリが共有ファイルシステム上に置かれている場合にのみ表示されます。<br><br>Web Server のルートディレクトリが置かれている共有ファイルシステムのマウントポイントを選択するか、入力してください。テンプレートサーバとターゲットサーバで、共有 Web Server のルートディレクトリのマウントポイントが同一である必要があります。多くの場合、ダイアログボックスに表示されたデフォルトのマウントポイントを選択する必要があります。 |
| Root Tag | Web Server のルートディレクトリが共有ファイルシステム上に置かれている場合にのみ表示されます。<br><br>ルートタグを選択するか、入力してください。これは、共有ファイルシステムのタグ名です。 |

3. 拡張が実行中であることを示す情報ボックスが表示されます。同じ Apache リソースインスタンスをクラスタ内の別のサーバに拡張する場合は、[Next Server] をクリックしてください。その場合は、リソース階層を拡張する操作を繰り返します。

   [Finish] をクリックした場合、Web Server リソースの拡張処理が正常に完了したかどうかが検証されます。

4. 最後のダイアログボックスで [Done] をクリックして、[Extend Resources Hierarchy] メニューの選

択を終了してください。

**注記**: 必ず両方のサーバで新しいインスタンスの機能をテストしてください。

# 6.1.4.3. リソース階層の拡張解除

1. [Edit] メニューの [Resource] から [Unextend Resource Hierarchy] を選択してください。

2. 拡張解除する Web Server リソースが存在するターゲットサーバを選択してください。Web Server が現在 in service になっているサーバは選択できません (このダイアログボックスは、右側のペインでリソースインスタンスを右クリックして [Unextend] 作業を選択した場合には表示されません)。

   [Next] をクリックしてください。

3. 拡張解除する Web Server 階層を選択し、[Next] をクリックしてください (このダイアログボックスは、いずれかのペインでリソースインスタンスを右クリックして [Unextend] 作業を選択した場合には表示されません)。

4. 拡張解除の対象として選択したターゲットサーバと Web Server リソース階層を確認する情報ボックスが表示されます。[Unextend] をクリックしてください。

5. Web Server リソースが正常に拡張解除されたことを示す別の情報ボックスが表示されます。

6. [Done] をクリックして終了してください。

# 6.1.4.4. リソース階層の削除

階層を out of service にする前に階層を削除した場合、リソース階層は LifeKeeper の保護から除外されますが、Apache インスタンスは現在アクティブなサーバで動作を続行します (ただし、手動で停止した場合やシステムを再起動した場合を除く)。別の IP アドレスを持つ同じ Apache 階層を再作成したり、以前使用した IP アドレスを使用して新しい Apache 階層を作成 (ただし別のサーバルートを使用) しようとすると、同じアドレスで動作中の Apache インスタンスとの競合が発生します。

LifeKeeper 環境内のすべてのサーバからリソース階層を削除するには、以下の手順を実行してください。

1. LifeKeeper GUI メニューから [Edit] を選択し、次に [Resouce] を選択してください。ドロップダウンメニューから、[Delete Resource Hierarchy] を選択してください。

2. 削除する Web Server リソース階層が存在するターゲットサーバの名前を選択してください。[Next] をクリックして次のダイアログボックスに進んでください。

   注記: 右側のペインで個々のリソースインスタンスを右クリックするか、またはリソースが 1 台のサーバ上のみにあるときに左側のペインでグローバルリソースを右クリックして [Delete Resource] 作業を選択した場合、ターゲットサーバのダイアログボックスは表示されません。

3. [Hierarchy to Delete] を選択してください (このダイアログボックスは、左側または右側のペインでリソースインスタンスを右クリックして、[Delete Resource] 作業を選択した場合には、表示されません。)このリストボックスには、ターゲットサーバ上にある、in service と out of service の両方の階層がすべて表示されます。Apache インスタンスを停止してリソース階層を LifeKeeper の保護から削除する場合は、削除前に、選択した階層を必ず out of service にする必要があります。

   [Next] をクリックしてください。

4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。[Next] をしてください。

5. Web Server リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。

6. [Done] をクリックして終了してください。

# 6.1.4.5. リソース階層のテスト

手動スイッチオーバを開始することによって、Apache リソース階層をテストできます。このテストでは、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

## GUI からの手動スイッチオーバの実行

スイッチオーバを手動で開始するには、LifeKeeper GUI から [Edit] の [Resource] を選択し、ドロップダウンメニューから [In Service] を選択します。例えば、バックアップサーバで in service 要求を実行すると、アプリケーション階層がバックアップサーバで in service になり、プライマリサーバでは out of service になります。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバに変わります。

Out of Service 要求を実行すると、アプリケーションは他のサーバで in service にならずに、out of service になります。

## リカバリ動作

プライマリサーバに障害が発生すると、Apache Recovery Kit ソフトウェアは以下の作業を実行します。

- プライマリサーバの 1 つ以上の物理ネットワークインターフェースで IP アドレスを in service にすることにより、Apache をバックアップサーバで in service にします。
- ファイルシステムを使用する場合は、そのサーバの共有ディスク上にファイルシステムをマウントします。
- Apache に関連するデーモンプロセスを開始します。

リカバリ後に、ブラウザの [Reload] / [Refresh] ボタンを使用して、再接続できます。

# 6.1.5. トラブルシューティング

本セクションでは、LifeKeeper Apache Web Server リソース階層の作成 / 拡張中、およびリソースの削除 / リストア中に表示される可能性のあるメッセージの一覧を示します。また、必要に応じて、エラーの原因や、エラー状態を解決するために必要な処置について補足説明します。他の SPS コンポーネントからメッセージが送出されることもあります。その場合は、メッセージカタログ(弊社のテクニカルドキュメンテーションサイトの「エラーコードの検索」内) を参照してください。メッセージカタログには、操作、管理、GUI など、SIOS Protection Suite for Linux を使用しているときに表示される可能性がある、すべてのエラーコードの一覧を示します。また、エラーコードの原因に関する補足説明や、問題解決のために必要な処置についても、必要に応じて記載します。この一覧から、受信したエラーコードを検索できます。また、該当する SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

本セクションのメッセージは、以下のトピックに分類されています。

- 階層の作成
- 階層の拡張
- 階層の削除、リストア、リカバリ

# 6.1.5.1. 階層作成時のエラー

## 階層作成時のエラー

Apache 階層の作成時に表示されるエラーメッセージを、説明とともに以下に示します。LifeKeeper Core および他の Recovery Kit が表示するエラーメッセージはこのガイドには記載していません。ここに記載されている問題を解決するために、階層の作成を中断できます。そして中断した時点から、Apache の設定に必要な新しい LifeKeeper リソースの作成も含めて、階層の作成を続行できます。

### Web Server バイナリの位置検証時

"Error: valid_httpd_path: Must specify absolute path to httpd executable."

有効な Apache httpd 実行可能ファイルの絶対パスを省略せずに入力してください。

"Error: valid_httpd_path: File does not exist at path specified."

指定した場所に、有効な Apache httpd 実行可能ファイルが存在しません。

"Error: valid_httpd_path: Httpd failed to display Server version."

指定した場所にある httpd 実行可能ファイルが、Apache 標準の「サーバのバージョン」を表示しません。

"Error: valid_httpd_path: Incorrect version $MAJOR.$MINOR.$POINT of Apache at $HTTPD_PATH.

指定した場所にある Apache httpd 実行可能ファイルが表示する「サーバのバージョン」が正しくありません。

### Web Server のルートディレクトリの検証時

"Error: valid_http_root: Cannot find Apache configuration file at $CONF_FILE."

指定したサーバルートディレクトリの conf/httpd.conf に、有効な Apache 設定ファイルが必

要です。デフォルト設定を使用して Apache を SuSE にインストールする場合、httpd.conf 設定ファイルは ServerRoot のサブディレクトリ conf に配置されません。この場合は、設定ファイルを /etc/httpd/conf ディレクトリに配置し直す必要があります。

"Error: valid_http_root: Must specify absolute path to Apache server root directory."

サーバルートディレクトリの絶対パスを省略せずに入力してください。

"Error: valid_http_root: Apache instance at $HTTP_ROOT is already under LifeKeeper protection."

各インスタンスには、一意のサーバルートディレクトリが必要であり、設定ファイルが conf/httpd.conf に必要です。指定されたサーバルートディレクトリは、すでに他の Apache インスタンスが使用しています。

"Syntax error on line <line number> of <configuration file path>, etc…"

Apache 設定ファイルに構文エラーがあります。これらのエラーメッセージは、httpd -T コマンドを使用して $CONF_FILE の構文をチェックするときに表示されます。詳細については、表示されたエラーメッセージを確認してください。

"Error: valid_http_root: Since $HTTPD_PATH is shareable on $HTTPD_PATH_SHARED, $HTTP_ROOT must be also."

httpd 実行可能ファイルが共有 / 共有可能ストレージに存在する場合は、サーバルートとすべての DocumentRoot ディレクトリも共有 / 共有可能ストレージに存在する必要があります。

"Error: valid_http_root: Since $HTTP_ROOT is shareable on $HTTP_ROOT_SHARED, all document root directories must shareable on this same filesystem."

サーバのルートディレクトリが共有 / 共有可能ストレージに存在する場合は、すべての DocumentRoot ディレクトリも共有 / 共有可能ストレージに存在する必要があります。

"Error: http_docs_shared: Since one/more Apache document root directories

are shareable on $docs_shared, $curr_root must be also.

いずれかの DocumentRoot ディレクトリが共有 / 共有可能ファイルシステムに存在する場合、すべての DocumentRoot ディレクトリが同じファイルシステムに存在する必要があります。

"Error: valid_http_root: Must include BindAddress or Listen directives for each Apache instance. Check the Apache configuration file at $CONF_FILE."

複数の Apache インスタンスを実行するには、各設定ファイルに BindAddress または Listen ディレクティブが必要です。詳細については、前のセクションにあるApache Web Server の設定に関する考慮事項 を参照してください。

"Error: valid_http_root: Default IP address * not allowed for LifeKeeper protection. Check the Apache configuration file at $CONF_FILE."

個々の Apache インスタンスについて、LifeKeeper が保護する IP アドレスを 1 つ以上指定する必要があります。

"Error: valid_http_root: A Listen directive is being used which specifies an IP address but no port. Check the Apache configuration file at $CONF_FILE."

Listen ディレクティブの正しい構文は、「Listen [IPaddress:] port number」です。これは、Apache では構文エラーとは見なされませんが、間違って解釈されます (IP アドレスの最初の数字がポート番号の指定と見なされる)。

"Error: valid_http_root: IP address $ip is not LifeKeeper protected."

Apache の設定ファイルが、LifeKeeper の保護下に設定されていない IP アドレスまたはドメイン名を参照しています。これらの LifeKeeper IP アドレスのリソースをあらかじめ作成しておく必要があります。

## Apache リソース階層の作成時

"Error: Could not find IP resource for $IP_ADD on machine $MACH."

このリソースを作成してから、Apache リソースの作成を続行する必要があります。

"Error: Create Apache file system hierarchy failure for filesystem $FSNAME used by server root $HTTP_ROOT."

"Error: Failure bringing Apache Resource $TAG into service on machine $MACH."

メッセージについては、Apache のエラーログを確認してください (デフォルトの位置は /var/log/httpd/error_log ですが、他のログがリストされることがあります)。

この問題の原因の多くは、Apache の設定ファイルのエラーです。問題を修正した後、このリソースを手動で in service にすることができます。

"LifeKeeper: RESTORE: *ERROR* Apache: The instance is Password Protected."

LifeKeeper Apache Web Server Recovery Kit は、SSL 対応 Web サーバに使用される、パスワードで保護された秘密キーファイルをサポートすることができません。これは、Apache を起動するたびに手動操作が必要になり、再起動とフェイルオーバの自動実行を妨げるからです。本書の「Apache Web Server の設定に関する考慮事項」セクションに、秘密キーファイルのパスワード保護を解除する方法 (SSLCertificateKeyFile ディレクティブで指定) が記載されています。このメッセージは、Apache とともに SSL モジュールを使用する環境にのみ適用されます。

# 6.1.5.2. 階層拡張時のエラー

Apache 階層の拡張時に表示されるエラーメッセージを、説明とともに以下に示します。これらのエラーメッセージは、GUI が「Executing the pre-extend script....」というメッセージを表示して、新しいシステムに階層を拡張する前の階層の検証時に表示されることに注意してください。

各メッセージの前には、

「Error - canextend(template_server, tag, app_type/resource_type, target_server) -」のようなエラーメッセージが表示されます。

各メッセージの後には、

「Error - extmgr(template_server, tag, target_tag, target_server) -」のようなエラーメッセージが表示されます。

## Web Server バイナリの位置検証時

階層作成時のエラーの下にある、Web Server バイナリの位置検証時のエラーリストを参照してください。

## ターゲットシステムの Apache 設定ファイルの検証時

"Cannot find Apache configuration file at $CONF_FILE on $TARGET_SYS."

指定したサーバのルートディレクトリの conf/httpd.conf に、有効な Apache の設定ファイルが必要です。

"DocumentRoot directory "$doc" in $CONF_FILE on $TARGET_SYS was not found in the configuration file on $TEMPLATE_SYS."

または

"DocumentRoot directory "$doc" in $CONF_FILE on $TEMPLATE_SYS was not found in the configuration file on $TARGET_SYS."

ターゲットサーバとテンプレートサーバの設定ファイルを比較するときに、これらのいずれとも一致しない DocumentRoot ディレクトリが 1 つ以上見つかりました。表示されたエラーメッセージの詳細を確認して、2 つの設定ファイルの違いを調べてください。DocumentRoot ディレクトリのパスを正しく入力しなかった場合、それぞれの設定ファイルには他方の設定ファイルに存在しないエントリがあるように見えるので、通常はこれらのエラーメッセージの両方が表示されます。

"IP:port combination "$ipp" in $CONF_FILE on $TARGET_SYS was not found in the configuration file on $TEMPLATE_SYS."

または

"IP:port combination "$ipp" in $CONF_FILE on $TEMPLATE_SYS was not found in the configuration file on $TARGET_SYS."

ターゲットサーバとテンプレートサーバの設定ファイルを比較するときに、一方のサーバで使用されるように設定されているが、他方のサーバでは設定されていない IP/ ポートの組み合わせが 1 つ以上見つかりました。使用される IP/ ポートの組み合わせは、Apache のさまざまな設定ディレクティブを使用して、IP アドレス、ポート、およびドメイン名で指定されます。比較されるのは、実際に使用される IP/ ポートの組み合わせであり、IP/ ポートの指定に使用されるディレクティブではありません。表示されたエラーメッセージの詳細を確認して、2 つの設定ファイルの違いを調べてください。

"SSLCertificateKeyFile "$file" in $CONF_FILE on $SYS1 was not found in the configuration file on $SYS2."

ターゲットシステムの Apache の設定ファイル内で SSLCertificateKeyFile として指定されたファイル名が、テンプレートシステムで指定されているファイル名と一致しません。これらの設定は同一である必要があります。このメッセージは、Apache とともに SSL モジュールを使用する環境にのみ適用されます。

"Apache SSLCertificateKeyFile exists on $SYS1 but not on $SYS2."

Apache の設定ファイル内に指定された SSLCertificateKeyFile が、一方のシステムには存在しますが、他方のシステムには存在しません。このファイルは、両方のノードに存在する必要があります。このメッセージは、Apache とともに SSL モジュールを使用する環境にのみ適用されます。

"WARNING: PHP configuration file $PHP_CONFIG appears to be different on $SYS1 and $SYS2."

ターゲットシステムの PHP モジュールの設定ファイルが、テンプレートシステムの PHP モジュールの設定ファイルと異なります。両方のサーバの設定を確認して、同一にする必要があります。このメッセージは、Apache とともに PHP モジュールを使用する環境にのみ適用されます。

## ターゲットサーバでの Apache リソース階層の作成時

階層作成時のエラーの下にある、Apache リソース階層作成のエラーリストを参照してください。

# 6.1.5.3. 階層リソースのリストア、削除、リカバリ時のメッセージとエラー

以下のトピックに含まれるエラーメッセージは、LifeKeeper のエラーログに出力されます。

このエラーログは、「lk_log log」と入力すると表示できます。

## Apache リソースを In Service にするとき (リストア)

"LifeKeeper: RESTORE: APACHE: RESTORING $TAG TO SERVICE START AT: <date>"

情報メッセージです。リストアの開始時刻を記録します。各リストアの開始時に記録されます。

"LifeKeeper: RESTORE APACHE RESOURCE $TAG END err=$err AT: <date>"

情報メッセージです。リストアの完了時刻を記録します。各リストアの完了時に記録されます。リストア時にエラーが発生した場合は、追加のメッセージがこれら 2 つのメッセージの間に記録され、err=$err に表示される値がゼロ以外の値になります。

"Apache: No instance information found for Tag=$TAG."

エラー: restore スクリプトに渡されたタグ値を持つインスタンスが定義されていないことを示します。LifeKeeper が既知のタグのみが In Service と Out of Service の動作の選択肢として使用できるため、GUI で発生する可能性は低いです。

"LifeKeeper: RESTORE: Apache: Tag=$TAG already running."

情報メッセージです。表示されたインスタンスがすでに起動して動作中であることを示します。

"LifeKeeper: RESTORE: Apache: Existing processes terminated for ID=$ID."

情報メッセージです。このインスタンス ID について、既存の httpd プロセスが実行中であることが検出されましたが、PidFile が見つからないか、無効です。このため、実行中のプロセスが強制終了されました。

"LifeKeeper: RESTORE: Apache: Invalid PidFile=$PIDFILE has been deleted."

情報メッセージです。このインスタンスについて既存の PidFile が見つかりましたが、その内容が無効です。このため、PidFile が削除されました。

"LifeKeeper: RESTORE: Apache: Tag=$TAG is being restarted."

情報メッセージです。表示されたインスタンスが部分的に動作中であるが、再起動が必要であることを示します。PidFile (親 httpd プロセスのプロセス ID を持つ) がまだ存在している場合、インスタンスは HUP 信号を使用して再起動されます。PidFile が見つからない場合、インスタンスは完全に停止されてから再起動されます。

"LifeKeeper: RESTORE: *ERROR* Apache: Error in web server configuration file $CONF_FILE for

instance $ID."

"LifeKeeper: RESTORE: *ERROR* Apache: Execute the following command to check the syntax of this file:"

"LifeKeeper: RESTORE: *ERROR* Apache: $HTTPD_PATH -t -d $SERVER_ROOT -f $CONF_FILE."

インスタンスの起動前に、httpd -t オプションを使用して、設定ファイルの構文がチェックされます。-d オプションは、ServerRoot ディレクトリをチェックします。モジュールを使用するように Apache を設定した場合、モジュールに関するその他のオプションも表示されることがあります。階層作成時に検出された構文エラーは LifeKeeper GUI に表示されますが、その後に持ち込まれた構文エラーは、GUI や LifeKeeper のログには表示されません。以下のコマンドを手動で実行して、設定の不具合を特定する必要があります (該当する場合は、モジュールのオプションを追加する)。

$HTTPD_PATH -t -d $SERVER_ROOT -f $CONF_FILE

"LifeKeeper: RESTORE: *ERROR* Apache: Error starting web server instance $INSTANCE."

"LifeKeeper: RESTORE: *ERROR* Apache: Restore of tag $TAG failed."

"LifeKeeper: RESTORE: *ERROR* Apache: Examine the Apache error log at $ERROR_LOG"

"LifeKeeper: RESTORE: *ERROR* Apache: to determine the cause of the problem."

パラメータを指定した httpd デーモンの実行時にエラーが発生しました。使用している httpd 実行可能ファイル、設定ファイル、および一般設定に問題がないかどうかをチェックしてください。

"LifeKeeper: RESTORE: *ERROR* Apache: Web server instance $ID did not start correctly."

"LifeKeeper: RESTORE: *ERROR* Apache: Restore of tag $TAG failed."

"LifeKeeper: RESTORE: *ERROR* Apache: Examine the Apache error log at $ERROR_LOG"

"LifeKeeper: RESTORE: *ERROR* Apache: to determine the cause of the problem."

多くの場合、Web サイトが予測どおりに応答しなくとも、httpd デーモンが起動したように見えることがあることに注意してください。restore スクリプトは、使用する IP/ ポートの組み合わせをすべてチェックし、設定されたすべてのサイトがフルに機能することを確認します。設定されたすべてのサイトがフルに機能しない場合、このメッセージが出力され、リストアは失敗します。

サイトが Out of Service 状態になっていても、1 つ以上の httpd プロセスが動作し続けている場合があることに注意してください (1 つ以上の Web サイトが動作している可能性があり、それらの Web サイトを強制終了したくないため、意図的にプロセスの動作を継続させています)。できるだけ早く問題を解決し、インスタンスを In Service にする必要があります。問題が解決されない場合、LifeKeeper が最終的にインスタンスのリカバリおよびリストアを自動的に試行します。LifeKeeper によるリカバリとリストアができない場合は、階層が別のサーバにフェイルオーバします。

## Apache リソースを Out of Service にするとき (削除)

"LifeKeeper: REMOVE: APACHE: REMOVE $TAG FROM SERVICE START AT: <date>"

情報メッセージです。削除の開始時刻を記録します。各削除の開始時に記録されます。

"LifeKeeper: REMOVE APACHE RESOURCE $TAG END err=$err AT: <date>"

情報メッセージです。削除の完了時刻を記録します。各削除の完了時に記録されます。

削除時にエラーが発生した場合は、追加のメッセージがこれら 2 つのメッセージの間に記録され、err=$err に表示される値がゼロ以外の値になります。

"LifeKeeper: REMOVE: *WARNING* APACHE: Error attempting to kill parent process for INSTANCE=$INSTANCE."

親 httpd プロセスを強制終了しようとするエラーがありました (親プロセスのプロセス ID は Pidfileに格納される)。

"LifeKeeper: REMOVE: *ERROR* APACHE: Error attempting to kill all processes for INSTANCE=$INSTANCE."

親 httpd プロセスが正常に強制終了されたように見えますが、このインスタンスの 1 つ以上のプロセスがまだ動作中です。通常、削除動作では、このインスタンスのすべてのプロセスを強制終了できます。強制終了できない場合、このメッセージが出力され、削除は失敗します。

## Apache リソースを In Service に戻すとき (リカバリ)

LifeKeeper Core は、Apache の quickCheck スクリプトを実行して、ローカルサーバで In Service 状態にあるすべての Apache インスタンスの健全性を定期的にチェックします。Apache の quickCheck スクリプトは、リストア時と削除時にインスタンスの状態をチェックするものと同じスクリプトを使用して、Web サイトをチェックします。インスタンスが完全に動作していない場合、recover スクリプトが実行され、インスタンスの再起動が試行されます。これは、以下に示す最初のメッセージを記録し、restore を起動します。restore スクリプトの結果に応じて、最後のエラーメッセージまたは成功メッセージを出力し、restore と同じ結果を返します。restore/recover が失敗した場合、このインスタンスは別のサーバにフェイルオーバされます。

"LifeKeeper: RECOVER: APACHE: Invoking restore for Apache instance "$ID" at: <date>"

"LifeKeeper: RECOVER: APACHE: Restore for Apache instance $ID returned error $RET at: <date>"

"LifeKeeper: RECOVER: APACHE: Restore for Apache instance $ID successful at: <date>"

# 6.2. DB2 Recovery Kit 管理ガイド

## DB2 Recovery Kit テクニカルドキュメンテーション

LifeKeeper for Linux DB2 Recovery Kit は、DB2 データベース・インスタンスの障害回復機能を提供します。LifeKeeper は、DB2 Universal Database 製品ファミリとともに使用して、長時間のシステム停止や人手の介入なしにデータベース・サーバの障害を効果的に回復することによって、DB2 運用環境の可用性を高めます。

## ドキュメントの内容

本書は以下のトピックについて説明しています。

- ドキュメンテーションとリファレンス 。LifeKeeper for Linux のドキュメントと所在の一覧です。
- 要件 。DB2 Recovery Kit を適切に設定、導入、運用するうえで必要なハードウェアとソフトウェアについて説明しています。LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SIOS Protection Suite インストールガイド を参照してください。
- 概要 。DB2 Recovery Kit の特徴と機能について説明しています。
- LifeKeeper for Linux DB2 Recovery Kit の設定 。DB2 Recovery Kit を適切に設定するうえで必要な手順について説明しています。
- LifeKeeper 設定作業 。LifeKeeper GUI を使用して DB2 リソース階層を作成し管理する作業について説明しています。
- トラブルシューティング 。LifeKeeper for Linux のエラー・メッセージの一覧とその説明です。
- 付録 。DB2 で RAW I/O を使用する設定手順です。

# 6.2.1. ドキュメンテーションとリファレンス

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- SIOS Protection Suite インストールガイド
- Optional Recovery Kit ドキュメンテーション

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

# 6.2.2. ハードウェア要件とソフトウェア要件

SPS for Linux DB2 Recovery Kit をインストールする前に、SPS の構成が次の要件を満たしている必要があります。SPS for Linux ハードウェアおよびソフトウェアの設定に関する具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

## ハードウェア要件

- サーバ 。製品媒体に付属する SPS for Linux リリースノート および SPS for Linux インストールガイド (製品メディアに収録) に記載の要件に従って設定されたLifeKeeper for Linux 対応サーバ。

- **IP** ネットワーク・インターフェース・カード 。各サーバは、イーサネット TCP/IP をサポートするネットワーク・インタフェースを 1 つ以上必要とします。ただし、LifeKeeper クラスタには 2 つの通信パスが必要で、独立した 2 つのサブネットを使用する 2 つの分離した LAN ベース通信パスが推奨され、これらの 1 つ以上をプライベート・ネットワーク として構成する必要があることを忘れないでください。TCP と TTY のハートビートの組み合わせもサポートされています。

## ソフトウェア要件

- **TCP/IP** ソフトウェア 。LifeKeeper 環境の各サーバには TCP/IP ソフトウェアが必須です。

- **IBM** ソフトウェア 。特定の Linux ディストリビューションとハードウェア・アーキテクチャにおける具体的な DB2 バージョンの要件については、SPS for Linux リリースノート を参照してください。

- LifeKeeper ソフトウェアクラスタの各サーバにインストールした LifeKeeper ソフトウェアのバージョン、および、適用した LifeKeeper ソフトウェア・パッチのバージョンが同一であることが絶対条件です。

- **LifeKeeper for Linux DB2 Recovery Kit** - DB2 Recovery Kit は CD-ROM 媒体で提供されます。Red Hat Package Manager（rpm）でパッケージされており、rpm を使用してインストール / アンインストールを行います。LifeKeeper for Linux DB2 Recovery Kit CD には、次のrpmファイルが格納されています。

  **steeleye-lkDB2**

  LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SPS for Linux インストールガイド を参照してください。

- **LifeKeeper for Linux NFS Recovery Kit** - DB2 EEE デプロイメントおよびマルチ・パーティション ESE デプロイメントを使用する場合に必要です。この recovery kit は CD の steeleye-lkNFS パッケージに格納されています。

  **重要:**「DB2 EEE またはマルチ・パーティション ESE と NFS に関する問題 」を参照してください。重要な設定情報が記載されています

# 6.2.3. 概要

## LifeKeeper for Linux DB2 Recovery Kit

Version 8 以降では、DB2 UDB Enterprise Edition（EE）と Enterprise-Extended Edition（EEE）を組み合わせ、DB2 UDB Enterprise Server Edition（ESE）という 1 つの製品を構成しています。旧バージョンは、標準的なリレーショナル・データベース管理システム としての Enterprise Edition（EE）と、マルチ・パーティション・データベースに対応する EE データベース・サーバの拡張版としての Enterprise-Extended Edition（EEE）の、2 種類の別個のエンタープライズ・レベル・データベース・サーバで構成されています。

LifeKeeper for Linux DB2 Recovery Kit は、EE、WE、WSE 環境のデータベース・マネージャ、および EEE 環境のデータベース・パーティション・サーバを保護します。組み合 わせの ESE 環境では、Recovery Kit はデータベース・マネージャとデータベース・パーティション・サーバの両方を保護します。

ユーザは、LifeKeeper クラスタ内の各マシンに DB2 管理サーバを定義することもできます。DB2 管理サーバを定義すると、LifeKeeper は、DB2 階層作成操作と DB2 階層回復操作の機能として、DB2 管理サーバの起動を試みます。

# 6.2.4. LifeKeeper for Linux DB2 Recovery Kit の設定

本セクションでは、LifeKeeper for Linux DB2 Recovery Kit の設定の詳細について説明します。DB2 Recovery Kit の設定と管理を開始する前に考慮すべき事項についても記載しています。LifeKeeper Core リソース階層の設定方法については、SPS for Linux テクニカルドキュメンテーション を参照してください。

# 6.2.4.1. DB2 で RAW I/O を使用する場合

DB2 で RAW I/O デバイスを使用する場合は、LifeKeeper Core CD から LifeKeeper Raw I/O Recovery Kit をインストールする必要があります。また、使用前に RAW I/O デバイスを 適切に設定する必要もあります。手順については、「付録 」を参照してください。

# 6.2.4.2. DB2 の使用

## DB2 プロセスの起動回数が多くなりすぎる場合の回避方法

場合によっては、LifeKeeper の保護下で DB2 8.x を使用する場合に、DB2 プロセスの起動回数が多くなりすぎる場合があります。カーネルのネットワーク・パラメータに次の変更を加えることで、この状況を緩和できます。上記のように、DB2 8.x を使用する予定の各 LifeKeeper クラスタ・システムで、*/etc/sysctl.conf* ファイルに次の行を追加します。

```
net.ipv4.tcp_syn_retries=1
```

次に **sysctl -p** を実行すると、この変更が有効になります。

## DB2 インスタンスが頻繁にクラッシュ (Panic) する場合の回避方法

systemdを使用している環境で DB2 を使用する場合、インスタンスがクラッシュすることがあります。systemd のパラメータに次の変更を加えることで、この状況を回避できます。各 LifeKeeper クラスタ・システムで、/etc/systemd/logind.conf ファイルに以下の設定を行ってください。

```
RemoveIPC=no
```

次に **systemctl restart systemd-logind** を実行すると、この変更が有効になります。 詳細はこちら をご覧ください。

# 6.2.4.3. DB2 シングル・パーティションの設定の検討事項

LifeKeeper for Linux DB2 Recovery Kit をシングル・パーティション環境またはワークグループ環境で運用する前に、以下の事項を考慮する必要があります。

1. LifeKeeper では、DB2 インスタンスのホーム・ディレクトリの場所に加えて、共有ド ライブに格納された関連するデータベース、テーブルスペース、リソースの場所も必要です。共有ドライブは、階層の作成時に自動的に保護されます。DB2 リソース 階層 の作成時に、DB2 データベース・マネージャが親リソースとして作成され、インスタ ンスのホーム・ディレクトリおよび実際のデータベースを含む共有ファイル・システムが従属リソースとして作成されます。したがって、DB2 階層作成後に、LifeKeeper で保護されていない共有ファイル・システム上にデータベースを作成する場合は、そのファイル・システム用のリソース階層を作成して、DB2 リソース階層の従属リソースにする必要があります。
2. プライマリ・システムのデータベース・マネージャが動作不能になると、サービスは定義済みのバックアップ・システムにフェイルオーバされます。バックアップ・システムのデータベース・サービスは、従属リソースがフェイルオーバされ、データベース・マネージャがサービスを開始すると直ちに利用可能になります。以前に接続されていた DB2 クライアントは接続を解除されるため、バックアップ・サーバに再接続する必要があります。コミットされていない SQL 文はすべてロールバックされるため、再入力する必要があります。

# 6.2.4.4. DB2 マルチ・パーティションの設定の検討事項

**DB2 マルチ・パーティションの制約：** あるマシン上に LifeKeeper DB2 リソース階層を作成すると、そのマシン上の DB2 マルチ・データベース・パーティション・サーバはすべて保護されます。保護するノードは、次のファイルを調べて判断されます。

**<インスタンス・ホーム>/sqllib/db2nodes.cfg**

この Recovery Kit では、将来、N-way フェイルオーバ機能が追加される予定です。

# 6.2.4.4.1. DB2 EEE またはマルチ・パーティション ESE と NFS に関する問題

DB2 インスタンスの実行中に DB2 インスタンスのホーム・ディレクトリの NFS エクス ポート・ポイントが使用できなくなると、システムはエクスポート・ポイントが再び使用可能になるまで待機し、応答しない状態になります。多くのシステム操作は、システムの再起動を含めて、正しく動作しません。DB2 マルチ・パーティション・クラスタ用 の NFS サーバも LifeKeeper で保護する必要があり、NFS リソースをシャットダウンする前に DB2 クラスタの全パーティションも手動でサービス休止にする場合以外は、NFS サーバを手動でサービス休止にしてはいけないことに注意してください。また、DB2 パー ティションは、NFS リソースがサービス中でないとサービスを開始できません。

不注意で NFS サーバを停止してクラスタをハングさせないよう、次のようにすることを推奨します。

**NFS に関する推奨事項**

**追加サーバの使用：** DB2 インスタンスのホーム・ディレクトリをマウントする NFS エク スポート・ポイント用に別のクラスタを用意することを強く推奨します。このクラスタの NFS エクスポート・ポイントは、LifeKeeper NFS Server Recovery Kit で保護する必要があります。

少なくとも 2 台の追加サーバを用意できない場合は、DB2 クラスタに追加サーバを 1 台追加して、前述の問題が発生する可能性を減らすことができます。この追加サー バは NFS 階層をエクスポートします。クラスタ内の別のノードの 1 つがバックアッ プの役目を果たします。この構成では、NFS 階層がバックアップ・ノードにフェイ ルオーバされた場合に、前述の問題が発生します。このクラスタの NFS エクスポート・ポイントは、LifeKeeper NFS Server Recovery Kit で保護する必要があります。

**追加サーバを使用できない場合：** これは、望ましくない選択肢です。ただし、DB2 マルチ・パーティションと同じクラスタで NFS サーバを実行する場合、NFS エクス ポート・ポイントは LifeKeeper NFS Server Recovery Kit で保護する必要があります。 現在、 LifeKeeper では、DB2 パーティションとそれを管理する NFS サーバとの関係が認識されないことに注意してください。したがって、クラスタ内のどのノードでも、LifeKeeper の停止または開始を行う前に、手動で次の手順を踏む必要があります。

1. 1 台のサーバで LifeKeeper を停止する場合は、クラスタ内の別のサーバで NFS サーバが実行中であることを確認する必要があります。そうでない場合、LifeKeeper の シャットダウン処理で、DB2 パーティションのサービスを停止しようとしてハングする可能性があります。一般に、LifeKeeper の再起動時に問題が発生しないようにするには、LifeKeeper を停止する前に、すべての DB2 パーティションを別のサーバに切り替えるか、手動でサービスを停止する必要があります。

2. クラスタ全体をシャットダウンするには、手動ですべての DB2 パーティション・ リソースのサービスを休止する必要があります。次に、すべての DB2 NFS サーバ・ リソースのサービスを休止して、最後に LifeKeeper をシャットダウンします。

3. LifeKeeper をシャットダウンする前に、忘れずに DB2 リソースのサービスを休止していれば、LifeKeeper を正常に再起動できるはずです。次に NFS サーバ・リソースのサービスを開始して、引き続き、再起動するすべての DB2 パーティションのサービスを開始します。

4. LifeKeeper をシャットダウンする前に DB2 パーティションのサービスを休止することを忘れた場合

は、LifeKeeper を再起動する前に、そのパーティションの NFS サーバ・リソースがクラスタ内のどこかで実行中であることを確認する必要があります。

# 6.2.4.4.2. 設定条件

マルチ・パーティション環境で DB2 Recovery Kit が正しく動作するには、以下の要件を 満たしている必要があります。

1. NFS 階層のために追加のクラスタを使用できない場合、LifeKeeper for Linux DB2 Recovery Kit によって、NFS が保護しているファイル・システム上のアクティブ inode の数が制限されることを忘れないでください。したがって、この状態を避けるため、トップレベルのディレクトリーを保護し、完全修飾ディレクトリー名を使用して インスタンスのホーム・ディレクトリーをエクスポートすることをお勧めします。トップレベル・ディレクトリーは、ユーザが直接そこにディレクトリーを変更（すなわち *cd <トップレベル・ディレクトリー>* の実行）できないように、保護されます。

2. 本書の「ソフトウェア要件」セクションで説明したように、IBM の最新 Fix Pack （EEE デプロイメントの場合）をインストールしていることを確認します。

3. *db2nodes.cfg* ファイルのホスト名の値を、 **hostname** コマンドが返す値と同じにします。

   例：

   db2nodes.cfg ファイル：

   **0 server1.sc.steeleye.com 0**

   さらに、サーバーの/etc/hosts ファイルのホスト名の値も db2nodes.cfg ファイルのホスト名の値と同じにする必要があります。

   また、サーバーの/etc/hosts ファイルに、そのファイルに含まれる各サーバー・エントリ のローカル・ホスト名と完全修飾ホスト名が両方記述されていることを確認する必要もあります。

   例：

   */etc/hosts* ファイル

   ```
   127.0.0.1 localhost localhost.localdomain

   9.21.55.53 server1.sc.steeleye.com server1
   ```

4. db2setup スクリプトを実行するときに、この時点で、DB2 Warehouse Control Database（DWCNTRL）または DB2 Sample Database の作成を 選択しないでください。 DB2 リソース階層を正しく作成するには、これらのデータベースを共有ファイル・システム上に作成する必要があります。db2setup スクリプト実行中に、どちらかのデータベースの作成を選択すると、データベースが共有ファイル・システムではなくホーム・ディレクトリーに作成されます。これらのデータベースを作成する場合は、 db2setup スクリプトでは作成せず、別の方法で共有ファイル・システムを指定して作成する必要があります。

   Version 8.1 以降では、セットアップ・スクリプトの実行中に DB2 ツール・カタログが作成されないようにします。必要に応じて、このデータベースは共有ファイル・ システム上に置き、セットアッ

プの完了後かつ階層の作成前に作成しなければなりません。

5. アクティブ／アクティブ環境、つまりマルチ・パーティション・サーバー環境では、 設定内の各サーバーは、フェイルオーバー・シナリオですべてのデータベース・インスタンスを実行できる必要があります。特定のシステム・リソース下で可能な DB2 インスタンスまたはパーティション・サーバーの最大数の決定方法については、『IBM Getting Started Guide』を参照してください。

6. 共有ファイル・システムを選択または作成して、そのファイル・システムをエクスポートします（つまり、/export/db2home）。このファイル・システムは DB2 インスタンスのホームとして使用されます。（注記：本要件のひとつの例外として、常に同一のサーバー上で動作しているパーティションがあります。そのケースにおいては、NFS のエクスポートは不要で、インスタンスのホームディレクトリーは、共有ストレージ上に配置することができます。）

7. LifeKeeper NFS リソース階層を作成して、エクスポートしたファイル・システムを保護します。このファイル・システムは、NFS 階層に従属リソースとして含める必要があります。

8. エクスポートされるサーバーも含めて、クラスタの各サーバーに共有ファイル・システムを NFS マウントします。マウント・オプションについては『DB2 Quickstart Guide』を参照してください。DB2 インスタンスの作成時には、インスタンスのホーム・ディレクトリーは NFS でマウントしたファイル・システム上に配置する必要があります。 ファイル・システムのマウント時には、NFS 階層作成時に使用し、 LifeKeeper によって保護されている切り替え可能な IP アドレスを使用します。さらに、ホーム・ディレクトリーのマウント・ポイントを LifeKeeper クラスタの全サーバー の/etc/fstab ファイルで指定しておく必要があります。 構成内の各サーバーは、同一のマウント・ポイント（すなわち/db2/home）にファイル・システムをマウントする必要があります。

   注記： DB2 リソース階層を作成する前に、NFS 階層を作成してテストすることをお勧めします。NFS 階層の作成とテストの詳細な手順については、LifeKeeper for Linux NFS Server Recovery Kit 管理ガイド を参照してください。

9. 構成内の全サーバーについて、次の DB2 環境変数をインスタンスのパーティション総数と同じ値に設定します。 この環境変数を設定するには、インスタンス・オーナーとしてログオンして、db2set コマンドを発行します。この環境変数を調整することで、 考えられるすべてのフェイルオーバー・シナリオに対応できます。

   **db2set DB2_NUM_FAILOVER_NODES=<**インスタンスのパーティション数**>**

10. 以下の DB2 ユーティリティーを使用して、既存の DB2 インスタンスと DB2 管理サーバーをアップデートします。

    db2iupdt および dasiupdt

11. インスタンスのデータを管理するデータベース・パーティション・サーバーを持つクラスタの各サーバーに LifeKeeper DB2 階層を作成する必要があります。 データベースとテーブルスペースは、共有ファイル・システム上に存在する必要があります。データベース・パーティション・サーバーそれぞれと、 NFS のエクスポートされたホーム・ディレクトリーには別個の LUN が必要です。従属リソースには、実際のデータベ ースとテーブルスペースが配置されるファイル・システムが 格納されます。

12. DB2 階層作成後に、保護されていない LifeKeeper ファイル・システム上にデータベ ースを作成する

場合は、 そのファイル・システム用のリソース階層を作成して、 DB2 リソース階層の従属リソースにする必要があります。 この階層は、db2node.cfg ファイルによってサーバー上で実行されることが示されているすべてのパーティション・サーバーを保護します。

13. フェイルオーバーが正しく実行されるようにするには、各データベース・パーティション・サーバーのファイル・システムに、 必ず固有の番号がつけられている必要があります。

    例：

    データベース・パーティション・サーバー node0 のマウント・ポイントは次のようにします。

    **/<FSROOT>/<db2 インスタンス名>/NODE0000**

    データベース・パーティション・サーバー node1 の マウント・ポイントは次のようにします。

    **/<FSROOT>/<db2 インスタンス名>/NODE0001**

    **注記：** この例では、2 台のパーティション・サーバーがあり、各サーバーのファイル・システムは別個の LUN にマウントされています。

14. あるマシンで DB2 階層を正しく作成するには、そのマシンのすべてのデータベース・パーティション・サーバーが動作している必要があります。

15. プライマリ・システムのデータベース・パーティション・サーバーが動作不能になると、サービスは定義済みのバックアップ・システムにフェイルオーバーされます 。バックアップ・システムのデータベース・サービスは、従属リソースがフェイルオーバーされ、データベース・パーティション・サーバーがサービスを開始すると直ちに利用可能に なります。以前に接続されていた DB2 クライアントは接続を解除されるため、バックアップ・サーバーに再接続する必要があります。 コミットされていない SQL 文はすべてロールバックされるため、再入力する必要があります。

# 6.2.4.5. すべての DB2 構成に共通する設定の検討事項

1. DB2 インスタンス名に使用する文字は英数字だけにする必要があります。

2. DB2 クライアントは、LifeKeeper が保護している IP アドレス経由でデータベースに接続するよう設定する必要があります。 ユーザは **$instancehome/sqllib/profile.env** に**“DB2SYSTEM=<浮動 IP>”** と定義して、クライアントの浮動 IP アドレスを カタログすることができます。

3. DB2 リソース階層を保護している構成内の各サーバの/etc/services ファイルは、 保護されているインスタンスに関して、同一のサービス・エントリを持つ必要があります。また、保護されている DB2 インスタンスのユーザ ID、グループ ID、 インスタ ンスのホーム・ディレクトリは、リソースが保護されるすべてのサーバで同じでな くてはいけません。

   DB2では、以下のエントリをデフォルトとして */etc/services* に追加します。

   DB2_db2inst1 60000/tcp

   *DB2_db2inst1_1 60001/tcp*

   *DB2_db2inst1_2 60002/tcp*

   *DB2_db2inst1_END 60003/tcp*

   *db2c_db2inst1 50001/tcp*

4. 回復は、システム・クラッシュ時など、DB2 が突然終了したあとに実行されます。次に、障害発生時に DB2 の回復にかかる時間を大幅に 短縮するためのヒントを挙げます。

   - º DB2 が処理するログ・レコードを制限します。これは、設定パラメータの

**SOFTMAX** と **LOGFILSIZ** を適切に設定することによって 実現できます。ログ・ファイルのサイズを 4MB（1000 個の 4KB ページ）にして、次のようにアク ティブなログ・スペースの量をログ・ファイルの 25%（1MB）に設定して ください。

**db2 UPDATE DB CFG FOR <db 名> USING SOFTMAX 25**

**db2 UPDATE DB CFG FOR <db 名> USING LOGFILSIZ 1000**

º 次のように、負荷を処理するのに十分な数のページ・クリーナが存在するようにします。

**db2 UPDATE DB CFG FOR <db 名> USING NUM_IOCLEANERS <数値>**

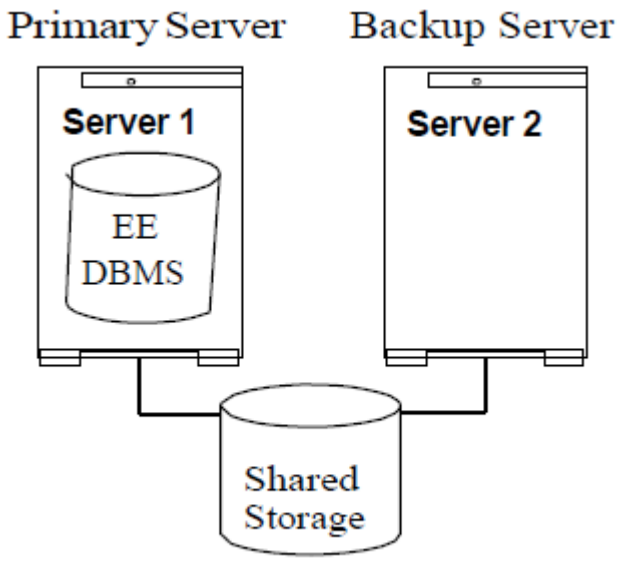
5. DB2 障害モニター(DB2 Fault Monitor)が有効になっている場合は、無効にしてください。

6. 全てのサーバに DB2 をインストールしてください。

# 6.2.4.6. 構成例

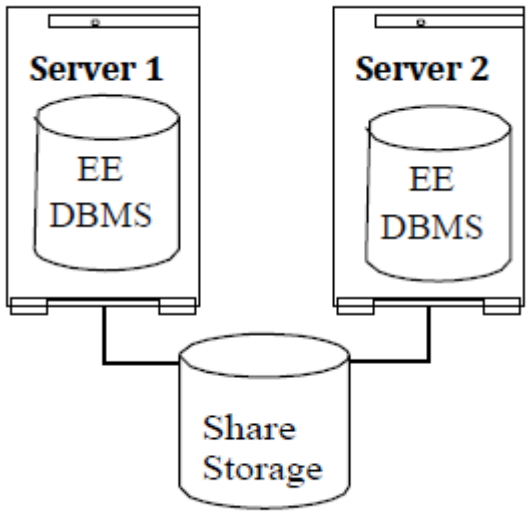
フェイルオーバが発生したときに、LifeKeeper for Linux DB2 Recovery Kit を使用しているとどの ような処理が行われるか、その例をいくつか示します。以下の図では、EE と EEE を使用してデ ータベース設定を示しています。必要に応じて、ESE に置き換えてください。

## 構成 1：DB2シングル・パーティションのアクティブ／スタンバイ構成



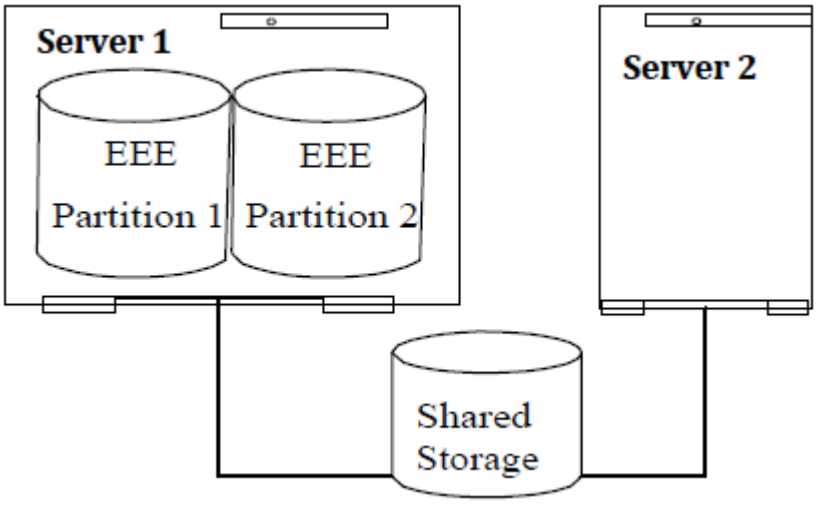

ある DB2 インスタンスがサーバ 1 で保護されていて、別の DB2 インスタンスがサーバ 2 で保護されています。障害発生時には、各サーバが別のサーバのリソースを引き継ぎます。

## 構成 2：DB2シングル･パーティションのアクティブ／アクティブ構成




ある DB2 インスタンスがサーバ 1 で保護されていて、別の DB2 インスタンスがサーバ 2 で保護されています。障害発生時には、各サーバが別のサーバのリソースを引き継ぎます。

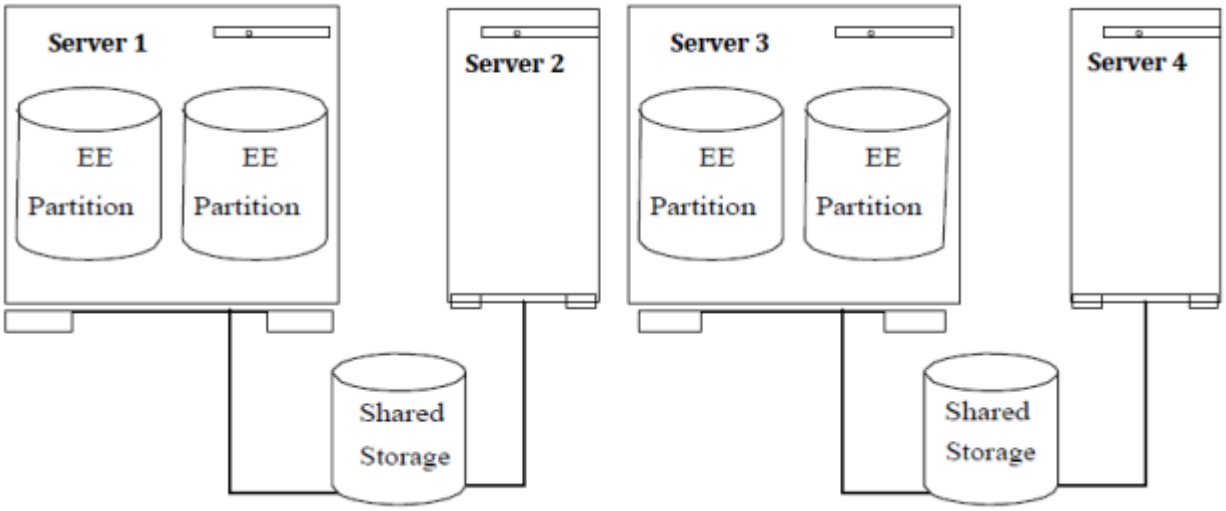
## 構成 3：DB2 マルチ･パーティションのアクティブ／スタンバイ構成（1 クラスタ）




2 つのデータベース･パーティション･サーバを持つ 1 つの DB2 インスタンスが、1 つ の LifeKeeper DB2 リソース階層を持つサーバ 1 で保護されています。障害発生時には、 サーバ 2 が DB2 リソース階層の所有権を引き継ぎます。

注記：以下に挙げるクラスタ群の構成では、どの場合も複数物理ノード上でマルチ･パー ティションを持つ 1 つの DB2 インスタンスだけを保護していることに注意してください。

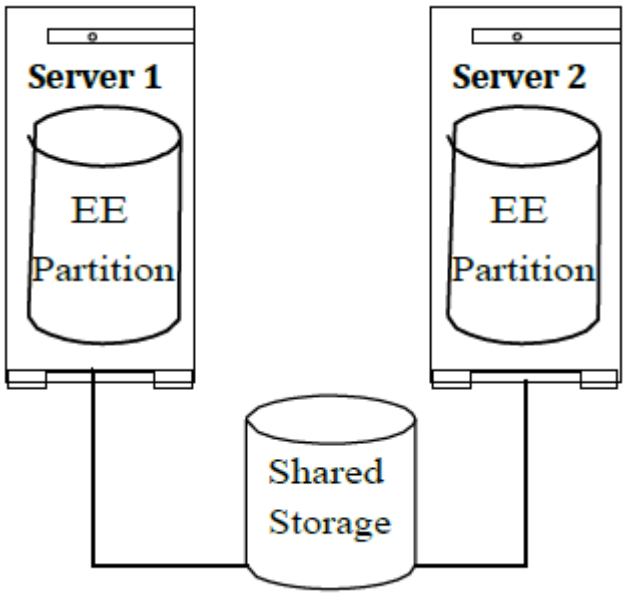
# 構成 4：DB2 マルチ・パーティションのアクティブ／スタンバイ構成（クラスタ群）




2 つのデータベース・パーティション・サーバを持つ 1 つの DB2 インスタンスがサーバ 1 で、2 つのデータベース・パーティション・サーバがサーバ 3 で保護されています。サーバ 1 上に LifeKeeper DB2 リソース階層が 1 つあり、それがサーバ 2 に拡張されていて、サーバ 3 にある別の DB2 リソース階層がサーバ 4 に拡張されています。サーバ 1 で障害が発生すると、サーバ 2 がそのリソースを引き継ぎます。サーバ 3 で障害が発生すると、サーバ 4 がそのリソースを引き継ぎます。

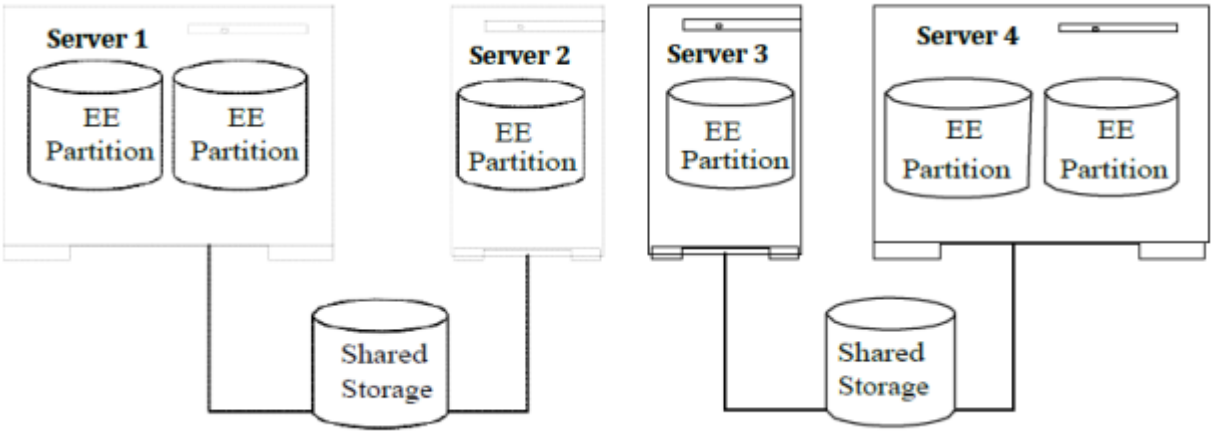
DB2 インスタンスのホーム・ディレクトリをエクスポートしているサーバとそのバック アップ・サーバが同時に動作不能になった場合、DB2 データベースにはアクセスできません。また、エクスポートされた DB2 インスタンス・ディレクトリ（プライマリおよび すべてのバックアップ）の NFS 階層が同時に動作不能になった場合、DB2 データベース は、NFS 階層が回復されるまで、アクセスできません。

## 構成 5：DB2 マルチ・パーティションのアクティブ／アクティブ構成（1 クラスタ）




1 つのデータベース・パーティション・サーバを持つ 1 つの DB2 インスタンスがサーバ 1 で、1 つのデータベース・パーティション・サーバがサーバ 2 で保護されています。 サーバ 1 に LifeKeeper DB2 リソース階層が 1 つあり、サーバ 2 に別の DB2 リソース階層があります。障害発生時には、各サーバが他のサーバのリソースを引き継ぎます。

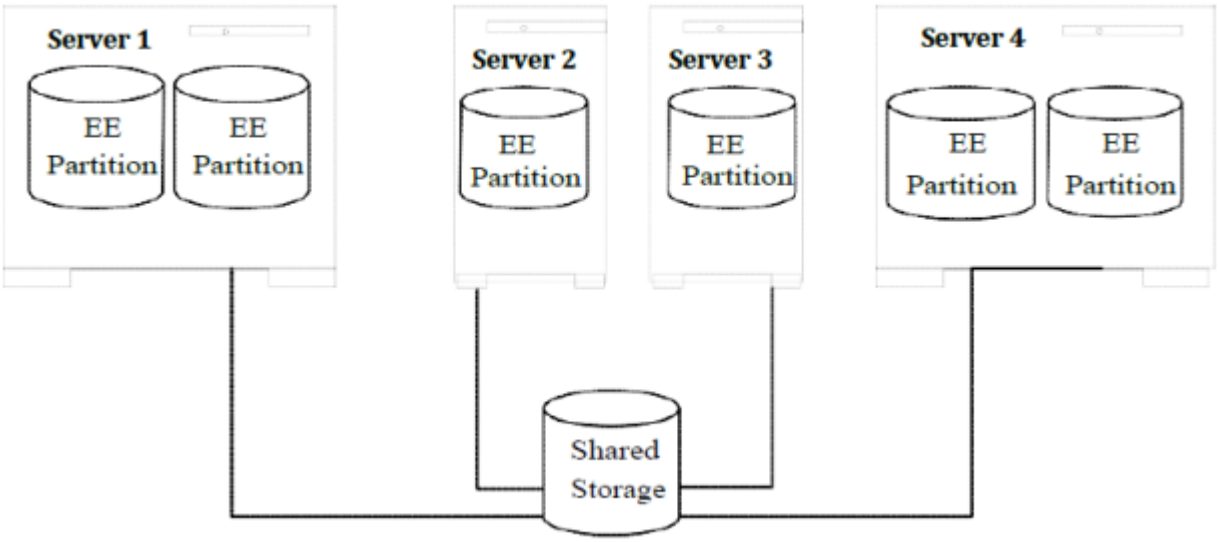
## 構成 6：DB2 マルチ・パーティションのアクティブ／アクティブ構成（クラスタ群）




2 つのデータベース・パーティション・サーバを持つ 1 つの DB2 インスタンスがサーバ 1 で、1 つのデータベース・パーティション・サーバがサーバ 2 で、1 つのデータベース・パーティション・サーバがサーバ 3 で、 2 つのデータベース・パーティション・ サーバがサーバ 4 で保護されています。クラスタの各サーバに LifeKeeper DB2 リソース 階層が 1 つあります。障害発生時には、サーバ 1 とサーバ 2 が互いのリソースを引き継ぎ、サーバ 3 とサーバ 4 が互いのリソースを引き継ぎます。

DB2 インスタンスのホーム・ディレクトリをエクスポートしているサーバとそのバック アップ・サーバが同時に動作不能になった場合、DB2 データベースにはアクセスできません。また、エクスポートされた DB2 インスタンス・ディレクトリ（プライマリおよびすべてのバックアップ）の NFS 階層が同時に動作不能になった場合、 DB2 データベース は、NFS 階層が回復されるまで、アクセスできません。

## 構成 7：DB2 マルチ・パーティション（4 ノード Fibre Channel クラスタ）




2 つのデータベース・パーティション・サーバを持つ 1 つの DB2 インスタンスがサーバ 1 で、1 つのデータベース・パーティション・サーバがサーバ 2 で、1 つのデータベー ス・パーティション・サーバがサーバ 3 で、2 つのデータベース・パーティション・ サーバがサーバ 4 で保護されています。クラスタの各サーバに LifeKeeper DB2 リソース 階層が 1 つあります。 障害発生時には、クラスタの各サーバが互いのバックアップになります。

DB2 インスタンスのホーム・ディレクトリをエクスポートしているサーバとそのバック アップ・サーバが同時に動作不能になった場合、DB2 データベースにはアクセスできません。また、エクスポートされた DB2 インスタンス・ディレクトリ（プライマリおよび すべてのバックアップ）の NFS 階層が同時に動作不能になった場合、 DB2 データベース は、NFS 階層が回復されるまで、アクセスできません。

# 6.2.5. LifeKeeper の設定作業

LifeKeeper for Linux DB2 Recovery Kit の管理作業はすべて、LifeKeeper グラフィカル・ユーザ・インタフェース（GUI）で実行できます。LifeKeeper GUI では、DB2 リソースの設定、管理、監視作業をガイド付きで行えます。

LifeKeeper for Linux DB2 Recovery Kit の設定に関しては、次の作業が可能です。

- リソース階層の作成 。DB2 リソース階層を作成します。
- リソース階層の削除 。DB2 リソース階層を削除します。
- リソース階層の拡張 。DB2 リソース階層をプライマリ・サーバからバックアップ・サーバに拡張します。
- リソース階層の拡張解除 。LifeKeeper クラスタの 1 つのサーバの DB2 リソース階層を 拡張解除（削除）します。
- 従属関係の作成 。既存のリソース階層と別のリソース・インスタンスとの間に子従属関係を作成し、クラスタ内のすべての対象サーバに従属関係の変化を通知します。
- 従属関係の削除 。リソースの従属関係を削除して、クラスタ内のすべての対象サーバ に従属関係の変化を通知します。
- In Service 。リソース階層をアクティブにします。
- Out of Service 。リソース階層を非アクティブにします。
- 表示 / 編集 。 リソース階層のプロパティを表示または編集します。

# 6.2.5.1. リソース階層の作成

プライマリ・サーバで次の作業を行います。

1. **[編集] > [サーバ] > [リソース階層の作成]** を選択します。

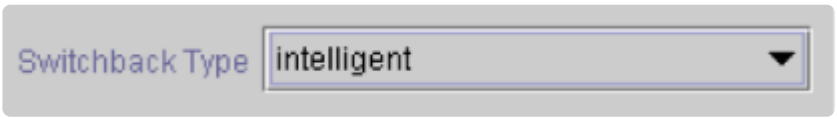
2. **[リカバリー・キットの選択]** ダイアログが表示されます。ドロップダウン・リストから **[DB2 データベース]** を選択します。

   **[次へ]** をクリックして次に進みます。

   **注記：** 階層作成手順のどの段階でも、**[キャンセル]** ボタンをクリックすると、作成プロセス全体がキャンセルされます。

3. **[スイッチバック・タイプ]** ダイアログが表示されます。スイッチバック・タイプによって、フェイルオーバ後、バックアップ・サーバで DB2 リソースがサービス中（アクティブ）になったとき、DB2 リソースをプライマリ・サーバにスイッチバックする方法が決まります。スイッチバック・タイプは intelligent または automatic です。インテリジェント・スイッチバックの場合、リソースをプライマリ・サーバに戻すために管理者の操作が必要ですが、自動スイッチバックの場合は、プライマリ・サーバがオンラインに戻り、LifeKeeper との通信パスが再確立されるとすぐにスイッチバックが実行されます。

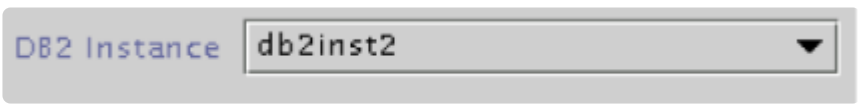
   **[次へ]** をクリックして次に進みます。

4. **[サーバ]** ダイアログが表示されます。DB2 リソースを作成するサーバの名前を選択します（通常は、これがプライマリ・サーバです）。クラスタ内のすべてのサーバがドロップダウン・リスト・ボックスに表示されます。

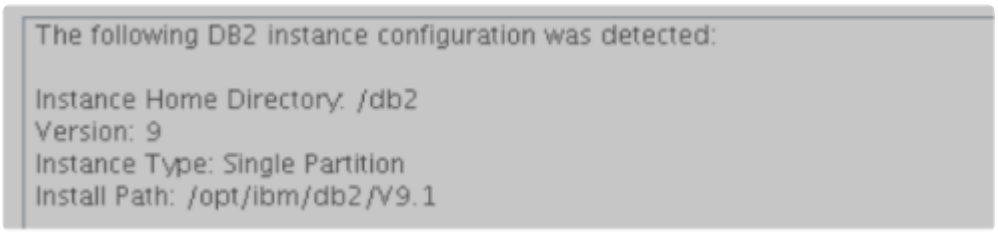
   **[次へ]** をクリックして次に進みます。

5. **[DB2 インスタンス]** ダイアログが表示されます。保護する DB2 インスタンスの名前を選択するか入力します。

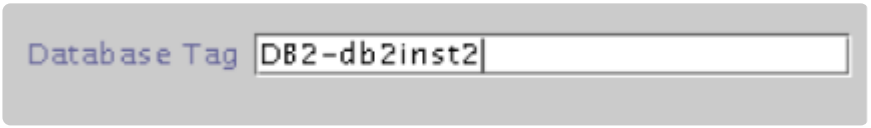
**[次へ]** をクリックして次に進みます。

6. 検出されたインスタンスに関する情報を示す情報ボックスが表示されます。



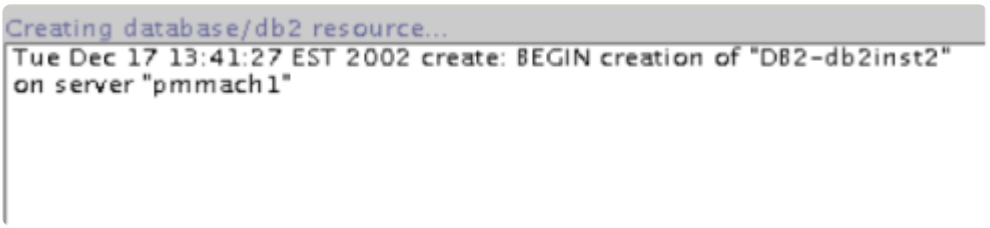

**[続行]** をクリックして次に進みます。

7. **[データベース・タグ]** ダイアログが表示されます。このダイアログでは、新しい DB2 データベース・リソース・インスタンス用の他と重複しないタグ名が自動的に入力されています。



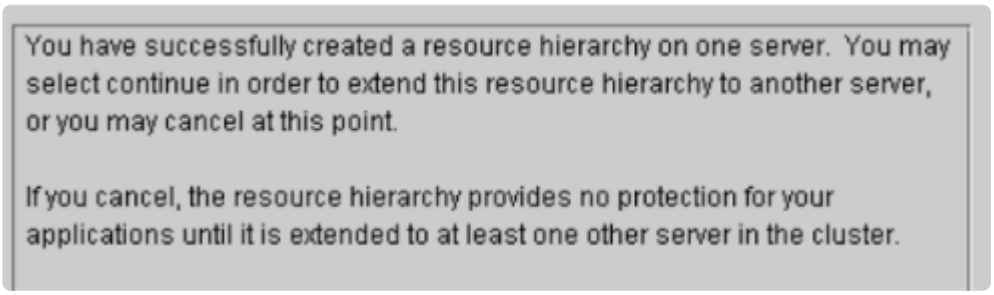

**[作成]** をクリックして次に進みます。

8. 階層の作成が開始されたことを示す情報ボックスが表示されます。

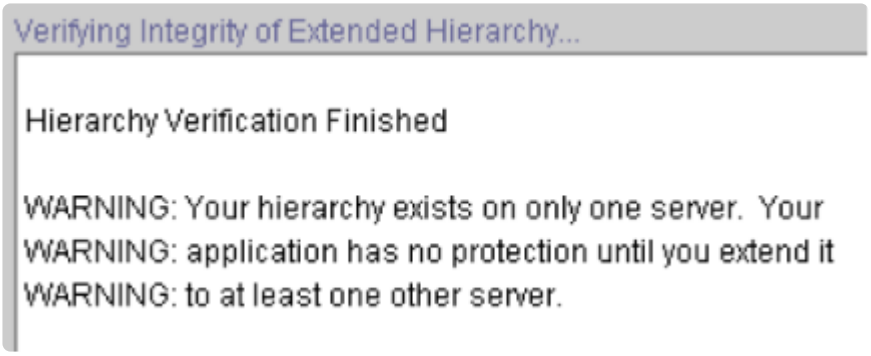



**[次へ]** をクリックして次に進みます。

9. DB2 リソース階層の作成に成功したことを示す情報ボックスが表示されます。リソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにリソース階層を拡張する必要があります。




リソースを拡張する場合は、**[続行]** をクリックします。 後でリソースを拡張する場合は、**[キャンセル]** をクリックします。




10. **[完了]** をクリックして、**[リソース階層の作成]** メニューを終了します。

# 6.2.5.2. リソース階層の削除

プライマリ・サーバで次の作業を行います。LifeKeeper 構成内のすべてのサーバから DB2 リソースを削除するには、次の手順を実行します。

1. **[LifeKeeper GUI]** メニューから **[編集]** を選択し、次に **[リソース]** を選択します。ドロップダウン・メニューから、**[リソース階層の削除]** を選択します。

2. DB2 リソース階層を削除するターゲット・サーバの名前を選択します。

   **注記：** グローバル・リソースの左ペインまたは個々のリソース・インスタンスの右ペインで右クリックして **[リソースの削除]** を選択した場合は、このダイアログは表示されません。

   **[次へ]** をクリックして次に進みます。

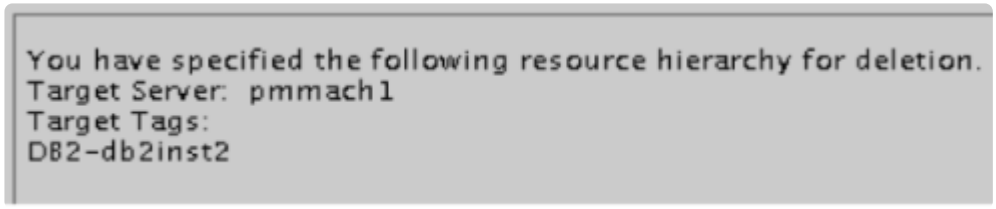
3. **[削除する階層]** を選択します。削除するリソース階層を特定して、強調表示にします。

   **注記：** グローバル・リソースの左ペインまたは個々のリソース・インスタンスの右ペインで右クリックして **[リソースの削除]** を選択した場合は、このダイアログは表示されません。

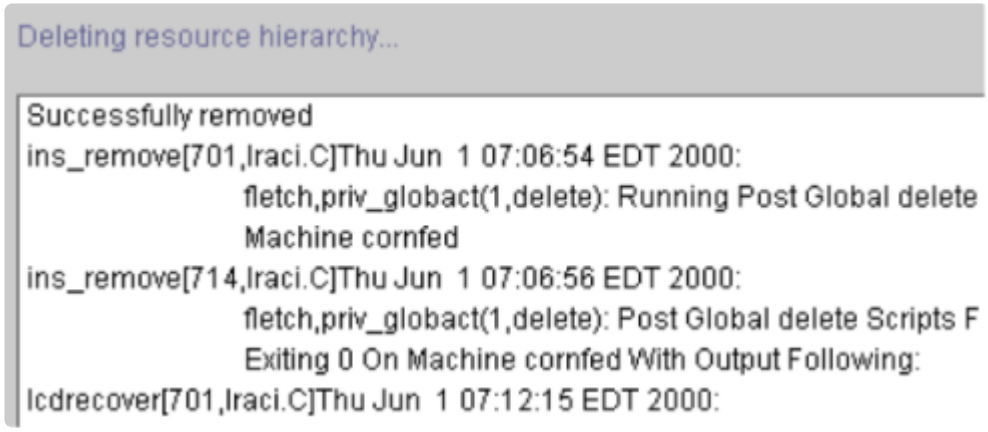
   **[次へ]** をクリックして次に進みます。

4. 選択したターゲット・サーバと、削除の対象として選択した階層を確認する情報ボ ックスが表示されます。




**[削除]** をクリックして次に進みます。

5. DB2 リソース・インスタンスの削除に成功したことを示す情報ボックスが表示されます。




6. **[完了]** をクリックして、**[リソース階層の削除]** メニューを終了します。

# 6.2.5.3. 階層の拡張

階層は、作成後に、クラスタ内の別のサーバに拡張する必要があります。リソース・インスタンスを拡張するには、次の 3 つの方法が可能です。

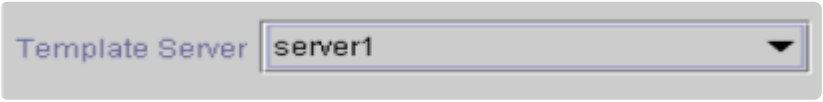
1. DB2 リソース階層の作成に成功したときに、**[続行]** を選択して、バックアップ・サーバにリソース階層を拡張する処理に進むことができます。

2. LifeKeeper GUI の左ペインまたは右ペインで拡張されていない階層を右クリックします。

3. LifeKeeper GUI で、ドロップダウン・メニューから **[編集] > [リソース] > [リソース階 層の拡張]** を選択します。選択すると、**[リソース階層の拡張]** ウィザードが起動されま す。**[リソース階層の拡張]** オプションの **[デフォルトを受け入れる]** ボタンは、 LifeKeeperの **[リソース階層の拡張]** の既定値をよく知っていて、値の入力や確認をしないで素早く LifeKeeper リソース階層を拡張したいユーザ向けです。GUI ダイアログを使用して対話的に段階を追って LifeKeeper リソース階層を拡張する場合は、 **[次へ]** ボタンを選択します。

   a. 最初に表示されるダイアログ・ボックスでは、DB2 リソース階層が現在サービス中であるテンプレート・サーバを選択します。 **[テンプレート・サーバ]** で選択す るテンプレート・サーバと、次のダイアログ・ボックスの **[拡張するタグ]** で選択する拡張するタグが、サービス中（アクティブ）のリソース階層を表すことを忘れないでください。選択したテンプレート・サーバでサービス中になっていない リソース・タグを選択すると、エラー・メッセージが表示されます。このダイアログのドロップダウン・ボックスには、クラスタ内の全サーバの名前が示されます。

   **注記：** DB2 リソース階層作成後、引き続きリソース階層の拡張作業に入った場合は、作成段階でテンプレート・サーバが特定されているため、このダイアログ・ボックスは表示されません。これは、GUI ウィンドウの左ペインの DB2 リソース・アイコンまたは右ペインの DB2 リソース・ボックスを右クリックして、 **[リソース階層の拡張]** を選択した場合も同様です。

   **注記：** 階層拡張の途中で **[キャンセル]** ボタンをクリックすると、拡張処理が取り消されます。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

   **[次へ]** をクリックして次に進みます。

   b. **[拡張するタグ]** を選択します。これは、テンプレート・サーバからターゲット・サーバに拡張する DB2インスタンスの名前です。ウィザードのドロップダウン・リスト・ボックスには、テンプレート・サーバ上に作成されているすべてのリソースが表示されます。

   **注記：** ここでも、DB2 リソース階層の作成後すぐにリソース階層の拡張作業に進んだ場合は、作成段階ですでにリソースのタグ名が特定されているため、このダイアログ・ボックスは表示されません。これは、GUI ウィンドウの左ペインの DB2 リソース・アイコンまたは右ペ

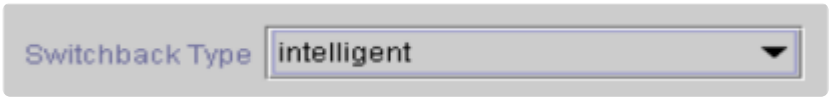
インの DB2 リソース・ボックスを右クリックし て、[リソース階層の拡張]を選択した場合も同様です。

**[次へ]** をクリックして次に進みます。

c. DB2 リソース階層を拡張するターゲット・サーバを選択します。

**[次へ]** をクリックして次に進みます。

d. **[スイッチバック・タイプ]** ダイアログが表示されます。スイッチバック・タイプによって、フェイルオーバ後、バックアップ・サーバで DB2 リソースがサービス中（アクティブ）になったとき、DB2 リソースをプライマリ・サーバにスイッ チバックする方法が決まります。スイッチバック・タイプは intelligent または automatic です。インテリジェント・スイッチバックの場合、リソースをプライマリ・サーバに戻すために管理者の操作が必要ですが、自動スイッチバックの場合 は、プライマリ・サーバがオンラインに戻り、LifeKeeper との通信パスが再確立されるとすぐにスイッチバックが実行されます。




**[次へ]** をクリックして次に進みます。

e. テンプレートの優先順位を選択または入力します。これはサーバで現在サービス中の DB2 階層の優先順位です。優先順位は、1～999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります（数字 1 が最高の優先順位に相当し ます）。拡張操作時には、別のシステムで既に使用中の優先順位をこの階層に対 して指定することはできません。デフォルト値を推奨します。

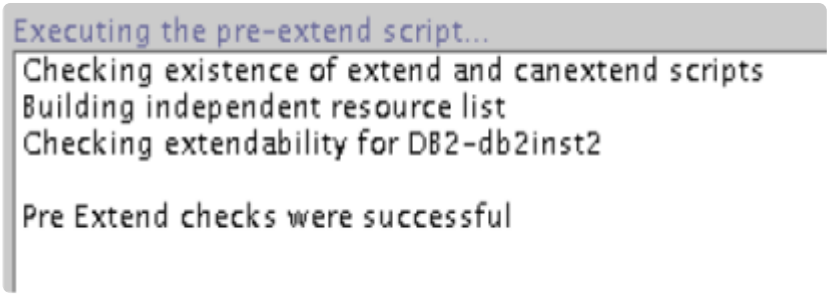
**注記：** このフィール ドは階層を最初に拡張するときだけ表示されます。

**[次へ]** をクリックして次に進みます。

f. ターゲットの優先順位を選択または入力します。これは、別サーバにある同等の階層に対する新しく拡張する DB2 階層の優先順位です。1～999 の範囲の未使用の値が有効で、リソースの連鎖フェイルオーバ・シーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位は高くなります（1 は最高の優先 順位を表わします）。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。

**[次へ]** をクリックして次に進みます。

g. 環境がチェックされ、このリソースを拡張するのに必要な要件がすべて満たされていることを示す情報ボックスが表示されます。満たされていない要件がある場合、**[次へ]** ボタンが使用できなくなり、**[戻る]** ボタンが有効になります。




リソースの拡張の設定を変更する場合は、**[戻る]** ボタンをクリックします。 後でリソースを拡張する場合は、**[キャンセル]** をクリックします。リソース階層の拡張を実行する場合は、**[次へ]** をクリックします。

**[終了]** をクリックして、DB2 リソース・インスタンスの拡張に成功したことを確認します。

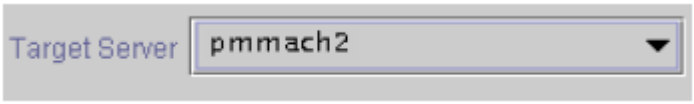
4. **[完了]** をクリックして、**[リソース階層の拡張]** メニューを終了します。

**注記：** 必ず両方のサーバで新しいインスタンスの機能をテストしてください。

# 6.2.5.4. 階層の拡張解除

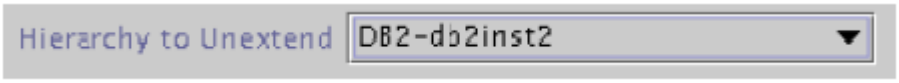
1. LifeKeeper GUI メニューで、**[編集] > [リソース] > [リソース階層の拡張解除]** を選択します。
2. リソースが現在サービス中（アクティブ）のサーバは選択できません。

   **注記：** グローバル・リソースの左ペインまたは個々のリソース・インスタンスの右ペ インで右クリックして **[拡張解除]** を選択した場合は、このダイアログは表示されません。

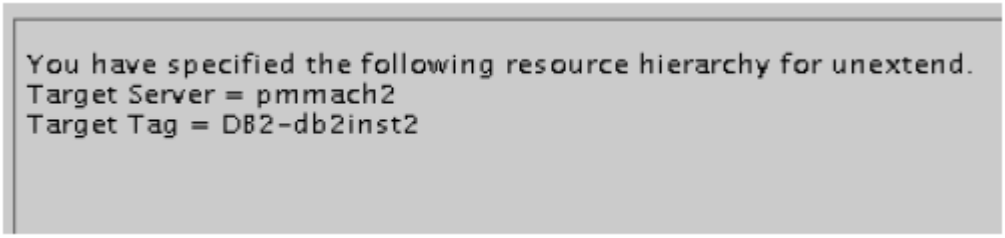
   
   

   **[次へ]** をクリックして次に進みます。
3. **[拡張解除する階層]** を選択します。

   **注記：** グローバル・リソースの左ペインまたは個々のリソース・インスタンスの右ペ インで右クリックして **[拡張解除]** を選択した場合は、このダイアログは表示されません。

   **[次へ]** をクリックして次に進みます。
4. 拡張解除するために選択したターゲット・サーバと DB2 リソース階層を確認する情報ボックスが表示されます。

   **[拡張解除]** をクリックします。
5. DB2 リソースの拡張解除に成功したことを示す別の情報ボックスが表示されます。
6. **[完了]** をクリックして、**[リソース階層の拡張解除]** メニューを終了します。

# 6.2.5.5. リソース階層のテスト

DB2 リソース階層のテストは、手動でサーバを切り替えて、プライマリ・サーバからバックアップ・サーバへのリソース・インスタンスのフェイルオーバをシミュレートすることによって行えます。

## LifeKeeper GUI からの手動切り替えの実行

LifeKeeper GUI で **[編集] > [リソース] > [サービス中]** を選択すると、手動切り替えを開始できます。たとえば、バックアップ・サーバでサービス中リクエストが実行されると、DB2 リソース階層がバックアップ・サーバ側でサービス中になり、プライマリ・サーバ側ではサービス休止になります。この時点で、元のバックアップ・サーバがプライマリ・サーバになり、元のプライマリ・サーバがバックアップ・サーバになります。

[サービス休止]要求を実行すると、リソース階層は他のサーバでサービス中にならずに停止されます。再同期中にリソースがサービス休止になった場合、サービス中に戻すことができるのは休止になったのと同じサーバ上のみです。

**重要：** バックアップ・サーバでリソース階層をサービス中にしたあとは、データベースに接続してみる必要があります。特に、RAW デバイスをテーブルスペース・コンテナとして使用している場合は接続を試みる必要があります。この作業は、バックアップ・サーバですべてのディスク・パーティションが認識されており、RAW バインディングが正しく確立されていることを確認するために必要です。

これは、すべてのディスク・パーティションがバックアップ・サーバ側から見えるようにして、RAW バインディングを正しく確立するために必要です。バックアップ・サーバで RAW バインディングが確立されていない場合、その原因として最も可能性が高いのは、新しいパーティションをプライマリ・サーバに作成して構成に追加したのに、バックアップ・サーバ側のパーティション・テーブルが更新されていないためです。

解決策は、パーティション・テーブルが正しくアップデートされるように、バックアップ・サーバを再起動することです。

# 6.2.6. トラブルシューティング

| 症状 | 考えられる要因 |
|---|---|
| DB2 EEE パーティション・サーバが 起動に失敗します | **db2nodes.cfg** ファイルのポート番号が **/etc/services** ファイルで設定されている範囲を超えている可能性があります。 **db2nodes.cfg** ファイルで設定されているポート番号を調べて、 **/etc/services** のポート番号の範囲におさまっているか確認します。 |
| LifeKeeperの「サービス中」操作または「サービス休止」操作でハングします | DB2 の環境変数 **DB2_NUM_FAILOVER_NODES** が適切に設定されていない可能性があります。設定内のすべてのサーバについて、この環境変数を、インスタンス内のパーティション総数と同じ値に設定します。<br><br>例： **db2set DB2_NUM_FAILOVER_NODES =**<インスタンス内のパーティション数> |
| LifeKeeperの「サービス中」操作がハングします | DB2 管理サーバで **dasupdt** コマンドを実行していない可能性があります。DB2 管理サーバで **dasupdt** コマンドを実行したか確認します。 |
| LifeKeeperが最初のスイッチオーバ操作に失敗します。 | DB2 Fenced ユーザーが、バックアップサーバ上で作成されていない可能性があります。指定したインスタンスに対する DB2 Fenced User がプライマリと同じユーザーおよびグループ IDと存在しているかを確認してください。保護対象のインスタンスは Administration Server groupのメンバーであることを必ず確認してください。 |
| 既存の DB2 リソース階層に新しいノードを追加する必要があります | 既存の LifeKeeper DB2 リソース階層に新しいノードを追加する方法の詳細については、nodesユーティリティのマニュアル・ページを参照してください。 |
| 管理サーバが起動に失敗します。 | 別の管理サーバが、指定されたポートですでに実行されていないか確認します。 |
| DB2 リソース階層の作成に長い時間がかかります。 | Usage: adddelpv <VG tag> [addlist\|dellist]大きな DB を持っている DB2 インスタンスを保護しようとしているために、リソース作成処理に時間がかかっている可能性があります。リソース作成操作を行う前にアクティベートを行ってください。 |

## エラーメッセージ

メッセージカタログ は、SIOS Protection Suite for Linux の使用時に検出されるすべてのメッセージのリストを示すとともに、適宜、エラーの原因とエラー状態を解決するために必要な処置の説明を提供します。このリストから受け取ったエラーコードを探すことができます。また、DB2 Recovery Kit の使用時に検出されるすべてのメッセージがリストされた DB2 Kit メッセージカタログ を直接参照してもかまいません。

# 6.2.7. 付録： DB2 で RAW I/O を使用する設定

DB2 インスタンスが LifeKeeper で保護されるように、DB2 で RAW I/O デバイスを設定するにはいくつか要件があります。

## 要件

- Linux OS が RAW I/O デバイスをサポートしている必要があります。ほとんどのディストリビューションでは、カーネル 2.4 でサポートされましたが、カーネル 2.2 からサポートしているディストリビューションもあります。

- すべての RAW I/O デバイスを共有ディスク･パーティションにバインドする必要があります。共有 SCSI ディスク･パーティションの数が必要になります。正確な数は、 RAW I/O デバイスに置くテーブルスペースの数によって決まります（RAW デバイスにテーブルスペースを書き込む場合のガイドラインについては、DB2 のドキュメ ントを参照してください）。

- DB2 Version 7.1 Fix Pack 3 以降または DB2 Version 8 もしくはそれ以降が必要です。

## RAW I/O の設定手順

以下の手順 1～4 は、『IBM Db2 Universal Database リリース･ノート バージョン 7.2/バー ジョン 7.1 フィックスパック 3』の第 7.3.1.1 節（「Linux でのロー I/O の使用法」）から抜き出したものです。 この例では、使用する RAW パーティションは/dev/sda5 です。ここ には重要なデータを置かないでください。

手順 4 または手順 5 は、複数論理ノードを使用しているかどうかで異なることに注意してください。

1. このパーティションの 4096 バイト･ページの数を計算し、必要に応じて端数を切り捨てます。例：

   例:

   ```
   # fdisk /dev/sda

   Command (m for help):p

   Disk /dev/sda:255 heads, 63 sectors, 1106 cylinders

   Units = cylinders of 16065 * 512 bytes
   ```

| Device Boot | Start | End | Blocks | ID | System |
|---|---|---|---|---|---|
| dev/sda1 | 1 | 23 | 4200997 | 83 | Linux |
| /dev/sda2 | 524 | 1106 | 4682947+ | 5 | Extended |
| /dev/sda5 | 524 | 1106 | 4682947 | 83 | Linux |

   ```
   Command (m for help):q
   ```

```
#
```

/dev/sda5 のページ数：

```
num_pages = floor( ((1106-524+1)*16065*512)/4096 )

num_pages = 11170736
```

2. このパーティションに、未使用の RAW デバイスをバインドします。これはマシンを再起動するたびに実行する必要があり、root アクセスを必要とするため、システム初期化ファイル（rc.local または boot.local）に RAW バインディングを追加することもできます。 このバインディングは、階層を LifeKeeper の保護下においたあと、削除する必要があります。LifeKeeper は、LifeKeeper が保護している RAW I/O デバイスの RAW バインディングを再確立します。

   次のように、raw -qa を使用して、どのロー・デバイス・ノードが使用中か確認します。

   ```
   raw /dev/raw/raw1 /dev/sda5

   /dev/raw/raw1:bound to major 8, minor 5
   ```

3. RAW デバイス・コントローラとディスク・パーティションにグローバル読み取りパーミッションを設定します。RAW デバイスにグローバル読み書きパーミッションを設定します。

   ```
   # chmod a+r /dev/rawctl

   # chmod a+r /dev/sdb1

   # chmod a+rw /dev/raw/raw1
   ```

4. **重要：** この手順は、DB2 EE を使用している場合、または、DB2 EEE 構成でフェイルオーバ後も複数論理ノード（MLN）を決して実行しない場合にだけ、あてはまります。 ある時点で MLN を実行する可能性がある場合は、手順 5 に進みます。

   ディスク・パーティションではなくロー・デバイスを指定して、DB2 にテーブルスペースを作成します。

   例：

   ```
   CREATE TABLESPACE dms1

   MANAGED BY DATABASE

   USING (DEVICE '/dev/raw/raw1' 11170736)
   ```

   ロー・デバイスのテーブルスペースは、DB2 がサポートしている他のページ・サイズでもサポートされます。

5. **重要：** この手順は、MLN を実行している場合、またはフェイルオーバ後のある時点 で MLN を実行

する場合に行う必要があります。

ディスク・パーティションではなくロー・デバイスを指定して、DB2 にテーブルスペースを作成し、各 DB2 インスタンス・パーティションに異なる RAW I/O デバイス を指定します。

例：

```
CREATE TABLESPACE dms1

MANAGED BY DATABASE

USING (DEVICE '/dev/raw/raw1' 11170736) on NODE (NODENUM)

USING (DEVICE '/dev/raw/<異なる RAW デバイス・ノード>' ####### ) on NODE
(NODENUM)
```

**注記：** 各 DB2 ノード（データベース・パーティション・サーバー）が異なる RAW I/O デバイスを使用する必要があるため、ON NODE を使用する必要があります。 フェイルオーバが正しく動作するように、異なるマシンでノードが実行されている場合でもON NODE を指定する必要があります。

# 6.3. Recovery Kit for EC2 管理ガイド

## Recovery Kit for EC2

Recovery Kit for EC2 は、障害の発生したプライマリサーバから Elastic IP をバックアップサーバに復旧する仕組みを提供します。また、複数のAvailability Zoneで IP Recovery Kit を動作可能にする仕組みも提供します。

Recovery Kit for EC2 の定義、シナリオ、および運用の比較と詳細については、運用の原則 を参照してください。

### SIOS Protection Suite ドキュメンテーション

以下は、SIOS Technology Corp が発行している SIOS Protection Suite for Linux 関連ドキュメントの一覧です。

- SPS for Linux テクニカルドキュメンテーション
- SPS for Linux リリースノート
- SIOS Technology Corp. ドキュメンテーション

詳細については、Amazon Elastic Compute Cloud (EC2) ドキュメント を参照してください。

「Amazon Web Services」、「Powered by Amazon Web Services」のロゴ、「AWS」、「Amazon EC2」、「EC2」、「Amazon Elastic Compute Cloud」、「Amazon Route 53」、「Amazon Virtual Private Cloud」および「Amazon VPC」は、米国その他の国における Amazon.com, Inc. またはその関連会社の商標です。なお、本文および図表中では、「™」、「®」は明記しておりません。(一部記載がございます。)

# 6.3.1. 運用の原則

Recovery Kit for EC2 には機能が 2 つあります。

1. ルートテーブルシナリオ (バックエンドクラスタ) は、LifeKeeper が保護する IP リソースに Amazon VPC™ 内のクライアントから接続できるように、ルートテーブルを管理します。

2. Elastic IP シナリオ (フロントエンドクラスタ) は、インターネットから使用できる Elastic IP を管理します。

## ルートテーブルシナリオ (バックエンドクラスタ):

ルートテーブルの管理と運用について理解しやすいように、図 1 に示すシナリオについて考えてみましょう。

この設定例は、1 つの Amazon VPC™ と 2 つの利用可能ゾーン (AZ) で構成されています。

各 AZ にサブネットが 2 つあります。

- 1 つ目のサブネット (以下「パブリックサブネット」) は、ルートテーブル別のインターネットゲートウェイを経由してインターネットに接続します (10.0.1.0/24 および 10.0.3.0/24 のルートテーブルを参照)。

- 2 つ目のサブネット (以下「プライベートサブネット」) は、ルートテーブル別のNATインスタンスを経由してインターネットに接続します (10.0.2.0/24 および 10.0.4.0/24 のルートテーブルを参照)。

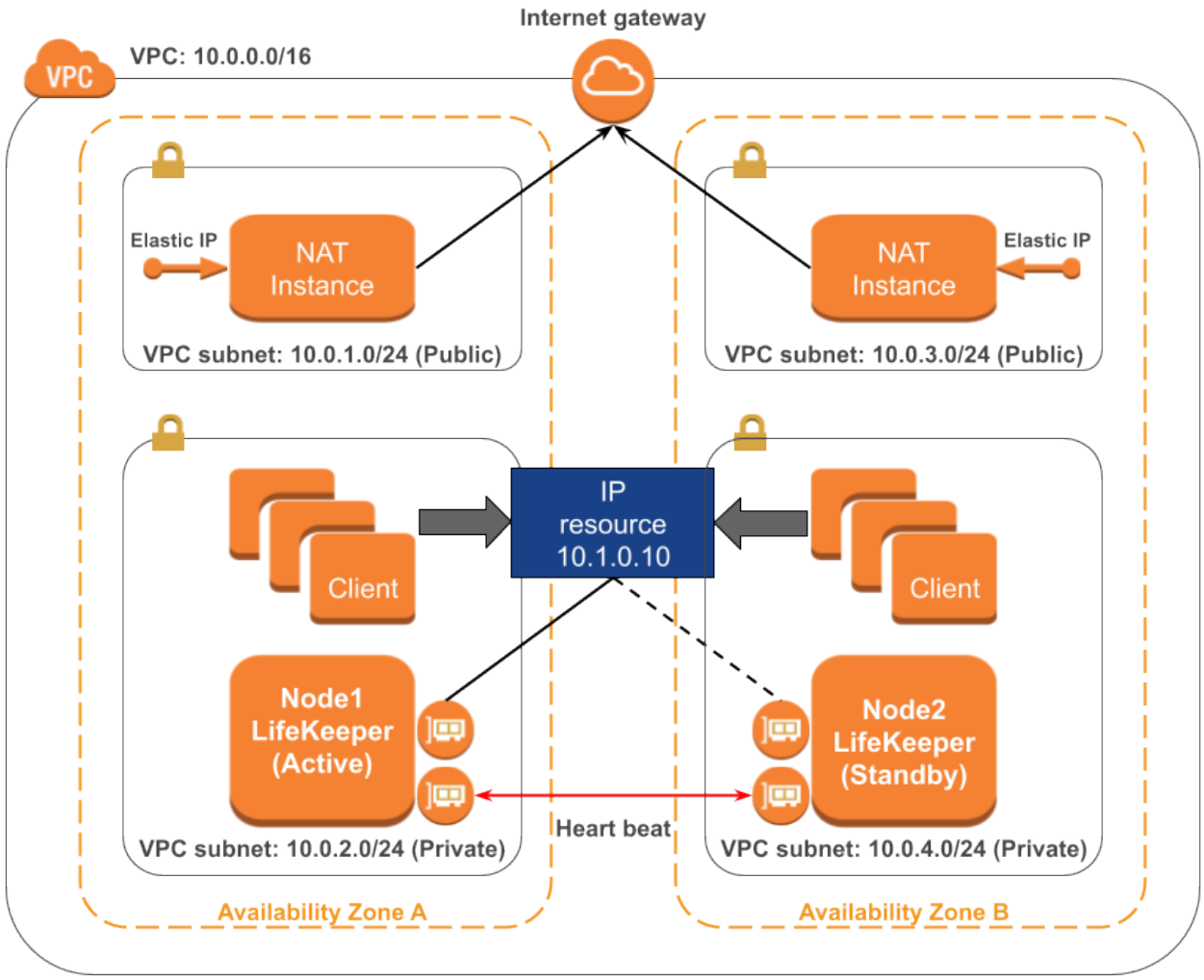
各パブリックサブネットには、NAT の Elastic IP を割り当てる EC2 インスタンス (以下「NAT インスタンス」) が 1 つあります。

各プライベートサブネットには、LifeKeeper のアクティブ / スタンバイ (以下それぞれ「Node1」および「Node2」) の EC2 インスタンスが 1 つあり、Node1/Node2 により保護されたアプリケーションを使用するクライアントが複数あります。

Node1/Node2 はそれぞれ、Elastic Network Interface (ENI) を 2 つ持ちます。

各インスタンスと各ノードとの間で通信が可能なように、ネットワーク ACL とセキュリティグループを設定します。

図 1: ルートテーブルシナリオ




## 10.0.1.0/24 および 10.0.3.0/24 のルートテーブル

| 接続先 | ターゲット | 注記 |
|---|---|---|
| 10.0.0.0/16 | ローカル | デフォルト |
| 0.0.0.0/0 | インターネットゲートウェイ | インターネットに接続するには、Elastic IP を割り当てる必要があります。 |

## 10.0.2.0/24 のルートテーブル

| 接続先 | ターゲット | 注記 |
|---|---|---|
| 10.0.0.0/16 | ローカル | デフォルト |
| 10.1.0.10/32 (IP リソース) | LifeKeeper のアクティブなノード上の Elastic Network Interface (ENI) | このターゲットは、スイッチオーバ中に Recovery Kit for EC2 により更新されます。 |
| 0.0.0.0/0 | NAT インスタンス (10.0.1.0) | NAT 経由でインターネットに接続 |

### 10.0.4.0/24 のルートテーブル

| 接続先 | ターゲット | 注記 |
|---|---|---|
| 10.0.0.0/16 | ローカル | デフォルト |
| 10.1.0.10/32 (IP リソース) | LifeKeeper のアクティブなノード上の Elastic Network Interface (ENI) | このターゲットは、スイッチオーバ中に Recovery Kit for EC2 により更新されます。 |
| 0.0.0.0/0 | NAT インスタンス (10.0.3.0) | NAT 経由でインターネットに接続 |

リソースのスイッチオーバが実行されると、LifeKeeper は Node1 の IP リソースを Out of Service にします。各プライベートサブネット内の 10.1.0.10/32 のターゲットエントリは、Node2 の ENI を反映するように更新されます。Node2 の IP リソースが In-Service になります。このため、IP アドレス 10.1.0.10 へのトラフィックは、プライベートサブネット内でのルートテーブル設定の変更により効果的に Node2 に転送されます。

パブリックサブネットを含む他のサブネットから IP アドレス 10.1.0.10 にアクセスする必要がある場合、それぞれのサブネットのルートテーブルのエントリに接続先 10.1.0.10/32 のルートを追加してください。LifeKeeperは、VPC内のルートテーブルで接続先が "10.1.0.10/32" になっている全てのエントリを制御します。

## Elastic IP シナリオ (フロントエンドクラスタ):

Elastic IP の管理と運用について理解しやすいように、図 2 に示すシナリオについて考えてみましょう。

この設定例は、1 つの Amazon VPC™ と 2 つの利用可能ゾーン (AZ) で構成されています。

各 AZ にサブネットが 1 つあります。

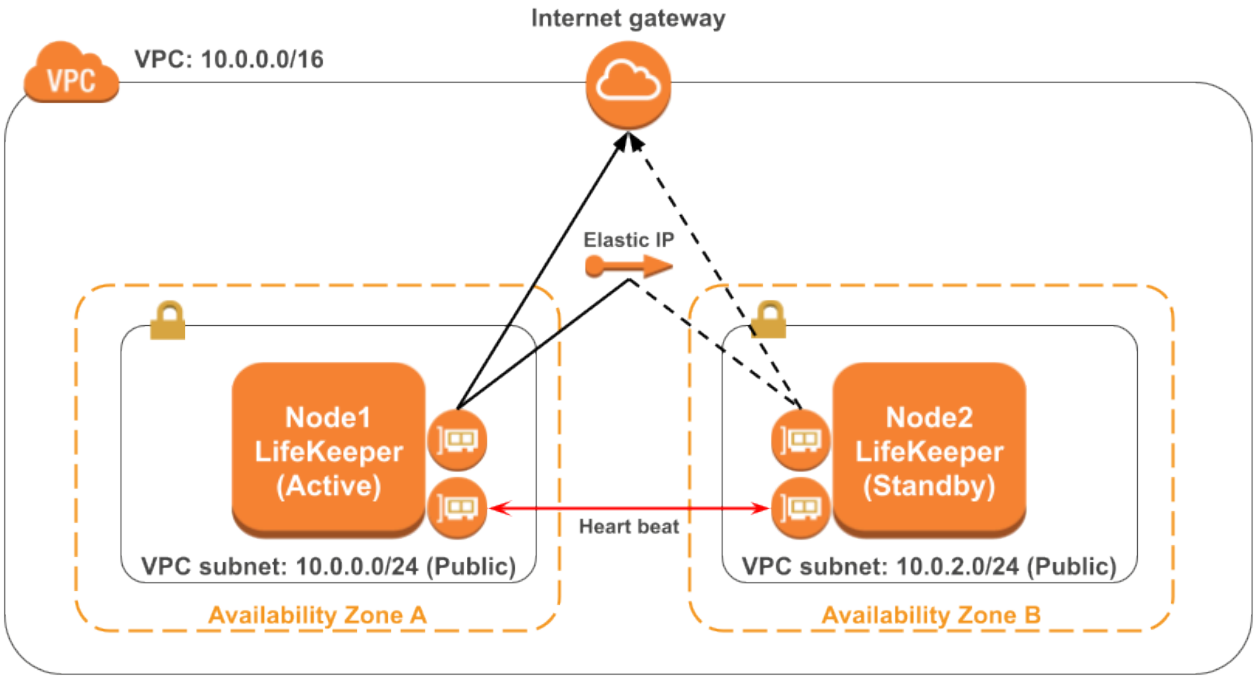
各サブネットは、ルートテーブル別のインターネットゲートウェイを経由して、インターネットに接続します。

サブネットには、LifeKeeper のアクティブ / スタンバイ (以下それぞれ「Node1」および「Node2」) の EC2 インスタンスが 1 つあります。

Node1/Node2 はそれぞれ、Elastic Network Interface (ENI) を 2 つ持ちます。

各ノード間で通信が可能なように、ネットワーク ACL とセキュリティグループを設定します。

図 2: Elastic IP シナリオ




システム管理者が、フロントエンドクラスタの Elastic IP アドレスを ENI に割り当てます。

Node1 をリソースのプライマリサーバと仮定すると、システム管理者は[リソース階層の作成](#) セクションで説明しているウィザードを使用して、Node1 上に EC2 リソース階層を作成します。

リソースのスイッチオーバが実行されると、Recovery Kit for EC2 は Node1 の ENI と Elastic IP との関連付けを解除します。Recovery Kit for EC2 は、Elastic IP が Node2 の ENI と関連付けられているかどうかを調べ、関連付けられていない場合は、Elastic IP を ENI に関連付けます。このため、インターネット上のクライアントは、スイッチオーバ後に Elastic IP 経由で Node2 に接続できます。

注記：EC2インスタンスを制御するために、スタンバイノードもエンドポイントにアクセスできる必要があります。即ち、VPC 外部への接続が必要ですので、ご注意ください。詳しくは「[要件](#) 」を参照してください。
なお、PrivateLink を利用すればエンドポイントへのアクセスにパブリック IP アドレスを必要としません。詳しくは「[VPC エンドポイント](#) 」を参照してください。

# 6.3.2. 要件

Recovery Kit for EC2 のインストール / アンインストールを試みる前に、Amazon Web Service のソフトウェア要件と Recovery Kit for EC2 のインストール / アンインストールの手順を理解しておく必要があります。

## Amazon Web Service およびソフトウェアの要件

Recovery Kit for EC2 をインストールして設定する前に、使用している設定が以下の要件を満たしていることを確認してください。

### Amazon Virtual Private Cloud (VPC):

- この Recovery Kit を使用するには、VPC を AWS 内に設定する必要があります。
- 異なるAvailability Zone (AZ) に作成された 2 つ以上のサブネット
- 各サブネットに、関連するルートテーブルがあります。
- パブリック (フロントエンド) クラスタを設定する場合、Elastic IP を 1 つ以上割り当てる必要があります。

### Amazon Elastic Compute Cloud (EC2):

- この Recovery Kit を使用するには、EC2 インスタンスが 2 つ以上必要です。
- インスタンスは、各サブネット上で関連付けられます。
- インスタンスは、Elastic Network Interface (ENI) に接続されます。
- AWS Command Line Interface (AWS CLI) を全ての EC2 インスタンスにインストールする必要があります。インストール方法は、「AWS Command Line Interface のインストール」を参照してください。
- 全ての EC2 インスタンスで、Amazon EC2 サービスのエンドポイント（AWS のリージョンとエンドポイント）にプロトコル HTTP および HTTPS を使用してアクセスできる必要があります。EC2 および OS の設定を適切に行ってください。
- Amazon EC2 インスタンスのメタデータを取得するため、IPアドレス 169.254.169.254 にプロトコル HTTP でアクセスできる必要があります。
- AWS CLI を用いているため、TCP ポート 443 でのアウトバウンド接続を有効にする必要があります。
- Auto Recovery 機能は LifeKeeper の回復機能と競合する可能性があるので、併用は推奨しません。

  **注記:** 設定ファイル `/etc/default/LifeKeeper` のパラメーター PATH に AWS CLI 実行ファイ

ルのパスが設定されていない場合、PATH に AWS CLI 実行ファイルのパスを追加してください。

### AWS Identity and Access Management (IAM):

LifeKeeperがAWSを操作するために、以下のアクセス権限を持ったIAM ユーザもしくはIAM ロールが必要です。EC2 インスタンスの root ユーザーからアクセスできるようにEC2の IAM ロール を設定するか、AWS CLI の設定 を適切に行ってください。

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress
- ec2:DescribeRouteTables
- ec2:ReplaceRoute

## LifeKeeper ソフトウェア:

各サーバに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。具体的な LifeKeeper の要件については、SPS for Linux テクニカルドキュメンテーション およびSPS for Linux リリースノート を参照してください。

### LifeKeeper Recovery Kit for EC2:

各サーバに同じバージョンの Recovery Kit for EC2 ソフトウェアとパッチをインストールする必要があります。

### LifeKeeper IP Recovery Kit:

ルートテーブル (バックエンドクラスタ) を保護する目的で Recovery Kit for EC2 を使用する場合、各サーバに同じバージョンの LifeKeeper for Linux IP Recovery Kit ソフトウェアとパッチをインストールする必要があります。

**注記:** 最新リリースの互換性と発注情報については、 SPS for Linux リリースノート を参照するか、営業担当者に問い合わせてください。LifeKeeper Recovery Kit for EC2 のインストール / アンインストール方法の具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

Recovery Kit for EC2を使う場合には、Quorum/Witnessの使用を推奨します。
詳細については、こちら を参照してください。

# 6.3.3. 設定

必要とする保護と柔軟性が得られるように LifeKeeper を設定するには、設定要件を把握している必要があります。設定を適切にプランニングするには、Amazon、Amazon Virtual Private Cloud (VPC)、Amazon Elastic Compute Cloud (EC2)、およびユーザシステムの階層設定オプションを理解する必要があります。このセクションでは、設定のプランニングに加え、Recovery Kit を設定するために必要な特定の作業についても説明します。

## Amazon EC2 特有の設定上の考慮事項

Recovery Kit for EC2 を正しく設定するために、以下のトピックを見直して、設定作業を完了するために必要な情報を用意してあることを確認してください。

- ユーザシステムのセットアップ

更なる設定上の考慮事項については、以下のトピックを参照してください。

- EC2 のリソース監視と設定に関する考慮事項
- EC2 のローカルリカバリと設定に関する考慮事項

## Amazon EC2 特有の設定上の考慮事項

このセクションでは、EC2 リソースの以下の設定タスクについて説明します。設定タスクは EC2 リソースインスタンスに特有のものであり、Recovery Kit ごとに異なります。

- リソース階層の作成：アプリケーションリソース階層を LifeKeeper クラスタに作成します。
- リソース階層の削除：リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。
- リソース階層の拡張：リソース階層をプライマリサーバからバックアップサーバへ拡張します。
- リソース階層の拡張解除：リソース階層を LifeKeeper クラスタ内の 1 つのサーバから拡張解除 (削除) します。
- EC2 設定のプロパティの確認および編集: EC2 リソースの設定を表示し、その一部を修正することが可能です。
- Recovery Kit for EC2 のチューニング Recovery Kit for EC2 の動作を調整します。

以下の作業は、SPS for Linux テクニカルドキュメンテーション の管理 セクションに記載されています。これらは、すべての Recovery Kit で同じ手順を使用する共通の作業です。

- リソース依存関係の作成：既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。

- リソース依存関係の削除：リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service：リソース階層を特定のサーバで In Service にします。
- Out of Service：リソース階層を特定のサーバで Out of Service にします。
- プロパティの表示 / プロパティの編集：特定のサーバでリソース階層のプロパティを表示または編集します。

このセクションの残りの部分では、LifeKeeper GUI の [Edit] メニューから作業を選択することによって、Recovery Kit を設定する方法を説明します。設定作業はツールバーから選択することもできます。

- 状況表示ウィンドウのリソース階層ツリー (左側のペイン) のグローバルリソースを右クリックすると、[Edit] メニューと同じドロップダウンメニューの選択項目が表示されます。これは階層がすでに存在している場合にだけ可能な方法です。
- 状況表示ウィンドウのリソース階層表 (右側のペイン) のリソースインスタンスを右クリックすると、サーバおよび特定リソースの状況に応じて、リソース階層の作成を除くすべての設定作業を実行できます。

# 6.3.3.1. Recovery Kit for EC2 のチューニング

Recovery Kit for EC2 で設定可能なパラメータはパラメーター一覧 を参照してください。

# 6.3.3.2. リソース階層の作成

プライマリサーバからリソースインスタンスを作成するには、以下の手順を完了する必要があります。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択してください。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択してください。

2. ダイアログボックスが表示され、クラスタ内にインストール済みの認識されている Recovery Kit がすべて、ドロップダウンリストボックスに表示されます。ドロップダウンリストから **[Amazon EC2]** を選択し、 **[Next]** をクリックしてください。

3. 以下の情報を入力するようにプロンプトが表示されます。(ダイアログボックスで [Back] ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、以前に入力した情報を訂正する必要がある場合に特に役立ちます。)

   **注記:** 階層作成の途中で [Cancel] ボタンをクリックすると、作成処理全体が取り消されます。

| フィールド | ヒント |
|---|---|
| Switchback Type | フェイルオーバ後にサーバが復帰したときに、このサーバに EC2 インスタンスを戻す方法を指定します。[intelligent] または [automatic] を選択できます。<br><br>• [intelligent] の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。<br><br>• [automatic] の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。<br><br>**注記:** スイッチバックタイプは、必要な場合、[Resource Properties] ダイアログボックスの [General] タブで後から変更できます。 |
| Server | [Server] で EC2 リソースの作成先のサーバ (通常、これをプライマリサーバまたはテンプレートサーバと呼ぶ) を選択します。クラスタ内のすべてのサーバがドロップダウンリストに表示されます。 |
| EC2 Resource type | EC2 Recovery Kit は、ルートテーブルと Elastic IP という 2 通りの AWS リカバリシナリオに対する保護を提供します。<br><br>ルートテーブルシナリオはローカル仮想 IP アドレスと組み合わせて使用し、通常はバックエンドクラスタに使用します。<br><br>Elastic IP シナリオは Elastic IP の保護に使用し、通常はフロントエンドクラスタに使用します。<br><br>使用する EC2 タイプを選択してください。 |
| IP | このフィールドは、ルートテーブルシナリオでのみ表示され設定されます。IP リソー |

| | |
|---|---|
| resource | スを選択してください。これは LifeKeeper が保護する仮想 IP アドレスであり、VPC のルートテーブルアドレスで設定されます。 **注記**：リストには ISP および IPv4 ベースの IP リソースのみが表示されます。 |
| Network Interface | このフィールドは、Elastic IP シナリオでのみ表示され設定されます。Elastic IP に関連付けるネットワークインターフェースを選択してください。 |
| Elastic IP | このフィールドは、Elastic IP シナリオでのみ表示され設定されます。ネットワークインターフェースに関連付ける Elastic IP を選択してください。 |
| EC2 Resource Tag | 作成している EC2 リソースインスタンスに固有の EC2 リソースタグ名を選択または入力します。このフィールドには、デフォルトのタグ名である ec2-<resource> が自動的に表示されます。<resource> はリソース名です。このタグは、後から変更できます。 |

4. [Create] をクリックしてください。[Create Resource Wizard] によって、EC2 リソースが作成されます。

5. この時点で情報ボックスが表示され、LifeKeeper は、EC2 リソース階層を作成するために有効なデータが指定されたかどうかを検証します。LifeKeeper が問題を検知した場合は、情報ボックスにエラーが表示されます。検証が正常に完了すると、リソースが作成されます。 **[Next]** をクリックしてください。

EC2 リソース階層が正常に作成されたことを示す情報ボックスが表示されます。リソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにそのリソース階層を拡張する必要があります。

**[Continue]** をクリックすると、[Pre-Extend configuration task] が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、リソース階層の拡張 を参照してください。

ここで **[Cancel]** をクリックすると、別のダイアログボックスが表示され、後で EC2 リソース階層を別のサーバに手動で拡張して LifeKeeper の保護下に置く必要があることが警告されます。

# 6.3.3.3. リソース階層の削除

LifeKeeper 環境のすべてのサーバからリソース階層を削除するには、次の手順を実行してください。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択してください。ドロップダウンメニューから **[Delete Resource Hierarchy]** を選択してください。

2. EC2 リソース階層から削除するターゲットサーバの名前を選択し、 **[Next]** をクリックしてください。

   **注記:** このダイアログボックスは、いずれかのペインでリソースインスタンスを右クリックして [Delete Resource] を選択した場合には、表示されません。

3. [Hierarchy to Delete] を選択してください。削除するリソース階層を指定して強調表示にしてから、**[Next]** をクリックしてください。

   **注記:** このダイアログボックスは、左右どちらかの インでリソースインスタンスを右クリックして [Delete Resource] を選択した場合には、表示されません。

4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[Delete]** をクリックして次に進んでください。

5. EC2 リソースが正常に削除されたことを確認する情報ボックスが表示されます。

6. **[Done]** をクリックして終了してください。

# 6.3.3.4. リソース階層の拡張

階層の作成後、クラスタ内の別のサーバに拡張する必要があります。3 通りのシナリオで、テンプレートサーバからターゲットサーバにリソースインスタンスを拡張できます。

- リソースの作成後、[Continue] をクリックして、別のサーバにリソースを拡張します。
- 次に説明するように [Edit] メニューから [Extend Resource Hierarchy] を選択します。
- 左側または右側のペインから拡張されていない階層を右クリックします。

どのシナリオでも同じダイアログボックスが表示されます (いくつかの例外については、以下に詳細を明記)。

1. LifeKeeperGUI メニューから [Extend] ウィザードを開始する場合は、 **[Edit]** を選択し、次に **[Resource]** を選択します。ドロップダウンメニューから **[Extend Resource Hierarchy]** を選択します。これで [Extend Resource Hierarchy] ウィザードが起動されます。拡張操作に慣れていない場合は、 **[Next]** をクリックします。LifeKeeper の [Extend Resource Hierarchy] のデフォルト値が分かっていて、入力と確認を省略する場合は **[Accept Defaults]** をクリックします。

2. [Pre-Extend Wizard] で以下の情報を入力します。

**注記:** 最初の 2 つのフィールドは [Edit] メニューの [Extend] から操作を開始した場合にだけ表示されます。階層を拡張する手順の間に [Cancel] をクリックすると、どの時点であってもそのサーバへの拡張処理がキャンセルされるので注意してください。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

| フィールド | ヒント |
|---|---|
| スイッチバックタイプ | スイッチバックタイプを選択します。ここでは、バックアップサーバへのフェイルオーバの後、EC2 インスタンスが In Service に戻ったときに、このサーバに EC2 インスタンスをスイッチバックする方法を指定します。[intelligent] または [automatic] を選択できます。<br>• intelligent の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。<br>• automatic の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。<br>スイッチバックタイプは、必要な場合 [Resource Properties] ダイアログボックスの [General] タブで後から変更できます。 |
| Template Priority | テンプレートの優先順位を選択するか、入力してください。これはサーバで現在 In Service の EC2 階層の優先順位です。1 ～ 999 の範囲で、まだ優先順位として使用されていない値が有効で、小さい数値ほど優先順位が高くなります (数値 1 が最高の優先順 |

| | |
|---|---|
| | 位)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を使用することを推奨します。<br><br>**注記:** このフィールドは、階層を最初に拡張するときにだけ表示されます。 |
| Target Priority | ターゲットの優先順位を選択するか、入力してください。これは、他のサーバにある同等の階層に対する、新しく拡張する EC2 階層の優先順位です。1 ～ 999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位が高くなります (数値 1 が最高の優先順位)。<br><br>**注記:** LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられます。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |

3. 情報ボックスが表示され、LifeKeeper が環境のチェックを正常に完了し、この EC2 リソースを拡張するためのすべての要件が満たされていることが示されます。満たされていない要件がある場合は、 **[Next]** ボタンは選択できなくなり、 **[Back]** ボタンが有効になります。 **[Back]** をクリックした場合、情報ボックスに表示されるエラーメッセージの内容に従って、リソースの拡張を変更できます。ここで **[Cancel]** をクリックした場合は、EC2 リソース階層を他のサーバに手動で拡張して、LifeKeeper の保護下に置く必要があります。 **[Next]** をクリックすると、[Extend Resource Hierarchy] 設定作業に入ります。

4. リソース階層構造の拡張を行うために、以下の情報の入力が要求されます。

| フィールド | ヒント |
|---|---|
| EC2 Resource Tag | EC2 リソースタグを選択するか、入力してください。これは、ターゲットサーバに拡張される EC2 リソースが使用するリソースタグ名です。<br>**注記:** このフィールドは編集できません。 |

5. 拡張が実行中であることを示す情報ボックスが表示されます。同じ EC2 リソースインスタンスをクラスタ内の別のサーバに拡張する場合は、 **[Next Server]** をクリックしてください。その場合は、リソース階層を拡張する操作を繰り返します。 **[Finish]** をクリックすると、LifeKeeper は EC2 リソースの拡張が正常に完了したことを確認します。

6. **[Done]** をクリックして、 **[Extend Resources Hierarchy]** メニューの選択を終了します。

**注記:** 必ずすべてのサーバで新しいインスタンスの機能をテストしてください。

# 6.3.3.5. ローカルリカバリと設定に関する考慮事項

## ローカルリカバリシナリオ (バックエンドクラスタ):

保護されているルートテーブルの障害が Recovery Kit for EC2 によって検出されると、結果として生じる障害によって EC2 ローカルリカバリスクリプトが起動されます。ローカルリカバリでは、すべてのルートテーブル内から指定された IP リソースエントリを収集し、そのエントリのターゲットをアクティブなサーバ上にある ENI に変更します。ローカルでリカバリの試みが失敗すると、LifeKeeper は EC2 リソース、および依存関係を持つすべてのリソースをスタンバイサーバにフェイルオーバします。このシナリオの設定については、運用の原則 セクションを参照してください。

**注記:** 対応する EC2 リソースが作成されると、Recovery Kit がルートテーブルの設定を保護するので、ルートテーブルを手動で変更しないでください。

以下の例で、典型的なローカルリカバリのシナリオを示します。Recovery Kit が、ルートテーブル内にある IP ルートのターゲット設定が正しくないことを検出すると、ローカルリカバリにより、アクティブなサーバ上の ENI にターゲットが置換されます。このプロセスでは、ルートテーブル B の 10.1.0.20/32 のエントリは何も変更されません。

| | |
|---|---|
| IP リソース | 10.1.0.10 |
| アクティブなノード上の ENI | eni-01234567 |

### ルートテーブル A – 前の状態

| 接続先 | ターゲット |
|---|---|
| 10.1.0.10/32 | eni-89abcdef |
| 10.0.0.0/16 | ローカル |

### ルートテーブル A – 後の状態

| 接続先 | ターゲット |
|---|---|
| 10.1.0.10/32 | eni-01234567 |
| 10.0.0.0/16 | ローカル |

### ルートテーブル B – 前の状態

| 接続先 | ターゲット |
|---|---|
| 10.1.0.10/32 | eni-89abcdef |
| 10.1.0.20/32 | eni-89abcdef |
| 10.0.0.0/16 | ローカル |

### ルートテーブル B – 後の状態

| 接続先 | ターゲット |
|---|---|
| 10.1.0.10/32 | eni-01234567 |
| 10.1.0.20/32 | eni-89abcdef |
| 10.0.0.0/16 | ローカル |

## Elastic IP シナリオ (フロントエンドクラスタ):

保護されている Elastic IP の障害が Recovery Kit for EC2 によって検出されると、結果として生じる障害によって EC2 ローカルリカバリスクリプトが起動されます。ローカルリカバリにより、Elastic IP がアクティブなノード上の ENI に割り当てられます。ローカルでリカバリの試みが失敗すると、LifeKeeper は EC2 リソース、および依存関係を持つすべてのリソースをスタンバイサーバにフェイルオーバします。このシナリオの設定については、運用の原則 セクションを参照してください。

# 6.3.3.6. リソース監視と設定に関する考慮事項

## ルートテーブルシナリオ (バックエンドクラスタ)

Recovery Kit は AWS CLI を使用して、保護されている IP リソースに VPC 内のクライアントから接続できるようにルートテーブルの設定を監視します。Recovery Kit は、VPC 内のすべてのルートテーブルに対して、IP リソース宛のターゲットがアクティブサーバの ENI に正しく設定されていることを確認します。正しいことが確認されない場合は、Recovery Kit が EC2 のローカルリカバリプロセスを実行します。

## Elastic IP シナリオ (フロントエンドクラスタ):

Recovery Kit は AWS CLI を使用して、Elastic IP と アクティブなサーバ上にある ENI との関連付けを監視します。Recovery Kit は、アクティブサーバに接続されている ENI に Elastic IP が正しく関連付けられていることを確認します。正しいことが確認されない場合は、Recovery Kit が EC2 のローカルリカバリプロセスを実行します。

**注記:** どちらのシナリオにおいても、AWS CLI でタイムアウトが発生した場合、フェイルオーバは実行されず、リソースは ISP 状態のままになります。タイムアウト関連のメッセージのみが LifeKeeper ログに記録されます。Recovery Kit はチェック間隔の経過後に監視を再度実行します。タイムアウト値を設定する方法の詳細については、EC2パラメータ一覧 を参照してください。

# 6.3.3.7. リソース階層の拡張解除

階層全体を拡張解除するには、以下の手順を実行してください。

1. **LifeKeeper GUI メニュー** から **[Edit]** を選択し、次に **[Resouce]** を選択してください。ドロップダウンメニューから **[Unextend Resource Hierarchy]** を選択してください。

2. EC2 リソースから、拡張解除の対象となるターゲットサーバを選択してください。EC2 リソースが現在 In Service になっているサーバは選択できません。 **[Next]** をクリックしてください。

   **注記:** 右側のペインから個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

3. 拡張解除する EC2 階層を選択してください。 **[Next]** をクリックしてください。

   **注記:** 左側のペインにあるグローバルリソースを右クリックするか、右側のペインにある個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

4. 拡張解除の対象として選択したターゲットサーバと EC2 リソース階層を確認する情報ボックスが表示されます。 **[Unextend]** をクリックしてください。

5. EC2 リソースが正常に拡張解除されたことを示す情報ボックスが表示されます。

6. **[Done]** をクリックして終了してください。

# 6.3.3.8. ユーザシステムのセットアップ

## ルートテーブルシナリオ (バックエンドクラスタ)

Recovery Kit for EC2 のルートテーブル保護オプションを使用すると、VPC 内のルートを自動更新することができます。フェイルオーバ中、Recovery Kit はターゲットサーバの仮想 IP アドレスが表す新しい Elastic Network Interface (ENI) の場所を反映するように、ルートテーブルを更新します。LifeKeeper が VPC 内のルートテーブルの保護、監視、および更新を実行できるようにするには、以下の設定手順を実行する必要があります。アクティブ / アクティブ設定の場合も同様です。

- LifeKeeper for Linux IP Recovery Kit で保護する仮想 IP アドレスは、VPC 内の割り当て済み CIDR の範囲外にする必要があります。
- 仮想 IP アドレスを LifeKeeper で保護してから、Recovery Kit for EC2 リソースを作成する必要があります。
- ENI のソース / ターゲットのチェックを無効にする必要があります。これは、インスタンスが仮想 IP アドレスのネットワークパケットを受信するために必要です。
- LifeKeeper IP リソースのブロードキャスト PING チェックを無効にする必要があります。LifeKeeper は、ローカルサブネット上の IP アドレスに対してブロードキャスト PING テストを実行することで、IP リソースを監視します。複数の利用可能ゾーンが存在する環境では、複数の利用可能ゾーンに異なるサブネットが存在するので、この機能は使用できません。この機能を無効にするには、 */etc/default/LifeKeeper* の設定ファイルの NOBCASTPING エントリを以下のように設定する必要があります。

  NOBCASTPING=1

- ルートテーブルには、仮想 IP アドレスおよびアクティブなサーバの ENI のルートエントリが必要です。>

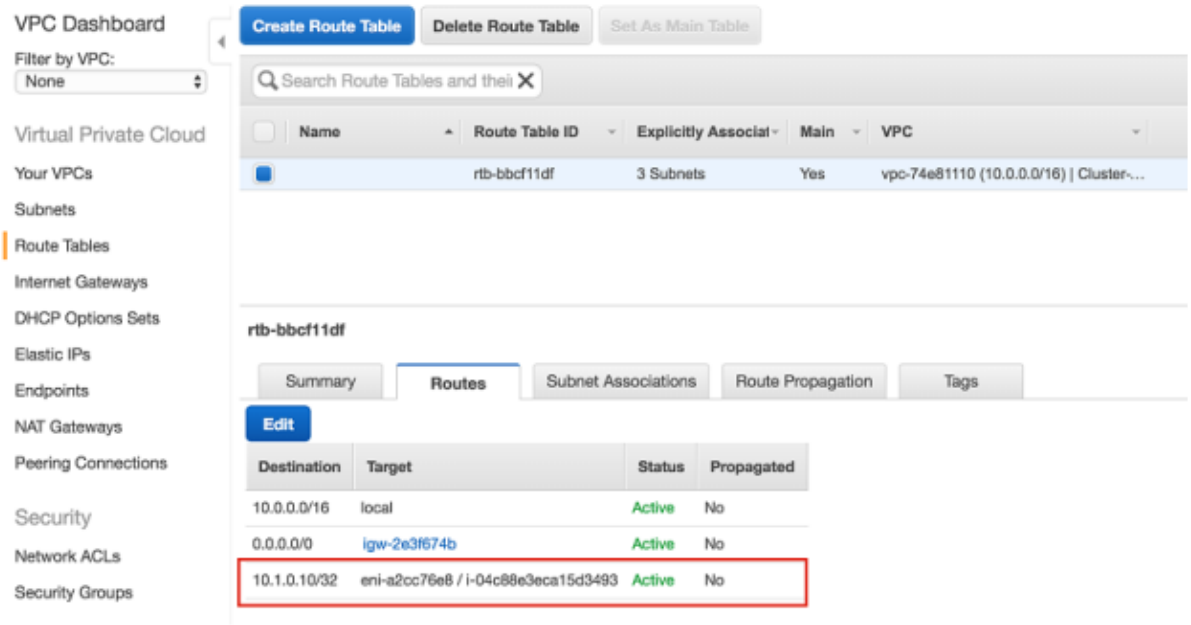
  **注記:**

  対応する EC2 リソースが作成されると、Recovery Kit がルートテーブルの設定を保護するので、ルートテーブルを手動で変更しないでください。

  **例**

  送信先: VIP 10.1.0.10/32

  ターゲット: eni-a2cc76e8




## Elastic IP シナリオ (フロントエンドクラスタ)

Recovery Kit for EC2 の Elastic IP (EIP) 保護オプションを使用すると、特定の ENI (アクティブサーバまたはバックアップサーバ上の EC2 が使用する ENI) と EIP の関連付けを再実行できます。

アクティブサーバまたはバックアップサーバ上の ENI と EIP との関連付けについて、LifeKeeper が保護、監視、および更新を実行できるようにするには、以下の設定手順を実行する必要があります。

- 1 つの ENI は、1 つの Elastic IP とのみ関連付けることができます。その他の EIP (EC2 リソースにより使用される EIP を除く、すべての EIP) は、その ENI と関連付けることができません。他の EIP を関連付けると、Recovery Kit はその ENI にすでに関連付けていた他の EIP の関連付けを解除します。

注記:

- AWS の Elastic Block Store (EBS) は 1 つの EC2 インスタンスにのみ接続可能なため、EBS を使用する HA クラスタ構成を作成する際は、DataKeeper for Linux を使用することを推奨します。
- /etc/default/LifeKeeper の RESRVRECTIMEOUT の値を、デフォルト値の 150 から 300 に増加させることを推奨します。RESRVRECTIMEOUT は、別のプロセスがすでにリソースをリザーブしているときに、リソースを「リカバリ」用にリザーブするまで、LifeKeeper プロセスが待機する間スリープ状態になっている秒数です。

# 6.3.4. トラブルシューティング

メッセージカタログ は、SIOS Protection Suite for Linux の使用時に検出されるすべてのメッセージのリストを示すとともに、適宜、エラーの原因とエラー状態を解決するために必要な処置の説明を提供します。受け取ったエラーコードをこのリストから探すことができます。また、Recovery Kit for EC2 の使用時に検出されるすべてのメッセージがリストされた Recovery Kit for EC2 メッセージカタログ に直接アクセスすることもできます。

# 6.4. Generic ARK for Load Balancer probe reply

## 概要

Generic ARK for Load Balancer probe reply（以下Gen LB）は、Microsoft Azure（以下Azure）や Google Cloud Platform（以下GCP）環境で Load Balancer の負荷分散対象インスタンスへのヘルスチェックプローブを待ち受けて応答する仕組みを提供します。

本書では Gen LB スクリプトの動作詳細を解説します。LifeKeeper の一般的な用語、使用方法等については、 SIOS Protection Suite for Linux テクニカルドキュメンテーション を参照してください。

また、本書の内容はあらゆる環境において正常に動作することを保証するものでありません。

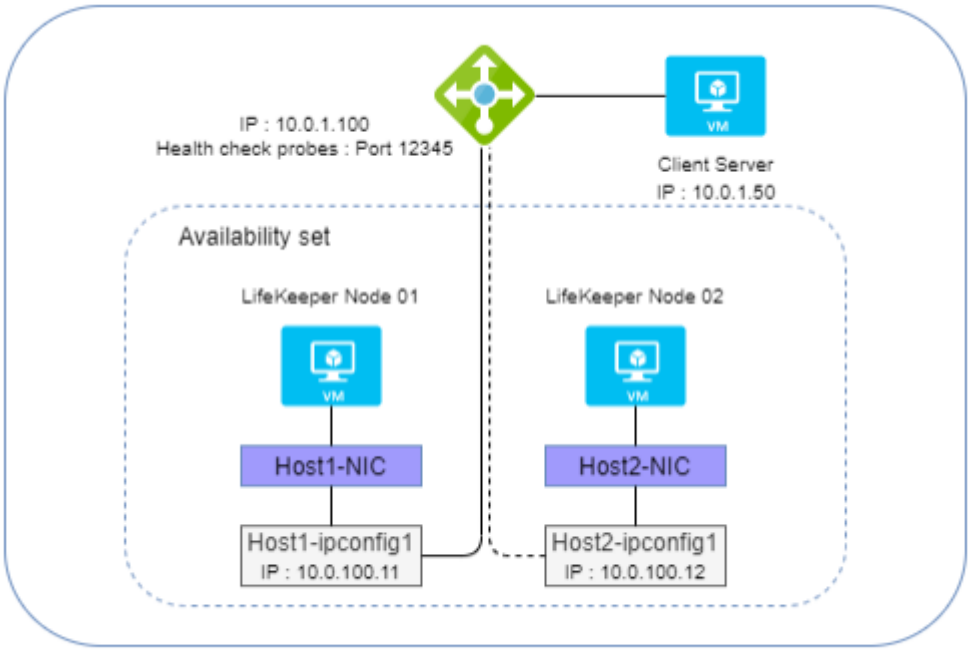
## 前提条件

- 本スクリプトはLifeKeeper for Linux v9.5.1以降で利用することができます。その他のバージョンやSingle Server Protectionでは利用できません。
- LifeKeeper for Linux v9.5.1 以降でサポートする OS･構成 をサポートしています。ただし LifeKeeper for Linux v9.5.1 + SAP Application Recovery Kit（SAP ARK）の組み合わせを GCP 上で利用する構成はサポート対象外です。サポートされる OS や構成につきましては、オンラインドキュメントの リリースノート および SIOS Protection Suite for Linux サポートマトリックス をご確認ください。またインストール手順につきましては、オンラインドキュメントのインストール手順をご確認ください。
- Cloud は Microsoft Azure と Google Cloud Platform のみサポートします。
- プローブのプロトコルは TCP のみをサポートします。

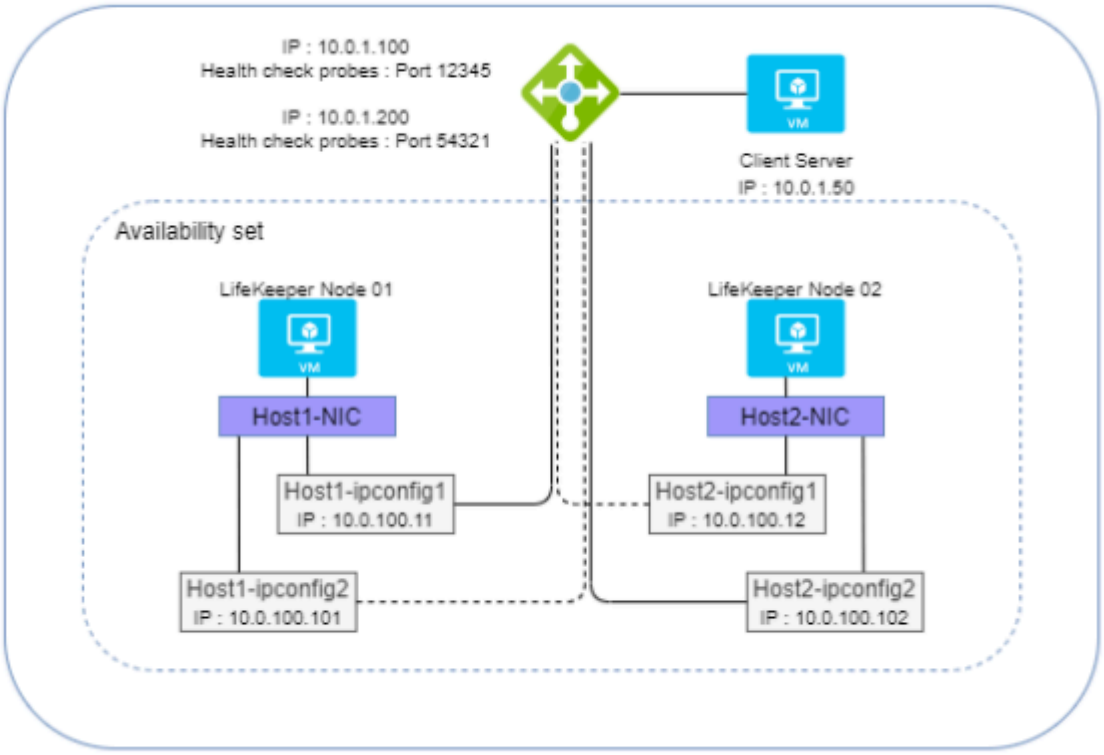
# 6.4.1. 環境構成例

## Azureを使用する場合の構成例

### Active / Standby構成

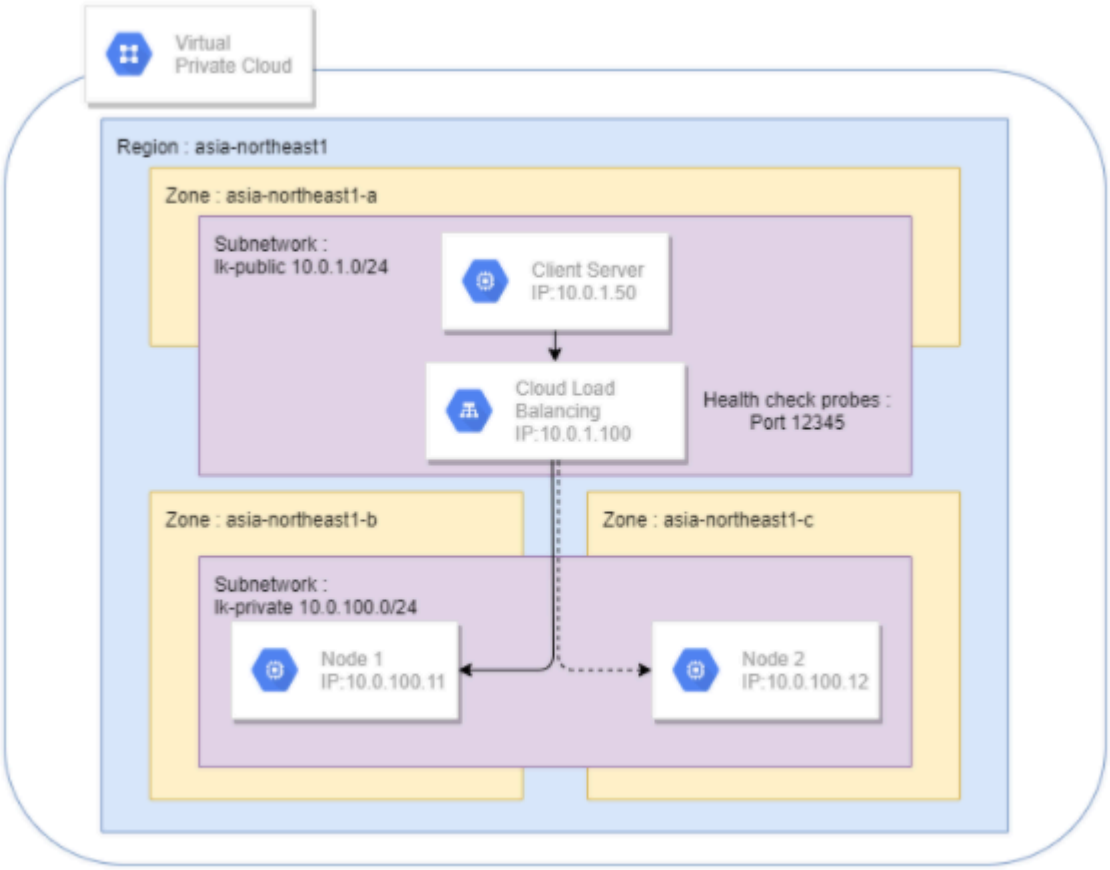



### Active / Active構成

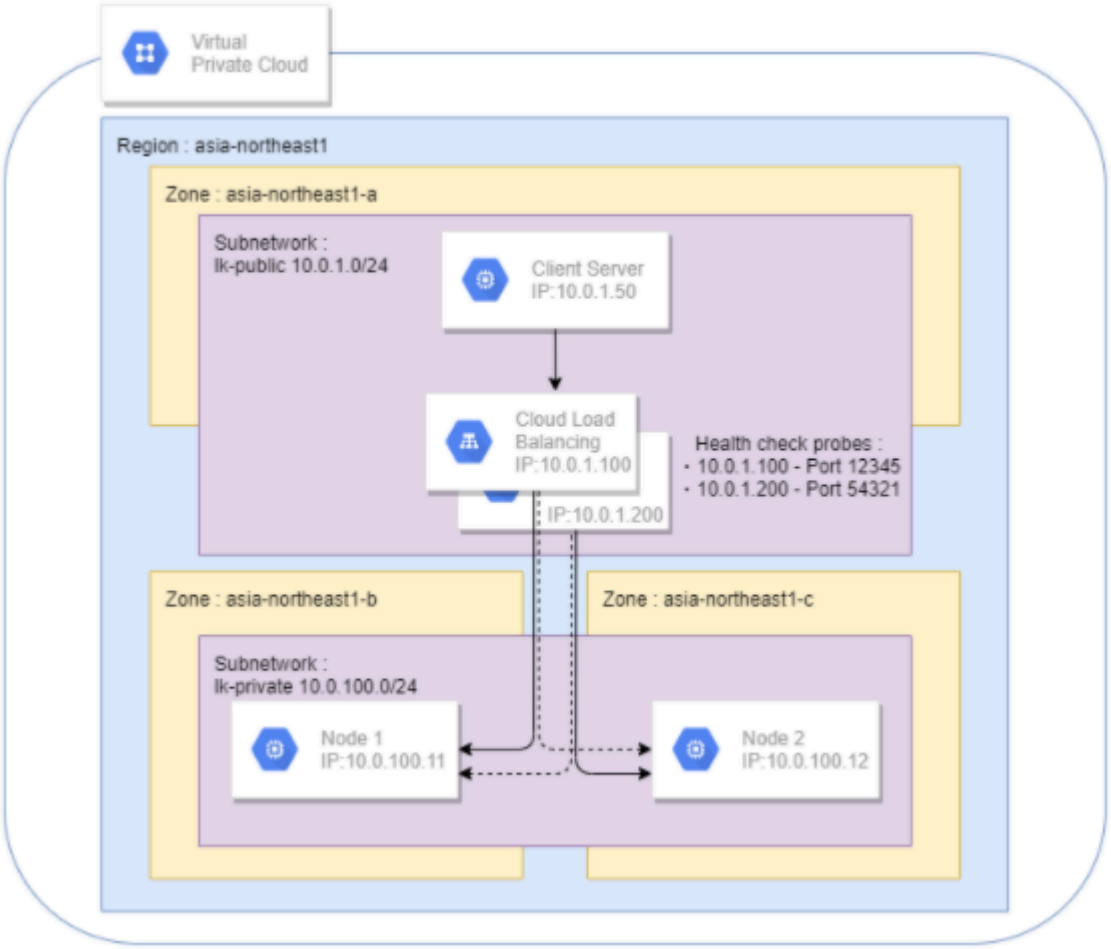


# GCPを使用する場合の構成例

## Active / Standby構成



## Active / Active構成



# 6.4.2. 基本動作

## 起動処理(restore)

Load Balancer probe reply リソースを ISP（サービス稼動状態）にするために、指定された Port 番号でヘルスチェックプローブを待ち受けるデーモンを起動します。この動作は LifeKeeper の GUI クライアントまたは CUI で Load Balancer probe reply リソースの起動処理を実行したときが該当します。起動処理に失敗した場合は、Load Balancer probe reply リソースは OSF（サービス障害状態）になります。スイッチオーバーやフェイルオーバーの切り替えにおいても同様の処理を行います。この動作の詳細は スクリプト仕様 をご参照下さい。

## 停止処理(remove)

Load Balancer probe reply リソースを OSU（サービス停止状態）にするために、デーモンを停止します。この動作は LifeKeeper の GUI クライアントまたは CUI で Load Balancer probe reply リソースの停止処理を実行したときが該当します。この動作の詳細は スクリプト仕様 をご参照下さい。

## 監視処理(quickCheck)

Load Balancer probe reply リソースが ISP のときに、デーモンが正しく起動されていることを確認します。監視処理で正しいことが確認されない場合はフェイルオーバーが実行されます。 この動作の詳細は スクリプト仕様 をご参照下さい。

## 回復処理(recover)

Recover 処理はありません。

# 6.4.3. スクリプト仕様

Generic リソース作成時に、本スクリプトを restore、remove、quickCheck に登録頂くことでそれぞれの動作で起動します。具体的には実際のスクリプトを参照して下さい。

説明の各所で実行コマンドラインや関連ファイル名 / パスを例示します。
具体的には実際のスクリプトを参照して下さい。

## 起動処理(restore)

以下の順序で Load Balancer probe reply リソースの起動を試みます。

1. コマンドラインオプションから自リソースのタグ名を取得します。
2. 自リソースのタグ名からリソース情報（AppInfo : < Port番号 > [応答メッセージ]）を取得します。失敗した際は、exit 1で処理を異常終了します。
3. タグ ID からデーモンが起動されていないことを確認します。既に起動されている場合は、exit 1 で処理を異常終了します。
4. Socket オブジェクトを生成し、Load Balancer からの接続を待ち受けます。 $HC_TIMEOUT ( スクリプトのパラメーター一覧 に記載) 秒待ち、接続がない場合は、exit 1 で処理を異常終了します。
5. 4 をデーモンとしてバックグラウンドで実行します。$HC_TIMEOUT ( スクリプトのパラメーター一覧 に記載) 秒待ち、接続がない場合は、$id-hc-ng フラグが生成されます。再度接続を確認したらフラグは削除され、タイムアウトもリセットされます。 **注記:** $id-hc-ng フラグは、quickCheck での異常検知に使用されます。
6. exit 0 で処理を正常終了します。

## 停止処理(remove)

以下の順序で Load Balancer probe reply リソースの停止を試みます。

1. コマンドラインオプションから自リソースのタグ名を取得します。
2. 自リソースの ID から、デーモンの PID を取得します。
3. PID からデーモンが起動されていることを確認します。既に停止している場合は exit 0 で処理を正常終了します。
4. TERM シグナルでプロセスを停止します。
5. exit 0 で処理を正常終了します。

# 監視処理(quickCheck)

以下の順序で Load Balancer probe reply リソースの監視を試みます。

1. コマンドラインオプションから自リソースのタグ名を取得します。
2. 自リソースのタグ名から、リソースの起動状態を取得します。ISP 以外の場合は、既に停止していると判断し、exit 0 で正常終了します。
3. 起動処理(restore)の⑤で生成される、$id-hc-ng フラグの有無を確認します。存在する場合は、Load Balancer からの接続が $HC_TIMEOUT 秒 ないことを意味し、exit 1 で処理を異常終了します。
4. exit 0 で処理を正常終了します

## スクリプトの実行時間

### restore

restore 実行時に、Load Balancer から $HC_TIMEOUT 秒 接続がないと処理がタイムアウトします。

### remove, quickCheck

これらのスクリプトの処理に要する時間に制限はありません。

# 6.4.4. スクリプトのパラメーター一覧

各スクリプト内で設定可能なパラメーター一覧を下記に記載します。

| パラメーター | 説明 |
| --- | --- |
| HC_TIMEOUT | Load Balancer からの接続要求のタイムアウト値を設定します。<br><br>指定された秒数 Load Balancer から接続がない場合は、異常と判断されフェイルオーバーが実施されます。初期値は 60 です。 */etc/default/LifeKeeper* ファイルに設定することで有効になります。<br><br>**設定例:**<br>HC_TIMEOUT=30 |

上記以外のパラメーターおよびコードの変更はできません。可能なパラメーター以外の変更を行った場合、LifeKeeper 製品サポートの対象外となります。予めご注意ください。

# 6.4.5. リソースの作成/拡張

Gen LB リソースの作成/拡張方法は、Generic Application を利用します。Generic Application の利用方法の詳細につきましては、 SIOS Protection Suite for Linux テクニカルドキュメンテーション を参照してください。

Gen LB リソース作成時の独自の設定として、Generic Application wizard の Application Information（AppInfo）の項目での Port 番号と応答文字列の指定があります。Port 番号（1024-65535の範囲）の設定は必須の設定項目です。応答文字列については任意で設定することができます。

AppInfo : <Port 番号:1024-65535> [応答文字列]
入力例.　AppInfo : 12345 “message”

Port番号と応答文字列のあいだは半角スペースです。

応答文字列に半角スペースを含むことはできません。

# 6.4.6. メッセージ一覧

スクリプトで出力するメッセージの一覧です。各メッセージは、イベントログに出力されます。
DEBUG は debug フラグ、もしくは debug_gen-lb フラグを設定することで出力されるようになります。

| コード | カテゴリー | メッセージ | 説明 |
|---|---|---|---|
| 127001 | ERROR | Daemon already started. | 既にデーモンが起動しているため、restore に失敗しています。 |
| 127002 | ERROR | Socket open failed. | 指定した Port が既に使用されているなど、何らかの理由で Socket の Open に失敗しています。 |
| 127003 | ERROR | Health probe not received. | restore 時に、Load Balancer からのプローブが途絶していることを示します。 |
| 127004 | DEBUG | Health check received. | restore 時に、Load Balancer からのプローブを受信したことを示します。 |
| 127005 | ERROR | Daemon stat failed. | デーモンの起動に失敗しています。 |
| 127006 | ERROR | Health probe not received. | デーモンによるプローブ待ち受け時に、Load Balancer からのプローブが途絶していることを示します。 |
| 127007 | DEBUG | Health check received. | デーモンによるプローブ待ち受け時に、Load Balancer からのプローブを受信したことを示します。 |
| 127008 | INFO | Daemon already stopped. | remove 時に、デーモンが起動していないことを示します。 |
| 127009 | ERROR | Resource out of service. | quickCheck 時にリソースが In Service（ISP）でないことを示します。 |
| 127010 | ERROR | Daemon not running. | quickCheck 時にデーモンが起動してないことを示します。 |
| 127011 | ERROR | Failed to accept the health probe:$! | 何らかの理由で、プローブの応答に失敗しています。 |
| 127012 | ERROR | Port number is not valid. | 指定された Port が有効でないため、restore に失敗しています。 |

# 6.5. LVM Recovery Kit 管理ガイド

## LVM Recovery Kit テクニカルドキュメンテーション

The SIOS Protection Suite (SPS) for Linux Logical Volume Manager (LVM) Recovery Kit は、他の SPS リカバリキットに、論理ボリュームサポートを提供します。SPS で保護されたアプリケーションは、ストレージ管理の簡素化、要件変更に応じた動的なボリュームのサイズ変更など、Logical Volume Manager が提供するメリットを活用できます。

LVM Recovery Kit は、単体で使用されることがなく、他の SPS リソースに依存して使われる点で、他のほとんどの SPS リカバリキットと異なります。そのため、SPS リカバリキットに通常付随する多くの作業 (階層の作成など) は、LVM Recovery Kit に直接該当しません。

## ドキュメントの内容

このガイドでは次の事項について説明します。

- ドキュメンテーションとリファレンス 。関連する SPS for Linux のドキュメントと、その入手場所、LVM 製品に関する多数の役立つドキュメントの参照情報の一覧を記載します。
- 要件 。LVM Recovery Kit を適切に設定、インストール、運用する上で必要なハードウェアとソフトウェアを説明します 。SPS for Linux ソフトウェアをインストールまたはアンインストールする方法の詳細については、SPS for Linux インストールガイドを参照してください。
- 概要 。LVM Recovery Kit と対応するリソースタイプの概要を説明します。
- SPS LVM階層の作成と管理 。SPS による LVM Recovery Kit の管理作業の詳細を説明します。
- トラブルシューティング 。情報メッセージとエラーメッセージ、および推奨される対処方法の一覧を示します。

# 6.5.1. ドキュメンテーションとリファレンス

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- SIOS Protection Suite インストールガイド
- Optional Recovery Kit ドキュメンテーション

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

LVM に関する情報は以下を参照してください。

- 論理ボリュームマネージャーの管理 Red Hat Enterprise Linux 7
- 論理ボリュームの設定および管理 Red Hat Enterprise Linux 8
- 論理ボリューム(LVM) | ストレージ管理ガイド | SUSE Linux Enterprise Server 12 SP5
- 論理ボリューム(LVM) | ストレージ管理ガイド | SUSE Linux Enterprise Server 15 SP2
- LVM HowTo (最新情報ではありません)

# 6.5.2. 要件

SPS for Linux LVM Recovery Kit をインストールする前に、SPS の構成が次の要件を満たしている必要があります。SPS for Linux ハードウェアおよびソフトウェアの設定に関する具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

# 6.5.2.1. ハードウェア要件とソフトウェア要件

## ハードウェア要件

- **サーバ** 。このリカバリキットを使用するには、SPS for Linux リリースノート および SPS for Linux インストールガイド (製品メディアに収録) に記載の要件に従って設定されたコンピュータが 2 台以上必要です。

- **データストレージ** 。LVM Recovery Kit は、共有ストレージと DataKeeper for Linux によって提供される複製されたストレージの両方と組み合わせて使用できます。Network Attached Storage (NAS) と合わせて使用することはできません。その他、論理ボリュームデバイス上のアプリケーションを保護しているリカバリキットの要件を超えるような、ストレージ設定に関する特別な要件はありません。

## ソフトウェア要件

- **オペレーティングシステム** 。LVM は、すべての主要 Linux ディストリビューションに含まれています。サポートされているディストリビューションとバージョンについては、SPS for Linux リリースノート を参照してください。

- **Logical Volume Manager** 。リカバリキットをインストールするには、lvmまたはlvm2 rpm パッケージがインストールされている必要があります。このリリースの SPS Logical Volume Manager Recovery Kit は、LVM バージョン1 および LVM バージョン2 (LVM2) の両方をサポートします。サポートされる LVM のバージョンは、Linux ディストリビューションで提供されるバージョンです。

- **SPS Software** 。各サーバに同一のバージョンの SPS Core ソフトウェア、LVM Recovery Kit を含む任意のリカバリキット、およびパッチをインストールする必要があります。具体的な SPS の要件については、SPS for Linux リリースノート を参照してください。

- **SPS for Linux Logical Volume Manager Recovery Kit** 。Logical Volume Manager Recovery Kit は SPS Installation Image File (sps.img)を提供します。これは、Red Hat Package Manager (rpm: `steeleye-lkLVM`) を使用してパッケージ、インストール、およびアンインストールします。

パッケージのインストール中、サポート対象バージョンの SPS Core パッケージと LVM パッケージの両方が、LVM Recovery Kit をインストールするシステム上にあることの確認が行われます。これらのパッケージに要求されるバージョンについては、SPS for Linux リリースノート を参照してください。

SPS Core ソフトウェアおよび LVM Recovery Kit をインストールまたは削除する手順については、SPS for Linux インストールガイド を参照してください。

LVM Recovery Kit は、LVM を使用した SPSで保護するディスクリソースを管理するクラスタ内の各サーバ上にインストールする必要があります。

LVM Recovery Kit は、LVM ボリューム上に配置されるアプリケーションの階層を作成および拡張する前にインストールする必要があります。

# 6.5.3. 概要

## LVM の操作

LVM は現在すべての主要 Linux ディストリビューションに含まれる標準的なボリューム管理製品です。LVM は、複数の物理ディスクまたはディスクパーティションをボリュームグループとして知られるエンティティにひとまとめにグループ化することができます。ボリュームグループはさらに分割されまたは仕切られて複数の論理ボリュームになります。論理ボリュームは、標準のブロックデバイスとしてアクセスできます。そのため、ブロックデバイスを直接扱うことのできるファイルシステムや任意のアプリケーションによって使用できます。

Logical Volume Manager は、主にストレージ管理の簡素化のために使用されます。論理ボリュームをストレージ要件の変化に応じて、動的にサイズ変更することができ、物理ディスクやパーティションが sda や sdc1 などと命名されるのとは違って、ボリュームグループと論理ボリュームは管理者が選んだIDによってわかりやすく命名されます。

次の図は、LVM エンティティ間の関係を示しています。ファイルシステムまたはアプリケーションは、論理ボリュームを使用します。論理ボリュームは、ボリュームグループをパーティションで仕切ることによって作成されます。ボリュームグループは、1 つ以上の物理ディスクパーティションまたはディスクの集合です。

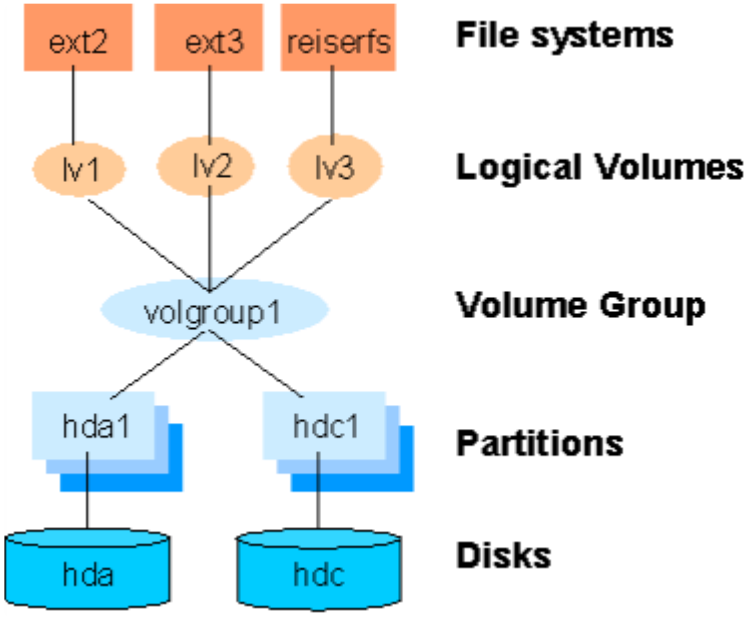



**図 1: Logical Volume Manager のエンティティ間の関係**

## SPS for Linux LVM Recovery Kit

SPS LVM Recovery Kit は、その他の SPS リカバリキットが Linux 論理ボリューム上で正常に連携するために必要なサポートを提供します。このサポートを実現するために、LVM Recovery Kit は 2 つの新しいリソースタイプをインストールします。lvmlv と lvmvgで、それぞれ論理ボリュームとボリュームグルー

プに連携します。 lvmlv および lvmvg リソースは、他の SPS リソースを動作できるようにするという、内部的な用途のためのみに存在しています。

図 1 に示すように、各ボリュームグループには1つ以上の論理ボリュームからの依存関係があります。逆に言えば、各論理ボリュームは依存関係のあるボリュームグループを1つは持っていなければいけません。これら 2つのLVM リソースを含む一般的な SPS 階層は、図 1　に示す関係に非常に近いようです。実際の SPS 階層の例を見るには、SPS LVM 階層の作成と管理 セクションにある図 2 を参照してください。

LVM Recovery Kit では、lvm パッケージで提供されるコマンドを使用して、SPS 内のボリュームグループと論理ボリュームリソースを管理します。ボリュームグループと論理ボリュームは、フェイルオーバやスイッチオーバの操作中に階層が in service になると設定あるいは起動し、階層が out of service になると設定解除となります。

# 6.5.3.1. LVM Recovery Kit の注意事項と制約事項

以下の注意事項と制約事項は、本バージョンの LVM Recovery Kit に適用されます。

## Raw I/O および全ディスクのサポート

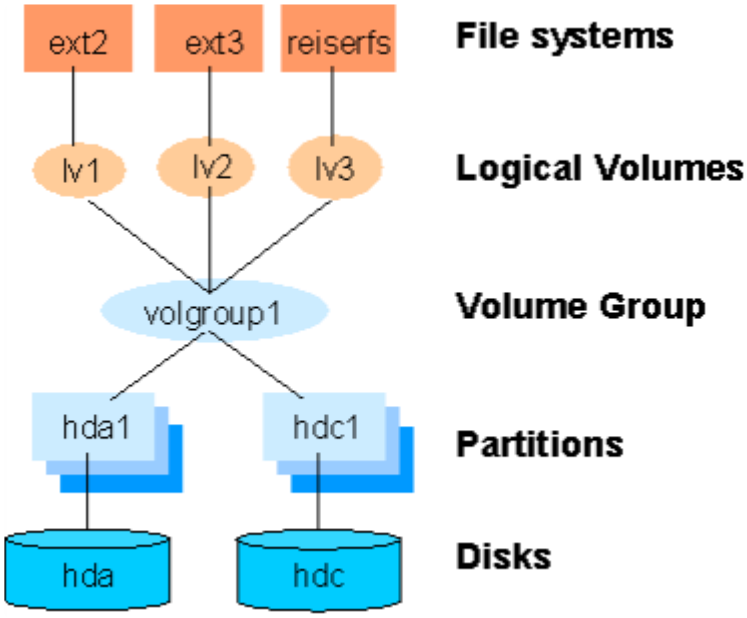
図 1 にはディスクパーティション上の様々なファイルシステムの下にある論理ボリュームとボリュームグループが示されていますが、LVM Recovery Kit を SPS Raw I/O Recovery Kit と組み合わせて使用した場合、論理ボリュームへの raw アクセスをサポート可能であることは重要です。このとき、複数のディスクパーティション (例: */dev/sdc1*) ではなく、1 つ以上のディスク全体 (例: */dev/sdc*) で構成されるボリュームグループを管理できることに注目してください。

LVM の使用上のオプションについて詳細を知るには、LVMをDataKeeperで使用する セクションも参照してください。

## ボリュームグループのアクティベーション

現在の LVM 実装においては、ボリュームグループがアクティブになると、同時にそのボリュームグループに関連するすべての論理ボリュームが自動的にアクティブになります。SPS にとってこれは、関連するリソースインスタンスがまだ Out-of-Service (OSU) にマーキングされているにも関わらず論理ボリュームがアクティブになることがあるということです。一般的なフェイルオーバやスイッチオーバ操作では、SPS は論理ボリュームをボリュームグループの後即座に in service にしようとします。その結果として restore スクリプトの呼び出しが成功の表示とともに即座に返ってきます。論理ボリュームを in service にしようとするこの不要な試みは使いやすさに何の影響も与えません。

# LVM Figure 1


ext2
ext3
reiserfs
File systems
lv1
lv2
lv3
Logical Volumes
volgroup1
Volume Group
hda1
hdc1
Partitions
hda
hdc
Disks

# 6.5.4. SPS LVM 階層の作成と管理

SPS LVM 階層は、論理ボリューム上に配置されるリソースの作成処理中に自動的に作成されます。LVM リソースタイプを含む階層の作成と拡張は、常に、より高位のリソースタイプの作成および拡張プロセスから開始されます。削除と拡張解除についても同様です。

以下の図は、LVM リソースを含む階層全体を示す LifeKeeper GUI のスクリーンショットです。階層内のリソースはタグによるデフォルト表示というより、SPS ID を明確にするために表示されていることに注意してください。

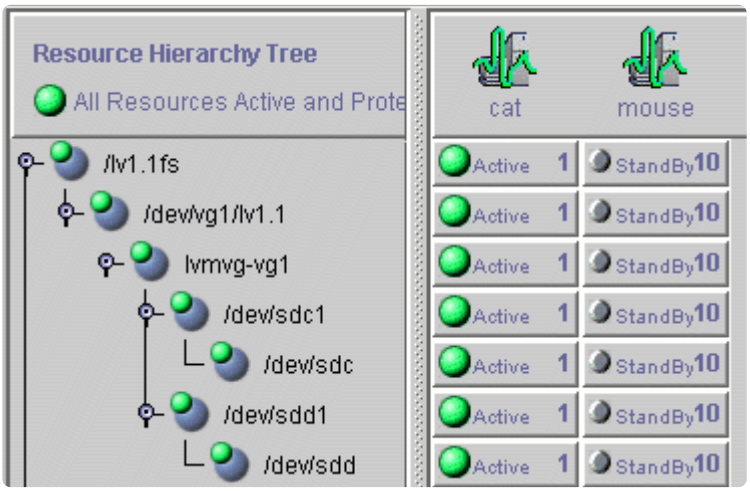



**図 2: LVM リソースを含む SPS 階層**

図 2 に示される階層はファイルシステム階層です。これは、[Edit] > **[Server] > [Create Resource Hierarchy]** メニューから [File System] リカバリキットを選択して作成されます。これは、LVM 論理ボリューム */dev/vg1/lv1.1* にマウントされたファイルシステムリソース `/lv1.1fs`で構成されています。その論理ボリュームは、SPS ID `lvmvg-vg1` で表される vg1 ボリュームグループの一部です。ボリュームグループ `vg1` は、*/dev/sdc1* および */dev/sdd1* という2つの物理ディスクパーティションで構成されています。階層内の、各ディスクパーティション配下には、さらにディスクデバイス、*/dev/sdc* および */dev/sdd* があります。

# 6.5.4.1. 階層作成の手順

ファイルシステムまたは高レベルアプリケーションが LVM 論理ボリュームを使用する階層を作成するには、次の高レベル手順に従う必要があります。

1. LVM ボリュームグループと論理ボリュームに必要な設定を決定してください。このとき、以下の点を考慮してください。

- ◦ そのボリュームグループに関連するすべてのディスクリソースを、SPS クラスタ内のサーバ間で一緒に移動させる必要があります。

- ◦ そのボリュームグループに関連するすべての論理ボリューム (およびファイルシステムまたはそれを使うアプリケーション) は、SPS クラスタ内のサーバ間で一緒に移動しなければなりません。

- ◦ LVM RKのインストールの時にlvm2-lvmetadは無効になります。ただし、LVM RKのインストール後にlvm2をインストールした場合は自分で無効にする設定が必要です。設定方法はOSのマニュアルを参照してください。

2. アプリケーションのプライマリサーバになるシステム上で、必要なボリュームグループと論理ボリュームを作成しアクティブにします。作成には LVM パッケージに含まれるツールを使いますが、その説明はドキュメンテーションとリファレンス トピックの参照先にある LVM *HowTo* ドキュメントで説明しています。

   共有ストレージを使用してる場合は、ボリュームグループに割り当てられたすべての物理ボリュームが、保護されたアプリケーションを実行しようとしている SPS クラスタ内のマシン間で適切に共有されていることを確認してください。DataKeeper で LVM を使用したい場合は、DataKeeper で LVM を使用する のトピックを参照してください。

3. 各論理ボリューム上でファイルシステムを作成します。そうせずに、Raw I/O を使用したい場合は、Raw デバイスをそれぞれの論理ボリュームデバイスにバインドします。

4. 使用するアプリケーションに関連する SPS リカバリキット管理ガイドに記載されている設定手順に従い、保護されたアプリケーションをファイルシステム上で設定します。

   適切なリカバリキット管理ガイドに記載の手順に従って、アプリケーション階層を作成および拡張します。

> **重要:** アプリケーション階層を階層が拡張された各クラスタノードに一時的に移動するには、手動で in-service 操作を行います。この手順はノードのフェイルオーバ操作より前に行わなければなりません。各クラスタノード上の LVM サブシステムに新しいボリュームグループと論理ボリュームの設定について教えるためです。これらの手動によるスイッチオーバを行ってから、アプリケーション階層を必要なプライマリクラスタノードに戻します。

# 6.5.4.2. DataKeeper で LVM を使用する

SPS for Linux は現在 LVM 上での DataKeeper の使用と DataKeeper 上での LVM の使用の両方をサポートしています。標準的な DataKeeper の設定では LVM 上での DataKeeper の使用がサポートされているので、SPS LVM Recovery Kit をインストールしないでください。必要なリカバリキットは、DataKeeper だけです。ただし、DataKeeper 設定上で LVM を使用する場合は、LVM Recovery Kit が必要になります。

SIOS では LVM 上での DataKeeper 使用を推奨していますが、DataKeeper 設定上で LVM が使用される場合は、2 段階の階層作成プロセスが必要になります。DataKeeper デバイス (階層) では、プライマリサーバ上で LVM ボリュームグループと論理ボリュームを作成する前に、DataKeeper の「Data Replication Resource」オプションを使用した設定を行う必要があります。必要なボリュームグループと論理ボリュームが作成できたら、残りの階層を保護されるアプリケーションに関連したリカバリキットの設定手順に従って作成します。完成した階層は以下に示す図 3 のようになります。 **注記**：データの整合性を保つため、DataKeeper 上に LVM を構成する場合には、DataKeeper ミラー 1 つだけ、または複数の ***同期*** ミラー、いずれかの構成でなければなりません。

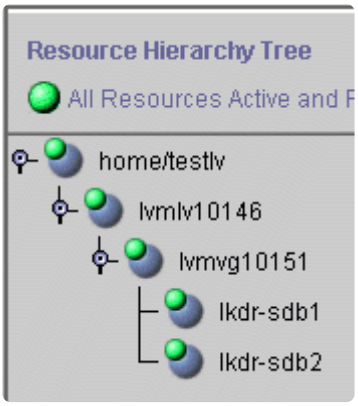



**図 3: DataKeeper 上の LVM を含む階層**

# 6.5.4.3. ボリュームグループの再設定

Logical Volume Manager を利用する最大のメリットの 1 つは、ストレージ要件の変化に応じて、論理ボリュームを動的にサイズ変更できることです。これは、LVM ボリュームグループ定義への物理パーティションやディスクの追加または削除を伴う場合があるため、LVM Recovery Kit には、そのような変更内容を反映して、既存のリソース階層を修正する仕組みが含まれています。

変更内容を反映して SPS 階層を修正する前に、すべてのボリュームグループ、論理ボリュームおよびファイルシステムの再設定を SPS 階層の外部で行う必要があります。その方法については、ドキュメンテーションとリファレンス セクションで紹介する LVM *HowTo* ドキュメントを参照してください。いずれかの手順において、SPS で保護されているリソースを **マウント解除** または **設定解除** する必要が生じた場合、必ず LifeKeeper GUI から **Out of Service** 操作を選択して行ってください。

> **重要:** 新しいデバイスは、SPS が再設定の実行を可能にする前に、両方のシステム (共有) から見えていなければいけません。

これらの変更後に SPS 階層を更新するには、まず、変更した ボリュームグループの **[Resource Properties]** ダイアログを開きます。このダイアログは、アクティブボリュームグループリソースを右クリックするか、 **[Properties]** を選択するか、 **[Edit] > [Resource] > [Properties]** メニューを選択し、**[Select Resource]** フィールドで適切なボリュームグループリソースを選択することで開けます。The resulting **[Resource Properties]** ダイアログは、以下の図 4 に示すように、下部に **[Resource Configuration]** ボタンがあります。

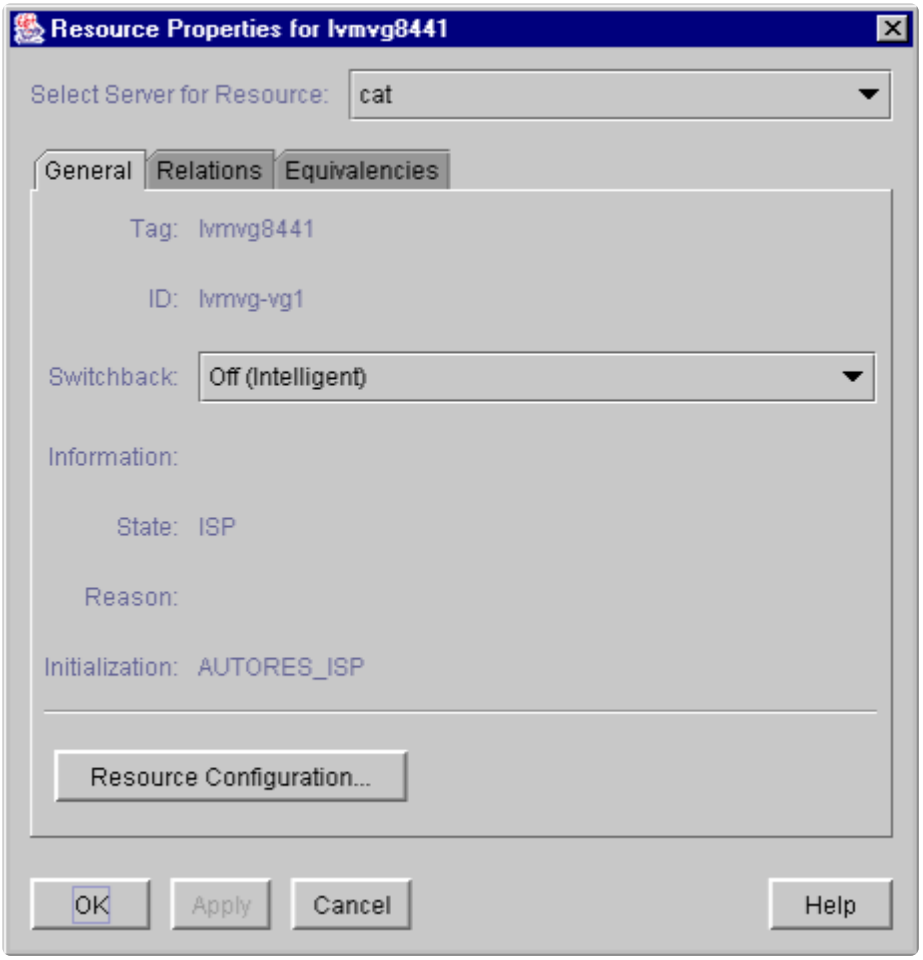



**図 4: LVM ボリュームグループ [Resource Properties] ダイアログ**

**[Resource Configuration]** ボタンをクリックすると、階層を再設定して、ボリュームグループリソースに加えられた変更を反映するための仕組みが開始されます。少し待つと、情報ボックスに SPS が検出したボリュームグループの変更内容が表示されます。以下の図 5 は、ボリュームグループに1つのディスクパーティションが追加された例を示しています。

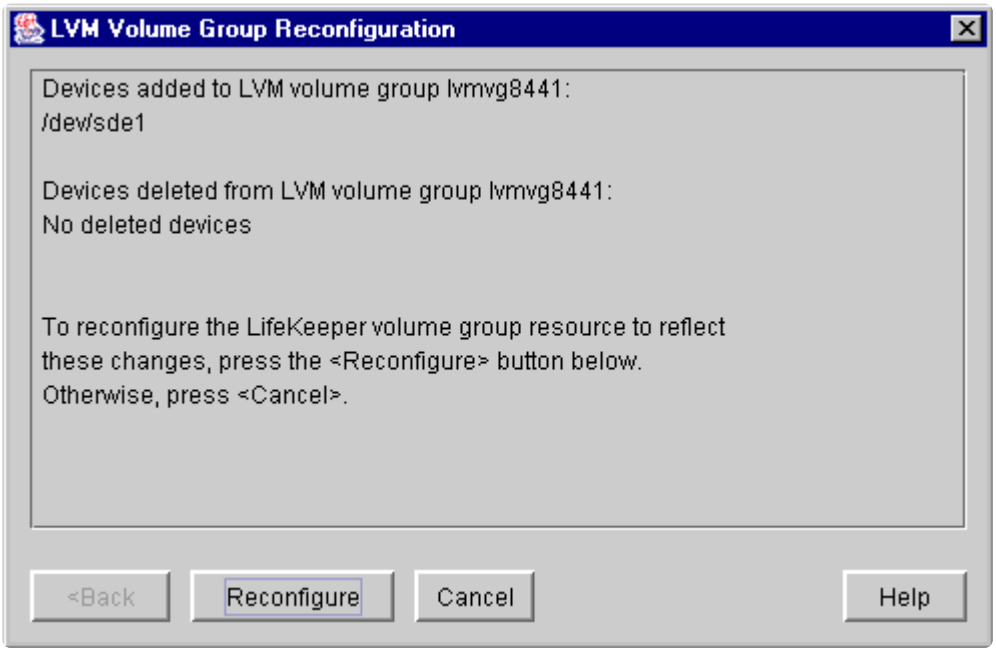


**図 5:** デバイスが追加された**LVM** ボリュームグループの再設定

情報ボックスの説明にあるように、検出された変更内容を反映して SPS ボリュームグループを再設定するには、単に **[Reconfigure]** ボタンをクリックしてください。SPS 階層の変更をそれ以上進めたくない場合は、 **[Cancel]** ボタンをクリックしてください。

**[Reconfigure]** ボタンをクリックした後、再設定手順の進捗状況を示す情報ボックスが表示されます (以下の **図6** ）処理が正常に完了すると、 **[Done]** ボタンが有効になります。 **[Done]** ボタンをクリックすると、情報ボックスが閉じ、 **[Resource Properties]** ダイアログに戻ります。

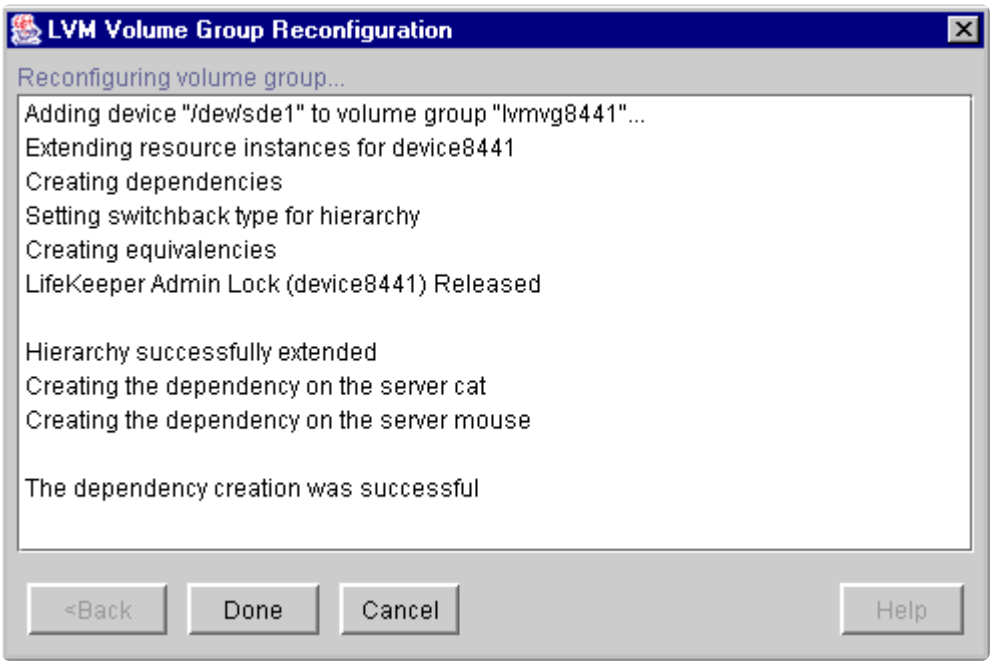



**図 6:** デバイスが追加された**LVM** ボリュームグループの再設定

以下の 2 つの図では、ボリュームグループからデバイスパーティションが削除された際に、再設定の進行中に表示される情報ボックスの例を示します。

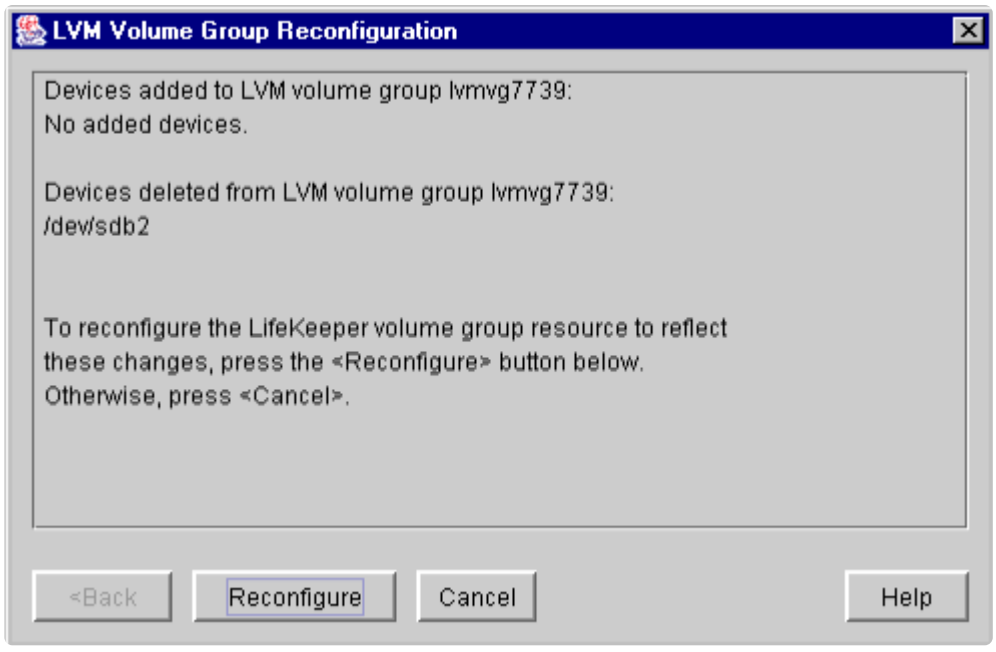


図 7: デバイスが削除されたLVM ボリュームグループの再設定

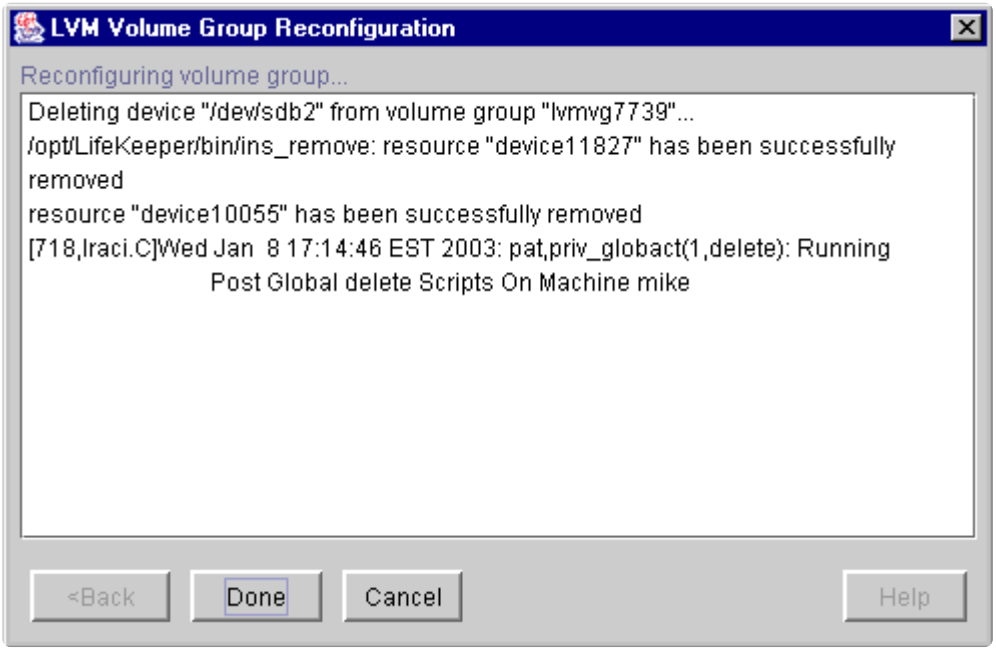



図 8: デバイスが削除されたLVM ボリュームグループの再設定

# 6.5.5. トラブルシューティング

## エラーメッセージ

このセクションでは、SPS LVM Recovery Kit の使用中に表示される可能性のあるメッセージの一覧を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

LVM Recovery Kit は、階層の作成および拡張時に他の SPS コンポーネントを利用するため、それらの他のコンポーネントによるメッセージが表示されることもあります。そのような場合は、メッセージカタログ を参照してください。メッセージカタログには、操作、管理、GUI など、SIOS Protection Suite for Linux を使用しているときに遭遇する可能性がある、すべてのエラーコードが列挙されています。また、エラーコードの原因に関する追加の説明や、問題解決のために必要な処置についても、必要に応じて記載されています。この一覧から、受信したエラーコードを検索できます。また、該当する SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

### LVM Recovery Kit のエラーメッセージ

| エラー番号 | エラーメッセージ |
|---|---|
| 110000 | <LVM resource type> resource type is not installed on <SPS server name>.<br><br>処置: LVM Recovery Kit を特定されたシステムにインストールしてください。 |
| 110001 | This script must be executed on <SPS server name>. |
| 110002 | Failed to create <device name> hierarchy. |
| 110003 | Failed to create dependency <resource tag>-<resource tag> on machine <SPS server name>. |
| 110004 | LifeKeeper internal ID <resource ID> already in use. |
| 110005 | <LVM resource type> constructor requires a valid argument. |
| 110006 | Usage: adddelpv <VG tag> [addlist\|dellist] |
| 110007 | 警告: Failure in updating list of LifeKeeper-controlled volume groups ( `/etc/lkvgs`). |
| 110008 | 警告: The device hierarchy for <device name>, with tag <device resource tag>, cannot be extended automatically.<br><br>処置: デバイス階層がまだ拡張されない場合、LifeKeeper GUI を使って拡張してください。そしてボリュームグループリソースからデバイス階層まで依存関係を作成してください。 |
| 110009 | Failed to create a dependency between volume group resource <volume group tag> and device resource <device tag>.<br><br>処置: LifeKeeper GUI を使って、依存関係を作ってください。 |
| 110010 | Failed to make the LVM logical volume <Logical Volume Path> active with error code <error |

| | |
|---|---|
| | code>. |
| 110011 | Failed to `vgscan` the LVM volume group <Volume Group Name> with error code <error code>. |
| 110012 | Failed to `vgimport` the LVM volume group <Volume Group Name> with error code <error code>. |
| 110013 | Failed to make the LVM volume group <Volume Group Name> active with error code <error code>. |

# 6.6. IP Recovery Kit 管理ガイド

## IP Recovery Kit テクニカルドキュメンテーション

SIOS Protection Suite for Linux Internet Protocol (IP) Recovery Kit は、LifeKeeper 環境で、障害が発生したプライマリーサーバからバックアップサーバへ IP アドレスを復旧するメカニズムを提供します。IP Recovery Kit は、LifeKeeper によって保護されたアプリケーションへの接続に使用できる IP アドレスを定義できます。他の LifeKeeper リソースと同様に、IP リソースのスイッチオーバは、障害が発生した結果として自動的に開始するか、管理者の操作によって手動で開始することができます。

IP Recovery Kit は、論理インターフェース機能を使用した TCP/IP プロトコルスイートの実装をサポートし、追加のスタンバイネットワークインターフェースカードやダミーの IP アドレスを必要とせずに、IP アドレスのスイッチオーバとフェイルオーバを提供できるようにします。リリース 7.4 からは、IP Recovery Kit は IPv4 と IPv6 の両方のアドレスをサポートします。

- SIOS Protection Suite for Linux リリースノート
- SIOS Protection Suite for Linux テクニカルドキュメンテーション
- アプリケーションリカバリーキット

# 6.6.1. 運用の原則

LifeKeeper は、仮想 IP アドレスをプライマリサーバの物理ネットワークインターフェースの 1 つに作成することによって、IP リソースを In Service が可能な状態にしています。ユーザはこの仮想 IP アドレスを使用してノードに接続します。

IP Recovery Kit ソフトウェアは、選択されたアドレス、ネットワークマスク、インターフェースが確実に正しく機能できるかをチェックします。ソフトウェアがチェックする要素は以下のとおりです。

- 未使用のリソース: 新しい IP アドレスが、LifeKeeper クラスター内に存在する他のどの IP リソースにも、割り当てられていないことを確認します。
- 一意のアドレス: このアドレスは、ネットワーク上で「現在は」アクティブにできません。リソース作成中のチェックに加えて、リソースを In Service にする直前にも一意であることの確認が行われます。ネットワーク上で重複するアドレスが検出された場合、そのリソースを In Service にしません。

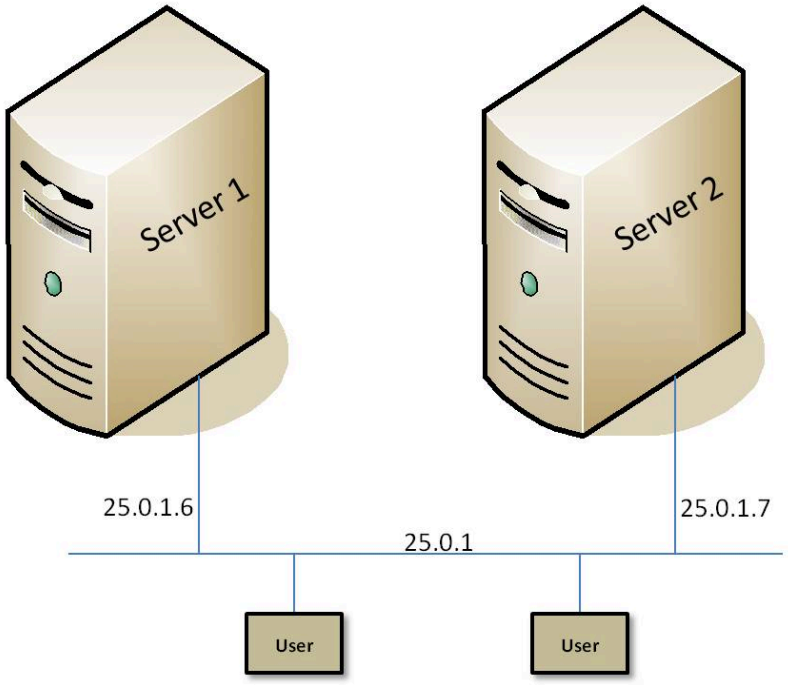
プライマリサーバに障害が発生すると、IP Recovery Kit は仮想 IP を、バックアップサーバの物理ネットワークインターフェースの 1 つに設定することによって、その IP リソースをバックアップサーバで In Service 状態にします。

リカバリ後はセッションコンテキストが失われているため、仮想 IP を利用するユーザは最初に接続するのに使用した手順とまったく同じ手順で再接続する必要があります。

手動のスイッチオーバでは、IP Recovery Kit はそのエイリアスアドレスをアクティブサーバのサービスから削除した後、バックアップサーバに追加します。

IP Recovery Kit の管理と運用について理解しやすいように、図 1 に示したシナリオについて考えてみましょう。この設定例には、Server1 と Server2 という 2 台のサーバで構成されています。各サーバは 1 つの LAN インターフェース (eth0) を備え、サブネット 25.0.1 に接続されています。ユーザシステムもこのサブネット上にあります。Server1 および Server2 上の LAN インターフェースのアドレスは、それぞれ 25.0.1.6 と 25.0.1.7 です。

**図 1.管理および運用のシナリオ**




システム管理者は、 *ipname* という名前の IP リソースのエイリアスアドレスとして 25.0.1.10 を使用することにしました。システム管理者は、 */etc/hosts* ファイル (および使用されている場合は DNS) に以下のようなエントリを作成します。

| | |
|---|---|
| 25.0.1.6 | server1 |
| 25.0.1.7 | server2 |
| 25.0.1.10 | ipname |

Server1 がこのリソースのプライマリサーバであれば、システム管理者はIP リソース階層の作成 セクションで説明しているウィザードを使用して、Server1 上に *ipname* 用の IP リソース階層を作成します。IP Recovery Kit は、 *ipname* (25.0.1.10) に関連付けられているアドレスを */etc/hosts* で見つけて使用可能であることを確認し、セカンダリアドレスを Server1 上の eth0 で設定することによって、アドレスを In Service にします。これで Server1 上の eth0 は、 *server1* と *ipname* の両方に応答するようになります。

LifeKeeper7.3 以前のバージョンでは、新しいエイリアスアドレスは `ifconfig` もしくは `ip addr show` コマンドで確認することができました。LifeKeeper 7.4 からは、 `ip addr show` コマンドを使用してください (詳細については IPv6 既知の問題 を参照してください)。

ユーザはその後、たとえば `telnet ipname` というように入力することで Server1 に接続できます。Server1 がクラッシュした場合、LifeKeeper は *ipname* のアドレスを Server2 の eth0 に自動的にスイッチオーバします。Server1 上のユーザセッションは終了します。ユーザが telnet ipname を再実行した場合は、Server2 に接続されます。

*ipname* がどこでアクティブになって In Service であるかにかかわらず、アドレスとしての *server1* と *server2* はアクティブで、使用することができます。ただし、これらは LifeKeeper のリカバリによって保護されていません。これらのアドレスは、切り替えられたアプリケーションに対してではなく、名前を使って特定のサーバに対して接続する必要がある場合は常に使用できます。例えば、リモートシステム管理や LifeKeeper コミュニケーションパスなどがあります(この場合、LifeKeeper コミュニケーションパスと

して、例えば 25.0.1.6 と 25.0.1.7 が使用されます)。

# IP リソースの監視

LifeKeeper は以下の方法を順番に実行して、管理下にある IP リソースの状態を定期的に監視しています。

1. 該当するネットワークインターフェースにおいて、IP リソースがエイリアスとして設定されているかを確認します。

2. IP リソースが設定されているネットワークインターフェースのリンク状態を確認し、物理ネットワークに正しく接続されているかを検査します。

3. 保護された IP アドレスをソースアドレスとしてブロードキャスト ping テストを行い、IP リソースがネットワーク上で正しくデータの送受信を行えるかを確認します。

   ブロードキャスト ping テストはデフォルトのテストメカニズムです。保護された IP アドレスをソースアドレスとして、ブロードキャスト ping パケットを IP リソースに関連付けられたサブネットのブロードキャストアドレス宛に送信し、テストを行っています。ローカルシステム以外の任意のアドレスから応答を受信するとテストは成功したと判断します。

   ブロードキャスト ping に応答を返せるシステムがネットワーク上にない環境 (多くのシステムのデフォルト設定がこのような環境です) では、LifeKeeper はアドレスリストを定義し、ブロードキャスト ping の代わりに ping を送信するように設定可能です。このリストが定義されている場合、ブロードキャスト ping テストは省略され、リスト内のすべての IP アドレスに対し、同時に ping が実行されます。リスト内のいずれかの IP アドレスからの応答が受信されれば、テストは成功したと判断されます。この技術は、大規模なネットワークにおけるブロードキャストストームの抑止に有効です。

IP リソースの定期的なチェックの中で、これらのテストが失敗すると、LifeKeeper に障害が通知されます。LifeKeeper はまずローカルリカバリを試み、IP リソースをローカルノード上でリストアしようとします。ローカルリカバリ手順の詳細については、IP ローカルリカバリと設定に関する考慮事項 を確認してください。ローカルリカバリが現在のノードでの IP リソースの起動に失敗した場合、LifeKeeper は IP リソースを含めたアプリケーション階層を、クラスタ内の別の LifeKeeper システムに移動させます。

LifeKeeper は同じヘルスチェック機能を使い、 In Service の状態になった直後に IP リソースが正しく動作するかを判定します。チェックの結果、障害が確認された場合、起動は失敗します。

IP ヘルスチェックメカニズムは設定の変更とチューニングを実行することが可能です。詳細については、IP 構成の確認および編集 およびIP Recovery Kit のチューニング を確認してください。

# 6.6.2. 要件

IP Recovery Kit のインストールやアンインストールを試みる前に、このパッケージのハードウェア / ソフトウェア要件と、インストールとアンインストールの手順を理解しておく必要があります。

## キットのハードウェア / ソフトウェア要件

LifeKeeper IP Recovery Kit をインストールして設定する前に、使用している設定が以下の要件を満たしていることを確認してください。

- **サーバ**：この Recovery Kit を使用するには、SIOS Protection Suite for Linux テクニカルドキュメンテーション 、およびSIOS Protection Suite for Linux リリースノート で説明されている LifeKeeper の要件に従って設定したコンピュータが 2 台以上必要です。

- **LifeKeeper ソフトウェア** 。各サーバに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。具体的な LifeKeeper の要件については、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート を参照してください。

- **LifeKeeper IP Recovery Kit** 。同じバージョンの Recovery Kit を各サーバにインストールする必要があります。

- **IP ネットワークインターフェース** 。各サーバに、イーサーネット TCP/IP をサポートするネットワークインターフェースが 1 つ以上必要です。IP スイッチオーバが正しく動作するには、ローカルネットワークに接続されているユーザシステムが標準の TCP/IP 仕様に準拠している必要があります。このインターフェースは設定する必要があります。 *ifcfg** ファイルが存在しない場合、インターフェースがダウンしたときに IP スイッチオーバが失敗する可能性があります。

  **注記**：各サーバが必要とするネットワークインターフェースが 1 つだけであっても、異種メディア要件、スループット要件、単一障害点の排除、ネットワークのセグメンテーション、ローカルリカバリのサポートといった多くの理由で、複数のインターフェースを使用すべきです。

  また、IPリソースの設定に使用するNICには、あらかじめOSの操作を使用して実IPアドレスを設定しておいてください。その実IPアドレスを使用して、IPネットワーク上の基本的な疎通確認等を実施するようにしてください。

  動作確認を実施しているチャンネルボンディング、ネットワークチーミングの構成については、こちら を参照してください。

- **TCP/IP ソフトウェア:** 各サーバに TCP/IP ソフトウェアも必要です。

最新のリリースの互換性および購入方法については、SPS for Linux リリースノート を参照するか、営業担当者にお問い合わせください。

LifeKeeper IP Recovery Kit ソフトウェアのインストール / アンインストール方法の具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

# 6.6.3. 設定

必要とする保護と柔軟性が得られるように LifeKeeper を設定するには、設定ルールを把握している必要があります。設定を適切にプランニングするには、ネットワーク設定、インターフェースの選択、ユーザシステムのセットアップ、階層オプション、IP 設定作業について理解することが必要です。このセクションでは、設定のプランニングに加え、Recovery Kit を設定するために必要な特定の作業についても、設定例を示して説明します。

## LifeKeeper での TCP/IP の設定

このセクションでは、TCP/IP の設定を開始する前に考慮する必要のある情報と、典型的な LifeKeeper IP 設定の例を紹介します。

LifeKeeper Core のリソース階層の設定に関する手順については、SPS for Linux テクニカルドキュメンテーション を参照してください。

## TCP/IP 固有の設定上の考慮事項

IP Recovery Kit を正しく設定するために、以下のトピックを見直して、設定作業を完了するために必要な情報を用意してあることを確認してください。

- インターフェースの選択
- ユーザシステムのセットアップ
- 一般的な IP プランニングの考慮事項

更なる設定上の考慮事項とサンプルについては、以下のトピックを参照してください。

- IP のリソース監視と設定に関する考慮事項
- IP のローカルリカバリと設定に関する考慮事項
- 設定例
- IP 依存関係の作成時のガイドライン

## LifeKeeper の設定作業

次の仮想 IP アドレスリソースの設定作業は、IP リソースインスタンス特有のもので、Recovery Kit ごとに異なるため、本セクションで説明しています。

- リソース階層の作成：アプリケーションリソース階層を LifeKeeper クラスタに作成します。
- リソース階層の削除：リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。

- リソース階層の拡張：リソース階層をプライマリサーバからバックアップサーバへ拡張します。
- リソース階層の拡張解除：リソース階層を LifeKeeper クラスタ内の 1 つのサーバから拡張解除 (削除) します。
- リソース階層のテスト：仮想 IP リソース階層が正しく構成されているか、動作するかをテストします。
- IP 設定の確認および編集：IP リソースの設定を表示し、それらを修正することが可能です。
- IP Recovery Kit のチューニング：IP Recovery Kit の動作を調整します。

次の作業は、すべての Recovery Kit に共通する手順のため、SPS for Linux テクニカルドキュメンテーション の管理 セクションに記載されています。

- リソース依存関係の作成 。既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除 。リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service 。リソース階層を特定のサーバで In Service にします。
- Out of Service 。リソース階層を特定のサーバで Out of Service にします。
- プロパティの表示 / 編集 。特定のサーバでリソース階層のプロパティを表示または編集します。

**注記:** このセクションの残りの部分では、LifeKeeper GUI の **[Edit]** メニューから作業を選択することによって、Recovery Kit を設定する方法を説明します。設定作業はツールバーから選択することもできます。状況表示ウィンドウのリソース階層ツリー(左側のペイン)のグローバルリソースを右クリックして、 **[Edit]** メニューと同じドロップダウンメニューの選択項目を表示することもできます。言うまでもなく、これは階層がすでに存在している場合にだけ可能な方法です。

状況表示ウィンドウのリソース階層表(右側のペイン)のリソースインスタンスを右クリックして、サーバおよび特定リソースの状況に応じて、リソース階層の作成を除くすべての設定作業を実行することもできます。

# 6.6.3.1. IP Recovery Kit のチューニング

IP Recovery Kit の動作のチューニング値をこちら で説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。IP Recovery Kit のコンポーネントはメモリに常駐していないため、 */etc/default/LifeKeeper* の値を変更すると、即時に反映されます。LifeKeeper の再起動を行う必要はありません。

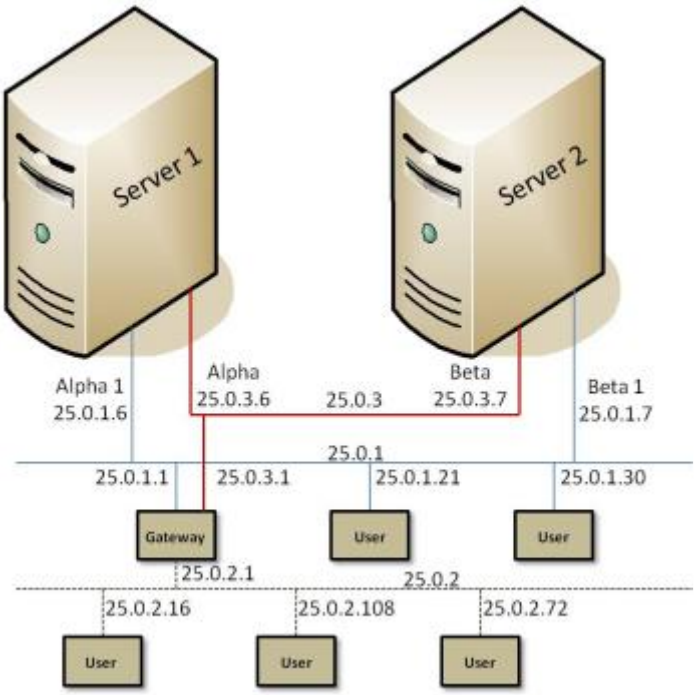
# 6.6.3.2. 設定例

このトピックでは、ネットワークの設定例を示し、2 種類の IP 設定例について説明します。最初の例は、単一の IP リソースに依存して既存のサブネットに設定されている、データベースアプリケーションの典型的な例を示しています。2 番目の例は、複数の IP リソースが設定されたアクティブ / アクティブのシナリオを示しています。

## ネットワーク設定

最初の 2 つの設定例は、以下の図に示すネットワーク設定を前提にしています。

ネットワーク設定




このネットワーク設定には以下のコンポーネントが存在します。

- サーバ。 Server 1 と Server 2 の 2 台のサーバがあり、それぞれに適切な LifeKeeper およびアプリケーションソフトウェアがインストールされています。
- インターフェース。 各サーバは、次のように設定された 2 つのイーサーネットインターフェース、eth0 と eth1 を備えています。

| インターフェース | Server 1 | Server 2 |
|---|---|---|
| eth0 | Server1<br><br>25.0.3.6 | Server2<br><br>25.0.3.7 |

| | | |
|---|---|---|
| eth1 | Server11<br><br>25.0.1.6 | Server21<br><br>25.0.1.7 |

- **ネットワーク。** ネットワークは、以下の 3 つのサブネットワークで構成されています。
  - 
    - 主にサーバ用の、トラフィックが少ないバックボーン(25.0.3)
  - 
    - サーバとクライアント両方のための、トラフィックが多いバックボーン(25.0.1)
  - 
    - トラフィックが多いクライアントネットワーク(25.0.2)

      1 つのゲートウェイで、すべての LAN 間の相互接続ルーティングを行います。アドレス解決には DNS(図示されていない)が使用されます。
- **ハートビート。** TCP ハートビートコミュニケーションパスは、片方または両方のサーバサブネットワークを使用して設定されます。

## 典型的な設定例

Server 1 と Server 2 は、共有ディスクに常駐する mydatabase と呼ばれるアプリケーションにアクセスします。mydatabase アプリケーションと、このアプリケーションへのアクセスに使用される IP リソースが一緒に切り替えられるようにするため、システム管理者は mydatabase アプリケーションリソースを作成し、依存関係で IP リソースをアプリケーション階層に追加します。

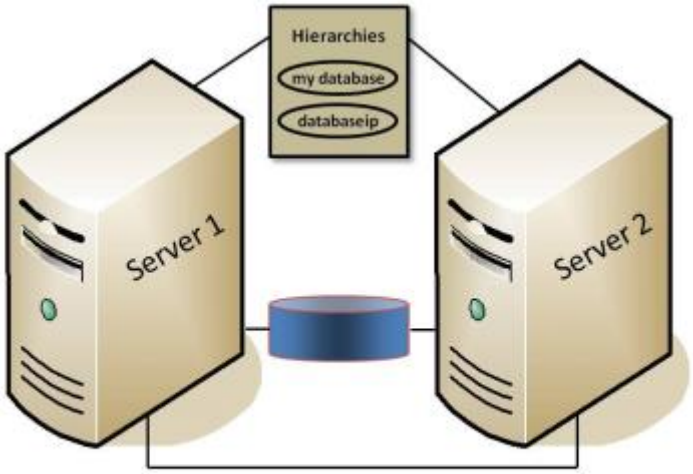
以下のような設定上の問題があります。

- **アプリケーション階層。** アプリケーション階層が存在しなければ、管理者はそれを IP リソースの親として指定できません。この例では、Server 1 がプライマリサーバです。アプリケーションリソースのタグは、mydatabase-on-server1 と mydatabase-on-server2 です。
- **IP リソース名。** システム管理者は、IP リソースの名前とアドレスを Server 1 と Server 2 の */etc/hosts* ファイル、および DNS データベースに追加します。この例では、IP リソース名は databaseip、そのネットワークアドレスは 25.0.1.2 です。名前と IP アドレスの関連付けが必要でない場合、この作業は不要です。
- **ルータ、ゲートウェイ、ユーザ。** databaseip は既存のサブネット上のアドレスであるため、追加の設定は不要です。IP リソースは、25.0.1 サブネット上にあります。すべてのユーザが、現在 25.0.1 サブネットに到達するために使用しているルートを経由して databaseip に接続します。たとえば、25.0.2 のユーザはゲートウェイ経由で接続し、25.0.1 のユーザは直接接続します。
- **IP インスタンスの定義。** システム管理者が [Resource Hierarchy Create] 画面で IP リソースとして databaseip を入力すると、ソフトウェアによっていくつかのテストが実行されます。Server 1 が、databaseip に付随するアドレスを特定可能かどうかが確認されます(アドレスは hosts ファイル内に

あるか、DNS から検索できます)。検索されたアドレス 25.0.1.2 がすでに使用されていないかも検査されます。IP リソースは 25.0.1 サブネット上にあるため、IP リカバリソフトウェアは、そのリソースが eth1 インターフェースに設定されていることを確認します。IP リソースに問題がなければ、ウィザードの残りのダイアログボックスに、図 3 の下の表に示したようなデフォルト値が設定されます。すべてのデフォルト値を選択すると、独立した IP リソース階層が ip-databaseip という名前で作成されます。

**注記:** それぞれの設定図に付随する表は、プライマリサーバ(Server 1)の場合は [Create Resource Hierarchy] ウィザードに、バックアップサーバ(Server 2)の場合は [Extend Resource Hierarchy] ウィザードに入力される適切な情報の例を示しています。ウィザードに入力する情報の詳細については、このセクションで後述するLifeKeeper 設定作業 セクションを参照してください。これらの表は、Recovery Kit を設定するときに役立つ参照情報となります。

**図 3.IP リソース作成時の典型的な設定例**




設定に関する注記:

1. アプリケーションリソースは mydatabase-on-server1 です。
2. IP リソースは databaseip、タグ名は ip-databaseip です。
3. mydatabase-on-server1 が障害を起こすと、LifeKeeper はそれを Server 2 に切り替えます (ipdatabaseip は、依存関係がある場合にのみ切り替えられます)。
4. Server 1 が障害を起こすと、両方のリソースが Server 2 に切り替えられます。
5. スイッチオーバの間に、databaseip のユーザは切断されます。ユーザは再度ログインすると、Server 2 のすべてのアプリケーションにアクセスできます。
6. 手動によるスイッチオーバの間、databaseip 以外の接続を使って Server 1 に接続しているユーザについては、Server 1 との接続が維持されます。

### Server 1 での IP リソース階層の作成

| | |
|---|---|
| Server: | Server1 |
| IP Resource: | databaseip |
| Netmask: | 255.255.252.0 |
| Network Interface: | eth1 |
| IP Resource Tag: | ip-databaseip |

**注記:** IP リソースをバックアップサーバへ拡張する前に IP 依存関係の作成時のガイドライン を参照してください。

### IP リソース階層の Server 2 への拡張

| | |
|---|---|
| Template Server: | Server1 |
| Tag to Extend: | databaseip |
| Target Server: | Server2 |
| Target Priority: | 10 |
| **IP Resource: | 25.0.1.2 |
| Netmask: | 255.255.252.0 |
| Network Interface: | eth1 |
| IP Resource Tag: | ip-databaseip |

** [Extend Wizard] では、その DNS 名に関連付けられている実際の IP アドレスが IP リソースとして表示されます。

## IP リソースのテスト

IP リソースが正常に作成されたことを確認するために、システム管理者は以下の作業を行う必要があります。

1. LifeKeeper の GUI メニューから、ip-databaseip が Server 1 で In Service (ISP)かどうかを確かめます。
2. リモートサーバから、ping または telnet を使用してアドレス databaseip に接続します。
3. Server 2 で in_service オプションを選択し、さらに ip-databaseip を選択して、手動のスイッチオーバをテストします。IP アドレスが Server 2 に移行されることを確認してください。

# アクティブ / アクティブ設定の例

同じネットワーク設定を使用した 2 番目の例では、各サーバで 1 つずつアクティブになった 2 つの IP リソースを示します。

## リソースアドレス

この例では、IP リソースは server1ip (アドレス 25.0.6.20)と server2ip (アドレス 25.0.6.21)です。これらのリソースのエントリが、各サーバの */etc/hosts* ファイルと、DNS データベースになければなりません。

## ルータの設定

選択されたアドレスは新しい(論理)サブネット上にあるため、eth0 または eth1 に設定できます。ただし、どちらも同じインターフェースを使用している必要があります。

この例では、eth0 を選択すると、すべてのユーザがゲートウェイを経由しなければならなくなります。eth1 を選択すると、サブネット 25.0.1 のユーザはリソースに直接アクセスできます(新しいサブネットが内部のルーティングテーブルに追加されていることが前提です)。サブネット 25.0.2 上のユーザには、やはりゲートウェイが必要です。この例の場合、選択されたインターフェースは eth1 です。

新しいサブネットをサポートするために、どの物理ネットワークが選択されたかにかかわりなく、ネットワーク管理者はルーティング情報をゲートウェイシステムに追加してから IP リソースを作成する必要があります

## 最初の IP リソース定義

管理者は、最初の IP リソースを Server 1 に作成します。eth0 は、各サーバで利用可能な最初のインターフェースであり、デフォルトとして提示されます。eth1 をインターフェースとして定義するときは、利用可能なインターフェースの一覧から選択します。

### Server 1 での IP リソース階層の作成

| | |
|---|---|
| Server: | Server1 |
| IP Resource: | server1ip |
| Netmask: | 255.255.252.0 |
| Network Interface: | eth1 |
| IP Resource Tag: | ip-server1ip |

**注記:** IP リソースをバックアップサーバへ拡張する前にIP 依存関係の作成時のガイドライン を参照してください。

### IP リソース階層の Server 2 への拡張

| | |
|---|---|
| Template Server: | Server1 |
| Tag to Extend: | server1ip |
| Target Server: | Server2 |

| | |
|---|---|
| Target Priority: | 10 |
| **IP Resouce: | 25.0.6.20 |
| Netmask: | 255.255.252.0 |
| Network Interface: | eth1 |
| IP Resource Tag: | ip-server1ip |

** [Extend Wizard] では、その DNS 名に関連付けられている実際の IP アドレスが IP リソースとして表示されます。

## 2 番目の IP リソース定義

管理者は、2 番目の IP リソースを Server 2 に作成します。eth0 は、各サーバで利用可能な最初のインターフェースであり、デフォルトとして提示されます。eth1 をインターフェースとして定義するときは、利用可能なインターフェースの一覧から選択します。

### Server 2 での IP リソース階層の作成

| | |
|---|---|
| Server: | Server2 |
| IP Resource: | server2ip |
| Netmask: | 255.255.252.0 |
| Network Interface: | eth1 |
| IP Resource Tag: | ip-server2ip |

**注記:** IP リソースをバックアップサーバへ拡張する前に IP 依存関係の作成時のガイドライン を参照してください。

### IP リソース階層の Server 1 への拡張

| | |
|---|---|
| Template Server: | Server2 |
| Tag to Extend: | server2ip |
| Target Server: | Server1 |
| Target Priority: | 10 |

| | |
|---|---|
| **IP Resouce: | 25.0.6.21 |
| Netmask: | 255.255.252.0 |
| Network Interface: | eth1 |
| IP Resource Tag: | ip-server2ip |

** [Extend Wizard] では、その DNS 名に関連付けられている実際の IP アドレスが IP リソースとして表示されます。

**注記:** サブネット 25.0.6 は Server 2 ではアクティブでないため、eth0 と eth1 の両方をプライマリネットワークインターフェースのために選択できます。Server 1(バックアップサーバ)では、最初の IP リソース 25.0.6.20 が In Service であるため、eth1 だけが選択可能です。管理者が定義を保存すると、LifeKeeper はアドレス 25.0.6.21 を Server2 の eth1 で In Service にします。

## IP リソースのテスト

管理者は以下のテストを行って、新しいリソースが両方のサーバで機能していることを確認する必要があります。

1. プライマリサーバの各リソースについて、ping または telnet を使用してそれぞれにアクセス可能であるか確認します。管理者は、すべてのユーザサイトからの接続をテストすることもできます。

2. 手動で ip-server1ip を Server 2 で In Service にして、スイッチオーバをテストします。両方のリソースが Server 2 で機能していることを確認してください。

3. 両方のリソースを Server 1 で In Service にします。両方のリソースが Server 1 で機能していることを確認してください。

4. ip-server2ip を、プライマリサーバの Server 2 で In Service に戻します。

# 6.6.3.3. IP リソース階層の作成

プライマリサーバからリソースインスタンスを作成するには、以下の手順を完了する必要があります。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択してください。

2. ダイアログボックスが表示され、クラスタ内にインストールされ認識されているすべての Recovery Kit が、ドロップダウンリストボックスに一覧で示されます。ドロップダウンリストから IP を選択し **[Next]** をクリックします。

3. 以下の情報を入力するようにプロンプトが表示されます。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、以前に入力した情報を訂正する必要がある場合に特に役立ちます。

   階層作成の途中で **[Cancel]** ボタンをクリックすると、作成処理全体が取り消されます。

| フィールド | ヒント |
|---|---|
| **Switchback Type** | バックアップサーバへのフェイルオーバー後にサーバが復帰したときに、このサーバに IP インスタンスを戻す方法を指定します。 *intelligent* または *automatic* を選択できます。 *intelligent* の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。 *automatic* の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。スイッチバックタイプは、必要な場合、[Resource Properties] ダイアログボックスの [General] タブで後から変更できます。 |
| **Server** | IP アドレスを配置する **サーバ** (通常これをプライマリサーバまたはテンプレートサーバと呼ぶ) を選択します。クラスタ内のすべてのサーバがドロップダウンリストに表示されます。 |
| **IP Resource** | 実際の **IP リソース** を選択または入力します。これは、LifeKeeper がこのリソースのために使用する IP アドレスまたはシンボル名です。クライアントアプリケーションは、この IP リソースを使用して、特定のネットワークインターフェース経由で親アプリケーションにログインします。シンボル名を使用する場合、そのシンボル名は、ローカルの */etc/hosts* ファイル内に存在するか、DNS によってアクセスできる必要があります。エイリアス名およびドメイン名は、上に示した条件を満たしていれば使用できます。この情報フィールドにデフォルト値はありません。<br><br>**注記:** シンボル名を使用する場合、このリソースを拡張するときに、実際の IP アドレスが IP リソースの指定としてダイアログボックスに表示されるため注意が必要です。 |
| **Netmask** | ネットワークマスク **(ネットマスク)** を選択または入力します。IP リソースは、このネットマスクをターゲットサーバで使用します。有効な値は、特定の IP リソースアドレスのクラスを表す標準のネットマスクです。<br><br>**注記:** ここで選択したネットマスクと IP アドレスの組み合わせによって、IP リソースが使用する、ネットワーク設定と一貫性があるサブネットが決まります。 |

| | |
|---|---|
| **Network Interface** | LifeKeeper に保護されて IP リソースが配置される **ネットワークインターフェース** を選択または入力します。これは、IP アドレスがインターフェースで接続する物理イーサーネットカードです。選択が有効かどうかは、既存のネットワーク設定と、IP リソースアドレスおよびネットマスクのために選択された値によって決まります。デフォルト値は、一連の有効な選択項目のうち、手順 4 と 5 で選択したアドレスとネットマスクの値に最もよく一致するインターフェースです。 |
| **IP Resource Tag** | 作成している IP リソースインスタンスに固有の **IP リソースタグ** 名を選択または入力します。このフィールドには、デフォルトのタグ名の ip-<resource> が自動的に入ります。<resource> はリソース名または IP アドレスです。このタグは必要に応じて変更できます。 |

4. **[Create]** をクリックします。 **[Create Resource Wizard]** によって、IP リソースが作成されます。

5. この時点で情報ボックスが表示され、LifeKeeper は、IP リソース階層を作成するために有効なデータが指定されたかどうかを検証します。LifeKeeper が問題を検知した場合は、情報ボックスにエラーが表示されます。検証に成功すると、リソースが作成されます。 **[Next]** をクリックします。

6. 別の情報ボックスが表示され、IP リソース階層が正常に作成されたことが示されます。IP リソース階層を LifeKeeper の保護下に置くには、その階層をクラスタ内の別のサーバに拡張する必要があります。

**[Continue]** をクリックすると、 **[Pre-Extend configuration task]** が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、階層の拡張 を参照してください。

ここで **[Cancel]** をクリックすると、別のダイアログボックスが表示され、IP リソース階層を LifeKeeper の保護下に置くには、いずれかの時点でこの手順に戻り、階層を別のサーバに拡張する必要があるという警告が表示されます。

# 6.6.3.4. リソース階層の削除

LifeKeeper 環境内のすべてのサーバからリソース階層を削除するには、以下の手順を実行します。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択します。ドロップダウンメニューから、 **[Delete Resource Hierarchy]** を選択します。

2. **[Target Server]** で、IP リソース階層を削除するターゲットサーバの名前を選択し、 **[Next]** をクリックします。(このダイアログボックスは、いずれかのペインでリソースインスタンスを右クリックして、 **[Delete Resource]** を選択した場合には表示されません。)

3. **[Hierarchy to Delete]** を選択します。削除するリソース階層を特定して、強調表示にします。 **[Next]** をクリックします。(このダイアログは、右ペインまたは左ペインのリソースインスタンス上で右クリックして **[Delete Resource]** を選択した場合には表示されません。)

4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[Delete]** をクリックして、次のダイアログボックスに進みます。

5. IP リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。

6. **[Done]** をクリックして、 **[Delete Resource Hierarchy]** メニューの選択を終了します。

# 6.6.3.5. リソース階層の拡張

**注記:** IP リソースをバックアップサーバへ拡張する前に、IP 依存関係の作成時のガイドライン セクションを参照してください。

階層を作成後、クラスタ内の別のサーバに拡張する必要があります。3 通りのシナリオで、テンプレートサーバからターゲットサーバにリソースインスタンスを拡張できます。最初のシナリオは、リソース作成後、[Continue] をクリックして、別のサーバにリソースを拡張する場合です。2 番目のシナリオは、以下に示すように [Edit] メニューから [Extend Resource Hierarchy] を選択する場合です。3 番目のシナリオは、左側または右側のペインから拡張されていない階層を右クリックする場合です。どのシナリオでも同じダイアログボックスが表示されます (いくつかの例外については、以下に詳細を明記する)。

1. LifeKeeperGUI メニューから ***[Extend wizard]*** を開始する場合は、 **[Edit]** を選択し、次に **[Resource]** を選択します。ドロップダウンメニューから **[Extend Resource Hierarchy]** を選択します。これで ***[Extend Resource Hierarchy wizard]*** が起動されます。拡張操作に慣れていない場合は、 **[Next]** をクリックしてください。LifeKeeper の **[Extend Resource Hierarchy]** のデフォルト値が分かっていて、入力と確認を省略する場合は **[Accept Defaults]** をクリックしてください。

2. ***Pre-Extend Wizard*** に以下の情報を入力します。 **注記**: 最初の 2 つのフィールドは **[Edit]** メニューの **[Extend]** から操作を開始した場合にだけ表示されます。階層を拡張する手順の間に **[Cancel]** をクリックすると、どの時点であってもそのサーバへの拡張処理がキャンセルされますので注意してください。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

| フィールド | ヒント |
|---|---|
| **Template Server** | IP リソースが現在サービス中のサーバを入力する。 |
| **Tag to Extend** | 拡張する IP リソースを選択する。これは、テンプレートサーバからターゲットサーバに拡張する IP インスタンスの名前です。ウィザードのドロップダウンリストボックスに、テンプレートサーバ上に作成した全リソースのうち、前のダイアログボックスで選択したリソースの一覧が表示されます。 |
| **Target Server** | IP リソース階層を拡張する **ターゲットサーバ** を選択してください。ドロップダウンボックスには、選択した階層に含まれないクラスタ内のサーバ名の一覧が表示されます。 |
| **Switchback Type** | **スイッチバックタイプ** を選択してください。これは、バックアップサーバへのフェイルオーバ後に IP インスタンスが In Service に戻る際に、このサーバに IP インスタンスをスイッチバックする方法を指定します。 *[intelligent]* または *[automatic]* を選択できます。intelligent の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。automatic の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。スイッチバックタイプは、必要に応じて [Resource Properties] ダイアログボックスの [General] タブで後から変更できます。 |
| **Template** | **テンプレートの優先順位** を選択するか、入力してください。これはサーバで現在サー |

| | |
|---|---|
| **Priority** | ビス中の IP 階層の優先順位です。優先順位は、1 ～ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります (1 が最高の優先順位)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を使用することを推奨します。 **注記:** このフィールドは、階層をはじめて拡張するときにだけ表示されます。 |
| **Target Priority** | **ターゲットの優先順位** を選択するか、入力してください。これは、新しく拡張する IP 階層が、他のサーバに存在する同等の階層に対して持つ相対的な優先順位です。1 ～ 999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位が高くなります (数値 1 が最高の優先順位)。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |

3. 情報ボックスが表示され、LifeKeeper による環境のチェックが正常に完了したこと、およびこの IP リソースを拡張するための要件がすべて満たされたことが示されます。満たされていない要件がある場合は、[Next] ボタンは選択できなくなり、[Back] ボタンが有効になります。[Back] をクリックした場合、情報ボックスに表示されるエラーメッセージの内容に従って、リソースの拡張を変更できます。ここで [Cancel] をクリックする場合は、いずれかの時点でこの手順に戻り、IP リソース階層を他のサーバに拡張して LifeKeeper の保護下に置く必要があります。[Next] をクリックすると、[Extend Resource Hierarchy] 設定作業に入ります。
4. リソース階層構造の拡張を行うために、以下の情報の入力が要求されます。

| フィールド | ヒント |
|---|---|
| **IP Resource** | [Create Resource Wizard] で作成した **IP リソース** またはアドレスと同じです。このダイアログボックスは情報の表示専用です。このボックスに表示される **IP リソース** は変更できません。 |
| **Netmask** | これは、このテンプレートサーバの IP リソースが作成されたときに選択された **ネットマスク** と同じで、ここではターゲットサーバの IP リソースによって使用されます。このダイアログボックスは情報の表示専用です。このボックスに表示される **ネットマスク** は変更できません。 |
| **Network Interface** | **ネットワークインターフェース** を選択するか、入力してください。これは、IP リソースがターゲットサーバで使用するネットワークインターフェース (つまりイーサーネットカード) の名前です。 |
| **IP Resource Tag** | **IP リソースタグ** を選択するか、入力してください。これは、ターゲットサーバに拡張される IP リソースが使用するリソースタグ名です。 |

5. 拡張が実行されていることを示す情報ボックスが表示されます。

   同じ IP リソースインスタンスをクラスタ内の別のサーバに拡張する場合は、 **[Next Server]** をクリックします。その場合は、リソース階層の拡張の操作が繰り返されます。

**[Finish]** をクリックすると、IP リソースの拡張が正常に完了したことが LifeKeeper によって確認されます。

6. **[Done]** をクリックして、 **[Extend Resources Hierarchy]** メニューの選択を終了します。

   **注記:** 必ずすべてのサーバで新しいインスタンスの機能をテストしてください。

# 6.6.3.6. 一般的な IP プランニングの考慮事項

IP リソース階層で使用する予定のアドレス、ネットマスク、関連するホスト名 / ドメイン名を選択したら、各サーバの */etc/hosts* ファイルに適切なエントリを追加します。DNS を使用している場合は、DNS にもエントリを追加します。

**注記:** DNS を使用している場合でも、すべての LifeKeeper サーバ上のローカル */etc/hosts* ファイルに IP リソースのエントリを記述することを強くお勧めします。それによって、リカバリ時間が短縮されます。ただし、IP インスタンスを作成するときに入力するリソース名が IP アドレス自体の場合、ホストファイルのエントリは不要です。

保護された IP アドレスは、恒久的な論理インターフェースを作成する場合のように、システムのブート時にアクティブになるようシステムに設定しないでください。LifeKeeper ソフトウェアはシステムに代わってそれらの管理を行います。

いずれかのリソースアドレスが新しい(論理)サブネット上にある場合は、それらのサブネットへのルーティングを処理するようにルータをアップデートしてください。

# 6.6.3.7. IP 依存関係の作成時のガイドライン

LifeKeeper で保護されるアプリケーションと LifeKeeper で保護される IP アドレス間の親 / 子の依存関係を、いつ、どのように作成するかは、一般に LifeKeeper で保護されるアプリケーションによって決まります。例えば、LifeKeeper で保護された Apache 環境では、親 / 子の依存関係は、(保護された IP アドレスをすでに作成している場合は) Apache リソース階層の作成時に作成されます。この依存関係が自動的に作成されない他のアプリケーションでは、次の手順に従うことをお勧めします。

1. アプリケーション / 親リソース階層を作成します。注記: この時点では、リソース階層をバックアップサーバに拡張しないでください。階層を拡張しないことを選ぶと警告メッセージが出されますが、この場合は拡張しないことが正しい操作です

2. IP リソース階層を作成します。注記:この時点では、IP リソース階層をバックアップサーバに拡張しないでください。階層を拡張しないことを選ぶと警告メッセージが出されますが、この場合は拡張しないことが正しい操作です

3. 親アプリケーションリソース階層と IP リソース階層の間の親 / 子依存関係を、[Create Resource Dependency] 設定作業を使用して作成します。(LifeKeeper for Linux のトピック、リソース依存関係の作成 を参照)

4. 最後に、アプリケーションリソース階層をバックアップサーバに拡張します。すでに依存関係が作成されているため、依存する IP リソースインスタンスも、親アプリケーションリソース階層の一部としてバックアップサーバに拡張されます。

上記の手順に従えば、余分な拡張 (つまり、IP リソースのバックアップサーバへの拡張) を行わずにすみます。

# 6.6.3.8. インターフェースの選択

IP リソースを作成するときには、IP リソースのアドレス、そのアドレスで使用するネットマスク、ネットワークインターフェースを選択します。すべての組み合わせが許されるわけではありません。指定するアドレスとネットマスクのペアと、現在 In Service になっているアドレスとネットマスクのすべてのペアによって選択肢が決まります。また、Recovery Kit の IP ローカルリカバリ機能を使用する予定がある場合、IP ローカルリカバリ セクションで追加の設定上の考慮事項を参照してください。

選択したアドレスとネットマスクのペアにより、リソースのサブネットが決まります。同じサブネット上で、別のアドレス (物理インターフェースアドレスまたは論理インターフェースアドレス) がいずれかのインターフェース上で現在 In Service である場合、そのインターフェースで IP リソースを設定する必要があります。ソフトウェアはテストを実行し、現在のネットワーク設定に基づいて、可能な選択肢を特定します。提示される選択肢からは、任意のものを選択できます。

IP リカバリキットソフトウェアが物理メディアのタイプを区別しないため、リソースが使用する物理ネットワークを決定し、適切にアドレスを選択してください。例えば、サブネット xx.yy.12 のイーサーネットバックボーンと、サブネット xx.yy.20 および xx.yy.30 のイーサーネット LAN に接続されたサーバがあるとします。最初のイーサーネットサブネット上にリソースを作成する場合は、xx.yy.20.120 のように、目的のサブネット上のアドレスを選択します。

一般的に、IP リカバリキットソフトウェアがあらゆる値のネットマスクの選択を許可したとしても、複数のネットマスクがパケットのミスルーティングを引き起こす原因となりうるため、同じ物理的なインターフェースに対して複数のネットマスクを選択することは避けなければなりません。

もう 1 つの考慮点として、すべての LifeKeeper サーバで、インターフェースの選択に一貫性を持たせる必要があります。サーバ A で、1 つのインターフェースに IP リソースをいくつか設定した場合は、サーバ B でも、1 つのインターフェースにそれらの IP リソースを設定する必要があります。

IP リソース階層を作成する時は、起動時に最初に立ち上がっていてプライマリおよびバックアップ側の両ノードで設定が一致しているネットワークインターフェースを利用するといいでしょう。例えば eth1 を利用する場合は、IP アドレスが割り当てられていないような構成、またはサーバ起動直後のいくつかのネットワークインターフェースが起動していないような時でも、eth1 は起動している必要があり、 *ifcfg-eth1* ファイルは両ノードで正しく設定されていなければなりません (`ifup/ifdown ifcfg-eth1` コマンドで確認する)。

# 6.6.3.9. IP ローカルリカバリと設定に関する考慮事項

HA の構成におけるネットワークインターフェースの冗長性を確保するために、Linux 標準の NIC のボンディングの利用をお勧めします。LifeKeeper IP Recovery Kit は、ボンディングインターフェース上に作成された仮想 IP アドレスを完全にサポートします。

## ローカルリカバリのシナリオ

IP Recovery Kit によって IP アドレスの障害が検出されると、結果として生じる障害によって IP ローカルリカバリスクリプトが起動されます。LifeKeeper は最初に、その IP アドレスを現在のネットワークインターフェース上で In Service の状態に戻そうとします。ローカルでリカバリの試みが失敗すると、LifeKeeper は IP リソースと依存関係を持つすべてのリソースのバックアップサーバへのフェイルオーバを実行します。

# 6.6.3.10. IP のリソース監視と設定に関する考慮事項

デフォルトでは、LifeKeeper IP Recovery Kit は、IP アドレスの論理サブネットでブロードキャスト ping を実行し、応答を確認することによって IP リソースを監視します。このテストが正しく動作するには、IP リソースと同じ論理サブネットに属する IP アドレスを持ち、ブロードキャスト ping に応答できる別のシステムが、物理ネットワーク上に少なくとも 1 つ存在する必要があります。この要件を満たすには、ネットワーク上で常に 1 つ以上のシステムが利用可能なようにする必要があります。同じ論理サブネット上にルータを配置すれば、通常は十分にこの要件が満たされます。多くのデバイスにおいて、デフォルトの設定ではブロードキャスト ping に応答しないことに注意してください。そのため、少なくとも 1 つのデバイスの設定を変更しておく必要があるでしょう。

この要件を満たせない場合は、ブロードキャスト ping を完全に無効にするか、ブロードキャスト ping に代わる手段として確認される IP アドレスの静的リストを設定してください。これらの設定方法に関する追加情報については、IP Recovery Kit のチューニング および IP 構成の確認および編集 を参照してください。

# 6.6.3.11. リソース階層のテスト

手動スイッチオーバを開始することによって、IP リソース階層をテストできます。これにより、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

## GUI からの手動スイッチオーバの実行

手動スイッチオーバを開始するには、LifeKeeper の GUI メニューから **[Edit]** 、 **[Resource]** の順に選択し、最後にドロップダウンメニューから **[In Service]** を選択します。例えば、バックアップサーバで [In Service] 要求を実行すると、アプリケーション階層はバックアップサーバで in service になり、プライマリサーバでは out of service になります。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバに変わります。

**[Out of Service]** 要求を実行すると、アプリケーションは他のサーバで in service にならずに、 out of service になります。

手動スイッチオーバでは、IP Recovery Kit はそのアドレスをアクティブサーバのサービスから削除した後、バックアップサーバに追加します。

スイッチオーバ後、その IP リソースのハードウェア (MAC) アドレスは別のアドレスになります。これは、IP リソースが別の LAN インターフェースに関係付けられるためです。ユーザシステムが再接続できるのは、ユーザシステムの TCP/IP ソフトウェアが、この新しいアドレスマッピングを特定してからです。IP Recovery Kit は、ARP (アドレス解決プロトコル) テーブルをアップデートして新しいマッピングを反映する必要があることを、接続されたすべてのサーバに自動的に通知します。

フル機能の TCP/IP 実装を実行しているユーザシステムは、すぐにアップデートされます。それより低機能の実装を備えたユーザシステムでは、アップデートが遅れたり、アドレス指定の仲介役としてルータが必要とされる場合があります。

# 6.6.3.12. リソース階層の拡張解除

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択します。ドロップダウンメニューから、 **[Unextend Resource Hierarchy]** を選択します。

2. **[Target Server]** で、IP リソースを拡張解除するターゲットサーバを選択します。IP アドレスが現在 In Service のサーバは選択できません。

   **注記:** 右側のペインから個々のリソースインスタンスを右クリックして拡張解除作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

3. **[IP Hierarchy to Unextend]** を選択してください。

   **注記:** 左側のペインからグローバルリソースを右クリックするか、右側のペインから個々のリソースインスタンスを右クリックして拡張解除作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. 拡張解除のために選択したターゲットサーバと IP リソース階層を確認する情報ボックスが表示されます。

   **[Unextend]** をクリックします。

5. IP リソースが正常に拡張解除されたことを確認する別の情報ボックスが表示されます。

6. **[Done]** をクリックして、 **[Unextend Resource Hierarchy]** メニューを終了します。

# 6.6.3.13. ユーザシステムのセットアップ

1 つの IP リソースが、2 台のサーバ間で IP Recovery Kit ソフトウェアによって切り替えられる場合、インターフェースが変更されるため、切り替えられる IP アドレスに対応する MAC アドレスが変化します。LAN 上のルータやユーザシステムは、この変更をそれぞれの ARP テーブルに反映しないと、新しい場所の IP アドレスにアクセスできません。オペレーティングシステムによっては、ネットワークインターフェースに新しい IP アドレスが追加されると、ARP パケットが自動的に送信され、サブネット上の全クライアントの ARP テーブルがアップデートされます。Linux にこの機能はありません。そのため LifeKeeper は、切り替え可能な IP アドレスをインターフェースに追加したら ARP パケットを送信し、クライアントの ARP キャッシュを強制的にアップデートする必要があります。

TCP/IP の実装は、このARP パケットに応じて必要な ARP のアップデートを実行する機能の点でさまざまです。以下に、重要な例の一部の説明を示します。

- **Linux TCP/IP の全機能の実装:** Linux や他のオペレーティングシステムのほとんどが備える TCP/IP の実装では、システムが ARP 要求パケットを受信したときに ARP キャッシュをアップデートする機能がサポートされています。LifeKeeper は前述のようにこの機能を利用して、それらのシステムの ARP キャッシュを強制的にアップデートします。

- **ARP キャッシュ:** 上記のような ARP の実装をサポートしていなくても ARP キャッシュはサポートしているユーザシステムは、通常、キャッシュ関連のタイマを備え、内容の新しさをあるレベルに維持しています。一部の実装では、タイマの値を減らすことによって、変更されたアドレスのマッピングがそのユーザシステムに反映されるのに必要な時間を最小限にすることができます。LAN 上のユーザ一数が少ない場合は、この方法を採用しても構いません。タイマの値を減らす必要がないシステムもあります。たとえば、Windows NT の TCP 実装が使用するタイマ値は 10 秒なので、値を変更する必要はないでしょう。

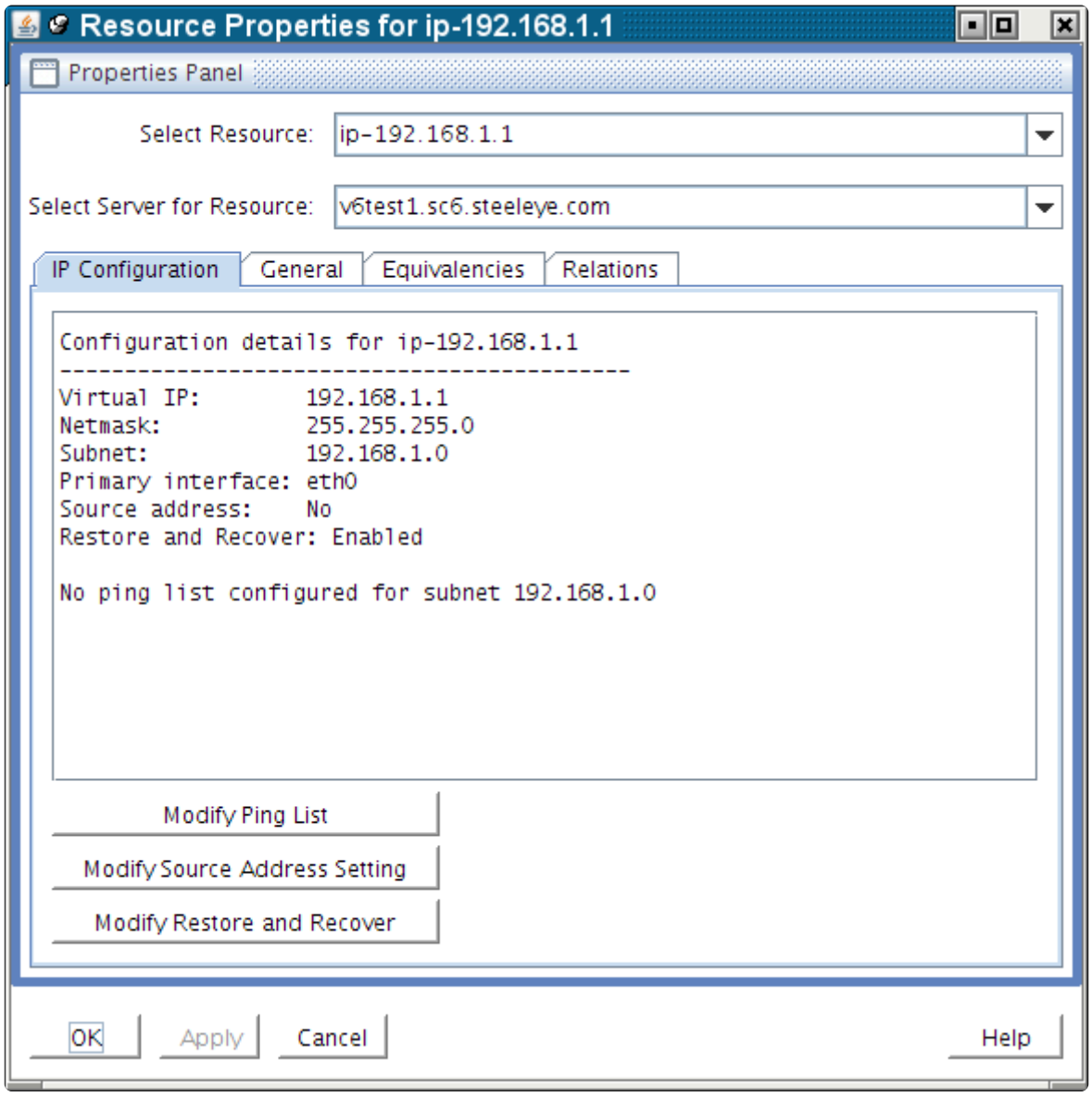
- 静的アドレスマッピング: 動的 ARP キャッシュを持たないシステムや、キャッシュのタイミングを調整できないシステムでは、ルータを使用してマッピングの変更を処理できます。そのようなユーザシステムは、ルータ (ゲートウェイ) 経由で IP リソースのサブネットにアクセスします。この設定では、キャッシュのアップデートが必要なのはリソースサブネットに直接接続されているルータだけで、ユーザシステム自体で必要な変更はありません。

# 6.6.3.14. IP 構成の確認および編集

**[IP Configuration Properties]** ページでは、特定の IP リソースに関する詳細な設定を確認し、選択された設定項目について修正を行うことができます。

**[IP Configuration Properties]** ページにアクセスするには、LifeKeeper GUI メニューから **[Edit]**、**[Resource]** の順に選択します。ドロップダウンメニューから **[Properties]** を選択します。次にプロパティを表示するリソースとサーバを **[Resource list]** と **[Server list]** から選択します。同様に特定の IP リソースインスタンスを右クリックし、コンテキストメニューからプロパティ画面にアクセスすることも可能です。

以下は IP リソースのプロパティ画面の例です。

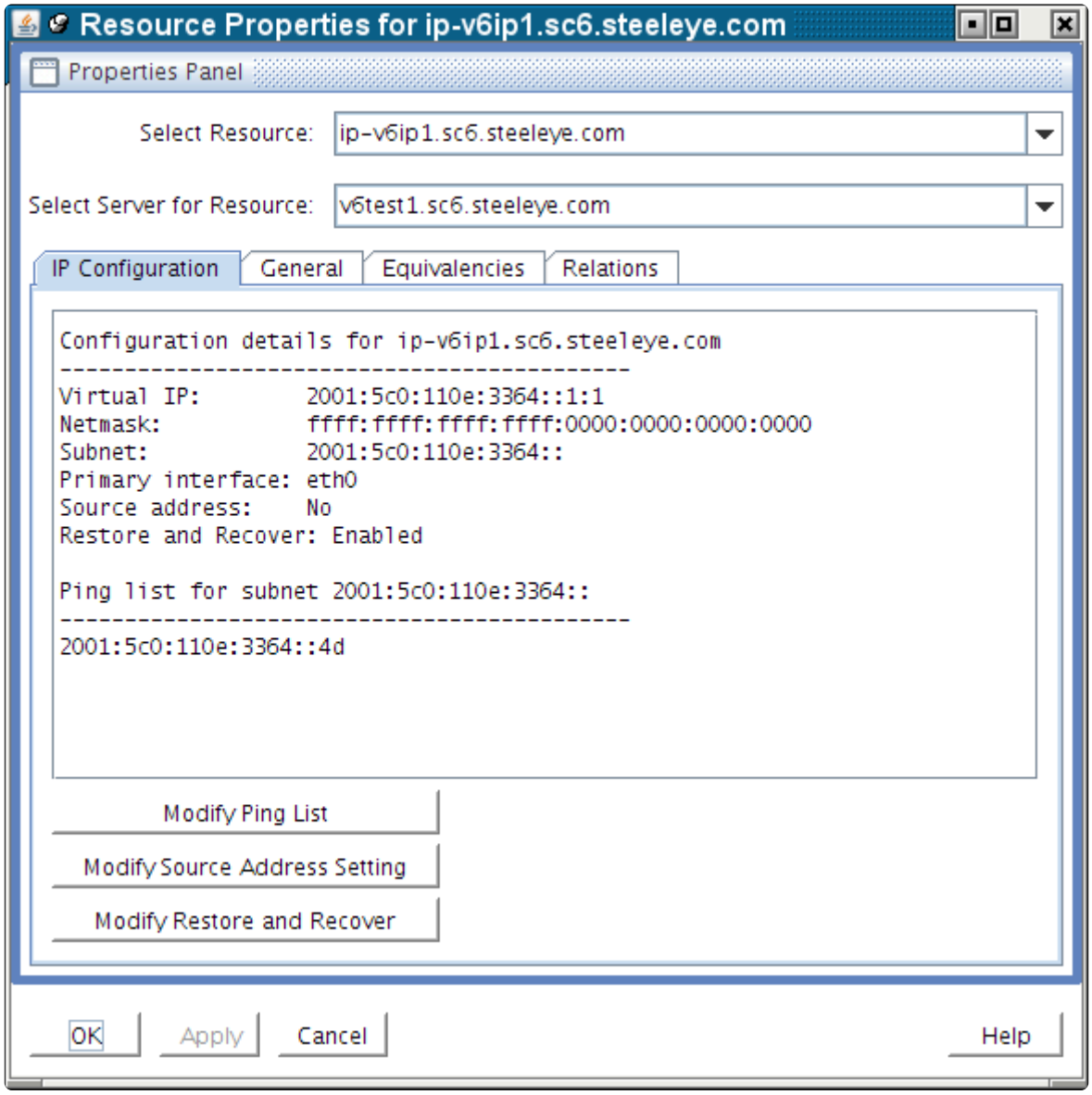



表示されたプロパティ画面にはタブが 4 つあります。1 番目のタブは **[IP Configuration]** で, 特定の IP リソースの設定情報が設定されています。残り 3 つのタブは、LifeKeeper のすべてのリソースタイプで共通の項目として使用できます。

[IP Configuration] タブには特定 IP リソースの以下の情報が表示されています。

| | |
|---|---|
| Virtual IP | IP リソースの仮想 IP アドレス |
| Netmask | 仮想 IP アドレスのネットマスク。この値により、サブネット分割が決定される |
| Subnet | 仮想 IP アドレスの論理サブネット、サブネット部分のビット数を含む |
| Primary interface | アクティブな場合、仮想 IP アドレスを設定する必要のあるネットワークインターフェース |
| Source address setting | 仮想 IP アドレスを同一サブネットへの外部通信 IP トラフィックのソースアドレスに使用するかどうかの設定 |
| Ping List | オプション用 Ping 先リスト。 IP ヘルスチェック時に、通常のブロードキャスト ping メカニズムの代替として ping を通して、この IP リソース (および同一サブネットの他の IP リソース) の動作をチェックする |

上記の図の例では、IP リソース上に Ping List は設定されていません。Ping List が設定されている場合、プロパティ画面は以下の図の例のようになります。

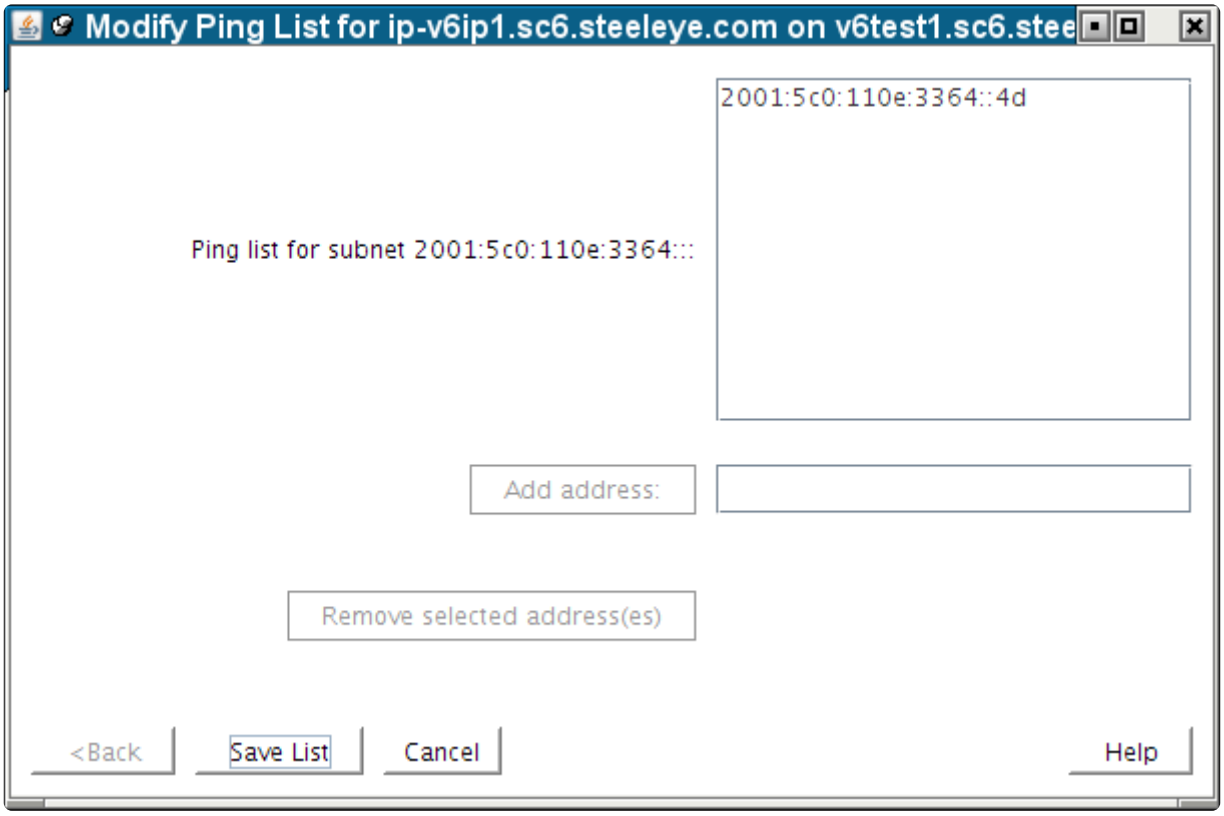



**[Modify Ping List]** および **[Modify Source Address Setting]** ボタンを使用して、この設定項目を修正することが可能です。以下のセクションで説明します。

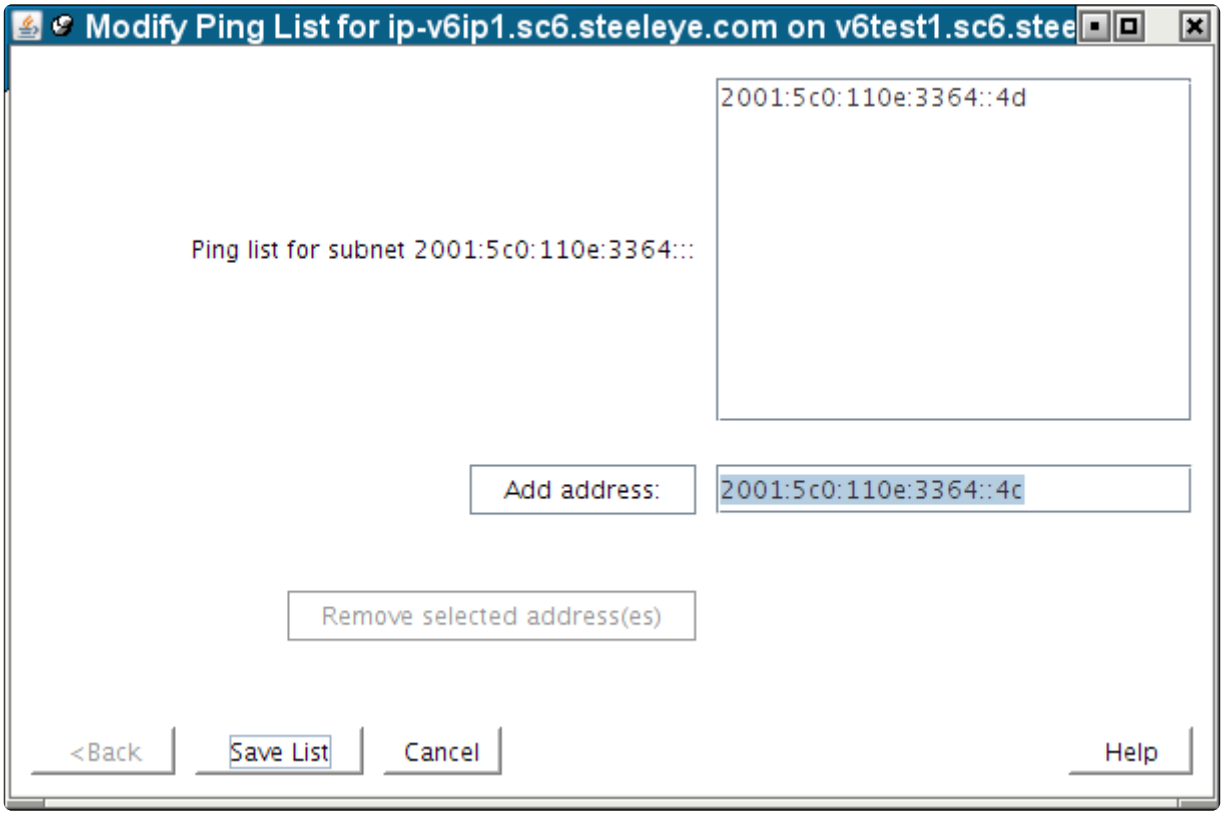
# Ping List の修正

IP リソースに関する Ping List 機能の使い方の説明については、IP リソースの監視 をご参照ください。

IP リソースの Ping List を作成または修正するには、 **[IP Configuration]** プロパティ画面の **[Modify Ping List]** ボタンをクリックします。以下の例のようにダイアログウィンドウが表示されます。

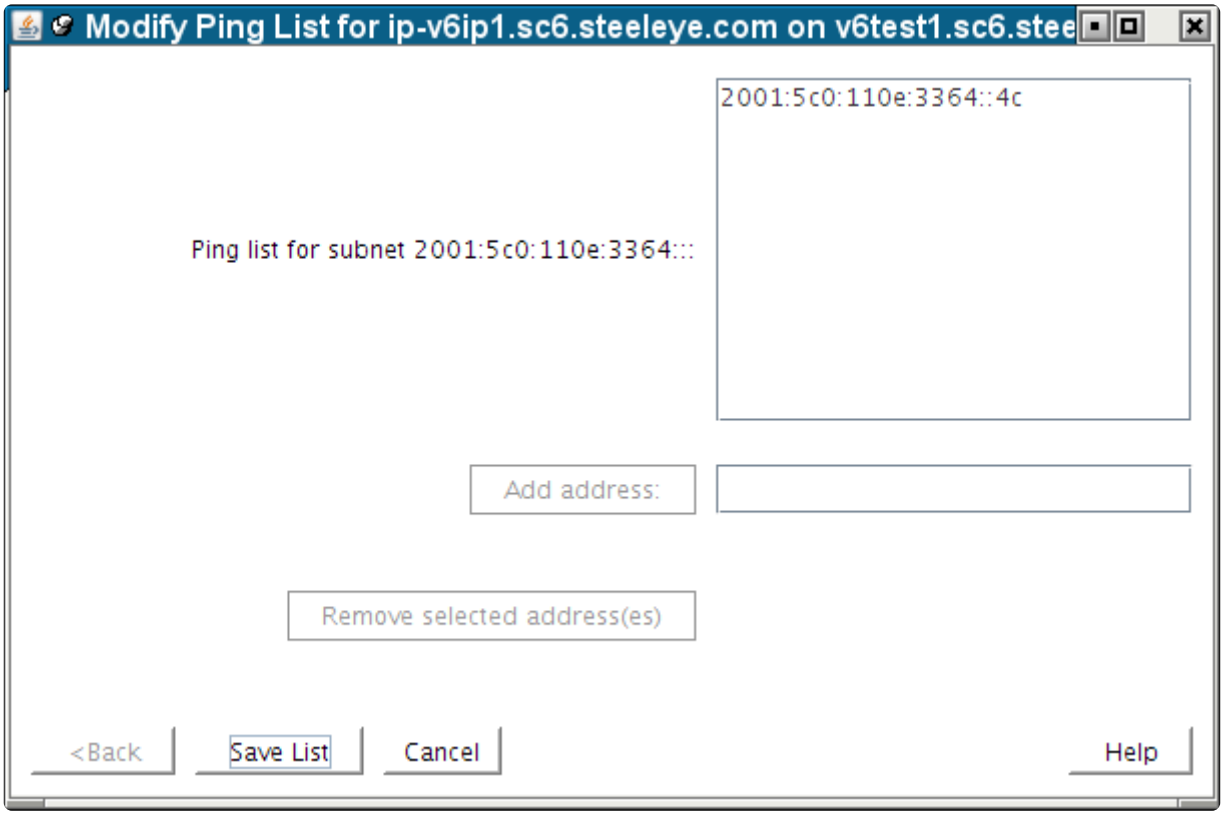



アドレスを Ping List に追加する場合、以下の 2 つの図に示すように [Add address] ボタンの横のフィールドにアドレスを入力し、 **[Add address]** をクリックします。 **[Add address]** ボタンは入力欄にアドレスを入力しない限りグレーアウトされています。



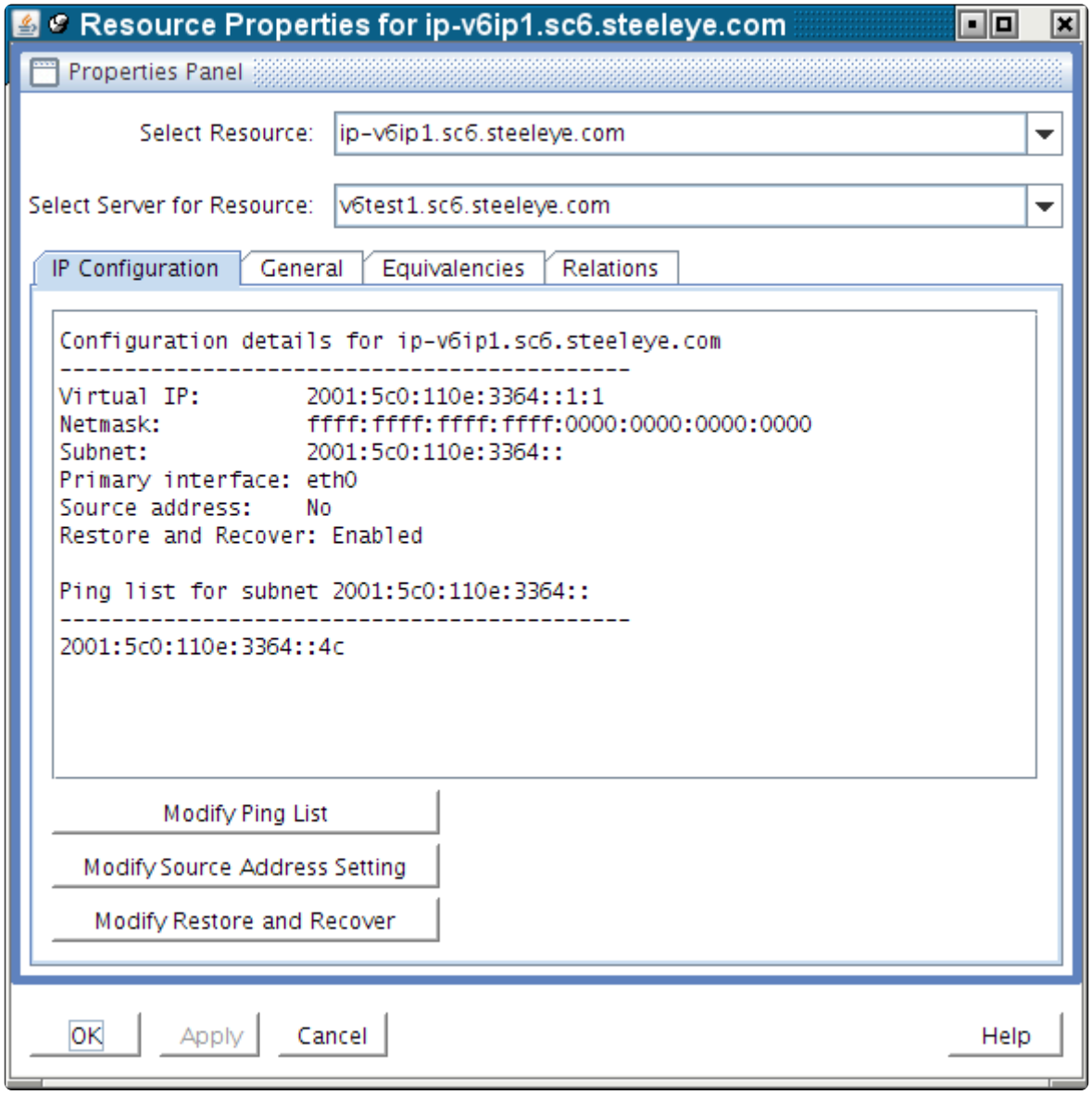

Ping List からアドレスを削除する場合は、削除したいアドレスを選択し、 **[Remove selected address(es)]** ボタンをクリックします。 **[Remove selected address(es)]** ボタンも同様に、リスト内で

アドレスを 1 つ以上選択しない限りグレーアウトされています。




修正済みリストを保存するため、**[Save List]** をクリックします。以下の確認ウィンドウが表示されます。

**[Done]** をクリックしてウィンドウを終了させ、 **[IP Configuration]** プロパティ画面に戻ると、Ping List が修正されているのが確認できます。




## Ping List の使用に関する重要事項

IP リソースの *Ping List* 構成はシステムごとに個別の設定となります。IP リソースが、Ping List 構成後に他のシステムに拡張された場合、同じ設定の Pinglist が拡張先 LifeKeeper システムに一緒にコピーされます。しかし、すでに IP リソースが拡張済みの場合は、Ping List の構成は IP リソースが定義されたそれぞれのシステムに個別に設定する必要があります。Ping List の修正は、IP リソースの状態にかかわらず実行できるため、それぞれのシステムの Ping List を変更するためにIP リソースのスイッチオーバを実行する必要はありません。

同一論理サブネットに複数の IP リソースが作られている場合、これらのリソースは同じ Ping List を共有する必要があります。これは [IP Configuration] プロパティ画面に反映され、Ping List 修正ダイアログボックスは IP リソースのサブネットごとに設定されます。

IP リソースに Ping List が定義されると、リソースのヘルスチェックはデフォルトのブロードキャスト ping メカニズムの代わりに、Ping List メカニズムを使用します。ブロードキャスト ping メカニズムを再度利用するためには、Ping List に定義した全 IP アドレスを削除する必要があります。

LifeKeeper の動作では、入力された Ping List の IP アドレスについて、フォーマットが正しいかどうか以

外の検査は実行しません。そのため、入力した IP アドレスが実際にネットワーク上に存在しており、常に正常に動作しているかについて、LifeKeeper がインストールされているシステムから ping を実行するなどの方法で確認する必要があります。また LifeKeeper がインストールされているシステムの IP アドレスは常に動作していることが予想されますが、これらの IP アドレスを Ping List に選択しないでください。これらのアドレスへの ping 実行は、NIC に関連付けられている実際の IP リソースの状態にかかわりなく、常に成功してしまう可能性があり、正しいチェックとなりません。

上記にて指摘している通り、IP リソースの Ping List を定義すると、LifeKeeper は自動的に Ping List メカニズムをブロードキャスト ping メカニズムの代わりに、対象の IP リソースおよび同一サブネットのアドレス設定をされた他の IP リソースに対して、使用します。この時、ブロードキャスト ping メカニズムを使用不可にするために、IP Recovery Kit のチューニング で解説されている NOBCASTPING 値の設定を行う必要はありません。しかしながら、ネットワーク上にブロードキャスト ping に対して回答可能なシステムが存在しない環境で設定する場合であれば、NOBCASTPING=1 と設定する必要があります。この設定を行わないと、IP リソース作成時にブロードキャスト ping によるチェックが実行されてしまうためです。この段階ではまだ Ping List を作成することができません。IP リソース作成が済み、Ping List を設定後であれば、デフォルトの NOBCASTPING=0 に戻して構いません。

リソース削除時に、Ping List の内容は削除されません。そのため、リソース削除後に以前と同一のサブネットアドレスを持つ IP リソースを作成した場合、以前設定した Ping List の内容が残ってしまうことに注意してください。

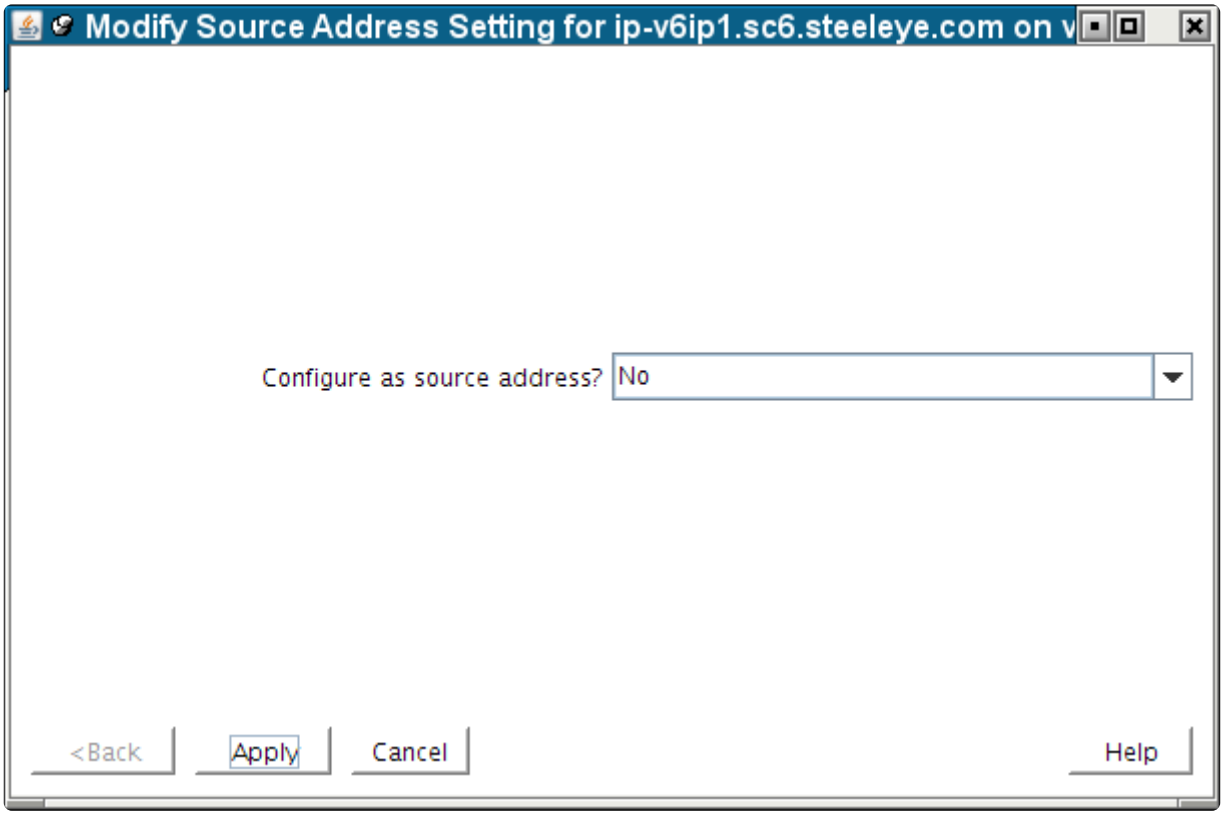
## ソースアドレス設定の修正

IP リソースのソースアドレス設定は、IP リソース起動時に、同じサブネットへの外部通信トラフィックのソースアドレスに仮想 IP アドレスを使用するかどうか決定します。この値のデフォルトは [No] と設定されており、同一サブネットへの外部への通信のソースアドレスは NIC に設定されたプライマリ IP アドレスが使用されます。この動作は大半のケースにおいて適切な設定です。仮想 IP アドレスはクライアントからの接続ポイントに使用されるため、通常は内向きのトラフィックとして生成されたものが使用されることになります。

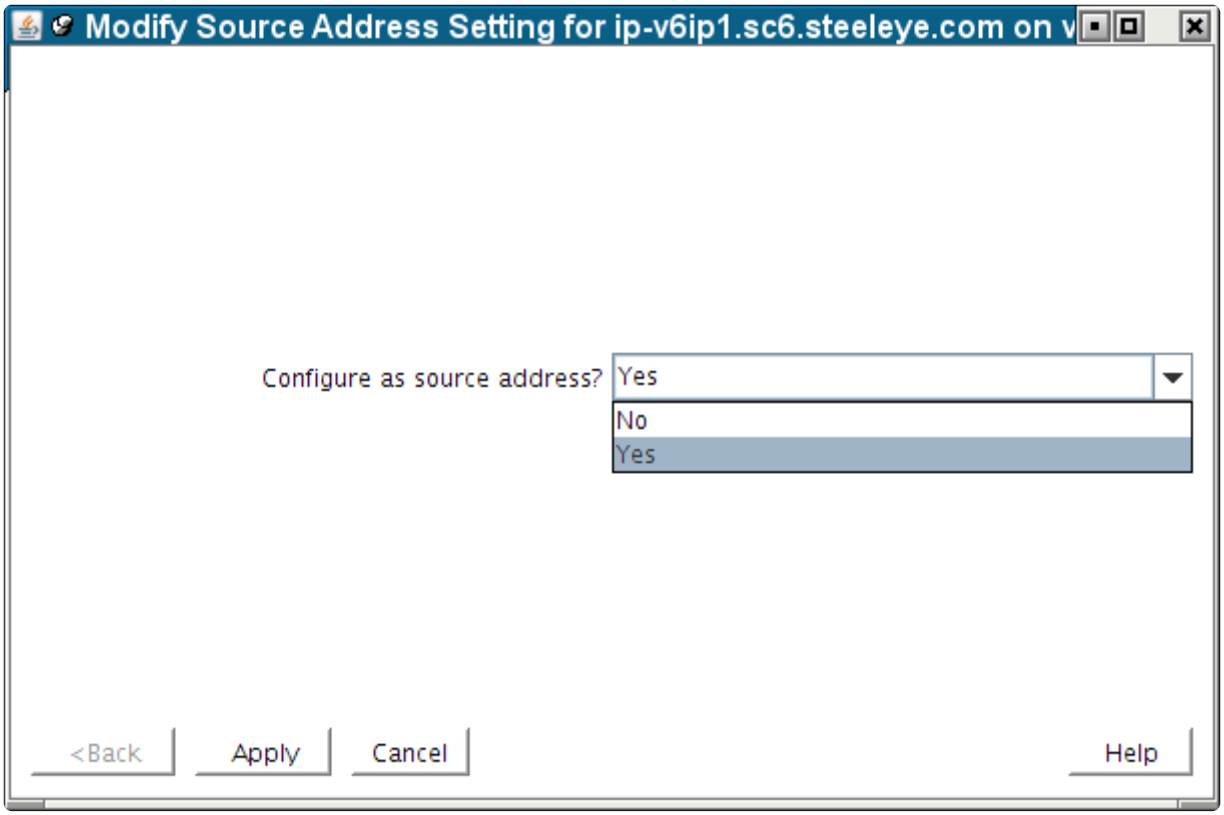
しかし特定の状況または設定のために、Lifekeeper システムからの接続に仮想 IP をソースとする必要がある場合もあり得ます。このため、ソースアドレスを変更するには、IP リソースが起動している状態で [Source Address Setting] を [Yes] にします。システムの TCP/IP ルートはこの設定を行うことで修正されます。

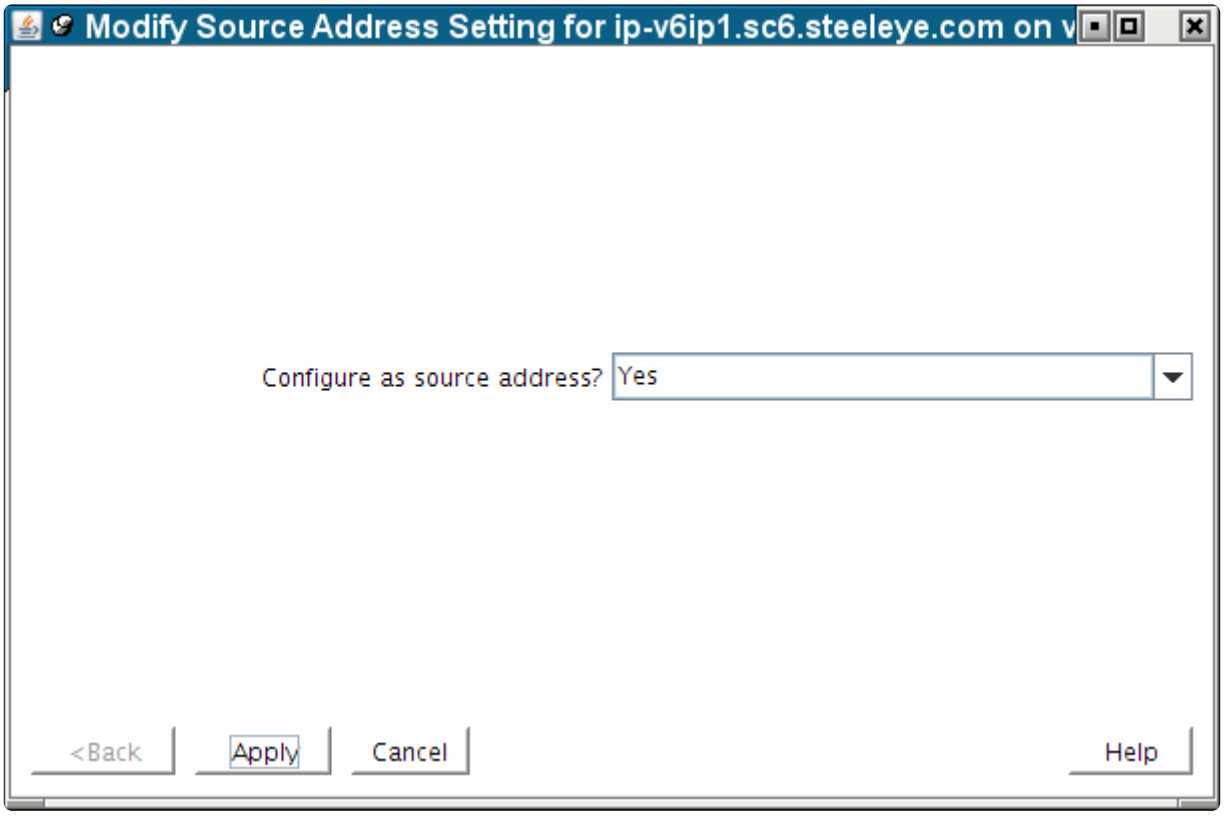
仮想 IP がシステムの実 IP アドレスと異なる論理サブネット上に存在する場合、そのサブネットへの全ての送信トラフィックは、ソースアドレス設定の修正なしに仮想 IP アドレスから行われることになります。さらに、CIDR/サブネットの範囲外の仮想IPでEC2 ルートテーブルを構成する場合、そのIPがサブネット内で構成されている唯一のIPであるため、トラフィックのソースアドレスは、ソースアドレス設定に関係なく仮想IP になります。

IP リソースのソースアドレス設定を修正するには、[IP Configuration] プロパティ画面の [Modify Source Address Setting] ボタンをクリックします。以下の例のようにダイアログウィンドウが表示されます。

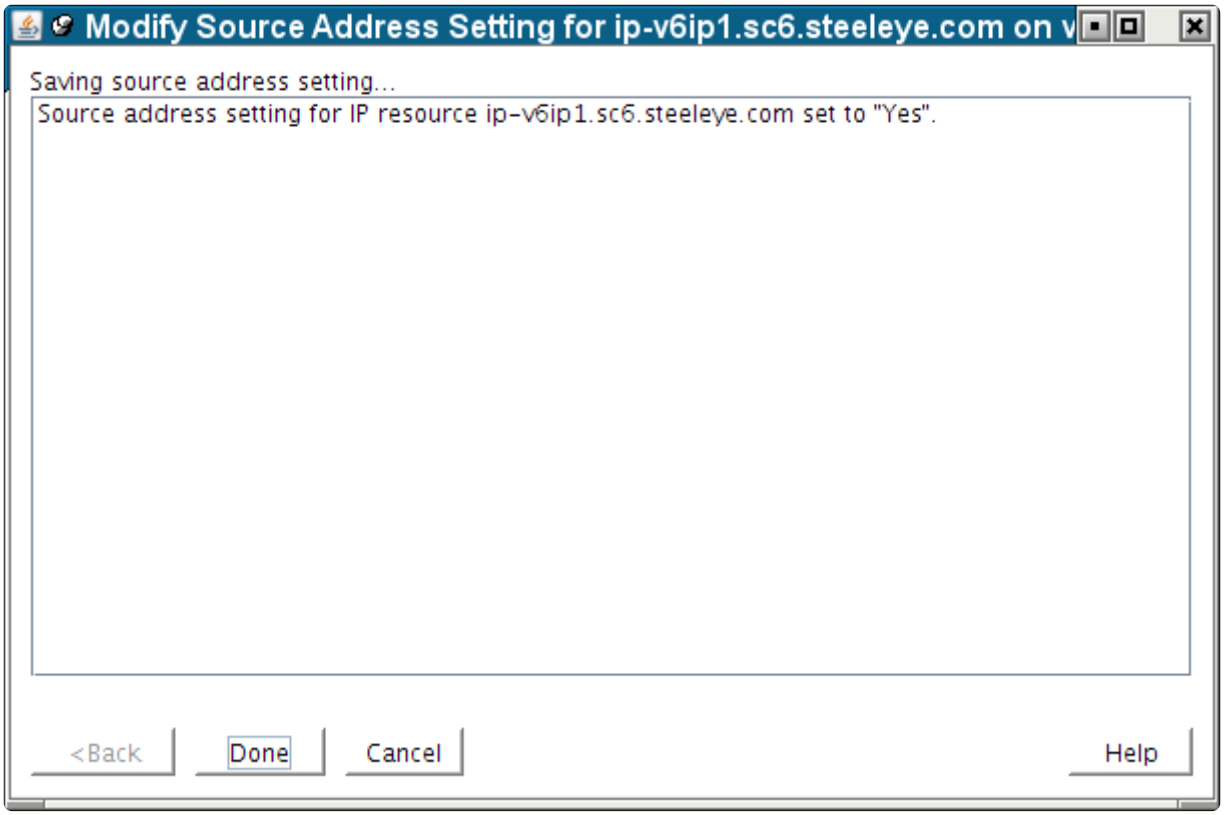



設定を変更するにはドロップダウンリストを選択し、 **[Yes]** または **[No]** を選択してください。

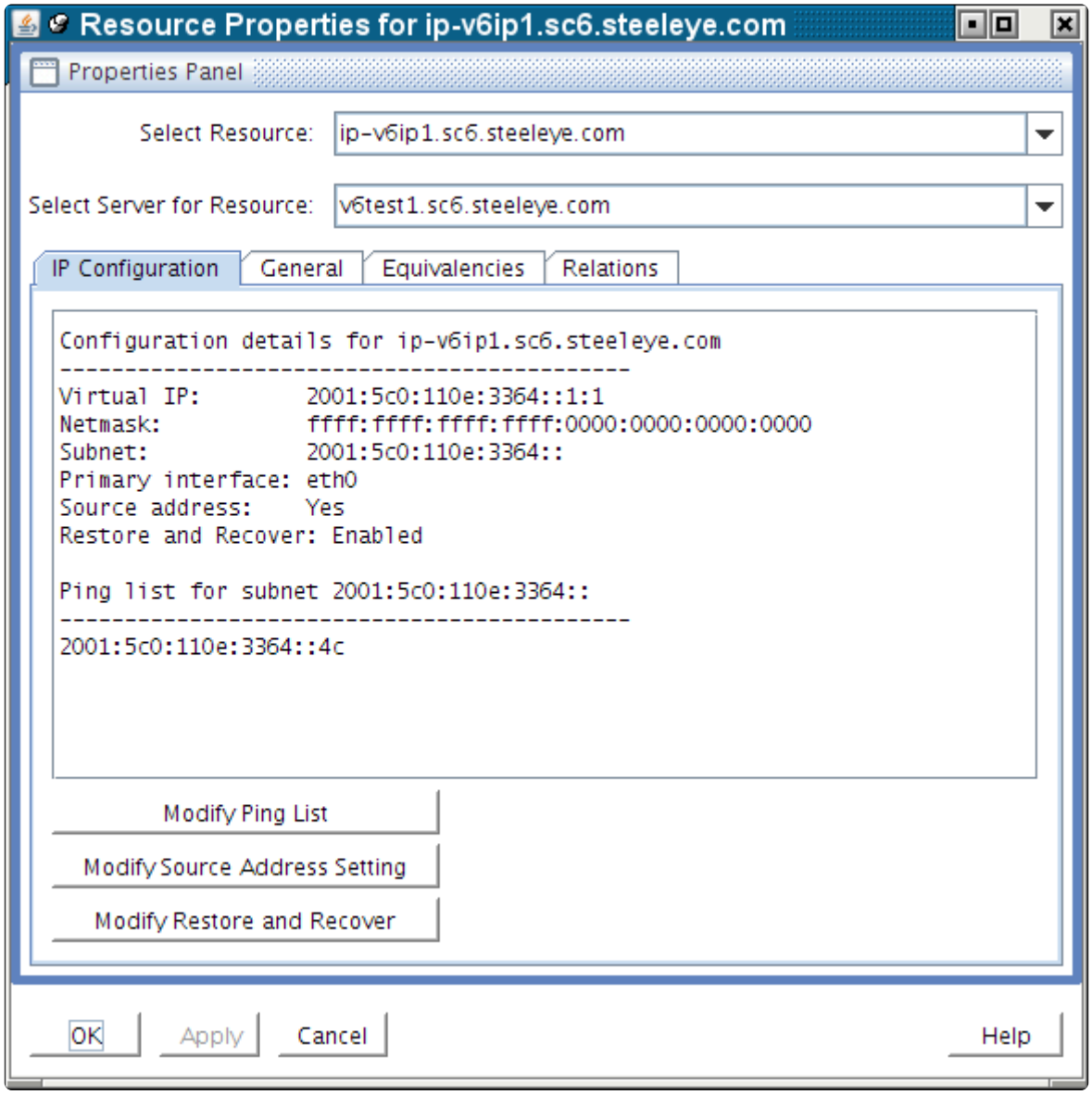





**[Apply]** をクリックし、変更を保存します。以下の確認ウィンドウが表示されます。




**[Done]** をクリックするとウィンドウが閉じ、 **[IP Configuration]** プロパティ画面に戻ります。ここで、変更した設定を確認できます。




## ソースアドレス設定に関する重要事項

IP リソースの **[Source Address]** 設定は、システムごとに個別の設定となります。 **[Source Address]** 設定の変更後に IP リソースを他のシステムに拡張する場合、その設定が拡張先の LifeKeeper システムに一緒にコピーされます。しかし、すでに IP リソースが拡張済みの場合は、IP リソースが定義されている各システム上で個別に **[Source Address]** 設定を変更する必要があります。

特定のサブネット内で、[Source Address] 設定を [Yes] にした場合、単一の IP リソースのみ設定が有効になります。外向けネットワークトラフィックのソースアドレスとして動作する IP アドレスは本来 1 つだけに限定されるためです。同じサブネットに属する複数の IP リソースが存在し、[Source Address] 設定が [Yes] になっている場合、最後に設定したリソースの IP アドレスが他の有効な送信トラフィックのソースアドレスとなり、その他のソースアドレス設定は無効になります。

**[Source Address]** 設定は、IP リソースの起動時のローカル TCP/IP 設定のみに影響します。したがってすでにリソースが起動中の場合は、ソースアドレス設定を変更後、リソースを一旦停止し、再起動させて、変更を TCP/IP 設定に反映させる必要があります。

**[Source Address]** 設定は、IP v4 アドレスにのみ影響があります。IP v6 アドレスには影響ありません。

# リストアおよびリカバリの修正

IP リソースに対する **[Restore and Recover]** の設定は、異なるサブネット間にディザスタリカバリシステムが存在するマルチサイトクラスタ環境では [Disable] にする必要があります。

この設定では既存の IP リソースに対するデフォルトのリストアおよびリカバリの動作を **[Enable]** もしくは **[Disable]** で選択することができます。 **[Enable]** オプションに設定すると、IP アドレスはアクティブになり、通常の監視およびリカバリプロセスが発生します。IP アドレスのリストアの現在のデフォルトの動作は **[Enable]** になっています。

**[Restore and Recover]** オプションを **[Disable]** に設定すると、リソースはネットワークもしくはネットワークアダプタ上で in service の状態になりますが、IP アドレスはアクティブになりません。この設定によりマルチサイトクラスタ (もしくは WAN) 環境での階層をディザスタリカバリシステム上で in service にすることが可能になります。

この設定はリソースを作成し拡張した後に選択することが可能です。

**アクティブな IP アドレス (ISP) に対する重要事項:** ISP およびアクティブな IP アドレスにおいて動作を **[Disable]** に設定しても、アクティブな IP アドレスは out of service になりません。

# 6.7. MySQL Recovery Kit 管理ガイド

## MySQL Recovery Kit のドキュメンテーション

SIOS Protection Suite for Linux MySQL Recovery Kit を使用すると、MySQL のリソースとデータベースのための LifeKeeper の障害回復保護機能を簡単に追加できます。これにより、プライマリデータベースサーバで障害が発生しても、わずかな時間で人手を介さずに指定されたバックアップサーバにて復旧することができます。

### SIOS Protection Suite のドキュメンテーション

以下の SPS 製品ドキュメンテーションを SIOS Technology Corp から取得可能です。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション (LifeKeeper GUI の **[Help]** メニューからもアクセス可能)

本ドキュメンテーションおよびオプションのリカバリキットに関連するドキュメンテーションは SIOS テクニカルドキュメンテーション の Web サイトで入手可能です。

# 6.7.1. 要件

## キットのハードウェア / ソフトウェア要件

リカバリソフトウェアをインストールおよびセットアップするには、サーバが一定のハードウェア / ソフトウェア要件を満たしている必要があります。LifeKeeper MySQL Recovery Kit をインストールまたは削除する具体的な方法については、SPS for Linux インストールガイド を参照してください。

設定が以下の要件を満たしていることを確認してください。

- **サーバ** 。この Recovery Kit には、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート で説明されている要件に従って設定した LifeKeeper 対応の 2 台以上のコンピュータが必要です。
- **LifeKeeper ソフトウェア** 。各サーバに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。具体的な LifeKeeper の要件については、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート を参照してください。
- **LifeKeeper IP Recovery Kit** 。リモートクライアントが MySQL データベースにアクセスする場合に、このキットが必要になります。同じバージョンの Recovery Kit を各サーバにインストールする必要があります。
- **IP ネットワークインターフェース** 。各サーバは、イーサーネット TCP/IP をサポートするネットワークインターフェースを 1 つ以上必要とします。IP スイッチオーバが正しく動作するには、ローカルネットワークに接続されているユーザシステムが標準の TCP/IP 仕様に準拠している必要があります。

> 注記: 各サーバが必要とするネットワークインターフェースが 1 つだけであっても、heterogeneous な要件、スループット要件、単一障害点の排除、ネットワークのセグメンテーションといった多くの理由のために、複数のインターフェースを使用すべきです。

- **TCP/IP ソフトウェア** 。各サーバは TCP/IP ソフトウェアも必要とします。
- **MySQL ソフトウェア** 。各サーバは、LifeKeeper と LifeKeeper MySQL Recovery Kit の設定に先だって MySQL ソフトウェアをインストールして設定しておく必要があります。各サーバには同じバージョンをインストールする必要があります。最新のリリースの互換性および購入方法については、SPS for Linux リリースノート を参照するか、営業担当者にお問い合わせください。

# 6.7.2. 設定上の考慮事項

本セクションでは、典型的な LifeKeeper MySQL 設定の定義および例と、MySQL の設定を開始する前に検討すべき情報について説明します。

LifeKeeper Core リソース階層の設定方法については、SPS for Linux テクニカルドキュメンテーション の リソース階層 セクションを参照してください。

---

MySQL の設定上の考慮事項

クライアント設定の考慮事項

設定の要件

設定例

アクティブ / スタンバイ設定

アクティブ / アクティブ設定

複数データベースサーバ環境に関する考慮事項

LifeKeeper で mysqld グループを使用する

Network Attached Storage の使用

systemd 環境で使用する場合の考慮事項

# 6.7.2.1. MySQL の設定上の考慮事項

LifeKeeper MySQL 環境に関して、次のような特有の考慮事項について検討する必要があります。

プライマリサーバーとバックアップサーバーで MySQL データベースサービスを運用するには、ファイルシステムとディスクパーティションが各サーバーからアクセスできなければなりません。MySQL Recovery Kit の設定を開始する前に、必ず次の準備手順を終了し、各サーバー上のデータベースをテストして稼働させる必要があります。以下の手順では、ユーザー「mysql」は MySQL サーバーを開始するオペレーティングシステムユーザーを表します。

1. すべてのサーバーに MySQL サーバーとクライアントコンポーネントをインストールします。必ず、すべてのサーバーで同じバーションの MySQL クライアントコンポーネントとサーバーコンポーネントが稼働するようにします。MySQL 実行可能ファイルは、ローカルまたは共有ドライブ上に置くことができます。

2. LifeKeeper が保護する MySQL データベースサーバーを稼働させたいソケットおよび / またはポート上で mysqld が稼働しているサーバーがあれば、`mysqladmin` コマンドを使用して各 MySQL サーバーを停止します。

3. MySQL データディレクトリの内容を共有の場所に移動します。デフォルトで、MySQL データディレクトリはローカルドライブにインストールされます。この場所は、ディストリビューションのメカニズムによって異なります。バイナリ RPM は、データディレクトリを `/var/lib/mysql` にインストールします(必ず内容のみを移動して、ディレクトリはそのまま残してください。これにより、MySQL データベースサーバーが必要に応じてこのディレクトリ内にログを書き込むことができます。ステップ 4 で説明されている「mysql」ユーザーが、この場所にログを書き込む権限を持っていることを確認します)。

4. インストール処理で Linux ユーザー「mysql」が作成されなかった場合は、このユーザーを作成します。セキュリティ上の理由から、MySQL サーバーを「root」として実行しないでください(セキュリティ問題の詳細については、『 *MySQL Administration Guide* 』を参照してください)。必ず、「mysql」ユーザーのみがデータベースディレクトリへの読み取り/書き込み権限を持つようにします。「mysql」ユーザーおよびグループはすべてのサーバー上に作成する必要があります。ユーザー ID とグループ ID は、すべてのサーバーで同じでなければなりません。

5. **重要:** OS 自動起動によって開始されるサーバーは、LifeKeeper の保護下には入れません。さらに、このサーバーは LifeKeeper の保護下にあるサーバーと同じポート番号またはソケットを使用できません。

6. ソケットを共有ディスク上のデータディレクトリに書き込むことを推奨します。ソケットをローカルディスクに書き込む場合、階層が存在するすべての LifeKeeper サーバー上の同じパスに存在することを確認します。「mysql」ユーザーが、この場所にソケットを書き込む権限を持っていることを確認します。

7. 設定に適した mysql デーモンスタートアップコマンドを使用して MySQL サーバーを開始します。*my.cnf* ファイルで単一インスタンスを定義している設定の場合は、以下のコマンドを使用してください。

```
<start command> --user=mysql --socket=<socket> --port=<port number> --dat
```

```
adir=<path to the data directory> --log &
```

<start command> は、mysql バージョン 3.x では `safe_mysqld`、バージョン 4.x では `mysqld_safe` です。

*my.cnf* ファイルで mysqld グループを使用した設定の場合は、以下のコマンドを使用してください。

```
mysqld_multi start <group number>
```

<group number> は mysqld グループの *my.cnf* ファイルで定義したインスタンス番号を表しています。LifeKeeper で mysqld グループを使用する際の詳細については、LifeKeeper で mysqldGroup を使用するを参照してください。

Systemd を採用しているディストリビューションで MySQL（5.7.6 以降）が Systemd を使用するように設定されている場合、systemctl コマンドを利用する必要があります。詳しくは Systemd 環境で使用する場合の考慮事項 を参照してください。

8. 「mysql」という名前で MySQL データベースユーザーを作成します。このユーザーにパスワードを指定し、「shutdown」権限を付与します。これは、1 台のサーバー上でのみ行います(ユーザーの作成と権限付与の詳細については、『 *MySQL Administration Guide* 』を参照してください)。

9. サンプル設定ファイル *my.cnf* を必要な場所( */etc* もしくは */<datadir>* )にコピーします。このファイルには、データベースサーバー用とクライアントプログラム用のオプションが含まれています。

   ファイルは、 *MySQL* データ ディレクトリまたは /etc ディレクトリのどちらかに置くことが可能です。 */etc/my.cnf* ファイルにはグローバルオプションが含まれます。マシン上で常に 1 つのデータベースのみが稼働する場合(つまりアクティブ / スタンバイ設定)、もしくは mysqld グループ機能を使用している場合は */etc* に *my.cnf* ファイルを置きます。(LifeKeeper で mysqld グループを使用する を参照してください。)ファイルが */etc* にある場合、そのファイルを各 LifeKeeper バックアップサーバーにコピーする必要があります。データディレクトリ内の *my.cnf* ファイルには、サーバー固有のオプションが含まれている必要があります。複数サーバーおよびアクティブ / アクティブ設定の場合、mysqld グループ機能を使用しない限りはこのファイルをリソースインスタンスごとにデータディレクトリに格納する必要があります。(LifeKeeper で mysqld グループを使用する を参照してください。)

   **注記:** サーバー固有のオプションを含む *my.cnf* ファイルは /etc ディレクトリと */<datadir>* ディレクトリの両方に存在してはいけません。 もし、サーバー固有のオプションを含む *my.cnf* ファイルが /etc ディレクトリに存在し、同時に、保護された *my.cnf* ファイルが */<datadir>* にインストールされている場合、コンフリクトの原因となる場合があります。詳細は MySQL のドキュメントのグローバルオプションとサーバー固有のオプションの設定を参照してください。

   以下のエントリを追加または編集します。

   a. ファイルの「client」セクションに、接続に使用するユーザーとパスワードを指定します。

      ```
      [client]
      user =clientuser
      ```

```
password =password
.
.
.
```

b. ファイルの「mysqld」セクションには、接続に使用するソケットとポートを、mysqld プロセスに対する pid ファイルの場所とともに指定します。ユーザー変数では、mysqld プロセスを開始するオペレーティングシステムユーザーを指定します。

```
[mysqld]
socket =/home1/test/mysql/mysql.sock
port =3307
pid-file =/home1/test/mysql/mysqld.pid
user =osuser
```

**注記:** このファイルが正しく保護され、ユーザー「mysql」によって所有されていることを確認します。

**注記:** MySQL 階層を作成した後、`my.cnf` ファイル内の情報を変更する必要がある場合は、変更を加える前に階層を out-of-service (つまり OSU 状態)にして mysql サーバーインスタンスを停止します。

**注記:** 上記の `my.cnf` 設定の例は、単一のデータベースインスタンス `mysqld` について説明しています。mysqld グループを使用した設定例については LifeKeeper で mysqld グループを使用する を参照してください。

**注記:** include文はサポートしていません。単一の`my.cnf`に全ての設定を記述してください。

# 6.7.2.2. クライアント設定の考慮事項

MySQL データベースクライアントを設定する上で、次のような考慮事項があります。

- クライアントがリモートホストから接続する場合、クライアントが接続に使用する LifeKeeper 保護下の IP アドレスを 1 つ作成します。
- クライアントは、LifeKeeper が保護する IP アドレス経由でデータベースサーバに接続するよう設定する必要があります。
- クライアントがドメイン名を使用して接続する場合は、各クライアントの hosts ファイル内に保護する IP アドレスのエントリを作成するか、DNS 内に名前を設定します。クラスタ内のすべてのクライアントおよびすべての LifeKeeper サーバから ping を実行して、保護する IP アドレスをテストします。
- 各ユーザは、それぞれのマシンのホームディレクトリに *my.cnf* ファイルを持つことができますが、L-ifeKeeper は */etc* ディレクトリまたはデータディレクトリにある *my.cnf* ファイルのみを使用します。 *my.cnf* ファイルにはクライアント接続情報(つまり、ポート、ソケット ID、ユーザ、パスワード)が格納されています。

# 6.7.2.3. 設定の要件

それぞれの例では、1 つまたは 2 つのデータベースインスタンス (databaseA、databaseB) が使用されています。データベースタグ名は、これらのデータベースインスタンスを LifeKeeper に記述するための任意の名前です。分かりやすい名前を指定することをお勧めします。mysqld グループを使用した設定の LifeKeeper が提示するデフォルトのタグ名は mysql または mysql<group number> です (LifeKeeper で mysqld グループを使用する を参照)。設定例を理解するには、次のような設定要件を留意してください。

- **LifeKeeper 階層。** LifeKeeper の管理を実行するときに、主要な階層は、管理するサーバ上に作成されている階層を参照します。設定の表については、最初の管理画面で入力される情報は Server 1 の観点からのものです。2 番目の画面が表示されると、2 台目のサーバを管理するときに作成される階層を参照します。設定例では、2 台目のサーバは Server 2 です。

- **1 台のサーバによってロックされる共有ディスク。** LifeKeeper を使用するとき、LifeKeeper の保護下にある共有ストレージリソースは、1 台のサーバが使用するためにリザーブしています。これは SCSI リザベーションを使用して行われます。共有デバイスがディスクアレイの場合、LUN 全体がリザーブされます。共有デバイスがディスクの場合は、ディスク全体がリザーブされます。こうすることで、クラスタ内の他のサーバが不用意にデータを破壊してしまうことを避けられます。サーバに障害が発生すると、優先順位が最も高いバックアップサーバがそれまでのリザベーションを解除して新たにリザベーションを行い、他のサーバをすべてロックします。

- **共有ディスク上のデータディレクトリ。** LifeKeeper MySQL Recovery Kit が正しく機能するためには、データベースインスタンスのデータディレクトリ (datadir) が常に共有ディスク上にある必要があります。データディレクトリは、ファイルシステム上になければなりません。このファイルシステムは、プライマリサーバおよびバックアップサーバの両方からマウントできなければなりません。データディレクトリ (datadir) はレプリケーションされたディスクもしくは Network Attached Storage (NAS) 上に存在することが可能です。

# 6.7.2.4. 設定例

本セクションの例は、MySQL データベースインスタンスの設定方法を示しています。各図は、設定の種類と MySQL パラメータの関係を示しています。また、各設定は、本書で説明している設定の規則と要件に従って、MySQL 設定と LifeKeeper ソフトウェアとの互換性を確保しています。

本セクションでは、設定の要件を説明し、次にそれぞれの設定例を示します。

- アクティブ / スタンバイ
- アクティブ / アクティブ

本セクションに挙げたものは可能な設定の例にすぎませんが、これらの設定を理解し、設定規則に従うことにより、使用するコンピューティング環境で実現可能なソリューションを明確にし設定する際に役立ちます。

設定の要件

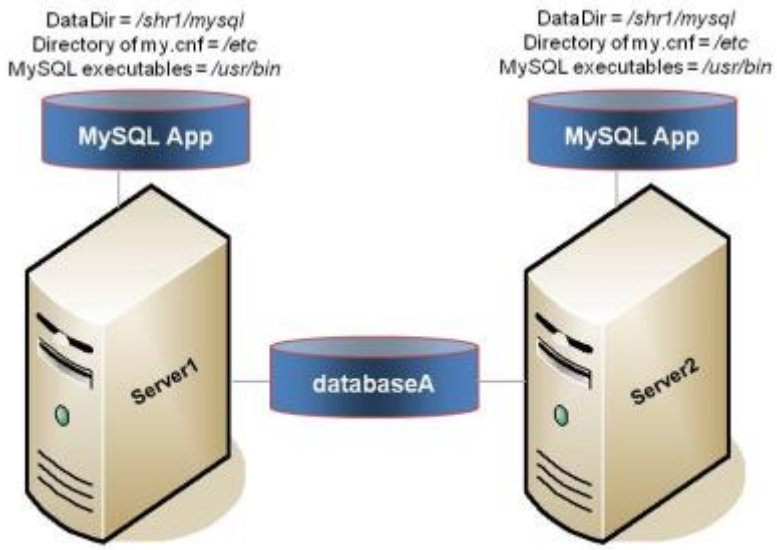
例 1 - アクティブ / スタンバイ設定

例 2 - アクティブ / アクティブ設定

# 6.7.2.5. アクティブ / スタンバイ設定

本セクションでは、アクティブ / スタンバイ設定の例を示します。この設定では、Server 1 がデータベースに排他的にアクセスしているため、Server 1 がアクティブであると見なされます。Server 2 は別の処理を実行します。Server 1 に障害が発生すると、Server 2 がデータベースにアクセス可能になり、LifeKeeper はデータベース操作を再確立します。

**図1.アクティブ / スタンバイ設定、例 1**




**設定に関する注記:**

- どちらのサーバも共有ディスク上の MySQL データディレクトリ(データベース databaseA が格納されている)を使用します。
- MySQL データディレクトリへのパスはどちらのサーバでも同じです。
- *my.cnf* 設定ファイルはローカルディスクの */etc* にあります。
- MySQL 実行ファイルは各サーバのローカルドライブの */usr/bin* にあります。
- Server 2 は、Server 1 がアクティブである間は、共有ディスク上のファイルとディレクトリにアクセスできません。

**Server 1 でのリソース階層の作成**

| | |
|---|---|
| サーバ: | Server1 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */usr/bin* |

| | |
|---|---|
| データベースタグ: | mysql-on-server1 |

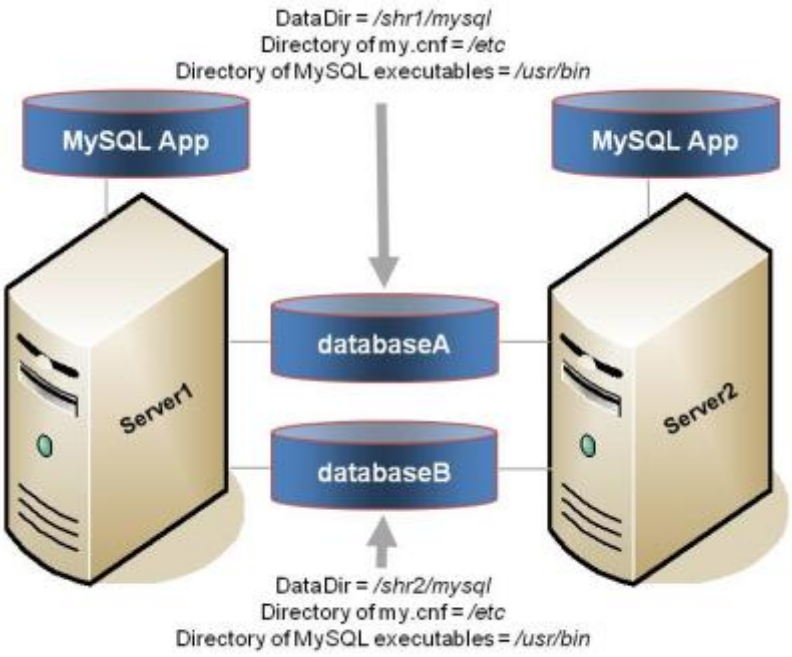
リソース階層の **Server 2** への拡張

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | mysql-on-server1 |
| ターゲットサーバ: | Server2 |
| ターゲットの優先順位: | 10 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */usr/bin* |
| データベースタグ: | mysql-on-server2 |

# 6.7.2.6. アクティブ / アクティブ設定

アクティブ / アクティブ設定は、異なるデータベースインスタンスを実行し、互いにバックアップの役割を果たす 2 台以上のサーバで構成されます。各データベースインスタンスは、別々の共有物理ディスク上に置かれていなければなりません。複数の MySQL データーベースインスタンス (同じバージョンもしくは異なるバージョン) をサポートする LifeKeeper の設定のために、SIOS では mysqld グループ 機能をサポートするバージョンの MySQL を使用することを推奨しています。これらの設定では *my.cnf* 設定ファイルは */etc* に存在します。mysqld グループ機能をサポートしないバージョンの MySQL では、 *my.cnf* 設定ファイルは各データベースインスタンスの共有ファイルシステムの MySQL データディレクトリに存在している必要があります(例: 下記図 2 では */shr1/mysql* および */shr2/mysql*)。

**図 2.アクティブ / アクティブ設定、例 2**




設定に関する注記:

- 各サーバは異なる共有ディスク上にある異なる MySQL データディレクトリ(データベースインスタンス database A と database B を含む)を使用します。
- MySQL データディレクトリへのパスは、サーバ上に定義されているインスタンスごとに異なります。
- *my.cnf* 設定ファイルは */etc* に配置され、各データベースインスタンスの mysqld グループセクションを含んでいます。各セクションはデータベースインスタンスに対する一意の MySQL データディレクトリ、ポート、ソケットを定義します。 *my.cnf* 設定ファイルはクラスタの全ノードで常に同一の内容であることに留意してください。mysqld グループをサポートしないバージョンの MySQL が起動しているシステムでは、各データベースインスタンスに対する *my.cnf* 設定ファイルは共有ドライブ上のデータディレクトリに配置されています。各設定ファイルはデータベースインスタンスに対する一意の MySQL データディレクトリ、ポート、ソケット定義を定義します。

- MySQL 実行ファイルは各サーバのローカルドライブの */usr/bin* にあります。
- 最初は、Server 1 で databaseA が、Server 2 で databaseB が稼動します。スイッチオーバが発生した場合は、1 台のサーバで両方のデータベースが稼動できます。

**Server 1** での **1** つ目のリソース階層の作成

| | |
|---|---|
| サーバ: | Server1 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */usr/bin* |
| データベースタグ: | mysql-shared.example.instance1 |

**1** つ目のリソース階層の **Server 2** への拡張

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | mysql-shared.example.instance1 |
| ターゲットサーバ: | Server2 |
| ターゲットの優先順位: | 10 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */usr/bin* |
| データベースタグ: | mysql-shared.example.instance1 |

**Server 2** での **2** つ目のリソース階層の作成**:**

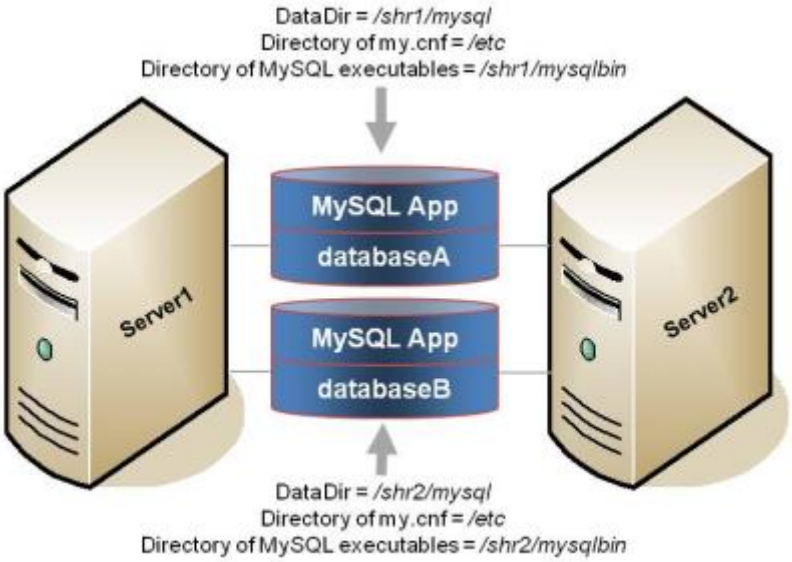
| | |
|---|---|
| サーバ: | Server2 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */usr/bin* |
| データベースタグ: | mysql-shared.example.instance2 |

**2** つ目のリソース階層の **Server 1** への拡張**:**

| | |
|---|---|
| テンプレートサーバ: | Server2 |

| | |
|---|---|
| 拡張するタグ: | mysql-shared.example.instance2 |
| ターゲットサーバ: | Server1 |
| ターゲットの優先順位: | 10 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */usr/bin* |
| データベースタグ: | mysql-shared.example.instance2 |

**図 3.アクティブ / アクティブ設定、例 2**




設定に関する注意:

- 各サーバは異なる共有ディスク上にある異なる MySQL データディレクトリ(データベースインスタンス database A と database B を含む)を使用します。
- MySQL データディレクトリへのパスは、サーバ上に定義されているデータベースインスタンスごとに異なります。
- *my.cnf* 設定ファイルは */etc* に配置され、各データベースインスタンスの mysqld グループセクションを含んでいます。各セクションはデータベースインスタンスに対する一意の MySQL データディレクトリ、ポート、ソケットを定義します。 my.cnf 設定ファイルはクラスタの全ノードで常に同一の内容であることを留意してください。mysqld グループをサポートしないバージョンの MySQL が起動しているシステムでは、各データベースインスタンスに対する my.cnf 設定ファイルがデータベースのデータディレクトリの共有ドライブ上に配置されています。各設定ファイルはデータベースインスタンスに対する一意の MySQL データディレクトリ、ポート、ソケットを定義します。
- データディレクトリを含む共有ディスクのそれぞれに MySQL 実行可能ファイルのコピーがありま

す。

- 最初は、Server 1 で databaseA が、Server 2 で databaseB が稼動します。スイッチオーバが発生した場合は、1 台のサーバで両方のデータベースが稼動します。

**Server 1 での 1 つ目のリソース階層の作成**

| | |
|---|---|
| サーバ: | Server1 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */shr1/mysqlbin* |
| データベースタグ: | mysql-shared.example.instance1 |

**1 つ目のリソース階層の Server 2 への拡張:**

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | mysql-shared.example.instance1 |
| ターゲットサーバ: | Server2 |
| ターゲットの優先順位: | 10 |
| my.cnf ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */shr1/mysqlbin* |
| データベースタグ: | mysql-shared.example.instance1 |

**Server 2 での 2 つ目のリソース階層の作成:**

| | |
|---|---|
| サーバ: | Server2 |
| my.cnf ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */shr2/mysqlbin* |
| データベースタグ: | mysql-shared.example.instance2 |

**2 つ目のリソース階層の Server 1 への拡張**

| | |
|---|---|
| テンプレートサーバ: | Server2 |
| 拡張するタグ: | mysql-shared.example.instance2 |

| | |
|---|---|
| ターゲットサーバ: | Server1 |
| ターゲットの優先順位: | 10 |
| *my.cnf* ファイルがあるディレクトリ: | */etc* |
| MySQL 実行可能ファイルがあるディレクトリ: | */shr2/mysqlbin* |
| データベースタグ: | mysql-shared.example.instance2 |

# 6.7.2.7. 複数データベースサーバ環境に関する考慮事項

複数の MySQL データベースサーバおよびデータベースがある場合、次のような設定上の考慮事項があります。

- アクティブ / アクティブまたは複数の(同一もしくは異なるバージョンの) MySQL インスタンスを稼動させる場合、可能であれば mysqld グループの機能を使用することを考慮してください。SIOS は、複数の MySQL データベースサーバ構成に対して mysqld グループ (mysqld_multi) を使用することを推奨します。

- アクティブ / アクティブまたは複数の MySQL インスタンスを稼働させる場合、共有ファイルシステムを */var/lib/mysql* としてマウントしないでください。mysql 起動コマンド (`safe_mysqld` or `mysqld_safe`)により、MySQL サーバが予期せずシャットダウンされるためです。

- mysqld グループの機能を使用しない場合、 *my.cnf* ファイルは、各アクティブ/アクティブもしくは複数サーバのデータディレクトリに保存する必要があります。mysqld グループを使用する構成では、 *my.cnf* ファイルはデータディレクトリではなく */etc* に保存されます。LifeKeeper および mysqld グループ機能に関する情報は LifeKeeper で mysqld グループを使用する を参照してください。

- MySQL 用の追加のポート番号は、 */etc/services* ファイルに指定する必要があります。

- 各 MySQL データベースサーバは、個別のポート上で稼働し、個別のソケットファイルにアクセスするように設定する必要があります。これらの設定オプションは、データディレクトリにある *my.cnf* ファイル内に指定されます。

- 各サーバは、個別の共有の位置からデータにアクセスするように設定する必要があります(つまり、各サーバは個別のデータディレクトリを使用する必要があります)。

# 6.7.2.8. LifeKeeper で mysqld グループを使用する

MySQL Application Recovery Kit は、mysqld_multi を介して管理される mysqld グループ機能を使用した *my.cnf* ファイルをサポートします。この MySQL の機能を使用すると、単一の my.cnf ファイル (通常 */etc* に保存される) を介して、容易に複数の MySQL インスタンスを設定できるようになります。本キットは mysqld グループのフォーマットを利用している my.cnf ファイルを検出し、管理者に対して保護対象とする mysqld グループの番号を選択することを要求します。管理者が選択できる選択リストは、my.cnf ファイルで定義されているグループ番号から、すでに本キットにより保護されているグループ番号を差し引いたものから決定します。

一般的には、mysqld グループ機能を使用して複数の MySQL インスタンスを設定および制御するほうが容易になります。SIOS は、アクティブ / アクティブもしくは複数インスタンスの設定時には、このアプローチを使用することを推奨します。

## my.cnf ファイル

mysqld グループ機能を使用する場合、以下が必須となります。

a. 単一の my.cnf ファイルが、データベースインスタンスの mysqld グループを定義するために使用される必要があります。

b. my.cnf ファイルは共有ストレージ上に配置しないでください。

c. 各クラスタノードに my.cnf ファイルの完全なコピーが必要です (*/etc/my.cnf* が望ましい)。

d. my.cnf ファイルに対して行ったすべての変更は、LifeKeper のすべてのクラスタノードに反映される必要があります。

Recovery Kit は、my.cnf ファイルが mysqld グループを使用していることを検知すると mysqld_multi コマンドを使用します。これに基づいて、LifeKeeper の制御下に配置する前に mysqld_multi を使用して MySQL インスタンスをテストすることができます。

以下は、mysqld_multi によって制御されている 2 つのデータベースインスタンスについて mysqld グループを使用した比較的複雑な my.cnf ファイルの解説です。`mysqld_multi` コマンド (および MySQL LifeKeeper Recovery Kit) は、管理者にさまざまな設定を行うためのオプションを提供しています。下記の例、`[mysqld1]` では、さまざまな MySQL コマンドの大部分についてデフォルトのロケーションを使用する比較的単純な MySQL インスタンスを定義しています。その次の例、`[mysqld55]`では、より多くの変更を行っています。コメントは各セクションにおいて LifKeeper がどのように MySQL と相互作用しているかについて説明しているので参考にしてください。

```
#The following client section defines which username/password combination wil
l be used for
#LifeKeeper  connections. The username/password combination needs to be define
d in each MySQL
```

```
# Database instance that will be described in this my.cnf file.
[client]
user     =  steeleye
password =  password
```

```
# This next section describes the default version of mysqld and mysqldadmin th
at mysqld_multi
# will use when processing mysqld_multi commands. The username/password combo
defines the
# MySQL account that mysqld_multi will use when working with the database inst
ances. This
# username and password combo needs to be defined in each MySQL Database insta
nce that will be
# controlled by mysqld_multi. See how to set up the multi_admin account in th
e MySQL Reference
# Manual, by issuing "mysqld_multi --example".
[mysqld_multi]
mysqld     = /usr/bin/mysqld_safe
mysqladmin = /usr/bin/mysqladmin
user       = multi_admin
password   = password
```

```
# The next section defines the first of two MySQL Database instances in this m
y.cnf file. Note
# that each section starts  with a [mysqldNN] where NN is the mysqld group num
ber (or instance).
# Each group name must have a number. There are a  number of directives that t
he LifeKeeper MySQL
# Recovery Kit will be looking for in these sections.
[mysqld1]
datadir  = /s11/mysql-data5077       #Defines where the data files for the ins
tance will live. For
                                     # LifeKeeper, this directory must be on L
ifeKeeper protected
                                     #   (shared or replicated) storage.
mysqld   = /usr/bin/mysqld_safe       # Defines specifically which mysqld comm
and will be used for
                                      #   starting the instance. This one is u
sing the
                                      #   default  mysqld_safe that came with
the distribution.
socket=/s11/mysql-data5077/moe.socket # Defines the location of the socket fo
r this instance.
                                      #   If the socket is not on LifeKeeper p
rotected storage, it
```

```
                                        #   needs to be defined in exactly the s
ame place on each
                                        #   node in the cluster and be owned by
the "user" defined
                                        #   below.
port     =  3307                       # Each instance needs its own, unique TC
P/IP port.
pid-file = /var/run/mysqld/mysqld.pid # The pid-file can be on LifeKeeper prot
ected or
                                       #   non-LifeKeeper protected storage.
log-error= /var/log/mysqld.log         # Location of the MySQL error log for th
is instance. Can be
                                       #   on LifeKeeper protected or non-LifeKe
eper protected
                                       #   storage.
user     = mysql                       # The Linux user name that will run the
MySQL processes.
```

```
#The next section defines the more complicated of the two MySQL instances. Ins
tance "55" is not
#using the default MySQL that came with the Linux distribution as it is using
the 5.5.12 version
#of MySQL that was installed from source. The binaries for this version were i
nstalled onto shared
#storage, and the binary directory is LifeKeeper protected.
[mysqld55]
datadir = /s11/mysql-data5512           # Same as above; this instance uses a
different data
                                        #   directory, and  this directory is
on LifeKeeper
                                        #   protected storage.
mysqld  =/s11/mysql5512/bin/mysqld_safe # For this instance, a different versi
on of mysqld_safe
                                        #   is used; the  one that is include
d with 5.5.12.
socket=/s11/mysql-misc5512/larry.socket # This instance has the socket on Life
Keeper protected
                                        #   storage, but not  in the default l
ocation (datadir).
port     = 3308                         # This instance has a unique TCP/IP po
rt as well.
pid-file = /var/run/mysqld/mysqld55.pid # This instance's pid-file is not on L
ifeKeeper protected
                                        #   storage.
log-error = /var/log/mysqld55.log       # This instance's log-error (error lo
g) is not on
```

```
                                        #   LifeKeeper protected  storage.
log-bin    = /s11/mysql-log5512/larry   # The log-bin directive specifies wher
e the binary
                                        #   transaction logs are  located for
this instance.
                                        #   These logs must be on LifeKeeper
protected storage
                                        #   (the recovery kit will enforce thi
s). By default,
                                        #   these logs are in the datadir.
user       = mysql                      # The Linux user name that will run th
e MySQL processes.
```

マルチインスタンス用 [mysqld<N>] の設定とシングルインスタンス用 [mysqld] の設定を併記する場合は、シングルインスタンス用の設定を最後に記載してください。

例：

```
[mysqld1]
(mysqld1の設定)

[mysqld2]
(mysqld2の設定)

[mysqld55]
(mysqld55の設定)

[mysqld]
(シングルインスタンス用mysqldの設定)
```

## mysqld_multi コマンド

例として mysql コマンドを実行します。

```
# mysqld_multi start 1
```

my.cnf で[mysqld1]として定義した mysqld グループ 1 インスタンスを、依存するすべての LifeKeeper 保護リソースが LifeKeeper のいずれかのノードで In Service であると仮定します。

以下のコマンドを実行します。

```
# mysqld_multi report 1
```

このインスタンスの状態がレポートされます (例：running (起動) または not running (停止))。このインスタンスが起動すると、LifeKeeper でのリソース作成が容易になります。

mysqld_multi 形式の my.cnf ファイルの設定方法に関する情報については以下のコマンドを実行してください。

```
# mysqld_multi --example
```

# 6.7.2.9. Network Attached Storage の使用

クラスタストレージとして NFS ファイルサーバ (Network Attached Storage) を使用するために LifeKeeper を設定するにあたっていくつかの特別な考慮事項があります。

## NAS Recovery Kit の使用

オプションの Network Attached Storage (NAS) Recovery Kit は、LifeKeeper for Linux で NFS サーバを共有ストレージアレイとして使用する場合に必要になります。各クラスタノードに NAS Recovery Kit (およびライセンス) をインストールしてください。詳細については、NAS Recovery Kit のドキュメンテーションを参照してください。

## エラーメッセージ

MySQL と Network Attached Storage (NAS) を使用する際に、システムクラッシュにより My SQL インスタンスがフェイルオーバ後に再起動できないといった事象に遭遇する可能性があります。MySQL エラーログで、エラーの原因が分かります。

### MySQL 5.0

```
110523 22:10:58 mysqld started
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
110523 22:10:58 InnoDB: Retrying to lock the first data file
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
```

### MySQL 5.5

```
110524 10:52:20 InnoDB: The InnoDB memory heap is disabled
110524 10:52:20 InnoDB: Mutexes and rw_locks use GCC atomic builtins
110524 10:52:20 InnoDB: Compressed tables use zlib 1.2.3
110524 10:52:20 InnoDB: Initializing buffer pool, size = 128.0M
110524 10:52:20 InnoDB: Completed initialization of buffer pool
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
110524 10:52:20 InnoDB: Retrying to lock the first data file
InnoDB: Unable to lock ./ibdata1, error: 11
InnoDB: Check that you do not already have another mysqld process
InnoDB: using the same InnoDB data or log files.
```

これは LifeKeeper によって制御されている NFS ファイルシステム上で、MySQL mysqld プロセスが

*ibdata1* ファイルを NFS lock に設定しているということを示しています。システムクラッシュによりロックがクリアされなかったので、LifeKeeper は MySQL インスタンスを In Service の状態に戻すことができません。MySQL は何か別のプロセスが *ibdata1* ファイルを使用しているとみなします。

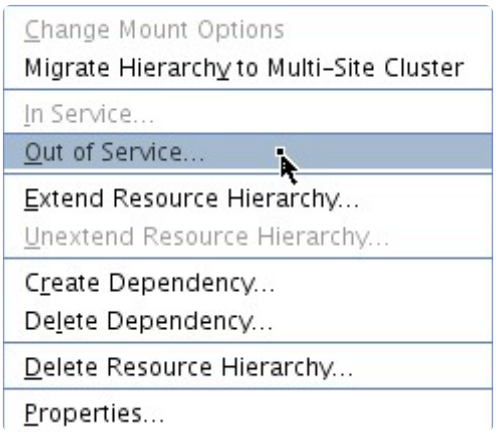
# 解決方法

この事象を解決するためには、ファイルシステムリソースを作成する前に「nolock」とよばれる NFS のオプションで *ibdata1* を格納する予定の NFS ファイルシステムをマウントしてください。デフォルトでは、NFS はファイルをロックさせるよう設定しています。「nolock」オプションがリソース作成前に使用されると、LifeKeeper はこのオプションを認識し、ファイルシステムリソースを In Service の状態にするたびにこのオプションを使用します。LifeKeeper は (クラスタノードから) ibdata1 を含んだファイルシステムへのアクセスを制御するので、通常、ロックは重要ではありません。テスト中に使用される NFS マウントオプションは「`rw,sync,tcp,nfsvers=3,nolock`」です。

MySQL のバイナリが格納されているファイルシステムといったような MySQL リソース階層によって使用されるその他のファイルシステム上で「nolock」を使用する必要はありません。

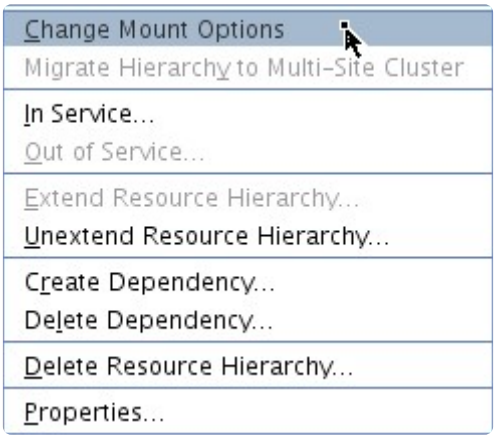
NAS ファイルシステムリソースがすでに「nolock」オプション設定なしに作成されている場合は以下の手順に従ってマウントオプションを変更してください。

1. LifeKeeper GUI を使用して変更する必要があるファイルシステムリソースに対して Out of Service 操作を行ってください。この操作は LifeKeeper GUI でファイルシステムリソース上にポインタを置き、右クリックしてドロップダウンメニューから **[Out of Service]** を選択することで可能です。この操作は親リソースも同じく Out of Service 状態にします。

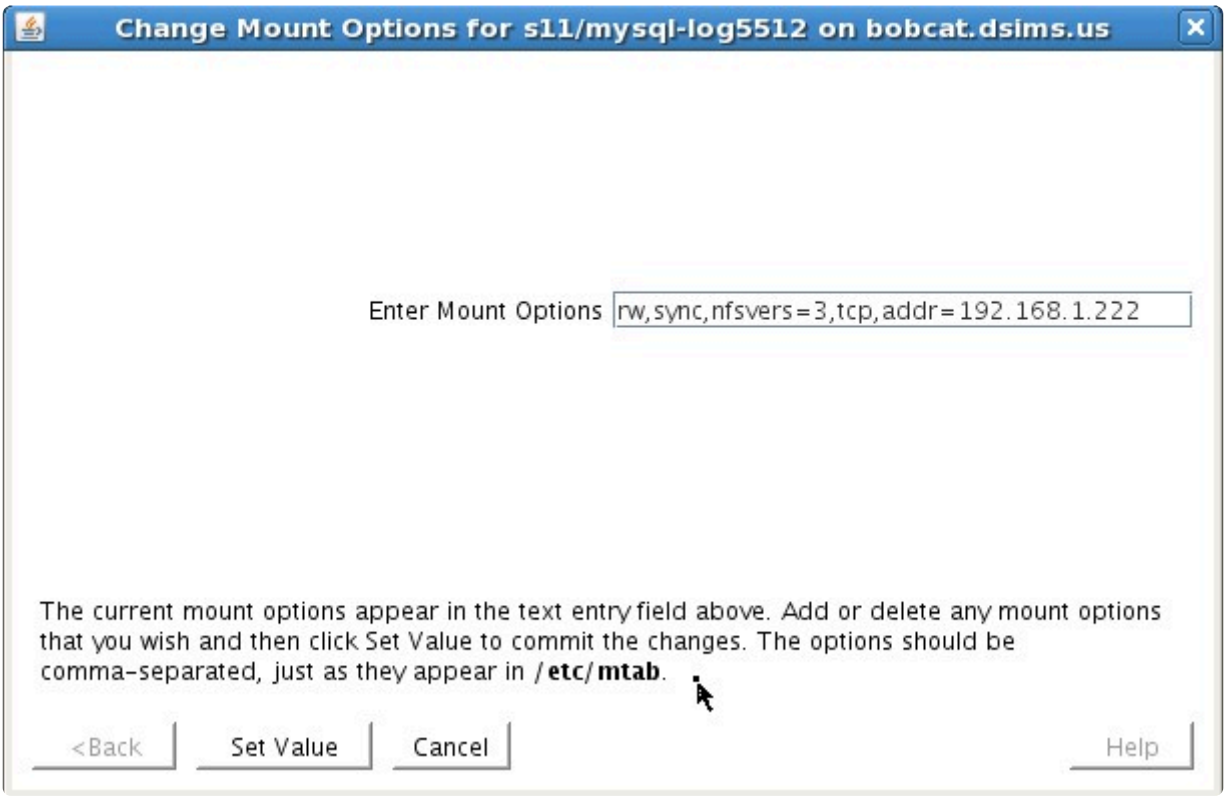



2. **[Out of Service]** 操作を実行し、プロセスを完了させてください。

3. 一旦ファイルシステムリソースが Out of Service 状態になったら、ポインタをリソース上に置き右クリックし、ドロップダウンメニューから **[Change Mount Options]** を選択してください。

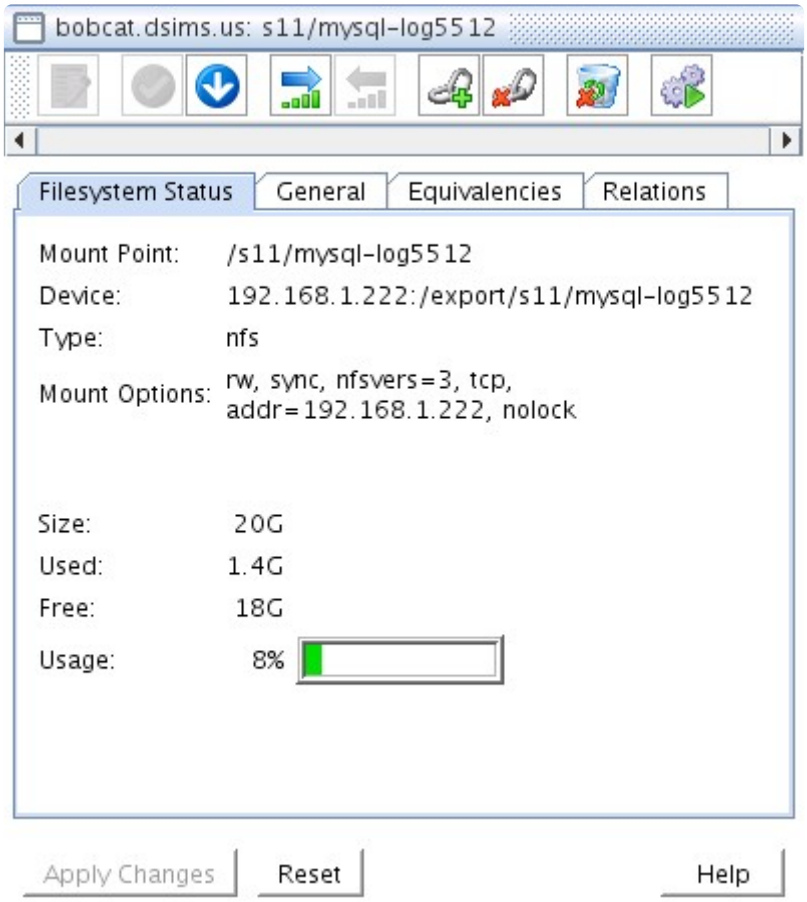



4. ポップアップウィンドウで、オプションの行に「**nolock**」を追加し、 **[Set Value]** をクリックしてください。クラスタの各ノードで手順 3 および 4 を繰り返す必要があります。




5. マウスを右クリックし、 **[In Service]** を選択し、NAS ファイルシステムリソースを In Service の状態に戻してください。

6. ファイルシステムリソースのプロパティパネルが、「nolock」が現在のマウントオプションであることを表示します。


bobcat.dsims.us: s11/mysql-log5512
Filesystem Status
General
Equivalencies
Relations
Mount Point: /s11/mysql-log5512
Device: 192.168.1.222:/export/s11/mysql-log5512
Type: nfs
Mount Options: rw, sync, nfsvers=3, tcp, addr=192.168.1.222, nolock
Size: 20G
Used: 1.4G
Free: 18G
Usage: 8%
Apply Changes
Reset
Help

# 6.7.2.10. systemd 環境で使用する場合の考慮事項

Systemd を採用しているディストリビューションで MySQL（5.7.6以降）が Systemd を使用するように設定されている場合、従来の mysqld_safe コマンドや mysqld_multi コマンドが使えません。これらのコマンドが使えない場合、LifeKeeper は systemctl コマンドを使って MySQL サービスの起動･停止を行います。従って、systemctl コマンドで起動･停止できる必要があります。

Managing MySQL Server with systemd を参考にして、MySQL の設定を行ってください。特に PID File の設定は、Systemd も LifeKeeper も必要とします。必ず Systemd の MySQL 設定と my.cnf の設定を同一にしてください。

リソース作成時は、以下の設定項目に注意してください。

| 設定項目 | 設定値 |
|---|---|
| *Location of my.cnf* | systemctl コマンドで MySQL サービスを起動する際に使用される my.cnf が格納されているディレクトリの絶対パス |
| *Location of MySQL executables* | mysqladmin コマンドがインストールされているディレクトリの絶対パス |

注記: "Location of my.cnf” の設定は、LifeKeeper 内部で my.cnf の設定を読み出すために使用されます。ここで設定した値が systemctl コマンドの起動で使われるわけではありません。デフォルトと異なるパスにある my.cnf を使う場合は、Systemd の MySQL 設定を正しく設定し、期待通り MySQL サービスが起動･停止できることを確認してください。
また、include 文はサポートしていません。単一の my.cnf に全ての設定を記述する必要があります。

# 6.7.3. インストール

## LifeKeeper での MySQL のインストール / 設定

---

LifeKeeper の設定作業

MySQL リソース階層の作成

リソース階層の削除

リソース階層の拡張

リソース階層の拡張解除

# 6.7.3.1. LifeKeeper の設定作業

設定作業は LifeKeeper GUI を使用して実行できます。以下の 4 つの作業は、MySQL リソースインスタンス特有のものであり、Recovery Kit ごとに異なるため、本セクションで説明しています。

- リソース階層の作成 。アプリケーションリソース階層を LifeKeeper クラスタに作成します。
- リソース階層の削除 。リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。
- リソース階層の拡張 。リソース階層をプライマリサーバからバックアップサーバへ拡張します。
- リソース階層の拡張解除 。リソース階層を LifeKeeper クラスタ内の 1 つのサーバから拡張解除 (削除) します。

以下の作業については、すべての Recovery Kit で同じ手順を使用する共通の作業であるため、 SPS for Linux テクニカルドキュメンテーション の管理 セクションを参照してください。

- リソース依存関係の作成 。既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除 。リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service リソース階層を特定のサーバで In Service の状態にします。
- Out of Service 。リソース階層を特定のサーバで Out of Service の状態にします。
- プロパティの表示 / 編集 。特定のサーバでリソース階層のプロパティを表示または編集します。

**注記:** このセクションの残りの部分では、LifeKeeper GUI の **[Edit]** メニューから作業を選択することによって、Recovery Kit を設定する方法を説明します。設定作業はツールバーから選択することもできます。状況表示ウィンドウの **[Resource Hierarchy Tree]** (左側のペイン)でグローバルリソースを右クリックし、ドロップダウンメニューに **[Edit]** メニューと同じ選択項目を表示することもできます。

状況表示ウィンドウの **[Resource Hierarchy Table]** (右側のペイン)でリソース・インスタンスを右クリックして、サーバおよび特定リソースの状態に応じて、リソース階層の作成を除くすべての設定作業を実行することもできます。

# 6.7.3.2. MySQL リソース階層の作成

> **重要:**
>
> MySQL データディレクトリファイル(datadir)が共有ディスク上にある LifeKeeper クラスタ環境では、共有ファイルシステムが primary/template サーバにマウントされる必要があります。ファイルシステムリソースが最初に作成される場合、共有ファイルシステムは必ず各サーバの同じマウントポイントにマウントします。また、重要事項として、リソースを作成するには稼働しているコミュニケーションパス(すなわちハートビート)が必須であることを念頭においてください。MySQL データディレクトリは、共有ディスク、レプリケーションされたディスク、もしくは Network Attached Storage (NAS) 上に存在することが可能です。

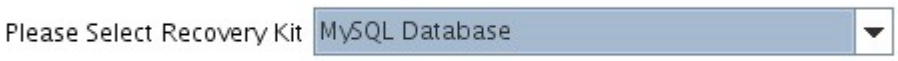
プライマリサーバからリソースインスタンスを作成するには、以下の手順を実行します。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択します。

   すでに行った選択を変更する場合、または MySQL リソース階層を作成するステップのいずれかでエラーメッセージが表示された場合、通常は 1 つ前に戻って、選択を変更するか訂正することができます。([Back] ボタンが有効な場合)。

   **重要:** リソースの作成時には、保護対象の MySQL データベースサーバデーモン (mysqld) が稼働していなければなりません。

   ダイアログボックスが表示され、ドロップダウンメニューに、クラスタにインストールされ認識されているすべての Recovery Kit が表示されます。ドロップダウンメニューから **[MySQL Database]** を選択します。

   **[Next]** をクリックします。

   階層作成の途中で **[Cancel]** ボタンをクリックすると、作成処理全体が取り消されます。

2. **[Switchback Type]** を選択します。この選択によって、バックアップサーバへのフェイルオーバの後、MySQL インスタンスが In Service に戻るときに、このサーバにどのようにスイッチバックされるのかが決まります。 ***[intelligent]*** または ***[automatic]*** を選択できます。 ***[intelligent]*** の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。 ***[automatic]*** の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。

スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。

**[Next]** をクリックします。

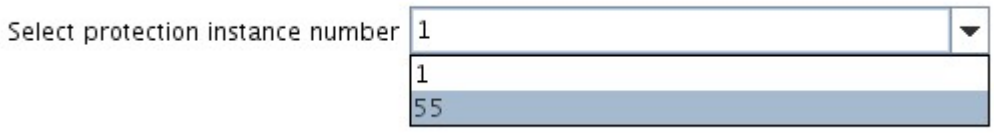
3. **[Server]** で、MySQL データベースインスタンスを配置するサーバ(通常これをプライマリサーバまたはテンプレートサーバと呼ぶ)を選択します。ドロップダウンメニューには、クラスタ内の全サーバが表示されます。

**[Next]** をクリックして次のダイアログボックスに進んでください。

4. **[Location of my.cnf]** を選択または入力します。これは MySQL 設定ファイル (my.cnf) が存在する場所のフルパス名 (ファイル名は除く) です。

**[Next]** をクリックして次のダイアログボックスに進んでください。

5. mysqld_multi 形式の my.cnf ファイルを保有している場合は **[Protection Instance Number]** を選択してください。従来の形式の my.cnf ファイルを保有している場合は、この画面は表示されません。




6. **[Location of MySQL executables]** の場所を選択または入力します。これは、MySQL データベースサーバデーモンの開始と監視に使用されるバイナリのフルパス名です。

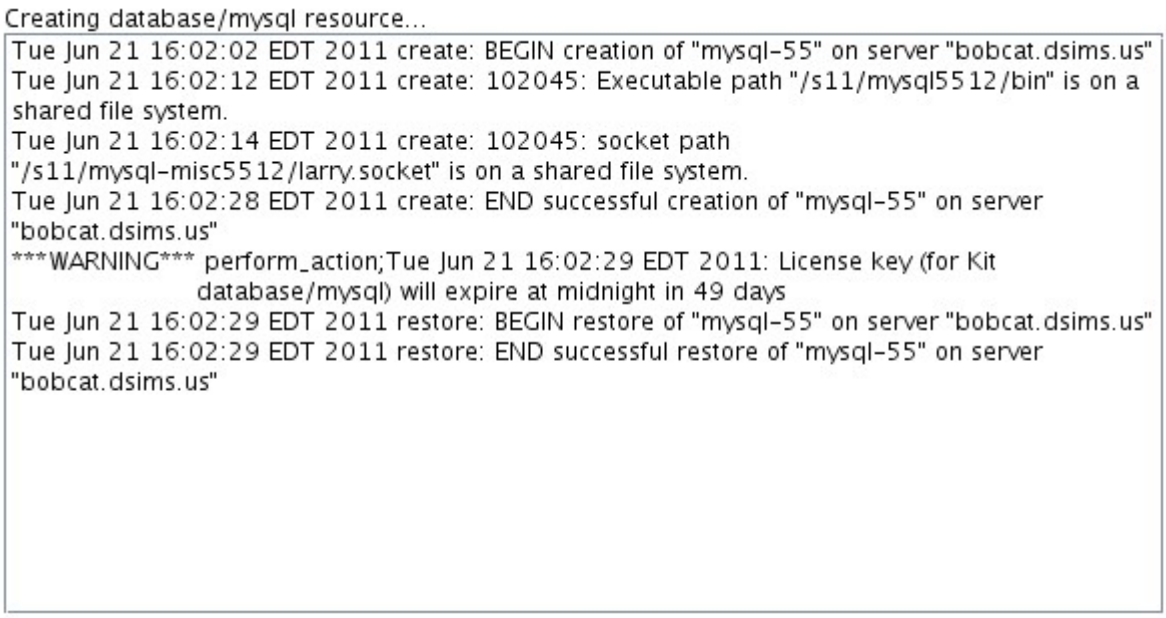
**注記:** この時点で、入力したデータが MySQL リソース階層の作成に妥当かどうかが検査されます。この検査で問題が検出された場合、エラーが画面に表示されます。ディレクトリのパスは妥当で、

MySQL 設定自体にエラーがある場合は、ここでそのエラーを訂正してから階層の作成を続けることができます。

**[Next]** をクリックして次のダイアログボックスに進んでください。

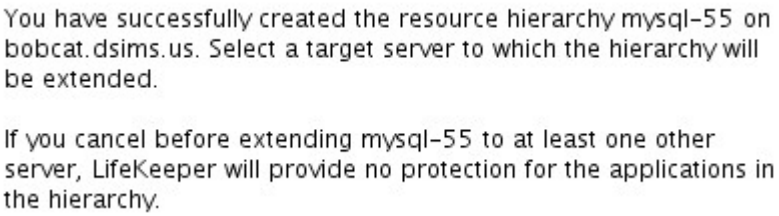
7. **[Database Tag]** を選択または入力します。これは MySQL 階層につけるタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。

**[Create]** をクリックすると、**[Create Resource Wizard]** により MySQL リソースが作成されます。




**注記:** この時点で、MySQL リソース階層が正常に作成されます。

8. 別の情報ボックスが表示され、MySQL リソース階層が正常に作成されたこと、MySQL リソース階層を LifeKeeper の保護下に置くには、その階層をクラスタ内の別のサーバに拡張する必要があることが示されます。




**[Continue]** をクリックすると、本セクションで後述する **[Pre-Extend Wizard]** が起動されます。

ここで **[Cancel]** をクリックすると、別のダイアログボックスが表示され、いずれかの時点でここに

戻り、MySQL リソース階層を別のサーバに拡張して、LifeKeeper の保護下に置く必要があることが警告されます。

```
Hierarchy Verification Finished

WARNING: Your hierarchy exists on only one server.  Your
WARNING: application has no protection until you extend it
WARNING: to at least one other server.
```

9. **[Done]** をクリックして終了します。

# 6.7.3.3. リソース階層の削除

LifeKeeper 環境内のすべてのサーバからリソース階層を削除するには、以下の手順を実行します。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択します。ドロップダウンメニューから、 **[Delete Resource Hierarchy]** を選択します。

1. **[Target Server]** で MySQL リソース階層を削除するターゲットサーバの名前を選択します。

   **注記:** 右側ペインで個々のリソースインスタンスを右クリックするか、リソースが 1 台のサーバ上のみにある場合に左側ペインでグローバルリソースを右クリックして **[Delete Resource]** 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックします。

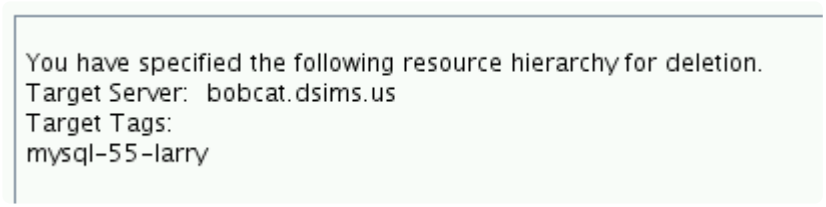
3. **[Hierarchy to Delete]** を選択します。削除するリソース階層を特定して、強調表示にします。

   **注記:** 左側ペインのグローバルリソースまたは右側ペインの個々のリソースインスタンスを右クリックして **[Delete Resource]** 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックします。

4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。




**[Delete]** をクリックします。

5. MySQL リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。

6. **[Done]** をクリックして終了します。

# 6.7.3.4. リソース階層の拡張

階層の作成後、クラスタ内の別のサーバに拡張する必要があります。3 通りのシナリオで、テンプレートサーバからターゲットサーバにリソースインスタンスを拡張できます。最初のシナリオは、リソース作成後、 **[Continue]** をクリックして、別のサーバにリソースを拡張する場合です。2 番目のシナリオは、以下に示すように **[Edit]** メニューから **[Extend Resource Hierarchy]** を選択する場合です。3 番目のシナリオは、左側または右側のペインから拡張されていない階層を右クリックする場合です。どのシナリオでも同じダイアログボックスが表示されます(いくつかの例外については、以下に詳細を明記する)。

1.
2. LifeKeeperGUI メニューから **[Extend]** ウィザードを開始する場合は、 **[Edit]** を選択し、次に **[Resource]** を選択してください。ドロップダウンメニューから **[Extend Resource Hierarchy]** を選択してください。これで **[Extend Resource Hierarchy]** ウィザードが起動されます。

3. 最初に表示されるダイアログボックスで、 MySQL リソース階層が現在 in service の **[Template Server]** を選択してください。ここで選択する **[Template Server]** と次のダイアログボックスで選択する **[Tag to Extend]** によって、 *in service* のリソース階層が表されることを認識しておくことが重要です選択したテンプレートサーバでサービス中でないリソースタグを選択すると、エラーメッセージが表示されます。このダイアログのドロップダウンボックスには、クラスタ内の全サーバの名前が表示されます。

   **注記:** MySQL リソース階層作成後、直ちに **[Extend Resource Hierarchy]** 作業に入った場合は、作成段階でテンプレートサーバが特定されているため、 このダイアログボックスは表示されません。 これは、GUI ウィンドウの左側ペインの MySQL リソースアイコンまたは右側ペインの MySQL リソースボックスを右クリックして、 **[Extend Resource Hierarchy]** を選択した場合も同様です。

   階層を拡張する手順の間に **[Cancel]** をクリックすると、どの時点であってもそのサーバへの拡張処理がキャンセルされますので注意してください。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

   たとえば、Server1 にリソースを作成し、それを Server2 に拡張しているとします。同じリソースを Server3 に拡張している最中に、気が変わっていずれかのダイアログボックス内の **[Cancel]** をクリックしたとします。この場合は、Server3 へのリソース拡張だけがキャンセルされ、Server2 への拡張はキャンセルされません。この階層から Server2 を削除したい場合は、Server2 からリソースを拡張解除する必要があります。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. **[Tag to Extend]** を選択してください。これは、テンプレートサーバからターゲットサーバに拡張する MySQL インスタンスの名前です。ウィザードのドロップダウンメニューには、前述のダイアログボックスで選択したテンプレートサーバ上に作成されたすべてのリソースが表示されます。

**注記:** ここでも、MySQL リソース階層作成後、ただちに [Extend Resource Hierarchy] 作業に入った場合は、作成段階で MySQL リソースのタグ名が特定されているため、 このダイアログボックスは表示されません。 GUI ウィンドウの左側ペインの MySQL リソースアイコンまたは右側ペインの MySQL リソースボックスを右クリックして、 **[Extend Resource Hierarchy]** を選択した場合も同様です。

**[Next]** をクリックしてください。

5. MySQL リソース階層を拡張する **[Target Server]** を選択してください。ドロップダウンボックスには、選択された階層に入っていないクラスタ内のサーバ名の一覧が表示されます。

**[Next]** をクリックしてください。

6. **[Switchback Type]** を選択してください。この選択によって、バックアップサーバへのフェイルオーバの後、MySQL インスタンスがサービス中に戻るときに、このサーバにどのようにスイッチバックされるのかが決まります。 ***[intelligent]*** または ***[automatic]*** を選択できます。[intelligent] の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。[automatic] の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。

スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。

**[Next]** をクリックしてください。

7. **[Template Priority]** を選択または入力してください。これはサーバで現在 in service の MySQL 階層の優先順位です。優先順位は、1 ~ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります(数字 1 が最高の優先順位に相当します)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。 **注記:** このフィールドは階層を最初に拡張するときだけ表示されます。

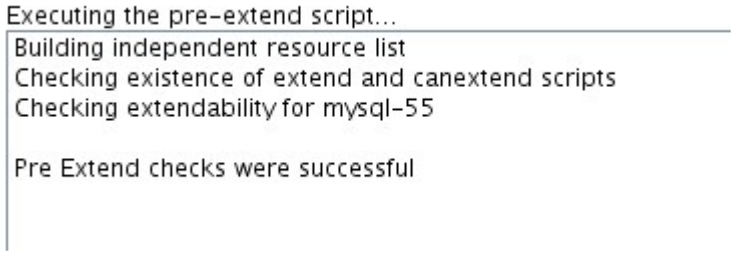
**[Next]** をクリックします。

8. **[Target Priority]** を選択または入力します。これは、別サーバにある同等の階層に対する新しく拡張する MySQL 階層の優先順位です。1 ~ 999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位は高くなります(1 は最高の優先順位を表す)。LifeKeeper のデフ

ォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。

**[Next]** をクリックしてください。

9. 情報ボックスが表示され、LifeKeeper が環境のチェックを正常に完了し、この MySQL リソースを拡張するためのすべての要件が満たされていることが示されます。満たされていない要件がある場合は、 **[Next]** ボタンは選択できなくなり、 **[Back]** ボタンが有効になります。

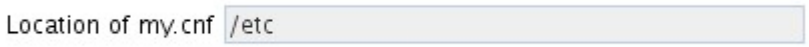



**[Back]** をクリックした場合、情報ボックスに表示されるエラーメッセージの内容に従って、リソースの拡張を変更できます。

ここで **[Cancel]** をクリックした場合は、別の時点でここに戻り、MySQL リソース階層を他のサーバに拡張して、LifeKeeper の保護下に置く必要があります。

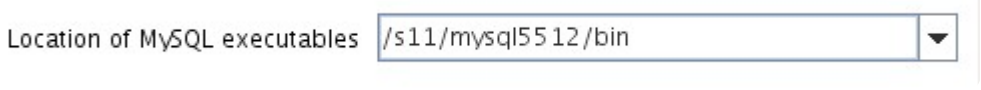
**[Next]** をクリックすると、 **[Extend Resource Hierarchy]** 設定作業に入ります。

10. このダイアログボックスは情報の表示専用です。このボックス内に表示される **[Location of my.cnf]** は変更できません。MySQL インスタンスは設定ファイルから場所情報を取得しています。




**[Next]** をクリックしてください。

11. **[Location of MySQL executables]** を選択または入力してください。これは、MySQL データベースサーバデーモンの開始と監視に使用されるバイナリのフルパス名です。

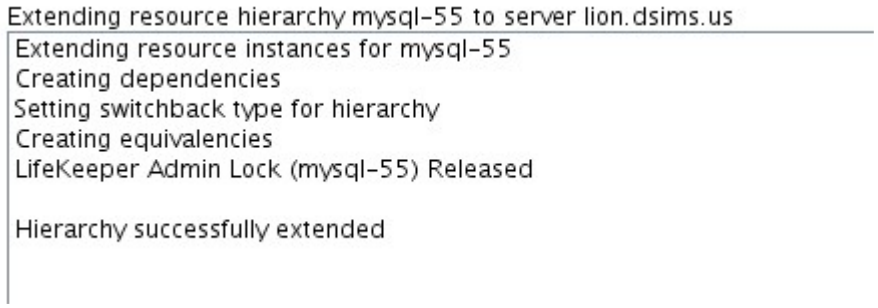



**[Next]** をクリックしてください。

12. **[Database Tag]** を選択または入力してください。これは MySQL 階層につけるタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。

**[Extend]** をクリックします。

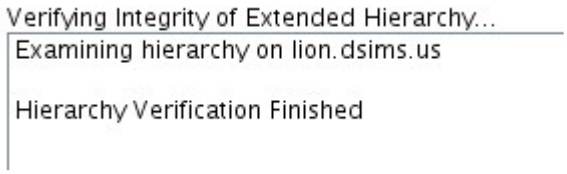
13. 拡張が実行されていることを示す情報ボックスが表示されます。




同じ MySQL リソースインスタンスをクラスタ内の別のサーバに拡張する場合は **[Next Server]** をクリックしてください。その場合は、 **リソース階層の拡張** の操作が繰り返されます。

**[Finish]** をクリックすると、LifeKeeper は MySQL リソースの拡張が正常に完了したことを確認します。

14. **[Finish]** をクリックした場合、次の画面が表示されます。




15. 最後のダイアログボックスで **[Done]** をクリックして終了してください。

**注記:** 必ず *両方の* サーバで新しいインスタンスの機能をテストしてください。

# 6.7.3.5. リソース階層の拡張解除

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択します。ドロップダウンメニューから、 **[Unextend Resource Hierarchy]** を選択します。

2. **[Target Server]** で、MySQL リソースを拡張解除するターゲットサーバを選択します。MySQL リソースが現在 In Service のサーバは選択できません。

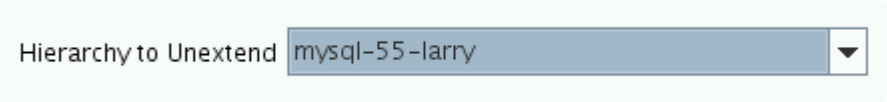
   **注記:** 右側のペインから個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックします。

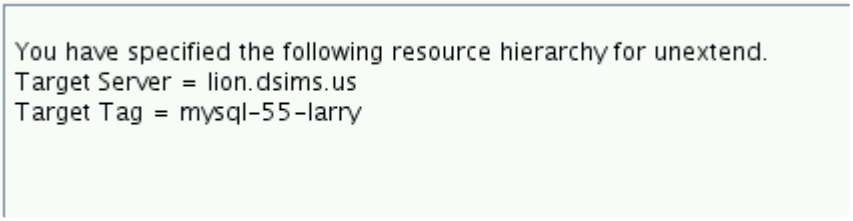
3. **[Hierarchy to Unextend]** で、拡張解除する MySQL 階層を選択してください。

   **注記:** 左側のペインからグローバルリソースを右クリックするか、右側のペインから個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

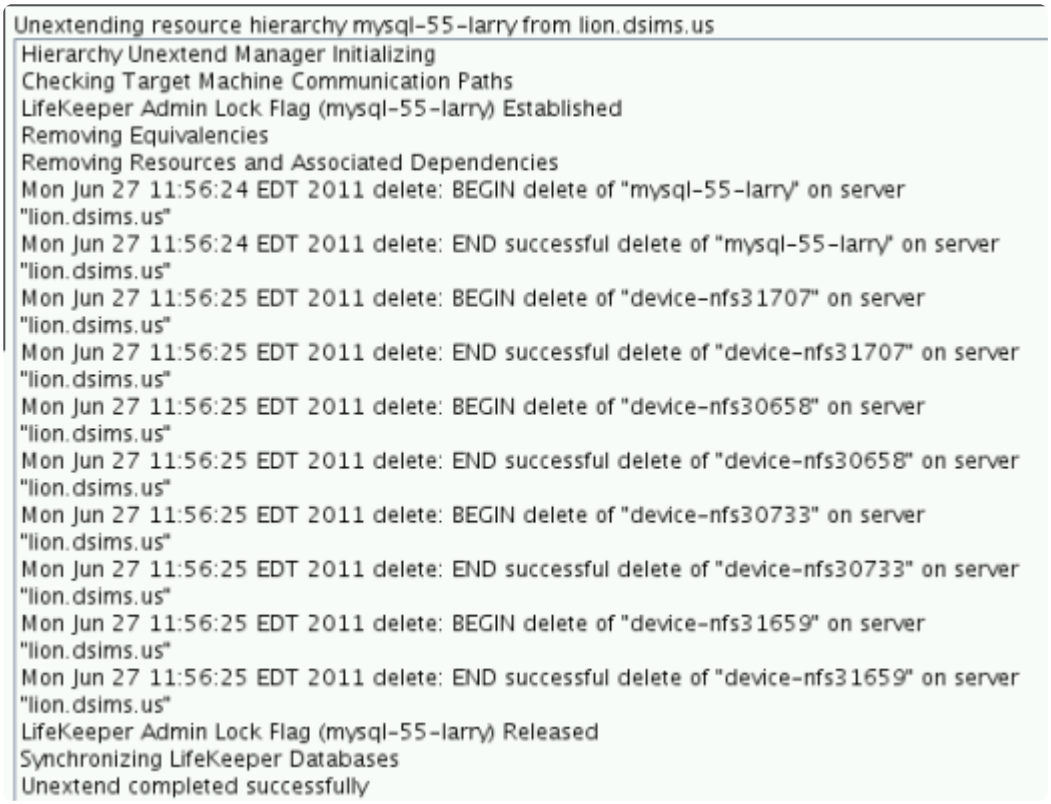
   **[Next]** をクリックします。

4. 拡張解除の対象として選択したターゲットサーバと MySQL リソース階層を確認する情報ボックスが表示されます。

   **[Unextend]** をクリックします。

5. MySQL リソースが正常に拡張解除されたことを確認する別の情報ボックスが表示されます。




6. **[Done]** をクリックして終了します。

# 6.7.4. 管理

## リソース階層のテスト

手動スイッチオーバを開始することによって、MySQL リソース階層をテストできます。これにより、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

GUI による手動スイッチオーバの実行

# 6.7.4.1. GUI による手動スイッチオーバの実行

手動スイッチオーバを開始することによって、MySQL リソース階層をテストできます。これにより、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

## GUI による手動スイッチオーバの実行

LifeKeeper GUI で、ドロップダウンメニューから **[Edit] > [Resource] > [In Service]** を選択すると、手動スイッチオーバを開始できます。例えば、バックアップサーバで **[In Service]** 要求を実行すると、アプリケーション階層はバックアップサーバで In Service になり、プライマリサーバでは停止されます。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバに変わります。

**[Out-of-Service]** 要求を実行すると、アプリケーションは他のサーバで In Service にならずに、out of service になります。

LifeKeeper では、ロールバックやバックアップアーカイブのような内部処理の調整や制御は行いません。テープアーカイブやリストアの作業は、アプリケーション管理者の責任です。

## リカバリ動作

プライマリサーバに障害が発生すると、MySQL Recovery Kit ソフトウェアは以下の処理を実行します。

- ファイルシステム (共有もしくは複製) をバックアップサーバ上にマウントします。
- MySQL に関連するデーモンプロセスを開始します。

# 6.7.5. トラブルシューティング

## 共通のエラーメッセージ

本セクションでは、SPS MySQL リソース階層の作成時や拡張時、またはリソースの削除時やリストア時に表示される可能性のあるメッセージの一覧を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

他の SPS コンポーネントからメッセージが出されることもあります。そのような場合は、メッセージカタログ を参照してください。メッセージカタログには、操作、管理、GUI など、SIOS Protection Suite for Linux の使用中に遭遇する可能性のあるすべてのエラーコードが列挙されています。また、エラーコードの原因に関する追加の説明や、問題解決のために必要な処置についても、必要に応じて記載されています。この完全なリストを検索すると、受信したエラーコードを見つけることができます。また、関連する SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

## MySQL 固有のエラーメッセージ

**注記:** エラーメッセージ列で、二重引用符で囲まれたすべて大文字の語はサーバ上のリソースの名前を示します (例えば、"SERVER" は実際には「Server1」という名前のサーバである場合があります)。

| エラー番号 | エラーメッセージ |
|---|---|
| 102001 | Usage: “SCRIPT NAME” sysname dbvarname cnfpath exepath instance |
| 102002 | Usage: “SCRIPT NAME” cnfpath |
| 102003 | Usage: “SCRIPT NAME” exepath cnfpath |
| 102004 | Unable to obtain a valid value for the “socket” variable in “PATH”/my.cnf<br>**処置:** my.cnf 設定ファイルの「mysqld」セクションには、「socket」のエントリがなければなりません。 |
| 102005 | Unable to obtain a valid value for the “port” in “PATH”/my.cnf<br>**処置:** my.cnf 設定ファイルの「mysqld」セクションには、「port」のエントリがなければなりません。 |
| 102006 | Unable to obtain the data directory location “PATH”<br>**処置:** データベースが指定したソケットとポートを使用して稼働していることを確認してください。 |
| 102007 | Must specify the absolute path to the my.cnf configuration file |
| 102008 | Must specify the absolute path to the MySQL executables |
| 102009 | The file my.cnf does not exist in the path specified |
| 102010 | The MySQL executables do not exist in the path specified |

| | |
|---|---|
| 102011 | LifeKeeper was unable to start the MySQL database server |
| 102012 | LifeKeeper successfully started the MySQL database server |
| 102013 | LifeKeeper was unable to stop the MySQL database server |
| 102014 | LifeKeeper successfully stopped the MySQL database server |
| 102015 | The port “PORT NUMBER” is in use on the target server “SERVER” |
| 102016 | The MySQL database server is not running on server “SERVER” |
| 102017 | Unable to open the configuration file “PATH”/my.cnf |
| 102018 | Unable to get the Data Directory information for resource “TAG” on server “SERVER” |
| 102019 | Unable to get the configuration file location information for resource “TAG” on server “SERVER” |
| 102020 | Unable to get the executable location information for resource “TAG” on server “SERVER” |
| 102021 | The argument for the configuration file path is empty |
| 102022 | The argument for the executable path is empty |
| 102023 | The path “PATH” for directive “DIRECTIVE” is not on a shared filesystem |
| 102024 | Unable to get the information for resource “TAG” on system “SYSTEM” |
| 102025 | The MySQL data directory “DATADIR” is already under LifeKeeper protection |
| 102026 | The port variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match |
| 102027 | The socket variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match |
| 102028 | Unable to obtain a valid value for the “user” variable in “PATH”/my.cnf<br>**処置**: my.cnf 設定ファイルの「client」セクションには、「user」変数の有効なエントリがなければなりません。 |
| 102029 | Unable to obtain a valid value for the “password“ variable in “PATH”/my.cnf<br>**処置**: my.cnf 設定ファイルの「client」セクションには、「password」変数の有効なエントリがなければなりません。 |
| 102030 | The user variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match |
| 102031 | The password variables in the file /etc/my.cnf on “SERVER1” and “SERVER2” do not match |
| 102032 | Unable to obtain the pid file location |

| | |
|---|---|
| | **処置**: my.cnf 設定ファイルの mysqld セクションには、「pid-file」変数のエントリがなければなりません。 |
| 102033 | Unable to obtain a valid value for the “user” variable in “PATH”/my.cnf<br>**処置**: my.cnf 設定ファイルの「mysqld」セクションには、「user」変数を使用して OS ユーザを指定する必要があります。 |
| 102034 | 警告: A my.cnf file exists at “PATH”, which may override the values specified in the file at “PATH”/my.cnf. |
| 102035 | The mysql system user “USER” does not exist on target server “SERVER” |
| 102036 | The mysql system user “USER” uids are different on target server “SERVER1” and template server “SERVER2” |
| 102037 | The mysql system user “USER” gids are different on target server “SERVER1” and template server “SERVER2” |
| 102038 | LifeKeeper was unable to stop the MySQL database server using a graceful shutdown. Issuing kill for pid(s): “PROCESS ID LIST”. |
| 102039 | LifeKeeper will ignore failed connection as possible max connections error, due to existence of process pid “PROCESS ID”. |
| 102040 | The mysql action for resource tag :TAG” returned: “COMMAND OUTPUT”. |
| 102041 | LifeKeeper was unable to start the MySQL database server using the defaults-file option. Retrying with individual options. |
| 102042 | The LifeKeeper “ACTION” action detected the flag “FLAG”, and will exit. |
| 102043 | END of “ACTION” action on due to a(n) “SIGNAL” signal. |
| 102044 | The file my.cnf does not exist in the stored path “PATH”. |
| 102045 | “DIRECTIVE” path “PATH” is on a shared filesystem. |
| 102046 | Starting mysqld daemon with databases from “PATH”. |

# 6.8. WebSphere MQ Recovery Kit 管理ガイド

## MQ Recovery Kit テクニカルドキュメンテーション

SIOS Protection Suite for Linux WebSphere MQ Recovery Kit は、WebSphere MQ キューマネージャとキューマネージャのストレージ場所に、耐障害性保護を提供します。このキットを使用すると、プライマリ WebSphere MQ サーバまたはキューマネージャ上で障害が発生した場合、人手の介入なしに迅速にプライマリサーバまたは指定のバックアップサーバで復旧することが可能です。

## ドキュメントの内容

このガイドでは次の事項について説明します。

- SIOS Protection Suite ドキュメンテーション 。SPS for Linux のドキュメンテーションの一覧と入手場所を記載します。
- 略語 。本ドキュメンテーション全体で使用される略語の一覧およびその意味を記載します。
- 要件 。WebSphere MQ Recovery Kit を適切に設定、インストール、運用する上で必要なハードウェアとソフトウェアを説明します 。SPS for Linux ソフトウエアをインストールまたはアンインストールする方法の具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。
- WebSphere MQ Recovery Kit の概要 。WebSphere MQ Recovery Kit の特徴および機能の概要と、この Recovery Kit でサポートされる WebSphere MQ ソフトウェアのバージョンの一覧を示します。
- WebSphere MQ を設定する際の考慮事項 。設定上の問題の概要と、WebSphere MQ Recovery Kit でサポートされるファイルシステムのレイアウトを示します。
- LifeKeeper 使用時の WebSphere MQ の設定。 WebSphere MQ のインストール手順と LifeKeeper とともに使用するための設定手順を示します。
- リソース作成後の設定変更 。WebSphere MQ の設定変更が LifeKeeper WebSphere MQ のリソース階層に与える影響に関する情報を記載します。
- WebSphere MQ の設定例 。一般的な WebSphere MQ 設定の例と、WebSphere MQ リソースの設定手順を示します。
- LifeKeeper の設定作業。 LifeKeeper GUI を使用して WebSphere MQ リソース階層を作成および管理するための作業を説明します。
- WebSphere MQ のトラブルシューティング。 情報メッセージとエラーメッセージ、および推奨される対処方法の一覧を示します。
- 付録 。WebSphere MQ の設定ファイルの例と、WebSphere MQ インストールを計画するために使

用できる設定シートが含まれます。

# 参考ドキュメント

MQドキュメンテーションは、以下の WebSphere MQ Library にあります。

http://www.ibm.com/software/integration/wmq/library/

# SPS ドキュメンテーション

以下は、SIOS Technology Corp が公開している SPS 関連情報の一覧です。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- SIOS Protection Suite インストールガイド
- Optional Recovery Kit ドキュメンテーション
- SPS for Linux IP Recovery Kit 管理ガイド

このドキュメンテーションと、その他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

# 6.8.1. 略語

本ドキュメンテーション全体で、使用される略語は次のとおりです。

| 略語 | 意味 |
|---|---|
| HA | Highly Available, High Availability (高可用性) |
| QMDIR | WebSphere MQ queue manager directory.このディレクトリは、キューマネージャのパーシステントキューデータを格納します。一般的に、 */var/mqm/qmgrs* 内に、キューマネージャー名をサブディレクトリ名として配置されます。このディレクトリの正確な場所は、グローバル設定ファイル *mqs.ini* 内で指定されます。<br><br>DataPath パラメータを定義する場合は、キューマネージャー名を含む DataPath 値でキューマネージャのパーシステントデータの場所が指定されます。定義しない場合は、上記のデフォルトの場所が使用されます。 |
| QMLOGDIR | WebSphere MQ queue manager log directory.このディレクトリは、キューマネージャのログデータを格納します。一般的に、 */var/mqm/log* 内に、キューマネージャー名をサブディレクトリ名として配置されます。このディレクトリの正確な場所は、キューマネージャ設定ファイル (*QMDIR/qm.ini*) 内で指定されます。 |
| MQUSER | すべての WebSphere MQ コマンドを実行するオペレーティングシステムのユーザ。このユーザは、 QMDIR の所有者です。このユーザは、MQGROUP 管理グループ mqm (下記を参照) のメンバである必要があります。 |
| MQGROUP | MQUSER が参加すべきオペレーティングシステムのユーザーグループ。このグループの名前は、mqm である必要があります。 |
| UID | オペレーティングシステムユーザの数値 ID。 |
| GID | オペレーティングシステムユーザグループの数値グループ ID。 |

# 6.8.2. 要件

WebSphere MQ Recovery Kit をインストールする前に、SPS の設定が次の要件を満たしている必要があります。SPS ハードウェアおよびソフトウェアの設定に関する具体的な手順については、SIOS Protection Suite 管理ガイド を参照してください。

# 6.8.2.1. ハードウェア要件とソフトウェア要件

## ハードウェア要件

- サーバ 。Recovery Kit には、SIOS Protection Suite インストールガイド に記載の要件に従って設定された 2 台以上のサーバが必要です。サポートされる Linux ディストリビューションについては、Linux の設定 を参照してください。

- データストレージ。WebSphere MQ Recovery Kit は、共有ストレージと DataKeeper 製品によって提供される複製されたストレージの両方と組み合わせて使用できます。また、Network Attached Storage (NAS) と使用することもできます。

## ソフトウェア要件

- **SPS** ソフトウェア 。各サーバにインストールする **SPS** ソフトウェアおよびパッチのバージョンは同じにする必要があります。

- **LifeKeeper WebSphere MQ Recovery Kit** 。WebSphere MQ v7.1 以降を実行するシステムには、WebSphere MQ Recovery Kit バージョン 7.5.1 以降が必要です。

- **LifeKeeper IP Recovery Kit** 。各サーバの LifeKeeper IP Recovery Kit のバージョンは同じにする必要があります。

- **IP** ネットワークインターフェース 。各サーバは、イーサーネット TCP/IP をサポートするネットワークインターフェースを 1 つ以上必要とします。IP スイッチオーバが正しく動作するには、ローカルネットワークに接続されているユーザシステムが標準の TCP/IP 仕様に準拠している必要があります。

  注記: 各サーバが必要とするネットワークインターフェースが 1 つだけであっても、異種メディア要件、スループット要件、単一障害点の排除、ネットワークのセグメンテーションといった多くの理由のために、複数のインターフェースを使用すべきです。

- **TCP/IP** ソフトウェア 。各サーバは TCP/IP ソフトウェアも必要とします。

- **WebSphere MQ** ソフトウェア 。IBM WebSphere MQ は、別途 IBM から注文する必要があります。サポートされる WebSphere MQ のバージョンについては、SPS リリースノート を参照してください。WebSphere MQ ソフトウエアは、WebSphere MQ Recovery Kit をインストールする前にクラスタの各サーバにインストールする必要があります。WebSphere MQ Recovery Kit を正常にインストールするには、次の WebSphere MQ パッケージがインストールされている必要があります。

  MQSeriesServer、MQSeriesSamples、MQSeriesClient、MQSeriesRuntime、MQSeriesSDK

  IBM WebSphere MQ バージョン 7.0.1 Fix Pack 6 以降、複数のバージョンの WebSphere MQ (例: MQ バージョン 7.0.1 Fix Pack 6 と 7.1) を同一サーバ上でインストールおよび実行できるようにする新機能が導入されました。この機能 (マルチインスタンスサポート) は、WebSphere MQ Recovery Kit 9.0.1 のバージョン以降でサポートされるようになりました。マルチインスタンス機能の一部として導入された単一の IBM WebSphere MQ インストールバージョン内で複数のキューマ

ネージャを保護する機能と、複数の IBM WebSphere MQ インストールバージョン内でキューマネージャを保護する機能、mqs.iniファイル 内での DataPath パラメータの使用は、このバージョンのリカバリキットでいずれもサポートされています。

- **オプションの C コンパイラ** 。WebSphere MQ Recovery Kit には、WebSphere MQ サンプルパッケージ内の修正版 amqsget0.c サンプルプログラムが含まれています。このプログラムは、デフォルトの 15 秒のタイムアウトではなく、0 秒のタイムアウトに対応するよう修正されました。このプログラムは、キューマネージャの PUT/GET テストを実行するために使用されます。このプログラムは RPM のインストール中にコンパイルされるため、C コンパイラがインストールされ、root ユーザの PATH に配置されている必要があります。

- **Syslog.pm**。 WebSphere MQ リソースで syslog ログを使用するには、Perl モジュール の Syslog.pm がインストールされている必要があります。このモジュールは、標準の Perl ディストリビューションに含まれているため、別途インストールする必要はありません。

# 6.8.2.2. LifeKeeper MQ クラスタのアップグレード

1. WebSphere MQ Recovery Kit を含むクラスタ内のすべてのノード上の SPS を、SIOS Protection Suite インストールガイド の「Upgrading SPS」セクションに記載の手順に従ってアップグレードします。

2. クラスタ内の各ノード上の IBM WebSphere MQ ソフトウェアを以下の手順でアップグレードします。

   a. 1 つまたは複数の LifeKeeper IBM WebSphere MQ リソース階層がアップグレードされるノード上で in service の場合、IBM WebSphere MQ ソフトウェアのアップグレードを実行する前に out of service にしなければなりません。この設定は、スタンドバイノードへのスイッチオーバによって可能です。

   b. IBM WebSphere アップグレード手順に従ってください。

3. IBM WebSphere ソフトウェアがノードにインストールされたら、LifeKeeper IBM WebSphere MQ リソース階層を in service にし (リストア)、各 Queue Manager の動作を確認します。

4. 各 Queue Manager の動作を確認したら、クラスタ内の他のすべてのノードをアップグレードします。

# 6.8.3. WebSphere MQ Recovery Kit の概要

WebSphere MQ (旧 MQSeries) は、IBM のソフトウェア製品で、一度だけの信頼性の高いメッセージ配信を行います。WebSphere MQ の中核要素は、メッセージを送受信するための1 つ以上のキューを処理するキューマネージャです。メッセージがキューに追加されると、メッセージは確実にパーシステントになり、一度だけ配布されます。

WebSphere MQ Recovery Kit を使用すると、コマンドサーバ、リスナー、およびパーシステントキューマネージャデータを含む WebSphere MQ キューマネージャを LifeKeeper で保護できます。キューマネージャリスナーの保護は、オプションでキューマネージャごとに無効化できます。これにより、クライアント接続を扱わない設定に対応したり、LifeKeeper のリカバリ試行を発動させることなく管理者がリスナーをシャットダウンしたりできます。

WebSphere MQ Recovery Kit は、保護対象の WebSphere MQ キューマネージャを、障害の発生したプライマリサーバからバックアップサーバにリカバリする仕組みを提供します。LifeKeeper はサーバレベル (ハートビートによる) またはリソースレベル (WebSphere MQ デーモンの監視による) で障害を検知して、保護対象の WebSphere MQ サービスの管理をバックアップサーバに移譲します。

また、WebSphere MQ Recovery Kit は、複数のバージョンの WebSphere MQ のインストールをサポートしており、同じシステム上にインストールして実行することができます。MQ のマルチバージョンはMQ ソフトウェアのバージョン 7.x、およびバージョン 8.x のキューマネージャをサポートし、いずれも Recovery Kit により保護することが可能です。WebSphere MQ Recovery Kit の 9.0.2 のバージョンにおける本サポート以前は、同じシステムにインストールして実行できる MQソフトウェアバージョンは 1 つのみでした (注記: インストールは、デフォルトの場所 /opt/mqm に行う必要がありました)。

# 6.8.3.1. WebSphere MQ リソース階層

一般的に、WebSphere MQ 階層は、WebSphere MQ キューマネージャリソースによって構成されます。また、ファイルシステム構成や IP リソースの有無に応じて、1 つまたは複数のファイルシステムリソースが含まれます。具体的な階層構成は、保護する内容によって変わります。IP リソースを WebSphere MQ リソース階層に含める場合、WebSphere MQ キューマネージャリソースを作成する前に管理者はその IP を作成し、またその IP リソースをプライマリサーバ上でアクティブにする必要があります。ファイルシステム階層は、WebSphere MQ キューマネージャリソースの作成中に自動的に作成されます。

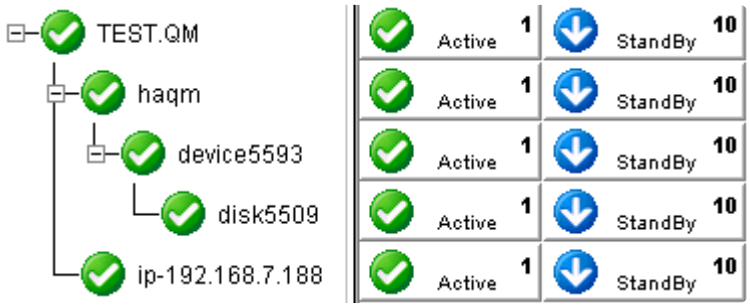



図 1 一般的な WebSphere MQ 階層 (シンボリックリンク)

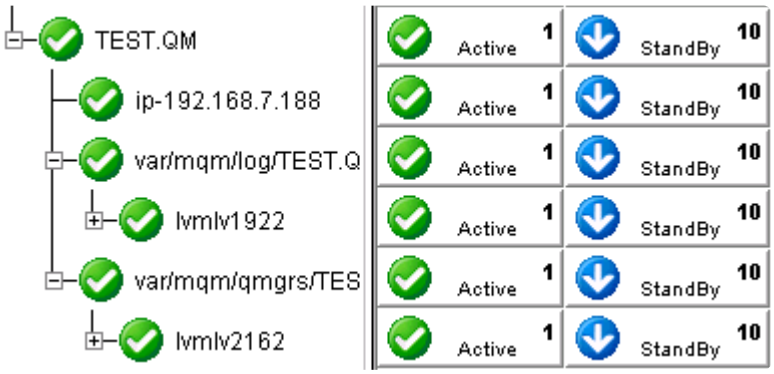



図 2 一般的な WebSphere MQ 階層 (LVM 設定)

# 6.8.3.2. Recovery Kit の機能

WebSphere MQ Recovery Kit は、次の機能を提供します。

- マルチインスタンスのキューマネージャ (MQ ソフトウェアの複数のバージョンで作成されたキューマネージャ) のサポート
- アクティブ / アクティブ設定のサポート
- LINEAR および CIRCULAR ログ (自動検出) のサポート
- サーバ接続およびクライアント接続によるエンドツーエンドアプリケーション健全性チェックのサポート
- オプションの PUT/GET テスト (テストキューを GUI とコマンドラインで定義可能) のサポート
- カスタマイズ可能なログレベルのサポート
- LifeKeeper でサポートされるすべてのストレージタイプをサポート
- オプションのリスナー保護のサポート (デフォルトで有効)
- 追加の syslog メッセージログ (ログ機能 local7) のサポート
- 複数レベルのコマンドサーバ保護 (デフォルトは「full」レベル)

# 6.8.4. WebSphere MQ を設定する際の考慮事項

本セクションでは、WebSphere MQ の設定を開始する前に考慮すべき事項について説明します。また、WebSphere MQ を LifeKeeper キューマネージャで設定および保護するための操作手順を示します。

WebSphere MQ を SPS でLinux ディストリビューションにインストールする方法については、WebSphere MQ ドキュメンテーション を参照してください。

# 6.8.4.1. 設定の要件

LifeKeeper 使用時の WebSphere MQ の設定 セクションには、LifeKeeper でキューマネージャを保護するための手順が記載されています。一般的に、LifeKeeper とともに使用する WebSphere MQ キューマネージャを正しく設定するには、次の要件が満たされている必要があります。

1. **カーネルパラメータの設定** 。Linux カーネルパラメータ（共有メモリやその他のカーネルリソースなど）の設定方法については、WebSphere MQ ドキュメンテーション を参照してください。

2. **MQUSER** および **MQGROUP** 。MQGROUP および MQUSERは、クラスタのすべてのサーバー上に存在する必要があります。Websphere MQ ソフトウェアでは、MQGROUP mqm が存在し、それが、プライマリグループメンバーシップがMQGROUP mqmに設定されていると定義されたMQUSER mqmを持っている必要があります。 もし、Websphere MQソフトウェアがインストールされるときにmqm userとmqm groupが存在しない場合には、mqm userとmqm groupは自動的に生成されます。 WebSphere MQソフトウェアをインストールする時には、 ほとんどのファイルとディレクトリは、mqm userとmqm groupに対して設定されたユーザーおよびグループの所有権を持ちます。Queue Managerデータとログディレクトリの中のファイルとディレクトリのユーザーおよびグループの所有権は、mqm userとmqm groupに対しても設定されます。さらに、Queue Managerが開始される時には、それがmqm userとして動作します。 そのため、MQUSER ユーザーID (uid) とMQGROUPグループID (gid) は、すべてのクラスターのサーバー上で同一でなくてはなりません。MQ Recovery Kitはリソースを拡張しようとする際に、この点を検証します。もし、この二つのIDが一致しない場合には、リソースの拡張に失敗します。注：もしローカルパスワードやグループファイル以外にNIS、LDAPもしくは他の認証ツールを使用している場合には、Websphere MQとLifeKeeperソフトウェアのインストールの前にMQUSERとMQGROUPを設定しなくてはなりません。また、ホームディレクトリを作成する必要があるかもしれません。もし、今回が以前にリリースされたWebSphere MQ Recovery Kitからのアップグレードである場合、MQUSER PATH環境変数の設定の変更が必要な場合があります。このMQUSER PATH環境変数は、WebSphere MQ ソフトウェアのデフォルトインストールロケーション、 /opt/mqmを含めるために変更する必要があります。もし、この変更が以前のリリースで行われている場合、今回のRecovery Kitのバージョンが正しく動作するために解除する必要があります。

3. 別のMQユーザーのサポート Websphere MQ ソフトウェアは常にmqm ユーザーとして動作しますが、別のユーザーがプライマリもしくはセカンダリメンバーシップをmqmグループの中に持つことを前提に、全てのMQコマンドを動作させるのに別のユーザー名を指定することが可能です。他のMQ Toolと統合する時には、WebSphere MQの開始に別のユーザー名が必要かもしれません。別のユーザーへの変更については、本文書の“LifeKeeper WebSphere MQ Recovery Kit のデフォルト設定の変更”の MQS_ALT_USER_NAME設定 を参照してください。

4. **コマンドサーバーの手動起動** 。LifeKeeper によってコマンドサーバーを起動するには、プライマリサーバー上で、次のコマンドを使用してコマンドサーバーの自動起動を無効にします。この手順を行わない場合、キューマネージャの起動時にコマンドサーバーが自動的に起動されます。

```
runmqsc QUEUE.MANAGER.NAME

ALTER QMGR SCMDSERV(MANUAL)
```

5. 別のMQユーザーのサポート **が共有ストレージ上にあること** 。キューマネージャディレクトリ

QMDIR およびキューマネージャログディレクトリ *QMLOGDIR* は、LifeKeeper でサポートされた共有ストレージ上にある必要があります。これにより、バックアップサーバー上の WebSphere MQ がデータにアクセスできます。詳細については、サポートされるファイルシステムレイアウト を参照してください。

6. **QMDIR および QMLOGDIR の権限** 。*QMDIR* および *QMLOGDIR* ディレクトリは、MQUSER と MQGROUP グループに所有されている必要があります。ARK は、ディレクトリの所有者を参照して MQUSER を自動的に決定します。また、シンボリックリンクを検出し、最終ターゲットに到達します。必要な場合、システムコマンド chown を使用して、これらのディレクトリの所有者を変更してください。

7. **キューマネージャの自動起動を無効化** 。LifeKeeper で保護するキューマネージャでは自動起動を無効化する必要があります。 systemctl コマンドを使って無効化してください。

8. **サーバー接続チャネルの認証** 。WebSphere MQ バージョン 7.1 以降では、チャネルの認証方法が変更されました。デフォルトでは、MQADMIN ユーザー (mqm) は匿名認証 (パスワードなし) を行うことはできないため、リソース階層の作成に失敗します (WebSphere MQ 7.1 以前のリリースで作成されたキューマネージャの認証は引き続き有効)。WebSphere MQ 7.1 以降、MQADMIN ユーザーによる認証を許可するには、チャネルの認証を無効にする方法があります。WebSphere MQ 8.0 では、追加の変更が (現在の設定を取得する「display authinfo(system.default.authinfo.idpwos)」を runmqsc コマンドで実行して) system.default.authinfo.idpwosのauthinfoに対して必要となります。「reqdamd」のchckclnt 設定は、「optional」に変更する必要があります。MQADMIN ユーザーによる匿名認証の許可に失敗すると、リソースの作成時に「 **MQCONNX ended with reason code 2035** 」エラーが発生します。チャネルの認証については、WebSphere MQ ドキュメンテーション を参照してください。

9. **MQSeriesSamples、MQSeriesSDK、およびMQSeriesClient パッケージ** 。LifeKeeper は、WebSphere MQ へのクライアント接続を使用して、リスナーとチャネルイニシエータが完全に機能していることを確認します。これは、リモートのキューマネージャおよびクライアントがキューマネージャに接続するための要件です。そのため、MQSeriesClient パッケージが、WebSphere MQ を実行するすべての LifeKeeper クラスタノード上にインストールされている必要があります。また、MQSeriesSDK および MQSeriesSamples パッケージは、クライアント接続テストと PUT/GET テストを実行するためにインストールされている必要があります。

10. **オプションの C コンパイラ** 。オプションの PUT/GET テストを実行するためには、C コンパイラがマシン上にインストールされている必要があります。C コンパイラがない場合、インストール中に警告が発生します。

11. **LifeKeeper のテストキュー** 。WebSphere MQ Recovery Kit は、オプションで PUT/GET テストを実行してキューマネージャの動作を確認します。リカバリキットはすべてのメッセージをこのキューから取得して破棄するため、専用のテストキューが作成されている必要があります。このキューでは、デフォルトのパーシスタンス設定が「**yes**」(DEFPSIST=yes) に設定されている必要があります。キューマネージャを LifeKeeper で保護している場合、「LIFEKEEPER.TESTQUEUE」という名前のテストキューが自動的に作成されます。また、次のコマンドを使用して、キューマネージャを保護する前に手作業でテストキューを作成することもできます。

```
su - MQUSER
```

```
runmqsc QUEUE.MANAGER.NAME

define qlocal(LIFEKEEPER.TESTQUEUE) DEFPSIST(YES) DESCR('LifeKeeper
test queue')
```

注記：デフォルトの「LIFEKEEPER.TESTQUEUE」以外の名前をテストキューに付けるには、このテストキューの名前を設定する必要があります。詳細については、リソースプロパティ設定の編集 を参照してください。

12. **リスナーオブジェクト用の TCP ポート** 。リスナーオブジェクトを runmqsc を介して変更して使用中の TCP ポートを反映します。次のコマンドを使用して、デフォルトリスナーの TCP ポートを変更します。

```
su - MQUSER
runmqsc QUEUE.MANAGER.NAME

alter LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1414)
IPADDR(192.168.1.100)
```

注記: リスナーオブジェクトは、デフォルトの MQ リスナー TCP ポート 1414 を使用する場合も変更が必要ですが、特定の IP アドレス (IPADDR) を指定する必要はありません。IPADDR を設定しない場合、リスナーはサーバー上のすべてのインターフェースにバインドされます。IPADDR を設定した場合は、IPADDR で定義されたアドレスを使用して LifeKeeper で仮想 IP リソースを作成することを強く推奨します。そうすることによって、MQ リスナーの起動時に確実に IP アドレスを使用できます。

13. **TCP ポート番号** 。各 WebSphere MQ リスナーは、異なるポート (デフォルトで 1414) を使用するか、すべてのインターフェースにバインドされたリスナーがない状態で異なる仮想 IP にバインドする必要があります。これには、クラスタ内の保護されたキューマネージャと保護されていないキューマネージャが含まれます。

14. ***mqs.ini* で設定されたキューマネージャ** 。アクティブ / アクティブ設定では、各サーバーは、グローバルキューマネージャー設定ファイル *mqs.ini* のコピーをそれぞれ保持しています。保護されたキューマネージャをクラスタ内のすべてのサーバー上で実行するには、キューマネージャがクラスタ内のすべてのサーバー上の *mqs.ini* 設定ファイルに設定されている必要があります。適切な QueueManager: スタンザをプライマリサーバーからコピーして、すべてのバックアップサーバー上の *mqs.ini* 設定ファイルに追加します。

# 6.8.4.1.1. サポートされるファイルシステムレイアウト

共有ストレージシステムとファイルシステムレイアウトに応じて、4 つの異なる設定がサポートされています。各設定の違いは、ファイルシステムレイアウトにあります。次のセクションでは、サポートされるファイルシステムレイアウトについて説明します。

# 6.8.4.1.1.1. 設定 1 – 共有ストレージ上の /var/mqm

この設定では、 */var/mqm* ディレクトリ全体が、LifeKeeper でサポートされる共有ストレージ (SCSI、SAN、NAS、または複製されたストレージ) にマウントされます。

注記：これは、アクティブ / パッシブ設定でのみ動作します。

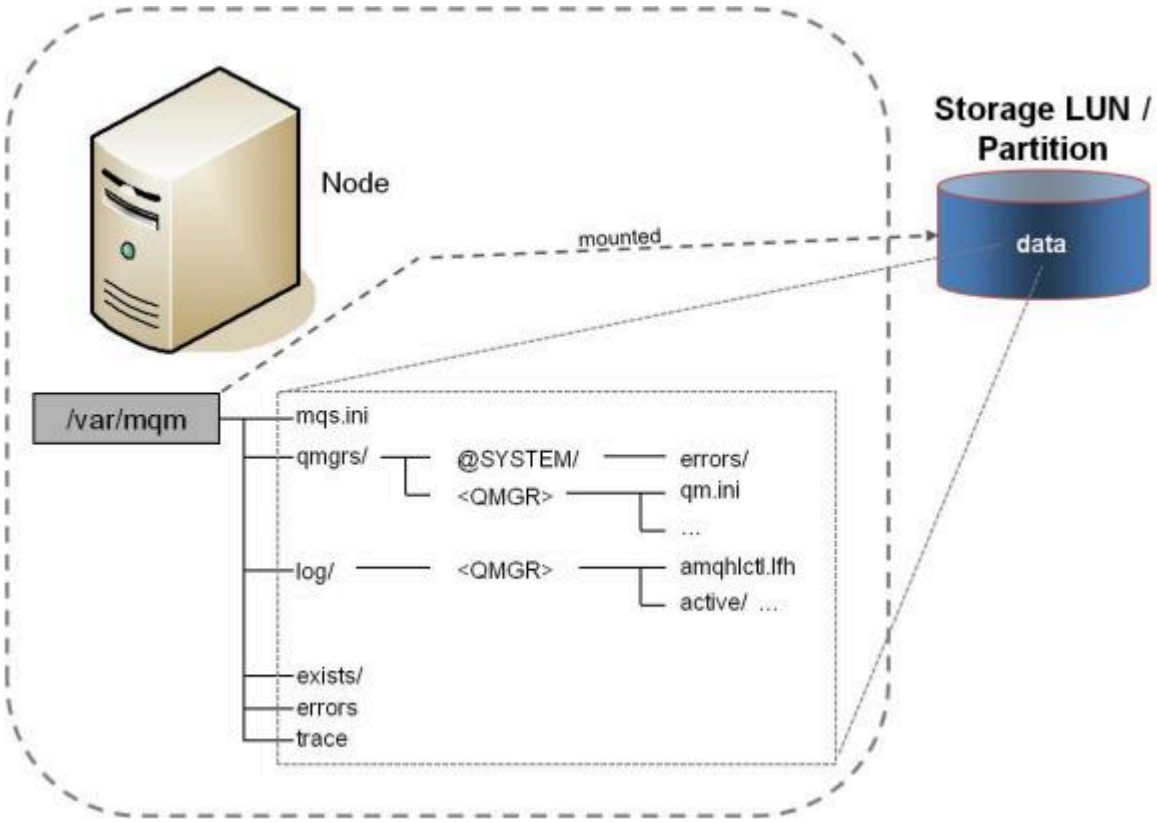



**図 3 -** ファイルシステムレイアウト **1 -** 共有ストレージ上の ***/var/mqm***

# 6.8.4.1.1.2. 設定 2 – 直接マウント

この設定では、*QMDIR* ディレクトリと *QMLOGDIR* ディレクトリは、共有ストレージ上にあります。これには、2 つの専用 LUN またはパーティションを用意するか、各キューマネージャで LVM を使用する必要があります。LVM を使用する場合、同一の LUN から 2 つの論理ボリュームを作成でき、2 つのディレクトリに個別にマウントできます。

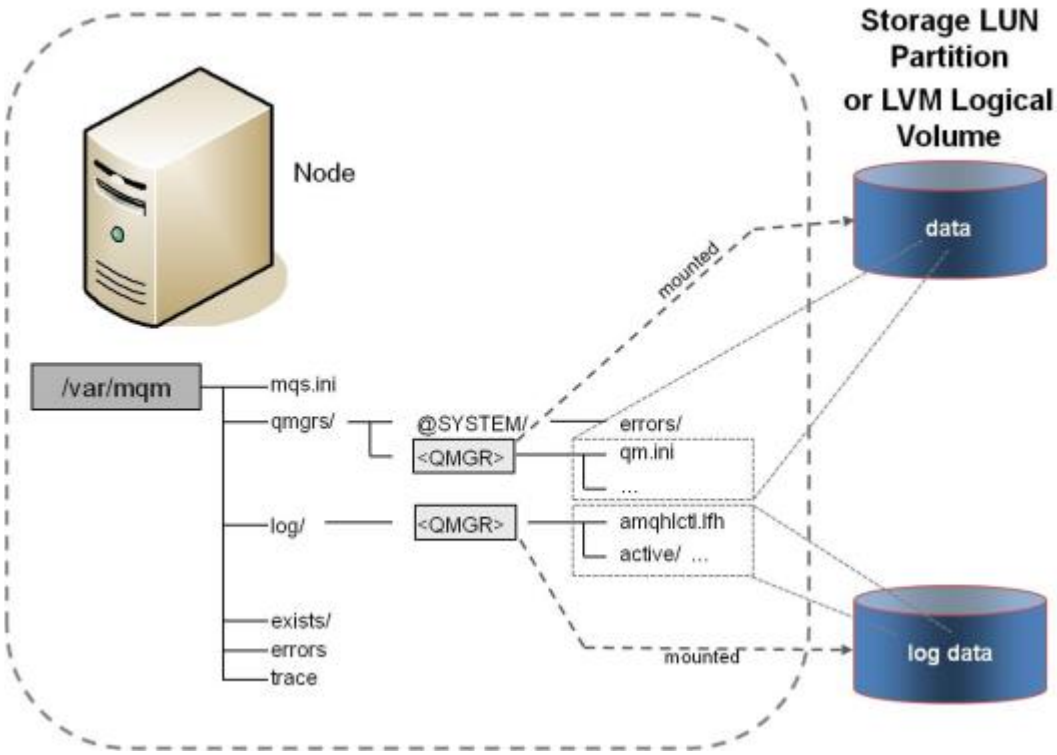



図 4 - ファイルシステムレイアウト 2 - 直接マウント

# 6.8.4.1.1.3. 設定 3 – シンボリックリンク

LVM を使用せず、多数のキューマネージャを使用するアクティブ / アクティブ設定では、シンボリックリンクの使用を推奨します。この場合、1 つまたは複数の専用マウントポイントが作成されます (例: */mq*)。LifeKeeper で保護された 1 つのファイルシステムをそこにマウントし、各キューマネージャ用にサブディレクトリを作成します (例: */mq/QUEUE!MANAGER!NAME/log* および */mq/QUEUE!MANAGER!NAME/qmgrs*)。その後、*QMDIR* ディレクトリおよび *QMLOGDIR* ディレクトリを、この場所にリンクします。

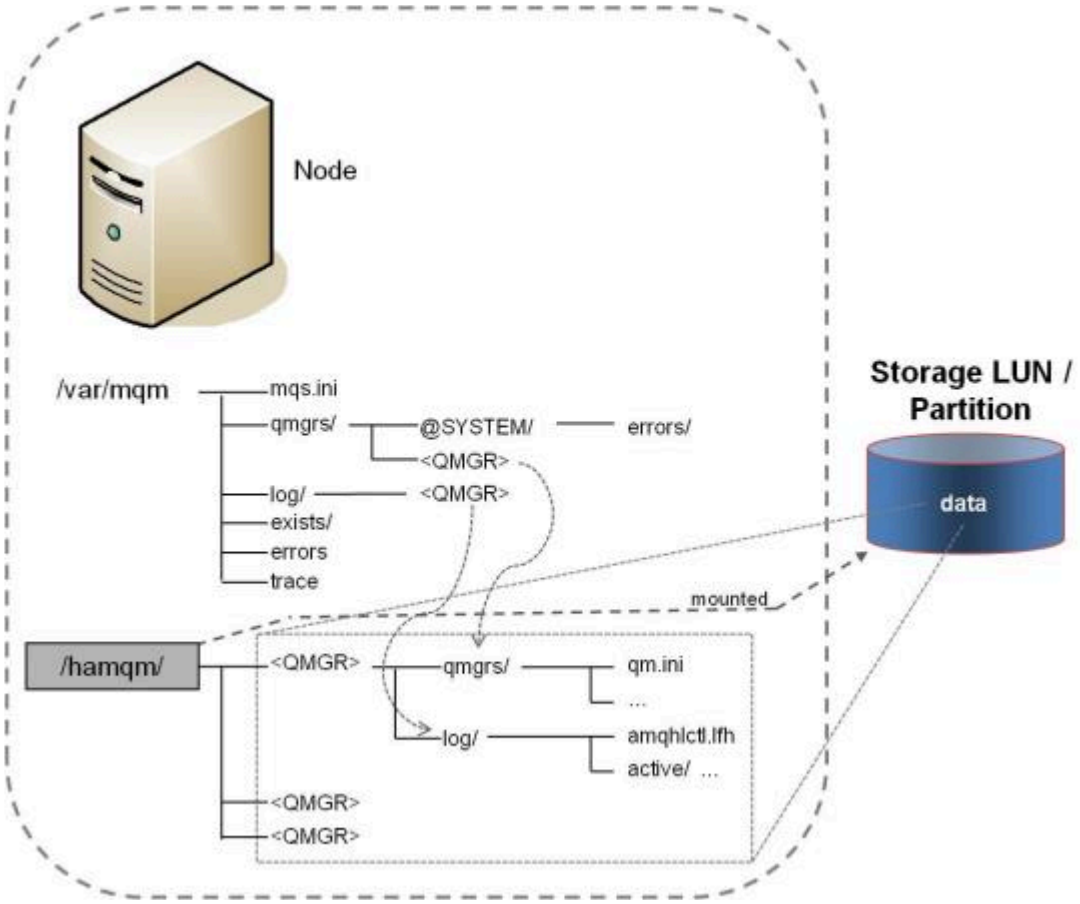



図 5 - ファイルシステムレイアウト 3 - シンボリックリンク

# 6.8.4.1.2. LifeKeeper 使用時の WebSphere MQ の設定

LifeKeeper for Linux WebSphere MQ リソース階層の作成を開始する前に実施すべき WebSphere MQ の設定上の考慮事項は多数あります。これらの変更は、Recovery Kit が PUT/GET テストを実行し、WebSphere MQ パーシステントデータに高可用性を提供するために必要です。WebSphere MQ キューマネージャは、リモートクライアントリクエストを TCP/IP を使って処理するため、仮想 IP リソースを WebSphere MQ リソース階層を作成する前に作成しておく必要があります。次の作業を実行し、LifeKeeper WebSphere MQ リソース階層を作成できるようにします。

1. インストールの計画をします (付録 C を参照)。

   WebSphere MQ をインストールする前に、インストールの計画をする必要があります。これには、MQUSER、MQUSER UID、MQGROUP GID の選択が含まれます。また、使用するファイルシステムレイアウトも決定しなければなりません (サポートされるファイルシステムレイアウト を参照)。このプロセスをスムーズに行うため、SIOS Technology Corp. では、必要なすべての情報を記入できるフォームを提供しています。付録 C – WebSphere MQ 設定シート を参照してください。インストール手順の準備として、このフォームに記入してください。

2. 各サーバでカーネルパラメータを設定します。

   WebSphere MQ では、共有メモリなど、特別な Linux カーネルパラメータ設定を必要とする場合があります。お使いのリリースのWebSphere MQの WebSphere MQ ドキュメンテーション を参照し、WebSphere MQ を実行するための最小要件を確認してください。カーネルパラメータの変更が再起動後も残るようにするためには、*/etc/sysctl.conf* 設定ファイルを使用できます。コマンド sysctl -p を起動スクリプト (boot.local) に追加する必要がある場合があります。SuSE では、insserv boot.sysctl を実行して *sysctl.conf* ファイル内のパラメータの自動設定を有効化できます。

3. 各サーバ上で MQUSER および MQGROUP を作成します。

   オペレーティングシステムコマンド groupadd および adduser を使用して、MQUSER と MQGROUP を作成します。その UID と GID は、手順 1 で使用した『WebSphere MQ 設定シート』に記載されています。

   選択した MQUSER の名前が *mqm* で、UID 1002 を持ち、MQGROUP GID が 1000 の場合、クラスタ内の各サーバ上で次のコマンドを実行できます (実際の設定に合わせて MQUSER、UID、GID の値を変更してください)。

   ```
   groupadd -g 1000 mqm
   useradd -m -u 1002 -g mqm mqm
   ```

   **注記**:これらの設定は、クラスタの全ノードで常に同一の内容である必要があります。NIS または LDAP を実行中の場合、ユーザとグループは一度だけ作成してください。中央ホームディレクトリサーバがない場合、ホームディレクトリを作成する必要がある場合があります。

4. PATH 環境変数（アップグレードの場合のみ) の設定を解除します。

   WebSphere MQ Recover Kit の以前のリリースをアップグレードする場合は、MQUSER PATH 環境変数の設定を変更しなければならない場合があります。Recovery Kit の以前のリリースでは、WebSphere MQ ソフトウェアのデフォルトのインストールの場所 (/opt/mqm) を含むように MQUSER PATH 環境変数を変更する必要がありました。この変更を以前のリリースで行った場合、Recover Kit が正常に動作するには、本バージョンでこの設定を解除する必要があります。

5. 必要なパッケージをインストールして、WebSphere MQ を各サーバ上にインストールします。

   MQSeries インストールでは、ライセンスのアクティベーション (`mqlicense_lnx.sh`) のために X11 ライブラリと Java が必要です。必要なソフトウェアパッケージをインストールします。

6. WebSphere MQ ソフトウェアと WebSphere MQ の Fix Pack を各サーバにインストールします。

   お使いの WebSphere MQ リリースの『WebSphere MQ ドキュメンテーション』に記載されている手順に従ってください。

7. **サーバ接続チャネルの認証。** WebSphere MQ バージョン 7.1 以降では、チャネルの認証方法が変更されました。デフォルトでは、MQADMIN ユーザー (mqm) は匿名認証 (パスワードなし) を行うことはできないため、リソース階層の作成に失敗します (WebSphere MQ 7.1 以前のリリースで作成されたキューマネージャの認証は引き続き有効)。WebSphere MQ 7.1 以降、MQADMIN ユーザーによる認証を許可するには、チャネルの認証を無効にする方法があります。WebSphere MQ 8.0 では、追加の変更が (現在の設定を取得する「display authinfo(system.default.authinfo.idpwos)」を runmqsc コマンドで実行して) system.default.authinfo.idpwosのauthinfoに対して必要となります。「reqdamd」のchckclnt 設定は、「optional」に変更する必要があります。MQADMIN ユーザーによる匿名認証の許可に失敗すると、リソースの作成時に「 **MQCONNX ended with reason code 2035** 」エラーが発生します。チャネルの認証については、WebSphere MQ ドキュメンテーション を参照してください。

8. MQ バージョン 7.1 以降を使用している場合、Queue Managerが使用されている指定チャネルで MQADMIN ユーザを有効化してください。

9. LifeKeeper と WebSphere MQ Recovery Kit を各サーバにインストールします。

   SPS をインストールする方法については、SIOS Protection Suite インストールガイド を参照してください。

10. 共有ストレージを準備し、その共有ストレージをマウントします。

    サポートされるファイルシステムレイアウトについては、サポートされるファイルシステムレイアウト を参照してください。ファイルシステムレイアウトとストレージタイプに応じて、ボリュームグループの作成、論理ボリューム、ファイルシステムの作成、または NFS 共有のマウントが必要です。

    以下は、NAS ストレージを使用するファイルシステムレイアウト 2 の例を示しています。

    ```
    node1:/var/mqm/qmgrs # mkdir TEST\!QM
    ```

```
node1:/var/mqm/qmgrs # mkdir ../log/TEST\!QM

node1:/var/mqm/qmgrs # mount 192.168.1.30:/raid5/vmware/shared_NFS/
TEST.QM/qmgrs ./TEST\!QM/

node1:/var/mqm/qmgrs # mount 192.168.1.30:/raid5/vmware/shared_NFS/
TEST.QM/log ../log/TEST\!QM/
```

11. *QMDIR* と *QMLOGDIR* の所有者とグループを MQUSER と MQGROUP に設定します。

    *QMDIR* と *QMLOGDIR* は、MQUSER と MQGROUP に所有されている必要があります。次のコマンドを使用して、ファイルシステム権限を設定してください。

    ```
    chown MQUSER QMDIR
    chgrp mqm QMDIR
    chown MQUSER QMLOGDIR
    chgrp mqm QMLOGDIR
    ```

    MQUSER、QMDIR、QMLOGDIR の値は、お使いのファイルシステムレイアウトと MQUSER のユーザ名によって異なります。手順 1 のシートを使用して、正しい値を決定してください。

    以下は、MQUSER mqm、およびキューマネージャ TEST.QM で、デフォルトの QMDIR および QMLOGDIR を使用する場合の例を示しています。

    ```
    node1:/var/mqm/qmgrs # chown mqm TEST\!QM/
    node1:/var/mqm/qmgrs # chgrp mqm TEST\!QM/
    node1:/var/mqm/qmgrs # chown mqm ../log/TEST\!QM/
    node1:/var/mqm/qmgrs # chgrp mqm ../log/TEST\!QM/
    ```

12. キューマネージャをプライマリサーバ上に作成します。

    『WebSphere MQ ドキュメント』に記載されている手順に従って、使用されている WebSphere MQ のバージョンに応じてキューマネージャを作成します。

    以下は、MQUSER mqm とキューマネージャ TEST.QM の例を示しています。

    ```
    node1:/var/mqm/qmgrs # su - mqm
    mqm@node1:~> crtmqm TEST.QM
    WebSphere MQ queue manager created.
    Creating or replacing default objects for TEST.QM.
    Default objects statistics : 31 created. 0 replaced. 0 failed.
    Completing setup.
    Setup completed.
    ```

    **注記**: すでに存在するキューマネージャを保護する場合、次の手順に従って、キューマネージャを共有ストレージに移動してください。

a. キューマネージャを停止します (`endmqm -i QUEUE.MGR.NAME`)。

b. キューマネージャディレクトリおよびキューマネージャログディレクトリの中身を手順 10 で作成した共有ストレージにコピーします。

c. グローバル設定ファイル (*mqs.ini*) およびキューマネージャ設定ファイル (*qm.ini*) を必要に応じて変更し、`QMDIR` と `QMLOGDIR` の新しい場所を設定します。

d. キューマネージャを起動して、機能を確認します (`strmqm QUEUE.MGR.NAME`)。

e. キューマネージャを停止します (`endmqm -i QUEUE.MGR.NAME`)。

13. オプション: 仮想 IP リソースをプライマリサーバ上の LifeKeeper で設定します。

SPS for Linux IP Recovery Kit 管理ガイド および SIOS Protection Suite インストールガイド に記載の手順とガイドラインに従ってください。

注記：サーバ接続によってのみキューマネージャにアクセスする場合、LifeKeeper 仮想 IP を設定する必要はありません。

14. リスナーオブジェクトをお使いの TCP アドレスおよびポートに合わせて変更します。

```
su - MQUSER
runmqsc QUEUE.MANAGER.NAME

alter LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1414)
IPADDR(192.168.1.100)
```

注記 :手順 13 で使用した IP アドレスと同じアドレスを使用して `IPADDR` の値を設定します。`IPADDR` は、WebSphere MQ がすべてのアドレスをバインドするよう設定してはなりません。

15. プライマリサーバ上のキューマネージャを起動します。

プライマリサーバ上で、キューマネージャ、コマンドサーバ (手動起動に設定されている場合)、リスナーを起動します。

```
su - MQUSER
strmqm QUEUE.MANAGER.NAME
strmqcsv QUEUE.MANAGER.NAME
runmqlsr -m QUEUE.MANAGER.NAME -t TCP &
```

16. キューマネージャが正常に起動したことを確認します。

```
su - MQUSER
echo 'display qlocal(*)' | runmqsc QUEUE.MANAGER.NAME
```

17. キューマネージャのスタンザを、バックアップサーバ上のグローバルキューマネージャ設定ファイル

*mqs.ini* に追加します。

**注記**: この手順は、ファイルシステムレイアウト 2 と 3 で必要です。

18. **オプション:** LifeKeeper テストキューを、プライマリサーバ上に作成します。

```
runmqsc TEST.QM

5724-B41 © Copyright IBM Corp. 1994, 2002. ALL RIGHTS RESERVED.

Starting MQSC for queue manager TEST.QM.

define qlocal(LIFEKEEPER.TESTQUEUE) defpsist(yes) descr('LifeKeeper
test queue')

1 : define qlocal(LIFEKEEPER.TESTQUEUE) defpsist(yes)
descr('LifeKeeper test queue')

AMQ8006: WebSphere MQ queue created.
```

19. LifeKeeper によってコマンドサーバを起動するには、プライマリサーバ上で、次のコマンドを使用してコマンドサーバの自動起動を無効にします。この手順を行わない場合、キューマネージャの起動時にコマンドサーバが自動的に起動されます。

```
su - MQUSER
runmqsc TEST.QM
ALTER QMGR SCMDSERV(MANUAL)
```

20. キューマネージャリソース階層をプライマリサーバ上に作成します。

    詳細については、 LifeKeeper の設定作業 を参照してください。

21. キューマネージャリソース階層をバックアップシステムに拡張します。

    詳細については、 LifeKeeper の設定作業 を参照してください。

21. 設定をテストします。

    HA WebSphere MQ インストールをテストするには、WebSphere MQ リソース階層のテスト に記載されている手順に従ってください。

# 6.8.4.1.3. リソース作成後の設定変更

SPS WebSphere MQ Recovery Kit は、WebSphere MQ コマンドを使用してキューマネージャを起動および停止します。以下に示すのは、このルールに対するいくつかの例外です。

---

QMDIR と QMLOGDIR の再配置

リスナーポートの変更

キューマネージャの IP の変更

# 6.8.4.1.3.1. QMDIR と QMLOGDIR の再配置

*QMDIR* と *QMLOGDIR* の場所が変わった場合、LifeKeeper の設定を変更する必要があります。設定は、次のいずれかの方法で変更できます。

1. キューマネージャのリソース階層を再作成する。

   この方法を行うには、キューマネージャ階層の削除と、キューマネージャ階層の作成が必要です。詳細については、「WebSphere MQ リソース階層の削除 」および「WebSphere MQ リソース階層の作成 」を参照してください。

2. ファイルシステム階層を手作業で新規作成し、その新しいファイルシステム階層を WebSphere MQ 階層に追加する。古いファイルシステム階層を WebSphere MQ 階層から削除し、古いファイルシステム階層を削除します。ファイルシステム階層を作成および削除する方法については、SIOS Protection Suite インストールガイド を参照してください。

# 6.8.4.1.3.2. リスナーポートの変更

キューマネージャのリスナーポートを変更するには、以下の手順を実行してください。

runmqsc 内のリスナーオブジェクトを変更した後、リスナーを停止し、起動してください。

```
su - MQUSER
runmqsc QUEUE.MANAGER.NAME

alter LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1415)
stop LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
start LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
```

詳細については、リソースプロパティ設定の編集 を参照してください。

# 6.8.4.1.3.3. キューマネージャの IP の変更

WebSphere MQ キューマネージャに関連する LifeKeeper で保護されている IP を変更するには、次の手順を実行します。

1. LifeKeeper GUI で新しい LifeKeeper 仮想 IP を作成します。
2. 新しい仮想 IP を WebSphere MQ 階層に追加します。
3. 古い仮想 IP を WebSphere MQ 階層から削除します。
4. 古い仮想 IP リソースを削除します。
5. 必要な場合、 runmqsc 内のリスナーオブジェクトを変更し、リスナーを再起動します。

```
su - MQUSER
runmqsc QUEUE.MANAGER.NAME

alter LISTENER(SYSTEM.DEFAULT.LISTENER.TCP) TRPTYPE(TCP) PORT(1414)
IPADDR(192.168.1.101)
stop LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
start LISTENER(SYSTEM.DEFAULT.LISTENER.TCP)
```

別の方法として、LifeKeeper の lk_chg_value 機能を使用して、IP を変更できます。詳細については、lk_chg_value(8) マニュアルページを参照してください。

# 6.8.4.1.4. WebSphere MQ の設定例

本セクションでは、一般的な WebSphere MQ 設定の定義と例を示します。各例には、LifeKeeper に適用される設定ファイルエントリが含まれています。

# 6.8.4.1.4.1. 共有ストレージ上の /var/mqm によるアクティブ / スタンバイ設定

アクティブ / スタンバイ設定では、Node1 はプライマリ LifeKeeper サーバです。このノードは、WebSphere MQ キューマネージャを保護します。すべてのストレージは、クラスタサーバ間の共有アレイに置かれます。Node2 は他のアプリケーションやサービスを処理することができますが、LifeKeeper から見た場合は、WebSphere MQ リソースのバックアップとしてのみ機能します。ディレクトリ */var/mqm* は、共有ストレージ上にあります。プライマリサーバは、処理可能な数のキューマネージャを上限なしで実行できます。

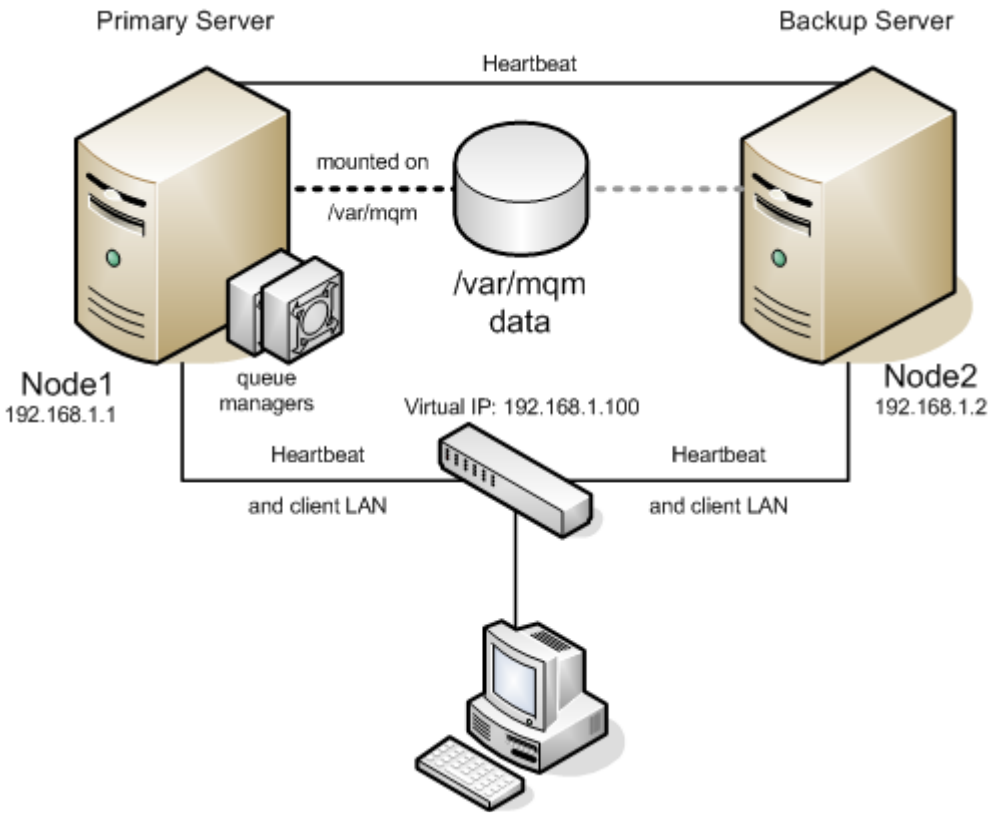



図 6 ローカルストレージにおけるアクティブ / スタンバイ設定

## 設定に関する注記

- クライアントは、LifeKeeper で保護され、クラスタ内のサーバ間で浮動するよう指定された IP 192.168.1.100 を使用して、WebSphere MQ サーバに接続します。
- ディレクトリ */var/mqm* は、共有ストレージ上にあります。
- 各キューマネージャにおいて、リスナーオブジェクトは一意のポート番号を含むよう変更されました。

# 6.8.4.1.4.2. NAS ストレージにおけるアクティブ / スタンバイ設定

アクティブ / スタンバイ設定では、Node1 はプライマリ LifeKeeper サーバです。このノードは、WebSphere MQ キューマネージャを保護します。すべてのストレージは、IP 10.0.0.100 の NAS サーバ上にあります。Node2 は他のアプリケーションやサービスを処理することができますが、LifeKeeper から見た場合は、WebSphere MQ リソースのバックアップとしてのみ機能します。ディレクトリ */var/mqm* は、NAS サーバの IP 10.0.0.100 から配置され、アクティブサーバのみにマウントされます。プライマリサーバは、処理可能な数のキューマネージャを上限なしで実行できます。

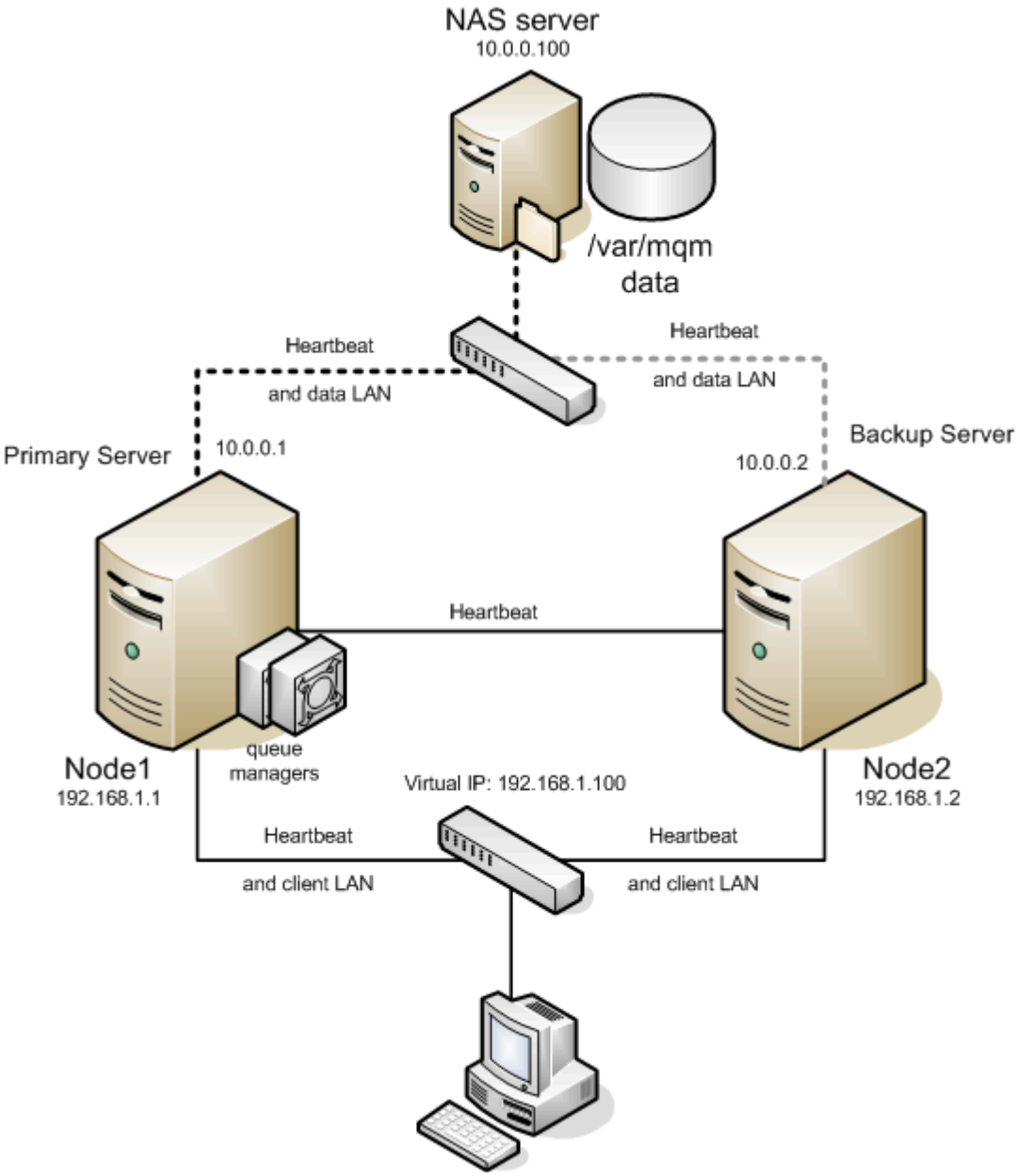



図 7 NFS ストレージにおけるアクティブ / スタンバイ設定

## 設定に関する注記

- クライアントは、LifeKeeper で保護され、クラスタ内のサーバ間で浮動するよう指定された IP 192.168.1.100 を使用して、WebSphere MQ サーバに接続します。
- ディレクトリ */var/mqm* は、NAS サーバ上にあります。
- アクティブサーバは、専用のネットワークインターフェースを使用して、ディレクトリ */var/mqm* を IP 10.0.0.100 を持つ NAS サーバからマウントします。
- 各ネットワークインターフェースには、ハートビートが設定されています。
- 各キューマネージャにおいて、リスナーオブジェクトは一意のポート番号を含むよう変更されました。

# 6.8.4.1.4.3. 共有ストレージにおけるアクティブ / アクティブ設定

以下のアクティブ / アクティブ設定において、Node1 と Node2 の両方が WebSphere MQ リソースに対するプライマリ LifeKeeper サーバです。各サーバは、互いにとってのバックアップサーバでもあります。この例では、Node1 はキューマネージャ QMGR1 のための共有ストレージアレイを保護します。Node2 はキューマネージャ QMGR2 のための共有ストレージアレイをプライマリサーバとして保護します。さらに、各サーバは互いのためのバックアップサーバとしても動作します。この例では、Node2 は Node1 上のキューマネージャ QMGR1 のためのバックアップであり、Node1 は、Node2 上のキューマネージャ QMGR2 のためのバックアップです。

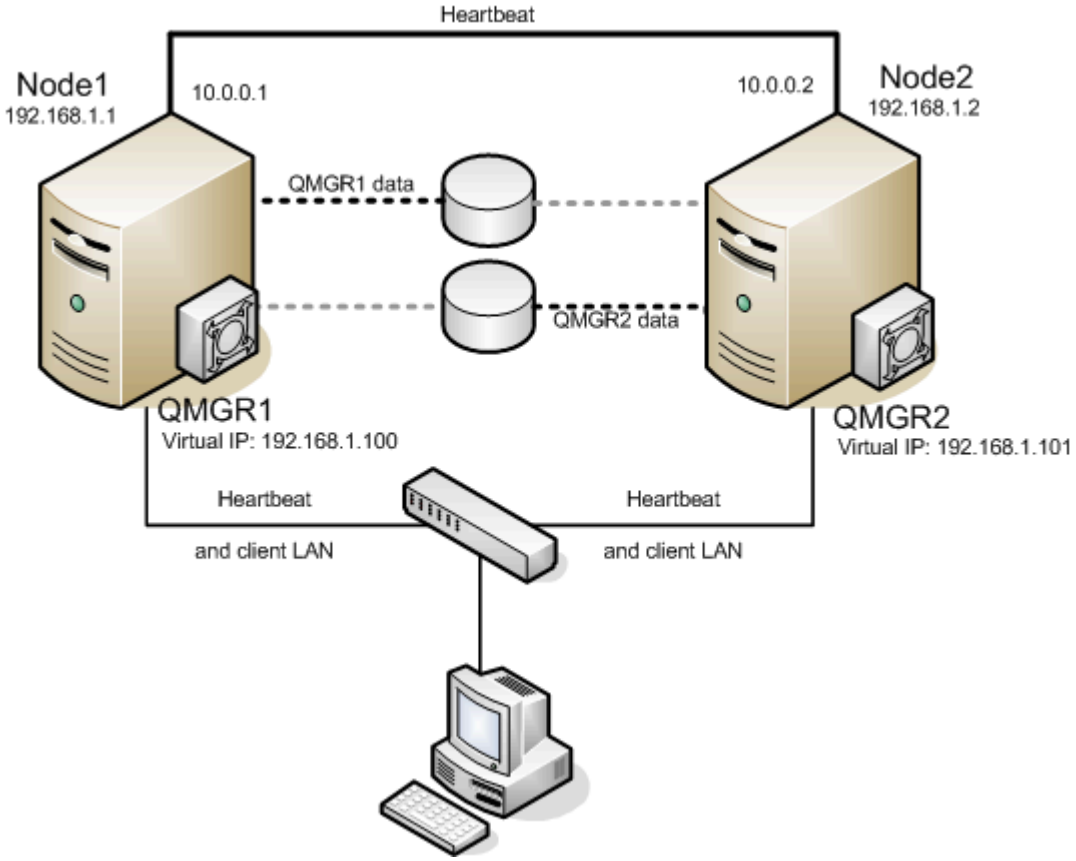



図 8 共有ストレージにおけるアクティブ / アクティブ設定

## 設定に関する注記

- クライアントは、LifeKeeper 仮想 IP 192.168.1.100 を使用して、キューマネージャ QMGR1 に接続します。
- クライアントは、LifeKeeper 仮想 IP 192.168.1.101 を使用して、キューマネージャ QMGR2 に接続します。
- 各ネットワークインターフェースには、ハートビートが設定されています。
- 各キューマネージャにおいて、リスナーオブジェクトは一意のポート番号を含むよう変更されまし

た。

- QMGR1 データは 2 つの論理ボリュームが設定された共有ストレージ上の 1 つのボリュームグループにあります。各論理ボリュームには、QMDIR または QMLOGDIR にマウントされるファイルシステムが含まれます。

- QMGR2 データは 2 つの論理ボリュームが設定された共有ストレージ上の 2 次ボリュームグループにあります。各論理ボリュームには、QMDIR または QMLOGDIR にマウントされるファイルシステムが含まれます。

# 6.8.4.1.4.4. NAS ストレージにおけるアクティブ / アクティブ設定

以下のアクティブ / アクティブ設定において、Node1 と Node2 の両方が WebSphere MQ リソースに対するプライマリ LifeKeeper サーバです。各サーバは、互いにとってのバックアップサーバでもあります。この例では、Node1 はキューマネージャ QMGR1 のための NFS マウントを保護します。Node2 はキューマネージャ QMGR2 のための NFS マウントをプライマリサーバとして保護します。さらに、各サーバは互いのためのバックアップサーバとしても動作します。この例では、Node2 は Node1 上のキューマネージャ QMGR1 のためのバックアップであり、Node1 は、Node2 上のキューマネージャ QMGR2 のためのバックアップです。

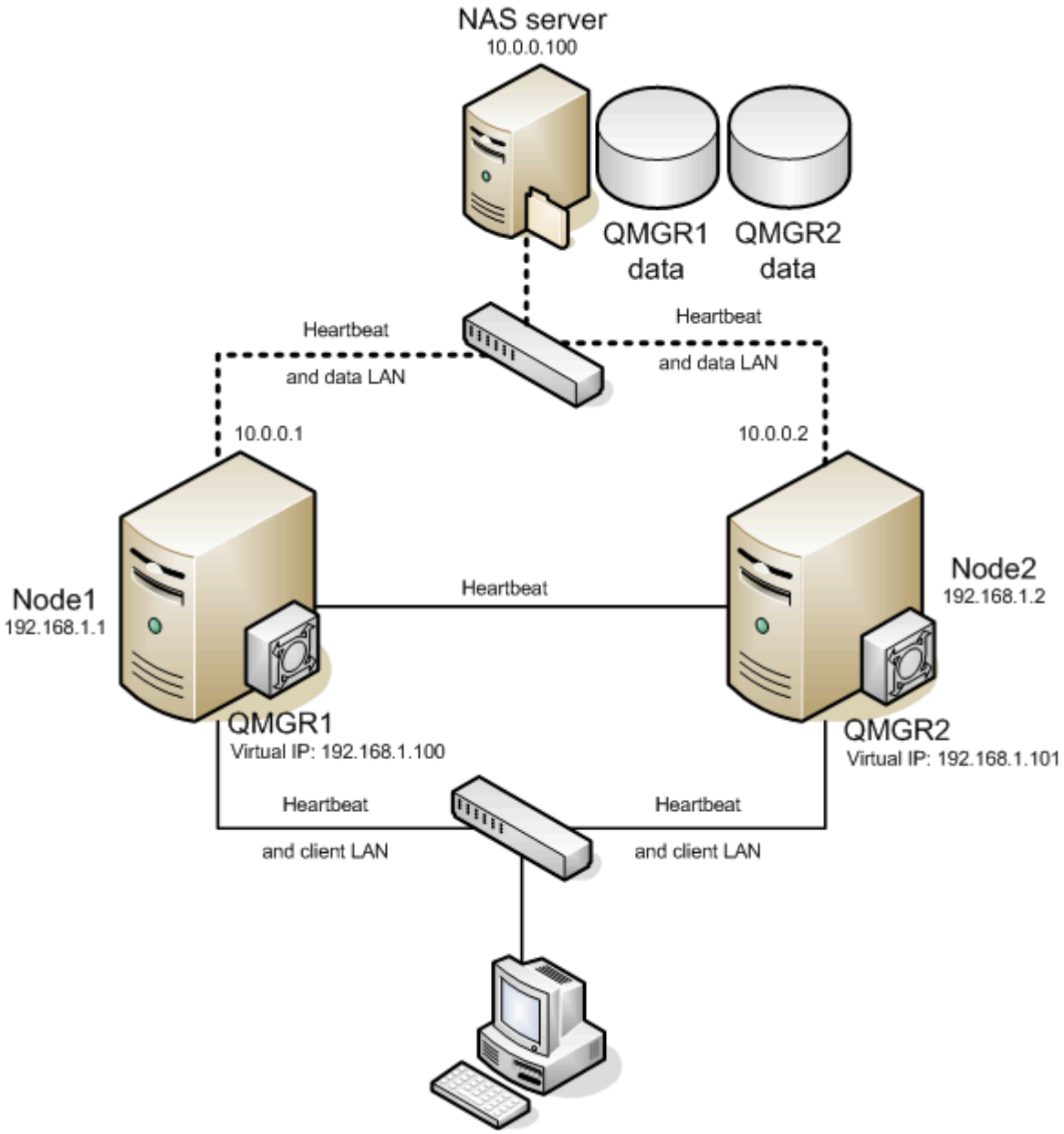



図 9 NFS ストレージにおけるアクティブ / アクティブ設定

## 設定に関する注記

- クライアントは、LifeKeeper 仮想 IP 192.168.1.100 を使用して、キューマネージャ `QMGR1` に接続します。
- クライアントは、LifeKeeper 仮想 IP 192.168.1.101 を使用して、キューマネージャ `QMGR2` に接続します。
- 各サーバは、NAS サーバにアクセスするために専用のネットワークインターフェースを持っています。
- 各ネットワークインターフェースには、ハートビートが設定されています。
- 各キューマネージャにおいて、リスナーオブジェクトは一意のポート番号を含むよう変更されました。
- `QMGR1` データは、NAS サーバ上の 2 つの NFS エクスポートに配置されています。これらのエクスポートは、*`QMDIR`* または *`QMLOGDIR`* にマウントされています。NAS サーバの IP は、10.0.0.100. です。
- `QMGR2` データは、NAS サーバ上の 2 つの NFS エクスポートに配置されています。これらのエクスポートは、*`QMDIR`* または *`QMLOGDIR`* にマウントされています。NAS サーバの IP は、10.0.0.100. です。

# 6.8.5. LifeKeeper の設定作業

SPS for Linux WebSphere MQ Recovery Kit の管理作業はすべて、LifeKeeper のグラフィカルユーザインターフェース (GUI) で実行できます。LifeKeeper の GUI では、WebSphere のリソースの設定、管理、監視の作業をガイド付きで行うことができます。

## 概要

以下の作業は、WebSphere MQ リソースインスタンス特有のものであり、Recovery Kit ごとに異なるため、本書で説明しています。

- リソース階層の作成 - WebSphere MQ リソース階層を作成します。
- リソース階層の削除 - WebSphere MQ リソース階層を削除します。
- リソース階層の拡張 - WebSphere MQ リソース階層をプライマリサーバからバックアップサーバに拡張します。
- リソース階層の拡張解除 - LifeKeeper クラスタ内にある 1 台のサーバの WebSphere MQ リソース階層を拡張解除（削除）します。
- リソースプロパティ設定の編集 – WebSphere MQ リソース階層の作成後に、LifeKeeper テストキュー、リスナー管理、停止タイムアウトなどの WebSphere MQ リソースパラメータを再設定します。

次の作業は、すべてのリカバリキットに共通する手順のため、SPS for Linux テクニカルドキュメンテーション の管理 セクションに記載されています。

- リソース依存関係の作成 。既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除 。リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service 。リソース階層を特定のサーバで in service にします。
- Out of Service 。リソース階層を特定のサーバで out of service の状態にします。
- プロパティの表示 / 編集 。特定のサーバでリソース階層のプロパティを表示または編集します。

**注記:** このセクションの残りの部分では、設定作業はすべて **[Edit]** メニューから実行されます。これらの設定作業の大半は次の方法でも実行できます。

- ツールバーから実行
- ステータス表示の左ペインのグローバルリソースを右クリックして実行

- ステータス表示の右ペインのリソースインスタンスを右クリックして実行

右クリックを使用すると、 **[Edit]** メニューを使用するときに要求される情報を入力せずにすみます。

# 6.8.5.1. WebSphere MQ リソース階層の作成

必要な設定作業を完了した後、次の手順に従って WebSphere MQ リソース階層を作成します。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。表示されたメニューから、 **[Create Resource Hierarchy]** を選択します。

   ***Create Resource Wizard*** ダイアログボックスが表示され、クラスタ内にインストールされ認識されているすべての Recovery Kit が、ドロップダウンリストボックスに一覧で示されます。

2. **[IBM WebSphereMQ]** を選択し、 **[Next]** をクリックします。

3. 以下の情報を入力するようにプロンプトが表示されます。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも **[Cancel]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| スイッチバックタイプ | **[Intelligent]** または **[Automatic]** を選択します。バックアップサーバへのフェイルオーバー後にサーバが復帰したときに、このサーバに WebSphere MQ インスタンスを戻す方法を指定します。スイッチバックタイプは、必要な場合、 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。<br><br>**注記:** スイッチバック方法は、WebSphere MQ リソースが使用する IP またはファイルシステムリソースのスイッチバック方法と一致する必要があります。この設定が WebSphere MQ リソースの設定と一致しない場合、作成処理によって、WebSphere MQ リソースの設定と一致するよう設定がリセットされます。 |
| Server | 階層を作成したい **サーバ** を選択します。 |
| Queue Manager Name | 保護したい WebSphere MQ キューマネージャを選択します。キューマネージャは、リソース階層を作成する前に作成しておく必要があります。すでに LifeKeeper によって保護されているキューマネージャはこの一覧から除外されます。キューマネージャは、グローバル設定ファイル *mqs.ini* から取得されます。 |
| Manage Listener | **[YES]** を選択すると、WebSphere MQ キューマネージャリスナーを保護および管理します。 **[NO]** を選択すると、LifeKeeper は WebSphere MQ リスナーを管理しません。<br><br>**注記:** この設定は後から変更できます。詳細については、リソースプロパティ設定の編集 を参照してください。 |
| Server Connection Channel | 接続テストに使用するサーバ接続チャネルを選択してください。デフォルトでは、チャネル SYSTEM.DEF.SVRCONN が使用されますが、MQ バージョン 7.1 以降では、MQ の チャネル認証を変更するには、デフォルト以外のチャネルが使用され、MQADMIN ユーザーが指定のチャネルで有効になっていることが必要になりました。<br><br>**注記:** リソースを作成する前に、サーバ接続チャネルが作成済みであることを確認して |

| | |
|---|---|
| | ください。詳細については、LifeKeeper 使用時の WebSphere MQ の設定 を参照してください。<br><br>**注記:** この設定は、後から変更できます。詳細については、リソースプロパティ設定の編集 を参照してください。 |
| Virtual IP | 階層に含める LifeKeeper の仮想 IP リソースを選択します。LifeKeeper 仮想 IP を WebSphere MQ 階層に含めない場合は、 **[None]** を選択してください。<br><br>**注記:** 仮想 IP が選択リストに表示されるには、プライマリノード上で ISP (アクティブ) である必要があります。 |
| IBM WebSphere MQ Resource Tag | LifeKeeper が提供するデフォルトのルートタグを選択するか、このサーバ上のリソースインスタンスに独自の名前を入力します。デフォルトは、キューマネージャの名前です。アルファベット、数字、および次の特別な文字を使用できます。- _ ./ |

4. **[Create]** をクリックします。 ***[Create Resource Wizard]*** が表示され、WebSphere MQ リソース階層が作成されます。LifeKeeper によって入力データが検査されます。問題が検出された場合、情報ボックスにエラーメッセージが表示されます。

5. 情報ボックスが表示され、WebSphere MQ リソース階層が正常に作成されたことと、フェイルオーバ保護を提供するには階層をクラスタ内の別のサーバに拡張する必要があることが示されます。 **[Next]** をクリックしてください。

6. **[Continue]** をクリックします。次に、LifeKeeper は、 ***[Pre-Extend Wizard]*** を開始します。リソース階層を別のサーバに拡張する方法の詳細については、 WebSphere MQ 階層の拡張 の手順 2 を参照してください。

# 6.8.5.2. WebSphere MQ 階層の拡張

この操作は **[Edit]** メニューから開始できます。または **[Create Resource Hierarchy]** オプションの動作が完了すると自動的に開始されます。その場合は、手順 2 を参照してください。

1. **[Edit]** メニューの **[Resource]** から **[Extend Resource Hierarchy]** を選択します。 **[Pre-Extend Wizard]** が表示されます。 **拡張** 操作に慣れていない場合は、 **[Next]** をクリックしてください。 **LifeKeeper** の **[Extend Resource Hierarchy]** のデフォルト値が分かっていて、入力と確認を省略する場合は **[Accept Defaults]** をクリックしてください。

2. ***[Pre-Extend Wizard]*** に以下の情報を入力します。

   注記: 最初の 2 つのフィールドは **[Edit]** メニューから **拡張** を開始した場合にのみ表示されます。

| フィールド | ヒント |
|---|---|
| Template Server | WebSphere MQ リソースが現在 in service になっているサーバを入力します。 |
| Tag to Extend | 拡張する WebSphere MQ リソースを選択します。 |
| Target Server | 拡張先のサーバを入力するか、選択します。 |
| Switchback Type | **[Intelligent]** または **[Automatic]** のいずれかを選択します。スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。<br><br>**注記:** スイッチバック方法は、WebSphere MQ リソースが使用する依存リソースのスイッチバック方法と一致する必要があることに注意してください。 |
| Template Priority | テンプレート階層の優先順位を選択または入力します。優先順位は、1 ~ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります(数字 1 が最高の優先順位に相当します)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。<br><br>**注記:** このフィールドは、階層をはじめて拡張するときにだけ表示されます。 |
| Target Priority | ターゲットサーバの階層の優先順位を選択または入力します。 |
| Queue Manager Name | この情報フィールドには、拡張されるキューマネージャの名前が表示されます。この値を変更することはできません。 |
| Root Tag | LifeKeeper は、ターゲットサーバ上の新しい WebSphere MQ リソースインスタンスに対して、デフォルトのタグ名を提供します。デフォルトのタグ名は、テンプレートサーバ上のこのリソースのタグ名と同一です。新しい名前を入力する |

| | |
|---|---|
| | 場合、ターゲットサーバ上で一意であることを必ず確認してください。アルファベット、数字、および以下の特別な記号を使用できます。- _ ./ |
| **注記:** リスナー管理、LifeKeeper テストキューの名前、シャットダウンタイムアウト値などのキューマネージャ管理パラメータはすべて、テンプレートサーバから取得されます。 | |

3. 拡張前のチェックが正常に終了したというメッセージが表示されたら、 **[Next]** をクリックします。

4. 拡張する階層に応じて、拡張されるリソースタグ（編集不可）を示す一連の情報ボックスが表示されます。 **[Extend]** をクリックします。

5. 「**Hierarchy extend operations completed (階層の拡張操作が完了しました)**」というメッセージが表示されたら、階層を別のサーバに拡張する場合は、 **[Next Server]** をクリックし、拡張作業が必要ない場合は、 **[Finish]** をクリックします。

6. 「**Hierarchy Verification Finished (階層の検査が完了しました)**」というメッセージが表示されたら、 **[Done]** をクリックします。

# 6.8.5.3. WebSphere MQ 階層の拡張解除

LifeKeeper クラスタ内にある 1 台のサーバからリソース階層を削除するには、次の手順を実行します。

1. **[Edit]** メニューの **[Resource]** から **[Unextend Resource Hierarchy]** を選択します。

2. **[Target Server]** で、WebSphere MQ リソースを拡張解除するターゲットサーバを選択します。WebSphere MQ リソースが現在 in service になっているサーバは選択できません。(このダイアログボックスは、右側のペインでリソースインスタンスを右クリックして、 **[Unextend]** 作業を選択した場合には、表示されません。) **[Next]** をクリックします。

3. 拡張解除する WebSphere MQ 階層を選択し、 **[Next]** をクリックします。 (このダイアログは、いずれかのペインでリソースインスタンスを右クリックして、 **[Unextend]** を選択した場合には表示されません。)

4. 拡張解除のために選択したターゲットサーバと WebSphere MQ リソース階層を確認する情報ボックスが表示されます。 **[Unextend]** をクリックします。

5. WebSphere MQ リソースが正常に拡張解除されたことを示す別の情報ボックスが表示されます。 **[Done]** をクリックして、 **[Unextend Resource Hierarchy]** メニューを終了します。

# 6.8.5.4. WebSphere MQ リソース階層の削除

WebSphere 階層を削除することによる、依存関係と保護対象のサービスへの影響を理解することは重要です。

- **依存関係**：リソース階層を削除する前に、依存関係を削除すると良い場合があります。依存しているファイルシステムは、削除されます。ファイルシステム以外の依存しているリソース (IP や Generic Application など) は、LifeKeeper GUI もしくは WebSphere MQ の削除スクリプトで削除しない限り、削除されません。WebSphere MQ キューマネージャの依存しているファイルシステムが LifeKeeper によって削除されないようにするには、WebSphere MQ 階層を削除する前に、手作業で依存関係を削除してください。

- **保護されるサービス**：WebSphere リソース階層を削除する前に out of service にした場合、そのキューマネージャに対する WebSphere デーモンは停止されます。階層が in service 中に削除された場合、WebSphere MQ デーモンは、階層が削除された後も (LifeKeeper 保護なしで) 実行とサービスの提供を継続します。

LifeKeeper 環境内のすべてのサーバからリソース階層を削除するには、以下の手順を実行します。

1. **[Edit]** メニューの **[Resource]** から **[Delete Resource Hierarchy]** を選択します。

2. **[Target Server]** で WebSphere MQ リソース階層を削除するターゲットサーバを選択し、 **[Next]** をクリックします。(このダイアログボックスは、いずれかのペインでリソースインスタンスを右クリックして [Delete Resource] を選択した場合には、表示されません。)

3. **[Hierarchy to Delete]** を選択します。(このダイアログは、右ペインまたは左ペインのリソースインスタンス上で右クリックして [Delete Resource] を選択した場合には表示されません。) **[Next]** をクリックします。

4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[Delete]** をクリックしてください。

5. WebSphere リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。

6. **[Done]** をクリックして終了します。

# 6.8.5.5. WebSphere MQ リソース階層のテスト

手動スイッチオーバを開始することによって、WebSphere MQ リソース階層をテストできます。これにより、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

**[Edit]** メニューの **[Resource]** から **[In Service]** を選択します。例えば、バックアップサーバで **In Service** 要求を実行すると、アプリケーション階層はプライマリサーバでは out of service になり、バックアップサーバで in service になります。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバになります。

**Out of Service** 要求を実行すると、アプリケーションは他のサーバで in service にならずに、out of service になります。

## 共有ストレージの設定のテスト

WebSphere MQ の共有ストレージの動作をテストするには、次の手順を行います。

1. プライマリサーバ上に一時的なテストキューを作成します。このとき、パーシスタンスはデフォルトの「**yes**」にします。

```
mqm@node1:/opt/mqm/samp/bin> runmqsc TEST.QM
5724-B41 © Copyright IBM Corp. 1994, 2002. ALL RIGHTS RESERVED.
Starting MQSC for queue manager TEST.QM.

define qlocal(TEST) defpsist(yes)
1 : define qlocal(TEST) defpsist(yes)
AMQ8006: WebSphere MQ queue created.
end

2 : end
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

2. 次のコマンドで、メッセージをプライマリノード上に作成されたテストキューに追加します。

```
mqm@node1:/opt/mqm/samp/bin> echo “HELLO WORLD on NODE1” | ./amqsput
TEST TEST.QM
Sample AMQSPUT0 start
target queue is TEST
Sample AMQSPUT0 end
```

3. 次のコマンドでテストキューを表示し、メッセージが格納されたことを確認します。

```
mqm@node1:/opt/mqm/samp/bin> ./amqsbcg TEST TEST.QM
```

「HELLO WORLD on NODE1」とその他の出力内容を含むメッセージが表示されます。次の行を探し、Persistence が 1 であることを確認します。

```
[…]
Priority :0 Persistence :1
[…]
```

4. リソース階層をスタンバイノードに切り替えます。

5. キューマネージャが現在アクティブになっているスタンバイサーバ上で、手順 3 を再び実行します。メッセージがスタンバイサーバ上でアクセス可能になります。アクセス可能でない場合、ストレージの設定を確認してください。

6. キューマネージャが現在アクティブになっているスタンバイサーバ上で、テストキューからメッセージを取得します。

```
mqm@node1:/opt/mqm/samp/bin> ./amqsget TEST TEST.QM
Sample AMQSGET0 start
message <HELLO WORLD on NODE1>
<now wait 15 seconds>
no more messages
Sample AMQSGET0 end
```

7. 手順 1 で作成したテストキューを削除します。

```
mqm@node1:/opt/mqm/samp/bin> runmqsc TEST.QM
5724-B41 © Copyright IBM Corp. 1994, 2002. ALL RIGHTS RESERVED.
Starting MQSC for queue manager TEST.QM.

delete qlocal(TEST)
1 : delete qlocal(TEST)
MQ8007: WebSphere MQ queue deleted.
end
2 : end
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

# 6.8.5.5.1. クライアント接続のテスト

クライアント接続をテストするには、次の手順を行ってください。

1. プライマリサーバ上で、amqsbcgc コマンドを使用してキューマネージャに接続します。

   ```
   export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/192.168.1.90(1414)'
   ```

   注記: IP 192.168.1.90 を LifeKeeper が保護するキューマネージャの仮想 IP に置き換えます。キューマネージャが 1414 以外のポートを使用している場合、1414 を使用されているポート番号に置き換えます。使用されているサーバ接続チャネルがデフォルトの SYSTEM.DEF.SVRCONN チャネルではない場合、SYSTEM.DEF.SVRCONN を使用されているサーバ接続チャネルに置き換えます。

   次の出力が表示されます。

   ```
   mqm@node1:/opt/mqm/samp/bin> ./amqsbcgc LIFEKEEPER.TESTQUEUE TEST.QM

   AMQSBCG0 - starts here
   **********************

   MQOPEN - 'LIFEKEEPER.TESTQUEUE'

   No more messages
   MQCLOSE
   ```

   次のようなメッセージが表示された場合、テストキュー LIFEKEEPER.TESTQUEUE が設定されていません。「LifeKeeper 使用時の WebSphere MQ の設定」の説明に従ってテストキューを作成し、テストを再度実行します。

   ```
   AMQSBCG0 - starts here
   **********************

   MQOPEN - 'LIFEKEEPER.TESTQUEUE'
   MQOPEN failed with CompCode:2, Reason:2085
   ```

2. リソース階層のスイッチオーバを実行します。

3. 手順 1 を前と同じサーバ (スイッチサーバ後はスタンバイサーバ) に対して再度実行します。

## PUT/GET テスト実行の確認

WebSphere MQ Recovery Kit が PUT/GET テストを含むすべてのチェックを実行するかどうかを確認するには、次の手順を行ってください。

1. キューマネージャがいずれかのサーバ上で in service (ISP) であることを確認します。

2. キューマネージャのログレベルをログレベルの変更 の説明に従って「**FINE**」に変更します。

3. キューマネージャが実行されているマシン上でログダイアログを開き、次のチェックが実行されるまで待ちます (最大 2 分間)。

4. ログを分析し、1 つのテストも欠かすことなく、すべてのチェックが実行されたことを確認します。PUT/GET テストは、次の理由で省略されることがあります。

   a. LifeKeeper テストキューが設定されていない (その場合、「LifeKeeper テストキュー名の変更」の説明に従ってテストキューを設定します)。

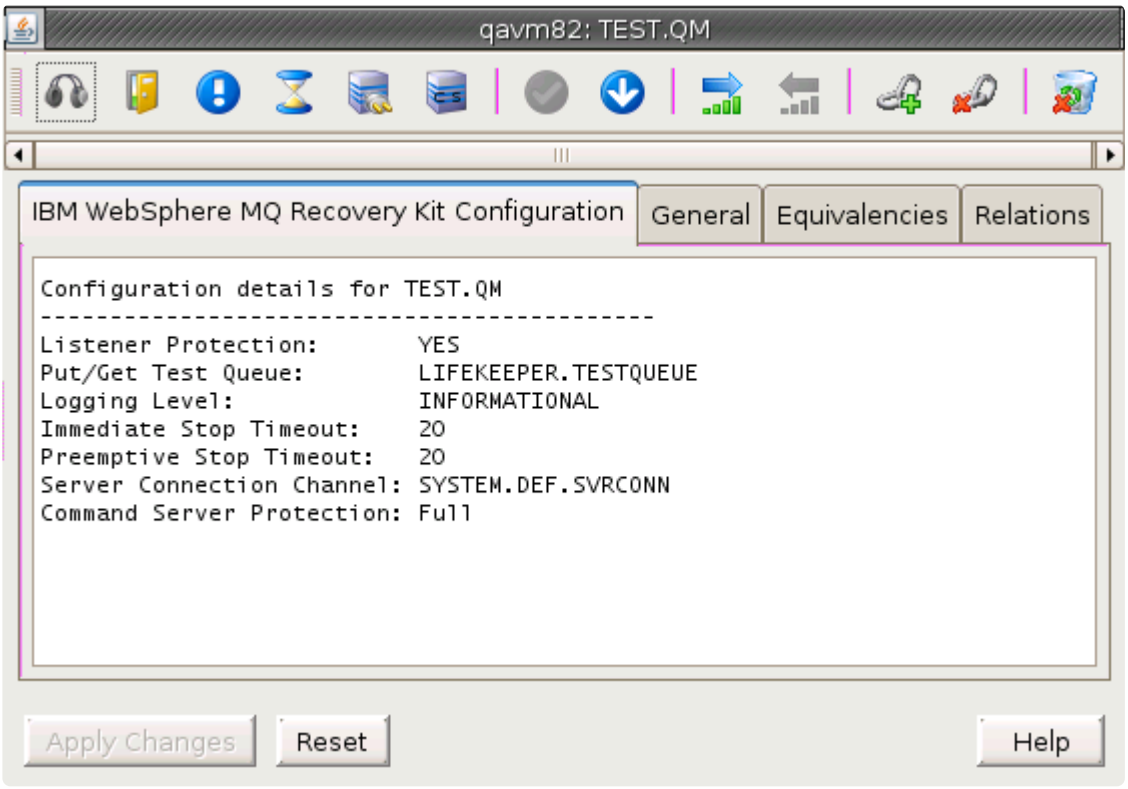
   b. LifeKeeper テストキューが存在しない (その場合、「LifeKeeper 使用時の WebSphere MQ の設定」の説明に従ってテストキューを作成します)。

   c. 変更された amqsget(c) 実行可能ファイルが使用できない (その場合、C コンパイラをインストールし、スクリプト */opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/compilesamples* を再実行します)。

5. デバッグレベルを「INFORMATIONAL」に再度設定します。

# 6.8.5.6. リソースのプロパティの表示

IBM WebSphere MQ リソースのプロパティを表示するには、表示するリソース / サーバの組み合わせのアイコンを右クリックします。リソースコンテクストメニューが表示されたら、 **[Properties]** をクリックします。以下のダイアログが表示されます。



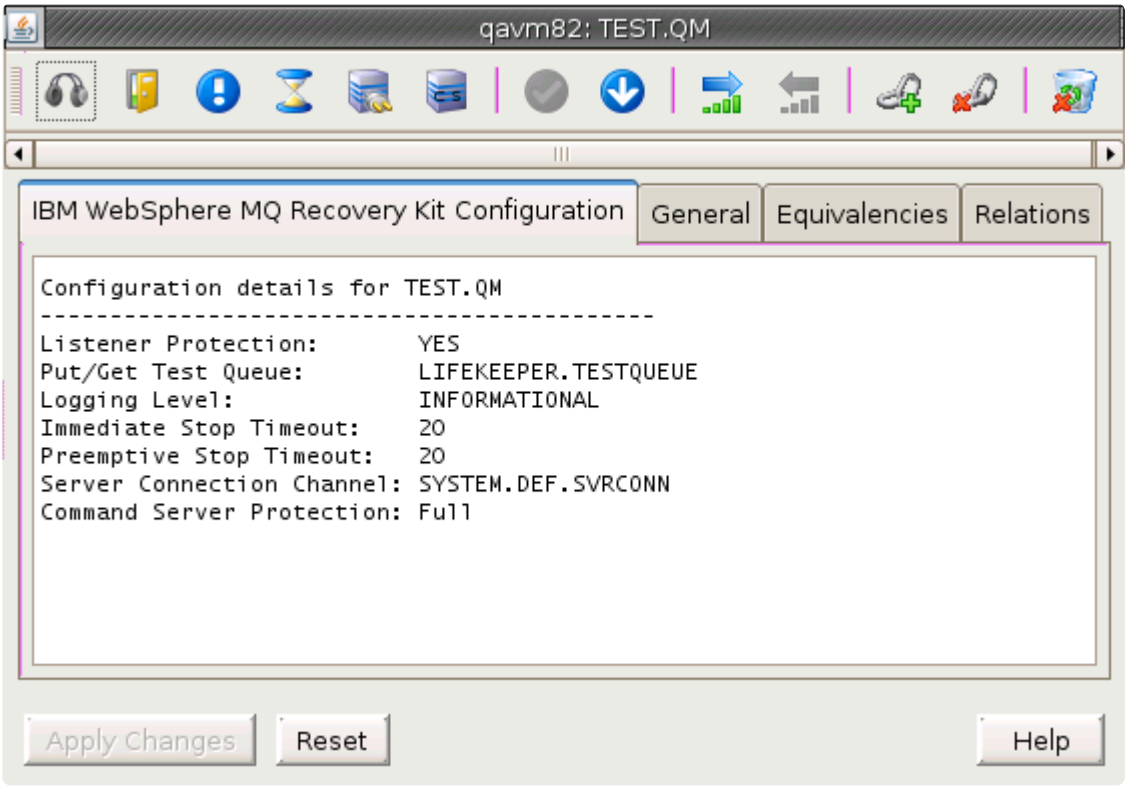

リソースのプロパティは、プロパティ画面(有効になっている場合) に表示されます。プロパティを表示するグローバルリソースのアイコンを右クリックすることもできます。 **[Resource Context]** メニュー が表示されたら、 **[Properties]** をクリックします。ダイアログが表示されたら、表示するリソースが存在するサーバを [Server] リストから選択します。

# 6.8.5.7. リソースプロパティ設定の編集

WebSphere MQ のプロパティ画面では、プロパティパネル (有効な場合) を使用して特定の WebSphere MQ リソースの詳細設定を表示および変更できます。特定の WebSphere MQ リソース設定プロパティは、 **[Resource Context Menu]** を使用して変更することもできます。

LifeKeeper GUI の [Properties Panel] から開く WebSphere MQ の [Configuration Properties] 画面で設定の詳細を編集する前に、GUI の [Properties Panel] が有効になっていることを確認する必要があります。GUI の [Properties Panel] を有効化するには、 **[View]** 、 **[Properties Panel]** を順にクリックします (チェックマークを入れて有効にします)。有効化後に **WebSphere MQ** リソース上で左クリックすると、その設定の詳細が LifeKeeper GUI の [Properties Panel] に表示されます。

以下に、LifeKeeper GUI の [Properties Panel] に表示される WebSphere MQ リソースのプロパティ画面の例を示します。




プロパティ画面にはタブが 4 つあります。1 つ目のタブ、 **[IBM WebSphere MQ Recovery Kit Configuration]** には、WebSphere MQ の各リソースの設定情報が表示され、リソースごとのアイコンを使用して変更できます。他の 3 つのタブは、LifeKeeper のすべてのリソースタイプで共通のため、その内容は SPS for Linux テクニカルドキュメンテーション の [Resource Properties] ダイアログ で説明されています。

次の表には、WebSphere MQ リソース固有のアイコンと、アイコンをクリックすることで変更可能な設定コンポーネントを示しています。

| | | |
|---|---|---|
| | リスナー保護の設定 | IBM WebSphere MQ リスナーの保護を、その他の保護対象の IBM WebSphere MQ キューマネージャコンポーネントに含めるかどうかを指定できます。 |

| | | |
|---|---|---|
| | **PUT/GET テストキューの設定** | 保護対象のキューマネージャに対して IBM WebSphere MQ Recovery Kit が PUT/GETテストを実行するために使用するキューの名前を変更できます。 |
| | **ログレベルの設定** | IBM WebSphere MQ Recovery Kit が保護対象のキューマネージャに使用するログレベルを変更できます。 |
| | **シャットダウンタイムアウトの設定** | 保護対象の IBM WebSphere MQ キューマネージャの即時シャットダウンタイマーおよびプリエンプティブシャットダウンタイマーのタイムアウトを秒単位で変更できます。 |
| | **サーバ接続チャネルの設定** | 保護対象の IBM WebSphere MQ キューマネージャにおいて、クライアント接続および PUT/GET テストに使用するサーバ接続チャネルを変更できます。 |
| | **コマンドサーバ保護の設定** | 保護対象の IBM WebSphere MQ キューマネージャのコマンドサーバコンポーネントの保護/リカバリレベルを指定できます。 |

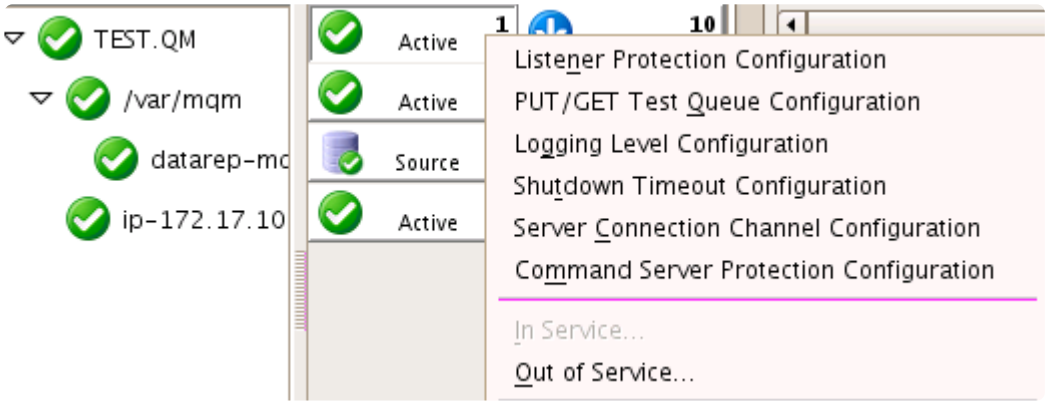
上記の各設定オプションの詳細を以下に示します。

| | |
|---|---|
| **リスナー管理** | LifeKeeper によるキューマネージャの保護の有無を指定します。リスナー管理が無効 (値: NO) の場合、LifeKeeper はリスナーを監視しないため、LifeKeeper リカバリアクションを発動させることなくリスナーを停止できます。リスナー管理が有効 (値: YES) の場合、LifeKeeper はリスナーを監視し、リスナーが実行中でない場合にリスナーを再起動します。リカバリが失敗すると、WebSphere MQ 階層のバックアップサーバへのフェイルオーバが開始されます。 |
| **LifeKeeper のテストキュー** | LifeKeeper は、キューマネージャの動作を監視するために PUT/GET テストを実行します。WebSphere MQ Recovery Kit は、 **専用の** テストキューに対してメッセージを追加および再取得します。障害が検出された場合、リカバリやフェイルオーバは実行されません。その代わり、Recovery Kit は受信するよう登録可能なイベントを送信します。これらのイベントは、putgetfail および putgetcfail と呼ばれます。これらのイベントに対応するため、ディレクトリ */opt/LifeKeeper/events/mqseries/putgetfail* および */opt/LifeKeeper/events/mqseries/putgetcfail* に通知スクリプトを追加できます。<br><br>**注記 1** : LifeKeeper テストキューがキューマネージャに設定されていない場合、PUT/GET テストは行われません。リカバリやフェイルオーバは実行されません。<br><br>**注記 2** : リスナーが保護されている場合、2 回目のクライアント接続チェックが実行されます。このチェックが失敗すると、キューマネージャのリカバリまたはフェイルオーバが試行されます。 |
| **ログレベル** | WebSphere MQ Recovery Kit のログレベルを 4 つのプリセット値に設定できます。<br><br>• ERROR<br><br>このログレベルでは、エラーのみがログに記録されます。情報メッセージはログに記録されません。<br><br>• INFORMATIONAL (デフォルト) |

| | |
|---|---|
| | このログレベルでは、リソースの起動、停止、リカバリに関する LifeKeeper の情報メッセージがログに記録されます。<br><br>• DEBUG<br><br>このログレベルでは、LifeKeeper の情報メッセージと、restore、remove、および recovery スクリプト内のすべての WebSphere MQ コマンドによるコマンド出力がログ記録されます。<br><br>• FINE<br><br>このログレベルでは、start、stop、recovery、および quickCheck スクリプトで実行された WebSphere MQ コマンドによるすべてのコマンド出力がログ記録されます。その他のデバッグメッセージもログ記録されます。<br><br>このデバッグレベルは、デバッグ時のみ設定することを推奨します。`quickCheck` アクションもログ記録されるため、WebSphere MQ キューマネージャで `quickCheck` が実行されるたびにログファイルが一杯になります。<br><br>デフォルトは、 *INFORMATIONAL* です。これは、他のリカバリキットの通常の LifeKeeper ログと同等です。<br><br>**注記:** ログレベルの設定に関わらず、起動、停止、リカバリ、またはチェックルーチン中に発生した WebSphere MQ エラーは、直前に実行されたコマンドの完全な出力とともに常にログに記録されます。 |
| **停止タイムアウト値** | WebSphere MQ Recovery Kit は、キューマネージャを 3 段階で停止します。<br><br>1. 即時停止<br>2. プリエンプティブ停止<br>3. すべてのキューマネージャプロセスを停止<br><br>指定されたタイムアウト値は、Recovery Kit が手順 1 と 2 において正常完了を待機する時間を決定します。このタイムアウトに達すると、シャットダウンプロセスの次の手順が開始します。即時シャットダウンタイムアウトとプリエンプティブシャットダウンタイムアウトのデフォルト値は ***20*** 秒です。 |
| **サーバ接続チャネル** | WebSphere MQ Recovery Kit では、サーバ接続チャネルを指定できます。キットはチャネル `SYSTEM.DEF.SVRCONN` をデフォルトで使用しますが、リソース作成中またはリソース作成後の任意のタイミングで代替チャネルを指定することができます。 |
| **コマンドサーバ** | WebSphere MQ Recovery Kit では、保護対象のキューマネージャのコマンドサーバコンポーネントについて、2 つのレベルの保護およびリカバリを提供します。すなわち、 **Full** レベルと **Minimal** レベルです。<br><br>**Full 保護の場合、コマンドサーバは起動、停止、監視、および復旧され、またリカバリが失敗した場合はフェイルオーバされます。 Full保護のリカバリ手順は、次のとおりです。** |

| | |
|---|---|
| | • コマンドサーバプロセスのみを再起動する。<br>• 上記が失敗した場合、コマンドサーバプロセスを含むキューマネージャ全体を再起動する。<br>• 上記のコマンドサーバ再起動がいずれも失敗した場合、スタンバイノードへのフェイルオーバを開始する。<br><br>**Minimal** 保護の場合、コマンドサーバはリストア中の起動、または、削除中の停止のみ実行されます。コマンドサーバの監視とリカバリは行われません。<br><br>**注記:** リストア中の **Recovery Kit** によるコマンドサーバの起動は、キューマネージャ **SCMDSERV** パラメータが手動起動に設定されている場合のみ行われます。リカバリ中、障害の発生したコマンドサーバの再起動は **SCMDSERV** の設定に関わらず行われます。ただし、コマンドサーバの保護レベルが **Minimal**に設定されている場合を除きます。 |

前述したように、これらの WebSphere MQ リソース設定コンポーネントは、プロパティーパネルでリソース固有のアイコンを使用するか **[Resource Context Menu]** を使用して変更できます。




上記のパラメータは、**LifeKeeper GUI** またはコマンドラインユーティリティを使用して、キューマネージャごとに独立して設定できます。

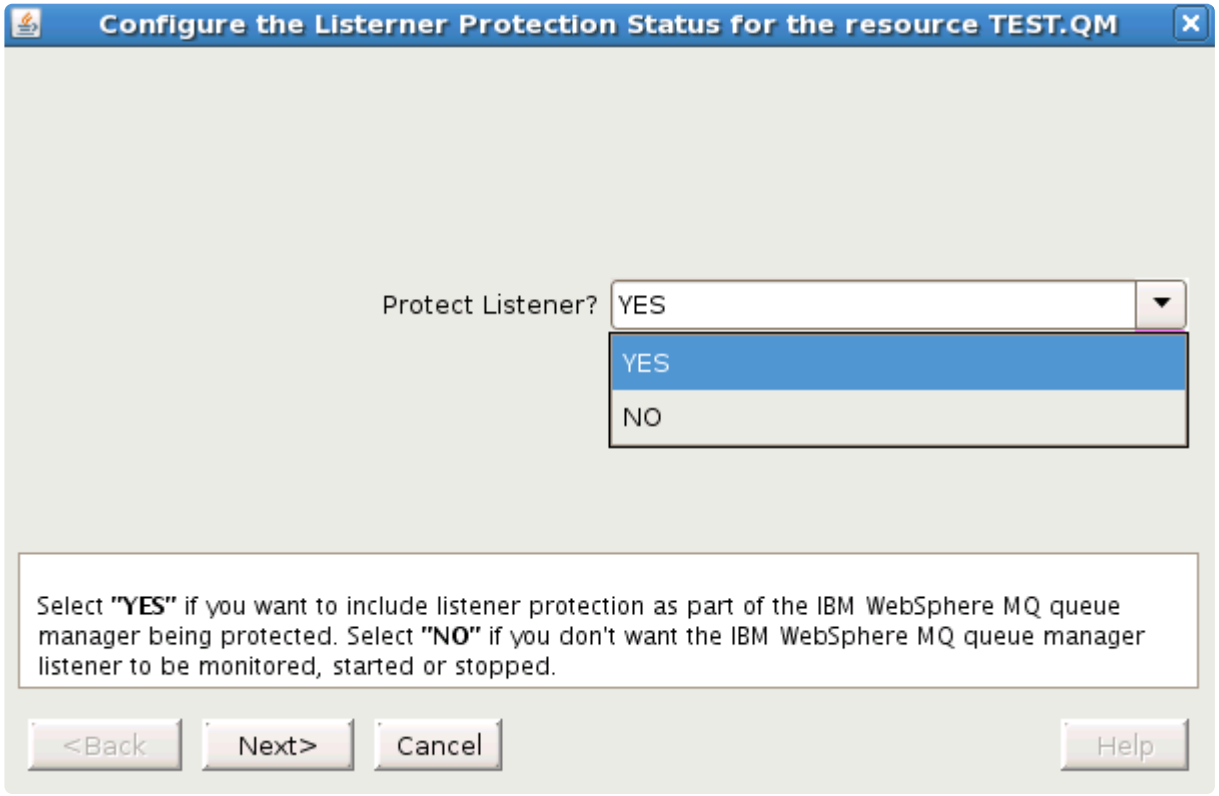
コマンドラインでパラメータを設定するには、次のスクリプトを使用します。

```
$LKROOT/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam
```

# 6.8.5.7.1. リスナー保護の有効化 / 無効化

## GUI

まず、 **WebSphere MQ** のリソースプロパティ画面 または上述の リソースコンテクストメニュー を開きます。リスナー保護の値を変更するには、リソースが in service である必要があります。 **[Listener Protection Configuration]** アイコンまたはメニュー項目をクリックします。以下のダイアログが表示されます。




**[YES]** を選択すると、LifeKeeper は WebSphere MQ リスナーを起動、停止、および監視します。 **[NO]** を選択すると、LifeKeeper は WebSphere MQ リスナーを起動、停止、および監視しません。 **[Next]** をクリックします。リスナー保護を有効化または無効化するかを確認するメッセージが表示されたら、 **[Continue]** をクリックします。リスナー管理を有効化する場合、LifeKeeper GUI はリスナーがすでに実行中であるかチェックします。実行されていない場合、リスナーを起動します。リスナーの起動に成功すると、LifeKeeper GUI は、クラスタ内の各サーバ上でリスナー管理を有効にします。リスナーが実行中でなく、起動できなかった場合、LifeKeeper GUI はクラスタ内の各サーバ上でリスナー管理を有効にしません。

## コマンドライン

LifeKeeper リスナー管理をコマンドラインで設定するには、次のコマンドを使用します。

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p LISTENERPROTECTION -v YES
```

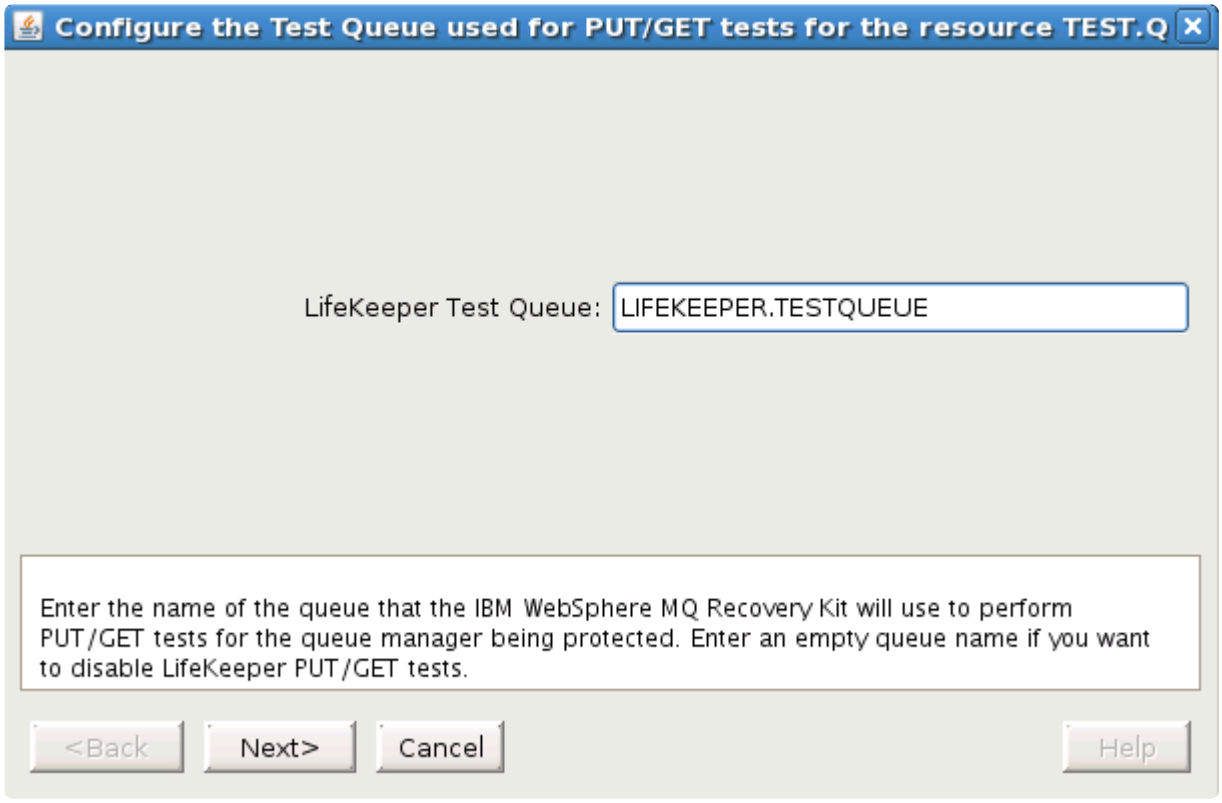
このコマンドを使用すると、キューマネージャ TEST.QM (-i) において、クラスタ (-c) 内の各ノード上の LifeKeeper リスナー管理 (-p) が **YES** (-v) に設定 (-s) されます。

**注記:** キューマネージャの名前（`-i`）または LifeKeeper TAG（`-t`）の名前のいずれかを使用できます。

# 6.8.5.7.2. LifeKeeper テストキュー名の変更

## GUI

まず、 **WebSphere MQ** のリソースプロパティ画面 または上述のリソースコンテクストメニューを開きます。 **[PUT/GET TESTQUEUE Configuration]** アイコンまたはメニュー項目をクリックします。以下のダイアログが表示されます。




LifeKeeper テストキューの名前を入力し、 **[Next]** をクリックします。新しい LifeKeeper テストキューを設定するかを確認するメッセージが表示されたら、 **[Continue]** をクリックします。LifeKeeper GUI により、クラスタ内の各サーバ上で選択されたキューマネージャに LifeKeeper テストキューが設定されます。テストキューに空の値を設定した場合、PUT/GET テストは実行されません。

## コマンドライン

LifeKeeper テストキューをコマンドラインで設定するには、次のコマンドを使用します。

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p TESTQUEUE -v "LIFEKEEPER.TESTQUEUE"
```

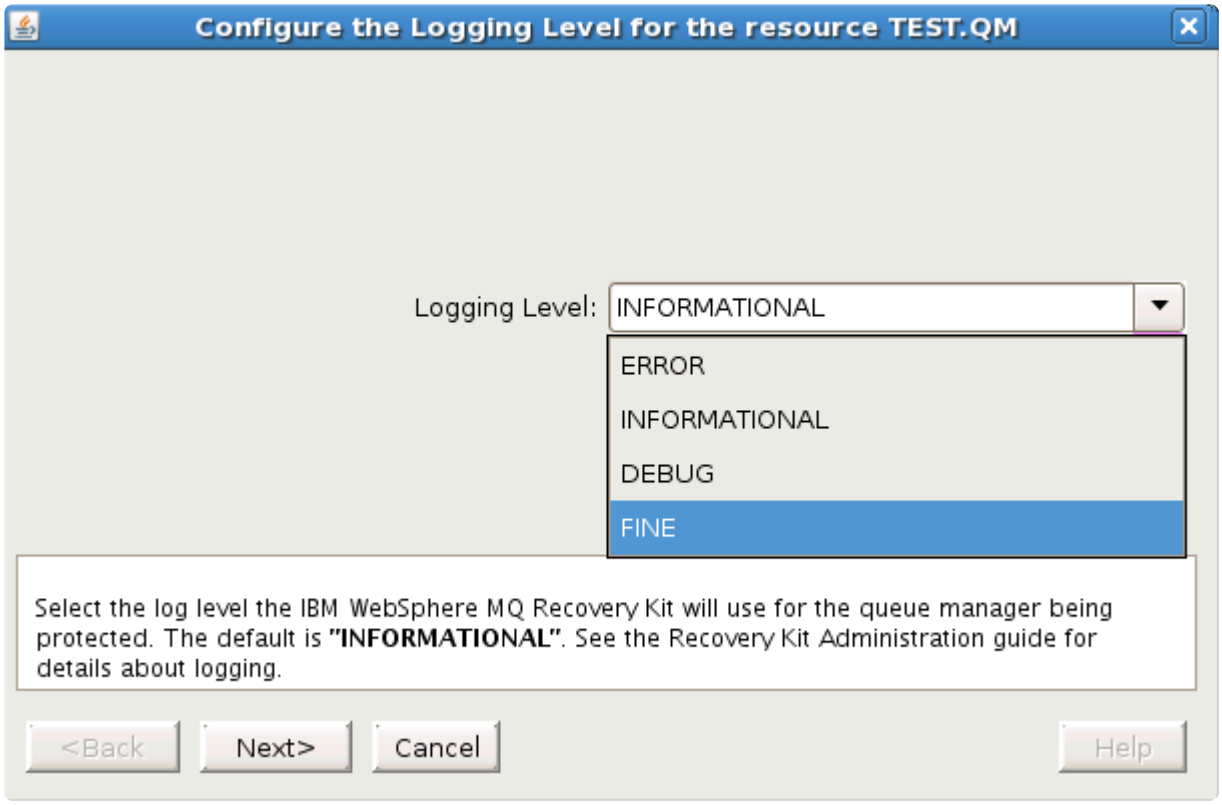
このコマンドを使用すると、キューマネージャ TEST.QM （-i）において、クラスタ (-c) 内の各ノード上の LifeKeeper テストキュー (-p) が LIFEKEEPER.TESTQUEUE （-v）に設定 (-s) されます。

注記: キューマネージャの名前 (-i) または LifeKeeper TAG (-t) の名前のいずれかを使用できます。

# 6.8.5.7.3. ログレベルの変更

## GUI

まず、 **WebSphere MQ** のリソースプロパティ画面 または上述の リソースコンテクストメニュー を開きます。 **[Logging Level Configuration]** アイコンまたはメニュー項目をクリックします。以下のダイアログが表示されます。




**[Logging Level]** を選択して **[Next]** をクリックしてください。新しい LifeKeeper ログレベルを設定するかを確認するメッセージが表示されたら、 **[Continue]** をクリックします。LifeKeeper GUI により、クラスタ内の各サーバ上で選択されたキューマネージャに LifeKeeper ログレベルが設定されます。

## コマンドライン

LifeKeeper ログレベルをコマンドラインで設定するには、次のコマンドを使用します。

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p DEBUG -v DEBUG
```

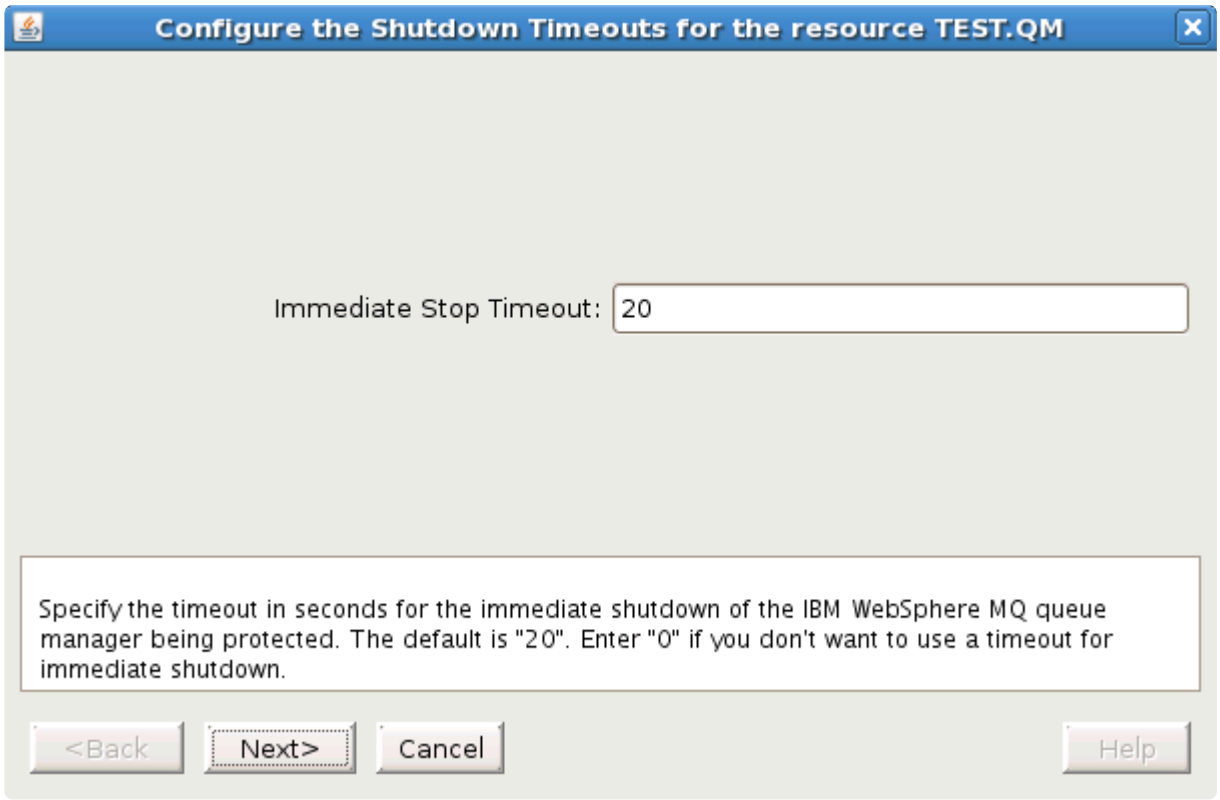
このコマンドを使用すると、キューマネージャ `TEST.QM` （`-i`）において、クラスタ (-c) 内の各ノード上の LifeKeeper ログレベル (-p) が `DEBUG` （`-v`）に設定 (-s) されます。

注記: キューマネージャの名前 (-i) または LifeKeeper TAG (-t) の名前のいずれかを使用できます。

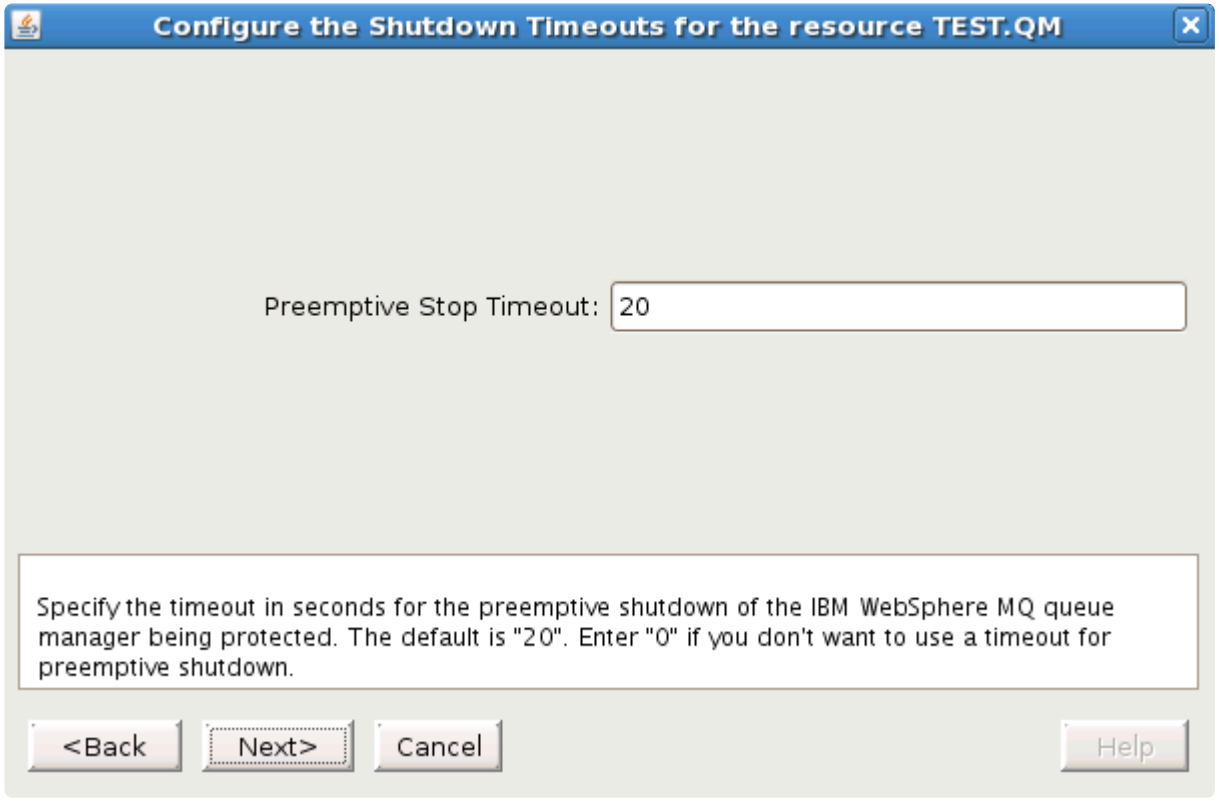
# 6.8.5.7.4. シャットダウンタイムアウト値の変更

## GUI

まず、WebSphere MQ のリソースプロパティ画面または上述のリソースコンテクストメニューを開きます。 **[Shutdown Timeout Configuration]** アイコンまたはメニュー項目をクリックします。以下のダイアログが表示されます。




即時シャットダウンタイムアウト値を秒単位で入力し、 **[Next]** をクリックします。即時シャットダウンタイムアウトを無効にするには、 **0** を入力します。以下のダイアログが表示されます。




プリエンプティブシャットダウンタイムアウト値を秒単位で入力し、 **[Next]** をクリックします。プリエンプティブシャットダウンタイムアウトを無効にするには、0 を入力します。新しい LifeKeeper タイムアウトパラメータを設定するかを確認するメッセージが表示されたら、 **[Continue]** をクリックします。LifeKeeper GUI により、クラスタ内の各サーバ上で、LifeKeeper 即時およびプリエンプティブタイムアウト値が設定されます。

## コマンドライン

LifeKeeper **プリエンプティブシャットダウン** タイムアウト値をコマンドラインで設定するには、次のコマンドを使用します。

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p PREEMPTIVE_TIMEOUT -v 20
```

このコマンドを使用すると、キューマネージャ TEST.QM (-i) において、クラスタ (-c) 内の各ノード上の LifeKeeper プリエンプティブシャットダウンタイムアウト (-p) が *20* 秒 (-v) に設定 (-s) されます。

LifeKeeper **即時シャットダウン** タイムアウト値をコマンドラインで設定するには、次のコマンドを使用します。

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p IMMEDIATE_TIMEOUT -v 20
```

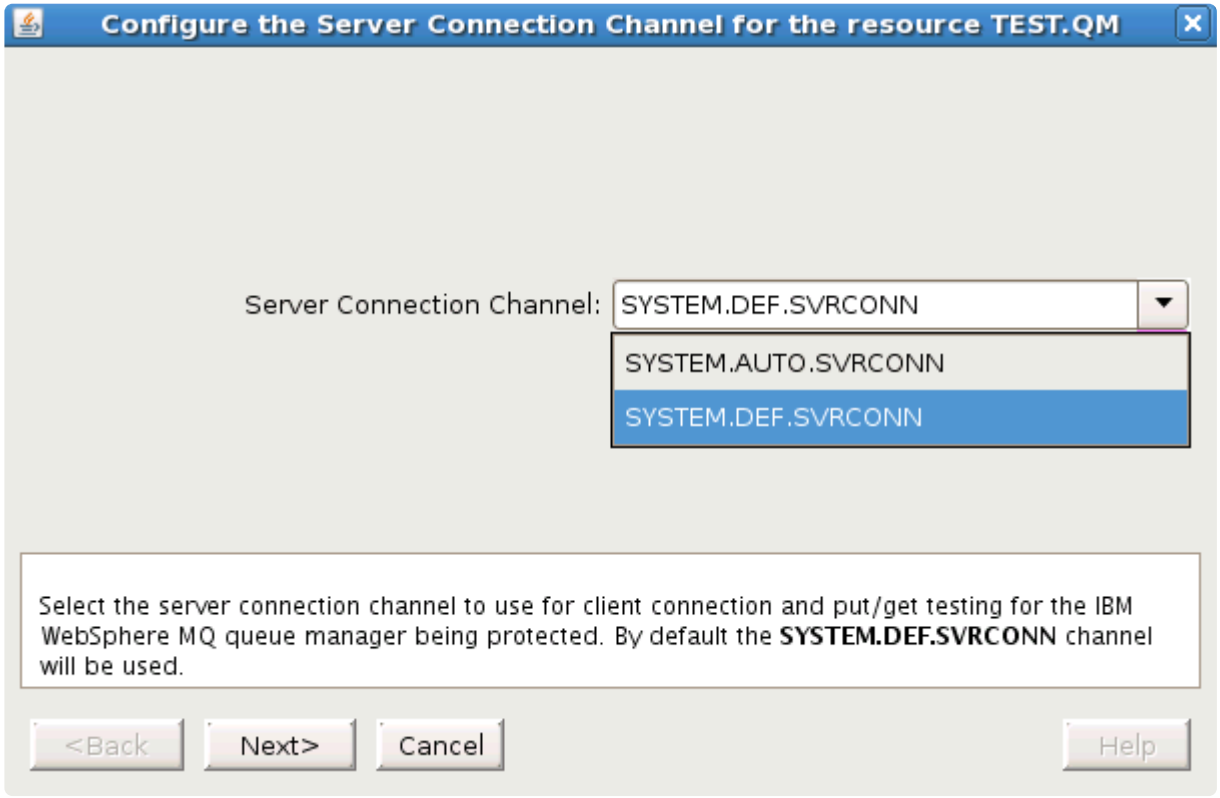
このコマンドを使用すると、キューマネージャ TEST.QM (-i) において、クラスタ (-c) 内の各ノード上の LifeKeeper 即時シャットダウンタイムアウト (-p) が *20 秒* (-v) に設定 (-s) されます。

**注記**：キューマネージャの名前 (-i) または LifeKeeper TAG (-t) の名前のいずれかを使用できます。

# 6.8.5.7.5. サーバ接続チャネルの変更

## GUI

まず、WebSphere MQ のリソースプロパティ画面または上述のリソースコンテクストメニューを開きます。 **[Server Connection Channel]** の値を変更するには、リソースが in service である必要があります。次に、 **[Server Connection Channel Configuration]** アイコンまたはメニュー項目をクリックします。以下のダイアログが表示されます。




使用するサーバ接続チャネルを **[Server Connection Channel]** で選択し、 **[Next]** をクリックします。新しい **サーバ接続チャネル** に変更するかどうかを確認するメッセージが表示されたら、 **[Continue]** をクリックします。LifeKeeper GUI により、クラスタ内の各サーバ上で選択したキューマネージャのサーバ接続チャネルが設定されます。

## コマンドライン

サーバ接続チャネルをコマンドラインで設定するには、次のコマンドを使用します。

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p CHANNEL -v LK.TEST.SVRCONN
```

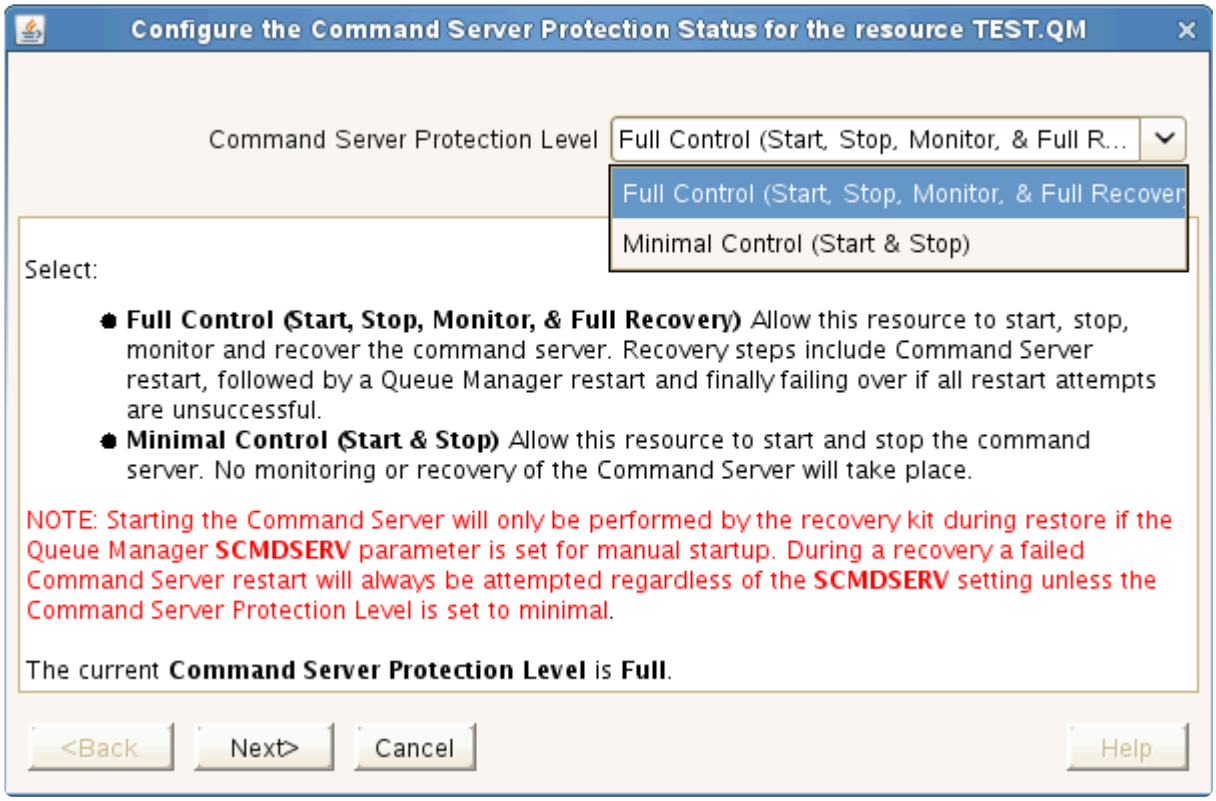
このコマンドを使用すると、キューマネージャ `TEST.QM` (-i) において、クラスタ (-c) 内の各ノード上のサーバ接続チャネル (-p) が *`LK.TEST.SVRCONN`* (-v) に設定 (-s) されます。

注記: キューマネージャの名前 (-i) または LifeKeeper TAG (-t) の名前のいずれかを使用できます。

# 6.8.5.7.6. コマンドサーバ保護設定の変更

## GUI

まず、WebSphere MQ の **リソースプロパティ画面** または上述の **リソースコンテクストメニュー** を開きます。 **[Command Server Protection Configuration]** アイコンまたはメニュー項目をクリックします。以下のダイアログが表示されます。




LifeKeeper が起動、停止、監視、およびリカバリが失敗した場合にフェイルオーバするようにするには、WebSphere MQ キューマネージャのコマンドサーバコンポーネントの **[Full Control]** を選択します。

LifeKeeper が起動と停止のみを行い、監視やリカバリは行わないようにするには、WebSphere MQ キューマネージャのコマンドサーバコンポーネントの **[Minimal Control]** を選択します。

詳細については、上述の表 を参照してください。保護制御を選択したら、 **[Next]** をクリックします。コマンドサーバ保護を現在の設定から新しい設定に変更するか確認するメッセージが表示されたら、 **[Continue]** をクリックして、クラスタ内の全てのノードを変更します。

## コマンドライン

**コマンドサーバ保護設定** をコマンドラインで設定するには、次のコマンドを使用します。

```
/opt/LifeKeeper/lkadm/subsys/appsuite/mqseries/bin/mq_modqmgrparam -c -s
-i TEST.QM -p CMDSERVERPROTECTION -v LEVEL
```

上記の LEVEL は **Full** または **Minimal** です。

このコマンドを使用すると、キューマネージャ TEST.QM (-i) において、クラスタ (-c) 内の各ノード上の LifeKeeper コマンドサーバ保護設定 (-p) が LEVEL (-v) に設定 (-s) されます。

**注記:** キューマネージャの名前 (-i) または LifeKeeper TAG (-t) の名前のいずれかを使用できます。

# 6.8.5.7.7. LifeKeeper WebSphere MQ Recovery Kit のデフォルト設定の変更

IBM WebSphere MQ Recovery Kit は、多数のデフォルト値を使用しています。デフォルト設定に問題がある場合、これらを調整および変更できます。デフォルト設定は、ほとんどの環境で問題無くできます。タイムアウト値に問題がある場合、次の表を使用して、調整可能なパラメータを特定できます。WebSphere MQ リソース階層に問題が発生するまでは、パラメータを変更しないことをお勧めします。

| /etc/default/LifeKeeper 内の変数名 | デフォルト値 | 説明 |
|---|---|---|
| MQS_QUICKCHECK_TIMEOUT_SC | 10 (秒) | サーバ接続チェックのタイムアウト。 |
| MQS_QUICKCHECK_TIMEOUT_CC | 10 (秒) | クライアント接続チェックのタイムアウト。 |
| MQS_QUICKCHECK_TIMEOUT_PUTGET | 10 (秒) | PUT/GET チェックのタイムアウト。 |
| MQS_QUICKCHECK_TIMEOUT_PS | 5 (秒) | publish/subscribe の使用有無チェックのタイムアウト |
| MQS_QUICKCHECK_TIMEOUT_CLUSTER | 5 (秒) | キューマネージャが WebSphere MQ クラスタの一部であるかのチェックのタイムアウト |
| MQS_QUICKCHECK_TIMEOUT | 40 (秒) | quickCheck スクリプトのタイムアウト (最低 10 秒以上) |
| MQS_QMGR_START_TIMEOUT | 60 (秒) | キューマネージャの起動コマンド完了のタイムアウト |
| MQS_CMDS_START_TIMEOUT | 30 (秒) | コマンドサーバの起動コマンド完了のタイムアウト |
| MQS_LISTENER_START_TIMEOUT | 30 (秒) | リスナーの起動コマンド完了のタイムアウト |
| MQS_LISTENER_LIST_TIMEOUT | 10 (秒) | リスナーの list コマンド完了のタイムアウト |
| MQS_CHECK_TIMEOUT_ACTION | ignore | サーバ接続チェックまたはクライアント接続チェックがタイムアウトになった場合のアクション。デフォルト値の ignore は、タイムアウトに関するメッセージはログに記録されるが、リカバリは開始されないことを表します。この変数を sendevent にすると、サーバ接続チェックがタイムアウトした際にローカルリカバリが開始されます。 |
| MQS_LISTENER_CHECK_DELAY | 2 (秒) | リスナーの起動から、リスナー正常起動 |

| | | |
|---|---|---|
| | | チェックを行うまでの時間 (秒)。デフォルトの 2 秒は、使用状態のポートを検出するのに十分です。 |
| NO_AUTO_STORAGE_DEPS | 0 | この変数を 1 に設定すると、リカバリキットは、キューマネージャとログディレクトリが共有ストレージにあるかどうかを確認しません。1 に設定した場合、リカバリキットは、リソース設定時のファイルシステム階層の作成も行いません。 |
| MQS_DSPMQVER_TIMEOUT | 5 (秒) | dspmqver コマンド (WebSphere MQ のバージョンを知るために必要) のタイムアウト。最低 2 秒必要。 |
| MQS_SKIP_CRT_MISSING_Q | 0 | 1 に設定すると、欠落したテストキューを自動作成しません。 |
| MQS_ALT_USER_NAME | 設定されていない、もしくはユーザーが”mqm”グループにメンバーシップを持っていない場合、mqm | すべてのWebSphere MQコマンドに別のユーザーネームを使用。デフォールトでは、ユーザー”mqm”が使用される。もし設定されている場合には、ユーザーはそのプライマリーグループをグループ”mqm”に設定するか、そのグループにセカンダリーメンバーシップを持たなくてはならない。 |

パラメータを変更するには、上の表内の適切な変数を */etc/default/LifeKeeper* に追加してください。行のシンタックスは、次のとおりです。

```
[…]
MQS_CHECK_TIMEOUT_ACTION=sendevent
[…]
```

カスタム設定を無効化してデフォルト値に戻すには、その行を /etc/default/LifeKeeper から削除するか、コメントアウトします。

# 6.8.6. WebSphere MQ のトラブルシューティング

## WebSphere MQ ログの場所

キューマネージャの名前が既知であり、キューマネージャが使用可能な場合、WebSphere MQ エラーログは、キューマネージャ設定ファイル (*qm.ini*) で定義される `LogPath` パラメータで指定されたディレクトリにあります。キューマネージャが使用できない場合、エラーログは、(*/var/mqm/qmgrs/@SYSTEM/errors*) にあります。クライアントアプリケーションでエラーが発生した場合、エラーログはクライアントのルートドライブ (*/var/mqm/errors*) にあります。

Message Queue Interface (MQI) 呼び出しが失敗したことを示すリターンコードがアプリケーションに返された場合、リターンコードの詳細について *WebSphere MQ Application Programming Reference Manual* を参照してください。

# 6.8.6.1. エラーメッセージ

このセクションでは、SPS MQ Recovery Kit の使用中に表示される可能性のあるメッセージの一覧を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

MQ Recovery Kit は、階層の作成および拡張時に他の SPS コンポーネントを利用するため、それらの他のコンポーネントによるメッセージが表示されることもあります。その場合、メッセージカタログ を参照してください。メッセージカタログでは、操作、管理、GUI など、SIOS Protection Suite for Linux を使用しているときに出会う可能性がある、すべてのエラーコードを列挙します。また、エラーコードの原因に関する補足説明や、問題解決のために必要な処置についても、必要に応じて記載します。この完全なリストを検索すると、受信したエラーコードを見つけることができます。また、適切な SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

## 共通のエラーメッセージ

| エラー番号 | エラーメッセージ | 処置 |
|---|---|---|
| 119001 | Queue manager with TAG "TAG" failed to start on server "SERVER" with return code "Code" | start コマンドは正常に実行されましたが、開始後のチェックが失敗しました。<br><br>SERVER 上の IBM WebSphere MQ アラートログを確認し、エラーを修正してください。 |
| 119002 | Queue manager with TAG "TAG" start command failed on server "SERVER" with return code "Code". | キューマネージャ TAG に対する start コマンドが 0 以外の値を返しました。<br><br>SERVER 上の IBM WebSphere MQ アラートログを確認し、エラーを修正してください。<br><br>リターンコード Code は、strmqm コマンドのリターンコードです。 |
| 119006 | Command server start command for queue manager "TAG" failed on server "SERVER" with return code "Code". | コマンドサーバに対する起動コマンドが、0 以外の値を返しました。<br><br>SERVER 上の IBM WebSphere MQ アラートログを確認し、エラーを修正してください。<br><br>リターンコード Code は、runmqsc コマンドのリターンコードです。<br><br>WebSphere MQ v6.0 の場合、コマンドサー |

| | | |
|---|---|---|
| | | バの起動タイプが「MANUAL」になっていることを確認してください。詳細については、設定の要件 を参照してください。 |
| 119007 | Listener for queue manager "TAG" failed to start on server "SERVER". | SERVER 上の IBM WebSphere MQ アラートログを確認し、エラーを修正してください。 |
| 119008 | Listener start command for queue manager with TAG "TAG" failed on server "SERVER" with return code "CODE". | SERVER 上の IBM WebSphere MQ アラートログを確認し、エラーを修正してください。 |
| 119013 | Could not create queue manager object for queue manager “QMGR” with TAG “TAG”. | LifeKeeper と WebSphere MQ のエラーログを確認してください。 |
| 119014 | Could not create listener object for queue manager “QMGR” with TAG “TAG”. | LifeKeeper と WebSphere MQ のエラーログを確認してください。 |
| 119015 | No value forthe “PARAMETER” specified. | LifeKeeper MQ Recovery Kit スクリプトを、正しい引数を使用して実行してください。 |
| 119016 | Instance with ID “ID” does not exist on server “SERVER”. | リソース階層を確認してください。 |
| 119017 | Instance with TAG “TAG” does not exist on server “SERVER”. | リソース階層を確認してください。 |
| 119018 | Invalid parameters specified. | 正しいオプションを使用してスクリプトを実行してください。 |
| 119019 | Too few parameters specified | 正しいオプションを使用してスクリプトを実行してください。 |
| 119021 | Failed to set “VALUE” for resource instance “TAG” on server “SERVER”. | LifeKeeper ログで、値の設定に関するエラーを確認してください。 |
| 119025 | Failed to update instance info for queue manager with TAG “TAG” on server “SERVER”. | サーバが動作中の状態に戻ったら、操作を再度実行して設定を同期してください。 |
| 119026 | The following program required does not exist or is not executable:"EXECUTABLE".Check failed. | プログラム EXECUTABLE が見つかりませんでした。すべてのインストール要件が満たされていることを確認し、必要なパッケージをすべてインストールしてください。<br><br>詳細については、設定の要件 を参照してください。 |
| 119032 | Script: usage error (error message) | 正しい引数を使用して、スクリプト Script を開始してください。 |
| 119033 | Script: error parsing config file "ConfigFile". | ConfigFile が存在し、読み取り可能であることを確認してください。 |

| | | |
|---|---|---|
| 119034 | CHECKTYPE check for queue manager with TAG "TAG" failed on server "SERVER" because the MQUSER could not be determined.This is probably because of a removed configuration file - ignoring. | タグ TAG のキューマネージャに対する CHECKTYPE チェックが失敗しました。<br><br>グローバル設定ファイル (mqs.ini) が存在し、読み取り可能であることを確認してください。<br><br>削除されている場合、mqs.ini 設定ファイルを再作成してください。 |
| 119035 | CHECKTYPE check for queue manager with TAG "TAG" failed on server "SERVER" because no TCP PORT directive found in config file "CONFIGFILE" - ignoring. | キューマネージャ設定ファイル (qm.ini) が存在し、インストール中に必要な TCP セクションが含まれていることを確認してください。<br><br>TCP セクションをキューマネージャ設定ファイルに追加してください。 |
| 119042 | “CHECKTYPE” check for queue manager with TAG “TAG” failed on server “SERVER” because no TCP PORT information was found via runmqsc. | リスナーオブジェクトのポート情報が定義済みであり、runmqsc でアクセス可能であることを確認してください。 |
| 119043 | TCP Listener configuration could not be read, reason:“REASON”. | MQ が実行中であり、リスナーオブジェクトのポート情報が定義済みで、runmqsc でアクセス可能であることを確認してください。 |
| 119044 | No TCP Listener configured, no TCP PORT information was found via runmqsc:“MESSAGE”. | リスナーオブジェクトのポート情報が定義済みであり、runmqsc でアクセス可能であることを確認してください。 |

## 作成

| エラー番号 | エラーメッセージ | 処置 |
|---|---|---|
| 001022 | END failed hierarchy “CREATE” of resource “TAG” on server “SERVER” with return value of “VALUE”. | サーバ SERVER 上の LifeKeeper ログに、リソース階層の作成に関するエラーがないか確認します。このエラーは、キューマネージャが起動していないことに関係する場合があります。 |
| 119020 | Create MQSeries queue manager resource with TAG “TAG” for queue manager “QMGR” failed. | LifeKeeper ログにリソースの作成に関するエラーがないか確認します。このエラーは、キューマネージャが起動していないことに関係する場合があります。 |
| 119022 | Failed to create dependency between “PARENT” and “CHILD”. | LifeKeeper ログに依存関係の作成に関するエラーがないか確認します。 |
| 119023 | Creating the filesystem hierarchies for | LifeKeeper ログにファイルシステム階層の作成に関 |

| | | |
|---|---|---|
| | queue manager with TAG “TAG” failed.File systems:“Filesystems”. | するエラーがないかを確認します。 |
| 119029 | No TCP section configured in "CONFIGFILE" on server "SERVER". | サーバ SERVER 上のキューマネージャ設定ファイルに TCP セクションを追加します。詳細については、「設定の要件 」を参照してください。 |
| 119031 | Queue manager "DIRTYPE" directory ("DIRECTORY") not on shared storage. | ディレクトリ DIRECTORY を共有ストレージに移動して、操作を再度実行してください。 |
| 119038 | Creation of queue manager resource with TAG "TAG" failed on server "SERVER". | サーバ SERVER 上の LifeKeeper ログのエラーを確認および修正し、操作を再度実行してください。 |
| 119039 | TCP section in configuration file "FILE" on line "LINE1" is located before LOG section on line "LINE2" on server "SERVER". | キューマネージャ設定ファイル内で、TCP セクションは LOG: セクションより後に配置することを推奨します。<br><br>TCP セクションをキューマネージャ設定ファイルの末尾に移動し、操作を再度実行してください。 |
| 119040 | Creation of MQSeries queue manager resource by create_ins was successful but no resource with TAG “TAG” exists on server “SERVER”.Sanity check failed. | LifeKeeper ログにリソース作成中のエラーがないか確認してください。 |
| 119041 | Creation of MQSeries queue manager resource was successful but no resource with TAG “TAG” exists on server “SERVER”.Final sanity check failed. | LifeKeeper ログにリソース作成中のエラーがないか確認してください。 |

## 拡張

| エラー番号 | エラーメッセージ | 処置 |
|---|---|---|
| 119024 | Instance "TAG" can not be extended from "TEMPLATESYS" to "TARGETSYS".Reason:REASON | REASON に記載されているエラーを修正し、操作を再度実行してください。 |
| 119027 | The user "USER" does not exist on server "SERVER". | プライマリサーバ上と同じ UID を持つユーザ USER を SERVER 上に作成し、操作を再度実行してください。 |
| 119028 | The user "USER" has a different numeric UID on server "SERVER1" (SERVER1UID) then it should be (SERVER2UID). | USER が全サーバ上で同一の UID を持つように UID を変更し、UID を変更したサーバ上の WebSphere MQ を再インストールしてから、操作を再度実行してください。 |
| 119029 | No TCP section configured in "CONFIGFILE" on server "SERVER". | サーバ SERVER 上のキューマネージャ設定ファイルに TCP セクションを追加します。 |

| | | |
|---|---|---|
| | | 詳細については、「設定の要件 」を参照してください。 |
| 119030 | Queue manager "QMGR" not configured in "CONFIGFILE" on server "SERVER". | 拡張対象のキューマネージャ QMGR は、ターゲットサーバ SERVER 上のグローバル設定ファイルに設定されていません。キューマネージャのスタンザをサーバ SERVER 上の設定ファイル CONFIGFILE に追加し、操作を再度実行してください。 |
| 119036 | Link "LINK" points to "LINKTARGET" but should point to "REALTARGET" on server "SERVER". | ファイルシステムレイアウト上、3 つのシンボリックリンクがテンプレートおよびターゲットサーバ SERVER 上の同じ場所を指している必要があります。<br><br>サーバ SERVER 上のリンク LINK を REALTARGET を指すように修正し、操作を再度実行してください。 |
| 119037 | Link "LINK" that should point to "REALTARGET" does not exist on system "SERVER". | ファイルシステムレイアウト上、3 つのシンボリックリンクは、ターゲットサーバにも存在する必要があります。<br><br>サーバ SERVER 上に REALTARGET への必要なリンク LINK を作成し、操作を再度実行してください。 |

## 削除

| エラー番号 | エラーメッセージ | 処置 |
|---|---|---|
| 119003 | Failed to stop queue manager with TAG “TAG” on server “SERVER". | サーバ SERVER 上のキューマネージャ TAG が Recovery Kit によって停止できませんでした。詳細な情報を確認し調査するには、ログレベルを DEBUG に変更してください。マシン負荷によっては、シャットダウンタイムアウト値を大きくする必要がある場合があります。 |
| 119004 | Some orphans of queue manager with TAG “TAG” could not be stopped on server "SERVER".Tried it “tries” times. | オーファンを手動で停止し、Queue Manager を再起動してください。より詳細な情報を得るには、ログレベルを DEBUG に変更します。 |
| 119010 | Listener for queue manager with TAG "TAG" failed to stop on server "SERVER". | このメッセージは、リスナーの監視が有効な場合のみ表示されます。より詳細な情報を得るには、ログレベルを DEBUG に変更します。 |

## リソース監視

| エラー番号 | エラーメッセージ | 処置 |
|---|---|---|
| 119005 | Queue manager with TAG “TAG” on server | SERVER 上の IBM WebSphere MQ アラートログでエラーを確認します。このメッセージは、キューマネージャがクラッシュしたことを示 |

| | | |
|---|---|---|
| | "SERVER" failed. | しています。 |
| 119009 | Listener for queue manager with TAG “TAG” failed on server "SERVER". | このメッセージは、リスナーの監視が有効な場合のみ表示されます。より詳細な情報を得るには、ログレベルを FINE に変更します。 |
| 119011 | “CHECKTYPE" PUT/GET Test for queue manager with TAG "TAG" failed on server "SERVER" with return code "Code" | このメッセージは、PUT/GET テストが有効で、テストキューが存在する場合のみ表示されます。より詳細な情報を得るには、ログレベルを FINE に変更し、SERVER 上の IBM WebSphere キューマネージャエラーログ (/var/mqm/errors) でエラーを確認し、修正します。ファイルシステムの容量が不足していないか確認します。 |
| 119012 | Client connect test for queue manager with TAG "TAG" on server "SERVER" failed with return code "Code". | このメッセージは、リスナーの管理が有効な場合のみ表示されます。<br><br>このメッセージは、リスナーまたはキューマネージャの問題を示しています。<br><br>ログでエラーを確認し、修正します。<br><br>リターンコード Code は、amqscnxc コマンドのリターンコードです。 |

## 警告メッセージ

| エラー番号 | エラーメッセージ | 処置 |
|---|---|---|
| 119201 | Listener for queue manager with TAG "TAG" is NOT monitored on server "SERVER". | これは、リスナーの管理が有効になっていないことを示す警告です。 |
| 119202 | Queue manager with TAG “TAG” is not running on server “SERVER” but some orphans are still active.This is attempt number “ATTEMPT” at stopping all orphans processes. | これは、MQ が正しく停止されなかったことを示す警告です。 |
| 119203 | Another instance of recover is running, exiting “EXITCODE”. | リカバリプロセスを開始したが、別のリカバリプロセスがすでに実行中のため、このリカバリプロセスを継続できません。 |
| 119204 | Queue manager server connect check for queue manager with TAG "TAG%" timed out after "SECONDS" seconds on server "SERVER". | このメッセージがよく表示される場合、/etc/defaul/LifeKeeper の MQS_QUICKCHECK_TIMEOUT_SC の値を大きくしてください。<br><br>詳細については、サーバ接続チャネルの変更を参照してください。 |
| 119205 | Queue manager client connect check for queue manager with TAG "TAG" timed out after | このメッセージがよく表示される場合 /etc/defaul/LifeKeeper の |

| | | |
|---|---|---|
| | "SECONDS" seconds on server "SERVER". | MQS_QUICKCHECK_TIMEOUT_CC の値を大きくしてください。<br><br>詳細については、サーバ接続チャネルの変更を参照してください。 |
| 119206 | Server "SERVER" is not available, skipping. | キューマネージャ設定の更新中に、サーバがオンラインではありませんでした。<br><br>サーバがオンラインに戻るのを待ってから、設定手順を再度実行してください。 |
| 119207 | "CHECKTYPE" PUT/GET test for queue manager with TAG "TAG" failed because test queue "QUEUE" does not exist (reason code "REASONCODE") - ignoring. | テストキュー QUEUE の設定を作成するか、既存のキューにテストキューを再設定します。<br><br>テストキューの作成については、「設定の要件」を参照してください。 |
| 119208 | Channel "CHANNEL" does not exist for queue manager with TAG "TAG" (reason code "REASONCODE") - ignoring. | 存在していないチャネル CHANNEL を作成してください。デフォルトでは、チャネル SYSTEM.DEF.SVRCONN が使用されます。<br><br>チャネルの作成方法については、WebSphere MQ ドキュメンテーションを参照してください。 |
| 119209 | PUT/GET test for queue manager with TAG "TAG" skipped because no test queue is defined. | キューマネージャ TAG に LifeKeeper テストキューを設定してください。 |
| 119210 | The following program required to perform the PUT/GET test does not exist or is not executable:"EXECUTBALE".Test skipped. | C コンパイラをシステムにインストールし、それが root ユーザの PATH 環境変数に配置されていることを確認してください。スクリプト LKROOT/lkadm/subsys/appsuite/mqseries/bin/compilesamples を実行し、修正済みサンプルプログラムの amqsget および amqsgetc をコンパイルします。 |
| 119211 | Queue manager "CHECKTYPE" PUT/GET test for queue manager with TAG "TAG" timed out after "SECONDS" seconds on server "SERVER". | このメッセージがよく表示される場合、/etc/default/LifeKeeper の MQS_QUICKCHECK_TIMEOUT_PUTGET の値を大きくしてください。<br><br>詳細については、サーバ接続チャネルの変更を参照してください。 |
| 119212 | QuickCheck for queue manager with TAG “TAG” timed out after SECONDS seconds on server “SERVER”. | このメッセージがよく表示される場合、/etc/default/LifeKeeper の MQS_QUICKCHECK_TIMEOUT の値を大きくしてください。 |

| | | |
|---|---|---|
| 119213 | mqseriesQueueManager::getMQVersion::ERROR unexpected dspmqver output (OUTPUT) – reading cached value instead (Queue QUEUE, Queuemanager QMGR). | runmqsc による MQ バージョンの読み取りに失敗しました。このメッセージがよく表示される場合、/etc/default/LifeKeeper の MQS_DSPMQVER_TIMEOUT の値を大きくしてください。 |
| 119214 | mqseriesQueueManager::getMQVersion:: ERROR unexpected output retrieving MQ version information (Queue QUEUE, Queuemanager QMGR). Unexpected results * | mqm ユーザとして次のコマンドを実行して、出力が生成されるかを確認してください: dspmqver -b -p1 -f2. また、mqm ユーザとして次のコマンドを実行して、出力を確認してください: dspmqinst |

# 6.8.7. 付録 A – mqs.ini 設定ファイルの例

```
#********************************************************************#
#*                                                                  *#
#* <START_COPYRIGHT>                                                *#
#* Licensed Materials - Property of IBM                             *#
#* 63H9336                                                          *#
#* (C) Copyright IBM Corporation 1994, 2000                         *#
#*                                                                  *#
#* <END_COPYRIGHT>                                                  *#
#********************************************************************#
#* Module Name: mqs.ini                                               *#
#* Type       : WebSphere MQ Machine-wide Configuration File          *#
#* Function   : Define WebSphere MQ resources for an entire machine   *#
#********************************************************************#
#* Notes      :                                                       *#
#* 1) This is the installation time default configuration             *#
#*                                                                    *#
#********************************************************************#
AllQueueManagers:
   #********************************************************************#
   #* The path to the qmgrs directory, below which queue manager data  *#
   #* is stored                                                        *#
   #********************************************************************#
   DefaultPrefix=/var/mqm

LogDefaults:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=1024
   LogType=CIRCULAR
   LogBufferPages=0
   LogDefaultPath=/var/mqm/log
QueueManager:
   Name=TEST.QM
   Prefix=/var/mqm
   Directory=TEST!QM
DefaultQueueManager:
   Name=TEST.QM
QueueManager:
   Name=TEST.QM.NEW
   Prefix=/var/mqm
   Directory=TEST!QM!NEW
QueueManager:
   Name=TEST.QM2
   Prefix=/var/mqm
```

```
   Directory=TEST!QM2
QueueManager:
   Name=MULTIINS_1
   Prefix=/var/mqm
   Directory=MULTIINS_1
   DataPath=/opt/webmq/MULTIINS_1/data

   InstallationName=Installation1


QueueManager:
   Name=MULTIINS_2
   Prefix=/var/mqm
   Directory=MULTIINS_2
   DataPath=/opt/webmq/MULTIINS_2/data

   InstallationName=Installation2
```

# 6.8.8. 付録 B – qm.ini 設定ファイルの例

```
#********************************************************************#
#* Module Name: qm.ini                                              *#
#* Type       : WebSphere MQ queue manager configuration file       *#
#  Function   : Define the configuration of a single queue manager *#
#*                                                                  *#
#********************************************************************#
#* Notes      :                                                     *#
#* 1) This file defines the configuration of the queue manager      *#
#*
*#
#********************************************************************#
ExitPath:
   ExitsDefaultPath=/var/mqm/exits/
   ExitsDefaultPath64=/var/mqm/exits64
#*                                                                  *#
#*                                                                  *#
Log:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=1024
   LogType=CIRCULAR
   LogBufferPages=0
   LogPath=/opt/MQ_log/MULTIINS_1
   LogWriteIntegrity=TripleWrite
Service:
   Name=AuthorizationService
   EntryPoints=14
ServiceComponent:
   Service=AuthorizationService
   Name=MQSeries.UNIX.auth.service
   Module=amqzfu
   ComponentDataSize=0
```

# 6.8.9. 付録 C – WebSphere MQ 設定シート

| | | |
|---|---|---|
| クラスタ名 | | |
| 連絡先<br><br>(クラスタ責任者のメールアドレスまたは電話番号) | | |
| LifeKeeper のバージョン | | |
| オペレーティングシステム | | |
| クラスタノード | 名前 | パブリック IP / ネットマスク |
| | | |
| | | |
| | | |
| | | |

| | | |
|---|---|---|
| キューマネージャーの名前 | | |
| LifeKeeper によるリスナー管理 | [ ] YES [ ] NO | |
| WebSphere MQ オペレーティングシステムユーザ | 名前 | 数値 (UID/GID) |
| ユーザ<br><br>(例: mqm/1002) | | |
| グループ<br><br>(例: mqm/200) | | |
| 仮想 IP / ネットマスク / ネットワークドライブ<br><br>(例: 192.168.1.1/24/eth0) | | |
| ファイルシステムレイアウト | __ 設定 1 – 共有ストレージ上の/var/mqm | |
| | __ 設定 2 – 直接マウント | |
| | __ 設定 3 – シンボリックリンク<br><br>__ 設定 4 – マルチインスタンスのキューマネージャ | |
| | __ その他 | |
| 共有ストレージのタイプ | __ NAS<br>(IP:_____) | |

| | |
|---|---|
| | __ SCSI/FC (タイプ:_____)<br>__ SDR |
| キューマネージャ */var/mqm/qmgrs/QM.NAME* の物理的場所<br>(デバイス、マウントポイント、または論理ボリューム)<br><br>(例: `LVM /dev/mqm_test_qm/qmgrs` ) | |
| キューマネージャ */var/mqm/log/QM.NAME* の物理的場所<br><br>(デバイス、マウントポイント、または論理ボリューム)<br>(例: `LVM /dev/mqm_test_qm/log` ) | |

# 6.9. NAS Recovery Kit 管理ガイド

## NAS Recovery Kit テクニカルドキュメンテーション

LifeKeeper for Linux Network Attached Storage Recovery Kit（以下、NAS Recovery Kit と呼ぶ）は、LifeKeeper 環境においてネットワーク・ファイル・システム（NFS）ソフトウェアの障害回復能力を高めるものです。NAS Recovery Kit は、LifeKeeper ユーザに対し、 エクスポートされた NFS ファイル・システムを LifeKeeper 階層の基本ストレージとして使用する機会を提供します。

## ドキュメントの内容

本書は以下のトピックについて説明しています。

- ドキュメンテーションとリファレンス 。LifeKeeper for Linux のドキュメントと所在の一覧です。
- 要件 。NAS Recovery Kit を適切に設定、導入、運用するうえで必要なハードウェアとソフトウェアについて説明しています。LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SIOS Protection Suite インストールガイド を参照してください。
- 概要 。NAS Recovery Kit の特徴と機能について説明しています。
- LifeKeeper for Linux NAS Recovery Kit の設定 。NAS Recovery Kit を適切に設定するうえで必要な手順について説明しています。
- LifeKeeper 設定作業 。LifeKeeper GUI を使用して NAS リソース階層を作成し管理する作業について説明しています。
- トラブルシューティング 。LifeKeeper for Linux のエラー・メッセージの一覧とその説明です。

# 6.9.1. ドキュメンテーションとリファレンス

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- SIOS Protection Suite インストールガイド
- Optional Recovery Kit ドキュメンテーション

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

# 6.9.2. ハードウェア要件とソフトウェア要件

SPS for Linux NAS Recovery Kit をインストールする前に、SPS の構成が次の要件を満たしている必要があります。SPS for Linux ハードウェアおよびソフトウェアの設定に関する具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

## ハードウェア要件

- サーバ 。製品媒体に付属する SPS for Linux リリースノート および SPS for Linux インストールガイド (製品メディアに収録) に記載の要件に従って設定されたLifeKeeper for Linux 対応サーバ。

- **IP** ネットワーク・インターフェース・カード 。各サーバは、イーサネット TCP/IP をサポートするネットワーク・インタフェースを 1 つ以上必要とします。ただし、LifeKeeper クラスタには 2 つの通信パスが必要で、独立した 2 つのサブネットを使用する 2 つの分離した LAN ベース通信パスが推奨され、これらの 1 つ以上をプライベート・ネットワーク として構成する必要があることを忘れないでください。TCP と TTY のハートビートの組み合わせもサポートされています。

## ソフトウェア要件

- **TCP/IP** ソフトウェア 。LifeKeeper 環境の各サーバには TCP/IP ソフトウェアが必須です。

- **LifeKeeper** ソフトウェア 。クラスタの各サーバにインストールした LifeKeeper for Linux ソフトウェアのバージョン、および適用した LifeKeeper for Linux ソフトウェア・パッチのバージョンが同一であることが絶対条件です。

- **LifeKeeper for Linux NAS Recovery Kit** - NAS Recovery Kit は CD で提供されます。 Red Hat Package Manager（rpm）でパッケージされており、rpm を使用してインストール／アンインストールを行います。LifeKeeper for Linux NAS Recovery Kit CD には、次の rpm ファイルが格納されています。

  **steeleye-lkNAS**

- **Linux** ソフトウェア - クラスタ内の各サーバに、 **util-linux** パッケージがインストールされている必要があり、LifeKeeper および LifeKeeper NAS Recovery Kit の構成前に構成しておく必要があります。NAS Recovery Kit が正常に機能するには、バージョン 2.9u 以降の **util-linux** パッケージが必要です。

  LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SPS for Linux インストールガイド を参照してください。

# 6.9.3. 概要

## LifeKeeper for Linux NAS Recovery Kit

LifeKeeper for Linux NAS Recovery Kit の第 1 の目的は、LifeKeeper ユーザに対し、共有ストレージとデータ・レプリケーションに代わる代替ストレージ方式を提供することです。NAS Recovery Kit を使用すると、LifeKeeper 保護下のサーバまたはクライアント上に LifeKeeper リソース階層を作成できます。このリソース階層には、Network Attached Storage デバイスまたはクラスタ内の NFS サーバからエクスポートしたネットワーク・ファイル・システム（NFS）をインポート（マウント）できます。エクスポートしたファイル・システムがマウントされたクラスタ内のノードで障害が検出されると、NAS Recovery Kit は、事前判別されたバックアップ・ノードへのフェイルオーバを開始します。

そのため、エクスポートしたファイル・システムが LifeKeeper サーバまたはクライアン トにマウントされたら、それを LifeKeeper 階層のストレージの基礎として完全に使用できます。

エクスポートしたファイル・システムをストレージ・メディアとして使用する場合、 LifeKeeper では、ファイル・システムがエクスポートされたサーバの保護を必要としません。ただし、可用性を高めるには、ユーザが LifeKeeper for Linux NFS Server Recovery Kit を使用して、ファイル・システムがエクスポートされたサーバで障害が発生しないように保護することが推奨されます。

NAS Recovery Kit のリソース階層は、LifeKeeper Core 製品（steeleye-lk パッケージ）から 入手可能な現存の File System Recovery Kit を使用して作成されます。

NAS Recovery Kit にはいくつかの利点がありますが、最も重要な 2 つの利点は、コストのかかる共有ストレージ・デバイスを使用する必要がなくなる点と、複数ノードのクラスタ構成が可能になる点です。

## NAS Recovery Kit　の制限事項

- このバージョンの NAS Recovery Kit では、NAS デバイスへのアクセスが失敗したときのローカル・リカバリーをサポートしていません。障害が検出されると、デフォルト・アクションによって、バックアップ・サーバへの階層の転送が開始されます。 リソース階層の構成に応じて、このアクションはハング・プロセスにつながることがあります。ハング・プロセスを回避するには、サーバを停止してバックアップ・ サーバへのフェイルオーバを強制するように、デフォルト・アクションを変更します。デフォルトの切り替え動作を変更するには、LifeKeeper デフォルト・ファイルで LKNASERROR の設定を変更します。LKNASERROR の詳細については、このドキュメントのセクション「LifeKeeper for NAS Recovery Kit の設定」を参照してください。

- NAS Recovery Kit は、Network Attached Storage デバイスを保護しません。このキットの目的は、LifeKeeper ストレージ・オプションを Network Attached Storage 領域まで拡張することです。

- NAS Recovery Kitでは、NFSファイル・システムを異なるマウント・ポイントへ複数回マウントすることは許可されません。ファイル・システムが/etc/fstabファイルで複数回検出されると、階層の作成試行は失敗します。

- NAS Recovery Kit で保護するファイル・システムは、ホスト名ではなく IP アドレス（たとえば、server1/dir ではなく 100.99.100.9/dir）を使用してマウントする必要があります。これにより、

DNS またはホスト・ファイル・ルックアップの潜在的な問題を回避できます。ホスト名を使用してマウントすると、「不良マウント」が検出されます。このような場合は、LifeKeeper がファイル・システムをアンマウントし、IP アドレスを使用して再マウントします。アンマウント・プロセスによって、マウント・ポイントを現在使用しているプロセスが消滅することがあります。

# 6.9.4. LifeKeeper for Linux NAS Recovery Kit の設定

本セクションでは、LifeKeeper for Linux NAS Recovery Kit の設定の詳細について説明します。NAS Recovery Kit の設定と管理を開始する前に考慮すべき事項についても記載しています。LifeKeeper Core リソース階層の設定方法については、SPS for Linux テクニカルドキュメンテーション を参照してください。

# 6.9.4.1. 設定上の考慮事項

LifeKeeper for Linux NAS Recovery Kit を操作する前に、次の事項を検討する必要があります。

1. エクスポートしたファイル・システムをマウントし、かつ NAS リソース階層を拡張するクラスタ構成内のサーバへ、NAS Recovery Kit をインストールします。ファイル・システムは、LifeKeeper で保護できる NFS サーバからエクスポートすることもできますし（推奨設定）、Network Attached Storage デバイスからエクスポートするこ ともできます。

2. このキットを適切に実行するために、サーバ名ではなくサーバの IP アドレスを使用して、エクスポートされた NFS ファイル・システムをマウントすること、およびファイル・システムを LifeKeeper 保護下に入れる前にマウント操作を実行することを強くお勧めします。さらに、LifeKeeper for Linux NFS Server Recovery Kit によって現在保護されているファイル・システムをマウントする場合は、NFS Server 階層を作成するために使用した IP アドレスを、LifeKeeper NAS サーバへファイル・システム をマウントするために使用することを強くお勧めします。

3. スプリットブレイン（つまり、クラスタ内の複数のノードにサービス中の保護された（ISP）階層がある）に関連する問題が発生しないようにするため、エクスポートしたファイル・システムへアクセスするために使用する同じネットワーク上のクラスタ内のノード間に、通信パスの 1 つを確立することを強くお勧めします。この推奨事項に従わないと、通信パスの障害が発生したときに、複数のノードで階層が ISP スプリットブレインになる可能性があります。スプリットブレインのシナリオからリカバリーするには、1 つの ISP 階層を除いたすべての階層をサービス休止状態にします。これにより、1 つのノードのみが、エクスポートされたファイル・システムへアクセスできるようになります。

4. NAS 階層をビルドするために使用するビルトイン・ファイル・システムのリカバリー・キットでは、フェイルオーバ条件でマウントされたファイル・システムを使用 している LifeKeeper 保護下に入っていないプロセスを検出したり、除去したりすることはできません。そのため、LifeKeeper 保護下に入れるプロセスのみが、NAS 保護下のファイル・システムを使用するように設定することを強くお勧めします。

5. LKNFSTIMEOUT という調整可能パラメータは、NFS マウント・ファイル・システムのステータスを判別するときに NAS Recovery Kit が使用するタイムアウトを秒単位で表します。この調整可能パラメータのデフォルト値は、2 分に設定されます。 LKNFSSYSCALLTO という調整可能パラメータは、マウント・ポイントのステータスを判別するときにアラームがシステム・コールを中断するように NAS Recovery Kit が使用するタイムアウトを秒単位で表します。下記の式を使用して、この調整可能パラメータの値を判別します。LKNFSSYSCALLTO の 3 倍の値に 5 を加算した値が、LKNFSTIMEOUT の値より小さくなければなりません。

6. 調整可能パラメータ LKNASERROR は、NAS デバイスへのアクセスが失敗したときに NAS Recovery Kit が実行するアクションを制御します。調整可能パラメータには、2 つの 値（ **switch** および **halt** ）があり、 **switch** がデフォルトです。値が **switch** に設定され、アクセスが失敗すると、NAS Recovery Kit は障害の検出時に、バックアップ・サーバへのリソース階層の転送を開始します。バックアップ・サーバへのリソース階層の転送試行は、NAS リソースより上にあるリソースが NAS ファイル・システム上の項目へアクセスしようとすると、停止する場合があります。この問題を回避するには、調整可能値を **halt** に設定します。これにより、アクセス障害が検出されると、即時にシステムが停止します。このアクションによって、バックアップ・サーバへのすべてのリソース

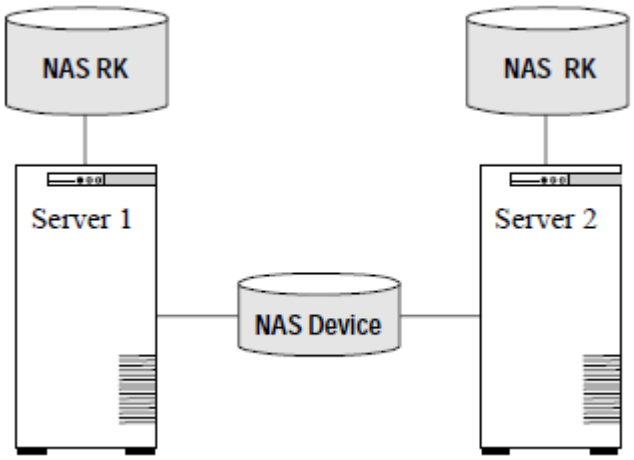
階層のフェイルオーバが強制されます。

7. ノード障害（すべてのコミュニケーションパスがダウンする）によるスプリットブレインが発生しないように、STONITHデバイスやQuorum/Witnessパッケージの使用を推奨します。スプリットブレインが発生して、クラスタ内の複数のノードでNASリソースがin serviceになるとデータ破損を引き起こす可能性があります。詳細は、SIOS Protection Suite のテクニカルドキュメンテーションを参照してください。

# 6.9.4.2. 構成例

フェイルオーバが発生したときに、 **LifeKeeper for Linux NAS Recovery Kit** を使用している とどのような処理が行われるか、その例をいくつか示します。

## 構成 1：アクティブ／スタンバイ設定の例



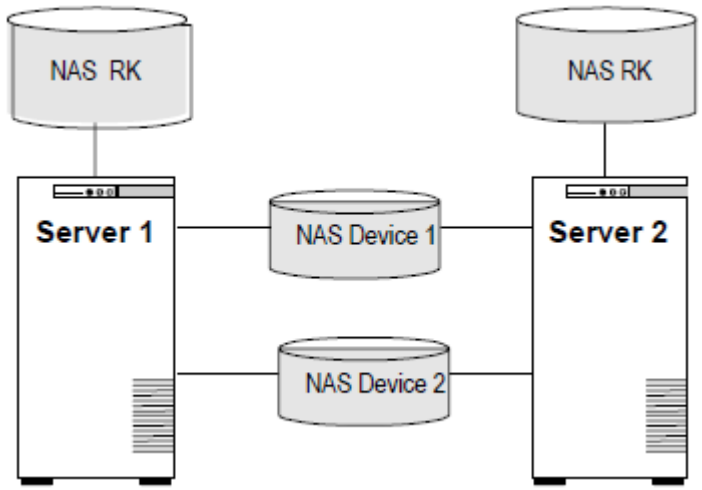

この構成では、サーバ 1 が NAS Recovery Kit ソフトウェアを実行しており、NAS デバイスからファイル･システムをインポート（マウント）したため、アクティブとみなされます。 サーバ 2 は別の処理を実行します。 サーバ 1 に障害が発生した場合、サーバ 2 が ファイル･システムへのアクセス権を獲得し、LifeKeeper のセカンダリ階層を使用して、 クライアントがファイル･システムへアクセスできるようにします。

**設定の注意点：**

- NAS ソフトウェアを両方のサーバにインストールしておく必要があります。
- ファイル･システムが NAS デバイスからインポートされました。
- サーバ 1 がアクティブである間、サーバ 2 は NAS デバイス上のファイルおよびディレクトリにアクセスできません。

**注記：** アクティブ／スタンバイ設定では、サーバ 2 でも NAS Recovery Kit が実行されていることがありますが、他の NAS リソースは LifeKeeper によって保護されていません。

# 構成 2：アクティブ／アクティブ設定の例




アクティブ／アクティブ設定は、NAS Recovery Kit ソフトウェアをアクティブに実行し、 かつファイル・システムを NAS デバイスからインポートする複数のシステムで構成されます。

**設定の注意点：**

- NAS ソフトウェアを両方のサーバにインストールしておく必要があります。
- 最初に、サーバ 1 がファイル・システムをインポートし、サーバ 2 が別のファイ ル・システムをインポートします。切り替え時には、1 つのシステムで両方のファイル・システムをインポートできます。

# 6.9.5. LifeKeeper の設定作業

LifeKeeper for Linux NAS Recovery Kit の管理作業はすべて、LifeKeeper グラフィカル・ユーザ・インタフェース（GUI）で実行できます。LifeKeeper GUI では、NAS リソースの設定、管理、監視作業をガイド付きで行えます。

LifeKeeper for Linux NAS Recovery Kit の設定に関しては、次の作業が可能です。

- リソース階層の作成 。NAS リソース階層を作成します。
- リソース階層の削除 。NAS リソース階層を削除します。
- リソース階層の拡張 。NAS リソース階層をプライマリ・サーバからバックアップ・サーバに拡張します。
- リソース階層の拡張解除 。LifeKeeper クラスタの 1 つのサーバの NAS リソース階層を拡張解除（削除）します。
- 従属関係の作成 。既存のリソース階層と別のリソース・インスタンスとの間に子従属関係を作成し、クラスタ内のすべての対象サーバに従属関係の変化を通知します。
- 従属関係の削除 。リソースの従属関係を削除して、クラスタ内のすべての対象サーバ に従属関係の変化を通知します。
- In Service 。リソース階層をアクティブにします。
- Out of Service 。リソース階層を非アクティブにします。
- 表示 / 編集 。 リソース階層のプロパティを表示または編集します。

**注記：** このセクションの残りの部分では、設定作業はすべて **[Edit]** メニューから実行することになっています。この設定作業の大半は次の方法でも実行できます。

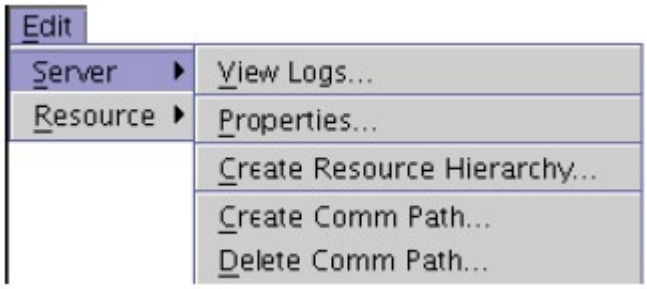
1. ツールバーから状態表示の左ペインにあるグローバル・リソースを右クリック。
2. 状態表示の右ペインにあるリソース・インスタンスを右クリック。

   ※右クリックの方法を使用すると、 **[Edit]** メニューを使用したときに必要な情報を入力しないですみます。

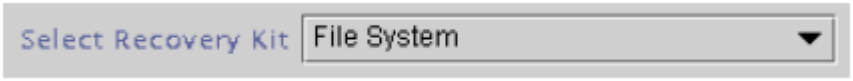
# 6.9.5.1. リソース階層の作成

プライマリ・サーバで以下を実行して、**[リソース作成ウィザード]** を開始します。

1. **[Edit] > [Server] > [Create Resource Hierarchy]** を選択します。



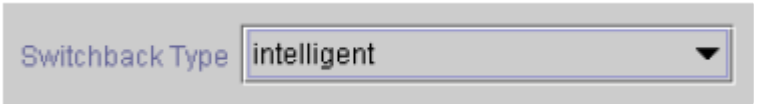

2. **[リカバリー・キットの選択]** ダイアログが表示されます。ドロップダウン・リストから **[ファイル・システム]** オプションを選択します。NAS Resource Hierarchy を、NFS マウント・システム・ファイルを使用して作成したファイル・システム階層にします。




**[Next]** をクリックして次に進みます。

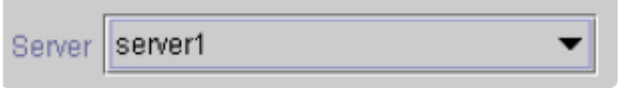
注記：階層作成手順のどの段階でも、**[Cancel]** ボタンをクリックすると、作成プロセス全体がキャンセルされます。

3. **[スイッチバック・タイプ]** ダイアログが表示されます。スイッチバック・タイプによって、フェイルオーバ後、バックアップ・サーバで NAS リソースがサービス中（アクティブ）になったとき、NAS リソースをプライマリ・サーバにスイッチバックする方法が決まります。スイッチバック・タイプは intelligent または automatic です。インテリジェント・スイッチバックの場合、リソースをプライマリ・サーバに戻すために管理者の操作が必要ですが、自動スイッチバックの場合は、プライマリ・サーバがオンラインに戻り、LifeKeeper との通信パスが再確立されるとすぐにスイッチバックが実行されます。




**[Next]** をクリックして次に進みます。

4. **[サーバ]** ダイアログが表示されます。NAS リソースを作成するサーバの名前を選択します（通常は、これがプライマリ・サーバです）。クラスタ内のすべてのサーバがドロップダウン・リスト・ボックスに表示されます。

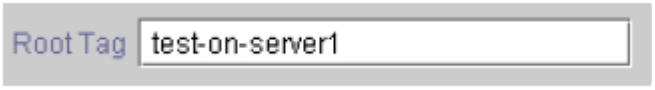


**[Next]** をクリックして次に進みます。

5. NAS (File System) Resource Hierarchy で保護する **[マウント・ポイント]** パスを選択します。すべての「ローカル」・ファイル・システム（つまり、共有ストレージを使用するファイル・システム） および NFS マウント・ファイル・システムがリストされます。ドロップダウン・リストから[NFS マウント・ファイル・システム]オプションを選択します。

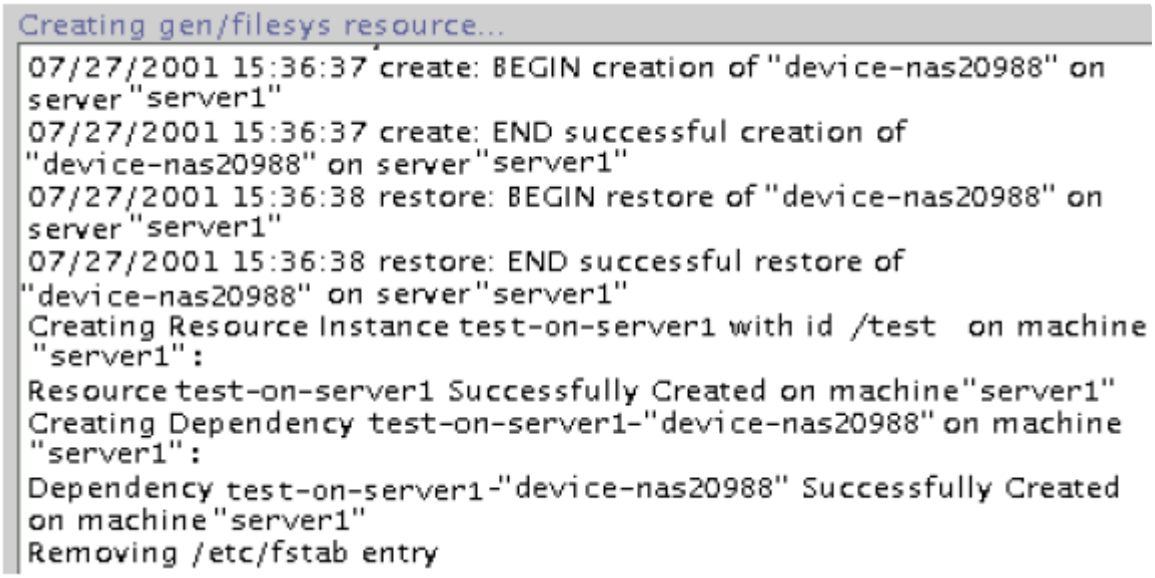
**[Next]** をクリックして次に進みます。

6. **[ルート・タグ]** ダイアログに、ターゲット・サーバ（つまり、上記で選択したサー バ）上のリソース・インスタンスの固有名が自動的に表示されます。デフォルトを 受け入れることもできますし、文字、数字、および特殊文字-、_、.、または/で構成 された固有のタグ名を入力することもできます。



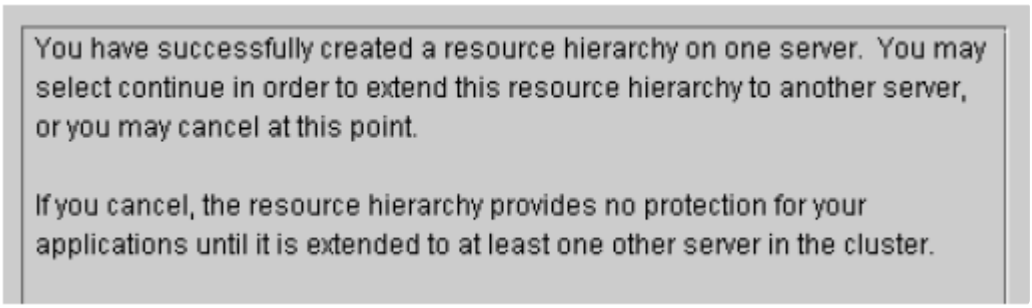

**[Create Instance]** をクリックします。

7. 階層の作成を開始することを示す情報ボックスが表示されます。

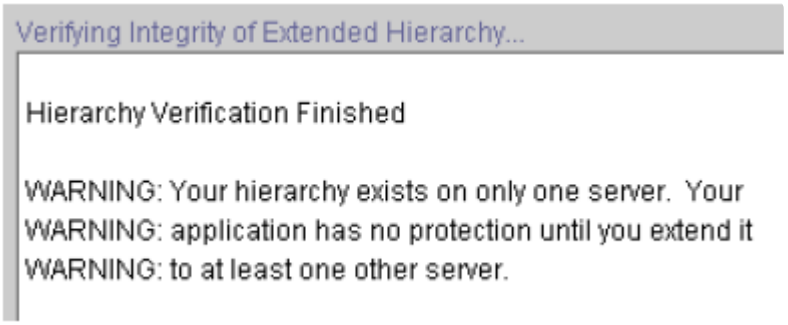



**[Next]** をクリックして次に進みます。

8. NAS リソース階層の作成に成功したことを示す情報ボックスが表示されます。リソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにリソース階層を 拡張する必要があります。




リソースを拡張する場合は、 **[Continue]** をクリックします。 後でリソースを拡張する場合は、 **[Cancel]** をクリックします。




9. **[Done]** をクリックして、[リソース階層の作成]メニューを終了します。 を LifeKeeper で保護するには、クラスタ内の別のサーバにリソース階層を拡張する必要があります。

# 6.9.5.2. リソース階層の削除

プライマリ・サーバで次の作業を行います。LifeKeeper 構成内のすべてのサーバから NAS リソースを削除するには、次の手順を実行します。

1. **[LifeKeeper GUI]** メニューから **[Edit]** を選択し、次に **[Resource]** を選択します。ドロップダウン・メニューから、 **[Delete Resource Hierarchy]** を選択します。

2. NAS リソース階層を削除する **ターゲット・サーバ** の名前を選択します。

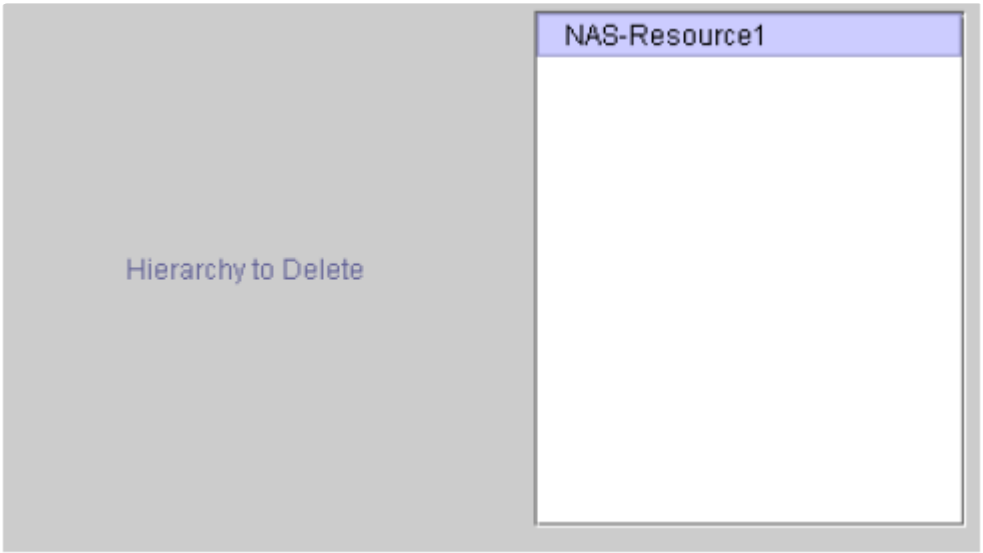
   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインから個々のリソース・インスタンスを右クリックして **[リソース階層の削除]** を選択した場合、このダイアログ・ボックスは表示されません。

   **[Next]** をクリックして次に進みます。

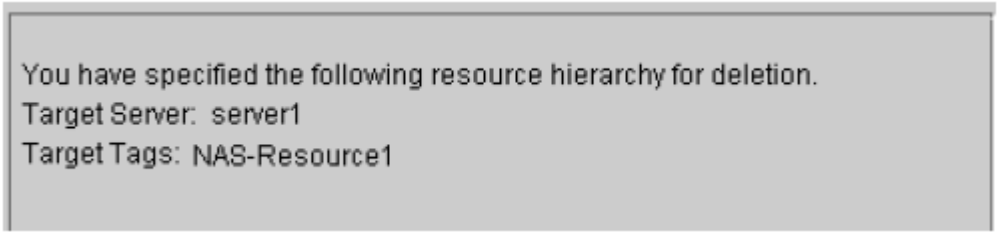
3. **[削除する階層]** を選択します。削除するリソース階層を特定して、強調表示にします。

   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインか ら個々のリソース・インスタンスを右クリックして **[リソース階層の削除]** を選択した場合、このダイアログ・ボックスは表示されません。

   **[Next]** をクリックして次に進みます。

4. 選択したターゲット・サーバと、削除の対象として選択した階層を確認する情報ボ ックスが表示されます。




**[Delete]** をクリックして次に進みます。

5. File System NAS リソース・インスタンスの削除に成功したことを示す情報ボックス が表示されます。

6. **[Done]** をクリックして、 **[リソース階層の削除]** メニューを終了します。

# 6.9.5.3. 階層の拡張

階層は、作成後に、クラスタ内の別のサーバに拡張する必要があります。リソース・インスタンスを拡張するには、次の 3 つの方法が可能です。

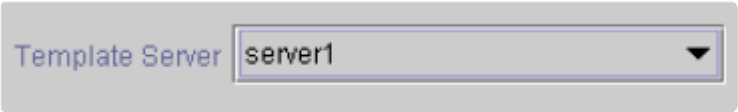
1. NAS リソース階層の作成に成功したときに、 **[Continue]** を選択して、バックアップ・サーバにリソース階層を拡張する処理に進むことができます。

2. LifeKeeper GUI の左ペインまたは右ペインで拡張されていない階層を右クリックします。

3. LifeKeeper GUI で、ドロップダウン・メニューから **[Edit] > [Resource] > [Extend Resource Hierarchy]** を選択します。選択すると、 **[リソース階層の拡張]** ウィザードが起動されます。 **[Extend Resource Hierarchy]** オプションの **[Accept Defaults]** ボタンは、 LifeKeeperの **[リソース階層の拡張]** の既定値をよく知っていて、値の入力や確認をしないで素早く LifeKeeper リソース階層を拡張したいユーザ向けです。GUI ダイアログを使用して対話的に段階を追って LifeKeeper リソース階層を拡張する場合は、 **[Next]** ボタンを選択します。

   a. 最初に表示されるダイアログ・ボックスでは、NAS リソース階層が現在サービス中であるテンプレート・サーバを選択します。 **[テンプレート・サーバ]** で選択するテンプレート・サーバと、次のダイアログ・ボックスの **[拡張するタグ]** で選択する拡張するタグが、サービス中（アクティブ）のリソース階層を表すことを忘れないでください。選択したテンプレート・サーバでサービス中になっていない リソース・タグを選択すると、エラー・メッセージが表示されます。このダイアログのドロップダウン・ボックスには、クラスタ内の全サーバの名前が示されます。

      **注記：** NAS リソース階層作成後、引き続きリソース階層の拡張作業に入った場合は、作成段階でテンプレート・サーバが特定されているため、このダイアログ・ボックスは表示されません。これは、GUI ウィンドウの左ペインの NAS リソース・アイコンまたは右ペインの NAS (File System) リソース・ボックスを右クリックして、 **[リソース階層の拡張]** を選択した場合も同様です。

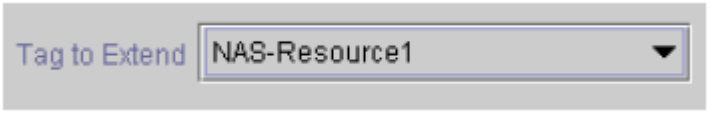
      **注記：** 階層拡張の途中で **[Cancel]** ボタンをクリックすると、拡張処理が取り消されます。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

      **[Next]** をクリックして次に進みます。

   b. **[拡張するタグ]** を選択します。これは、テンプレート・サーバからターゲット・サーバに拡張する NAS インスタンスの名前です。ウィザードのドロップダウン・リスト・ボックスには、テンプレート・サーバ上に作成されているすべてのリソースが表示されます。

      注記： ここでも、NAS リソース階層の作成後すぐにリソース階層の拡張作業に進んだ場合は、作成段階ですでにリソースのタグ名が特定されているため、このダイアログ・ボックスは

表示されません。これは、GUI ウィンドウの左ペインの NAS (File System) リソース・アイコンまたは右ペインの NAS (File System) リソース・ボックスを右クリックして、[リソース階層の拡張]を選択した場合も同様です。

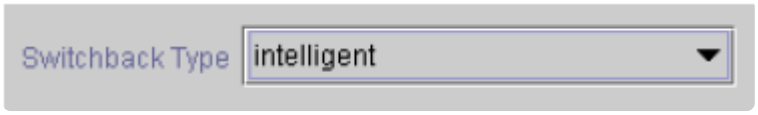



**[Next]** をクリックして次に進みます。

c. NAS リソース階層を拡張するターゲット・サーバを選択します。

**[Next]** をクリックして次に進みます。

d. **[スイッチバック・タイプ]** ダイアログが表示されます。スイッチバック・タイプによって、フェイルオーバ後、バックアップ・サーバで NAS リソースがサービス中（アクティブ）になったとき、NAS リソースをプライマリ・サーバにスイッ チバックする方法が決まります。スイッチバック・タイプは intelligent または automatic です。インテリジェント・スイッチバックの場合、リソースをプライマリ・サーバに戻すために管理者の操作が必要ですが、自動スイッチバックの場合 は、プライマリ・サーバがオンラインに戻り、LifeKeeper との通信パスが再確立されるとすぐにスイッチバックが実行されます。




**[Next]** をクリックして次に進みます。

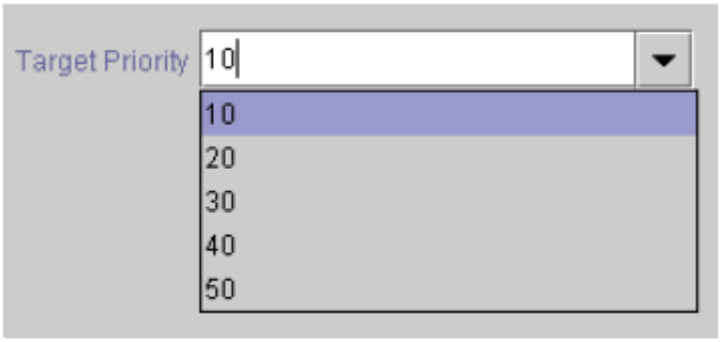
e. テンプレートの優先順位を選択または入力します。これはサーバで現在サービス中の NAS 階層の優先順位です。優先順位は、1～999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります（数字 1 が最高の優先順位に相当し ます）。拡張操作時には、別のシステムで既に使用中の優先順位をこの階層に対 して指定することはできません。デフォルト値を推奨します。

**注記：** このフィール ドは階層を最初に拡張するときだけ表示されます。

**[Next]** をクリックして次に進みます。

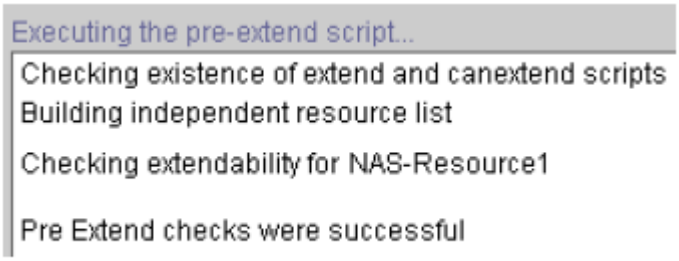
f. ターゲットの優先順位を選択または入力します。これは、別サーバにある同等の階層に対する新しく拡張する NAS 階層の優先順位です。1～999 の範囲の未使用の値が有効で、リソースの連鎖フェイルオーバ・シーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位は高くなります（1 は最高の優先 順位を表わします）。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連

続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。




**[Next]** をクリックして次に進みます。

g. 環境がチェックされ、このリソースを拡張するのに必要な要件がすべて満たされていることを示す情報ボックスが表示されます。満たされていない要件がある場合、 **[Next]** ボタンが使用できなくなり、 **[戻る]** ボタンが有効になります。




リソースの拡張の設定を変更する場合は、 **[Back]** ボタンをクリックします。 後でリソースを拡張する場合は、[キャンセル]をクリックします。リソース階層の拡張を実行する場合は、 **[Next]** をクリックします。

**[Finish]** をクリックして、NAS リソース・インスタンスの拡張に成功したことを確認します。

4. **[Done]** をクリックして、 **[リソース階層の拡張]** メニューを終了します。

**注記：** 必ず両方のサーバで新しいインスタンスの機能をテストしてください。

# 6.9.5.4. 階層の拡張解除

1. LifeKeeper GUI メニューで、 **[Edit] > [Resource] > [Unextending Resource Hierarchy]** を選択します。

2. NAS リソースを拡張解除する **[ターゲット・サーバ]** を選択します。リソースが現在サービス中（アクティブ）のサーバは選択できません。

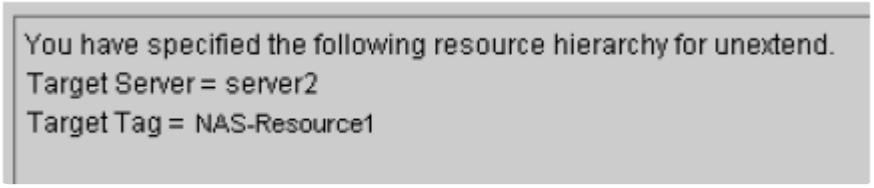
   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインから個々のリソース・インスタンスを右クリックして拡張解除作業を選択した場合、 このダイアログ・ボックスは表示されません。

   **[Next]** をクリックして次に進みます。

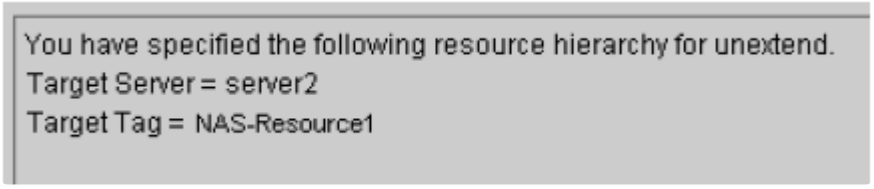
3. **[拡張解除する階層]** を選択します。

   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインから個々のリソース・インスタンスを右クリックして拡張解除作業を選択した場合、 このダイアログ・ボックスは表示されません。

   **[Next]** をクリックして次に進みます。

4. 拡張解除するために選択したターゲット・サーバと NAS リソース階層を確認する情報ボックスが表示されます。

   **[Unextend]** をクリックします。

5. NAS リソースの拡張解除に成功したことを示す別の情報ボックスが表示されます。

6. **[Done]** をクリックして、 **[リソース階層の拡張解除]** メニューを終了します。

# 6.9.5.5. リソース階層のテスト

NAS リソース階層のテストは、手動でサーバを切り替えて、プライマリ・サーバからバックアップ・サーバへのリソース・インスタンスのフェイルオーバをシミュレートすることによって行えます。

## LifeKeeper GUI からの手動切り替えの実行

LifeKeeper GUI で **[Edit] > [Resource] > [In Service]** を選択すると、手動切り替えを開始できます。たとえば、バックアップ・サーバでサービス中リクエストが実行されると、NAS リソース階層がバックアップ・サーバ側でサービス中になり、プライマリ・サーバ側ではサービス休止になります。この時点で、元のバックアップ・サーバがプライマリ・サーバになり、元のプライマリ・サーバがバックアップ・サーバになります。

**[Our of Service]** を実行した場合、リソース階層はサービス休止になりますが、別のサーバでサービス中にはなりません。

# 6.9.6. トラブルシューティング

| 症状 | 考えられる要因 |
|---|---|
| LifeKeeper のフェイルオーバ操作 がアンマウント・ビジー・エラーで失敗しました。 | NAS 階層をビルドするために使用するファイル・システム・キットでは、フェイルオーバ条件でマウントされたファイル・システムを使用している LifeKeeper 保護下に入っていないプロセスを検出したり、除去したりすることはできません。そのため、LifeKeeper 保護下に入れるプロセスのみが、NAS 保護下のファイル・システムを使用するように設定することを強くお勧めします。この障害が発生したら、ファイル・システムを使用してプロセスを識別し、消滅させる必要があります。 **fuser -m** コマンドを使用すると、ファイル・システムへ現在アクセスしているプロセスを判別できます。この使用法の詳細については、 **fuser** のマニュアル・ページを参照してください。 |
| LifeKeeper が、サーバ名を使用してマウントされたファイル・システムの ローカル・リカバリーを行います。 | NAS Recovery Kit の保護下にあるファイル・システムが IP アドレスではなくホスト名を使用してマウントされたら、 NAS リソースの作成後に、LifeKeeper が次のようなメッセージをログに記録します。<br><br>`. . . WARNING: Mon Aug 26 11:27:01 2002:`<br><br>LifeKeeper protected filesystem resource “tmp/mnt-on-tom.brown.com” (/tmp/mnt) is in service but not mounted<br><br>. . . Attempting Local Recovery of resource<br><br>LifeKeeper は、この時点で IP アドレスを使用してファイル・システムを再マウントします。ただし、問題が発生したら、LifeKeeper は NAS リソースをバックアップ・サーバへフェイルオーバするか（拡張された場合）、またはリソースをサービス休止状態にします（リソースが拡張されていない場合）。<br><br>**推奨される対策：** ローカル・リカバリーが成功したら、それ以上処置は必要ありません。ただし、ローカル・リカバリーが失敗したら、次の操作を行う必要があります。<br><br>1. LifeKeeper で NAS リソースを削除します。<br><br>2. ホスト名ではなく IP アドレスを使用してファイル・システムを再マウントします。<br><br>3. NAS リソースを作成し直します。 |

# 6.9.6.1. エラーメッセージ

本セクションでは、LifeKeeper NAS リソース階層の作成、拡張時、またはリソースの削除、復元時に表示される可能性のあるメッセージの一覧を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

その他の LifeKeeper コンポーネントからもメッセージが表示される可能性があります。 そのような場合は、適切な LifeKeeper コンポーネントのマニュアルを参照してください。

このセクションのメッセージは、次のトピックに分類されています。

- 共通エラー・メッセージ
- 階層の作成
- 階層の拡張
- 階層の削除、復元、リカバリー

## 共通エラーメッセージ

| エラー番号 | エラーメッセージ |
|---|---|
| 000002 | Usage error （使用法のエラー） |
| 000010 | Error getting resource information （リソース情報取得時のエラー） |
| 000011 | Both Tag and ID name not specified （タグ名および ID 名が指定されていません） |
| 000019 | Resource not found on local server （ローカル・サーバにリソースが見つかりません） |
| 000022 | END failed hierarchy <tag name> in service on server <server name> （サーバ <サーバ名> でサービス中の階層 <タグ名> の END が失敗しました） |
| 000026 | END failed ACTION for <tag name> on server <server name> due to <signal> signal （<シグナル> シグナルのためにサーバ <サーバ名> の <タ グ名> ACTION の END が失敗しました） |

## 階層の作成

| エラー番号 | エラーメッセージ |
|---|---|
| 000012 | Usage error（使用法のエラー） |
| 000013 | Error getting resource information（リソース情報取得時のエラー） |
| 000014 | Resource with either matching tag <tag name> or ID exists （タグ <タグ名> または ID が一致するリソースが存在します） |
| 000015 | ins_create failed on server <server name> （サーバ <サーバ名> で ins_create に失敗しました） |

| | |
|---|---|
| 000018 | Error creating resource <tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の作成エラー） |
| 000021 | Removing resource instance < tag name> from server <server name> due to an error during creation （作成時のエラーにより、リソース・インスタンス <タグ 名> をサーバ <サーバ名> から削除しています） |
| 000023 | Error bringing resource < tag name> in service on server <server name> （サーバ <サーバ名> でリソース <タグ名> をサービス中 にするときにエラー） |
| 000024 | Failed resource creation of resource < tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の作成に失敗しました） |
| 000027 | Removing file system dependency from <parent tag> to <child tag> on server <server name> due to an error during creation （作成時のエラーにより、サーバ <サーバ名> の <親タグ> から <子タグ> へのファイル・システム従属関係を削除し ています） |
| 000028 | Removing file system hierarchy <filesys tag> created by <parent tag> on server <server name> due to an error during creation （作成時のエラーにより、サーバ <サーバ名> の <親タグ> によって作成されたファイル・システム階層 <ファイルシステム・タグ> を削除しています） |
| 000029 | Switchback type mismatch between parent <parent tag> and child <child tag> on server <server name> （サーバ <サーバ名> の親 <親タグ> と子 <子タグ> のスイ ッチバック・タイプが一致していません）<br><br>**処置：** スイッチバック・タイプの不一致は予想外の動作を 引き起こすことがあります。ins_setas コマンドを使用して スイッチバック・タイプを手動で変更することによって、 この不一致をなくすことができます。 |
| 000030 | create: tag name not specified （作成時：タグ名が指定されていません）<br><br>または<br><br>extend: tag name not specified （拡張時：タグ名が指定されていません） |

## 階層の拡張

| エラー番号 | エラーメッセージ |
|---|---|
| 000003 | Template resource < tag name> on server <server name> does not exist （サーバ <サーバ名> にテンプレート・リソース <タグ名> が存在しません） |
| 000004 | Template resource < tag name> cannot be extended to server <server name> because it already exists there （テンプレート・リソース <タグ名> がサーバ <サーバ名> に すでに存在するため拡張できません） |
| 000005 | Cannot access canextend script on server <server name> （サーバ <サーバ名> の canextend スクリプトにアクセスできません） |
| 000006 | Cannot access extend script <path to extend> on server <server name> （サーバ <サーバ名> の extend スクリプト <extend へのパス> にアクセスできません） |

| | |
|---|---|
| 000007 | Cannot access depstoextend script <path to depstoexend> on server <server name> （サーバ <サーバ名> の depstoextend スクリプト <depstoextend へのパス> にアクセスできません） |
| 000008 | Cannot extend resource < tag name> to server <server name> （サーバ <サーバ名> にリソース <タグ名> を拡張できません） |
| 000009 | Either <templatesys> or <templatetag> argument missing （<テンプレート・システム> または <テンプレート・タグ> 引数が指定されていません） |
| 000014 | Resource with either matching tag <tag name> or ID exists （タグ <タグ名> または ID が一致するリソースが存在します） |
| 000015 | ins_create failed on server <server name> （サーバ <サーバ名> で ins_create に失敗しました） |
| 000018 | Error creating resource <tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の作成エラー） |
| 000025 | END failed resource extension of < tag name> on server <server name> due to a "<signal>" signal - backing out changes made to server （"<シグナル>" シグナルのためにサーバ <サーバ名> の <タグ名> リソース拡張の END が失敗しました - サーバに加えた変更を元に戻しています） |
| 000030 | create: tag name not specified （作成時：タグ名が指定されていません）<br><br>または<br><br>extend: tag name not specified （拡張時：タグ名が指定されていません） |

### 復元

| エラー番号 | エラーメッセージ |
|---|---|
| 000023 | Error bringing resource < tag name> in service on server <server name> （サーバ <サーバ名> でリソース <タグ名> をサービス中にするときにエラー） |

### リソース監視

| エラー番号 | エラーメッセージ |
|---|---|
| 000001 | Calling sendevent for resource < tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の sendevent を呼び出 しています） |

## NAS Recovery Kit のエラーメッセージ

| エラー番号 | エラーメッセージ |
|---|---|

| | |
|---|---|
| 107001 | Creation of NAS device with tag id <tag id> on server <LifeKeeper server name> failed. （サーバ <LifeKeeper サーバ名> でタグ ID <タグ ID> を使用してNAS デバイスを作成しようとしましたが、失敗しました。） |
| 107002 | Error getting list of IP addresses for NFS server device <NFS server name> on server <LifeKeeper server name>.（サーバ <LifeKeeper サーバ名> で NFS サーバ・デバイス <NFSサーバ名> の IP アドレスのリストを取得中にエラーが発生しました。） |
| 107003 | Error attempting to find active address to NFS server <NFS server name> on server <LifeKeeper server name>. （サーバ <LifeKeeper サーバ名> で NFS サーバ <NFS サーバ名> の アクティブ・アドレスを検出試行中にエラーが発生しました。） |
| 107004 | Error in format of device ID <resource device>. （デバイス ID <リソース・デバイス> のフォーマット・エラー。） |
| 107005 | Cannot bring NAS resource <tag id> in service on server <LifeKeeper server name>. （サーバ <LifeKeeper サーバ名> で NAS リソース <タグ ID> をサービス中の状態にできません。）<br><br>**処置：** 問題を修正したあと、手動でリソースをサービス中にしてください。 |
| 107006 | create: Device not specified. （作成：デバイスが指定されていません。） |
| 107007 | Null Device returned by getId on <LifeKeeper server name> （<LifeKeeper サーバ名> で getId によってヌル・デバイスが戻されました。） |
| 107008 | Cannot open /etc/mtab file on <LifeKeeper server name>. （<LifeKeeper サーバ名> で/etc/mtab ファイルを開けません。） |
| 107009 | Illogical settings for NAS defaults on <LifeKeeper server name>. Using defaults of 120 for LKNFSTIMEOUT and 5 for LKNFSSYSCALLTO. （<LifeKeeper サーバ名> 上の NAS デフォルトの設定が非論理的 です。LKNFSTIMEOUT には 120 のデフォルトを、 LKNFSSYSCALLTO には 5 を使用しています。）<br><br>**処置：** LKNFSSYSCALLTO の 3 倍の値に 5 を加算した値が LKNFSTIMEOUT より小さくなるように、NAS デフォルト値をリセットします。 |
| 107010 | Error: detected conflict in expected tag name <tag id> on target machine <LifeKeeper server name>. （エラー：ターゲット・マシン<LifeKeeper サーバ名> 上の予想さ れたタグ名 <タグ ID> で競合が検出されました。）<br><br>**処置：** 競合するリソースを削除し、階層を再拡張してください。 |
| 107011 | Error: mkdir of "/tmp/nas_mntpt.2915" on "mouse" failed: "permission denied". （エラー："mouse"上の"/tmp/nas_mntpt.2915"の mkdir が失敗しました："権限拒否"） |
| 107012 | Error: Exported file system <NFS exported file system name> cannot be accessed on <server name>. （エラー：<サーバ名> で、エクスポートされたファイル・システム <NFS エクスポート・ファイル・システム名> にアクセスできません。）<br><br>考えられる原因<br><br>- LifeKeeper ノードが、NFS サーバのエクスポートされたシステムのリストに含まれていない。 |

| | |
|---|---|
| | - エクスポートされたシステムのリストに、 **showmount** コマンド で表示されない矛盾したエントリが含まれている。 (つまり、エクスポートされたシステムのリストによってシステムが全システムと特定のシステムの両方にエクスポートされる場合、 **showmount** では特定のシステムのみが報告されます。)<br><br>**処置：** エクスポートされたファイル･システムのアクセス問題を訂正し、階層を再拡張してください。 |
| 107013 | Error: Mount authorization check for "172.25.113.25:/ export" on "fred" appears to be hung. Exiting. (エラー："fred"での"172.25.113.25:/ export"のマウント認証チェックが停止した可能性があります。終了します。)<br><br>**処置：** アクセス問題を訂正し、階層を再拡張してください。 |
| 107015 | Exported file system <tag> cannot be accessed on <LifeKeeper server name>. (サーバ <LifeKeeper サーバ名> でエクスポートされたファイルシステム <tag名> にアクセスできませんでした。) |

# 6.9.6.2. LifeKeeper GUI 関連のエラー

| エラー番号 | エラーメッセージ |
|---|---|
| 104901 | The mount point %s is mounted （マウント・ポイント %s はマウントされています）<br><br>**処置：** マウントされていないマウント・ポイントを指定してくだ さい。 |
| 104902 | The mount point %s is not an absolute path（マウント・ポイント %s が絶対パスでありません）<br><br>**処置：** スラッシュで始まるマウント・ポイントを指定してくださ い。 |
| 104903 | The mount point %s is not empty.（マウント・ポイント %s が空でありません）<br><br>**処置：** 存在しないマウント・ポイントまたは空のマウント・ポイ ントを指定してください。 |

# 6.10. NFS Recovery Kit 管理ガイド

## NFS Server Recovery Kit のドキュメンテーション

SIOS Protection Suite for Linux NFS Server Recovery Kit は、SIOS Protection Suite 環境でネットワークファイルシステム(NFS)ソフトウェアの障害回復能力を高めます。プライマリ NFS サーバに障害が発生した場合には、人手の介入なしに迅速に指定のバックアップサーバでシステムを復旧することが可能です。

SIOS Protection Suite for Linux NFS Server Recovery Kit の概要

## SIOS Protection Suite のドキュメンテーション

以下は、SIOS Technology Corp が発行している SIOS Protection Suite for Linux 関連ドキュメントの一覧です。

- SPS for Linux テクニカルドキュメンテーション
- SPS for Linux リリースノート
- SIOS Technology Corp. ドキュメンテーション

## 参考ドキュメント

以下は SIOS Protection Suite NFS Server Recovery Kit に関連する参考ドキュメントの一覧です。

- NFS オンラインドキュメンテーション
- *Managing NFS and NIS, Hal Stern, O'Reilly & Associates, Inc. 1991*

# 6.10.1. 概要

## LifeKeeper for Linux NFS Server Recovery Kit 概要

NFS Server Recovery Kit は、Filesystem Recovery Kit (steeleye-lk パッケージの一部として提供) および IP Recovery Kit (steeleye-lkIP) と階層的に連携して、高可用性 NFS サービスを実現します。

NFS Server Recovery Kit は、IP リソースと共有マウントポイントを含むファイルシステムリソースがクラスタ内の同じサーバで常に In Service であることを保証します。LifeKeeper が保護する IP リソースを使用してファイルシステムをマウントするクライアントは、実際のエクスポートサービスが手動で、または障害に対する対処としてクラスタ内のサーバ間で切り替えられても、事実上中断なしにボリューム上のファイルの処理を継続できます。NFSセッションの復帰時間はクライアント - サーバー間の状態に依存します。例としてNFSv3においては、TCPのプロトコルタイムアウトはUDPに比べて長くなります。NFSで使用する適切なトランスポート層プロトコルを決定するには、OSベンダーからの推奨や各トランスポート層プロトコルのメリットとデメリット、ご利用の環境を踏まえてご検討ください。

※NFSv4においては、OSベンダーからTCPの利用が強く推奨されており、NFS ARKはTCPとNFSv4で検証されています。

NFS Server Recovery Kit の 7.4 では、NFS v4 **擬似ファイルシステム** のエクスポートがサポートされ、サーバ上のエクスポートされるすべてのオブジェクトへシームレスにアクセスできるようになります。7.4 以前のバージョンではクライアントはアクセスするために各共有サーバのファイルシステムをマウントする必要がありました。NFS v4 で、サーバは NFS アクセスのためにエクスポートされる各サーバディレクトリもしくはファイルシステムに対してエクスポートの制御を指定します。またこれらのエクスポートの制御から、サーバは、エクスポートされたディレクトリ間のすべてのエクスポートされたデータを **1 つのディレクトリツリー** として表示します。このツリーは **擬似ファイルシステム** として知られており、NFS v4 サーバの擬似ルートで起動されます。擬似ファイルシステムにより、NFS v4 クライアントが、その実装に応じて、サーバのエクスポートされたすべてのデータにアクセスするためにサーバの擬似ルートを単独でマウントできるようになります。

スイッチオーバまたはフェイルオーバを実行している間、ファイルシステム上のファイルはすべて一時的に使用できなくなりますが、リソースの転送が完了すると再び使用可能になり、クライアント側に切り替えは意識されません。スイッチオーバの場合、これには 5 ~ 30 秒かかります。フェイルオーバの場合、リカバリ時間は、ファイルシステムの修復にかかる時間に左右されます。ディスクボリュームは、障害に極めて強く、数秒で修復が可能なジャーナリングファイルシステム (JFS) でフォーマットすることを強く推奨します。

# 6.10.2. 要件

LifeKeeper NFS Server Recovery Kit をインストールして設定する前に、次の要件を満たしていることを確認します。

- **サーバー** 。この Recovery Kit には、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート で説明されている要件に従って設定した 2 台以上のサーバーが必要です。サポート対象の Linux ディストリビューションについては、上記のリリースノートを参照してください。

- **LifeKeeper ソフトウェア** 。クラスタの各サーバーにインストールする LifeKeeper ソフトウェアのバージョン、および、適用する LifeKeeper ソフトウェアパッチのバージョンが同一であることが絶対条件です。具体的な LifeKeeper の要件については、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート を参照してください。

- **LifeKeeper IP Recovery Kit** 。各サーバーの LifeKeeper IP Recovery Kit のバージョンは同じにする必要があります。

- **IP ネットワークインターフェース** 。各サーバーは、イーサーネット TCP/IP をサポートするネットワークインターフェースを 1 つ以上必要とします。IP スイッチオーバが正しく動作するには、ローカルネットワークに接続されているユーザーシステムが標準の TCP/IP 仕様に準拠している必要があります。

  **注記**：各サーバーが必要とするネットワークインターフェースが 1 つだけであっても、異種メディア要件、スループット要件、単一障害点の排除、ネットワークのセグメンテーションといった多くの理由のために、複数のインターフェースを使用すべきです。

- **TCP/IP ソフトウェア** 。各サーバーは TCP/IP ソフトウェアも必要とします。

- **nfs-utils ソフトウェア** 。各サーバーでは、LifeKeeper と LifeKeeper NFS Server Recovery Kit の設定に先だって、nfs-utils パッケージの高可用性バージョンをインストールして設定しておく必要があります (OS のディストリビューションによっては、 `nfs-utils` は別パッケージで提供されています)。各サーバーに同じバージョンをインストールする必要があります。

  LifeKeeper インストールサポートセットアップスクリプトは HA クラスタ環境で使用する `nfs-utils` を設定します。さらに、 **NFS v2/v3 エクスポート** を保護するためにこのキットを使用する場合は `nfs-lock.service` および `nfs-server.service` がシステムブート時に自動的に起動されるように設定する必要があります。 **NFS v4** ルートエクスポートを保護する場合は、自動起動しないように設定してください。

- LifeKeeper NFS Server Recovery Kit をインストールまたは削除する具体的な方法については、SPS for Linux インストールガイド を参照してください。

# 6.10.3. 設定上の考慮事項

下記セクションでは、典型的な LifeKeeper NFS 設定の例も含めて NFS Server Recovery Kit の設定と管理を開始する前に考慮すべき事項についても記載しています。

LifeKeeper Core のリソース階層の設定方法については、 SPS for Linux テクニカルドキュメンテーション を参照してください。

- LifeKeeper における NFS サーバの設定
- 特定の設定上の考慮事項
- 設定例

# 6.10.3.1. 特定の設定上の考慮事項

LifeKeeper NFS Server Recovery Kit を使用する前に、次の事項を考慮する必要があります。

- LifeKeeper で保護する NFS ファイルシステムはプライマリサーバ (NFS リソースを作成するサーバ) によってエクスポートする必要があります。これは、NFS が実行中で、基礎ファイルシステムがマウントされていることを意味します。

  **注記:** NFS 経由で */home* ディレクトリを共有している場合は、*/home* が基礎ファイルシステムです。

- **読み書き可能なファイルシステムをエクスポートするときは、*sync* オプションを使用してください。** このオプションを使用すると、ファイルシステムのすべての書き込みをディスクにコミットしてから書き込み要求が完了します。LifeKeeper の保護下にある NFS ファイルシステムで *sync* オプションを使用しない場合、フェイルオーバ時にデータが失われる可能性があります。

- 基礎ファイルシステムは共有デバイス上にある必要があり、書き込み権限でマウントする必要があります。

- 基礎ファイルシステムがすでに LifeKeeper によって保護されている場合、そのファイルシステムはプライマリサーバ上で In Service で、最高の優先順位を持っている必要があります。基礎ファイルシステムが LifeKeeper の保護下にない場合、Recovery Kit によって保護下に置かれます。

- NFS Server Recovery Kit は、プライマリサーバで作成され In Service の IP リソースを必要とします。また、この IP リソースは、プライマリサーバで最高の優先順位を持っている必要があります。

- NFS リソースを作成する前に、クライアントは LifeKeeper が保護している IP アドレスを使用して NFS ファイルシステムをマウントできる必要があります。

- NFS ファイルシステムリソースを拡張するときには、各サーバでファイルシステムを同じマウントポイントでマウントする必要があります。

- NFSv4 ルートエクスポートを保護する場合、*/var/lib/nfs* は、マウント時に書き込み権限を持っている NFSv4 ルートファイルシステムへ移動されます。継続したアクセスを提供するため、*/var/lib/nfs* から NFSv4 ルートへのシンボリックリンクが作成されます。このため、NFSv4 でのアクティブ / アクティブ設定、NFS v2/v3 と NFSv4 の混在設定はサポートされていません。

- Oracle Recovery Kit は、共有データベースストレージ用に NFSv3 をサポートしています。NFSv4 のファイルロック機構のため、現時点では NFSv4 は Oracle Recovery Kit によってサポートされていません。

- NFS クライアントに Linux kernel 3.12 以降を採用しているシステムを使用する場合、kernel の仕様によりクライアント側は通信断時にロックを失います。その為、スイッチオーバやフェイルオーバ時にロックの引き継ぎが保証されません。

- NFSv4 において、クライアントは一定時間サーバと通信を行えない場合、ロックを失います。この値は、LifeKeeper では変数 NFS_V4_LEASE_TIME で扱っており、デフォルトでは

10(秒) が設定されています。スイッチオーバやフェイルオーバを実行しクライアントとサーバの通信が NFS_V4_LEASE_TIME に設定された時間より長く停止した場合は、そのクライアントが設定したファイルのロックは強制的に解除されます。この時間を変更する場合、/etc/default/LifeKeeper に、環境変数 NFS_V4_LEASE_TIME を適切な値で設定してください。
なお NFS_V4_LEASE_TIME の値を大きくした場合、上記の問題は起こりにくくなりますが、サーバが切り替わったあとのクライアントとの再接続に時間がかかるようになります。

- Red Hat Enterprise Linux 7以降、CentOS 7以降、Oracle Linux 7以降では NFSv2 はサポートされません

- Red Hat Enterprise Linux 8以降では NFS over UDP はサポートされません。

# 6.10.3.2. LifeKeeper における NFS サーバーの設定

本セクションでは、NFS Server Recovery Kit の設定と管理を開始する前に考慮すべき事項および典型的な LifeKeeper NFS 設定例について記載しています。

LifeKeeper Core のリソース階層の設定に関する手順については、SPS for Linux テクニカルドキュメンテーション を参照してください。

## NFS

次の表は、NFS Server Recovery Kit で重要な NFS ファイル、コマンド、デーモンについて説明しています。

| NFS コンポーネント | 説明 |
|---|---|
| *exports(5) (/etc/exports)* | NFS クライアントにエクスポートされたファイルシステムのアクセス制御リスト。ファイルの各行に、エクスポートポイント、ファイルシステムをマウントできるクライアントのリスト(オプション)、マウントパラメータのリスト(オプション) が記述されています。<br><br>**注記:** LifeKeeper が保護する NFS リソースを作成すると、ファイルシステムのエクスポート情報は *exports* ファイルから削除され、LifeKeeper によって管理されます。NFS リソースを削除すると、 *exports* ファイルにエクスポート情報がリストアされます。 |
| */var/lib/nfs* | 現在のエクスポート、クライアントマウント、ロックの状態およびその他の NFS 情報を含んだディレクトリです。バージョン 7.4 以降では、このディレクトリは、NFS v4 擬似ファイルシステムを保護する際に NFS エクスポートディレクトリに移動されています。 */var/lib/nfs* はプライマリおよびスタンバイシステムの両方で新しいロケーションへのシンボリックリンクに置き換えられます。 |
| */var/lib/nfs/etab* | NFS にエクスポートしたファイルシステムの現在の表を含むファイル。このファイルは **exportfs** コマンドによって管理され、ユーザーがこのファイルを直接編集することはありません。<br><br>**注記:** バックアップサーバーで NFS リソースのサービスを再開すると、その NFS ファイルシステムは、プライマリサーバーの *etab* ファイルから削除され、バックアップサーバーの *etab* ファイルに挿入されます |
| */var/lib/nfs/rpc_pipefs* | NFS のユーザー空間の通信に対するカーネルに使用されます。このディレクトリは、LifeKeeper のインストール中に */var/lib* に再配置されます。 |
| **exportfs(8)** */(usr/sbin/* | */var/lib/nfs/etab* のエクスポートされたファイルシステムの表を管理するコマンド。 |

| | |
|---|---|
| *exportfs)* | |
| **rpc.mountd(8)** *(/usr/sbin/ rpc.mountd)* | マウント要求を認証し、クライアントにファイルシステムのマウントを許可する場合にはファイルハンドルを返すデーモン。 |
| **rpc.nfsd(8)** *(/usr/sbin/ rpc.nfsd)* | クライアントのファイルシステム要求を処理するデーモン。 |
| **rpc.quotad(8)** *(/usr/sbin/ rpc.rquotad)* | NFS 経由でリモートマウントされたローカルファイルシステムのユーザーのクォータを返す rpc サーバー。 |
| rpc.lockd(8) *(/sbin/rpc.lockd)* | クライアントのファイルロック要求を処理するデーモン。 |
| rpc.statd(8) *(/usr/sbin/ rpc.statd)* | NFS クライアントとサーバーのステータスを監視し、ステータスを通知するデーモン。このデーモンは、NFS ファイルロックを適切に機能させるために、実行しておく必要があります。 |
| portmap/ rpcbind | RPC プログラム番号をポート番号に変えるデーモンプロセス。このデーモンは、NFS のために実行しておく必要があります。このプロセスが失敗するとスタンバイノードへのスイッチオーバを強制します。いくつかのシステムでは、この機能はポートマップにより提供されます。 rpcbindにより提供される場合もあります。 |
| rpc.idmapd | NFS v4 で、ユーザー名 / グループ名とユーザー ID/ グループ ID を一致させるデーモンプロセスです。このプロセスは NFS v4 では必ず起動させる必要がありますが NFS v2/ v3 では必要ありません。 |

## エクスポートの考慮事項

特定のエクスポートされたファイルシステムに対する LifeKeeper の保護は、exports(5) のマニュアルページで説明されている形式のエクスポートオプションによって決まります。特に、ホスト制限のフォーマットに注意してください。正当なホスト制限は、単独ホスト、ネットグループ、ワイルドカードで指定されたホスト名およびネットマスクの 4 つのみです。

特に、ワイルドカードで指定された IP アドレス (172.13.4.* など) は不正であり、スイッチオーバまたはフェイルオーバでファイルハンドルの陳腐化を招く恐れがあります。
exportfs -v を実行し、返されたエクスポートの記述とマニュアルページに説明されているフォーマットを手動で比較することで、慎重にチェックを行ってください。 (申し訳ございませんが、exportfs ではチェックは行えず、一部の不正なエクスポートフォーマットを受け入れてしまいます)。

# RPC.MOUNTD 再起動

複数の NFS リソース階層がある特定の条件下では、
rpc.mountd は、使用可能なエクスポートの一覧を適切に公表できません。そのため、リストア時の NFS Recovery Kit は停止し、rpc.mount を再開して、エクスポートの適切な一覧をすべてのクライアントが利用できるようにします。rpc.mount を停止し、再開するアクションは、 */etc/default/LifeKeeper*

内の RESTARTMOUNTD エントリによって制御されます。デフォルトでは、このエントリは次のように true に設定され、NFS リストア時に必ず rpc.mount の停止と再起動が行われます。

**RESTARTMOUNT=true**

このアクションをオフにするには、次のように設定します。

**RESTARTMOUNT=false**

# NFS リソース階層

LifeKeeper が保護する NFS リソースを作成すると、LifeKeeper によって次の階層が作成されます。

- NFS ファイルシステムリソース (親またはルート)
- HA-NFS リソース
- ファイルシステムリソース (基礎ファイルシステム)

NFS リソースを作成する前に、IP アドレスリソースを作成する必要があります。

NFS リソースを作成する前に、ファイルシステムリソースを作成することができます。この場合は、ファイルシステムリソースに割り当てる名前を選択できます。そうでない場合は、NFS リソース作成時に、NFS Server Recovery Kit によって自動的にファイルシステムリソースが作成されます。

# サービス自動起動の設定

NFSv2/NFSv3 領域を保護する場合は、以下のコマンドを実行して nfs-server.service を自動起動に設定してください。

```
systemctl start nfs-server.service
systemctl enable nfs-server.service
```

また、NFSv4 領域を保護する場合は、nfs-server.service を自動起動に設定しないでください。

## rpcbind/portmapの自動起動設定

rpcbindあるいはportmapの自動起動が有効になっていることを確認してください。

```
systemctl is-enabled rpcbind.socket
```

有効になっていない場合は、以下のコマンドで自動起動を有効にしてください。

```
systemctl start rpcbind.socket
systemctl enable rpcbind.socket
```

# NFSv2/v3 と NFSv4 の扱い

LifeKeeper は、オプションに /etc/exports にて "fsid=0" が指定されている領域を NFSv4 として保護します。 "fsid=0" の指定がない場合は NFSv2/3 として保護します。なお、この2つを混在することはできません。

# 6.10.3.3. 設定例

本セクションの設定例は、共有(または複製) ディスク上での NFS インスタンスの設定方法を示しています。それぞれの図は、設定の種類と NFS パラメータとの関係を示しています。また、各設定は、このセクションで説明している設定の規則と要件に従って、NFS 設定と LifeKeeper ソフトウェアとの互換性を確保しています。

本セクションでは、まず設定の要件を説明し、次に **NFS v2/v3** でのアクティブ / スタンバイ 、 **NFS v2/v3** でのアクティブ / アクティブ 、 **NFS v4** でのアクティブ / スタンバイ の設定例を示します。

> 注記: NFS v4 におけるアクティブ / アクティブはサポートされていません。また、NFS v2/v3 と NFS v4 の組み合わせもサポートされていません。

本セクションに挙げたものは可能な設定の例にすぎませんが、これらの設定例を理解し、設定規則に従うことは、ユーザのコンピューティング環境で実現可能なソリューションを明確にし、設定する上で役立ちます。

## 設定の要件

NFS タグ名は、LifeKeeper が保護するファイルシステムを示す任意の名前です。LifeKeeper によって示される標準タグ名は「 *nfs-<export point>* 」です。

設定例について理解するには、基礎のファイルシステムが必ず共用または複製ディスク上になければならないことを考慮に入れる必要があります。このファイルシステムはペアになったシステムのそれぞれからマウントできる必要があります。

## 設定例

- NFS v2/v3 でのアクティブ / スタンバイ設定
- NFS v2/v3 でのアクティブ / アクティブ設定

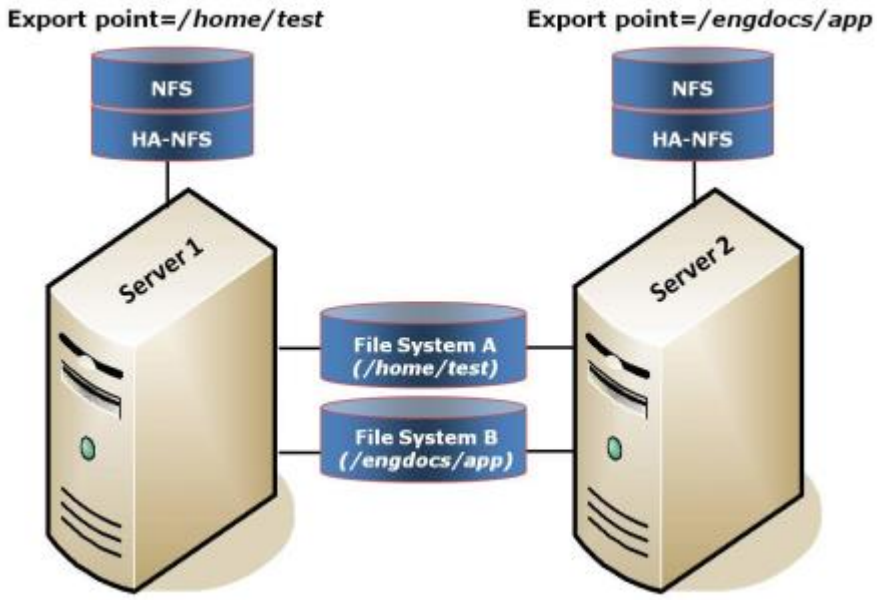
## 関連情報

- 設定上の考慮事項

# 6.10.3.3.1. NFS v2/v3 でのアクティブ / アクティブ

**NFS v2/v3** でのアクティブ / アクティブの例では、複数のシステムが NFS を実行し、ファイルシステムをエクスポートします。

## NFS v2/v3 でのアクティブ / アクティブ構成例




### 設定に関する注記

- NFS ソフトウェアを両方のサーバにインストールしておく必要があります。
- 最初は、 *Server 1* が */home/test* をエクスポートし、 *Server 2* が */engdocs/app* をエクスポートしています。スイッチオーバ時には、1 つのシステムで両方のファイルシステムをエクスポートできます。
- ファイルシステム A は、エクスポートポイント、 */home/test* の基礎ファイルシステムです。ファイルシステム B は、エクスポートポイント、 */engdocs/app* の基礎ファイルシステムです。
- 基礎ファイルシステムは異なる共有ディスク上にあります。

### *Server 1* での 1 つ目のリソース階層の作成

| | |
|---|---|
| Server: | *Server1* |
| Export Point: | */home/test* |
| IP tag: | ip-172.17.100.202 |
| NFS Tag: | *nfs-/home/test* |

### 1 つ目のリソース階層の Server 2 への拡張

| | |
|---|---|
| Template Server: | *Server1* |
| Tag to Extend: | *nfs-/home/test* |
| Target Server: | *Server2* |
| Target Priority: | 10 |

### Server 2 での 2 つ目のリソース階層の作成

| | |
|---|---|
| Server: | *Server2* |
| Export Point: | */engdocs/app* |
| IP Tag: | ip-172.17.100.203 |
| NFS Tag: | *nfs-/engdocs/app* |

### 2 つ目のリソース階層の Server 1 への拡張

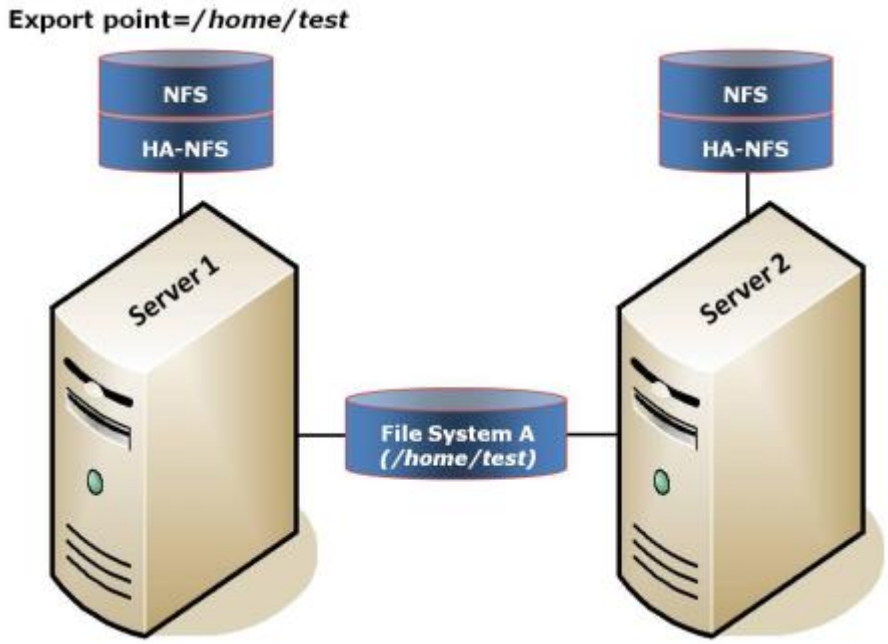
| | |
|---|---|
| Template Server: | *Server2* |
| Tag to Extend: | *nfs-/engdocs/app* |
| Target Server: | *Server1* |
| Target Priority: | 10 |

# 6.10.3.3.2. NFS v2/v3 でのアクティブ / スタンバイ

本セクションでは、NFS v2/v3 でのアクティブ / スタンバイ設定の例を示します。この設定では、NFS を実行し、ファイルシステムをエクスポートしている *Server 1* がアクティブと見なされます。 *Server 2* は別の処理を実行します。 *Server 1* に障害が発生した場合、 *Server 2* がファイルシステムへのアクセス権を獲得し、LifeKeeper のセカンダリ階層を使用して、クライアントがファイルシステムをアクセスできるようにします。

**注記:** アクティブ / スタンバイ設定では、サーバ 2 でも NFS が実行されていることがありますが、他の NFS リソースは LifeKeeper によって保護されていません。

## NFS v2/v3 でのアクティブ / スタンバイ設定例




### 設定に関する注記:

- NFS ソフトウェアを両方のサーバにインストールしておく必要があります。
- 基礎ファイルシステム (*File System A*) は共有 (または複製) ディクス上に置く必要があります。
- NFS エクスポートポイントは */home/test* です。
- エクスポートされたファイルシステムは、プライマリサーバとバックアップサーバの両方で同じマウントポイントにする必要があります。
- *Server 2* は *Server 1* がアクティブである間は、共有ディスク上のファイルとディレクトリにアクセスできません。

### Server 1 でのリソース階層の作成

| | |
|---|---|
| Server: | *Server1* |
| Export Point: | */home/test* |
| IP Tag: | ip-172.17.100.202 |
| NFS Tag: | *nfs-/home/test* |

### Server 2 へのリソース階層の拡張

| | |
|---|---|
| Template Server: | *Server1* |
| Tag to Extend: | *nfs-/home/test* |
| Target Server: | *Server2* |
| Target Priority: | 10 |

# 6.10.4. 設定作業

以下の設定作業は LifeKeeper GUI から実行できます。次の 4 つの作業は、NFS サーバリソースインスタンスに特有で、Recovery Kit ごとに異なるため、本セクションで説明しています。

**注記:** 本セクションの残りの部分では、LifeKeeper GUI の **[Edit]** ニューから作業を選択することによって、Recovery Kit を設定する方法を説明します。設定作業はツールバーから選択することもできます。状況表示ウィンドウの **[Resource Hierarchy Tree]** (左側のペイン)のグローバルリソースを右クリックして、 **[Edit]** メニューと同じドロップダウンメニューの選択項目を表示することもできます。言うまでもなく、これは階層がすでに存在している場合にだけ可能な方法です。

状況表示ウィンドウの **[Resource Hierarchy Table]** (右側のペイン)のリソースインスタンスを右クリックして、サーバおよび特定リソースの状況に応じて、リソース階層の作成を除くすべての設定作業を実行することもできます。

- リソース階層の作成 。アプリケーションリソース階層を LifeKeeper クラスタに作成します。
- リソース階層の削除 。リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。
- リソース階層の拡張 。リソース階層をプライマリサーバからバックアップサーバへ拡張します。
- リソース階層の拡張解除 。リソース階層を LifeKeeper クラスタ内の 1 つのサーバから拡張解除(削除)します。
- リソース階層のテスト 。手動でスイッチオーバを行って、NFS リソース階層をテストします。

以下の作業については、すべての Recovery Kit で同じ手順を使用する共通の作業であるため、 SPS for Linux テクニカルドキュメンテーション の 管理 セクションを参照してください。

- リソース依存関係の作成 。既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除 。リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service 。リソース階層を特定のサーバで In Service の状態にします。
- Out of Service 。リソース階層を特定のサーバで Out of Service の状態にします。
- プロパティの表示 / 編集 。特定のサーバでリソース階層のプロパティを表示または編集します。

# 6.10.4.1. NFS リソース階層の作成

プライマリサーバからリソースインスタンスを作成するには、以下の手順を実行します。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択します。

   すでに行った選択を変更する場合、または NFS リソース階層の作成中にエラーメッセージが表示された場合、 **[Back]** ボタンを使用して選択を変更するか、修正を行ってください ( **[Back]** ボタンが有効な場合)。

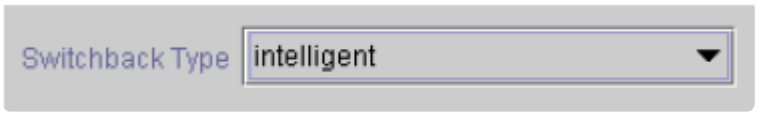
   ダイアログボックスが表示され、ドロップダウンメニューに、クラスタにインストールされ認識されているすべての Recovery Kit が一覧で示されます。ドロップダウンメニューから **[NFS]** を選択します。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

   注記 : 階層作成の途中で **Cancel** ボタンをクリックすると、作成処理全体が取り消されます。

2. **[Switchback Type]** を選択します。これによって、バックアップサーバへのフェイルオーバ後、NFS インスタンスがサービスに戻ったときにどのようにプライマリサーバにスイッチバックされるかが決まります。 **[intelligent]** または **[automatic]** を選択します。 **[Intelligent switchback]** の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。 **[Automatic switchback]** の場合、プライマリサーバがオンラインに戻り、LifeKeeper の保護下に入るとすぐに自動的にスイッチバックが行われます。

   スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

3. **[Server]** で NFS リソースを作成するサーバ (通常、これをプライマリサーバまたはテンプレートサーバと呼ぶ) を選択します。ドロップダウンメニューには、クラスタ内の全サーバが表示されます。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. **[Export Point]** ダイアログボックスのドロップダウンリストに、次の条件に合致する NFS ファイルシステムのエクスポートポイントが表示されます。

- ◦ エクスポートポイントが NFS によってエクスポートされている。
- ◦ エクスポートポイントが共有ドライブ上にある。
- ◦ 基礎ファイルシステムが LifeKeeper に保護されている場合、基礎ファイルシステムは In Service で、かつ、 **Server** ダイアログボックスで選択したサーバ上で最高の優先順位を持っている必要がある。
- ◦ NFSv4 の基準:
  - º バインドマウントを使用した NFSv4 のルートエクスポートのために、バインドマウントは通常のエクスポートと同様に共有ドライブ上にある必要があります。そして、ファイルシステムが LifeKeeper に保護されている場合は In service で、かつ、[Server] ダイアログボックスで選択したサーバ上で最も高い優先順位を持っている必要があります。
  - º

NFS v4 ルートエクスポートがすでに保護されている場合、選択肢はありません (v4 単体のみが保護対象で、v4 と v2/v3 の混在環境は保護対象にならない)。

- º すでに

NFS v2/v3 が保護されている場合は、NFS v4 は選択肢に表示されません。

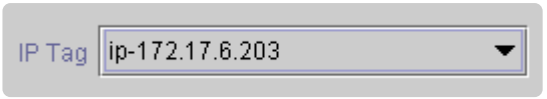
- º 何も保護されていない場合は、v2/v3 および v4 の両方が含まれたリストが表示されます。

ドロップダウンリストから保護する NFS エクスポートポイントを選択します。

**[Next]** をクリックして次のダイアログボックスに進んでください。

5. **[IP Tag]** ダイアログボックスのドロップダウンリストに、現在 LifeKeeper が保護していて、NFS リソースを作成するサーバで in service の仮想 IP アドレスに対応するタグが表示されます。

保護されている NFS ファイルシステムにアクセスするためにクライアントが使用する仮想 IP アドレスの **タグ** を選択します。




**注記**：この時点で、保護される IP リソースが利用可能かどうかが LifeKeeper によってチェックされます。また、NFS リソース階層の作成に使用する入力データが妥当かどうかもチェックされます。この 2 つのチェックのどちらかで問題が発生した場合 [ERROR] ボックスが表示されます。デ

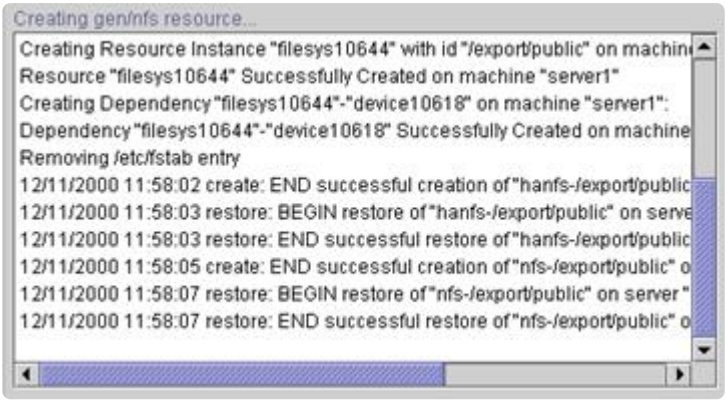
ィレクトリのパスは妥当で、NFS 構成自体にエラーがある場合は、そのエラーを修正して、階層の作成を続けることができます。ここで必要な LifeKeeper IP リソースを作成することもできます。

注記：仮想 IP アドレス依存関係を持つ別の LifeKeeper Recovery Kit を使用している場合は、NFS リソース用に別の仮想 IP アドレスを作成することもできます。そうしない場合、仮想 IP リソースがバックアップサーバにフェイルオーバされると、その IP リソースに依存するすべてのリソースが同時にフェイルオーバされます。

**[Next]** をクリックして次のダイアログボックスに進んでください。

6. **[NFS Tag]** を選択または入力します。これは NFS 階層に付けるタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。

**[Create]** ボタンをクリックすると、 ***[Create Resource Wizard]*** によって NFS リソースが作成されます。




**[Next]** をクリックすると階層の拡張 で説明する ***[Pre-Extend Wizard]*** が起動します。

注記：この時点で、NFS リソース階層が正常に作成されます。ただし、新しい NFS インスタンスの起動に失敗したというエラーメッセージが表示されることがあります。新しい NFS 階層を別のシステムに拡張するには、起動する(in service にする) 必要があることに注意してください。起動に失敗した場合は階層が削除されますが、この時点で表示されるエラーに基づいて問題を修正することも可能です。エラーの修正が不可能であれば、リソース作成のキャンセルのみを選択することになります。

階層の拡張 に進む前に、新しい階層を In Service にしてください。

注記: RHEL7.1 以降および SLES12SP1 以降の環境では、 NFS リソースを作成した後に nfs-server.service の自動起動を無効に設定してください。 また、NFS リソースは起動時に rpcbind.service が動作している必要があるため、必要に応じて rpcbind.service を自動起動するように設定してください。

# 6.10.4.2. リソース階層の削除

LifeKeeper 環境内のすべてのサーバからリソース階層を削除するには、以下の手順を実行します。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択します。ドロップダウンメニューから **[Delete Resource Hierarchy]** を選択します。

2. **[Target Server]** で、NFS リソース階層を削除するサーバの名前を選択します。

   **注記:** 次のどちらかの場所で右クリックして **[Delete Resource]** を選択した場合、このダイアログボックスは表示されません。

   - 個々のリソースインスタンスの右ペイン
   - リソースが 1 台のサーバだけにあるとき、グローバルリソースの左ペイン

   **[Next]** をクリックして次のダイアログボックスに進んでください。

3. **[Hierarchy to Delete]** を選択します。削除するリソース階層を特定して、強調表示にします

   **注記:** 左側ペインのグローバルリソースまたは右側ペインの個々のリソースインスタンスを右クリックしてリソースの削除作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。

**[Delete]** をクリックして、次のダイアログボックスに進みます。

5. NFS リソースの削除に成功したことを示す別の情報ボックスが表示されます。

6. **[Done]** をクリックして、 **[Delete Resource Hierarchy]** メニューの選択を終了します。

# 6.10.4.3. リソース階層の拡張

階層の作成後、クラスタ内の別のサーバに拡張する必要があります。リソースインスタンスをテンプレートサーバからターゲットサーバに拡張するには方法が 2 つあります。最初の方法は、リソース作成後、[Continue] をクリックして、別のサーバにリソースを拡張する方法です。もう 1 つの方法は、次に説明するように [Edit] メニューから [Extend Resource Hierarchy] を選択する方法です。どちらの方法でも、同じダイアログボックスが表示されます (一部例外があり、これについては次で説明)。

1. LifeKeeper GUI メニューから **[Extend Wizard]** に入っている場合は、 **[Edit]** を選択し、次に **[Resource]** を選択します。ドロップダウンメニューから **[Extend Resource Hierarchy]** を選択します。これで **[Extend Resource Hierarchy]** ウィザードが起動されます。

2. 最初に表示されるダイアログボックスで、NFS リソース階層が現在 In Service の **[Template Server]** を選択します。ここで選択するするテンプレートサーバと次のダイアログボックスで選択する拡張するタグによって、 In Service のリソース階層が表されることを認識しておくことが重要です。選択したテンプレートサーバで In Service でないリソースタグを選択すると、エラーメッセージが表示されます。このダイアログのドロップダウンボックスには、クラスタ内の全サーバの名前が示されています。

   **注記:** NFS リソース階層作成後、直ちに **[Extend Resource Hierarchy]** 作業に入った場合は、作成段階でテンプレートサーバが特定されているため、このダイアログボックスは表示されません。これは、GUI ウィンドウの左ペインの NFS リソースアイコンまたは右ペインの NFS リソースボックスを右クリックして **[Extend Resource Hierarchy]** を選択した場合も同様です。

   **注記:** 階層拡張の途中で **[Cancel]** ボタンをクリックすると、その特定のサーバへの拡張処理が取り消されます。また、取り消しポイントを超えて現在拡張されている階層 (IP アドレスまたはファイルシステム) の依存リソースもすべて拡張解除されます。ただし、すでに NFS リソース階層を別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

   例えば、Server 1 にリソースを作成し、そのリソースを Server 2 へ拡張したとします。同じリソースを Server 3 に拡張している最中に、拡張を取り消すためいずれかのダイアログボックス内の [Cancel] ボタンをクリックしたとします。この場合は、Server 3 へのリソース拡張だけがキャンセルされ、Server 2 への拡張はキャンセルされません。この階層から Server 2 を削除したい場合は、Server 2 からリソースを拡張解除する必要があります。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

3. **[Tag to Extend]** を選択します。これは、テンプレートサーバからターゲットサーバに拡張する NFS インスタンスの名前です。ウィザードのドロップダウンメニューには、前述のダイアログボックスで選択したテンプレートサーバ上に作成されたすべてのリソースが表示されます。

   **注記:** ここでも、NFS リソース階層作成後、ただちに **[Extend Resource Hierarchy]** 作業に入った場合は、作成段階で NFS リソースのタグ名が特定されているため、このダイアログボックスは表示

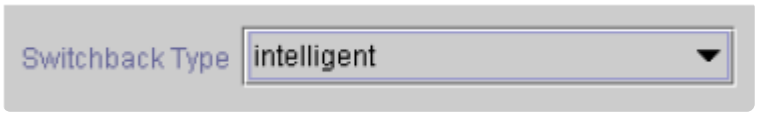
されません。これは、GUI ウィンドウの左ペインの NFS リソースアイコンまたは右ペインの NFS リソースボックスを右クリックして、 **[Extend Resource Hierarchy]** を選択した場合も同様です。

**[Next]** をクリックして次のダイアログボックスに進んでください。

4. NFS リソース階層の拡張先の **[Target Server]** を選択します。ドロップダウンボックスに、選択された階層にまだ入っていないクラスタ内のサーバ名の一覧が表示されます。

[Next] をクリックして、次のダイアログボックスに進んでください。

5. **[Switchback Type]** を選択します。これによって、バックアップサーバへのフェイルオーバ後、NFS インスタンスがサービスに戻ったときに、どのようにプライマリサーバにスイッチバックされるかが決まります。[intelligent] または [automatic] を選択できます。[intelligent] の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。[automatic] の場合、プライマリサーバがオンラインに戻り、LifeKeeper の保護下に入るとすぐに自動的にスイッチバックが行われます。




スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。

**[Next]** をクリックして次のダイアログボックスに進んでください。

6. **[Template Priority]** を選択または入力します。これはサーバで現在 In Service の NFS 階層の優先順位です。優先順位は、1 ～ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。 **注記:** このフィールドは階層を最初に拡張するときだけ表示されます。

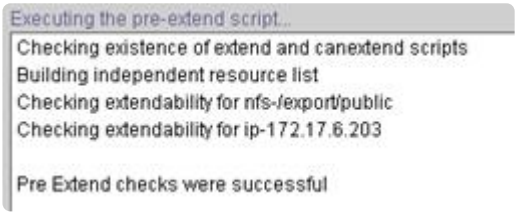
**[Next]** をクリックします。

7. **[Target Priority]** を選択または入力します。これは、別サーバにある同等の階層に対する新しく拡張する NFS 階層の優先順位です。1 ～ 999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位は高くなります (1 は最高の優先順位を表す)。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当て

ることはできません。

**[Next]** をクリックします。

8. 環境がチェックされ、この NFS リソースを拡張するための要件がすべて満たされていることを示す情報ボックスが表示されます。満たされていない要件がある場合、**[Next]** は選択できず、**[Back]** ボタンが有効になります。




**[Back]** をクリックした場合、情報ボックスに表示されるエラーメッセージの内容に従って、リソースの拡張を変更できます。

ここで **[Cancel]** をクリックした場合、NFS リソース階層を LifeKeeper で保護するには、いずれかの時点でここに戻り、別のサーバに NFS リソース階層を拡張する必要があります。

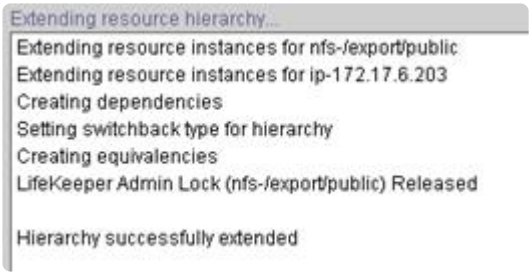
**[Next]** をクリックすると、**[Extend Resource Hierarchy]** 設定作業に入り **[NFS Tag]** ダイアログボックスが表示されます。

9. この画面にはテンプレートサーバ、拡張するタグ、ターゲットサーバ、標準の NFS タグに関する情報が表示されます。NFS タグは、NFS 階層の拡張に付けるタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。

**[Next]** をクリックして次のダイアログボックスに進んでください。

10. 拡張を実行中であることを確認する情報ボックスが表示されます。

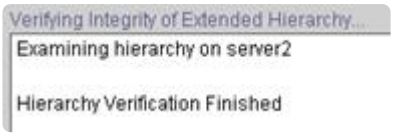
**注記:** IP リソースをターゲットサーバにまだ拡張していない場合、NFS リソースの拡張処理中に NFS Server Recovery Kit によって拡張されます。この場合、拡張の実行に関する情報ボックスが表示される前に、IP リソースの拡張に関連したダイアログボックスがいくつか表示されます。




同じ NFS リソースインスタンスをクラスタ内の別のサーバに拡張する場合は、 **[Next Server]** をクリックします。その場合は、リソース階層の拡張の操作が繰り返されます。

**[Finish]** をクリックした場合、NFS リソースの拡張処理が成功したか検証が行われます。

11. **[Finish]** をクリックした場合、次の画面が表示されます。




12. **[Done]** をクリックして終了します。

**注記:** 必ず両方のサーバで新しいインスタンスの機能をテストしてください。

# 6.10.4.4. リソース階層のテスト

NFS リソース階層をテストする前に、次に説明しているようにクライアントの設定を検証する必要があります。その後、手動でスイッチオーバを行って、NFS リソース階層のテストを行うことができます。これにより、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

## クライアントの設定の検証

一般に、クライアントは、NFS リソース階層の作成作業中に選択した LifeKeeper が保護する IP アドレスを使用してファイルシステムをマウントする必要があります。正しい IP アドレスを選択しているかクライアント側で確認することはありません。したがって、クライアントがファイルシステム用の正しい IP アドレスを確実に使用するように、慎重に次の検証手順を実行する必要があります。

クライアントの設定を検証するには、次の手順に従います。

1. セカンダリシステムで In Service の NFS インスタンスがないことを確認します。
2. LifeKeeper が保護する正しい IP アドレスを使用して、クライアントにファイルシステムをマウントします。
3. 手動でセカンダリシステムへのスイッチオーバを実行して、切り替えた NFS インスタンスをセカンダリシステムで現在 In Service の唯一の NFS インスタンスにします。
4. スイッチオーバが完了したら、クライアントがそのファイルシステムにアクセスできることを確認します。

## GUI による手動スイッチオーバの実行

依存関係を定義すると、自動的に LifeKeeper によってアプリケーションの起動と終了が制御され、障害が検出された場合にはフェイルオーバが開始されます。保守作業など、管理上の理由からスイッチオーバを手動で実行することもできます。

LifeKeeper GUI で、ドロップダウンメニューから [Edit] > [Resource] > [In Service] を選択すると、手動スイッチオーバを開始できます。例えば、バックアップサーバで In Service 要求を実行すると、アプリケーション階層はバックアップサーバで In Service になり、プライマリサーバでは Out of Service になります。

[Out of Service] 要求を実行すると、アプリケーションは他のサーバで In Service にならずに、Out of Service になります。

**注記:** NFS リソースを Out of Service にするには、NFS リソースと関連する HA-NFS リソースの両方を Out of Service にする必要があります。

アプリケーション側の動作については、すべての動作はアプリケーションのドキュメントで規定されています。LifeKeeper 側では、ロールバックやバックアップアーカイブのような内部処理の調整や制御は行い

ません。テープアーカイブやリストアの作業は、アプリケーション管理者の責任です。

## リカバリ動作

プライマリサーバに障害が発生すると、NFS Server Recovery Kit は次の作業を実行します。

- NFS デーモンが実行されていない場合、それらを開始します。
- NFS ファイルシステムをエクスポートします。

# 6.10.4.5. リソース階層の拡張解除

リソース階層の拡張解除を行うには、次の手順を実行します。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択します。ドロップダウンメニューから **[Unextend Resource Hierarchy]** を選択します。

2. **[Target Server]** で、NFS リソースを拡張解除するサーバを選択します。NFS リソースが現在 In Service のサーバは選択できません。

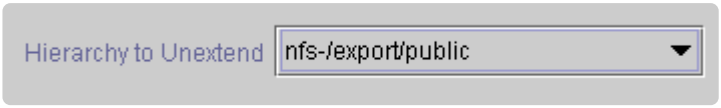
   **注記:** 右側のペインから個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

3. **[Hierarchy to Unextend]** で、拡張解除する NFS 階層を選択します。

   **注記:** 左側のペインからグローバルリソースを右クリックするか、右側のペインから個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

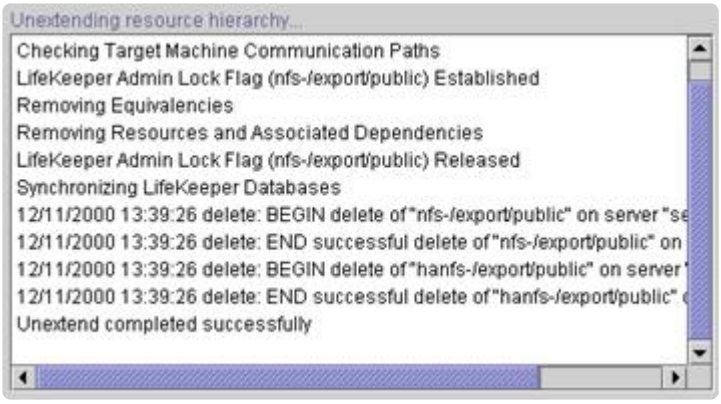
   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. 拡張解除するために選択したターゲットサーバと NFS リソース階層を確認する情報ボックスが表示されます。

   **[Unextend]** をクリックします。

5. NFS リソースの拡張解除に成功したことを示す別の情報ボックスが表示されます。




6. **[Done]** をクリックして、 **[Unextend Resource Hierarchy]** メニューを終了し、LifeKeeper GUI に戻ります。

   階層が拡張解除されて 1 つのサーバだけになった場合、「**One or More Resources Unprotected**」という警告メッセージが表示されます。

# 6.10.5. トラブルシューティング

本セクションでは、LifeKeeper NFS リソース階層の作成時や拡張時、またはリソースの削除時やリストア時に表示される可能性のあるメッセージの一覧を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

他の SPS コンポーネントからメッセージが出されることもあります。そのような場合は、メッセージカタログ を参照してください。メッセージカタログには、操作、管理、GUI など、SIOS Protection Suite for Linux を使用しているときに遭遇する可能性がある、すべてのエラーコードが列挙されています。また、エラーコードの原因に関する追加の説明や、問題解決のために必要な処置についても、必要に応じて記載されています。この完全なリストを検索すると、受信したエラーコードを見つけることができます。また、関連する SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

下記トラブルシューティングのヘルプをクリックしてください。

- HA nfs-utils のインストールと設定
- NFS 階層作成時のエラー
- NFS 階層拡張時のエラー
- 階層リソースのリストア、削除、リカバリ時のメッセージとエラー
- 階層削除のメッセージとエラー

# 6.10.5.1. HA nfs-utils のインストールと設定

LifeKeeper NFS Server Recovery Kit のバージョン 5.0.0 以降では、高可用性の有効な nfs-utils パッケージがインストールおよび設定されている必要があります。(いくつかの OS のディストリビューションバージョンでは nfs-utils が別のパッケージで提供されます。)LifeKeeper NFS Server Recovery Kit のインストールでは、HA が有効な nfs-utils パッケージの存在が検査されます。正しく設定された nfs-utils パッケージを検出できなかった場合は、LifeKeeper インストールサポートのセットアップスクリプトを再実行するか、サーバをリブートする必要があります。

LifeKeeper インストールサポートのセットアップスクリプトによって、 `nfs-utils` が HA 環境で使用できるように設定され、 **NFS v4** エクスポート設定のセットアップが実行されます。使用するどの NFS バージョンでも HA NFS の質問には **YES** と回答する必要があります。

**NFS v4** の設定において、 *rpc_pipefs* を */var/lib/nfs* から */var/lib* へ移動させる必要あります。この設定を行うためには、カーネルモジュールのアンロードおよび設定とブート時のスクリプトの追加もしくは修正が必要とされます。変更後、インストールサポートがカーネルモジュールのアンロードおよびリロードを行うことができない場合はシステムのリブートが必要になる可能性があります。この事象が発生した場合、システムのリブートが必要であることがユーザに通知されます。LifeKeeper の操作を正常に行うためにはシステムのリブートを含めた `rpc_pipefs` のセットアップを完了させることが必要です。

# 6.10.5.2. 階層削除のメッセージとエラー

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 106015 | Unable to restore the entry for export point "EXPORT POINT" in /etc/exports on server “SERVER" | エントリを手動でリストアします。 |
| 106021 | An entry for export point "EXPORT POINT" already exists in /etc/exports. The entry that was being used by the NFS Server Recovery Kit has been placed in the file "FILENAME" | /etc/exports のエクスポートエントリが正しいか確認します。 |
| 106049 | Restore statedir from /var/lib/.nfs.LK to /var/lib/nfs failed on server SERVER. | NFS ステートディレクトリ */var/lib/nfs* のリストアに失敗しました。 */var/lib/.nfs.LK* を */var/lib/nfs* へ移動させ、手動でディレクトリをリストアしてください。 |

# 6.10.5.3. 階層のリストア、削除、リカバリ時のメッセージ

## NFS リソースを In Service にする (リストア)

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 106007 | Cannot bring NFS or HANFS resource "TAG" In Service on server "SERVER" | 他のエラーメッセージを参考にして、実行するべき処置を決定します。問題を修正した後、手動でリソースを In Service にしてください。 |
| 106010 | NFS is not running on server "SERVER". LifeKeeper will attempt to restart NFS. | このダイアログボックスは情報の表示専用です。LifeKeeper は自動的に NFS デーモンの再起動を試みます。デーモンの再起動時に LifeKeeper に問題が発生した場合は、NFS の起動に失敗したというメッセージが表示されます。 |
| 106011 | Starting NFS on server “SERVER" failed | NFS デーモンの再起動中に LifeKeeper に問題が発生しました。手動で NFS の再起動をしてください。 |
| 106012 | The export point "EXPORT POINT" is not exported on server "SERVER". LifeKeeper will attempt to export the entry. | LifeKeeper によって、エクスポートポイントがエクスポートされていないことが検出され、エクスポートが試みられます。 |
| 106013 | Unable to export "EXPORT POINT" on server "SERVER" | ファイルシステムを手動でエクスポートします。 |
| 106014 | Usage: USAGE STRING | 誤った引数で実行されたコマンドの使用。コマンドラインのみ。 |
| 106019 | Executing command: "COMMAND" | このメッセージは、LifeKeeper が NFS デーモンを再起動するとき、またはエクスポートポイントをエクスポート、アンエクスポートするときに表示されます。問題が発生した場合、この情報が役に立ちます。 |
| 106024 | Unable to stop and restart rpc.mountd on "SERVER!" | 階層リストア中に rpc.mountd デーモンプロセスを再開する必要がありましたが、このプロセスが失敗しました。プロセスを手動で停止および再開して、エラーとその処置を判別してください。 |
| 106027 | Open of "ABC" on server "SERVER" failed: "File not found" | 記述された理由により、ファイルを開けませんでした。 |
| 106028 | Mount of /proc/fs/nfsd failed on server "SERVER!" | 2.6 以降のカーネルでは、クライアント認証に /proc/fs/nfsd が使用されますが、そのマウント試行が失敗しました。手動で /proc/fs/nfsd をマウントし、失敗内容を判別してください。 |
| 106029 | Unable to get exclusive lock on "/var/lib/nfs/rmtab" on server | ファイルの問題が示された 20 秒後に更新するための、排他ロックを /var/lib/nfs/rmtab 上に配置できません。 |

| | | |
|---|---|---|
| | "SERVER" | |
| 106030 | Unable to restore client info for "123.45.678.90" on server "SERVER!": "ABC" | クライアントでのフェイルオーバの試みは、リストされた理由によってサーバへのロックに失敗しています。 |
| 106037 | Attempts to get exclusive lock on "/var/lib/nfs/rmtab" on server "SEVER" failed: "ABC" | 更新のために /var/lib/nfs/rmtab に排他ロックを配置できません。原因については、エラーメッセージを参照してください。 |
| 106039 | Open of "FILE" on server "SERVER" failed: “ABC” or Attempt to get exclusive lock on "FILE" on server "SERVER" failed: "ABC | ファイルを開くかファイルの排他ロックを取得しようとしましたが、失敗しました。原因については、エラーメッセージを参照してください。 |
| 106040 | Multiple virtual IP addresses detected. In this release NFS lock failover only supports one virtual IP address. | NFS リソース階層を作成し直して、1 つの仮想 IP アドレスのみを使用するか、または LifeKeeper デフォルトファイルで FAILOVERNFSLOCKS を false に設定してください。 |
| 106052 | Unable to mount rpc_pipefs on "SERVER". Reason: "REASON". | REASON により rpc_pipefs は、SERVER でマウントされませんでした。 |
| 106053 | *rpc_pipefs* successfully mounted on "SERVER" | SERVER で *rpc_pipefs* が正常にマウントされました。 |

## NFS リソースを Out of Service にする(削除)

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 106008 | Unable to unexport the export point "EXPORT POINT" on server "SERVER" | **exportfs(8)** コマンドを使用して、エクスポートを解除します。 |
| 106014 | Usage: USAGE STRING | 誤った引数で実行されたコマンドの使用。コマンドラインのみ。 |
| 106019 | Executing command: "COMMAND" | このメッセージは、LifeKeeper が NFS デーモンを再起動するとき、またはエクスポートポイントをエクスポート、アンエクスポートするときに表示されます。問題が発生した場合、この情報が役に立ちます。 |

## NFS リソースを In Service に戻す (ローカルリカバリ)

LifeKeeper Core は、NFS の「quickCheck」スクリプトを実行して、ローカルサーバで In Service のすべての NFS インスタンスの状態を定期的にチェックします。このスクリプトは次の事項を検証します。

- ファイルシステムがエクスポートされているか

- NFS/HA-NFS デーモンが実行中かどうか

インスタンスが完全に動作していない場合、「recover」スクリプトが実行され、インスタンスの再起動が試みられます。これは、エラーメッセージを記録し、「restore」を起動して、(「restore」スクリプトの結果に応じて) 最後のエラーメッセージまたは成功メッセージを出力し、「restore」と同じ結果を返します。ローカルリカバリに失敗した場合、このインスタンスは別のサーバにフェイルオーバされます。

# 6.10.5.4. NFS 階層拡張時のエラー

NFS 階層の拡張時に表示されるエラーメッセージを、説明とともに挙げます。これらのエラーメッセージは、GUI が「Executing the pre-extend script....」というメッセージを表示して、新しいシステムに階層を拡張する前に階層の検証を行っているときに表示されることに注意してください。

ターゲットサーバへの **NFS** リソース階層の作成時

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 106016 | "REQUIRED SOFTWARE" cannot be found or does not have the expected permissions on server "SERVER" | NFS は、プライマリサーバとすべてのバックアップサーバにインストールする必要があります。nfs-utils がインストールされていることを確認してください。 |
| 106017 | The file system "FILE SYSTEM" on template server "SERVER" has a different mount point "MOUNT POINT" on server "SERVER" | リソースは各サーバに同じマウントポイントで作成する必要があります。ターゲットサーバからファイルシステム階層を拡張解除するか、同じマウントポイントでテンプレートサーバとターゲットサーバに再作成します。 |
| 106018 | Unable to copy the file "FILENAME" from server "SERVER" to server "SERVER" | 考えられる原因:<br>• サーバ間のコミュニケーションパスがダウンしています。<br>• ターゲットサーバのファイルシステムに空き容量がありません。<br>• ファイルシステムの問題。steeleye-lkNFS パッケージを再インストールする必要があります |
| 106020 | The generated id "nfs-/export" conflicts with an existing resource id | nfs または hanfs リソース用に内部で生成されたリソース ID は、重複しています。 |
| 106022 | The export point “EXPORT POINT" is in /etc/exportson the target server "SERVER" | ターゲットサーバの */etc/exports* ファイルからエクスポートポイントを削除してからリソースを拡張します。 |
| 106023 | The export point "EXPORT POINT" is exported on the target server "SERVER" | リソース拡張前にターゲットサーバのエクスポートポイントをエクスポート解除してください。 |
| 106051 | Unable to create active/active configurations with NFS v4 exports. Either "TEMPLATE SERVER" or "TARGET SERVER" currently protects an NFS v4 root export. | すでに片方もしくは両方のサーバが NFS v4 エクスポートを保護しており、NFS v4 エクスポートでのアクティブ/アクティブをサポートしていないため TEMPLATE SERVER から TARGET SERVER へ NFS リソースを拡張することができません。 |

# 6.10.5.5. NFS 階層作成時のエラー

NFS 階層の作成時に表示されるエラーメッセージを、説明とともに挙げます。リストするメッセージは、nfs リソースと hanfs リソースの作成時に表示されるメッセージです。LifeKeeper Core および他の Recovery Kit が表示するエラーメッセージはこのガイドには記載していません。ここに記載されている問題を解決するために、LifeKeeperを停止することがあるかもしれません。そして中断した時点から階層の作成を続行する際に、任意の新しいリソースを作成するといったことが必要な場合があります。NFS 設定に必要な新しい LifeKeeper リソースの作成も可能です。

**注記:** 次のエラーメッセージの説明で、コマンドラインのみというのは、コマンドラインでコマンドを入力した場合にだけそのメッセージが表示されることを示しています。LifeKeeper GUI を使用している場合、そのメッセージは表示されません。さらに、階層作成の終わりに、リソースのリストアが開始されます。そのプロセス中に発生する可能性のあるメッセージとエラーについては、「リソース階層のリストア、停止、ローカルリカバリ時のメッセージとエラー」を参照してください。

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 106000 | Export point not specified | リソース階層を作成するときには、NFS ファイルシステムのエクスポートポイントを指定する必要があります。 *コマンドラインのみ* |
| 106001 | The path "EXPORT POINT" is not exported by NFS | 指定したエクスポートポイントが現在 NFS によってエクスポートされていません。 **exportfs(8)** を使用してパスをエクスポートし、パスが */var/lib/nfs/etab* ファイルに含まれていることを確認します。 *コマンドラインのみ* |
| 106002 | create: The export point “EXPORTPOINT” on “server1” for client “CLIENT” either does not contain an FSID export option or the value is not unique. A unique FSID will be generated and "EXPORTPOINT" will be re-exported using the new FSID value. | LifeKeeper 保護下のすべてのエクスポートポイントでは、高可用性の NFS に固有の fsid=エクスポートオプションを使用する必要があります。選択したエクスポートは、この要件を満たしていなかったため、固有の値が生成されてから、選択したエクスポートポイントの再エクスポートが行われました。注記:「*」または「world」のクライアントは、エクスポートポイントをすべてのクライアントが使用できることを示します。 |
| 106003 | Unable to create the export entry file for "EXPORT POINT" in LifeKeeper | 考えられる原因:<br>• <br>◦ ターゲットサーバのファイルシステムに空き容量がありません。<br>◦ ファイルシステムの問題。Steeleye-lkNFS パッケージを再インストールする必要があります。 |
| 106004 | The export point "EXPORT POINT" is not on a shared file | エクスポートポイントが共有ファイルシステムのものか確認します。 *コマンドラインのみ* |

| | | |
|---|---|---|
| | system on server "SERVER" | |
| 106005 | Unable to create the HA-NFS hierarchy "TAG" with child resource "TAG" on server "SERVER" | 他のエラーメッセージを参考にして、実行するべき処置を決定します。 |
| 106006 | Unable to remove entry for export point "EXPORT POINT" from /etc/exportson server "SERVER" | */etc/exports* ファイルが存在しているか、読み込み可能かを確認します。 |
| 106014 | Usage: USAGE STRING | 誤った引数で実行されたコマンドの使用。コマンドラインのみ |
| 106016 | "REQUIRED SOFTWARE" cannot be found or does not have the expected permissions on server "SERVER" | NFS は、プライマリサーバとすべてのバックアップサーバにインストールする必要があります。 **nfs-utils** がインストールされていることを確認してください。 |
| 106019 | Executing command: "COMMAND" | このメッセージは、LifeKeeper が NFS デーモンを再起動するとき、またはエクスポートポイントをエクスポート、アンエクスポートするときに表示されます。問題が発生した場合、この情報が役に立ちます。 |
| 106020 | The generated id "nfs-/export" conflicts with an existing resource id | nfs または hanfs リソース用に内部で生成されたリソース ID は、重複しています。 |
| 106025 | An unknown error has occurred while running "newtag" on server "Server1" | hanfs リソースの nfs のタグを生成するためにコマンド newtag を実行しているときに、予想外のエラーが発生しました。 |
| 106026 | Adding dependency between HA-NFS resource "hanfs-/export" and filesysresource "export" on server "server1" failed. | 選択した hanfs リソースと filesys リソース間の依存関係の作成が、不明な理由で失敗しました。詳細については、出力を参照してください。 |
| 106027 | Open of "ABC" on server "SERVER" failed: "File not found" | 記述された理由により、ファイルを開けませんでした。 |
| 106029 | Unable to get exclusive lock on "/var/lib/nfs/rmtab" on server "SERVER" | ファイルの問題が示された 20 秒後に更新するための、排他ロックを /var/lib/nfs/rmtab 上に配置できません。 |
| 106031 | Re-export of "/export" to add FSID option failed on server "SERVER" | エクスポートポイント「/export」には fsid 引数が含まれておらず、引数を生成した後に再エクスポートしようとしましたが失敗しました。fsid 引数を手動で /etc/exports エントリに追加し、再エクスポートして、失敗内容を判別してください。 |
| 106032 | Dependent IP resource tag name not specified | NFS リソース階層を作成するときには、保護する IP アドレスのリソースタグを指定する必要があります。コマンドラインのみ |

| | | |
|---|---|---|
| 106033 | Selected IP resource "TAG" does not exist on server “SERVER" | NFS リソースを作成する前に、指定のサーバに IP リソースを作成する必要があります。また、コマンドを入力するときに、IP リソースを正しく入力したか確認します。コマンドラインのみ |
| 106034 | Adding dependency between NFS resource "TAG" and IP resource "TAG" on server "SERVER" failed | NFS リソースを作成しているサーバで IP リソースが In Service か確認します。コマンドラインのみ |
| 106035 | Creation of HA-NFS resource "TAG" on server "SERVER" failed | 他のエラーメッセージを参考にして、実行するべき処置を決定します。 |
| 106036 | Adding dependency between IP resource "TAG" and HA-NFS resource "TAG" on server "SERVER" failed | HA-NFS リソースを作成するサーバで IP リソースが In Service か確認します。コマンドラインのみ |
| 106037 | Attempts to get exclusive lock on "/var/lib/nfs/rmtab" on server "SEVER" failed: "ABC" | 更新のために/var/lib/nfs/rmtab に排他ロックを配置できません。原因については、エラーメッセージを参照してください。 |
| 106038 | Unable to create directory "ABC" on server "SERVER": "ABC" | エクスポートされたファイルシステムでディレクトリを作成しようとしましたが、失敗しました。原因については、エラーメッセージを参照してください。 |
| 106039 | Open of "FILE" on server "SERVER" failed: “ABC” or Attempt to get exclusive lock on "FILE" on server "SERVER" failed: "ABC | ファイルを開くかファイルの排他ロックを取得しようとしましたが、失敗しました。原因については、エラーメッセージを参照してください。 |
| 106041 | The selected IP resource "ip-123.45.678.90" is not ISP on server "SERVER" | 選択した IP リソースはサーバに存在していますが、現在 In Service ではありません。サーバで IP リソースを In Service の状態にして、作成を再試行してください。コマンドラインのみ |
| 106048 | Multiple NFS v4 root exports found on "SERVER". | 1 つの NFS v4 擬似ファイルシステムのみサポートされている SERVER で、複数の NFS v4 擬似ファイルシステムが見つかりました。 |
| 106050 | Unable to protect more than one NFS v4 export or a combination of NFS v4 and NFS v3 exports. | NFS v4 と NFS v2/v3 エクスポートの組み合わせの保護はサポートされません。 |

# 6.11. Oracle Recovery Kit 管理ガイド

## Oracle Recovery Kit ドキュメンテーション

SIOS Protection Suite for Linux Oracle® Recovery Kit は、 SPS 環境で Oracle ソフトウェアの障害回復能力を高めます。Recovery Kit ソフトウェアは、Oracle データベースのデータ整合性と LifeKeeper SIOS Protection Suite によって実現される可用性の向上を結び付けるメカニズムを提供します。

## ドキュメントの内容

このガイドでは次の事項について説明します。

SPS for Linux テクニカルドキュメンテーション (LifeKeeper GUI 内の [Help] メニューからも参照できます) 。 SPS for Linux のすべてのドキュメントおよび情報が提供されている場所のリスト。

要件 。リカバリソフトウェアをインストールおよびセットアップするには、サーバが一定のハードウェア／ソフトウェア要件を満たしている必要があります。LifeKeeper Oracle Recovery Kit のインストールまたは削除の詳しい手順については、SIOS Protection Suite Installation Guide を参照してください。

Recovery Kit の設定 。 必要とする保護と柔軟性が得られるように SPS を設定するには、設定ルールを把握している必要があります。設定を適切にプランニングするには、ネットワーク設定、インターフェースの選択、ユーザシステムのセットアップ、階層オプション、Oracle 設定作業についての理解が必要です。この「要件」セクションでは、設定のプランニングだけでなく、設定の例を示し、Recovery Kit を設定するために必要な特定のタスクについて説明します。

トラブルシューティング このセクションでは、情報メッセージとエラーメッセージ、および推奨される対処方法の一覧を示します。

# 6.11.1. 要件

LifeKeeper Oracle Recovery Kit のインストール / アンインストールを試みる前に、ハードウェア / ソフトウェア要件とインストール / アンインストールの手順を理解しておく必要があります。

## キットのハードウェア / ソフトウェア要件

設定が以下の要件を満たしていることを確認してください。

**サーバ** 。この Recovery Kit には、SPS テクニカルドキュメンテーション および SPS リリースノート で説明されている LifeKeeper の要件に従って設定した 2 台以上のサポートされているコンピュータが必要です。

**共有ストレージ** 。Oracle データベースは、SPS 環境の共有ディスク上に置く必要があります。共有ストレージのアーキテクチャにより、クラスタの各ノードに適切な LifeKeeper 共有ストレージまたはマルチパスストレージキットをインストールする必要があります。NFS バックデータベースストレージの例では、LifeKeeper NAS Kit のインストールが必要です。

**LifeKeeper ソフトウェア** 。クラスタの各サーバにインストールする LifeKeeper ソフトウェアのバージョン、および適用する LifeKeeper ソフトウェアパッチのバージョンが同一であることが絶対条件です。LifeKeeper の特定の要件については、SPS リリースノート および SPS テクニカルドキュメンテーション を参照してください。

**LifeKeeper IP Recovery Kit** 。リモートクライアントが Oracle データベースにアクセスする場合に、この Recovery Kit が必要になります。同じバージョンの Recovery Kit を各サーバにインストールする必要があります。

**IP ネットワークインターフェース** 。各サーバは、イーサーネット TCP/IP をサポートするネットワークインターフェースを 1 つ以上必要とします。IP 切り替えが正しく動作するには、ローカルネットワークに接続されているユーザシステムが標準の TCP/IP 仕様に準拠している必要があります。

**注記:** 各サーバが必要とするネットワークインターフェースが 1 つだけであっても、異種メディア要件、スループット要件、単一障害点の排除、ネットワークのセグメンテーションといった多くの理由のために、複数のインターフェースを使用すべきです。

**TCP/IP ソフトウェア** 。各サーバは TCP/IP ソフトウェアも必要とします。

**Oracle ソフトウェア** 。Oracle ソフトウェアが各サーバにインストールおよび設定されていなければ、LifeKeeper と LifeKeeper Oracle Recovery Kit の設定はできません。各サーバには同じバージョンをインストールする必要があります。最新のリリースの互換性および購入方法については、SPS リリースノート を参照するか、営業担当者にお問い合わせください。

LifeKeeper Oracle Recovery Kit をインストールまたは削除する具体的な手順については、SPS インストールガイド を参照してください。

# 6.11.2. LifeKeeper での Oracle の設定

本セクションでは、Oracle の設定を開始する前に検討すべき情報と、一般的な Oracle の設定例を示します。

以下の作業は、すべての Recovery Kit を通じて同一の手順からなる共通の作業のため、SPS for Linux テクニカルドキュメンテーション の 管理 セクションで説明しています。

- リソース依存関係の作成 。既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除 。リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service 。リソース階層を特定のサーバで in service にします。
- Out of Service 。リソース階層を特定のサーバで out of service にします。
- プロパティの表示 / 編集 。特定のサーバでリソース階層のプロパティを表示または編集します。

2.6 カーネルを使用した Linux ディストリビューションへの Oracle のインストール手順については、Linux ディストリビューションの Web サイトを参照してください。

LifeKeeper Core のリソース階層の設定に関する手順については、SIOS Technology の Web サイトにある SPS for Linux テクニカルドキュメンテーション も参照してください。

# 6.11.2.1. Oracle 特有の設定上の考慮事項

注記 : Raw I/O で Oracle を使用する場合は、あらかじめ Raw I/O デバイスを適切にセットアップしておく必要があります。手順については、付録 を参照してください。(Raw I/O は LifeKeeper Single Server Protection のオプションではありません。)

Oracle Recovery Kit を設定する前に以下の準備作業を完了し、必ず各サーバから Oracle が使用するファイルシステムとディスクパーティションをアクセスできるようにしてください。

1. **個人用初期化ファイルのプロンプトの削除** 。Oracle Recovery Kit が正しく動作するためには、Oracle ユーザの個人用初期化ファイル (つまり、.profile、.bash_profile) のプロンプトをすべて削除 (またはコメントとして除外) する必要があります。このファイルは Oracle ユーザが使用するシェル専用です。対話型にはできません。

注記 :「stty」ステートメントを個人用初期化ファイルで使用する場合は、対話型ターミナルが使用されているかどうか確認する「if」ステートメントに入れる必要があります。

2. **カーネルパラメータの設定** 。linux カーネルパラメータ（共有メモリやその他のカーネルリソースなど）の設定方法については、Oracle ドキュメンテーションを参照してください。これらのパラメータの設定方法の例を次に示します。

クラスタ内の *各サーバ* で次のようにします。

a. LifeKeeper を設定する前に、 */etc/sysctl.conf* で次のように ipcs の限度を設定します。

```
# changes for Oracle
kernel.shmmax = <value>
kernel.shmmni = <value>
kernel.shmall = <value>
kernel.sem = <value>
```

b. `sysctl -p` を実行して、上記の変更をカーネルに反映します。

c. 一部のディストリビューションでは、システム初期化ファイル (boot.local または rc.local) に `sysctl -p` を追加して、再起動のたびに変更をカーネルに反映させる必要があります。

3. **$ORACLE_HOME ディレクトリ** 。 *$ORACLE_HOME* ディレクトリおよび関連ファイルをローカルディスク上に設定するときは、 *$ORACLE_HOME* ディレクトリと各ファイルがすべてのサーバで同じになるように設定します。Linux の標準ユーティリティを使用して、ディレクトリとファイルを作成し、各サーバにコピーしてください。

注記 : 一部のアクティブ / アクティブ設定では、 *$ORACLE_HOME* の場所が異なります。

4. **場所** 。 *$ORACLE_HOME* ディレクトリは共有ディスクまたは非共有ディスク上に置くことができます。ディレクトリを共有メディアに置く利点は、 *$ORACLE_HOME* にも同じ共有ディスクが使用されていれば (例えば、アクティブ / スタンバイ設定など)、パラメータファイル*Oracle_HOME/dbs/<initSID.ora* や *spfileSID.ora>* の設定が 1 回で済むことです。共有ディレクトリの

不利な点は、一度に 1 台のサーバしかファイルシステムへの直接アクセスができないことです。SCSI リザベーションは、LifeKeeper によって保護された共有ドライブへのアクセスを一度に 1 台のサーバにしか許可しません。2 つ以上の Oracle インスタンス (SID) がクラスタ内で個々に保護されるアクティブ / アクティブ設定のクラスタを作成する場合は、 *$ORACLE_HOME* をローカルの非共有ストレージ上に設定する必要があります。

5. **ユーザおよびグループ ID** 。Oracle ユーザ (oracle) とグループ (dba) をすべてのサーバに作成する必要があります。ユーザ ID とグループ ID の番号は、すべてのサーバで同じになるようにしてください。

6. **データベース、アーカイブファイル、ログファイル、制御ファイル** 。すべてのデータベース、アーカイブファイル、ログファイル、制御ファイルを共有ファイルシステムまたはディスクパーティションに作成する必要があります。これらのロケーションは、Oracle パラメータファイル *init<SID>.ora* または *spfileSID.ora* に設定されます。データベースパラメータの編集については、Oracle ドキュメンテーションを参照してください。このパス名は全サーバで同じになるようにしてください。Oracle は内部的にこの情報を制御ファイルに保持しているため、Oracle が稼動中でなければ、SYSTEMS データベースのスペースとパスは変更できません。

**注記** : Oracle ログのアーカイブは、デフォルトでは有効になっていません。LifeKeeper Oracle 階層の作成前に有効になっている場合は、LifeKeeper がアーカイブファイルの場所を検知し、必要に応じて別のファイルシステム階層を作成します。しかし、ログアーカイブが LifeKeeper Oracle 階層作成後に有効になる場合は、共有アーカイブロケーションを保護するために手動でファイルシステム階層を作成および拡張し、Oracle リソースからこの新しいファイルシステム階層へ依存関係を作成してください。

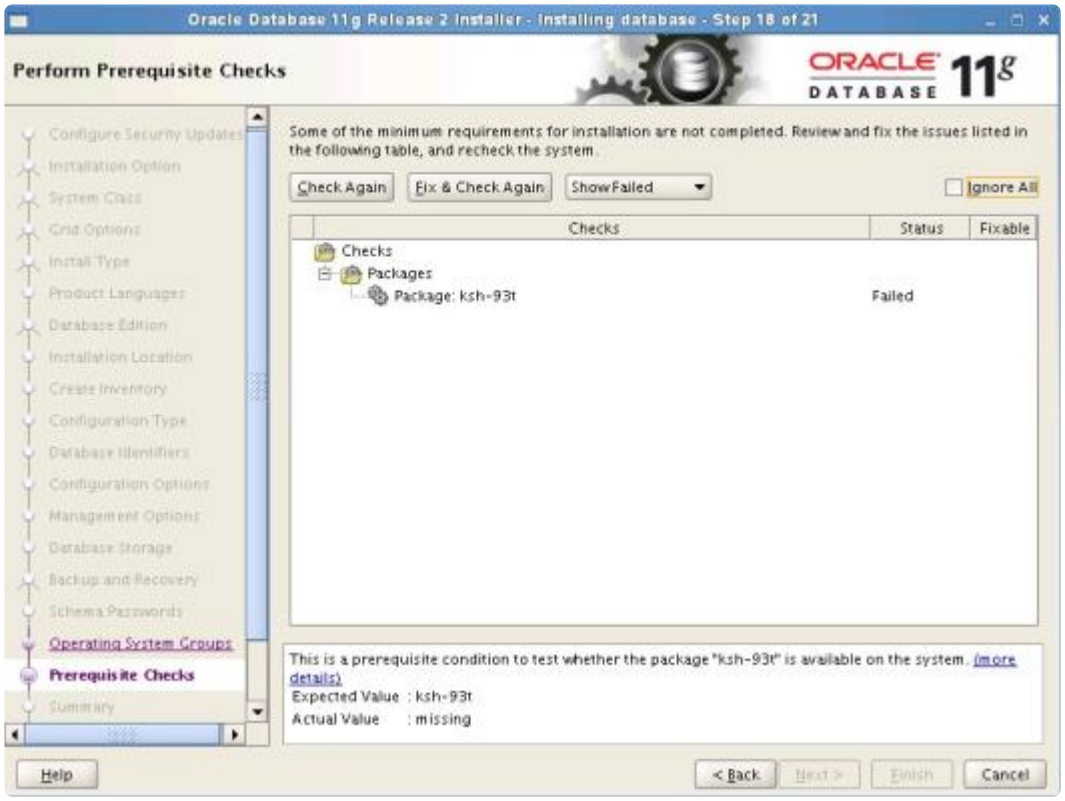
**注記** : ロック付きのストレージアプリケーションを使用し、NFS マウントオプションへの推奨設定を行う場合には、 *rw,nolock,bg,hard,nointr,tcp,nfsvers=3,timeo=600,rsize=32768,wsize=32768,actimeo=0* など追加の nolock オプションを SPS で設定する必要があります。

7. Oracle の新規インストール時にデータベースインストールプログラムでデータベースを作成しない場合、データベースインスタンスの最後の設定が容易になります。インストーラによってデータベースの作成が求められた場合は、 **[No]** を選択してください。インストールが完了したら、 **Oracle Database Creation Assistant** (dbca) を起動してください。dbca は、データベースコンポーネント作成時のインストール先の制御がより優れています。dbca の起動時に、Flash Recovery Area が LifeKeeper で保護するストレージ上に作成されるように指定してください 。

**重要:** Flash Recovery Destination は、共有ドライブ上に配置してください。

runInstaller でデータベースの作成を行った場合は、手動で Flash Recovery Area を再配置する必要があります。( **注記** : `runInstaller` でデータベースを作成できるようにはしないでください。)

8. `runInstaller` ユーティリティを使用した Oracle のインストール中、Oracle データベースのインストールに進む前にインストーラが Linux のパッケージおよび設定を確認するポイントがあります。LifeKeeper 7.2 (もしくは、それ以降のバージョン) がインストールされている場合に、ksh パッケージが見つからないというメッセージが表示されます。




このメッセージが表示された場合、画面右上の **[Ignore All]** にチェックを入れてください。LifeKeeper のインストールにより ksh パッケージが削除され、Public Domain Korn Shell の `pdksh` に置き換えられます。Oracle のインストールは `pdksh` を使用してください。

注記：バージョン 8.0 以降、Oracle ARK では `pdksh` は必要ありません。ただし、LifeKeeper Core ではまだ `pdksh` は必要であるため、 **[Ignore All]** にチェックを入れる必要があります。

注記：バージョン 8.1 以降の LifeKeeper では、独自の `pdksh` パッケージが提供されているため、Oracle の ksh の要件と競合しなくなりました。

9. データベースエンジンの調整。データの整合性とパフォーマンスのためにデータベースエンジンを調整するときのガイドラインについては、Oracle のマニュアルを参照してください。特に、障害回復能力の向上のためにアプリケーションを最適化するときは、メモリキャッシングとチェックポイントの頻度の調整が重要です。チェックポイントの間隔はコミットされていないデータベーストランザクションの数を判定します。その結果、システム障害の発生時に失われるデータベーストランザクションの数が決まります。

10. oratab ファイルのデータベースエントリ。/etc/oratab ファイルにはデータベースに対応したエントリが必要です。LifeKeeper 設定ルーチンは、このファイルの内容を使用して、 *$ORACLE_HOME* と *$ORACLE_SID* の値を関係付けます。通常は、Oracle インストールプログラムが必要なエントリを作成します。ただし、Oracle ソフトウェアが共有ファイルシステムにインストールされた設定では、管理者が oratab ファイルを、Oracle のインストールが実行されたサーバから、他の各サーバの /etc ディレクトリにコピーして、すべてのサーバで利用可能にする必要があります。

注記：サーバごとに設定可能な oratab は 1 つだけです。ファイルフォーマットについては、「Oracle Product Manual」を参照してください。

注記 : *oratab* ファイルは */etc* などの場所に収容できます。デフォルトでは、Oracle ARK は *oratab* ファイルをまず */etc* で検索した後、 */var/opt/oracle* でこのファイルを検索します。このいずれかのデフォルトの場所に *oratab* ファイルがない場合は、 ORACLE_ORATABLOC を */etc/default/LifeKeeper* で *oratab* が含まれるディレクトリに設定する必要があります。

11. 自動スタートアップの無効化。LifeKeeper により制御されるデータベースは LifeKeeper により起動されるため、自動スタートアップアクションはすべて無効にしてください。LifeKeeper は、階層が作成されたときに自動スタートアップを無効にします。これは、oratab ファイルを変更することによって実現されます。

12. Listener 設定ファイル、listener.ora。エントリに新しい行を組み込まないでください（例えば、SID_NAME=xx は 1 行にします）。

13. Oracle データベースのユーザ名およびパスワード。LifeKeeper はローカルセッションおよび OS 認証を使用して、Oracle データベースを制御します。セキュリティ上の理由からローカル OS 認証をオフにする場合、LifeKeeper は指定されたユーザ名およびパスワードを使用できます。Oracle データベースユーザは、保護されるデーターベースに対して sysdba 権限として接続可能である必要があり、各サーバの Oracle データベースが同一のユーザ名とパスワードを保有している必要があります。この構成がリソースの作成中にスキップされた場合、LifeKeeper は Oracle データベースリソースを制御するためにユーザ名およびパスワードを使用しません。このパラメータはリソース作成後にいつでも追加、変更、削除することが可能です。

LifeKeeper の保護下に置かれたら、LifeKeeper およびデータベースのユーザ権限を sysdba から sysoper に下げることができます。詳細については、Oracle データベースアカウントに対するユーザ名 / パスワードの変更 を参照してください。

**Oracle ユーザ名およびパスワードの作成に関する情報**

a. Oracle データベースが起動しているノード上で、dba グループの一部であるユーザで Linux にログインしてください。(「oracle」アカウントが最も一般的です。) sqlplus ユーティリティを使用して以下のコマンドを実行し、管理ユーザとしてデーターベースに接続してください。

```
$ sqlplus / as sysdba
```

b. この機能に対する新規ユーザを作成してください。

```
SQL> CREATE USER lkdba IDENTIFIED BY “password”;
```

c. このユーザに SYSDBA 権限を与えてください。

```
SQL> GRANT SYSDBA to lkdba;
```

d. Oracle が設定されると、LifeKeeper クラスタの各ノードが $ORACLE_HOME のローカルコピーを保有するので、クラスタの各ノードでこれらのコマンドを実行してください。LifeKeeper Oracle 階層の作成後、ノード上でデータベースを In Service にし、 CREATE および GRANT コマンド (上記) を実行してOracle でユーザをセットアップしてください。

注意 : 同じファイルシステムに 2 つのデータベースを設定しないでください。どうしてもそうしなければ

ならない場合は、十分な注意が必要です。その場合には、両方のデータベースを LifeKeeper の保護下に置き、両方の階層が同じプライマリサーバとバックアップサーバを持つようにする必要があります。

# 6.11.2.2. LifeKeeper 保護用の Oracle Net Listener の設定

Oracle データベースがリモートクライアント接続に対応する場合、Oracle データベースサーバに加えて、Oracle Listener を保護することができます。Oracle ネットワーク設定ユーティリティを使用して Oracle ネットワーク設定ファイル（例えば、listener.ora 、tnsnames.ora など）を作成する方法については、Oracle ドキュメンテーションを参照してください。

注記: 複数のリソースに対しての共有 Oracle Listener の作成方法については、このドキュメントの付録の「複数のリソースに対する共有 Oracle Listener の作成」を参照してください。

## Listener の設定

1. クライアントが接続するための切り替え可能 IP アドレスを選択する必要があります。このアドレスは DNS に登録すると良いでしょう。(IP のリソース階層を作成する方法の詳細については、**LifeKeeper IP Recovery Kit ドキュメンテーション** を参照してください。リソース依存関係を作成する方法の詳細については、「GUI Administration Tasks」の リソース依存関係の作成 トピックを参照してください。)

2. *listener.ora*ファイルで、データベースサービス名の HOST にこの切り替え可能な IP アドレスを指定してください。(*listener.ora* ファイルの詳細については、Oracle ドキュメンテーションを参照してください。)データベースサービス名の HOST には、切り替え可能な IP アドレスの代わりに DNS 名を使用できますが、LifeKeeper のベストプラクティスとしては推奨されません。切り替え可能な IP アドレスを使用することにより、DNS ルックアップの問題の影響によって quickCheck や restore または remove 処理中に LifeKeeper がリスナーの実行ステータスを決定できなくなることを防止できます。また、リスナーが 1 つしか定義されていない場合でも SID_LIST_LISTENER を定義する必要があります。

listener.ora のサンプルフォーマット:

```
.
.
.
SID_LIST_LISTENER =
     (SID_LIST =
     (SID_DESC =
             (SID_NAME = <SID Name>)
       )
)
.
.
.
<listener name>=
     (DESCRIPTION_LIST =
         (DESCRIPTION =
```

```
               (ADDRESS = (PROTOCOL = TCP)(HOST = <vip>)(PORT = <port number>))
        )
)
.
.
.
```

tnsnames.ora ファイルまたは Oracle Names の HOST に切り替え可能な IP アドレスを指定してください。(tnsnames.ora ファイルの詳細については、Oracle ドキュメンテーションを参照してください。)データベースサービス名の HOST には、切り替え可能な IP アドレスの代わりに DNS 名を使用できますが、LifeKeeper のベストプラクティスとしては推奨されません。切り替え可能な IP アドレスを使用することにより、DNS ルックアップの問題の影響によって quickCheck や restore または remove 処理中に LifeKeeper がリスナーの実行ステータスを決定できなくなることを防止できます。

```
    .
    .
    .
    <SID Name>=
       (DESCRIPTION =
          (ADDRESS_LIST =
             (ADDRESS = (PROTOCOL = TCP)(HOST = <vip>)(PORT = <port number>))
             )
             (CONNECT_DATA =
             (SID = <SID Name>)
          )
       )
```

これらのサンプルファイルは、Oracle 10g および 11g の両方で正常に動作します。

listener.ora

```
SID_LIST_LISTENER =
   (SID_LIST =
   (SID_DESC =
      (SID_NAME = ORA11A)
   )
)
LISTENER =
   (DESCRIPTION_LIST =
      (DESCRIPTION =
         (ADDRESS = (PROTOCOL = TCP)(HOST = 192.0.2.0)(PORT = 1521))
      )
   )
```

tnsnames.ora

```
ORA01 =
   (DESCRIPTION =
      (ADDRESS_LIST =
         (ADDRESS = (PROTOCOL = TCP)(HOST = 192.0.2.0)(PORT = 1521))
      )
         (CONNECT_DATA =
         (SID = ORA01)
      )
   )
```

通常の listener.ora の場所は *$ORACLE_HOME/network/admin* になります。最も一般的なポート番号は 1521 です。データベースのグローバル名は、作成時に定義されます。 *$ORACLE_HOME* ディレクトリが非共有ストレージ上にインストールされている場合は、listener.ora のコピーは両システム上に必要となるので注意してください。

注記: Oracle Net は、サービスのリスナーに障害が発生した場合に、クライアント接続を自動的に他のリスナーにフェイルオーバするオプションを提供しています。この機能を利用するには、*tnsnames.ora* ファイルで FAILOVER パラメータを「**ON**」に設定します。こうしておけば、LifeKeeper によって保護された Oracle SID のリスナーに障害が発生しても、LifeKeeper が保護下にあるリスナーを回復するまで、クライアント接続は別のリスナーを通じて継続されます。

## エラー

下記エラーが発生した場合は、以下に記載の解決策を確認してください。

### Oracle listener fails with TNS-00511: No Listener Linux Error:111: Connection refused

listener.ora ファイルを下記の例のように設定した場合、LifeKeeper はリスナーを起動することができません。

```
 (ADDRESS_LIST=
        (ADDRESS=(PROTOCOL=tcp)(HOST=<hostname>)(PORT=1521))
        (ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521))
        (ADDRESS=(PROTOCOL=tcp)(HOST=<vip>)(PORT=1521))
  )
```

<hostname> は、ホストのホスト名で、 <vip> は仮想 IP アドレスです。

エラーメッセージをスクロールしていくと下記エラーが表示されます。

TNS-12542: TNS:address already in use TNS-12560: TNS:protocol adapter error

### 解決策

listener.ora を下記のように変更してください。

```
(ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=<ip>)(PORT=1521))
      (ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521))
      (ADDRESS=(PROTOCOL=tcp)(HOST=<vip>)(PORT=1521))
)
```

<ip> はホストの IP アドレスで、 <vip> は、仮想 IP アドレスです。

**注記:** セキュリティ上の問題がある場合は、 vipname が仮想 IP の DNS 名でも設定可能です。

```
(ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=tcp)(HOST=<vipname>)(PORT=1521))
)
```

これは tnsnames.ora ファイルの変更にも適用されます。

# 6.11.2.3. LifeKeeper での透過的アプリケーションフェイルオーバの設定

サーバがフェイルオーバしたり、Oracle データベースに障害が発生したりすると、ユーザは深刻な影響を受けてしまいます。通常は、データベースへの接続が切断されるだけでなく、作業中のデータもほとんど失われます。フェイルオーバが完了（または Oracle データベースが回復）すると、クライアントはアプリケーションを再起動してデータベースに再接続する必要があります。Oracle の透過的アプリケーションフェイルオーバ（TAF）機能を使用すると、障害によってはそれにマスクをかけて、その影響を減らしたりなくしたりすることができます。LifeKeeper 環境で TAF を設定する際には、LifeKeeper サーバ側で行う作業と Oracle クライアント側で行う作業があります。

この TAF 機能の効果的な利用をお考えの場合は、クライアントアプリケーションで Oracle Call Interface（OCI）から API を呼び出す際に、フェイルオーバを意識した呼び出しを行う必要があります。また、tnsnames.ora ファイルに Oracle Net のパラメータを指定して、適切な TAF の設定を行う必要があります。TAF モードは、tnsnames.ora の CONNECT_DATA セクションに FAILOVER_MODE パラメータを指定して設定します。TAF のメカニズムでは、フェイルオーバ時のクライアント接続の動作を制御するサブパラメータをいくつかサポートしています。LifeKeeper for Linux Oracle Recovery Kit では、TAF 設定時に以下のサブパラメータを指定できます。

**TYPE= (SELECT または SESSION)**

> クライアント接続を TAF でどのようにフェイルオーバするかを指定します。 **SELECT** を指定すると Oracle は、移行中に発行されたすべての select ステートメントを記録します。新しい接続が確立されるとすべての select ステートメントが再実行され、カーソルを元の位置に配置してクライアントが行のフェッチを継続できるようにします。SESSION を指定すると接続のみが新規に作成され、作業中のデータは失われます。

**METHOD= (BASIC)。**

> このメソッドを指定すると TAF は、一次接続が失敗したときのみ再接続を試みます。他に PRECONNECT メソッドがありますが、LifeKeeper では現在のところ **PRECONNECT** メソッドの使用はサポートしておりません。

**DELAY=（秒数）**

> 障害発生後、TAF が再接続を試みる間隔を秒数で指定します。クライアントアプリケーションや環境に合った値を指定する必要があります。

**RETRIES=（試行回数）**

> 障害発生後、TAF が再接続を試みる回数を指定します。 **DELAY** と **RETRIES** を組み合わせて使用する際には、サーバで障害が発生したときに Oracle が完全に回復できる、十分な時間を設定する必要があります。そうすることで TAF は、サーバのフェイルオーバの完了後、十分に時間的余裕を持って再起動できるようになります。

以下は、クライアントシステム用サンプル tnsnames.ora ファイルからの抜粋です。

```
LKproDB=
    (DESCRIPTION=
        (ADDRESS_LIST=
        (ADDRESS = (PROTOCOL=TCP) (HOST=<switchableIP>) (PORT=<port number>))
        )
        (CONNECT_DATA=
            (SID=LKroDB)
            (SERVER=DEDICATED)
            (FAILOVER_MODE=
                (TYPE=SELECT)
                (METHOD=BASIC)
                (DELAY=5)
                (RETRIES=30)
            )
        )
    )
```

通常の tnsnames.oraの場所は *$ORACLE_HOME/network/admin* になります。最も一般的なポート番号は 1521 です。 tnsnames.oraファイルはユーザのホームディレクトリにも同様に配置されます。 *$ORACLE_HOME* ディレクトリが非共有ストレージ上にインストールされている場合は、 listener.ora および tnsnames.ora のコピーが両システム上に必要になるので注意してください。

Oracle データベースを保護している LifeKeeper サーバでは、LifeKeeper によって保護されている切り替え可能な IP アドレスを使用してリスナーを設定する必要があります。Oracle Net とリスナーの設定の詳細については、前述の「LifeKeeper 保護用の Oracle Net Listener の設定」セクションを参照してください。

# 6.11.2.4. Oracle Multitenant 構成による Pluggable Database の設定

Oracle データベースが Oracle Multitenant アーキテクチャに対応し、Oracle データベースがコンテナ・データベース（以下、CDB と呼ぶ）を保護する場合、Oracle データベースサーバに加えて、プラガブル・データベース（以下、PDB と呼ぶ）を保護することができます。

## CDB と PDB の確認

1. PDB を保護するためには Oracle リソースが作成されている必要があります。また、保護している Oracle リソースが CDB である必要があります。CDB であるかの確認はデータベースへ接続後に以下のコマンドを使って確認することができます。

   ```
   SQL> select CDB from V$DATABASE;
   ```

2. PDB を保護するにあたって、CDB の中に PDB がマウントされている必要があります。PDB がマウントされているかの確認はデータベース接続後に以下のコマンドで確認することができます。

   ```
   SQL> show pdbs;
   ```

## Oracle PDB リソース階層の作成

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択してください。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択してください。

   **重要:** リソースを作成するときは、Oracle リソースが作成されており、In Serviceである必要があります。

   ダイアログボックスが表示され、クラスタ内にインストール済みの認識されているすべての Recovery Kit がドロップダウンリストボックスに表示されます。ドロップダウンリストから **[Oracle Pluggable Database]** を選択してください。 **[Next]** をクリックして次のダイアログボックスに進んでください。

   **注記:** ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、以前に入力した情報を訂正する必要がある場合に特に役立ちます。

   階層作成手順のどの段階でも、 **[Cancel]** をクリックすると、作成プロセス全体がキャンセルされます。

2. 以下の情報を入力するように要求されます。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも **[Cancel]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| **Server** | Oracle PDB リソースを作成する LifeKeeper サーバを選択します。 |
| **Switchback Type** | intelligent または automatic を選択します。フェイルオーバ後、バックアップサーバで Oracle PDB リソースが In Service (アクティブ) になるとき、プライマリサーバにスイッチバックされる方法が決まります。インテリジェントスイッチバック (intelligent) の場合、リソースをプライマリサーバにスイッチバックするときに管理者の介入が必要ですが、自動スイッチバック (automatic) の場合は、プライマリサーバがオンラインに戻り、LifeKeeper のコミュニケーションパスが再確立されるとすぐにスイッチバックが実行されます。<br><br>**注記:** スイッチバック方法は、Oracle PDB リソースが使用する依存リソースのスイッチバック方法と一致する必要があります。 |
| **ORACLE_SID** | 保護しているOracleデータベースのSIDを指定します。 |
| **Oracle PDBs** | 保護したいPDBを指定します。このフィールドでは複数選択が可能です。 |
| **PDB Tag** | プライマリサーバ上の新しい Oracle PDB リソースに対する固有のタグ名です。デフォルトのタグ名は「pdb-<ORACLE_SID>」となっています.。別の固有のタグ名を入力することもできます。タグ名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |

3. **[Next]** をクリックしてください。 **[Create Resource Wizard]** が表示され、Oracle PDB リソース階層が作成されます。LifeKeeper によって入力データが検査されます。問題が検出された場合、情報ボックスにエラーメッセージが表示されます。

4. Oracle PDB リソース階層が正常に作成されたこと、フェイルオーバによる保護を実現するにはその階層をクラスタ内の別のサーバに拡張する必要があることを示すメッセージが表示されます。**[Next]** をクリックします。

5. **[Continue]** をクリックします。 **[Pre-extend Wizard]** が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、「Oracle PDB リソース階層の拡張」の **手順 2** を参照してください。

# Oracle PDB リソース階層の拡張

この操作は **[Edit]** メニューから開始できます。または、 **[Create Resource Hierarchy]** オプションが完了すると自動的に開始されます。その場合は下の手順 2 を参照してください。

1. **[Edit]** メニューの **[Resource]** から **[Extend Resource Hierarchy]** を選択します。 **[Pre-Extend Wizard]** が表示されます。拡張操作に慣れていない場合は、 **[Next]** をクリックします。LifeKeeper のリソース階層の拡張作業のデフォルト値が分かっていて、入力と確認を省略する場合は、**[Accept Defaults]** をクリックします。

2. **[Pre-Extend Wizard]** で以下の情報を入力します。

   **注記:** 最初の 2 つのフィールドは **[Edit]** メニューの **[Extend]** から操作を開始した場合にだけ表示さ

れます。

| フィールド | ヒント |
|---|---|
| **Template Server** | Oracle PDB リソースが現在 In Service のサーバを選択します。 |
| **Tag to Extend** | 拡張する Oracle PDB リソースを選択します。 |
| **Target Server** | 拡張先のサーバを入力または選択します。 |
| **Switchback Type** | フェイルオーバ後、バックアップサーバで Oracle PDB リソースが In Service (アクティブ) になるとき、プライマリサーバにスイッチバックされる方法が決まります。intelligent または automatic を選択できます。スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。<br><br>**注記:** スイッチバック方法は、Oracle PDB リソースが使用する依存リソースのスイッチバック方法と一致する必要があることに注意してください。 |
| **Template Priority** | テンプレートの優先順位 を選択または入力します。これはサーバで現在 In Service の Oracle PDB 階層の優先順位です。優先順位は、1 ~ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。<br><br>**注記:** このフィールドは階層を最初に拡張するときだけ表示されます。 |
| **Target Priority** | これは、新しく拡張する Oracle PDB 階層が、他のサーバに存在する同等の階層に対して持つ相対的な優先順位です。1 ~ 999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |

3. 拡張前のチェックが正常に終了したというメッセージが表示されたら、 **[Next]** をクリックします。

4. 拡張する階層に応じて、拡張されるリソースタグ (一部編集不可) を示す一連の情報ボックスが表示されます。

5. **[Extend Wizard]** でTag名が正しいかを確認し **[Extend]** をクリックします。

6. 「Hierarchy extend operations completed (階層の拡張操作が完了しました)」というメッセージが表示されたら、階層を別のサーバに拡張する場合は、 **[Next Server]** をクリックし、拡張作業が必要ない場合は、 **[Finish]** をクリックします。

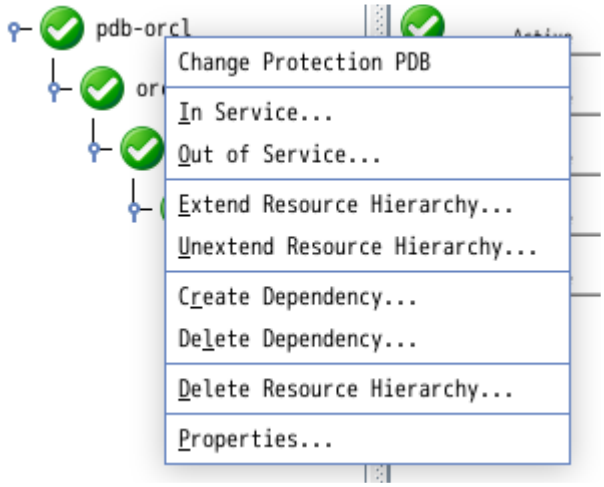
7. 「Hierarchy Verification Finished (階層の検査が完了しました)」というメッセージが表示されたら、

**[Done]** をクリックします。

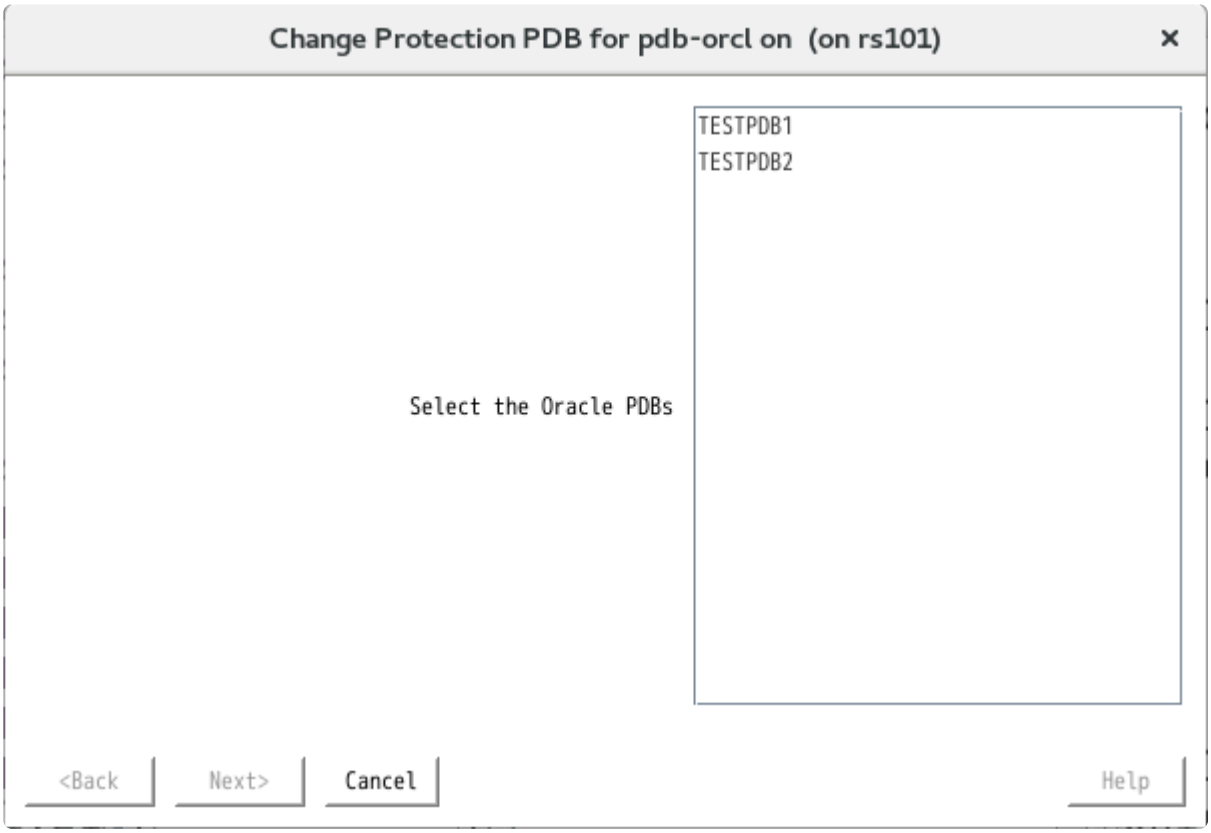
## 保護する PDB の変更

階層作成後、以下の手順で保護する PDB を変更してください。

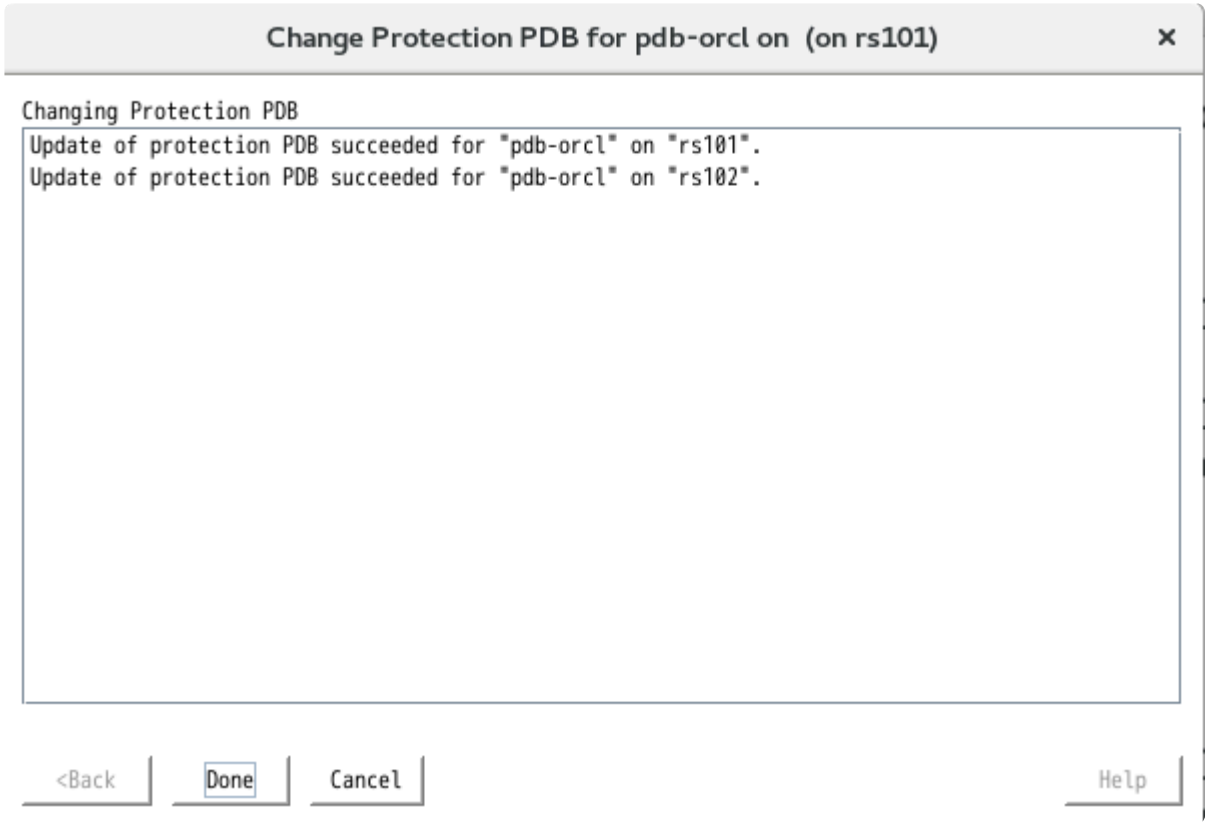
1. LifeKeeper GUI から Oracle PDB リソース階層を右クリックし、**[Change Protection PDB]** を選択してください。




2. 保護したい PDB を選択してください。（複数選択可能です）




3. **[Next]** をクリックすると設定が変更されます。




4. **[Done]** をクリックして設定変更を完了してください。

# 6.11.2.5. 設定例

以下の図は、LifeKeeper SPS 環境における Oracle のアクティブ / スタンバイおよびアクティブ / アクティブの設定例を示しています。

本セクションの例を見れば、Oracle データベースインスタンスをローカルディスクおよび共有ディスク上に設定する方法がわかります。それぞれの表は、設定タイプと Oracle パラメータの関係を示しています。これらの設定は、本書で説明している、Oracle の設定と LifeKeeper ソフトウェアの互換性を保証する設定ルールおよび要件にも適合しています。

本セクションでは、まず設定の要件を説明し、次にそれぞれの設定例を示します。

- アクティブ / スタンバイ
- アクティブ / アクティブ

本セクションの例は、設定のサンプルにすぎませんが、これらの設定を理解し、設定ルールに従うことにより、利用する環境に適した効果的なソリューションの定義およびセットアップが容易にできるようになります。

# 6.11.2.5.1. 設定の要件

それぞれの例では、1 つまたは 2 つのデータベース （ **databaseA** 、 **databaseB** ）が使用されています。デフォルトでは、Oracle データベースのシステム識別子（SID）と一致するタグ名が LifeKeeper から示されます。ただし、以下の例では、databaseA-on-server1 のように SID とサーバ名で構成されたタグ名が使用されています。

設定例を理解するには、次のような設定要件を留意しておいてください。

- **LifeKeeper 階層** 。LifeKeeper の管理を実行するときに、プライマリサーバは Oracle インスタンスが現在実行されている場所を参照します。LifeKeeper 階層を作成するときのシステム管理はこのサーバで行われます。この設定例では、プライマリサーバは Server 1、バックアップサーバまたは代替サーバは Server 2 です。

- **1 台のサーバのみによってロックされる共有ディスク** 。共有ストレージリソースが LifeKeeper の保護下にあると、そのリソースに一度にアクセス可能なサーバは 1 つだけになります。共有デバイスがディスクアレイである場合、LUN 全体が保護されます。共有デバイスがディスクである場合、ディスク全体が保護されます。共有デバイスがディスクの場合は、ディスク全体がリザーブされます。サーバに障害が発生すると、最も優先順位の高いバックアップサーバが自身の保護を確立し、他のサーバをすべてロックします。

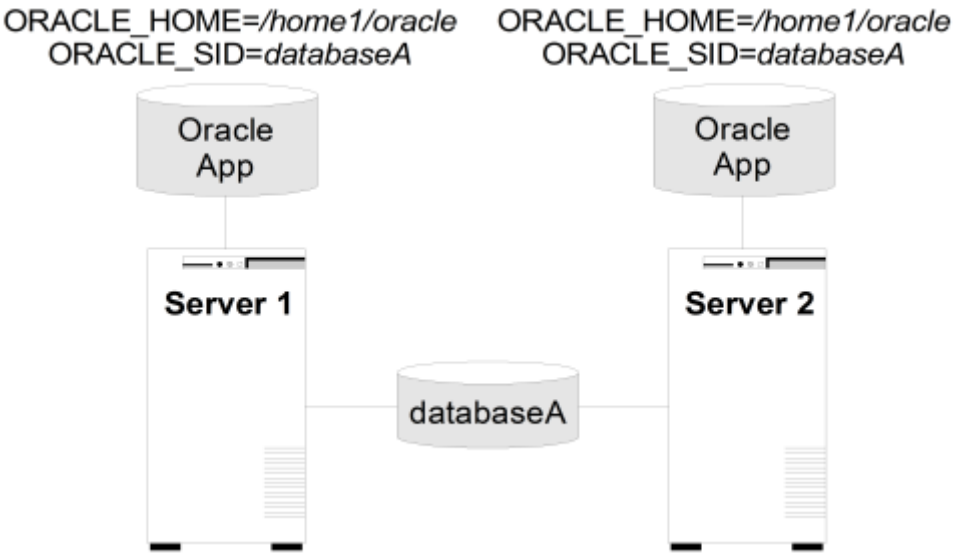
- **共有ディスク上のデータベース** 。LifeKeeper Oracle Recovery Kit が正しく機能するには、データベースが常に共有デバイス上にあることが必要です。データベースは 1 つまたは複数のファイルシステムおよびディスク上に存在することができます。

**注記:** それぞれの設定図に付随する表は、プライマリサーバ（Server 1）の場合は [Create Resource Hierarchy] ウィザードに、バックアップサーバ（Server 2）の場合は [Extend Resource Hierarchy] ウィザードに入力される適切な情報の例を示しています。ウィザードに入力する情報の詳細については、LifeKeeper の設定作業 を参照してください。これらの表は、Recovery Kit を設定するときに役立つ参照情報となります。

# 6.11.2.5.2. アクティブ / スタンバイ設定

このセクションでは、2 つのアクティブ / スタンバイ設定の例（図 1 と図 2）を紹介します。これらの構成では、Server 1 がデータベースへの排他アクセスを持っているため、アクティブと見なされます。Server 2 は別の処理を実行します。Server 1 に障害が発生すると、Server 2 がデータベースにアクセス可能になり、LifeKeeper がデータベースの稼動を再確立します。

**図 1アクティブ / スタンバイ設定、例 1**




**設定に関する注記:**

各サーバは独自の $ORACLE_HOME ディレクトリを非共有ディスク上に置いています。各サーバには同じバージョンの Oracle アプリケーションが導入されています。

$ORACLE_HOME パスはどちらのサーバでも同じです。

データベース databaseA は共有ディスク上にあります。

Server1 でのリソース階層の作成:

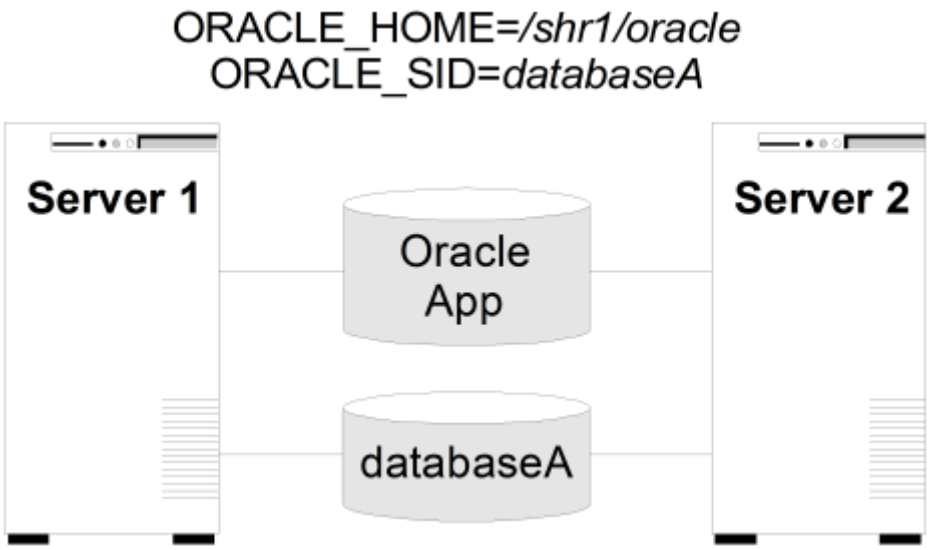
| | |
|---|---|
| サーバ: | Server1 |
| データベースの ORACLE_SID | databaseA |
| データベースのユーザ名 | system |
| ユーザ名のパスワード | ********** |
| データベースの ORACLE_HOME | /home1/oracle |
| データベースタグ: | databaseA-on-server1 |

Server 2 へのリソース階層の拡張:

| | |
|---|---|
| テンプレートサーバ: | Server1 |

| | |
|---|---|
| 拡張するタグ: | databaseA-on-server1 |
| ターゲットサーバ: | Server2 |
| ターゲットの優先順位: | 10 |
| データベースタグ: | databaseA-on-server1 |

図 2アクティブ / スタンバイ設定、例 2




**設定に関する注記:**

1.
2. どちらのサーバも共有ディスク上の $ORACLE_HOME ディレクトリを使用します。
3. $ORACLE_HOME パスはどちらのサーバでも同じです。
4. データベース databaseA は共有ディスク上にあります。
5. Server 1 がアクティブである間、Server 2 は共有ディスクのファイルおよびディレクトリにアクセスできません。
6. $ORACLE_HOME はデータベースと同じ共有ディスク上または別のディスク上に置くことができます。

Server1 でのリソース階層の作成:

| | |
|---|---|
| サーバ: | Server1 |
| データベースの ORACLE_SID | databaseA |
| データベースのユーザ名 | system |
| ユーザ名のパスワード | ********** |
| データベースの ORACLE_HOME | /shr1/oracle |

| | |
|---|---|
| データベースタグ: | databaseA-on-server1 |

Server 2 へのリソース階層の拡張:

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | databaseA-on-server1 |
| ターゲットサーバ: | Server2 |
| ターゲットの優先順位: | 10 |
| データベースタグ: | databaseA-on-server1 |

# 6.11.2.5.3. アクティブ / アクティブ設定

アクティブ / アクティブ設定は、それぞれが別々のデータベースインスタンスを実行する 2 台以上のサーバで構成されます。各データベースは、別々の共有ディスク上にある必要があります。

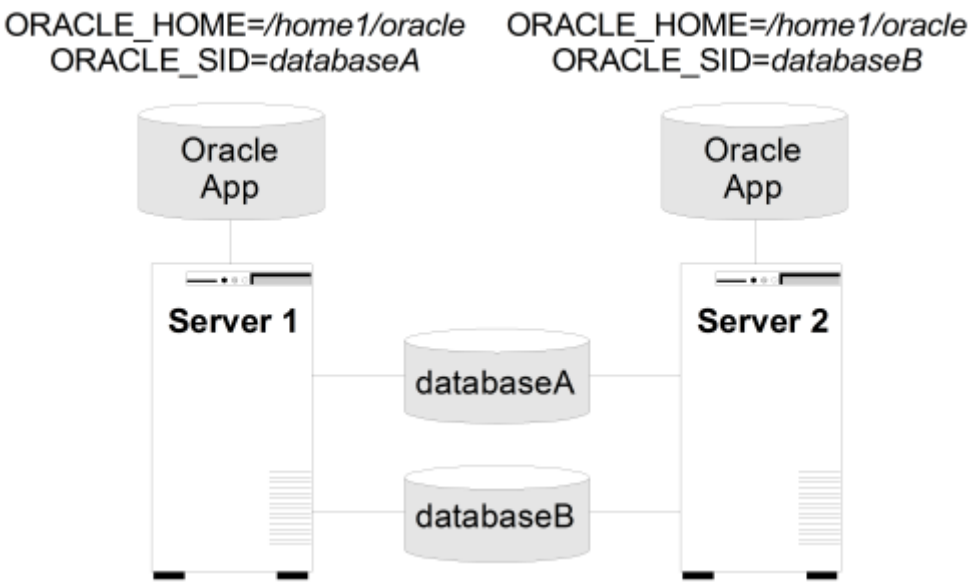
$ORACLE_HOME は、設定の要件に応じて、非共有ディスクまたは共有ディスク上に置くことができます。例えば、共通の$ORACLE_HOME を使用する任意のサーバ上の複数のデータベースインスタンスは、$ORACLE_HOME が非共有ディスク上にある必要があります。$ORACLE_HOME ディレクトリが共有ディスク上にある場合は、別々の共有ディスクでなければなりません。

本セクションでは、図 3 と図 4 に示されている 2 種類のアクティブ / アクティブ設定の例を検討します。

- 共有リソース上のデータベースと、非共有リソース上の一般的な $ORACLE_HOME。
- 共有リソース上のデータベースと、同じ共有リソース上の適切な $ORACLE_HOME。

**注記:** 非共有ストレージ上で複数の $ORACLE_HOME を使用しているサーバの複数のデータベースインスタンスは除外されます。

図 3アクティブ / アクティブ設定の例




**設定に関する注記:**

1. サーバは独自の $ORACLE_HOME ディレクトリを非共有ディスク上に配置します。各サーバには同じバージョンの Oracle アプリケーションが導入されています。
2. $ORACLE_HOME パスはどちらのサーバでも同じです。
3. 2 つのデータベース、databaseA と databaseB は共有ディスク上にあります。

4. 両方の Oracle インスタンスのエントリが含まれている oratab ファイルは、両サーバの /etc/ にあります。

5. 最初は、Server 1 で databaseA が、Server 2 で databaseB が稼動します。スイッチオーバが行われると、1 つのシステムが両方のデータベースを実行できます。

6. 詳細については、ローカルストレージでの Oracle バイナリのインストール後の Oracle データベース階層の作成 を参照してください。

Server 1 での 1 つ目のリソース階層の作成:

| | |
|---|---|
| サーバ: | Server1 |
| データベースの ORACLE_SID: | databaseA |
| データベースのユーザ名 | system |
| ユーザ名のパスワード | ********** |
| データベースの ORACLE_HOME | /home1/oracle |
| データベースタグ: | databaseA-on-server1 |

1 つ目のリソース階層の Server 2 への拡張:

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | databaseA-on-server1 |
| ターゲットサーバ: | Server2 |
| ターゲットの優先順位: | 10 |
| データベースタグ: | databaseA-on-server1 |

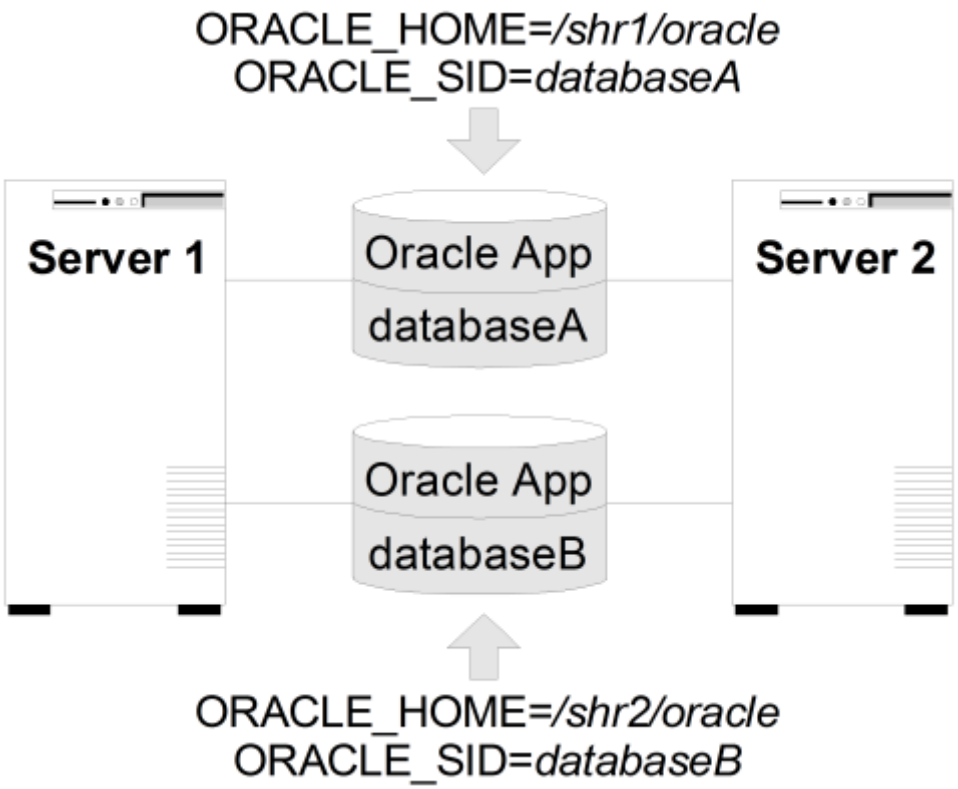
Server 2 での 2 番目のリソース階層の作成:

| | |
|---|---|
| サーバ: | Server2 |
| データベースの ORACLE_SID: | databaseB |
| データベースのユーザ名 | system |
| ユーザ名のパスワード | ********** |
| データベースの ORACLE_HOME | /home1/oracle |
| データベースタグ: | databaseB-on-server2 |

2 つ目のリソース階層の Server 1 への拡張:

| | |
|---|---|
| テンプレートサーバ: | Server2 |
| 拡張するタグ: | databaseB-on-server2 |
| ターゲットサーバ: | Server1 |
| ターゲットの優先順位: | 10 |
| データベースタグ: | databaseB-on-server2 |

図 4アクティブ / アクティブ設定、例 2




**設定に関する注記:**

1. 両サーバが別々の共有ディスク上の $ORACLE_HOME ディレクトリを使用します。
2. Oracle アプリケーションはどちらのサーバでも同じです。$ORACLE_HOME は、サーバに定義されたインスタンスごとに異なります。
3. 2 つのデータベース、databaseA と databaseB は共有ディスク上にあります。
4. 両方の Oracle インスタンスのエントリが含まれている oratab ファイルは /etc/ にあります。

5. Oracle インスタンスごとに一意のログインが必要です。それぞれのログインに対する Id および gid は Server 1 および Server 2 で同一である必要があります。

6. 最初は、Server 1 で databaseA が、Server 2 で databaseB が稼動します。スイッチオーバが行われると、1 つのシステムが両方のデータベースを実行できます。

Server 1 での 1 つ目のリソース階層の作成:

| | |
|---|---|
| サーバ: | Server1 |
| データベースの ORACLE_SID: | databaseA |
| データベースのユーザ名 | system |
| ユーザ名のパスワード | ********** |
| データベースの ORACLE_HOME | /shr1/oracle |
| データベースタグ: | databaseA-on-server1 |

1 つ目のリソース階層の Server 2 への拡張:

| | |
|---|---|
| テンプレートサーバ: | Server1 |
| 拡張するタグ: | databaseA-on-server1 |
| ターゲットサーバ: | Server2 |
| ターゲットの優先順位: | 10 |
| データベースタグ: | databaseA-on-server1 |

Server 2 での 2 番目のリソース階層の作成:

| | |
|---|---|
| サーバ: | Server2 |
| データベースの ORACLE_SID: | databaseB |
| データベースのユーザ名 | system |
| ユーザ名のパスワード | ********** |
| データベースの ORACLE_HOME | /shr2/oracle |
| データベースタグ: | databaseB-on-server2 |

2 つ目のリソース階層の Server 1 への拡張:

| | |
|---|---|
| テンプレートサーバ: | Server2 |
| 拡張するタグ: | databaseB-on-server2 |
| ターゲットサーバ: | Server1 |
| ターゲットの優先順位: | 10 |
| データベースタグ: | databaseB-on-server2 |

# 6.11.3. LifeKeeper の設定作業

設定作業は LifeKeeper GUI を使用して実行できます。以下の 4 つの作業は、Oracle リソースインスタンス特有のものであり、Recovery Kit ごとに異なるため、本書で説明しています。

- リソース階層の作成 。 アプリケーションリソース階層を LifeKeeper クラスタに作成します。
- リソース階層の削除 。 リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。
- リソース階層の拡張 。 リソース階層をプライマリサーバからバックアップサーバへ拡張します。
- リソース階層の拡張解除 。 リソース階層を LifeKeeper クラスター内の 1 つのサーバから拡張解除（削除）します。
- Oracle 設定の参照 - [Resource Properties] ダイアログを参照できます。
- ユーザ名 / パスワードの変更 。 Oracle データベースを保護するためにログインするユーザ名およびパスワードの変更します。
- リソース階層のテスト 。 手動スイッチオーバを実施することで、Oracle リソース階層をテストできます。

以下のタスクはすべての Recovery Kit で手順が同じであるためSPS for Linux テクニカルドキュメンテーション の管理 セクション で説明します。

- リソース依存関係の作成 .既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除 .リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service .リソース階層を特定のサーバで In Service にします。
- Out of Service .リソース階層を特定のサーバで Out of Service にします。
- 表示 / 編集 特定のサーバでリソース階層のプロパティを表示または編集します。

**注記:** 本セクションの残りの部分では、LifeKeeper GUI の **[Edit]** メニューから特定の作業を選択して Recovery Kit を設定する方法について説明します 。設定作業はツールバーから選択することもできます。状況表示ウィンドウの [Resource Hierarchy Tree]（左側のペイン） のグローバルリソースを右クリックして、[Edit] メニューと同じドロップダウンメニューの選択項目を表示することもできます。言うまでもなく、これは階層がすでに存在している場合にだけ可能な方法です。

状況表示ウィンドウの [Resource Hierarchy Table]（右側のペイン） のリソースインスタンスを右クリックして、サーバおよび特定リソースの状況に応じて、 *リソース階層の作成* を除くすべての設定作業を実行することもできます。

# 6.11.3.1. リソース階層の作成

**注記:** Oracle Net リモートクライアントアクセスを利用するには、クライアントの接続に使用する IP アドレスが、Oracle リソース階層に依存関係を持っている LifeKeeper の保護下にある必要があります。（詳細については、Lifekeeper 保護用の Oracle Net Listener の設定 を参照してください。）

プライマリサーバからリソースインスタンスを作成するには、以下の手順を実行します。

1.
2. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択してください。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択してください。

   **重要:** リソースを作成するときは、Oracle アプリケーションが実行されている必要があります。

   ダイアログボックスが表示され、クラスタ内にインストール済みの認識されているすべての Recovery Kit がドロップダウンリストボックスに表示されます。ドロップダウンリストから **[Oracle Database]** を選択してください。 **[Next]** をクリックして次のダイアログボックスに進んでください。

   **注記:** ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、以前に入力した情報を訂正する必要がある場合に特に役立ちます。

   階層作成手順のどの段階でも、 **[Cancel]** をクリックすると、作成プロセス全体がキャンセルされます。

3. **[Switchback Type]** を選択してください。この選択によって、バックアップサーバへのフェイルオーバの後、Oracle インスタンスが In Service に戻ったときに、このサーバにどのようにスイッチバックされるのかが決まります。intelligent または automatic を選択できます。Intelligent switchback では、インスタンスをプライマリ／オリジナルサーバにスイッチバックするときに、管理者の介入が必要になります。自動スイッチバックの場合は、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後にスイッチバックが行われます。

   スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。 **[Next]** をクリックして次のダイアログボックスに進んでください。

4. Oracle データベースを配置したい **サーバ** を選択してください（通常これは、プライマリサーバまたはテンプレートサーバと呼ばれます）。クラスタ内のすべてのサーバがドロップダウンリストに表示されます。 **[Next]** をクリックして次のダイアログボックスに進んでください。

5. このOracleデータベースの **ORACLE_SID** を選択してください。これは、設定するデータベースの Oracle システム識別子を指定するタグ名です。 */etc/oratab* にこのデータベースのエントリが存在しなければなりません。 **[Next]** をクリックして次のダイアログボックスに進んでください。

6. **ORACLE_SID** の **[Username]** を入力してください。これは ORACLE_SID にログインする間に指定する Oracle データベースのユーザ名になります。このユーザ名は、フルコントロールを与えるためにデータベースに対して sysdba 権限で接続可能である必要があります。 **[Next]** をクリックし

て次のダイアログボックスに進んでください。（このフィールドは空欄のままでも問題ありません。空欄にした場合は、LifeKeeper は Oracle データベースユーザリソースを制御するためにユーザ名およびパスワードを使用しません。次のステップの **[Input Password]** がスキップされます。）

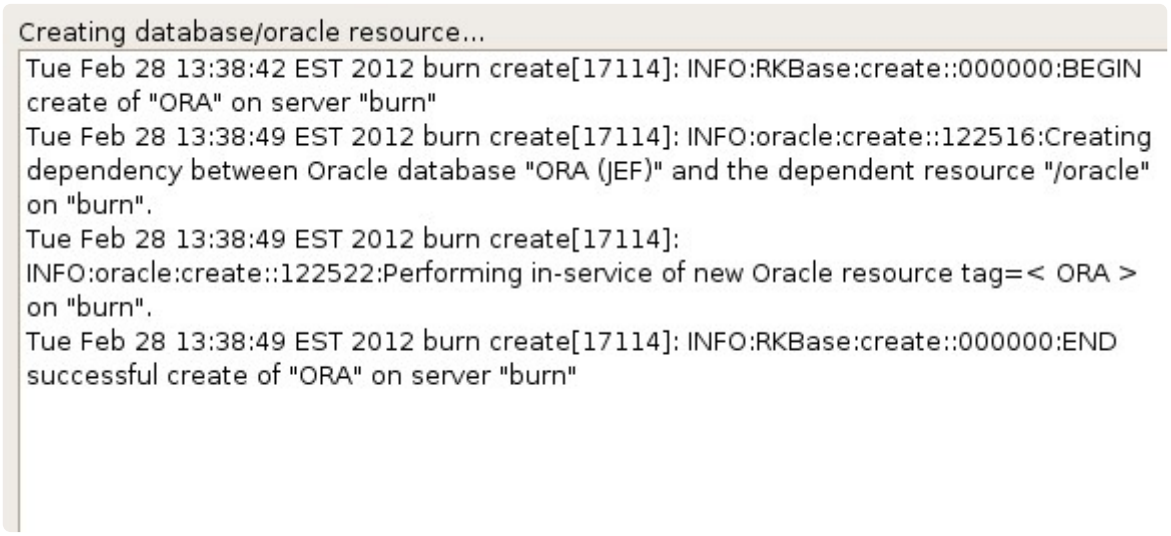
7. **[Password]** を入力してください。このパスワードは ORACLE_SID にログインする際に指定するパスワードです。このパスワードは、LifeKeeper により暗号化されて保存されます。 **[Next]** をクリックして次のダイアログボックスに進んでください。

8. Oracle リソースに依存して含まれる Listener の **タグ名** を選択してください。リストが、現在サーバ上で保護されているすべての Listener リソースを表示します。Oracle SID に対して要求される Listener に一致した Listener リソースタグを選択してください。Listener リソースが存在しない場合は、 **[None]** を選択してください。

9. **[Database Tag]** を選択または入力します。これは、LifeKeeper が Oracle 階層に与えるタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。

   **[Create]** をクリックすると、 **[Create Resource]** ウィザードによって Oracle リソースが作成されます。

10. この時点で情報ボックスが表示され、LifeKeeper は、Oracle リソース階層を作成するのに有効なデータが提供されているか検査します。LifeKeeper が問題を検知した場合は、インフォメーションボックスにエラーが表示されます。認証に成功すると、リソースが作成されます。




**[Next]** をクリックして **[Pre-extend]** ダイアログボックス （このドキュメントで後述）に進んでください。リソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにリソース階層を拡張する必要があります。

# 6.11.3.2. リソース階層の削除

リソース階層を SPS 環境から削除するには、次の手順を実行します。

1.
2. [LifeKeeper GUI] メニューから **[Edit]** を選択し、次に **[Resouce]** を選択してください。ドロップダウンメニューから、 **[Delete Resource Hierarchy]** を選択してください。

3. Oracle リソース階層を削除する **[Target Server]** の名前を選択してください。

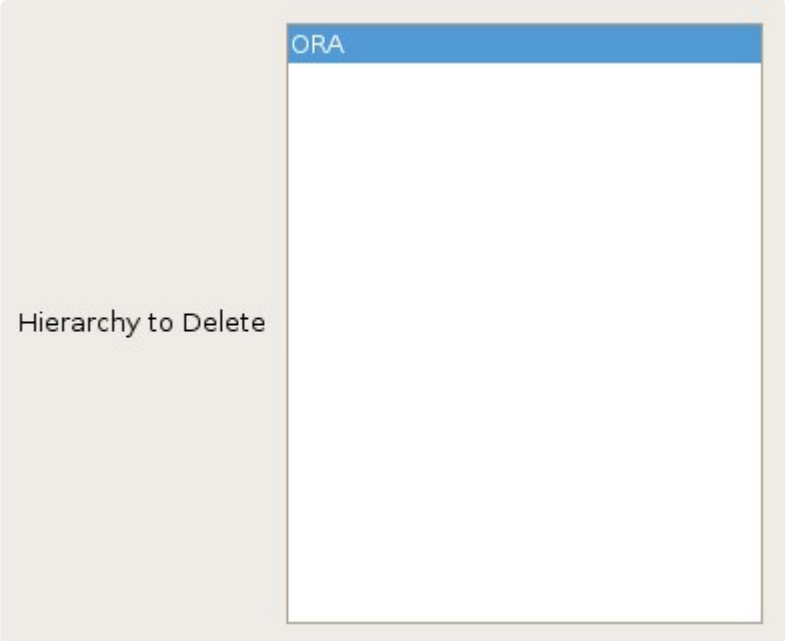
   **注記:** 右側ペインで個々のリソースインスタンスを右クリックした場合、またはリソースが 1 台のサーバ上のみにあり、左側ペインでグローバルリソースを右クリックし、[Delete Resource] 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. [Hierarchy to Delete] を選択してください。削除するリソース階層を指定して、強調表示にしてください。

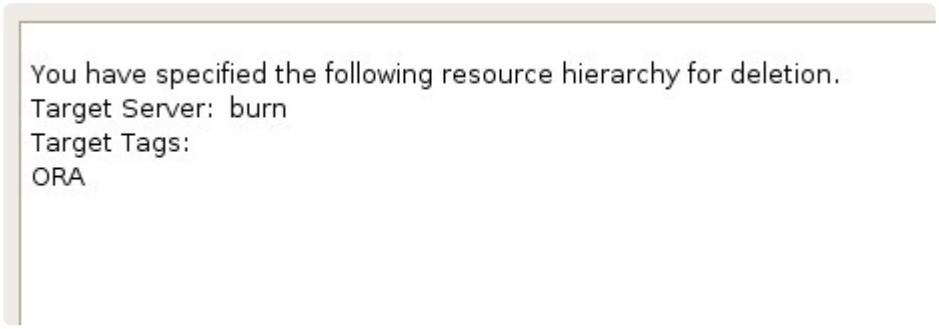
   **注記:** 左側のペインからグローバルリソースを右クリックするか、右側のペインから個々のリソースインスタンスを右クリックして [Delete Resource] 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

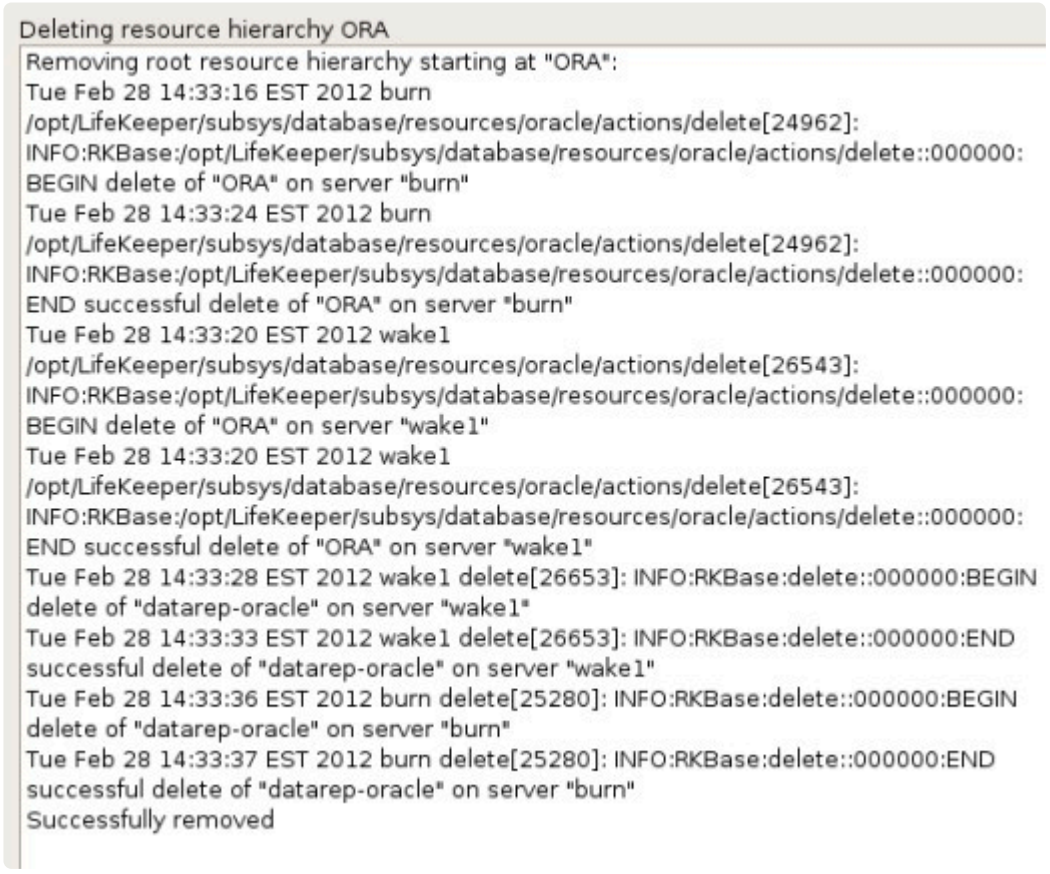
5. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示され

ます。




**[Delete]** をクリックしてリソースを削除し、最後のダイアログボックスに進んでください。

6. Oracle リソースが正常に削除されたことを示す別の情報ボックスが表示されます。




7. **[Done]** をクリックして、 **[Delete Resource Hierarchy]** メニューを終了します。

**注記:** 複数のリソースに対する共有 Oracle Listener の作成方法については、付録の「複数のリソースに対する共有 Oracle Listener の作成」を参照してください。

# 6.11.3.3. リソース階層の拡張

階層の作成後、クラスタ内の別のサーバに拡張する必要があります。3 通りのシナリオで、テンプレートサーバからターゲットサーバにリソースインスタンスを拡張できます。最初のシナリオは、リソース作成後、表示された情報ダイアログボックスで **[Next]** をクリックすることで、「**続けて**」、そのリソースを別のサーバに拡張する場合です。2 番目のシナリオは、以下に示すように [Edit] メニューから **[Extend Resource Hierarchy]** 作業を入力する場合です。3 番目のシナリオは、左側または右側のペインから拡張されていない階層を右クリックする場合です。どのシナリオでも同じダイアログボックスが表示されます（いくつかの例外については、以下に詳細を明記します）。

1.
2. LifeKeeper GUI メニューから **Extend ウィザード** を起動している場合は、 **[Edit]** を選択し、次に **[Resource]** を選択してください。ドロップダウンメニューから **[Extend Resource Hierarchy]** を選択してください。これで ***Pre-Extend Resource Hierarchy ウィザード*** が起動されます。

3. 最初のダイアログボックスが表示され、現在 Oracle リソース階層がサービス中の **テンプレートサーバ** を選択するように指示されます。ここで選択する **テンプレートサーバ** と次のダイアログボックスで選択する **[Tag to Extend]** によって、サービス中のリソース階層が表されることを認識しておくことが重要です。選択したテンプレートサーバでサービス中でないリソースタグを選択すると、エラーメッセージが表示されます。このダイアログのドロップダウンボックスには、クラスタ内の全サーバの名前が示されています。

   注記 : Oracle リソース階層の作成に続いて **[Pre-Extend Resource Hierarchy]** 作業に入った場合、ウィザードが作成段階にあるテンプレートサーバをすでに確認しているため、このダイアログボックスは表示されません。GUI ウィンドウの左側ペインの Oracle リソースアイコンまたは右側ペインの Oracle リソースボックスを右クリックして、 **[Extend Resource Hierarchy]** を選択した場合も同様です。

   階層を拡張する手順の間に **[Cancel]** をクリックすると、どの時点であってもそのサーバへの拡張プロセスがキャンセルされるので注意してください。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

   例えば、Server 1 にリソースを作成し、そのリソースを Server 2 へ拡張したとします。同じリソースを Server 3 に拡張している最中に、気が変わっていずれかのダイアログボックス内の **[Cancel]** ボタンをクリックしたとします。この場合は、Server3 へのリソース拡張だけがキャンセルされ、Server2 への拡張はキャンセルされません。この階層から Server2 を削除したい場合は、Server2 からリソースを拡張解除する必要があります。

   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. **[Tag to Extend]** を選択してください。これは、テンプレートサーバからターゲットサーバに拡張する Oracle インスタンスの名前です。ウィザードのドロップダウンメニューには、前のダイアログボックスで選択した、テンプレートサーバ上に作成されているすべてのリソースが表示されます。

**注記:** ここでも、Oracle リソース階層の作成に続いて [Pre-Extend Resource Hierarchy] 作業に入った場合、ウィザードが作成段階にある Oracle リソースのタグ名をすでに確認しているため、 **このダイアログボックスは表示されません** 。GUI ウィンドウの左側ペインの Oracle リソースアイコンまたは右側ペインの Oracle リソースボックスを右クリックして、 **[Extend Resource Hierarchy]** を選択した場合も同様です。

**[Next]** をクリックして次のダイアログボックスに進んでください。

5. Oracle リソース階層を拡張する **ターゲットサーバ** を選択してください。ドロップダウンボックスに、選択された階層にまだ入っていないクラスタ内のサーバ名の一覧が表示されます。

**[Next]** をクリックして次のダイアログボックスに進んでください。

6. **[Switchback Type]** を選択します。この選択によって、バックアップサーバへのフェイルオーバの後、Oracle インスタンスがサービス中に戻ったときに、このサーバにどのようにスイッチバックされるのかが決まります。 *intelligent* または *automatic* を選択できます。Intelligent switchback では、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに、管理者の介入が必要になります。automatic switchback の場合は、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後にスイッチバックが行われます。

スイッチバックタイプは、必要な場合 [Resource Properties] ダイアログボックスの [General] タブで後から変更できます。

**[Next]** をクリックして次のダイアログボックスに進んでください。

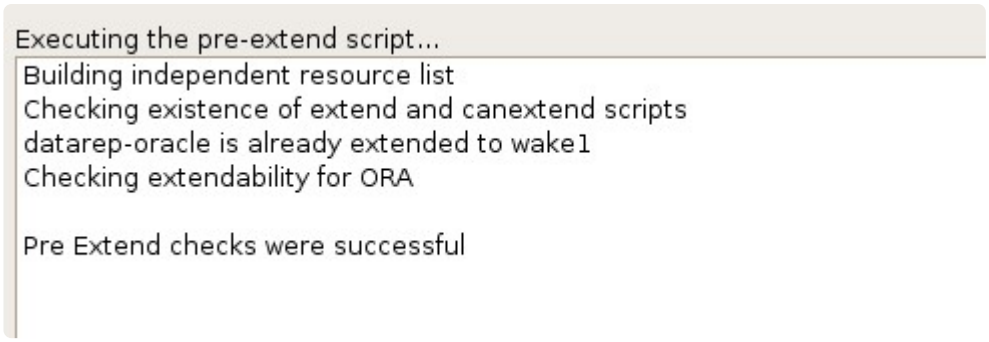
7. **[Template Priority]** を選択または入力します。これはサーバ上で現在サービス中の Oracle 階層の優先順位です。優先順位は、1～999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります（数字 1 が最高の優先順位に相当します）。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。注記:このフィールドは階層を最初に拡張するときだけ表示されます。

8. **[Target Priority]** を選択または入力します。これは、他のサーバにある同等の階層に対する、新しく拡張する Oracle 階層の優先順位です。1～999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位は高くなります（1 は最高の優先順位を表します）。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ

優先順位を割り当てることはできません。

**[Next]** をクリックします。

9. 情報ボックスが表示され、LifeKeeper が環境のチェックを正常に完了し、この Oracle リソースを拡張するためのすべての要件が満たされていることが示されます。満たされていない要件がある場合は、[Next] ボタンは選択できなくなり、[Back] ボタンが有効になります。



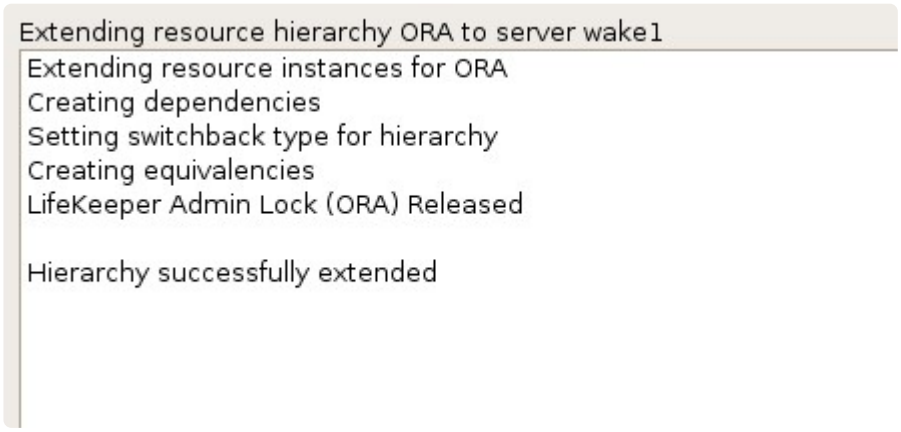

**[Back]** をクリックした場合、情報ボックスに表示されるエラーメッセージの内容に従って、リソースの拡張を変更できます。

ここで **[Cancel]** をクリックすると、いずれかの時点でここに戻り、Oracle リソース階層を別のサーバに拡張して、LifeKeeper の保護下に置く必要があります。

**[Next]** をクリックすると、[Extend Resource Hierarchy] 設定作業に入ります。

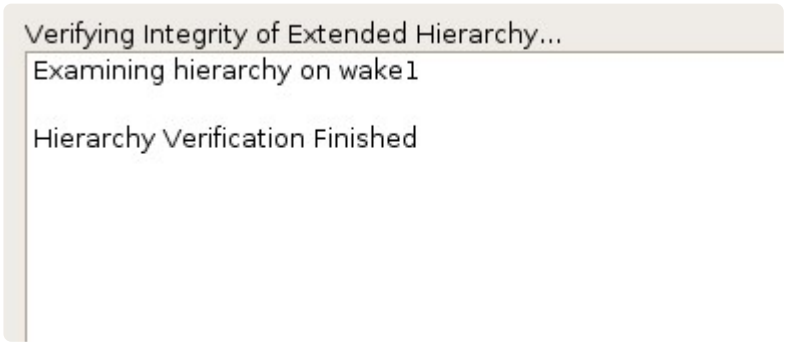
10. **[Database Tag]** が表示されたら、 **[Extend]** をクリックしてください。

11. 拡張が実行されていることを示す情報ボックスが表示されます。



同じ Oracle リソースインスタンスをクラスタ内の別のサーバに拡張したい場合は、 **[Next Server]** をクリックします。その場合は、 **[Extend Resource Hierarchy]** 操作が繰り返されます。

**[Finish]** をクリックすると、Oracle リソースの拡張が正常に完了したことを示す別のダイアログボックスが表示されます。




12. **[Done]** をクリックして、 **[Extend Resources Hierarchy]** メニューを終了します。

    **注記**：必ず両方のサーバで新しいインスタンスの機能をテストしてください。

# 6.11.3.4. リソース階層の拡張解除

1.
2. **[LifeKeeper GUI]** メニューから **[Edit]** を選択し、次に **[Resouce]** を選択してください。ドロップダウンメニューから **[Unextend Resource Hierarchy]** を選択してください。

3. Oracle リソースを拡張解除したい **Target Server** を選択してください。Oracle が現在 In Service のサーバは選択できません。

   **注記:** 右側のペインから個々のリソースインスタンスを右クリックして **拡張解除** 作業を選択した場合、このダイアログボックスは表示されません。

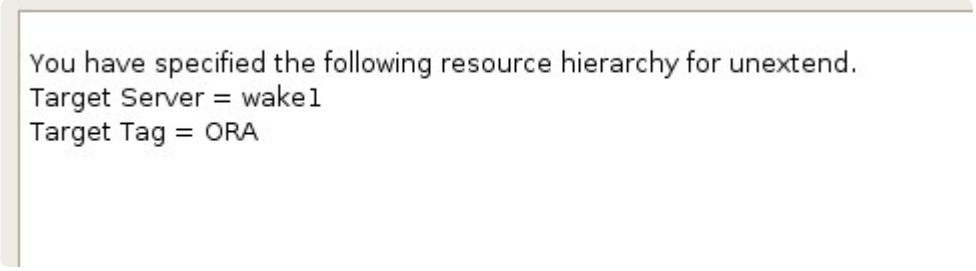
   **[Next]** をクリックして次のダイアログボックスに進んでください。

4. **拡張解除する Oracle 階層** を選択してください。 **注記:** 左側のペインからグローバルリソースを右クリックするか、右側のペインから個々のリソースインスタンスを右クリックして **拡張解除** 作業を選択した場合、このダイアログボックスは表示されません。

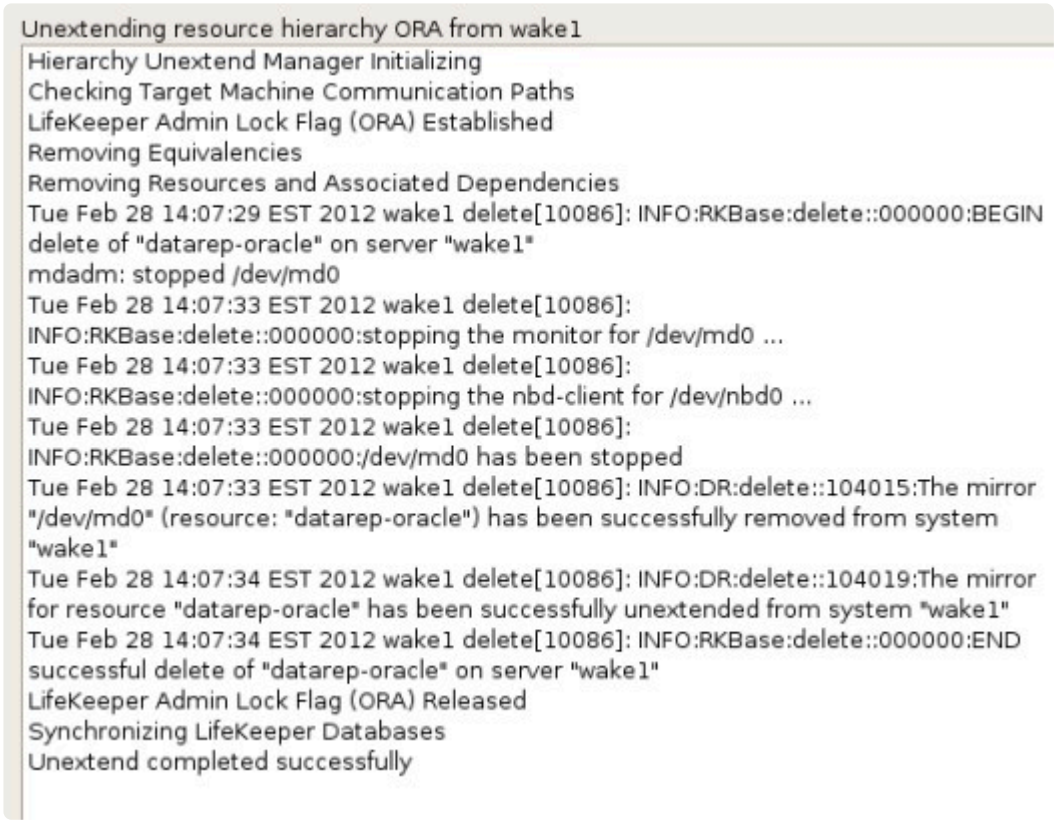
   **[Next]** をクリックして次のダイアログボックスに進んでください。

5. 拡張解除のために選択したターゲットサーバと Oracle リソース階層を確認する情報ボックスが表示されます。

   **[Unextend]** をクリックしてください。

6. Oracle リソースが正常に拡張解除されたことを示す別の情報ボックスが表示されます。



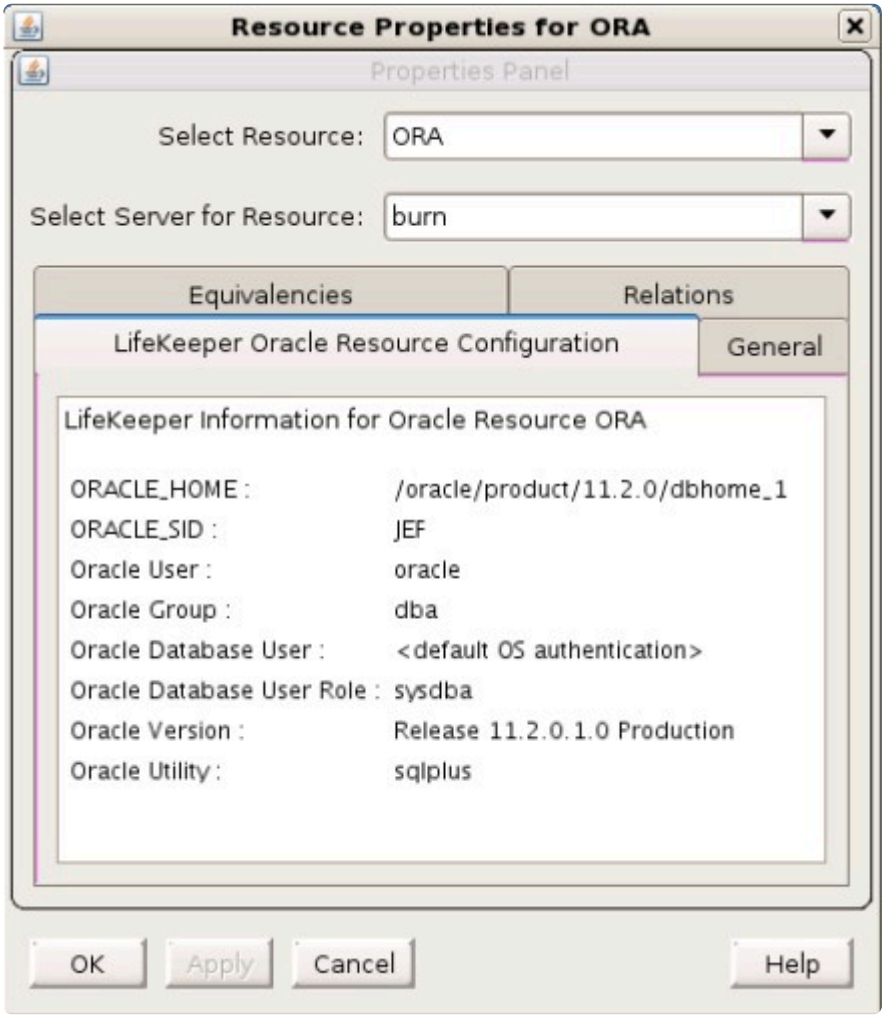

7. **[Done]** をクリックして、 **[Unextend Resource Hierarchy]** メニューの選択を終了します。

# 6.11.3.5. Oracle 設定の参照

**[Resource Properties]** ダイアログは **[Edit]** メニューもしくはリソースコンテキストメニューから参照可能です。このダイアログには、サーバ上の特定のリソースに対するプロパティが表示されます。 **[Edit]** メニューからアクセスした場合は、リソースおよびサーバを選択することができます。リソースコンテキストメニューからアクセスする場合はサーバを選択することができます。

**[Configuration]** タブからは以下の Oracle 設定を参照することができます。

- ORACLE_HOME
- ORACLE_SID
- Oracle User
- Oracle Group
- Oracle Database User
- Oracle Database User Role
- Oracle Version
- Oracle Utility



# 6.11.3.6. Oracle データベースアカウントに対するユーザ名 / パスワードの変更

階層作成後、以下のいずれかの手順でユーザ名およびパスワードを変更してください。

$ORACLE_HOME が共有（もしくは複製）ストレージ上にある場合（一般的なアクティブ-パッシブ構成）：

1.
2. Oracle データベースインスタンスが動作可能なシステム上で LifeKeeper の設定ファイル /etc/default/LifeKeeper を編集し、以下の行をファイルに追加してください。

   ```
   LK_ORA_NICE=1
   ```

   Oracle リソースが定義されているクラスタ内の各システムで同様の処理を行ってください。

3. sqlplusを使用して、Oracle ユーザのパスワードを変更してください。

   ```
   SQL> ALTER USER {username} IDENTIFIED BY {newpassword};
   ```

4. LifeKeeper GUI から、Oracle データベースリソース階層を右クリックし、 **[Change Username / Password]** を選択してください。

5. **[Username]** を入力し、 **[Next]** を選択してください。

6. **[Password]** を入力し、 **[Next]** を選択してください。

7. **[database user role]** を選択し、 **[Apply]** をクリックしてください。認証後、ユーザ名およびパスワードが更新されます。

8. **[Done]** を選択してください。

9. すべてのクラスタノード上で LifeKeeper の設定ファイルを編集し、以下の変更を行ってください。

   ```
   LK_ORA_NICE=0
   ```

$ORACLE_HOME がローカルストレージ上にあり、クラスタの各ノードがそれらの $ORACLE_HOME のコピーを保有している場合（一般的なアクティブ-アクティブ構成）：

1.
2. Oracle データベースインスタンスが動作可能なシステム上で LifeKeeper の設定ファイル /etc/default/LifeKeeper を編集し、以下の行をファイルに追加してください。

   ```
   LK_ORA_NICE=1
   ```

   Oracle リソースが定義されているクラスタ内の各システムで同様の処理を行ってください。

3. sqlplus を使用して、Oracle ユーザのパスワードを変更してください。

   ```
   SQL> ALTER USER {username} IDENTIFIED BY {newpassword};
   ```

4. LifeKeeper GUI から、Oracle データベースリソース階層を右クリックし、 **[Change Username / Password]** を選択してください。

5. sys ユーザのように一時的に代用するユーザ名を **[Username]** に入力し、 **[Next]** を選択してください。

6. **[Password]** を入力し、 **[Next]** を選択してください。

7. **[database user role]** を選択し、 **[Apply]** をクリックしてください。認証後、ユーザ名およびパスワードが更新されます。

8. **[Done]** を選択してください。

9. バックアップシステムで Oracle データベースリソースを「In Service」にしてください。

10. バックアップシステムでデータベースが起動したら、sqlplus を使用して、Oracle アカウントパスワードを変更してください。

    ```
    SQL> ALTER USER {username} IDENTIFIED BY {newpassword};
    ```

    このパスワードを変更する場合、ステップ 2 で設定した新しいパスワードを使用します。この手順は $ORACLE_HOME のセキュリティトークンをリセットします。

11. データベースをクラスタ内の各ノード上で「In Service」にし、手順 8 を繰り返してください。

12. すべてのクラスタノード上でパスワードが変更されたら、適切なノードで Oracle データベースを「In Service」に戻してください。

13. LifeKeeper GUI から、Oracle データベースリソース階層を右クリックし、 **[Change Username / Password]** を選択してください。

14. **[Username]** を入力し、 **[Next]** を選択してください。

15. **[Password]** を入力し、**Next** を選択してください。

16. **[database user role]** を選択し、 **[Apply]** をクリックしてください。認証後、ユーザ名およびパスワードが更新されます。

17. **[Done]** を選択してください。

18. すべてのクラスタノード上で LifeKeeper の設定ファイルを編集し、以下の変更を行ってください。

    ```
    LK_ORA_NICE=0
    ```

# 6.11.3.7. リソース階層のテスト

まず手動スイッチオーバを実施することで、Oracle リソース階層をテストできます。これにより、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートします。

## GUI による手動スイッチオーバの実行

LifeKeeper GUI メニューで [Edit] の [Resource] から [In Service] を選択すると、手動でのスイッチオーバを開始できます。例えば、バックアップサーバで In Service 要求を実行すると、アプリケーション階層はバックアップサーバで In Service になり、プライマリサーバでは out of service になります。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバに変わります。

Out of Service 要求を実行すると、アプリケーションは他のサーバで In Service にならずに、out of service になります。

## リカバリ動作

プライマリサーバに障害が発生すると、Oracle Recovery Kit ソフトウェアは以下の作業を実行します。

- バックアップサーバのいずれかの物理ネットワークインターフェースで論理インターフェースを In Service にすることによって、Oracle をバックアップサーバで In Service にします。（注記: これは、IP リソースインスタンスが Oracle 階層の子として定義されている場合にのみ実行されます。）
- ファイルシステムをそのサーバの共有ディスク上にマウントします。
- Oracle に関連するデーモンプロセスを開始します。

リカバリ後はセッションコンテキストが失われているため、Oracle ユーザは最初の接続に使用したのとまったく同じ手順で再接続する必要があります。

# 6.11.4. トラブルシューティング

メッセージカタログ は、SIOS Protection Suite for Linux の使用時に検出されるすべてのメッセージのリストを示すとともに、適宜、エラーの原因とエラー状態を解決するために必要な処置の説明を提供します。このリストから受け取ったエラーコードを探すことができます。また、Oracle Recovery Kit の使用時に検出されるすべてのメッセージがリストされた Oracle Kit メッセージカタログ または Oracle Listener メッセージカタログ を直接参照してもかまいません。

- Oracle 既知の問題と制限

# 6.11.4.1. Oracle 既知の問題と制限

## 制御ファイルのスイッチオーバの失敗

$ORACLE_HOME ディレクトリが回復されない場合、データベース制御ファイルが正しくセットアップされていない可能性があります。自動スイッチオーバの場合は、データベースの作成中に制御ファイルを共有デバイス上に設定する必要があります。制御ファイルを別々のサーバに置く場合は、変更が必要になったときに両方のサーバを手動でアップデートする必要があります。

## 出力の打ち切り

Oracle の一部のバージョンでは、show parameters control_files を sqldba モードで実行するときに、出力が打ち切られることがあります。使用している Oracle のバージョンがそのような動作をする場合は、以下の点を確認してください。

- controlfile パラメータ。パラメータが、*$Oracle_HOME/dbs/init#SID.ora* ファイルにあることを確認します。
- controlfile デバイス。controlfile デバイスが、新規行ではなく、継続行で指定され、各デバイスがコンマで区切られていることを確認します。

Oracle 関連のデバイスが正しく設定されていない場合は、LifeKeeper Application 管理の下で利用可能なファイルシステムアプリケーションを使用して、手動で設定できます。

## 共有ドライブ上に配置された Flash Recovery Destination

構成のセクションでも記載されているように、Flash Recovery destination が、共有ドライブ上に配置されていることが重要です。Flash Recovery Area がどこに配置されているかを確認する場合は、SYSDBA として以下のクエリーを発行してください。

```
SQL> SELECT substr(Name,1,30) Name,
  (SPACE_LIMIT/1024/1024/1024) Space_Limit_GB,
       SPACE_USED/1024/1024/1024 Space_Used_GB,
       SPACE_RECLAIMABLE, NUMBER_OF_FILES
FROM V$RECOVERY_FILE_DEST;

NAME                SPACE_LIMIT_GB SPACE_USED_GB SPACE_RECLAIMABLE
------------------- -------------- ------------- -----------------
NUMBER_OF_FILES
---------------
/U01/flash_recovery_area    3.76171875    .156448364           0
              4
```

下記は、このタスクを完了させるために、 *$ORACLE_HOME/dbs/spfile<sid>* に変更を行う場合の例です。

```
SQL> ALTER SYSTEM SET DB_RECOVERY_FILE_DEST='/oracledb/oracle/flash_recovery_a
rea' scope=both;
System altered.
SQL> show parameter DB_RECOVERY_FILE_DEST;
NAME                          TYPE        VALUE
------------------------------------------------------------------------
db_recovery_file_dest         string      /oracledb/oracle/flash_recovery_area
db_recovery_file_dest_size    big integer 2G
SQL> commit;
```

## データベースに接続できない場合のフェイルオーバの回避

リソースのヘルスチェックの実行時に、Oracle ARK は実行中のデータベースプロセスをチェックし、データベースへの接続を試行します。最大許容接続に達したことで生じるヘルスチェック障害を回避するには、 */etc/default/LifeKeeper* で以下を設定します。

```
LK_ORA_NICE=1
```

**注記:** LK_ORA_NICE を設定することで、非稼働データベースによって生じる他のタイプの接続エラーをマスクできます。この設定を使用する場合は、注意してください。

## *oratab* の非伝統的な場所

デフォルトでは、Oracle ARK は *oratab* ファイルをまず */etc* で検索した後、 */var/opt/oracle* でこのファイルを検索します。このいずれかのデフォルトの場所に *oratab* ファイルがない場合は、 ORACLE_ORATABLOC を */etc/default/LifeKeeper* で *oratab.* が含まれるディレクトリに設定する必要があります。

## サポートされていない NFS バージョン 4

Oracle Recovery Kit は共有データベースストレージで NFSv3 をサポートしています。現時点では、NFSv4 のファイルロックメカニズムにより NFSv4 はサポートされていません。

## ローカルストレージでの Oracle バイナリのインストール後の Oracle データベース階層の作成

各 LifeKeeper クラスタノードのローカルストレージに Oracle バイナリ *（$ORACLE_BASE）* をインストールした場合、LifeKeeper GUI で Oracle データベース階層の作成時に以下のようなメッセージが表示されます。

```
BEGIN create of <SID> on server <server1（サーバ server1 でSID の作成を開
始）>
Creating resource instance <SID> on server <server1（サーバ server1 でリソー
スインスタンス SID を作成）>
```

Setting resource state for <SID> on server <server1> to "ISP". (サーバ server1 で SID のリソース状態を「ISP」に設定)

ORACLE_HOME「/opt/oracle/app/oracle/product/11.2.0/dbhome_1」が共有ファイルシステムにありません。ORACLE_HOME ディレクトリおよび関連ファイルがすべてのサーバで同じであるか確認してください。詳細については、LifeKeeper Oracle Recovery Kit のドキュメントを参照してください。
Creating dependent file system resource "/u00" on <server1>. (server1 で依存ファイルシステムリソース「/u00」を作成)
Creating dependency between Oracle database "SID (SID)" and the dependent resource "/u00" on <server1>. (server1 で Oracle データベース「SID (SID) 」と依存リソース「/u00」間の依存関係を作成)
Creating dependency between Oracle database "SID (SID)" and the listener resource "LSNR.LISTENER" on <server1>. (server1 で Oracle データベース「SID (SID) とリスナーリソース「LSNR.LISTENER」間の依存関係を作成)
Performing in-service of new Oracle resource tag=< SID > on <server1>. (server1 で新しい Oracle リソースタグ = SID の In Service を実行)
END successful create of <SID> on server <server1 (サーバ server1 で SID の作成が正常終了) >

また、LifeKeeper ログにも似たような警告が示されます。この警告が不要で、階層を拡張して作業を続行した後、データベースリソースを別のノードで In Service にしようとした場合、[LifeKeeper GUI] ダイアログに以下のようなメッセージが表示されます。

Put resource "OST" in-service (リソース「OST」を In Service にする)
BEGIN restore of "OST" on server "cae-qa-v39" (サーバ「cae-qa-v39」で「OST」のリストアを開始)
Begin the "start [ start.normal ]" of the database "OST" on "cae-qa-v39". (「cae-qa-v39」でデータベース「OST」の「start[start.normal]」を開始)
The "start [ start.normal ]" attempt of the database "OST" appears to have failed on "cae-qa-v39". (「cae-qa-v39」でデータベース「OST」の「start [start.normal]」試行が失敗したようです)
ORA-01078: failure in processing system parameters (ORA-01078: システムパラメータの処理障害)
LRM-00109: could not open parameter file ‘/opt/oracle/app/oracle/product/11.2.0/dbhome_1/dbs/initOST.ora’ (LRM-00109: パラメータファイル「/opt/oracle/app/oracle/product/11.2.0/dbhome_1/dbs/initOST.ora」を開くことができませんでした)
Begin the "start [ start.force ]" of the database "OST" on "cae-qa-v39". (「cae-qa-v39」でデータベース「OST」の「start[start.force]」を開始)
The "start [ start.force ]" attempt of the database "OST" appears to have failed on "cae-qa-v39". (「cae-qa-v39」でデータベース「OST」の「start [start.force]」試行が失敗したようです)
select ‘alter database datafile ‘||file#||’ end backup;’ from v\$backup where status = ‘ACTIVE’ (ステータス =「ACTIVE」の v\$backup から「alter

database datafile '||file#||' end backup;」を選択してください)

また、ダイアログボックスに以下のようなメッセージが表示されることもあります。

Put resource "OST" in-service (リソース「OST」を In Service にする)
BEGIN restore of "OST" on server "cae-qa-v39" (サーバ「cae-qa-v39」で「OST」のリストアを開始)
Begin the "start [ start.normal ]" of the database "OST" on "cae-qa-v39". (「cae-qa-v39」でデータベース「OST」の「start[start.normal]」を開始)
122528 Initial inspection of "COMMAND" failed, verifying failure or success of received output.」(122528 「COMMAND」の初期点検が失敗し、受信出力の成否を検証できませんでした)
Logon failed with "" for "OST" on "cae-qa-v39". (「cae-qa-v39」の「OST」で「」でログオンが失敗しました) Please check username/password and privileges. (ユーザ名 / パスワードおよび権限をチェックしてください)
The "start [ start.normal ]" attempt of the database "OST" appears to have failed on "cae-qa-v39". (「cae-qa-v39」でデータベース「OST」の「start [start.normal]」試行が失敗したようです)
ERROR: (エラー: )

ORA-01031: insufficient privileges (ORA-01031: 不十分な権限)

Enter password: (パスワードを入力してください: )

ERROR: (エラー: )

ORA-01005: null password given; logon denied (ORA-01005: null パスワードが返されました。ログオンが拒否されます)

この問題を解決するには、***$ORACLE_BASE/admin*** をデータベースインスタンスが作成されたプライマリシステムからバックアップシステム（階層が拡張された場所）の ***$ORACLE_BASE/admin*** にコピーしてください。また、このディレクトリの所有権を Oracle ユーザ名および Oracle グループ（通常 **oracle:oinstall**）に変更してください。

また、すべての ****{$ORACLE_SID}**** (この例では ***OST***) ファイルをプライマリファイルの ***$ORACLE_BASE/product/11.2.0/dbhome_1/dbs***からバックアップシステムの ***ORACLE_BASE/product/11.2.0/dbhome_1/dbs*** にコピーしてください。

例えば、これらはプライマリシステムからバックアップへコピーされたファイルであり、ORACLE SID が OST であるとします。

-rw-r—— 1 oracle oinstall 1544 2012-05-09 11:02 hc_OST.dat
-rw-r—— 1 oracle oinstall 24 2012-01-31 10:22 lkOST
-rw-r—— 1 oracle oinstall 1536 2012-03-05 09:02 orapwOST
-rw-r—— 1 oracle oinstall 2560 2012-05-09 10:58 spfileOST.ora

SID が ORA01 のもう 1 つの例では、以下のファイルがコピーされます。

```
-rw-r--- 1 oracle oinstall 1536 2010-09-08 18:25 orapwORA01
-rw-r--- 1 oracle oinstall 24 2010-09-08 18:25 lkORA01
-rw-r--- 1 oracle oinstall 2560 2010-09-08 18:30 spfileORA01.ora
-rw-r--- 1 oracle oinstall 1544 2010-09-08 18:30
hc_ORA01.dat
```

およびディレクトリ

```
peshm_ORA01_0/:
```

## フェイルオーバ後も Oracle リスナーがプライマリサーバ上で稼動したままになる

ネットワーク障害に起因した Oracle アプリケーションのフェイルオーバ後に、プライマリサーバ上でリスナー・プロセスが稼動し続けることがあります。

# 6.11.4.1.1. データベース作成時の問題

| | |
|---|---|
| 問題: | dbca を使用してデータベースを作成したときに、次のメッセージが表示された。「ORA-00439 feature not enabled: string」(ORA-00439 この機能は使用できません：文字列) |
| 処置: | 環境変数 $ORACLE_SID の値を調べて、作成している SID と同じか確認してください。 |
| 問題: | スクリプトからデータベースを作成したときに、次のメッセージが表示された。「ORA-01092 ORACLE instance terminated.Disconnection forced」(ORA-01092 Oracle インスタンスが終了しました。強制的に切断されます) |
| 処置: | bdump ディレクトリにある警告を参照してください。「ORA-12714 invalid national character set specified」(ORA-12714 無効な各国語キャラクタセットが指定されました) というメッセージが表示される場合には、環境変数 $ORA_NLS33 の値を調べて、正しいロケーションに設定されているか確認してください。 |
| 問題: | dbca で生成されたスクリプトでデータベース作成中に問題が発生したら、次の手順を実行してください。 |
| 処置: | 1. 次のディレクトリがなければ、必ず作成してください<br><br>*bdump*<br>*cdump*<br>*udump*<br>*<oracle data base directory>/oradata sid*<br>*<oracle data base directory>/dbs*<br>*<oracle data base directory>/admin/<SID>*<br><br>*bdump* ディレクトリと *udump* ディレクトリのパスを調べる場合は、初期設定ファイル（*init<SID>.ora*）を確認してください。<br><br>2. ファイル *$ORACLE_HOME/dbs/orapw* が存在するか確認してください。存在しない場合には、orapwd ユーティリティを使用して作成してください。 |

# 6.11.4.1.2. データベース起動時の問題

| | |
|---|---|
| 問題: | sqlplus を使用してデータベースを起動中に、次のメッセージが表示された。「ORA-03113 end-of-file on communication channel」(ORA-03113 通信チャネルでファイルが終わりになりました) |
| 処置:: | 初期設定ファイル (*init<SID>ora*) とパスワードファイル (*orapw<SID>*) がディレクトリ *$ORACLE_HOME/dbs* にあるか確認してください。 |
| 問題: | データベースの起動時に、次のメッセージが表示された。「ORA-01092 ORACLE instance terminated.Disconnection forced」(ORA-01092 Oracle インスタンスが終了しました。強制的に切断されます) |
| 処置:: | *bdump* ディレクトリにある警告を参照してください。「ORA-12701 CREATE DATABASE character set is not known」(ORA-12701 CREATE DATABASE で指定したキャラクタセットが不明です) というメッセージが表示される場合には、環境変数 $ORA_NLS33 の値を調べて、正しいロケーションに設定されているか確認してください。 |

# 6.11.4.1.3. LifeKeeper の inqfail エラーログ

フェイルオーバーに続いて、inqfail エラーが LifeKeeper エラーログに表示された場合は、filesystemio の設定を変更する必要があります。

**注記:** ディスク ID およびサーバ名はそれぞれの設定で異なります。

この問題を解決するには、filesystemio=「SETALL」 を filesystemio=「ASYNCH」へ設定変更する必要があります。

この設定を探すには、以下の SQL コマンドでオプションを確認してください。

1. SQL> show parameter filesystemio;
2. 以下のコマンドを使用して設定を変更してください。

   SQL> alter system set filesystemio_options=<XXXXXXX> scope=spfile;

   <XXXXXXX> は以下に設定できます。

   <XXXXXXX> = {none | setall | direction | asynch}

   NONE – 最適化なし

   ASYNC - 非同期 I/O を有効化

   DIRECTIO – ダイレクト I/O を有効化

   SETALL – 使用可能なすべての機能を有効化

   **重要:** パラメータの再設定の後、Oracle を再起動する必要があります。

# 6.11.5. 付録

Raw I/O を使用するための Oracle のセットアップ

階層作成後のテーブルスペースの追加

複数のリソースに対する共有 Oracle Listener の作成

Listener 保護レベルのアップデート

Listener リカバリレベルのアップデート

保護された Listener のアップデート

# 6.11.5.1. Raw I/O を使用するための Oracle のセットアップ

以下の手順に従って、ファイルの代わりに共有 Raw I/O デバイスを使用する Oracle データベースを作成します。

1. データベースを作成するために最低限必要なファイルの数とサイズを決めます。これには、制御ファイル、テーブルスペース、REDO ログが含まれます。それらのアイテムの一部をファイルとし、それ以外を Raw I/O デバイスとする混合セットアップも可能です。すべての Raw I/O デバイスが共有ディスクパーティションを使用する必要があります。

2. 必要な数の Raw I/O デバイスを使用して、Raw I/O セットアップを作成します。

   a. Oracle データベースを作成するときに指定するサイズ以上のサイズで RAW デバイスを作成します。

   b. `raw` コマンドを使用して、システム初期化ファイル（つまり、`boot.local` または `rc.local`）にRAW デバイスマッピングを作成します。どの RAW デバイスがどの Oracle ファイルを表すのかを識別するために、意味のあるコメントを追加してください。これは、システムのリブートを行う場合に、マッピングを再確立できるようにするために必要です。Raw I/O デバイスが LifeKeeper の保護下に置かれたら、これらのマッピングは手動でファイルから削除してください。

3. 以下のコマンドを使用して、RAW デバイスを Oracle データベースに書き込み可能にします。
   ```
   chown oracle:dba /dev/raw[0-9]*
   ```
   ここで、オーナおよびグループは、Oracle インスタンスの構成に固有です。

4. マッピングを含むファイルを実行して、RAW デバイスの設定をアクティブにします。

5. データベース作成スクリプトがすでに存在している場合は、ステップ 6 へ進みます。データベース作成スクリプトがない場合は、Oracle Java GUI ツールの 1 つ（`dbassist` または `dbca`）を使用して作成できます。どちらのツールを使用する場合も「**Save As Script**」を選択してください。データベースの作成を選択しないでください。

   **注記:**

   - `dbca` で、スクリプトを生成するには、「**New Database**」テンプレートを選択する必要があります。必要であれば、ファイル名を共有デバイスに変更し、構成の値を調整します。
   - この時点で DB 作成プロセスを開始しないでください。`dbassist` ツールは、各テーブルスペースに指定されたファイルがすでに存在しているかどうかをチェックし、ファイルが存在している場合は処理を行いません。`dbca` ツールはファイルを上書きするかどうかを確認するメッセージを表示しますが、RAW デバイスでは失敗します。どちらの場合も、これらのツールからRAW デバイスを直接使用することはできません。

6. データベース作成スクリプト（既存のもの、もしくは dbassist または dbca で作成したもの）を編集する必要があります。該当するファイル名（パスを含む）を、Raw I/O デバイスのフルパス名に置き換えます。対象となるファイルには（最低限）データベース作成ファイル（CREATE DATABASE コマンド用）とテーブルスペース作成ファイル（CREATE TABLESPACE コマンド用）があります。選択した dbassist または dbca のオプションによっては、さらに別のファイルを編集しなければならない場合があります。また、必要に応じて、初期化ファイルを編集し、制御ファイルを Raw I/O デバイスに変更します。初期化ファイルは、作成スクリプトが存在するディレクトリにあります。データファイルの場合、編集結果は以下のようになります。

```
. . .

CREATE DATABASE “LK”

     maxdatafiles 254

     maxinstances 8

     maxlogfiles 32

     character set US7ASCII

     national character set US7ASCII

DATAFILE ‘/dev/raw/raw1’ SIZE 260M AUTOEXTEND ON NEXT 10240K

logfile ‘/ora/LK/redo01.log’ SIZE 500K,

       ‘/ora/LK/redo02.log’ SIZE 500K,

       ‘/ora/LK/redo03.log’ SIZE 500K;

. . .
```

   Raw I/O デバイスのサイズは、格納するデータのために Oracle が必要とする最低限のサイズにします。

7. ここで、ステップ 5 で作成したスクリプトを実行してデータベースを作成します。

8. 作成ログをチェックして、データベースまたはテーブルスペースのエラーが発生していないか確認します。

9. 作成スクリプトでデータベースを作成中に問題が発生するか、後でRAW デバイスにテーブルスペースを追加する場合、適切なツール（dbassist または dbca）でデータベースを作成する必要があります。その後、sql ユーティリティから次のようなコマンドを実行して、Raw I/O デバイスデータファイルを追加します。

```
tablespace RAWTS
```

```
DATAFILE '/dev/raw/raw217' SIZE 50M REUSE
DEFAULT STORAGE (INITIAL 50K NEXT 50K
MINEXTENTS 1 MAXEXTENTS 4) ONLINE
```

10. udev の規則を追加して、RAW デバイスの権限および所有権をリブート、切り替え、フェイルオーバがあっても変わらないようにします。

    Linux 2.6 カーネルでは、udev システムは、RAW デバイスなどの特殊ファイルの作成をカーネルが制御するためのデフォルトの方法です。RAW デバイスを Oracle で使用する場合、固有の所有権および権限の設定が必要です。このような固有の設定は、カーネルのデフォルトの設定と競合します。固有の設定をアドレス指定するには、udev の規則を使用して、LifeKeeper が保護する Oracle SID で使用する Oracle ユーザおよび Oracle グループにRAW デバイスの所有権を設定する必要があります。

    **注記:** 一部の OS ディストリビューションでは、デバイスを作成するための規則とデバイス権限を設定するための規則を分ける必要があります。詳細については、OS ディストリビューションの udev ドキュメントを参照してください。

    OS ディストリビューションで動作する udev 規則の例を下記に示します。

    ```
    KERNEL=="raw10", RUN+="/bin/chown oracle:oinstall /dev/raw/raw10"

    KERNEL=="raw[3-5]*", OWNER="oracle", GROUP="oinstall", MODE="660"
    ```

    リソース階層を In Service にする前に、作成した udev の規則を LifeKeeper クラスタのすべてのノードに適用する必要があります。

# 6.11.5.1.1. 階層作成後のテーブルスペースの追加

LifeKeeper で Oracle 階層を作成した後に Raw I/O デバイスでテーブルスペースを追加する場合は、GUI を使用して LifeKeeper Raw I/O 階層を作成し、Oracle リソース（親）と Raw I/O リソース（子）の間に依存関係を手動で作成する必要があります。

# 6.11.5.2. 複数のリソースに対する共有 Oracle Listener の作成

ご利用のシステム構成で次のいずれかが当てはまる場合、Oracle Listener を作成できます。

- 複数の Oracle SID に対して、複数の Listener が定義されている。
- Oracle Listener が設定において重要なコンポーネントである。
- 複数の Oracle SID に対して 1 つの Listener が定義されている。

> ✻ 注記: 複数の Listener を SPS で保護する場合、LISTENER/SID_LIST_LISTENER スタンザのコレクションはすべて一意である必要があります。（例：11g インストールの listener.ora ファイルにはスタンザ LISTENER_11G_1/SID_LIST_LISTENER_11G_1 が含まれます。）

このプロセスにより、LifeKeeper 内の Listener を保護し、さまざまな Listener と SID の組み合わせに対応できるようになります。

複数のリソースに対して 1 つの共有 Oracle Listener を作成する場合は、下記の手順に従ってください。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択します。

   **重要:** リソースを作成するときは、Oracle アプリケーションが実行されている必要があります。

2. ダイアログボックスが表示され、クラスタ内にインストール済みの認識されているすべての Recovery Kit がドロップダウンリストボックスに表示されます。ドロップダウンリストから **[Oracle Database Listener]** を選択します。 **[Next]** をクリックして次のダイアログボックスに進んでください。

3. 次の表で説明している情報を入力します。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、以前に入力した情報を訂正する必要がある場合に特に役立ちます。階層作成手順のどの段階でも、 **[Cancel]** をクリックすると、作成プロセス全体がキャンセルされます。

| フィールド | ヒント |
|---|---|
| スイッチバックタイプ | intelligent または automatic を選択します。これは、サーバがフェイルオーバした後、どのように Listener リソースがサーバにスイッチバックされるかを指示します。データレプリケーションを使用する場合は、スイッチバックタイプは intelligent に選択してください。<br><br>**注記:** スイッチバックタイプは Listener リソースによって使用される依存リソース（IP およびボリュームリソース）と一致している必要があります。そうでない場合は、作成に失敗します。 |
| サーバ | 階層を作成したいサーバを選択してください。 |

| | |
|---|---|
| Listener 設定ファイルのパス | Oracle Listener 設定ファイルへのフルパスを選択してください。 |
| Listener 名 | このリソースインスタンスで保護する Oracle Listener の名前を選択してください。 |
| Listener の実行パス | Oracle Listener プログラムの実行パスを選択します。このパスは Oracle Listener を起動･停止･監視･回復する際に必要となります。 |
| Listener の保護レベル | 以下のレベルの 1 つを選択してください。<br><br>Full Control（開始、停止、監視および回復）<br><br>Intermediate Control（開始、監視および回復）<br><br>Minimal Control（開始および監視のみ） |
| Listener のリカバリレベル | 指定された listener に対するリカバリレベルを選択してください。<br><br>Standard (On) - 標準の LifeKeeper リカバリを有効にします。Standard (On) が選択されている場合は、すべての listener 障害がローカルで発生し、必要であれば、有効なバックアップサーバへのフェイルオーバを行います。<br><br>Optional (off) - オプションの LifeKeeper リカバリを有効にします。Optional (Off) を選択した場合は、すべての listener 障害がローカルで発生しますが、有効なバックアップサーバへのフェイルオーバは行いません。<br><br>**注記** ：どちらのリカバリレベルもlistener障害が発生した際に、ローカルリカバリは実施しますが、フェイルオーバ実施の有無に違いがあります。 |
| IP アドレス名 | このリソース階層の依存関係として保護される IP アドレスのリソース名を選択してください。選択された listener に関連した IP アドレスは選択リストに表示されます。IP リソースが設定に必要がない場合は [None] を選択してください。 |
| Listener タグ | サーバ上のリソースに対するユニークな名前を入力してください。名前に対する有効な文字は、アルファベット、数字および特殊記号 - _ . / になります。 |

4. **[Create]** ボタンを選択し、階層の作成を開始します。インフォメーションボックスが表示され、LifeKeeper がデータベース listener リソース階層を作成するために提供した有効なデータを確認します。LifeKeeper が問題を検知した場合は、インフォメーションボックスにエラーが表示されます。認証に成功すると、リソースが作成されます。

5. ***[Pre-Extend Wizard]*** ダイアログはリソース階層の作成が完了した場合に表示され、以下の情報を入力することになります。 **Extend** のオペレーションに慣れていない場合は、それぞれのダイアログで選択をした後 **[Next]** を選択してください。LifeKeeper の **[Extend Resource Hierarchy]** 作業

のデフォルト値が分かっていて、入力と確認を省略する場合は、 **[AcceptDefaults]** をクリックします。

| フィールド | ヒント |
|---|---|
| ターゲットサーバ | 階層が拡張されるターゲットサーバを選択してください。リソースが少なくとも 1 つのサーバへ拡張される前にキャンセルを選択した場合は、LifeKeeper は階層のアプリケーションに対して保護をしません。 |
| スイッチバックタイプ | バックアップサーバへフェイルオーバした後、 in service にする場合に、どのように Oracle Listener インスタンスがこのサーバにスイッチバックするかを指示します。intelligent または automatic を選択できます。スイッチバックタイプは、必要な場合 [Resource Properties] ダイアログボックスの [General] タブで後から変更できます。<br>注記: スイッチバックストラテジーは、Oracle Listener リソースによって使用される依存リソースのものと一致している必要があります。 |
| テンプレートの優先順位 | このフィールドは、作成時に拡張しなかった場合に表示されます。<br>リソースに対するカスケーディングフェイルオーバシーケンスにおける、テンプレートサーバの優先順位を 1 から 999 までの間の値を入力してください。低い数値は高い優先順位となります。<br>LifeKeeper は、階層が作成されたサーバの値を「1」に割り当てています。2 つのサーバは与えられたリソースに対して同じ優先順位をもつことはできません。 |
| ターゲットの優先順位 | リソースに対するカスケーディングフェイルオーバシーケンスにおける、ターゲットサーバの優先順位を 1 から 999 までの間の値を入力してください。低い数値は高い優先順位となります。<br>LifeKeeper は、階層が拡張される最初のサーバのデフォルト値を 10 にすることを推奨します。 |

6. pre-extend チェックに成功したというメッセージを受けた後、 [Next] をクリックし、以下の情報を入力してください。

| フィールド | ヒント |
|---|---|
| Listener 設定ファイルのパス | Oracle Listener 設定ファイルに対するフルパスを選択してください。 |
| Listener の実行パス | Oracle Listener プログラムの実行パスを選択します。このパスは Oracle Listener を起動･停止･監視･回復する際に必要となります。 |
| Listener | このフィールドはプライマリサーバ上の新しい Oracle Listener リソースに対してユニークな名前 |

| | |
|---|---|
| タグ | で自動的に割り当てられます。デフォルトのネーミングパターンが表示されます。他のユニークな名前を入力することもできます。Listener タグに対する有効な文字は、アルファベット、数字および特殊記号 - : . / になります。 |

7. **[Extend]** をクリックします。 **[Hierarchy Integrity Verification]** ウィンドウが、 **Hierarchy Verfication Finished（階層完了）** のメッセージと共に表示されます。 **[Next Server]** または **[Finish]** をクリックしてください。

# 6.11.5.2.1. Listener 保護レベルのアップデート

1. リソース を選択後、 **[Resource]** ツールバーから ボタンを選択し、リソースの保護レベルをアップデートしてください。

2. 以下の情報を入力してください。

| フィールド | ヒント |
|---|---|
| **Listener** の保護レベル | 以下の 1 つを選択してください。<br><br>Full Control（開始、停止、監視および回復）<br><br>Intermediate Control（開始、監視および回復）<br><br>Minimal Control（開始および監視のみ） |

3. **[Update]** を選択し、現在の状態から新しい状態へと **保護レベル** を変更してください。現在の状態を変更しない場合は、 **[Cancel]** を選択してください。

# 6.11.5.2.2. Listener リカバリレベルのアップデート

1. **listener** を選択後、 **[Resource]** ツールバーから ボタンを選択し、リソースのリカバリレベルをアップデートしてください。

2. 以下の情報を入力してください。

| フィールド | ヒント |
|---|---|
| **Listener のリカバリレベル** | 指定された listener に対するリカバリレベルを選択してください。<br><br>Standard (On) - 標準の LifeKeeper リカバリを有効にします。Standard (On) が選択されている場合は、すべての listener 障害がローカルで発生し、必要であれば、有効なバックアップサーバへのフェイルオーバを行います。<br><br>Optional (off) - オプションの LifeKeeper リカバリを有効にします。Optional (Off) を選択した場合は、すべての listener 障害がローカルで発生しますが、有効なバックアップサーバへのフェイルオーバは行いません。<br><br>**注記** ：どちらのリカバリレベルもlistener障害が発生した際に、ローカルリカバリは実施しますが、フェイルオーバ実施の有無に違いがあります。 |
| **アップデートの確認** | [Update] ボタンを選択し、リカバリレベルを現在の状態から新しい状態へと変更してください。<br><br>現在の状態を変更しない場合は [Cancel] を選択してください。 |

3. **[Update]** をクリックし、リカバリレベルを現在の状態から新しい状態へと変更してください。現在の状態を変更しない場合は **[Cancel]** を選択してください。

# 6.11.5.2.3. 保護された Listener のアップデート

1. **Listener** を選択後、 **[Resource]** ツールバーから ボタンを選択し、保護された Listener のアップデートを行ってください。

2. 以下の情報を入力してください。

| フィールド | ヒント |
|---|---|
| **Listener 名** | リソースインスタンスに対して、保護する Oracle Listener の名前を選択してください。 |
| **IP アドレス名** | このリソース階層の依存関係として追加する IP アドレスのリソース名を選択してください。<br><br>この設定に対して追加する IP リソースが存在しない場合は [None] を選択してください。 |

3. **[Update]** を選択し、 **保護された Listener** および **IP の割り当て** を現在の状態から新しい状態へと変更してください。現在の状態を変更しない場合は **[Cancel]** を選択してください。

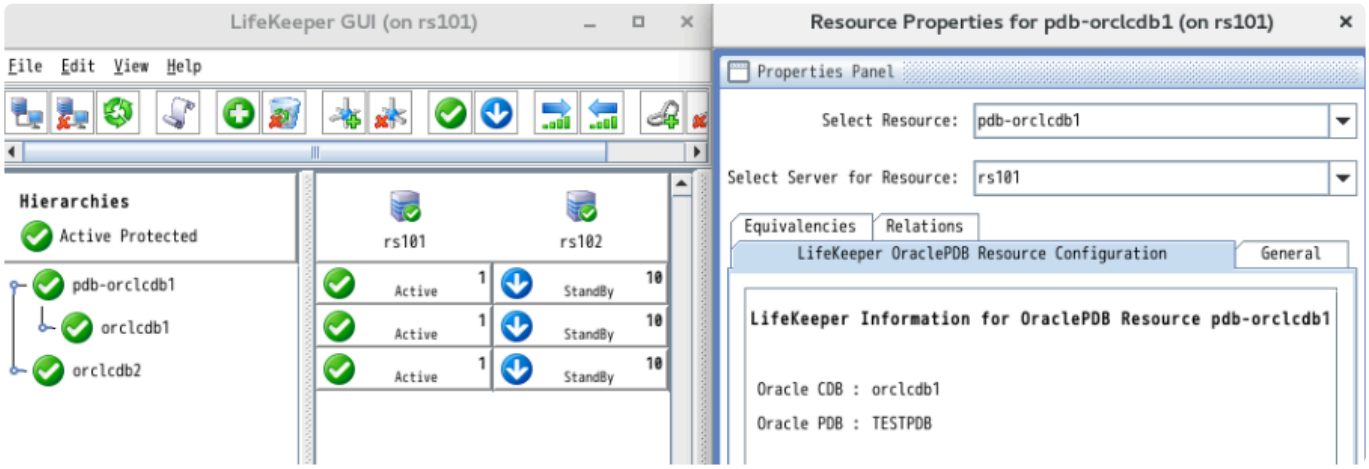
   **[Update]** を選択した場合、特定のリソースに対する保護された Listener がアップデートされている状態がダイアログに表示されます。 **[Finish]** をクリックしてください。

# 6.11.5.3. Pluggable Database の移行

LifeKeeper で保護されている、コンテナ・データベース（以下、CDB）間でプラガブル・データベース（以下、PDB）を移行する手順を紹介します。事前にデータベースのバックアップを取得しておくことをおすすめします。

## 移行手順

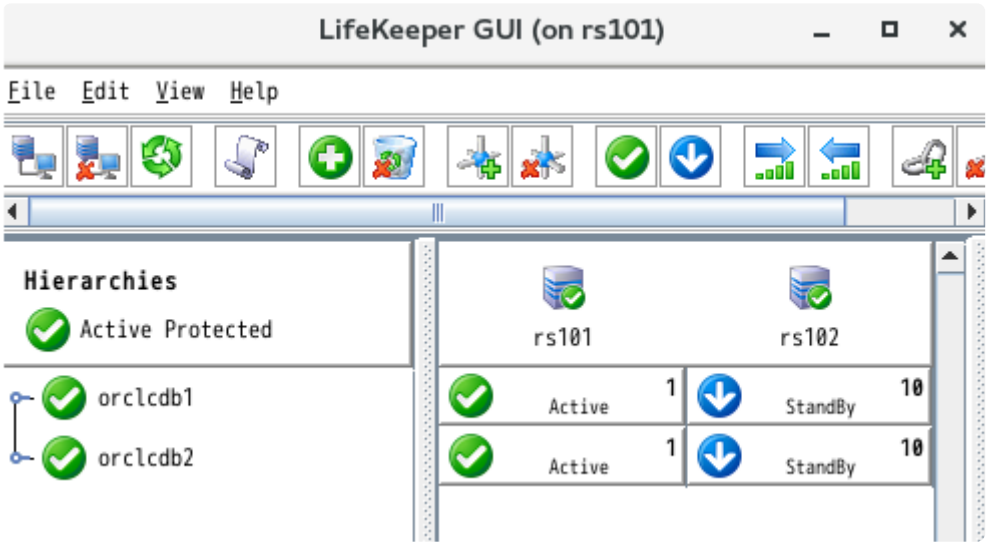
本手順では、PDB のプラグ/アンプラグを使用した方法による PDB の移行を示します。
下図の構成の場合を例に、ソース CDB （ORCLCDB1）から ターゲット CDB （ORCLCDB2）へ PDB （TESTPDB） を移行する手順です。




1. 移行前にソース CDB に接続し、対象の PDB の情報を確認します。

```
SQL> COLUMN NAME FORMAT A8

     SELECT NAME, DBID, GUID FROM V$CONTAINERS WHERE NAME='<PDB>';
```

2. Oracle PDB リソースを Out of Service にし、リソースを削除します。
   複数の PDB を保護している Oracle PDB リソースがある場合は、リソース設定 [Change Protection PDB] （保護するPDBの変更 を参照）で対象の PDB を保護対象から削除します。




3. ソース CDB に接続し対象の PDB をアンプラグします。（PDB が停止していない場合、PDB を停止します。）

```
SQL> ALTER PLUGGABLE DATABASE <PDB> UNPLUG INTO '/home/
oracle/<PDB>.xml';

SQL> DROP PLUGGABLE DATABASE <PDB>;
```

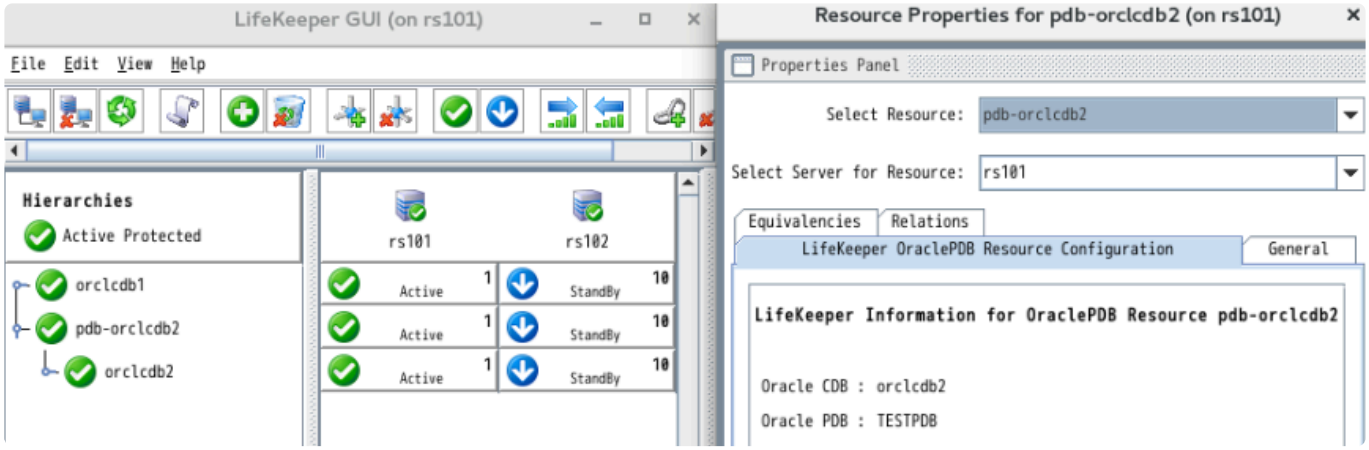
4. ターゲット CDB に接続し対象の PDB をプラグし、PDB を起動します。

```
SQL> CREATE PLUGGABLE DATABASE <PDB> USING '/home/oracle/.xml'
<COPY|NOCOPY>;

SQL> ALTER PLUGGABLE DATABASE <PDB> OPEN;
```

5. 移行後の PDB の情報が移行前と一致することを確認します。

```
SQL> COLUMN NAME FORMAT A8

     SELECT NAME, DBID, GUID FROM V$CONTAINERS WHERE NAME='<PDB>';
```

6. ターゲット CDB を指定して Oracle PDB リソースを作成します。
既に PDB を保護している Oracle PDB リソースがある場合は、リソース設定 [Change Protection PDB]（保護するPDBの変更 を参照）で対象の PDB を保護対象に追加して一つのリソースで管理することも可能です。


LifeKeeper GUI (on rs101)
File Edit View Help
Hierarchies
Active Protected
orclcdb1
pdb-orclcdb2
orclcdb2
rs101
rs102
Active 1
StandBy 10
Active 1
StandBy 10
Active 1
StandBy 10
Resource Properties for pdb-orclcdb2 (on rs101)
Properties Panel
Select Resource: pdb-orclcdb2
Select Server for Resource: rs101
Equivalencies
Relations
LifeKeeper OraclePDB Resource Configuration
General
LifeKeeper Information for OraclePDB Resource pdb-orclcdb2
Oracle CDB : orclcdb2
Oracle PDB : TESTPDB

# 6.12. PostgreSQL Recovery Kit 管理ガイド

## PostgreSQL Recovery Kit のドキュメンテーション

## 概要

SIOS Protection Suite for Linux PostgreSQL Recovery Kit は、SQLに準拠した、POSTGRESをベースとするオブジェクトリレーショナルデータベース管理システム (ORDBMS) です。PostgreSQL は誕生以来進化を続け、最も高度なオープンソースリレーショナルデータベース管理システムの 1 つになりました。

SPS for Linux PostgreSQL Recovery Kit は、LifeKeeper 内の PostgreSQL インスタンスを保護する機能を提供します。PostgreSQL ソフトウェア、LifeKeeper Core、および PostgreSQL Recovery Kit はクラスタ内の 2 台以上のサーバにインストールされます。PostgreSQL データベースインスタンスが LifeKeeper の保護下に入ったら、クライアントは LifeKeeper が保護する IP アドレスを使用してデータベースに接続します。LifeKeeper が保護する IP アドレスは別々に作成し、親 PostgreSQL リソースインスタンスと子 IP アドレスリソースの間に依存関係を手動で作成する必要があります。PostgreSQL サーバに障害が発生した場合、LifeKeeper はまずローカルサーバ上での復旧を試みます。ローカルリカバリに失敗した場合、バックアップサーバに切り替えます。

PostgreSQL リソース階層

## SIOS Protection Suite のドキュメンテーション

SIOS Technology Corp の Web サイトでは以下の SIOS Protection Suite 製品のドキュメンテーションを提供しています。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- Optional Recovery Kit ドキュメンテーション

## PostgreSQL のドキュメンテーション

*Administration Guide* 、 *User Guide* 、 *Reference Guide* を含む PostgreSQL のドキュメンテーションは以下の Web サイトで入手できます。

http://www.postgresql.org/docs

# 6.12.1. PostgreSQL リソース階層

以下の例は典型的な PostgreSQL リソース階層を示しています。

上記の例の依存関係は、以下の保護されるリソースに対応しています。

| リソース | **PostgreSQL** ソフトウェアコンポーネント |
|---|---|
| *LKIP.EXAMPLE.COM* | クライアント接続に使用される切り替え可能な IP アドレスを保護します。 |
| *var/lib/pgsql/data* | データベースデータディレクトリ (PGDATA) を保護します。 |
| *var/lib/pgsql/exec* | PostgreSQL サーバとクライアントの実行可能ファイルを保護します (実行可能ファイルが共有ファイルシステムにインストールされている場合)。 |
| *var/lib/pgsql/log* | データベースログファイルディレクトリを保護します (ログパスが共有ファイルシステム上にある場合)。 |
| *var/lib/pgsql/pg_xlog* | データベーストランザクションログディレクトリ (PGDATA/pg_xlog) を保護します。トランザクションディレクトリは Write-Ahead-Log ディレクトリとしても参照されます。 |
| *var/lib/pgsql/socket_path* | データベースソケットディレクトリを保護します(ソケットパスが共有ファイルシステム上にある場合)。 |

フェイルオーバが発生した場合、LifeKeeper はバックアップサーバでファイルシステム、IP アドレス、およびデータベースリソース(すべてのリソース依存関係を含む)を In Service の状態にします。クライアントは切断され、サーバに再接続する必要があります。まだコミットされていない SQL ステートメントはすべて再入力する必要があります

# 6.12.2. 要件

## PostgreSQL Recovery Kit の要件

LifeKeeper for Linux PostgreSQL Recovery Kit をインストールする前に、LifeKeeper 設定が次の要件を満たしている必要があります。LifeKeeper のハードウェアとソフトウェアのインストールと設定に関する具体的な手順については、SPS for Linux インストールガイド を参照してください。

### ハードウェア要件

- **サーバ** - サーバは、要件にしたがって設定する必要があります。要件は、 SPS for Linux テクニカルドキュメンテーション 、および SPS for Linux リリースノート に記載されています。

- **IP ネットワークインターフェースカード** - 各サーバは、イーサネット TCP/IP をサポートするネットワークインターフェースカードを 1 つ以上必要とします。ただし、LifeKeeper クラスタのベストプラクティスでは 2 つ以上のコミュニケーションパスが必要であることに注意してください。LAN を使用した 2 つのコミュニケーションパスは、ハートビート用に別々のサブネットに接続することをお勧めします。また、少なくともどちらか 1 つのサブネットは、プライベートネットワークとして設定します。TCP と TTY のハートビートの組み合わせもサポートされています。

### ソフトウェア要件

- **TCP/IP** ソフトウェア – LifeKeeper 環境の各サーバには TCP/IP ソフトウェアが必要です。

- **PostgreSQL** ソフトウェア – クラスタ内の全サーバに必ず同じバージョンの PostgreSQL ソフトウェアをインストールします。PostgreSQL ソフトウェアは、http://www.postgresql.org/download に記載されているいずれかのミラーサイトからダウンロードできます。

- **LifeKeeper** ソフトウェア – クラスタ内の各サーバにインストールする LifeKeeper ソフトウェアのバージョン、および、適用する LifeKeeper ソフトウェアパッチのバージョンが同一であることが絶対条件です。

- **LifeKeeper for Linux PostgreSQL Recovery Kit** – PostgreSQL Recovery Kit は、FTP ダウンロードにより SPS for Linux インストールイメージファイル (*sps.img*) で提供されます。Red Hat Package Manager (rpm) でパッケージされており、rpm を使用してインストール / アンインストールを行います。

# 6.12.3. 設定上の考慮事項

本セクションでは、PostgreSQL Recovery Kit の設定と管理を開始する前に考慮すべき事項について説明します。

PostgreSQL の保護: ベストプラクティス

DataKeeper でのミラーファイルシステムの使用

# 6.12.3.1. Postgre SQL の保護: ベストプラクティス

アクティブ / スタンバイ設定では、バックアップサーバは PostgreSQL をアクティブに実行せず、プライマリサーバに障害が発生した場合に備えてスタンバイします。アクティブ / アクティブ設定では、障害時、他のサーバに対するバックアップとして各サーバがアクティブに PostgreSQL インスタンスを実行します。以下に、アクティブ / スタンバイもしくはアクティブ / アクティブ設定で PostgreSQL リソースインスタンスを保護するときに従わなければならない具体的な要件を示します。

1. PostgreSQL の DataDir および Write-Ahead-LogPath (*PGDATA/pg_xlog*) を 1 つ以上の共有ファイルシステムにインストールしてください。DataDir および WAL-Path は、リソースインスタンスを保護するすべてのサーバ間で共有してください。

- ◦ PostgreSQL オペレーティングシステムユーザは、データディレクトリと、Write-Ahead-LogPath を含むディレクトリを所有している必要があります。

- ◦ PostgreSQL データベースは、`initdb` ユーティリティを使用して作成してください。`initdb` ユーティリティは、PostgreSQL オーナとして、-D <datadir> オプションを使用して実行する必要があります。

- ◦ デフォルトの PostgreSQL インスタンスの自動スタートアップは無効にするか、またはデフォルトの PostgreSQL インスタンスが LifeKeeper で使用するポート以外のポートで稼働するように制限する必要があります。

- ◦ LifeKeeper で保護する PostgreSQL インスタンスの自動スタートアップは無効にする必要があります。LifeKeeper は、保護するインスタンスの開始と停止を制御します。

- ◦ PostgreSQL インスタンスは、階層の作成に先だって手動で開始する必要があります。インスタンスは、`pg_ctl` ユーティリティにバックエンドオプション – o “ – p <port>” を指定して開始してください。

2. `StartupLogPath`、`SocketPath` 、`ExecutablePath` は、プライマリサーバの任意の共有ファイルシステム上か、各クラスタノードのローカルファイルシステム上に配置することができます。

- ◦ PostgreSQL オペレーティングシステムユーザはソケットパスを含むディレクトリを所有している必要があります。

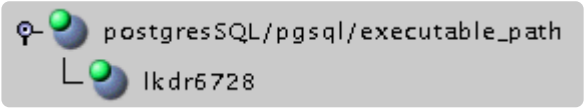
- ◦ PostgreSQL オペレーティングシステムユーザは StartupLogPath を含むディレクトリでの書き込み権限を所有している必要があります。

3. アクティブ / スタンバイもしくはアクティブ / アクティブのどちらかで複数のインスタンスを起動する場合は、各インスタンスに一意のポートおよびソケットパスを割り当ててください。

# 6.12.3.2. DataKeeper でのミラーファイルシステムの使用

PostgreSQL Recovery Kit では、SIOS DataKeeper を共有ファイルシステムとして使用できます。ミラーファイルシステムは、PostgreSQL インストールパス、ログパス、データディレクトリ、および実行可能パスに使用できます。

例えば、PostgreSQL リソースの依存ファイルシステムは、データディレクトリのファイルシステムとその依存関係を示す、以下の SDR リソースミラーと似ています。



# 6.12.4. インストール

## LifeKeeper での PostgreSQL のインストール / 設定

PostgreSQL データベースと LifeKeeper ソフトウェアのインストールと設定には以下の手順をお勧めします。各ステップは、詳細作業にリンクしています。

1. PostgreSQL ソフトウェアのインストール
2. PostgreSQL データベースの作成
3. LifeKeeper Core と PostgreSQL Recovery Kit のインストール
4. PostgreSQL リソースのための LifeKeeper 設定項目

これらの作業を実行すると、LifeKeeper リソース階層を作成して PostgreSQL データベースを保護できるようになります。

## リソース設定作業

前のセクションで説明したセットアップ作業を終了したら、PostgreSQL リソース階層を作成して拡張できるようになります。

LifeKeeper for Linux PostgreSQL Recovery Kit の設定に関して、以下の作業を行うことができます。

- Create Resource Hierarchy - PostgreSQL リソース階層を作成します。
- Delete Resource Hierarchy - PostgreSQL リソース階層を削除します。
- Extend Resource Hierarchy - PostgreSQL リソース階層をプライマリサーバからバックアップサーバに拡張します。
- Unextend Resource Hierarchy - LifeKeeper クラスタ内の 1 つのサーバから PostgreSQL リソース階層を拡張解除 (削除) します。
- Viewing PostgreSQL Configuration Settings - Resource Properties ダイアログを参照します。

LifeKeeper Core のリソース階層 (ファイルシステムや IP リソースなど) の設定方法については、SPS for Linux テクニカルドキュメンテーション の 管理作業 セクションを参照してください。

以下の作業については、 すべての Recovery Kit で同じ手順を使用する共通の作業であるため、 SPS for Linux テクニカルドキュメンテーション の 管理 セクションを参照してください。

- Create a Resource Dependency - 既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。

- Delete a Resource Dependency - リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service - リソース階層を特定のサーバで In Service にします。
- Out of Service - リソース階層を特定のサーバで Out of Service にします。
- View / Edit - 特定のサーバでリソース階層のプロパティを表示または編集します。

**注記:** このセクションでの設定作業は **[Edit]** メニューを使用して実行します。この設定作業の大半は次の方法でも実行できます。

- ツールバーから
- 状況表示の左側ペインにあるグローバルリソースを右クリックする
- 状況表示の右側ペインにあるリソースを右クリックする

右クリックの方法を使用すると、Edit メニューを使用するときに要求される情報を入力せずにすみます。

# アップグレード

PostgreSQL Recovery Kit の旧バージョンからのアップグレード

# 6.12.4.1. PostgreSQL ソフトウェアのインストール

同一のパラメータ / 設定を使用してクラスタ内の全サーバに PostgreSQL ソフトウェアをインストールする必要があります。詳細については PostgreSQL のサイト を参照してください。以下は、LifeKeeper を PostgreSQL と連携させるための追加の推奨事項および注意事項です。

- PostgreSQL クライアントソフトウェアパッケージがインストールされている必要があります。これらのパッケージには、PostgreSQL `psql` クライアントユーティリティが含まれていなければなりません。
- PostgreSQL サーバソフトウェアパッケージがインストールされている必要があります。これらのパッケージには、PostgreSQL `pg_ctl` および `initdb` ユーティリティが含まれていなければなりません。
- PostgreSQL クライアントパッケージとサーバパッケージのバージョンは、全サーバで同じでなければなりません。
- 全サーバに、PostgreSQL オペレーティングシステムユーザが以下のように存在している必要があります。
    - ° PostgreSQL オペレーティングシステムユーザを PostgreSQL ソフトウェアのインストールとサブディレクトリの所有者として指定します。
    - ° PostgreSQL オペレーティングシステムユーザには

`pg_ctl` ユーティリティを使用する権限が必要です。PostgreSQL オペレーティングシステムユーザは `pg_ctl` コマンドを使用してポストマスタインスタンスを開始および停止できなければなりません。

- ° PostgreSQL オペレーティングシステムユーザ名には英数字 (a-z,A-Z, 0-9) のみを使用する必要があります。
- ° PostgreSQL オペレーティングシステムユーザのユーザ ID とグループ ID は、全サーバで同じでなければなりません。

- PostgreSQL データベース管理者ユーザは、`psql` ユーティリティを使用した LifeKeeper クライアント接続のため、PostgreSQL のデータベース内に存在している必要があります。
    - ° また PostgreSQL データベース管理者ユーザは、データベース (

*template1* ) への接続権限を有している必要があり、またインスタンスに定義されたデータベースのリストを取得できる必要があります。

- ° この PostgreSQL データベース管理者ユーザは、システムテーブルの閲覧権限および一般的なクエリを作成できる権限を保有している必要があります。
- ° PostgreSQL データベース管理者ユーザは PostgreSQL オペレーティングシステムユーザと同じ名前を持つことができますが、異なるものです。

    - º 例:PostgreSQL オペレーティングシステムユーザ=postgres で PostgreSQL データベース管理者ユーザ=lkpostgres、もしくは、PostgreSQL オペレーティングシステムユーザ=postgres で PostgreSQL データベース管理者ユーザ=postgres。

- PostgreSQL サーバーデーモンは LifeKeeper が制御するので、システム起動時の自動起動を無効にする必要があります。

# 6.12.4.2. PostgreSQL データベースの作成

データベースを作成するには、『 PostgreSQLのサイト 』の手順に従います。さらに次の推奨事項にも留意してください。

- PostgreSQL データディレクトリは、`initdb` ユーティリティを使用し、 *-D <data dir>* オプションを指定して初期化する必要があります。`initdb` コマンドは PostgreSQL オペレーティングシステムユーザとして実行する必要があります。
- PostgreSQL インスタンスデータディレクトリは共有ファイルシステムに存在している必要があります。
- PostgreSQL トランザクションログディレクトリは共有ファイルシステム上に存在している必要があります。
- PostgreSQL データベース名には英数字のみを使用する必要があります。
- データベースを作成したら、PostgreSQL データベースインスタンスの自動スタートアップを無効にする必要があります。LifeKeeper の保護下に入ったら、LifeKeeper がデータベースの開始と停止を行います。
- PostgreSQL インスタンスは、階層の作成に先だって手動で開始する必要があります。インスタンスは、`pg_ctl` ユーティリティにバックエンドオプション *-o " -p <port>"* を指定して開始してください。

パスワードの保護を無効にする場合(インスタンスへの接続に、パスワード保護を使用しない)

- PostgreSQL データベースインスタンスがパスワード保護されない場合や、PostgreSQL の管理ユーザからローカルクライアント接続にパスワードを必要としない場合は、エントリがローカルトラスト接続を許可した状態で存在している必要があります。以下は PostgreSQL のデータベース管理ユーザのローカルクライアント接続を有効にするために *pg_hba.conf* に記述するエントリのサンプルです。

```
====================
.
.
Local all postgres trust
.
.
======================
```

パスワード保護を有効にする(インスタンスへの接続にパスワードを使用)

- パスワードで保護されるデータベースインスタンスは、リソースが保護されるクラスタの各サーバ上の *.pgpass* 認証ファイルに存在するための PostgreSQL データベース管理ユーザに対するパスワードエントリが必要になります。 *.pgpass* ファイルはパスワードを要求する各 PostgreSQL データベ

ース管理ユーザに対して有効でテスト済みのエントリを保有している必要があります。
- *.pgpass* ファイルは PostgreSQL オペレーティングシステムユーザのホームディレクトリに配置してください。ファイルへのアクセス権を制限するために適切な権限を設定してください。
- 以下は、*<hostname>:<port>:<database>:<user>:<password>*のフォーマットで示した有効な *.pgpass* ファイルの一例です。

```
==================
*:5443:*:lifekeeper:jh43tmp2009
==================
```

注記：*.pgpass* ファイルは、ユーティリティ psql による自動の (非ターミナルまたはスクリプト) 接続のために必要です。 *.pgpass* ファイルは、パスワード保護されたインスタンスが保護される各サーバ上に存在している必要があります。

# 6.12.4.3. LifeKeeper ソフトウェアのインストール

PostgreSQL ソフトウェアをインストールし、データベースを作成したら、LifeKeeper Core ソフトウェアと必要なパッチをインストールし、続いて PostgreSQL Recovery Kit をインストールできます。

LifeKeeper パッケージのインストール方法については、SPS for Linux インストールガイド を参照してください。

# 6.12.4.4. PostgreSQL リソースのための LifeKeeper 設定項目

PostgreSQL Recovery Kit には、特定のシナリオでリソース保護をカスタマイズできるように、設定項目のための環境変数が用意されています。これらの変数の値を変更するには、ファイル */etc/default/LifeKeeper* を編集します。新しい設定を反映するためにプロセスを再起動する必要はありません。デフォルト値は PostgreSQL Recovery Kit がインストールされるほとんどの環境で動作する値となっています。

## ·LKPGSQL_CONN_RETRIES

この設定項目は、データベースが開始するまで PostgreSQL Recovery Kit が待機する時間を制御します。この時間は、「(LKPGSQL_CONN_RETRIES * 5) = データベースインスタンスの開始まで待機する合計時間(秒単位)」の式を用いて Recovery Kit により計算されます。この変数の設定は、リソースの In Service 要求とリソースのローカルリカバリの両方に影響します。

## ·LKPGSQL_DISCONNECT_CLIENT

この設定項目は、postmaster がクラッシュした場合にアクティブなクライアントが切断されるかどうかを制御します。値を 1 (true) に設定すると、アクティブなクライアントはリソースのローカルリカバリが進行している間は切断されます。値を 0 (false) に設定すると、アクティブなクライアントはリソースのローカルリカバリが進行している間も切断されません。この変数は、リソースのローカルリカバリイベントのみに影響し、postmaster プロセスが稼働していない場合のローカルリカバリイベントの場合にのみ適用されます。

## ·LKPGSQL_SDIRS

この設定項目は、PostgreSQL データベースをシャットダウンする際のクライアント切断動作を制御します。カンマで区切られた設定項目をデフォルトファイルに追加する必要があります。このオプションを設定することにより、指定されたリソースインスタンスもしくは保護されるデータディレクトリと一致するインスタンスが、クライアントに対してシャットダウン時の切断を要求しません。

```
LKPGSQL_SDIRS=/protected/pgsql-datadir

LKPGSQL_SDIRS=/protected/pgsql-datadir,/otherprotected/pgsql-datadir
```

*/protected/pgsql-datadir* および */otherprotected/pgsql-datadir* は、LifeKeeper の保護下の PostgreSQL データディレクトリです。

**注記:** このオプション LKPGSQL_SDIRS および LKPGSQL_IDIRS は排他的なオプションです。LKPGSQL_SDIRS もしくは LKPGSQL_IDIRS の値は、階層の作成中に選択した保護対象の datadir の値と一致していなければなりません。

# ·LKPGSQL_IDIRS

この設定項目は、PostgreSQL データベースをシャットダウンする際のクライアント切断動作を制御します。カンマで区切られた設定項目をデフォルトファイルに追加する必要があります。このオプションを設定することにより、指定されたリソースインスタンスもしくは保護されるデータディレクトリと一致するインスタンスが、クライアントに対してシャットダウン時の即時切断を強制します。

```
LKPGSQL_IDIRS=/protected/pgsql-datadir

LKPGSQL_IDIRS=/protected/pgsql-datadir,/otherprotected/pgsql-datadir
```

*/protected/pgsql-datadir* および */otherprotected/pgsql-datadir* は、LifeKeeper の保護下の PostgreSQL データディレクトリです。

**注記:** このオプション LKPGSQL_SDIRS および LKPGSQL_IDIRS は排他的なオプションです。LKPGSQL_SDIRS もしくは LKPGSQL_IDIRS の値は、階層の作成中に選択した保護対象の datadir の値と一致していなければなりません。

# 6.12.4.5. PostgreSQL リソース階層の作成

プライマリサーバで次の手順を実行します。

1. **[Edit]** メニューの **[Server]** から **[Create Resource Hierarchy]** を選択します。

   **[Create Resource Wizard]** ダイアログボックスが表示されます。

2. ドロップダウンリストから **[PostgreSQL Database]** を選択し、 **[Next]** をクリックします。.

3. 以下の情報を入力するように要求されます。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも **[Cancel]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| **Switchback Type** | **[intelligent]** または **[automatic]** を選択します。フェイルオーバ後、バックアップサーバで PostgreSQL リソースが In Service (アクティブ) になるとき、プライマリサーバにスイッチバックされる方法が決まります。インテリジェントスイッチバック (intelligent) の場合、リソースをプライマリサーバにスイッチバックするときに管理者の介入が必要ですが、自動スイッチバック (automatic) の場合は、プライマリサーバがオンラインに戻り、LifeKeeper のコミュニケーションパスが再確立されるとすぐにスイッチバックが実行されます。<br><br>**注記:** スイッチバック方法は、PostgreSQL リソースと依存関係を構築しているリソースのスイッチバック方法と一致する必要があります。 |
| **PostgreSQL Executable Location** | このフィールドは、PostgreSQL 実行可能ファイルを含むディレクトリパスを指定するために使用されます。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| **PostgreSQL Client Executable Location** | このフィールドは、PostgreSQL 実行可能ファイル `psql` を含むディレクトリパスを指定するために使用されます。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| **PostgreSQL Administration Executable Location** | このフィールドは、PostgreSQL 実行可能ファイル `pg_ctl` を含むディレクトリパスを指定するために使用されます。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| **PostgreSQL Data Directory** | このフィールドは、LifeKeeper の保護下に置かれる PostgreSQL データディレクトリ (datadir) の場所を指定するために使用されます。指定されたディレクトリは、共有ファイルシステムに存在し、常駐している必要があります。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| **PostgreSQL Port** | このフィールドは、ポストマスタデーモンがクライアントアプリケーションからの接続をリスニングする TCP/IP ポート番号を指定するために使用されます。 |

| | |
|---|---|
| **PostgreSQL Socket Path** | このフィールドは、ポストマスタデーモンがクライアントアプリケーションからの接続をリスニングする UNIX ドメインソケットへのフルパスを指定するために使用されます。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| **PostgreSQL Database Administrator User** | このフィールドは、指定したデータベースインスタンスとそのインスタンスへの接続および管理者権限に対する PostgreSQL データベース管理ユーザ名を指定するために使用されます。 |
| **PostgreSQL Logfile** | このフィールドは、PostgreSQL の起動や停止に使用する `pg_ctl` の -l オプションで使われるログファイルパスを指定するために使用されます。 |
| **PostgreSQL Database Tag** | プライマリサーバ上の新しい PostgreSQL データベースリソースに対する固有のタグ名です。デフォルトのタグ名は「pgsql」の後にデータベースインスタンスのポート番号を付けたものです。別の固有のタグ名を入力することもできます。タグ名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |

4. **[Create]** をクリックします。 **[Create Resource Wizard]** によって、PostgreSQL リソース階層が作成されます。LifeKeeper によって入力データが検査されます。問題が検出された場合、情報ボックスにエラーメッセージが表示されます。

5. PostgreSQL リソース階層が正常に作成されたこと、フェイルオーバによる保護を実現するにはその階層をクラスタ内の別のサーバに拡張する必要があることを示すメッセージが表示されます。
**[Next]** をクリックします。

6. **[Continue]** をクリックします。 **[Pre-extend Wizard]** が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、PostgreSQL リソース階層の拡張 の手順 2 を参照してください。

# 6.12.4.6. PostgreSQL リソース階層の削除

LifeKeeper 設定内のすべてのサーバから PostgreSQL リソース階層を削除するには、次の手順を実行します。

1. **[Edit]** メニューの **[Resource]** から **[Delete Resource Hierarchy]** を選択します。
2. **[Target Server]** で、PostgreSQL リソース階層を削除するサーバの名前を選択します。

   **注記:** 左側ペインのグローバルリソースまたは右側ペインの個々のリソースインスタンスを右クリックして **[Delete Resource]** 作業を選択した場合、このダイアログボックスは表示されません。
3. **[Hierarchy to Delete]** を選択します。(このダイアログは、右ペインまたは左ペインのリソースインスタンス上で右クリックして **[Delete Resource]** を選択した場合には表示されません。) **[Next]** をクリックします。
4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[Next]** をクリックします。
5. PostgreSQL リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。
6. **[Done]** をクリックして終了します。

# 6.12.4.7. PostgreSQL リソース階層の拡張

この操作は **[Edit]** メニューから開始できます。または **[Create Resource Hierarchy]** の処理を完了すると自動的に開始されます。その場合は、下記の **手順 2** を参照してください。

1. **[Edit]** メニューの **[Resource]** から **[Extend Resource Hierarchy]** を選択します。 ***[Pre-Extend Wizard]*** が表示されます。 **拡張** 操作に慣れていない場合は、 **[Next]** をクリックします。 LifeKeeper のリソース階層の拡張作業のデフォルト値が分かっていて、入力と確認を省略する場合は、 **[Accept Defaults]** をクリックします。
2. ***[Pre-Extend Wizard]*** で以下の情報を入力します。

   **注記:** 最初の 2 つのフィールドは **[Edit]** メニューの **[Extend]** から操作を開始した場合にだけ表示されます。

| フィールド | ヒント |
|---|---|
| Template Server | PostgreSQL リソースが現在 In Service のサーバを選択します。 |
| Tag to Extend | 拡張する PostgreSQL リソースを選択します。 |
| Target Server | 拡張先のサーバを入力または選択します。 |
| Switchback Type | フェイルオーバ後、バックアップサーバで PostgreSQL リソースが In Service (アクティブ) になるとき、プライマリサーバにスイッチバックされる方法が決まります。 **[intelligent]** または **[automatic]** を選択できます。スイッチバックタイプは、必要な場合 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。<br><br>**注記:** スイッチバック方法は、PostgreSQL リソースと依存関係を構築しているリソースのスイッチバック方法と一致する必要があることに注意してください。 |
| Template Priority | **テンプレートの優先順位** を選択または入力します。これはサーバで現在 In Servic の PostgreSQL 階層の優先順位です。優先順位は、1 ~ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります(数字 1 が最高の優先順位に相当します)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。<br><br>**注記:** このフィールドは階層を最初に拡張するときだけ表示されます。 |
| Target Priority | これは、別サーバにある同等の階層に対する、新しく拡張する PostgreSQL 階層の優先順位です。1 ~ 999 の範囲の未使用の値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |

3. 拡張前のチェックが正常に終了したというメッセージが表示されたら、 **[Next]** をクリックします。

4. 拡張する階層に応じて、拡張されるリソースタグ(一部編集不可) を示す一連の情報ボックスが表示されます。

5. ***[Extend Wizard]*** で以下の情報を入力します。

| フィールド | ヒント |
|---|---|
| PostgreSQL Executable Location | このフィールドは、PostgreSQL 実行可能ファイルを含むディレクトリパスを指定するために使用されます。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| PostgreSQL Database Tag | プライマリサーバ上の新しい PostgreSQL データベースリソースに対する固有のタグ名です。デフォルトのタグ名は語「pgsql」の後にデータベースインスタンスのポート番号を付けたものです。別の固有のタグ名を入力することもできます。タグ名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |

6. 「Hierarchy extend operations completed」というメッセージが表示されたら、階層を別のサーバに拡張する場合は、 **[Next Server]** をクリックし、これ以上拡張作業が必要ない場合は、 **[Finish]** をクリックします。

7. 「Hierarchy Verification Finished」のメッセージが表示されたら、 **[Done]** をクリックします。

# 6.12.4.8. PostgreSQL リソース階層の拡張解除

LifeKeeper クラスタ内にある 1 つのサーバからリソース階層を削除するには、次の手順を実行します。

1. **[Edit]** メニューの **[Resource]** から **[Unextend Resource Hierarchy]** を選択します。

2. **[Target Server]** で、PostgreSQL リソースを拡張解除するサーバを選択します。リソースが現在 In Service のサーバは選択できません。(このダイアログボックスは、右側のペインでリソースインスタンスを右クリックして、拡張解除作業を選択した場合には、表示されません。) **[Next]** をクリックします。

3. 拡張解除する PostgreSQL 階層を選択し、 **[Next]** をクリックします。(このダイアログボックスは、いずれかのペインでリソースインスタンスを右クリックして、拡張解除作業を選択した場合には、表示されません。)

4. ターゲットサーバと拡張解除の対象として選択した PostgreSQL リソース階層を確認する情報ボックスが表示されます。 **[Unextend]** をクリックします。

5. PostgreSQL リソースが正常に拡張解除されたことを確認する別の情報ボックスが表示されます。 **[Done]** をクリックして、 **[Unextend Resource Hierarchy]** メニューを終了します。

# 6.12.4.9. PostgreSQL 設定の参照

リソースプロパティダイアログは [Edit] メニューもしくはリソースコンテキストメニューから参照可能です。このダイアログはサーバ上の特定のリソースに対するプロパティを表示します。 **[Edit]** メニューからアクセスした場合は、リソースおよびサーバを選択することができます。 **リソースコンテキスト** メニューからアクセスする場合はサーバを選択することができます。

**[Configuration]** タブからは以下の PostgreSQL 設定を参照することができます。

- Executable Path
- Client Executable Name
- Admin Executable Name
- Bind Setting
- Startup Log Location
- PostgreSQL Operating System User Name
- PostgreSQL Database Administrator User
- Version Number
- Data Directory
- Socket Location
- Port Number
- OS Daemon Name

# 6.12.4.10. PostgreSQL Recovery Kit の旧バージョンからのアップグレード

旧バージョンの SPS for Linux の PostgreSQL ソフトウェアからのアップグレード時、既存の SPS PostgreSQL リソースのインスタンスは変更されます。SPS ソフトウェアがサーバ上でアップデートされる際に、次の保持される値が、SPS 内部の情報のフィールドに自動的に追加されます。

- **Client Executable Location ( *psql* )** -保護対象のデータベースインスタンスへの接続に使用される *psql* もしくは同等のクライアントユーティリティの場所。アップグレード後、この値は LifeKeeper GUI のプロパティ表示で確認することが可能です。また、この値は LifeKeeper コマンドラインの `set_value` ユーティリティを使用して確認することも可能です。

  `set_value` は、LifeKeeper PostgreSQL Recovery Kit が内部のリソース情報の項目の値をアップデートするために提供される LifeKeeper ユーティリティの名前です。このユーティリティの使用は本トピックで説明されている問題、もしくは SIOSTechnology Corp. 社のサポートチームによる依頼および指示に限るものとします。

  **注記**：`set_value`ユーティリティは、正確なエラーチェックは行わないので、一般的な使用には向いていません。

- **Administration Executable Location (*pg_ctl*)** - 保護対象のデータベースインスタンスの起動、停止、ステータス確認に使用する *pg_ctl* または同等の管理ユーティリティの場所です。アップグレード後、この値は LifeKeeper GUI のプロパティ表示で確認することが可能です。また、この値は LifeKeeper コマンドラインの set_value ユーティリティを使用して確認することも可能です。

- **PostgreSQL Database Administrator User** - LifeKeeper が保護するインスタンスの PostgreSQL データベース管理者ユーザです。このユーザは保護対象のデータベースインスタンスへの接続および管理権限を持っている必要があります。デフォルト値は、PostgreSQL のデータディレクトリを保持している PostgreSQL オペレーティングシステムユーザになります。アップグレード後、この値は LifeKeeper GUI のプロパティ表示で確認することが可能です。また、この値は SPS コマンドラインの set_value ユーティリティを使用して確認することも可能です。

- **PostgreSQL Daemon Name (postmaster)** - バックエンドで起動しているデーモン名です。この値は、データベースの最初のステータス確認時に決定されます。デフォルト値は postmaster です。アップグレード後、この値は LifeKeeper GUI のプロパティ表示で確認することが可能です。また、この値は LifeKeeper コマンドラインの set_value ユーティリティを使用して確認することも可能です。

- **Default Test Database (template1)** - データベースインスタンスが基本的な接続確認を行っている間に LifeKeeper が使用するデータベースです。アップグレード後、デフォルトのテストデータベースは **template1** に設定されます。

- **PostgreSQL Maximum Monitoring Hangs (`LKPGSQL_QCKHANG_MAX`)** - この設定は、リストアやリカバリのためのフェイルオーバアクションが開始される前に、無制限な数のコネクションがハングすることからの保護を提供します。PostgreSQL Recovery Kit の監視の一部は、保護されたデータベースへの接続を必要とします。どれだけの数のコネクションのハングを許容するかは、リソー

ス作成時に **LKPGSQL_QCKHANG_MAX** によって決定されます。8.1.2 以前のバージョンでは、デフォルトの値は 15 でした。バージョン 8.1.2 （またはそれ以降のバージョン）へのアップグレード後は、デフォルトの値は 2 です。この値はリソース作成時にリソース情報と紐付けて保存されますので、手動でアップデートしない限り、バージョン 8.1.2 にアップグレードする前に作成されたリソースはいずれもデフォルト値 15 のままになります。バージョン 8.1.2 （またはそれ以降のバージョン）で作成されたリソースはデフォルト値の 2 となります。値は set_value ユーティリティ を使用して確認することもできます。

重要な注記: SPS for Linux PostgreSQL Recovery Kit ソフトウェアをアップグレードした後、プライマリサーバからバックアップサーバへリソースインスタンスのフェイルオーバをシュミレーションするために 手動でスイッチオーバを行い PostgreSQL リソース階層のテストを行ってください。

| | |
|---|---|
| アップグレードに関する重要事項 | アップグレードを行った後、リソースを起動できない場合は、以下の事項を確認してください:<br><br>**Client Executable name is not found or incorrect (クライアントの実行可能ファイル名が見つからない、または不正です)**<br><br>この値は set_value ユーティリティを使用してアップデートできます。クライアントの実行可能ファイルをアップデートする構文は以下の通りです。<br><br>• /opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value <tag> 'clientexe' <full path to the psql utility><br><br>• 例: /opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value pgsql-5443 'clientexe' '/pgsql/clientutils/psql'<br><br>• **Administration Executable name is not found or incorrect (管理実行可能ファイル名が見つからない、または不正です)**<br><br>• この値は set_value ユーティリティを使用してアップデートできます。管理実行可能ファイルをアップデートする構文は以下の通りです。<br><br>• /opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value <tag> 'osexe' <full path to the pg_ctl utility> |

| | |
|---|---|
| | • 例: `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value pgsql-5443 'osexe' '/pgsql/adminutils/pg_ctl'` |
| | 複数のハング･イベントからの復旧までのインターバルの短縮 ( LKPGSQL_QCKHANG_MAX )<br><br>**8.1.2 以前のバージョンでは、Maximum Monitoring Hangs の値は大きすぎました。**<br><br>既存の PostgreSQL リソースインスタンスの **Maximum Monitoring Hangs** の値は、set_value ユーティリティを使用して見る、またはセットすることができます。<br><br>**Maximum Monitoring Hangs** ( LKPGSQL_QCKHANG_MAX ) の値をセットする構文は以下の通りです。<br>• `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value <tag> 'hangmax' <number>.`<br><br>• 例: `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value pgsql-5443 'hangmax' 3.`<br><br>• 注記: クラスタ内の全てのノードで値を更新するためには、-c オプションを付与してください。(`set_value -c <tag>…`).<br><br>この値を **確認する** 構文は以下の通りです。<br>• `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value -l <tag> 'hangmax'`<br><br>• **例:** `/opt/LifeKeeper/lkadm/subsys/database/pgsql/bin/set_value -l pgsql-5443 'hangmax'` |

# 6.12.5. 管理

## データベース管理者ユーザの更新

ユーザの更新 オプションにより、LifeKeeper の管理者は、LifeKeeper PostgreSQL リソースインスタンスに対する現在の PostgreSQL データベース管理者ユーザを変更することができます。

データベース管理者ユーザの更新

## PostgreSQL リソース階層のテスト

PostgreSQL リソース階層のテストは、手動でサーバをスイッチオーバすることで、プライマリサーバからバックアップサーバへのリソースインスタンスのフェイルオーバをシミュレートして実行できます。

LifeKeeper GUI からの手動スイッチオーバの実行

## EnterpriseDB Postgres Plus Advanced Server 環境

EnterpriseDB Postgres Plus Advanced Server の保護

## FUJITSU Software Symfoware/Enterprise Postgres 環境

Symfoware Server / Enterprise Postgres の保護

# 6.12.5.1. LifeKeeper GUI からの手動スイッチオーバの実行

LifeKeeper GUI で **[Edit] > [Resource] > [In Service]** を選択すると、手動スイッチオーバを開始できます。例えば、バックアップサーバで [In Service] 要求 が実行されると、PostgreSQL リソース階層がバックアップサーバ側で In Service になり、プライマリサーバ側では Out of Service になります。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバに変わります。

[Out of Service] 要求を実行した場合、リソース階層は Out of Service になりますが、別のサーバで In Service にはなりません。

**重要:** バックアップサーバでリソース階層を In Service にしたら、データベースに接続できるかどうか試す必要があります。パスワード保護されたインスタンスに関しては *.pgpass* ファイルがバックアップ上で有効であることが重要です。 *.pgpass* ファイルが有効であるか確認するためには、データベースへのクライアント接続が `psql` ユーティリティと PostgreSQL データベース管理者ユーザの両方を使用して実行できることを確認してください。パスワード認証を要求されなければ、接続が有効な *.pgpass* ファイルが存在していることが分かります。

# 6.12.5.2. EnterpriseDB Postgres Plus Advanced Server の保護

EnterpriseDB Postgres Plus Advanced Server リソースを保護するにあたって、追加の LifeKeeper 設定は必要ありません。

| 問題 | 解決方法 |
| --- | --- |
| EnterpriseDB Postgres Plus Advanced Server のインストール中に、 **[Configuration Mode]** ダイアログでオプションの **[PostgreSQL-compatible defaults and samples]** を選択した場合は、LifeKeeper が使用する「edb」データベースが作成されません。 | 「createdb」ユーティリティを使用して「*edb* 」データベースを手動で追加してください。<br><br>`createdb -p <port> -h <socket path> edb` コマンドは PostgreSQL オペレーティングシステムユーザとして実行する必要があります。以下はサンプルです。<br><br>`su - postgres`<br><br>`postgres@server1 ~>createdb -p 5435 -h /var/lib/postgres edb` |

# 6.12.5.3. Symfoware Server/Enterprise Postgres の保護

Symfoware Server/Enterprise Postgres を保護するにあたって、サポート範囲及びご利用の環境について以下に解説します。

| サポート範囲 |
| --- |
| • PostgreSQL と互換性のある機能をサポートします。<br>• ミラーリング機能はサポートされません。ミラーを作成しデータに冗長性を持たせたい場合は、DataKeeper をご利用ください。<br>• Symfoware Server の Native インターフェースはサポートされません。Open インターフェース (Symfoware 12.2)、Postgres (Symfoware 12.3 以降) をご利用ください。<br>• WebAdmin は非対応です。詳細は Fujitsu Software の Symfoware Server クラスタ運用ガイドをご参照ください。<br>• 以下の機能はサポートされません。<br>◦ WAL の2重化<br>◦ 暗号化機能（格納データの暗号化）<br>◦ データ秘匿化<br>◦ 並列検索<br>◦ インメモリ機能 |

| 構成上の注意 |
| --- |
| DB 管理ユーザー (OS ユーザー) のログイン時の環境設定ファイル (.bash_profile 等) に、Symfoware のコマンド (pg_ctl,psql 等) の実行に必要な環境変数 LD_LIBRARY_PATH を設定してください。 |

# 6.12.5.4. データベース管理者ユーザの更新

**ユーザの更新** オプションは、保護されたリソースが存在する全システム上の PostgreSQL データベース管理者ユーザに対して保存された値を更新します。 **[Update User]** オプションは **LifeKeeper のリソースツールバー** もしくは **LifeKeeper のリソースコンテキストメニュー** から実行することが可能です。

PostgreSQL データベース管理者ユーザを更新するためには、プライマリサーバで以下の手順を実行してください。

注記 : **[Update User]** メニューおよびツールバーオプションは out of service のリソースに対して無効になります。

1. ツールバーの **[Update User]** アイコンを選択するか、リソースコンテキストメニューから [Update User] を選択してください。

   **[Update User Wizard]** のダイアログが表示されます。

2. 以下の情報を入力するように要求されます。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも **[Cancel]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| **[PostgreSQL Database Administrator User]** と入力 | このダイアログは、指定したデータベースインスタンスとそのインスタンスへの接続および管理者権限に対する PostgreSQL データベース管理者ユーザ名を指定するために使用されます。<br><br>**注記:** 認証スクリプトは、指定した値を使用して接続確認を行います。パスワード保護されたインスタンスは PostgreSQL データベース管理者ユーザの *.pgpass* ファイルの有効なエントリを要求します。 |
| **Confirm Update Action** | このダイアログでは、以前のユーザから新しいユーザに対して行う変更について確認をします。 |

3. **[Update]** をクリックしてください。現在保護されているリソースがあるすべてのサーバ上で PostgreSQL データベース管理者ユーザが更新されます。

# 6.12.6. トラブルシューティング

本セクションでは、SPS PostgreSQL リソース階層の作成時や拡張時、またはリソースの削除時やリストア時に表示される可能性のあるメッセージの一覧を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

他の SPS コンポーネントからメッセージが出されることもあります。そのような場合は、メッセージカタログ を参照してください。メッセージカタログには、操作、管理、GUI など、SIOS Protection Suite for Linux を使用しているときに遭遇する可能性がある、すべてのエラーコードが列挙されています。また、エラーコードの原因に関する追加の説明や、問題解決のために必要な処置についても、必要に応じて記載されています。この完全なリストを検索すると、受信したエラーコードを見つけることができます。また、関連する SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

- 全般項目
- 設定項目

# 6.12.6.1. 全般項目

以下のエラーメッセージおよび状態は PostgreSQL Recovery Kit 使用時に表示される可能性があります。

| エラー | 解決方法 |
|---|---|
| Unable to protect PostgreSQL database using the same port as another LK protected PostgreSQL database. | PostgreSQL のこのバージョンに *postgresql.conf* ファイルが含まれていることを確認してください。 *postgresql.conf* ファイルで、エントリ listen_address= に、データベースインスタンスで使用される IP アドレスを設定してください。<br><br>**注記:** *postgresql.conf* ファイルの listen_address= のフォーマットは重要です。構文エラーにより、データベースサーバの起動に失敗することもあります。 |
| Unable to perform a manual switchover of version 8.X when clients are connected. | デフォルトの (スマート) シャットダウンオプションがスイッチオーバでクライアントの切断に失敗しました。接続されたクライアントにより、シャットダウンに失敗し続けてしまう場合は、LKPGSQL_SDIRS が設定されていないかどうかを確認してください。障害が継続するようであれば、LifeKeeper の設定項目 LKPGSQL_IDIRS を設定してください。 |
| Unable to connect from a remote client to the database server. | PostgreSQL のリモートホストログインを有効にするには、 *PostgreSQL Administration Guide* の *pg_hba.conf* ファイルの設定に関する説明を参照してください。 |
| psql: connectDBStart() — connect() failed: No such file or directory. Is the postmaster running at 'localhost' and accepting connections on Unix socket '<port>'?" | ソケットファイルが存在し、インスタンスが現在実行されていることを確認します。ソケットファイルが */tmp* にある場合、 */tmp* ディレクトリをクリーンアップする cron ジョブによって削除された可能性があります。リソースを out of service にしてから、 in service に戻します。次に、PostgreSQL ソケットファイルを残すように cron ジョブを修正します。 |
| PostgreSQL resource hierarchy fails to come in service but the database is running. | データベースが、指定された期間内に LifeKeeper クライアント要求に応答できなかった可能性があります。 */etc/default/LifeKeeper* の設定項目 LKPGSQL_CONN_RETRIES を調整して、PostgreSQL データベースインスタンスのリカバリと再起動に許容される秒数を増やします。 |
| PostgreSQL resource hierarchy fails local recovery following a postmaster crash with active client connections. | 多数のアクティブクライアントが PostgreSQL に接続しているとき、データベースはクライアント接続が終了するまで正しく再起動できない場合があります。この場合、ローカルリカバリが正常に行われるよう強制的にクライアント接続を終了させることをお勧めします。<br>LKPGSQL_DISCONNECT_CLIENT 変数を */etc/default/LifeKeeper* に設定して、この場合の PostgreSQL リソース階層の動作を制御することができます。値を 1 ( **true** ) に設定すると、クライアントプロセスに SIGTERM シグナルが送信され、データベースから強制的に切断されます。この処置は、ポストマスタプロセスがローカルリカバリ中に稼働していない場合のみに取ることができます。 |

| | |
|---|---|
| Unable to connect to database with error “WARNING: password file “*/home/<osuser>/.pgpass*” has world or group read access” | *.pgpass* ファイルの権限を、u=rw(0600) にしてください。 *.pgpass* ファイルの権限および所有者を変更してください。 |
| FATAL: syntax error in file “*/<datadir>/postgresql.conf*” line 50, near token “.17” | *postgresql.conf* ファイルの listen_address= のエントリが正しい引用を含んでいません。エントリが正しいかどうか、またはエントリが正しい引用を含んでいるかを確認してください。 |

# 6.12.6.2. 設定項目

| 設定項目 | 機能 |
|---|---|
| LKPGSQL_KILLPID_TIME | プロセス ID が停止した後、そのプロセスに対する再チェックを行うまでの時間 |
| LKPGSQL_CONN_RETRIES | 旧 LKPGSQLMAXCOUNT – 操作 (開始もしくは停止) を行った後、クライアント接続を試みる回数 |
| LKPGSQL_ACTION_RETRIES | アクションコマンドに失敗するまで、開始と停止を試行する回数 |
| LKPGSQL_STATUS_TIME | status コマンドのタイムアウト値 |
| LKPGSQL_QCKHANG_MAX | データベースインスタンスがフェールオーバ / sendevent を発生させるまでに許容される、クイックチェックがハングする回数 |
| LKPGSQL_CUSTOM_DAEMON | postgres デーモン (デフォルト postmaster) に対する追加の別名の指定を許可します。 |
| LKPGSQL_IDIRS | 旧 LKPGSQL_IPORTS – immediate オプションのみを使用してシャットダウンするインスタンスに対する datadir のエントリを含みます。 |
| LKPGSQL_SDIRS | Smart オプションを使用してシャットダウンするインスタンスに対する datadir のエントリを含みます。 |
| LKPGSQL_DISCONNECT_CLIENT | データベースに障害が発生している間の PostgreSQL リソース階層の振る舞いを制御します。値を 1 **(true)** に設定すると、クライアントプロセスに SIGTERM シグナルが送信され、データベースから強制的に切断されます。この処置は、postmasterプロセスがローカルリカバリ中に稼働していない場合のみに取ることができます。<br><br>※注:PostgreSQL 8.2以降では、本パラメータをご利用いただけません。 |
| LKPGSQL_DISCONNECT_CLIENT_BYTAG | LKPGSQL_DISCONNECT_CLIENT と類似していますが、この設定は、処置を、この設定項目で指定したカンマで区切られたタグのリストに限定します。<br><br>※注:PostgreSQL 8.2以降では、本パラメータをご利用いただけません。 |
| LKPGSQL_RESUME_PROC | プロセスの停止状態が検出されたときに (プロセス状態が T であるとき) 再開するか無視するかを決めます。 |
| LKPGSQLDEBUG | PostgreSQL database kit および postgres データベースのデバッグを有効にします。0 ~ 5 の値が有効です。数字が大きくなるほどデバッグ情報が増えます。 |

| | |
|---|---|
| | この設定項目は、オプション -d <LKPGSQLDEBUG> を使用して postmaster データベースへと渡されます。 |

# 6.13. Postfix Recovery Kit 管理ガイド

## Postfix Recovery Kit テクニカルドキュメンテーション

Postfix はさまざまな役割を演じます。 そのすべてが、電子メールの適切な流れには不可欠なものです。 Postfix はメールの着信がないかネットワークをリスニングし、メールを別のサーバに送信し、ローカルメールをローカルプログラムに配信します。

LifeKeeper for Linux Postfix Recovery Kit は、LifeKeeper 環境で障害の発生したプライマリサーバからバックアップサーバに Postfix を回復する仕組みを提供します。LifeKeeper もPostfix も、 大幅な時間の損失や人手の介入なしにフェイルオーバ処理を通じてデータの整合性を保証します。

## ドキュメントの内容

本書は以下のトピックについて説明しています。

- ドキュメンテーションとリファレンス 。LifeKeeper for Linux のドキュメントと所在の一覧です。
- 要件 。Postfix Recovery Kit を適切に設定、導入、運用するうえで必要なハードウェアとソフトウェアについて説明しています 。LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SIOS Protection Suite インストールガイド を参照してください。
- LifeKeeper for Linux Postfix Recovery Kit の設定 。 Postfix Recovery Kit を適切に設定するうえで必要な手順について説明しています。
- Postfix の 構成確認 。Postfix のリソース階層を構成する前に、Postfix の構成確認方法を 記述 します。
- LifeKeeper 設定作業 。LifeKeeper GUI を使用して Postfix リソース階層を作成し管理する作業について説明しています。
- スプール (メールボックス) の依存関係作成 。Postfix リソースと Mailbox Spool ファイル･システムリソースの間で手動で依存関係をつくる方法を記述します。
- リソース階層のテスト 。LifeKeeper GUI とコマンドラインインターフェースを使用して、Postfix リソース階層をテストするためのステップを記述します。
- トラブルシューティング 。LifeKeeper for Linux のエラー･メッセージの一覧とその説明です。

## ドキュメンテーションとリファレンス

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション

- SIOS Protection Suite インストールガイド
- Optional Recovery Kit ドキュメンテーション

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

## Postfix のドキュメンテーションとリファレンス

以下は、Postfix アプリケーションおよび LifeKeeper Postfix Recovery Kit に関連する参考資料の一覧です。

- Postfix マニュアルページ
- Red Hat Postfix リファレンスマニュアル

# 6.13.1. ハードウェア要件とソフトウェア要件

SPS for Linux Postfix Recovery Kit をインストールする前に、SPS の構成が次の要件を満たしている必要があります。SPS for Linux ハードウェアおよびソフトウェアの設定に関する具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

## ハードウェア要件

- サーバ 。 製品媒体に付属する SPS for Linux リリースノート および SPS for Linux インストールガイド (製品メディアに収録) に記載の要件に従って設定されたLifeKeeper for Linux 対応サーバ。

- データストレージ 。 Postfix Recovery Kit は共有ストレージ、LifeKeeper DataReplication が提供するデータレプリケーションストレージのどちらもデータストレージとして使用する事ができます。

## ソフトウェア要件

- **TCP/IP** ソフトウェア 。LifeKeeper 環境の各サーバには TCP/IP ソフトウェアが必須です。

- **LifeKeeper** ソフトウェア 。クラスタの各サーバにインストールした LifeKeeper for Linux ソフトウェアのバージョン、および適用した LifeKeeper for Linux ソフトウェア・パッチのバージョンが同一であることが絶対条件です。

- **LifeKeeper for Linux IP Recovery Kit** 。同じバージョンのRecovery Kit を各サーバにインストールする必要があります。

- **IP** ネットワークインタフェース 。 各サーバには、イーサーネットTCP/IP をサポート するネットワークインタフェースが1 つ以上必要です。IP 切り替えが正しく機能するためには、ローカルネットワークに接続したユーザシステムが標準TCP/IP 仕様に準拠している必要があります。

  **注記：** 各サーバに必要なネットワークインタフェースが1 つだけであっても、異種メディア要件、スループット要件、シングルポイントオブフェイラの排除、ネットワークのセグメンテーションといった多くの理由で、複数のインタフェースを使用すべきです。

- **Postfix** ソフトウェア 。 各サーバには、LifeKeeper とPostfix Recovery Kit を設定する前に、PostfixソフトウェアのLinuxディストリビューション版をインストールして設定しておく必要があります。各サーバには同じバージョンをインストールする必要があります。最新リリースの互換性および発注情報については、SPS for Linux リリースノート を参照するか販売代理店にお問い合わせください。

  LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SPS for Linux インストールガイド を参照してください。

# 6.13.1.1. Recovery Kit のインストール

LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SPS for Linux インストールガイド を参照してください。

# 6.13.2. LifeKeeper for Linux Postfix Recovery Kit の設定

本セクションでは、LifeKeeper for Linux Postfix Recovery Kit の設定の詳細について説明します。Postfix Recovery Kit の設定と管理を開始する前に考慮すべき事項についても記載しています。LifeKeeper Core リソース階層の設定方法については、SPS for Linux テクニカルドキュメンテーション を参照してください。

# 6.13.2.1. 保護対象

Postfix Recovery Kit では以下のオブジェクトを保護対象とします。

- Postfix サービスデーモン
- Postfix のネットワークソケット

  仮想 IP アドレスを一つまたは複数作成し Postfix の設定ファイルで指定します。all を指定した場合はローカルループバックアドレスを死活監視に使用します。また、 Postfix の smtpd プロセスが使用する SMTP および SMTPS ポートは 25(465) をサポートします。

- Postfixが利用するキュー領域

  スプール(メールボックス)領域がキュー領域と別のファイルシステム上であり、そちらも合わせて保護する場合は別途ファイル・システムリソースを作成し、Postfix Recovery Kitで作成したリソース階層との依存関係を作成する必要があります。こちらは、スプール(メールボックス)の依存関係作成を参照してください。

# 6.13.2.2. 設定要件

- SMTP サービスが使用する IP アドレスを指定している場合、その IP アドレスは LifeKeeper で保護している仮想 IP アドレスである必要があります。
- main.cf

  以下の項目はPostfix Recovery Kit から参照されるため指定する必要があります。

  - º mail_owner
  - º setgid_group
  - º daemon_directory
  - º command_directory
  - º process_id_directory
  - º inet_interfaces

    all または仮想 IP アドレスを指定します。仮想 IP アドレスは複数指定可能です。
  - º queue_directory
  - º mail_spool_directory
- master.cf

  以下の項目は必須で指定する必要があります。

  - º smtpd を起動するための、smtp(s) サービスエントリ
- queue_directory 値のディレクトリは共有ストレージ上にある必要があります。そのディレクトリのファイルシステムは LifeKeeper により保護(ファイルシステムリソース) されている必要があります。
- メールボックスに共有されるデータが格納される場合、mail_spool_directory 値のディレクトリも共有ストレージ上にある必要があります。
- postfix オーナーID は全てのクラスタサーバ上で一致している必要があります。
- postdrop(setgid group) のグループID は全てのクラスタサーバ上で一致している必要があります。
- PostfixサービスはLifeKeeperが制御するので、システム起動時の自動起動を無効にする必要があります。

# 6.13.2.3. ポート定義とPostfix Recovery Kit

Postfix Recovery Kit はPostfix 設定ファイルのmaster.cf 内のエントリでSMTP サービス プロ セスで指定されたポートをリスニングします。ポートがサービス名(たとえば smtp) として指定 されている場合は、/etc/services ファイルのポート番号が調べられます。smtp であれば25 番ポ ート、smtps であれば465 ポートが使用されます。

```
smtp      inet  n       -       n       -       -       smtpd
```

# 6.13.2.4. DNS、Postfix とLifeKeeper

DNSは、メール配信用のバックアップ(代替) ホストを指定する仕組み(MX レコード) を備えています。この仕組みによって、ホストはnullクライアントのように、メールを受信するように設定されていない別のホストのメール処理を代行することもできます。 MXレコードは、すべてのメールをハブマシンまたはメールサーバに強制的に送信する仕組みも可能にします。 MXレコードでは、あるドメイン名のメール交換者(つまり、指定されたホスト名のメールを処理、または転送するホスト)が指定されます。これは、たとえばDNSサーバに次のような項目を追加することによって行います。

```
himalaya.sc.steeleye.com IN     MX      10  relay.steeleye.com.
```

この例では、サーバhimalaya.sc.steeleye.comは、このサーバのメールをrelay.steeleye.comに配信することを指定したMXレコードを持っています。 LifeKeeperで保護するサーバはMXレコードを持っていてはいけません。代わりに、Postfixリソース階層作成時に使用されるLifeKeeperが保護する仮想IPアドレスをMXレコード用として使用する必要があります。

# 6.13.2.5. 構成例

## 構成 1：アクティブ／スタンバイ設定の例

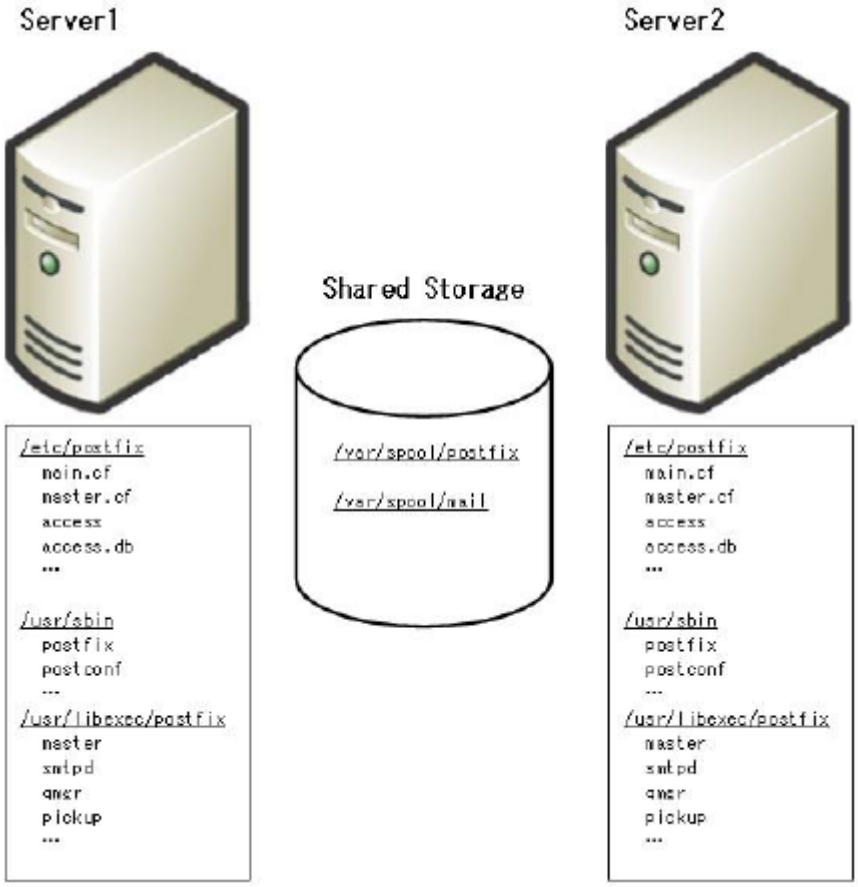



**図1: 典型的なLifeKeeper アクティブ/スタンバイPostfix 環境 1**

- Postfix の設定ファイルは個々のサーバ内に設定
- Postfix の実行ファイルおよび、デーモンプログラムは個々のサーバ内に設定
- キュー領域 /var/spool/postfix は共有ストレージ上に設定
- スプール(メールボックス)領域 /var/spool/mail は共有ストレージ上に設定

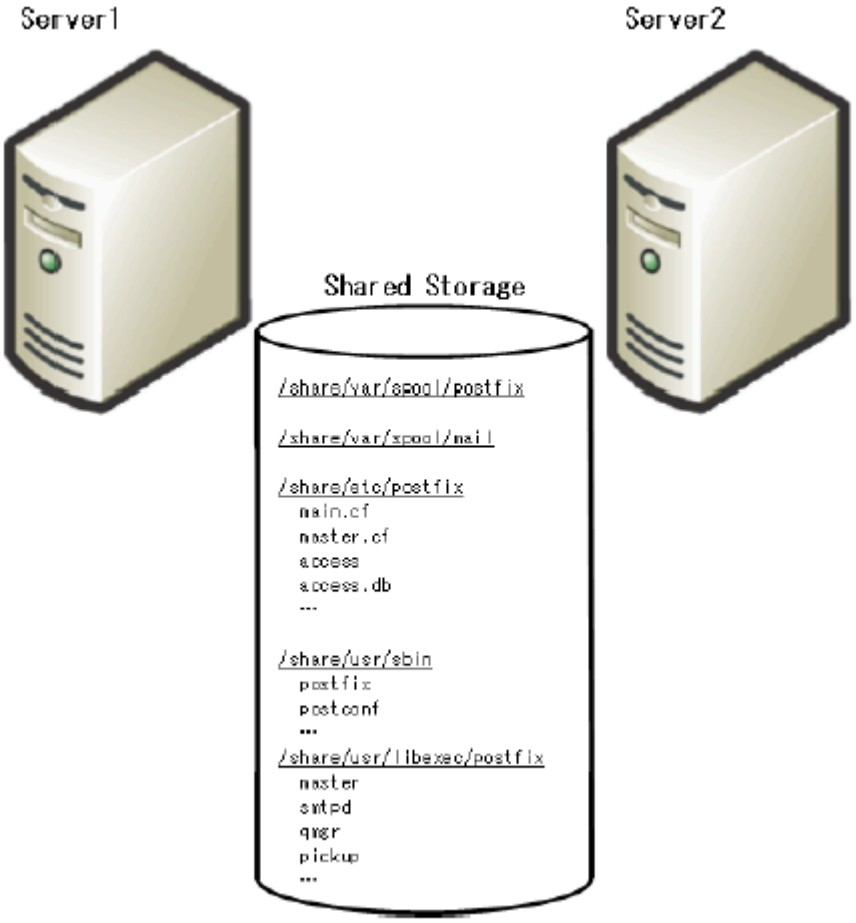



**図2: 典型的なLifeKeeper アクティブ/スタンバイPostfix 環境 2**

- Postfix の設定ファイルは共有ストレージ上に設定
- Postfix の実行ファイルおよび、デーモンプログラムは共有ストレージ上に設定
- キュー領域 /var/spool/postfix は共有ストレージ上に設定
- スプール(メールボックス)領域 /var/spool/mail は共有ストレージ上に設定

# 構成 2：アクティブ／アクティブ設定の例

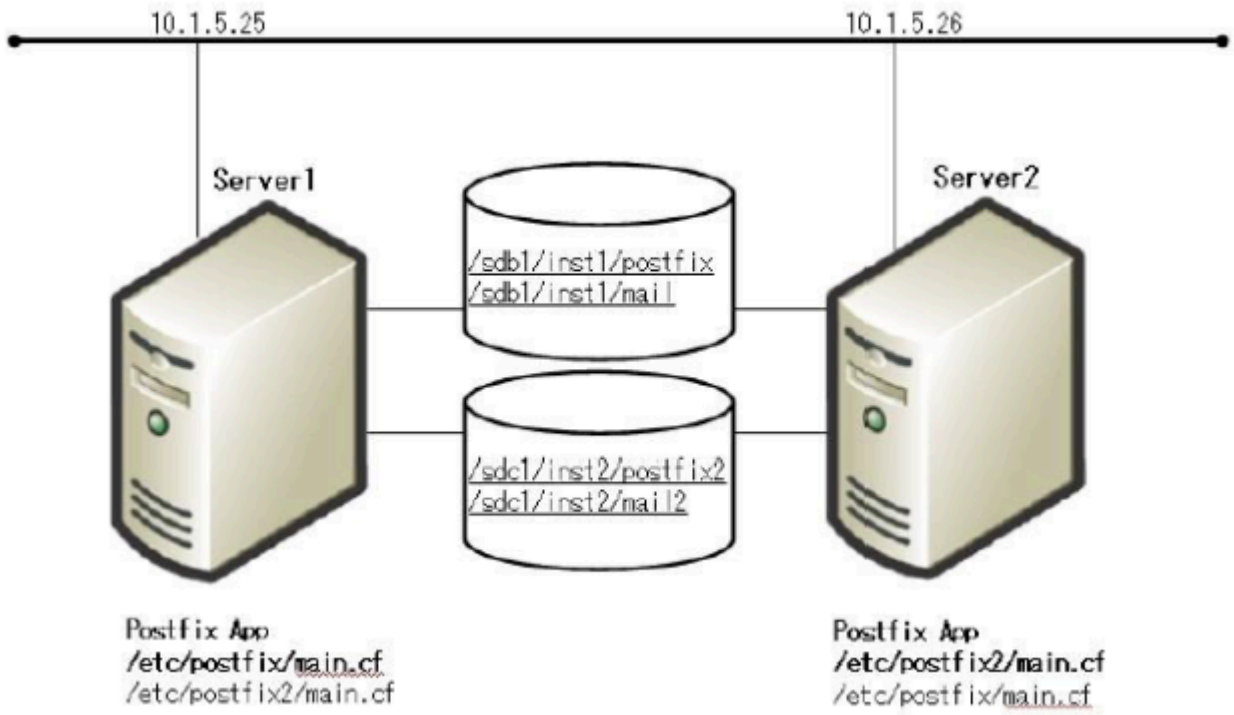



**図3: 典型的なLifeKeeper アクティブ/アクティブPostfix 環境 3**

[インスタンス1 はServer1 上でアクティブ]

Postfixの設定ファイルパス：/etc/postfix

Postfix の実行ファイルパス：/usr/sbin

キュー領域：/sdb1/inst1/postfix

メールスプール領域：/sdb1/inst1/mail

<main.cf>

inet_interfaces = 10.1.5.25, localhost

[インスタンス2 はServer2 上でアクティブ]

Postfix の設定ファイルパス：/etc/postfix2

Postfix の実行ファイルパス：/usr/sbin

キュー領域：/sdc1/inst2/postfix2

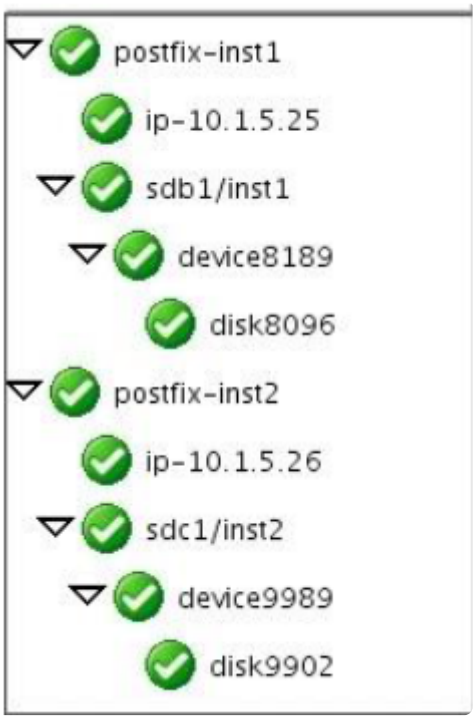
メールスプール領域：/sdc1/inst2/mail2

<main.cf>

inet_interfaces = 10.1.5.26

alternate_config_directories = /etc/postfix2

LifeKeeper GUIで Postfix リソースの階層構造は下記のようになります。



# 6.13.3. Postfix の構成確認

このセクションでは、Postfix Recovery Kit でのリソース作成前、つまりクラスタシステムを構築する前に、単体システムにおける確認作業の実施方法を構成例 1 を例に示します。

## Postfix の構成確認手順

1. Postfix の設定

   Postfix の設定ファイルは各サーバにあります。

```
main.cf (extract)

daemon_directory = /usr/libexec/postfix
command_directory = /usr/sbin
process_id_directory = pid
inet_interfaces = localhost , 192.168.0.10
mail_spool_directory = /var/spool/mail
queue_directory = /var/spool/postfix
```

```
master.cf (extract)

smtp      inet  n       -       n       -       -       smptd
```

2. SMTP サービスが待ち受ける仮想IP アドレスを起動

   Postfix のSMTP サービスが待ち受ける仮想IP アドレスを起動します。 LifeKeeper によりIP リソースを作成するか、以下のように”ifconfig”コマンドに より手動で起動します。

```
# ifconfig eth0:1 192.168.0.10 netmask 255.255.255.0 up
```

3. キュー領域のための共有ファイルシステムをマウント

```
# mkfs.ext3 /dev/sda1
# mount -t ext3 /dev/sda1 /mnt/queue
# mkdir -p /mnt/queue/postfix
```

```
# cp -rp /var/spool/postfix/* /mnt/queue/postfix/
# mv /var/spool/postfix /var/spool/postfix.org
# ln -s /mnt/queue/postfix /var/spool/postfix
# postfix check
```

4. メールスプール領域のための共有ファイルシステムをマウント

```
# mkfs.ext3 /dev/sdb1
# mv /var/spool/mail /var/spool/mail.org
# mkdir -p /var/spool/mail
# mount -t ext3 /dev/sdb1 /var/spool/mail
```

5. Postfix の起動

```
# postfix –c /etc/postfix start
postfix/postfix-script: starting the Postfix mail system
```

6. プロセスとソケットの確認

```
# netstat -pltun | grep master
tcp    0   0   127.0.0.1:25      0.0.0.0:*     LISTEN     15931/master
tcp    0   0   192.168.0.10:25   0.0.0.0:*     LISTEN     15931/master

# ps -ef | grep -v grep | grep postfix
root      15931      1   0   16:11   ?   00:00:00   /usr/libexec/postfix/master
postfix   15932   15931   0   16:11   ?   00:00:00   pickup -l -t fifo -u
postfix   15933   15931   0   16:11   ?   00:00:00   qmgr -l -t fifo -u
```

7. Postfix の停止

```
# postfix -c /etc/postfix stop
postfix/postfix-script: stopping the Postfix mail system
```

5～7 の操作により、正しく起動ができなかった場合は Postfix のエラーメッセージ等を参照し、問題を解決してから再度同じ作業を実施して問題がなくなるまで確認します。また、この操作を全てのクラスタシステムのサーバ上で実施し、確認をします。

# 6.13.3.1. LifeKeeper の設定作業

LifeKeeper for Linux Postfix Recovery Kit の管理作業はすべて、LifeKeeper グラフィカル・ユーザ・インタフェース（GUI）で実行できます。LifeKeeper GUI では、Postfix リソースの設定、管理、監視作業をガイド付きで行えます。

LifeKeeper for Linux Postfix Recovery Kit の設定に関しては、次の作業が可能です。

- リソース階層の作成 。Postfix リソース階層を作成します。
- リソース階層の拡張 。Postfix リソース階層をプライマリ・サーバからバックアップ・サーバに拡張します。
- リソース階層の拡張解除 。LifeKeeper クラスタの 1 つのサーバの Postfix リソース階層を拡張解除（削除）します。
- リソース階層の削除 。Postfix リソース階層を削除します。
- 従属関係の作成 。既存のリソース階層と別のリソース・インスタンスとの間に子従属関係を作成し、クラスタ内のすべての対象サーバに従属関係の変化を通知します。
- 従属関係の削除 。リソースの従属関係を削除して、クラスタ内のすべての対象サーバ に従属関係の変化を通知します。
- In Service 。リソース階層をアクティブにします。
- Out of Service 。リソース階層を非アクティブにします。
- 表示 / 編集 。 リソース階層のプロパティを表示または編集します。

**注記：** このセクションの残りの部分では、設定作業はすべて **[Edit]** メニューから実行することになっています。この設定作業の大半は次の方法でも実行できます。

1. ツールバーから状態表示の左ペインにあるグローバル・リソースを右クリック。
2. 状態表示の右ペインにあるリソース・インスタンスを右クリック。

   ※右クリックの方法を使用すると、 **[Edit]** メニューを使用したときに必要な情報を入力しないですみます。

# 6.13.3.1.1. リソース階層の作成

必要なセットアップ作業を完了したあと、Postfix リソース階層を作成するために、以下の手 順を行います。

**重要:** Postfix リソース・インスタンスを作成する前に、仮想 IP アドレスを LifeKeeper の保護下に置く必要があります。

1. **[LifeKeeper GUI]** メニューから **[編集]** を選択し、次に **[サーバ]** を選択します。 ドロップダウンメニューから、 **[リソース階層の作成]** を選択します。

   **[リソース作成ウィザード]** ダイアログ・ボックスが表示され、クラスタにインストールされている全ての Recovery Kit がリストボックスに表示されます。

2. **[Postfix Mail Server]** を選択します。

3. 次の表で説明している情報を入力します。 どのダイアログ・ボックスでも、 **[戻る]** がアクティブになっているときには、直前のダイアログ・ボックスに戻ることができます。 これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。 いつでも **[キャンセル]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| Switchback Type | [インテリジェント]または[自動]を選択します。これによって、バックアップサーバへのフェイルオーバ後、Postfix インスタンスがこのサー バでサービスに戻ったときにどのようにスイッ チバックされるかが決まります。スイッチバック・タイプは、必要な場合、[リソース・プロパティ]ダイアログ・ボックスの[一般]タブで後から変更できます。<br><br>**注記：** スイッチバックの方法は、Postfix リソースで使用するプリントサーバ、IP リソース、フ ァイル・システムリソースのスイッチバック方 法と一致している必要があります。一致していない場合、Postfix リソース作成時に、Postfix リソースに対して選択された設定に一致するように設定がリセットされます。 |
| Server | 階層を作成するサーバを選択します。 |
| Postfix Binary Location | Postfix デーモンが置かれてディレクトリパス名を入力してください。 |
| Postfix server Config File Location | Postfix の設定ファイル(main.cf) が置いてあるディレクトリを入力します。 |
| Queue Root | Postfix キューディレクトリのディレクトリパス名を入力してください。 |

| | |
|---|---|
| Directory | デフォルトは前のウィザードで指定した構成ファイルから決定されます。<br><br>Postfix キューディレクトリは共有ディスクの上になければなりません。Postfix キューディレクトリがシンボリックリンクであるならば、ダイアログ･ボックスはディレクトリの最初の場所を指しているシンボリックリンクのルートディレクトリを示します。 |
| Root Tag | LifeKeeper が示すデフォルト･ルート･タグを選択するか、このサーバ上のリソース･インスタンスの重複しない名前を入力します。デフォルトは postfix-on-<キューディレクトリパス>です。英字、数字、および特殊文字（- _ . /）を使用できます。 |

4. **[作成]** をクリックします。**[リソース作成ウィザード]** が表示され、Postfix リソース階層が作成されます。LifeKeeper によって入力データが検証されます。問題が検出された場合、情報ボックスにエラーメッセージが表示されます。

5. 情報ボックスが表示され、Postfix リソース階層の作成に成功したこと、フェイルオーバによって保護するには、クラスタ内の別のサーバにリソース階層を拡張する必要があることを示すメッセージが表示されます。**[次へ]** をクリックします。

6. **[続行]** をクリックします。**[事前拡張ウィザード]** が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、次の「Postfix リソース階層の拡張 」の手順2 を参照してください。

# 6.13.3.1.2. リソース階層の拡張

この操作は **[編集]** メニューから開始できます。また、**[リソース階層の作成]** の処理を完了すると自動的に開始されます。その場合は、手順2を参照してください。

1. **[編集]** メニューの **[リソース]** から **[リソース階層の拡張]** を選択します。**[事前拡張ウィザード]** が表示されます。拡張操作に慣れていない場合は、**[次へ]** をクリックします。

2. **[事前拡張ウィザード]** で次の情報を入力します。

   **注記：** 最初の 2 つのフィールドは、**[編集]** メニューの **[拡張]** から操作を開始した場合にだけ表示されます。

| Field | Tips |
|---|---|
| Template Server | Postfix リソースが現在サービス中のサーバを入力します。 |
| Tag to Extend | 拡張するPostfix リソースを選択します。 |
| Target Server | 拡張先のサーバを入力または選択します。 |
| Switchback Type | **[インテリジェント]** または **[自動]** を選択します。スイッチバック・タイプは、必要な場合、**[リソース・プロパティ]** ダイアログ・ボックスの **[一般]** タブで後から変更できます。<br><br>**注記：** スイッチバック方法は、Postfix リソースが使用する従属リソースのスイッチバック方法と一致する必要があることを忘れないでください。 |
| Template Priority | テンプレート階層の優先順位を選択または入力します。優先順位は1～999 の範囲の未使用の値が有効で、数値が小さいほど優先順位は高くなります（1は最高の優先順位を表わします）。 拡張操作時には、別のシステムで既に使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。<br><br>**注記** ：これは階層を最初に拡張するときだけ表示されます。 |
| Target Priority | ターゲット・サーバの階層の優先順位を選択、または入力します。 |

3. 拡張前のチェックが成功したというメッセージが表示されたら、**[次へ]** をクリックします。

4. LifeKeeper はPostfix リソースを拡張するための一連のダイアログ・ボックスを示します。そして、その幾つかは編集することはできません。

| Field | Tips |
|---|---|
| Root Tag | ターゲット・サーバの新しいPostfix リソース・インスタンスのデフォルト・タグ名が表示されます。デフォルト・タグ名はテンプレート・サーバ上のこのリソースのタグ名と同じです。新しいタグ名を入力する場合は、ターゲット・サーバ上で重複しない名前にします。英字、数字、および特殊文字（- _ . /）を使用できます。 |
| Binary Directory (Information Only) | このダイアログ・ボックスは、情報の目的だけです。ボックスに現れるBinary Directory を変えることができません。 |
| Configuration Directory (Information Only) | このダイアログ・ボックスは、情報の目的だけです。ボックスに現れるConfiguration Directory を変えることができません。 |

5. 「Hierarchy extend operations completed (階層の拡張操作が完了しました)」という メッセージが表示されたら、階層を別のサーバに拡張する場合は、**[次のサーバ]** を クリックし、拡張作業が必要ない場合は、**[完了]** をクリックします。

6. 「Hierarchy Verification Finished (階層の検査が完了しました)」というメッセージ が表示されたら、**[完了]** をクリックします。

# 6.13.3.1.3. リソース階層の拡張解除

LifeKeeper クラスタの1 つのサーバのリソース階層を削除するには、次の手順を実行します。

1. **[編集]** メニューの **[リソース]** から **[リソース階層の拡張解除]** を選択します。
2. Postfix リソースを拡張解除する **[ターゲット・サーバ]** を選択します。リソースが現在サービス中の Postfix サーバは選択できません。（このダイアログ・ボックスは、右側のペ インでリソース・インスタンスを右クリックして、**[拡張解除]** を選択した場合には、表示されません。）**[次へ]** をクリックします。

3. 拡張解除するPostfix 階層を選択し、**[次へ]** をクリックします。（このダイアログ・ボックスは、いずれかのペインでリソース・インスタンスを右クリックして、**[拡張解除]** を選択した場合には、表示されません。）

4. 拡張解除するために選択したターゲット・サーバとPostfix リソース階層を確認する情報ボックスが表示されます。**[拡張解除]** をクリックします。

5. Postfix リソースの拡張解除に成功したことを示す別の情報ボックスが表示されます。**[完了]** をクリックして、**[リソース階層の拡張解除]** メニューを終了します。

# 6.13.3.1.4. リソース階層の削除

Postfix　階層を削除すると、従属関係および保護されているサービスに何が起きるかを理解しておくことが重要です。

- **依存関係:** Postfixリソース階層を削除するとき、Postfixリソースだけは削除されます。従属するIPとファイルシステム資源は削除されません。

- **保護されたサービス:** Postfixリソースを削除する前にPostfixリソースを停止した場合、リソースを削除した後Postfixデーモンは停止します。 Postfixリソースの動作中にリソースを削除した場合、Postfixデーモンは動き続けます。リソース階層の削除をした後はLifeKeeperの保護対象となりません。

LifeKeeper環境内のすべてのサーバからリソース階層を削除するには、以下の手順を実行します。

1. **[編集]** メニューの **[リソース]** から **[リソース階層の削除]** を選択します。

2. Postfixリソース階層を削除する **[ター ゲット・サーバ]** を選択し、 **[次へ]** をクリックします。（このダイアログ・ボックスは、いずれかのペインでリソース・インスタンスを右クリックして、 **[リソースの削除]** を選択した場合には、表示されません。）

3. **[削除する階層]** を選択します。このダイアログ・ボックスは、左側または右側のペインでリソース・インスタンスを右クリックして、 **[リソースの削除]** を選択した場合には、表示されません。） **[次へ]** をクリックします。

4. 選択したターゲット・サーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[削除]** をクリックします。

5. Postfixリソースの削除に成功したことを示す別の情報ボックスが表示されます。

6. **[完了]** クリックして終了します。

# 6.13.3.1.5. スプール (メールボックス) の依存関係作成

Postfix のキューディレクトリとスプール(メールボックスディレクトリが共有ストレージの同じファイルシステム(LUN)上である場合、Postfix リソース階層の作成、Postfix リソース階層の拡張により両ディレクトリは保護対象となります。スプール(メールボックス)ディレクトリが共有ストレージの別のファイルシステム(LUN)上である場合、 スプール(メール ボックス) ディレクトリ配下のファイルシステムを LifeKeeper により保護対象とし、作成されたファイル・システムリソースと、Postfix リソース階層と依存関係を作成する必要があります。

以下のような手順で行います。

1. 対象のファイルシステムをマウントする
2. LifeKeeper GUI の **[Edit]** 、 **[Server]** 、 **[Create Resource Hierarchy]** を選択
3. **[Select Recovery Kit]** で **[File System]** を選択
4. **[Switchback Type]** スイッチバック・タイプを選択
5. **[Server]** プライマリ・サーバを選択
6. **[Mount Point]** マウントポイントを選択
7. **[Root Tag]** タグの名前を入力

これによりプライマリ・サーバでファイル・システムリソースが作成されるため、バックアップ・サーバにも拡張します。そしてこの作成されたファイル・システムリソースを Postfixのリソース階層内に依存させます。Lifekeeper for Linux テクニカルドキュメンテーションの依存関係の作成 を参照してください。

# 6.13.3.1.6. リソース階層のテスト

Postfix リソース階層のテストは、手動でサーバを切り替えて、プライマリ・サーバからバックアップ・サーバへのリソース・インスタンスのフェイルオーバをシミュレートすることによって行えます。

## LifeKeeper GUI からの手動切り替えの実行

LifeKeeper GUI で **[Edit] > [Resource] > [In Service]** を選択すると、手動切り替えを開始できます。たとえば、バックアップ・サーバでサービス中リクエストが実行されると、NAS リソース階層がバックアップ・サーバ側でサービス中になり、プライマリ・サーバ側ではサービス休止になります。この時点で、元のバックアップ・サーバがプライマリ・サーバになり、元のプライマリ・サーバがバックアップ・サーバになります。

**[Our of Service]** を実行した場合、リソース階層はサービス休止になりますが、別のサーバでサービス中にはなりません。

## コマンドラインインターフェースによる手作業の切り替えの実行

LifeKeeper GUI による切り替え操作を実施する場合は、以下のコマンドをその操作を実施したいシステム上で入力します。

```
# /opt/LifeKeeper/bin/perform_action -t [tag-name] -a [restore|remove]
```

- **-t**

  起動や停止をする LifeKeeper により保護されているサービスのタグ名を指定します。タグ名は LifeKeeper GUI 画面や、lcdstatus コマンドにより確認することができます。

- **-a**

  実行されるリソースのアクションを指定します。起動を実施する場合 restore、停止する場合 remove とします。

perform_action コマンドの詳細については man オンラインマニュアルをご参照ください。

## リカバリ動作

LifeKeeper はサービスを提供中のシステム(プライマリ・サーバ) で以下のような障害が発生した場合、リカバリ動作を実施します。

- Postfix サービス障害 (Postfix リソース)
- IP アドレス障害 (Postfix リソースに依存するIP リソース)
- 共有ファイルシステム障害(Postfix リソースに依存するファイル・システムリ ソース)

- ノード障害 (サーバのハングアップや、電源断等)

プライマリ・サーバに障害が発生すると、Postfix Mail Server Recovery Kit は次の作業を実行します。

- バックアップサーバの物理ネットワークインタフェースに対し論理インタフェースをサービス中にすることによって、バックアップサーバでエイリアス IP(仮想 IP) アドレスをサービス中にします。
- バックアップサーバで共有ディスク上のファイルシステムをマウントします。
- Postfix 関連のデーモン・プロセスを開始します。

リカバリ後はセッション・コンテキストが失われるため、Postfix ユーザは最初に接続し たときと正確に同じ手順を使用して再接続する必要があります。

# 6.13.4. トラブルシューティング

この項には、LifeKeeper Postfix 階層の作成、拡張、削除、復元処理時に表示される可能性のあるメッセージの一覧が記載されています。また、該当する場合には、エラーの原因とエラーの解決に必要な処置についても説明されています。他のLifeKeeperスクリプトやユーティリティがメッセージを表示することもありえます。その場合は、そのスクリプトやユーティリティのマニュアルを参照してください。この項ではメッセージを次のように分類しています。

リソース階層作成時のエラーメッセージ

リソース階層拡張時のエラーメッセージ

サービスを起動、停止、回復時に出力されるエラーメッセージ

# 6.13.4.1. リソース階層作成時のエラーメッセージ

| エラー | エラーメッセージ |
|---|---|
| No config path | コンフィグレーションパスが指定されていません。コンフィグレーションパスを指定してください。 |
| main.cf not found in the configuration path | 指定されたコンフィグレーションパスにはmain.cf が見つかりません。パスを見直し、正しいパスを指定してください。 |
| master.cf not found in the configuration path | 指定されたコンフィグレーションパスには master.cf が見つかりません。パスを見直し、正しいパスを指定してください。 |
| A value of inet_interfaces must be IPv4 or “all” | main.cf 内の inet_interfaces パラメータには IPv4 アドレスまたはall を指定してください。 |
| No execute path | 実行パスが指定されていません。postfix のコマンドが置かれている実行パスを指定してください。 |
| Postfix command invalid | 指定された実行パスには postfix が見つからないか、有効ではありません。postfix のインストール状況および、コマンドについて確認し正しい実行パスを指定してください。 |
| *<queue directory>* is not found. This directory must exist on a shared filesystem | 指定されたキューディレクトリは共有ファイルシステム上にありません。共有ファイルシステム上にキューを保持するように設定してください。 |
| *<tag name>* not in service on the server | 選択されたタグ名はサーバ上でアクティブではありません。IPリソースを作成し、仮想 IP アドレスがサーバ上でアクティブとなっているかを確認してください。 |
| Could not find IP resouce for “*<IP address>*” | main.cf 内の inet_interfaces パラメータに指定された IP address の IP リソースが見つかりません。IP リソースを作成してください。 |

# 6.13.4.2. リソース階層拡張時のエラーメッセージ

| エラー | エラーメッセージ |
| --- | --- |
| postfix id does not match between servers | postfixのオーナーID がクラスタサーバ間において一致していません。postfix のオー ナーID が全てのサーバで一致するよう設定してください。 |
| postdrop gid does not match between servers | postdrop のグループID がクラスタサーバ間において一致していません。postdrop の グループID が全てのサーバで一致するよう設定してください。 |

# 6.13.4.3. サービスを起動、停止、回復時に出力されるエラーメッセージ

| エラー | エラーメッセージ |
|---|---|
| Master process of postfix is not running | postfixのmasterデーモンが起動していません。 maillogなどからpostfixの動作に異常が生じていなかったか確認してください。 |
| Failed in a check by postfix command | postfixのcheckオプションによるエラーが発生しました。 postfixの環境および、設定内容を確認してください。 |
| Couldn't start postfix resource | postfixのサービスを起動することができませんでした。 maillogなどからpostfixの動作に異常が生じていなかったか確認してください。 |
| Failed in a stop process by kill command | postfixの停止時にkillコマンドによる停止を試みましたが失敗しました。 maillogなどからpostfixの動作に異常が生じていなかったか確認してください。 |
| PID <*pid*> does not exist. postfix may have already stopped | 起動されているはずのpostfixのプロセスIDは存在していません。 maillogやpostfixのプロセス稼動状況を確認してください。既に停止されているか、別のIDにて起動されているなどが原因として考えられます。 |
| Check script was not able to be connected to a socket *(vip:port)* | サービスが待ち受けているソケットに接続することができませんでした。postfixの設定内容を確認してください。 |
| Execute files (postfix or postconf command) is not an executable file | postfixまたはpostconf実行ファイルが存在しない、もしくは実行権がありません。 リソース作成時に指定した実行パス内にて、これらの実行ファイルを確認してください。 |
| Configuration files (main.cf or master.cf) does not exist | main.cfまたはmaster.cf実行ファイルが存在しない、もしくは実行権がありません。 リソース作成時に指定したコンフィグレーションパス内にて、これらのファイルを確認してください。 |
| The postfix owner <*owner name*> does not exist | postfixのオーナーがサーバ上に存在していません。 |
| The postdrop group id does not match an attribute of queue directory | postdropのグループIDがキューディレクトリ内maildropのgidと一致していません。　postfixの設定内容ついて確認してください。 |

# 6.14. Recovery Kit for Route 53 管理ガイド

## Recovery Kit for Route 53

Recovery Kit for Route 53は、障害の発生したプライマリサーバからバックアップサーバに復旧する際、依存関係にあるIPリソースの仮想IPアドレス・実IPアドレス情報に対応するAmazon Route 53 DNS 情報を更新する仕組みを提供します。

## SIOS Protection Suite ドキュメンテーション

以下は、SIOS Technology Corp が発行している SIOS Protection Suite for Linux 関連ドキュメントの一覧です。

- SPS for Linux テクニカルドキュメンテーション
- SPS for Linux リリースノート
- SIOS Technology Corp. ドキュメンテーション

詳細については、Amazon Route 53 ドキュメント を参照してください。

「Amazon Web Services」、「Powered by Amazon Web Services」のロゴ、「AWS」、「Amazon EC2」、「EC2」、「Amazon Elastic Compute Cloud」、「Amazon Route 53」、「Amazon Virtual Private Cloud」および「Amazon VPC」は、米国その他の国における Amazon.com, Inc. またはその関連会社の商標です。なお、本文および図表中では、「™」、「®」は明記しておりません。(一部記載がございます。)

# 6.14.1. 要件

Recovery Kit for Route 53 をインストールして設定する前に、使用している設定が以下の要件を満たしていることを確認してください。

- AWS Command Line Interface (AWS CLI) を全ての EC2 インスタンスにインストールする必要があります。インストール方法は、「AWS Command Line Interface のインストール」を参照してください。
- Amazon Route 53 サービスのエンドポイント route53.amazonaws.com にプロトコル HTTPS を使用してアクセスできる必要があります。EC2 および OS の設定を適切に行ってください。
- Amazon Route 53 に適切なドメイン名を登録してください。
- LifeKeeperがAWSを操作するために、以下のアクセス権限を持ったIAM ユーザーもしくはIAM ロールが必要です。Amazon EC2 インスタンスの root ユーザーからアクセスできるようにEC2の IAM ロール を設定するか、AWS CLI の設定 を適切に行ってください。
  - route53:GetChange
  - route53:ListHostedZones
  - route53:ChangeResourceRecordSets
  - route53:ListResourceRecordSets

**注記:** 設定ファイル `/etc/default/LifeKeeper` のパラメーター PATH に AWS CLI 実行ファイルのパスが設定されていない場合、PATH に AWS CLI 実行ファイルのパスを追加してください。

## LifeKeeper ソフトウェア:

各サーバに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。具体的な LifeKeeper の要件については、SPS for Linux テクニカルドキュメンテーション およびSPS for Linux リリースノート を参照してください。

Recovery Kit for Route 53 を使う場合には、Quorum/Witnessの使用を推奨します。
詳細については、こちら を参照してください。

# 6.14.2. 設定

必要とする保護と柔軟性が得られるように LifeKeeper を設定するには、設定要件を把握している必要があります。設定を適切にプランニングするには、Amazon、Amazon Virtual Private Cloud (VPC)、Amazon Elastic Compute Cloud (EC2)、およびユーザシステムの階層設定オプションを理解する必要があります。このセクションでは、設定のプランニングに加え、Recovery Kit を設定するために必要な特定の作業についても説明します。

## Route53 リソース 特有の設定上の考慮事項

このセクションでは、Route53 リソースの以下の設定タスクについて説明します。設定タスクは Route53 リソースインスタンスに特有のものであり、Recovery Kit ごとに異なります。

- リソース階層の作成: アプリケーションリソース階層を LifeKeeper クラスタに作成します。
- リソース階層の削除: リソース階層を LifeKeeper クラスタ内のすべてのサーバから削除します。
- リソース階層の拡張: リソース階層をプライマリサーバからバックアップサーバへ拡張します。
- リソース階層の拡張解除: リソース階層を LifeKeeper クラスタ内の 1 つのサーバから拡張解除 (削除) します。
- Recovery Kit for Route 53のチューニング: Recovery Kit for Route 53 の動作を調整します。

以下の作業は、SPS for Linux テクニカルドキュメンテーション の管理 セクションに記載されています。これらは、すべての Recovery Kit で同じ手順を使用する共通の作業です。

- リソース依存関係の作成: 既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除: リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service: リソース階層を特定のサーバで In Service にします。
- Out of Service: リソース階層を特定のサーバで Out of Service にします。
- プロパティの表示 / プロパティの編集: 特定のサーバでリソース階層のプロパティを表示または編集します。

このセクションの残りの部分では、LifeKeeper GUI の [Edit] メニューから作業を選択することによって、Recovery Kit を設定する方法を説明します。設定作業はツールバーから選択することもできます。

- ◦ 状況表示ウィンドウのリソース階層ツリー (左側のペイン) のグローバルリソースを右クリックすると、[Edit] メニューと同じドロップダウンメニューの選択項目が表示されます。これは階層がすでに存在している場合にだけ可能な方法です。

- ◦ 状況表示ウィンドウのリソース階層表 (右側のペイン) のリソースインスタンスを右クリックすると、サーバおよび特定リソースの状況に応じて、リソース階層の作成を除くすべての設定作業を実行できます。

# 6.14.2.1. リソース階層の作成

プライマリサーバからリソースインスタンスを作成するには、以下の手順を完了する必要があります。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択してください。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択してください。

2. ダイアログボックスが表示され、クラスタ内にインストール済みの認識されている Recovery Kit がすべて、ドロップダウンリストボックスに表示されます。ドロップダウンリストから **[Amazon Route53]** を選択し、 **[Next]** をクリックしてください。

3. 以下の情報を入力するようにプロンプトが表示されます。(ダイアログボックスで [Back] ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、以前に入力した情報を訂正する必要がある場合に特に役立ちます。)

   **注記:** 階層作成の途中で [Cancel] ボタンをクリックすると、作成処理全体が取り消されます。

| フィールド | ヒント |
|---|---|
| Switchback Type | フェイルオーバ後にサーバが復帰したときに、このサーバに EC2 インスタンスを戻す方法を指定します。[intelligent] または [automatic] を選択できます。<br>• <br>◦ [intelligent] の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。<br>◦ [automatic] の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。<br><br>**注記:** スイッチバックタイプは、必要な場合、[Resource Properties] ダイアログボックスの [General] タブで後から変更できます。 |
| Server | [Server] で Route53 リソースの作成先のサーバ (通常、これをプライマリサーバまたはテンプレートサーバと呼ぶ) を選択します。クラスタ内のすべてのサーバがドロップダウンリストに表示されます。 |
| Domain name (Route53 hosted zone) | Route53のホストゾーンがドロップダウンリストに表示されます。使用するドメイン名を選択してください。 |
| Host Name (Not FQDN) | 使用するホスト名を入力してください。 |
| IP resource | IP リソースを選択してください。これは LifeKeeper が保護する仮想 IP アドレスもしくは実IPアドレスです。 |
| Route53 Resource Tag | 作成している Route53 リソースインスタンスに固有の Route53 リソースタグ名を選択または入力します。このフィールドには、デフォルトのタグ名である route53-<host name> |

| | |
|---|---|
| | が自動的に表示されます |

4. [Create] をクリックしてください。[Create Resource Wizard] によって、Route53 リソースが作成されます。

5. この時点で情報ボックスが表示され、LifeKeeper は、Route53 リソース階層を作成するために有効なデータが指定されたかどうかを検証します。LifeKeeper が問題を検知した場合は、情報ボックスにエラーが表示されます。検証が正常に完了すると、リソースが作成されます。 **[Next]** をクリックしてください。

Route53 リソース階層が正常に作成されたことを示す情報ボックスが表示されます。リソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにそのリソース階層を拡張する必要があります。

**[Continue]** をクリックすると、[Pre-Extend configuration task] が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、リソース階層の拡張 を参照してください。

ここで **[Cancel]** をクリックすると、別のダイアログボックスが表示され、後で Route53 リソース階層を別のサーバに手動で拡張して LifeKeeper の保護下に置く必要があることが警告されます。

# 6.14.2.2. リソース階層の削除

LifeKeeper 環境のすべてのサーバからリソース階層を削除するには、次の手順を実行してください。

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Resouce]** を選択してください。ドロップダウンメニューから **[Delete Resource Hierarchy]** を選択してください。

2. Route53 リソース階層から削除するターゲットサーバの名前を選択し、 **[Next]** をクリックしてください。

   **注記:** このダイアログボックスは、いずれかのペインでリソースインスタンスを右クリックして [Delete Resource] を選択した場合には、表示されません。

3. [Hierarchy to Delete] を選択してください。削除するリソース階層を指定して強調表示にしてから、**[Next]** をクリックしてください。

   **注記:** このダイアログボックスは、左右どちらかの インでリソースインスタンスを右クリックして [Delete Resource] を選択した場合には、表示されません。

4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[Delete]** をクリックして次に進んでください。

5. Route53 リソースが正常に削除されたことを確認する情報ボックスが表示されます。

6. **[Done]** をクリックして終了してください。

# 6.14.2.3. リソース階層の拡張

階層の作成後、クラスタ内の別のサーバに拡張する必要があります。3 通りのシナリオで、テンプレートサーバからターゲットサーバにリソースインスタンスを拡張できます。

- リソースの作成後、[Continue] をクリックして、別のサーバにリソースを拡張します。
- 次に説明するように [Edit] メニューから [Extend Resource Hierarchy] を選択します。
- 左側または右側のペインから拡張されていない階層を右クリックします。

どのシナリオでも同じダイアログボックスが表示されます (いくつかの例外については、以下に詳細を明記)。

1. LifeKeeperGUI メニューから [Extend] ウィザードを開始する場合は、 **[Edit]** を選択し、次に **[Resource]** を選択します。ドロップダウンメニューから **[Extend Resource Hierarchy]** を選択します。これで [Extend Resource Hierarchy] ウィザードが起動されます。拡張操作に慣れていない場合は、 **[Next]** をクリックします。LifeKeeper の [Extend Resource Hierarchy] のデフォルト値が分かっていて、入力と確認を省略する場合は **[Accept Defaults]** をクリックします。

2. [Pre-Extend Wizard] で以下の情報を入力します。

   **注記:** 最初の 2 つのフィールドは [Edit] メニューの [Extend] から操作を開始した場合にだけ表示されます。階層を拡張する手順の間に [Cancel] をクリックすると、どの時点であってもそのサーバへの拡張処理がキャンセルされるので注意してください。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

| フィールド | ヒント |
|---|---|
| Switchback Type | スイッチバックタイプを選択します。ここでは、バックアップサーバへのフェイルオーバの後、Route53 インスタンスが In Service に戻ったときに、このサーバにRoute53 インスタンスをスイッチバックする方法を指定します。[intelligent] または [automatic] を選択できます。<br><br>* intelligent の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。<br><br>* automatic の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。<br><br>スイッチバックタイプは、必要な場合 [Resource Properties] ダイアログボックスの [General] タブで後から変更できます。 |
| Template Priority | テンプレートの優先順位を選択するか、入力してください。これはサーバで現在 In Service |

| | |
|---|---|
| | の Route53 階層の優先順位です。1 ～ 999 の範囲で、まだ優先順位として使用されていない値が有効で、小さい数値ほど優先順位が高くなります (数値 1 が最高の優先順位)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を使用することを推奨します。<br><br>**注記:** このフィールドは、階層を最初に拡張するときにだけ表示されます。 |
| Target Priority | ターゲットの優先順位を選択するか、入力してください。これは、他のサーバにある同等の階層に対する、新しく拡張する Route53 階層の優先順位です。1 ～ 999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位が高くなります (数値 1 が最高の優先順位)。<br><br>**注記:** LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられます。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |

3. 情報ボックスが表示され、LifeKeeper が環境のチェックを正常に完了し、この EC2 リソースを拡張するためのすべての要件が満たされていることが示されます。満たされていない要件がある場合は、 **[Next]** ボタンは選択できなくなり、 **[Back]** ボタンが有効になります。 **[Back]** をクリックした場合、情報ボックスに表示されるエラーメッセージの内容に従って、リソースの拡張を変更できます。ここで **[Cancel]** をクリックした場合は、EC2 リソース階層を他のサーバに手動で拡張して、LifeKeeper の保護下に置く必要があります。 **[Next]** をクリックすると、[Extend Resource Hierarchy] 設定作業に入ります。

4. リソース階層構造の拡張を行うために、以下の情報の入力が要求されます。

| フィールド | ヒント |
|---|---|
| Route53 Resource Tag | Route53リソースタグを選択するか、入力してください。これは、ターゲットサーバに拡張される Route53 リソースが使用するリソースタグ名です。<br><br>**注記:** このフィールドは編集できません。 |

5. 拡張が実行中であることを示す情報ボックスが表示されます。同じ Route53リソースインスタンスをクラスタ内の別のサーバに拡張する場合は、 **[Next Server]** をクリックしてください。その場合は、リソース階層を拡張する操作を繰り返します。 **[Finish]** をクリックすると、LifeKeeper は Route53 リソースの拡張が正常に完了したことを確認します。

6. **[Done]** をクリックして、 **[Extend Resources Hierarchy]** メニューの選択を終了します。

   **注記:** 必ずすべてのサーバで新しいインスタンスの機能をテストしてください。

# 6.14.2.4. リソース階層の拡張解除

階層全体を拡張解除するには、以下の手順を実行してください。

1. **LifeKeeper GUI メニュー** から **[Edit]** を選択し、次に **[Resouce]** を選択してください。ドロップダウンメニューから **[Unextend Resource Hierarchy]** を選択してください。

2. EC2 リソースから、拡張解除の対象となるターゲットサーバを選択してください。 Route53 リソースが現在 In Service になっているサーバは選択できません。 **[Next]** をクリックしてください。

   **注記:** 右側のペインから個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

3. 拡張解除する Route53 階層を選択してください。 **[Next]** をクリックしてください。

   **注記:** 左側のペインにあるグローバルリソースを右クリックするか、右側のペインにある個々のリソースインスタンスを右クリックして **[Unextend]** 作業を選択した場合、このダイアログボックスは表示されません。

4. 拡張解除の対象として選択したターゲットサーバと Route53 リソース階層を確認する情報ボックスが表示されます。 **[Unextend]** をクリックしてください。

5. EC2 リソースが正常に拡張解除されたことを示す情報ボックスが表示されます。

6. **[Done]** をクリックして終了してください。

# 6.14.2.5. Recovery Kit for Route 53 のチューニング

Recovery Kit for Route 53 で設定可能なパラメータは パラメータ一覧 を参照してください。

# 6.14.2.6. リソースの監視とリカバリ動作

Route53 リソースは作成時に登録した DNS A レコードの取得と仮想IPアドレスとの関連づけの正常性を監視しています。リソースの監視内容として、以下のような処理を行っています。

1. Route 53 の A レコードで設定されているアドレスを API で取得します。取得に失敗した場合は、デフォルトで 2 秒の間隔を置いてから、情報の再取得を行います。
   3回取得を試行して取得できなかった場合、ログに記録を残し監視処理を終了します。

2. Route53リソースと依存関係がある IP リソースから IP アドレスを取得して、 Route53 の A レコードで設定されているアドレスと IP リソースの IP アドレスを比較します。
   一致した場合、正常であると判断して処理を正常終了します。一致しなかった場合、異常であると判断しリカバリを開始します。

# 6.14.2.7. ユーザシステムのセットアップ

Route53 リソース作成の際必要とするIPリソースは、仮想IPリソースと実IPリソース（NIC 用に構成されたプライマリ IP アドレスのためのリソース）のいずれかを使用することができます。

## 仮想IPリソースを使用する場合

Route53リソースの子リソースに仮想IPリソースを使用する場合、リソースのスイッチオーバー時に仮想IPアドレスに対する通信がバックアップサーバに対して行われるよう、ルートテーブルを再設定する必要があります。
そのため、Recovery Kit for EC2 を併用してください。詳細はRecovery Kit for EC2のマニュアルを参照してください。

## 実IPリソースを使用する場合

Route53リソースの子リソースに実IPリソースを使用する場合、追加で設定する項目はありません。
ただしスイッチオーバー毎に接続先IPアドレスが変更されてしまうため、接続には必ずRoute53リソースが保護しているホスト名で接続を行わなければならないことに注意してください。

# 6.14.3. トラブルシューティング

メッセージカタログ は、SIOS Protection Suite for Linux の使用時に検出されるすべてのメッセージのリストを示すとともに、適宜、エラーの原因とエラー状態を解決するために必要な処置の説明を提供します。受け取ったエラーコードをこのリストから探すことができます。

## Route53リソース起動にともなうレコードの更新に時間がかかる場合があります

Route53のDNSレコードの更新の伝播速度に関して、Amazonでは以下のような情報を提供しています。

Amazon Route 53 よくある質問
Q: Amazon Route 53 での DNS 設定の変更が全体に適用されるには、どのくらいの時間がかかりますか?
http://aws.amazon.com/jp/route53/faqs/

Route53リソースではRoute53 APIを使用してDNSへのレコード更新の状況について確認しています。INSYNCステータスを得られた場合には更新が完了したと判断し、PENDINGステータスが帰ってきた場合には、更新状況の確認のリトライを行います。このような仕様のため、Route53リソースの起動処理としては正しくレコード更新がかけられているにも関わらず、DNSへの更新情報の伝播に時間がかかった場合には、Route53リソースとしては起動失敗の状態になってしまう場合があります。

もし、Route53リソースの起動に失敗した場合には、Route53の管理コンソールをみてAレコードが正しく更新されていることを確認してください。更新されている場合、該当するDNSサービスの更新は完了しており、DNSサービスの更新を反映するには、LifeKeeperのみ更新が必要となります。もう一度LifeKeeper GUIからRoute53リソースを起動してください。

常時前述のようなケースでRoute53リソースの起動ステータスが失敗する場合には、/etc/default/LifeKeeperファイルの「ROUTE53_CHANGEID_TRY_COUNT」の値をデフォルトの５回から１～２回程度値を増やして、状況が改善するかご確認ください。この設定変更にはLifeKeeperやOSの再起動は必要ありません。

## DNSレコードのTTLの値を適切に設定してください

スイッチオーバーやフェイルオーバーが発生した後のクライアントからのアクセスは、TTLの設定によって決められた時間が過ぎるまで各クライアントが保持するDNS情報のキャッシュを用いて行います。そのため、TTLの値が長く設定されている場合、切り替わり前のアドレスへのアクセスが発生する機会が増えることとなり、予期せぬ問題が生じる可能性があります。また、TTLの値が短く設定されている場合、頻繁に名前解決が行われるので、ネットワークへの負荷がかかります。環境に応じて、可能な限り短くTTLの値を設定してください。

TTLの設定は、/etc/default/LifeKeeper ファイルで「ROUTE53_TTL」パラメータを設定してください。単位は秒です。

# 6.15. Samba Recovery Kit 管理ガイド

## Samba Recovery Kit テクニカルドキュメンテーション

LifeKeeper for Linux Samba Recovery Kit は異種ネットワーク環境にある Linux サーバ上で 動作する Samba のファイルとプリンタの共用を保護し耐障害性を高めます。この Kit により、プライマリ Samba サーバで障害が発生しても、わずかな時間で人手を介さずに指定されたバックアップ･サーバに現状を回復することができます。

## ドキュメントの内容

本書は以下のトピックについて説明しています。

- ドキュメンテーションとリファレンス 。LifeKeeper for Linux のドキュメントと所在の一覧です。
- 要件 。Samba Recovery Kit を適切に設定、導入、運用するうえで必要なハードウェアとソフトウェアについて説明しています。LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SIOS Protection Suite インストールガイド を参照してください。
- Samba Recovery Kitの概要 。Samba Recovery Kit の特徴と機能について簡単に説明しています。
- LifeKeeper for Linux Samba Recovery Kit の設定 。Samba Recovery Kit を適切に設定するうえで必要な手順について説明しています。
- LifeKeeper 設定作業 。LifeKeeper GUI を使用して Samba リソース階層を作成し管理する作業について説明しています。
- Samba 階層の管理 。Samba リソース作成後に必要になる可能性がある作業について説明しています。
- トラブルシューティング 。LifeKeeper for Linux のエラー･メッセージの一覧とその説明です。

## ドキュメンテーションとリファレンス

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- SIOS Protection Suite インストールガイド
- Optional Recovery Kit ドキュメンテーション

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

# 6.15.1. 要件

SPS for Linux Samba Recovery Kit をインストールする前に、SPS の構成が次の要件を満たしている必要があります。SPS for Linux ハードウェアおよびソフトウェアの設定に関する具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

- サーバー 。Recovery Kit は、SPS for Linux インストールガイド で説明されている要件に従って 2 つ以上のサーバーを設定する必要があります。対応している Linux ディストリビューションについては、SPS for Linux リリースノート を参照してください。

- **LifeKeeper ソフトウェア** 。各サーバーに同じバージョンの LifeKeeper ソフトウェアと パッチをインストールする必要があります。

- **LifeKeeper IP Recovery Kit** 。各サーバーの LifeKeeper IP Recovery Kit のバージョンは同 じにする必要があります。

- **IP ネットワークインタフェース** 。 各サーバーには、イーサーネットTCP/IP をサポートするネットワークインタフェースが 1 つ以上必要です。IP 切り替えが正しく機能するためには、ローカルネットワークに接続したユーザーシステムが標準TCP/IP 仕様に準拠している必要があります。

  **注記** ：各サーバーが必要とするネットワーク･インタフェースが 1 つだけであっても、異種メディア要件、スループット要件、シングル･ポイント･オブ･フェイルの排除、ネットワークのセグメンテーションといった多くの理由のために、複数のインタフェースを使用すべきです。

- **TCP/IP ソフトウェア** 。各サーバーは TCP/IP ソフトウェアも必要とします。

- **Samba ソフトウェア** 。Samba は LifeKeeper for Linux がサポートしているすべての Linux ディストリビューションに付属しています。付属しているバージョンの Samba を使用してください

# 6.15.2. Recovery Kit のインストール

LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SPS for Linux インストールガイド を参照してください。

# 6.15.3. Samba Recovery Kit の概要

Samba は、SMB (Server Message Block) プロトコルを使用するアプリケーションの集まりで、異種ネットワークで Microsoft Windows 製品の動作するサーバおよびクライアント と Linux サーバとの通信を可能にします。

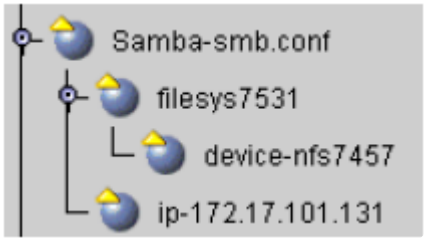
Samba Recovery KitはLinux サーバ上の Samba によるファイルとプリンタの共用を保護します。Samba には、クライアントの認証、ネットワーク・コンピュータの参照支援、WINS ネーム・サーバ解決など、その他のサービスも用意されていますが、LifeKeeperの今回のリリースでは、これらの追加サービスは保護されません。これらのSambaサービスは、

「Samba の複数インスタンスの実行 」のセクションで示されている規則にのっとっている限り、保護されていないSamba インスタンスとしてLifeKeeperサーバ上で共存すること も可能です。

Samba Recovery Kit は、保護された Samba のファイルとプリンタの共用を障害の発生したプライマリ・サーバからバックアップ・サーバに回復する仕組みを提供します。LifeKeeper は、サーバ・レベル（ハートビート経由）またはリソース・レベル（Samba デーモンを監視）で障害を検出して、Samba リソースの制御をバックアップ・サーバに移します。

## Samba リソース階層

一般的な Samba 階層は、1 つの Samba リソース、1 つ以上のファイル・システム・リソ ース、1 つ以上の IP リソースから構成され、1 つのプリント・サービス・リソースが存在することもあります。Samba のファイルの共用を保護しているリソース階層の例を次に示します。




この Samba-smb.conf 階層は 1 つのファイルの共用 filesys7531 (これはパーティション device-nfs7457 に従属しています) と 1 つの IP アドレス 172.17.101.131 を保護しています。次のセクションでは Samba リソースの設定方法について説明します。

# 6.15.4. LifeKeeper for Linux Samba Recovery Kit の設定

LifeKeeper for Linux の Samba リソース階層を作成する前に検討する必要がある設定上の検討事項がいくつかあります。Linux サーバ上の Samba サービスは **smbd** および **nmbd** という 2 つのデーモン・プロセスによって提供されます。このデーモン・プロセスは以下に説明する Samba 設定ファイルで定義された値によって制御されています。

# 6.15.4.1. Samba 設定ファイル

Samba 設定ファイルにはさまざまな命令を多数記述できますが、ここでは LifeKeeper の設定に影響を及ぼす項目に絞って説明します。設定ファイルに関しては、次のようにい くつか重要な点があります。

- 設定ファイルは提供する共用 (すなわちサービス) に対応するセクションから構成 されます。設定ファイルの各セクションには、その共用に特有の設定オプション(すなわち命令) が記述されます。
- 指定された命令は Samba Recovery Kit によって妥当性が検査されます。命令が適切に 設定されていない場合は、Samba リソースの作成に失敗します。
- Samba のデフォルト設定ファイルは、通常、smb.conf という名前で、/etc または /etc/samba（Linux ディストリビューションによって異なる）内にあります。
- 設定ファイルの名前はクラスタ内で重複してはなりません。また、アクティブ／アクティブ設定の場合は、各サーバの異なるディレクトリ内にある必要があります。名前または場所が重複してはいけないのは、Samba Recovery Kit がバックアップ・サーバの同じ場所に拡張するときに、設定ファイルのコピーが作成されるためです。
- アクティブ／スタンバイ設定で Samba デーモンのインスタンスを 1 つだけ実行する 場合は、デフォルト設定を使用できます（これを推奨します）。この場合、Samba デーモンの自動起動を無効にする必要があります。
- アクティブ／スタンバイ設定で複数のSambaインスタンスを実行する場合、またはアクティブ／アクティブ設定を使用する場合は、Samba設定ファイルの名前を重複させることはできません。Sambaの複数インスタンスを実行する場合の要件と詳細については、「Sambaの複数インスタンスの実行 」を参照してください。

次のセクションでは、LifeKeeper で Samba の共用を保護するのに必要なオプションも含めて、設定ファイルのセクションについて説明します。

# 6.15.4.2. 設定ファイルの [global] セクション

[global] セクションは、LifeKeeper Samba リソース階層で使用するすべての設定ファイルに必須の特別なセクションです。名前が示すように、このセクションで設定したオプションは、他のセクションでそのオプションが明示的に指定されない限り、他のすべてのセクションに適用されます。LifeKeeper では、特定の命令を[global]セクションで定義する必要があります。そのうち一部の命令はデフォルトの設定ファイルには記述されていないため、ユーザが追加する必要があります。オプションは次のとおりです。

- **netbios name** - LifeKeeper Samba階層を構成するリソース・セットに付ける一意の名前。クライアントはこの名前を使用して、interfaces命令で定義されたIPアドレス経由で共用に接続します (たとえば、NetBIOS name = server1_print1)。

- **interfaces** - Linux Samba サーバが認識し応答するネットワーク・アドレスの一覧。interfaces 命令を適切に設定するうえで必要な要件は次のとおりです。

  - º Samba　サーバがサービスするすべてのサブネットを記述する必要があります。これは LifeKeeper が保護するアドレスでなくてはならず、クラスタ内で重複してはなりません(その他の Samba 設定ファイルで同じ IP アドレスを使用してはなりません)。

  - º interfaces 命令では複数のフォーマット、IP アドレス(ドット形式またはホスト 名)、ネットワーク・インタフェース名を指定でき、ワイルドカードを使用でき ます。ただし、Samba Recovery Kit の場合、ワイルドカードは使用せず IP アドレスはドット形式 (100.25.104.25) で指定する必要があります。

  - º IP アドレスとあわせてサブネット・マスクを指定することもできますが、LifeKeeper では使用されません。

  - º この命令で指定したアドレス用の LifeKeeper IP リソースは、Samba リソース階層の作成前に作成しておく必要があります。また、この命令のアドレスにネットワ ーク・マスクを適用する場合は、作成時の IP リソースで使用されるマスクと一致している必要があります。

  - º その他の保護されていない Samba インスタンスも、interfaces 命令を使用して、LifeKeeper が使用するものとは異なる IP アドレスを指定します。

    **注記：** 後述するbind interfaces only 命令を使用するために、interfaces命令で localhost アドレス 127.0.0.1 を指定して、ユーティリティ **smbpasswd** が正しく動作するようにしなくてはならない場合があります。localhost アドレスが必要かどうか判断する方法については、「Samba の複数インスタンスの実行 」を参照してください。

- **lock directory** (または lock dir) - 全サーバの Samba インスタンスで使用する一意のロック・ファイル・ディレクトリの名前と場所。これは、クラスタ内の全サーバに存在する既存のディレクトリでなくてはなりません。

  **注記：** この命令は **lock dir** と呼ばれることもあります。 Samba Recovery Kit はどちらの名前でも処理します。

- **bind　interfaces only** - この命令は、 **smbd** プロセスと **nmbd** プロセスが interfaces 命令で定義

されたアドレスに対する SMB リクエストだけを処理するようにします。これは Yes に設定する必要があります。 システムで動作している保護されていない Samba インスタンスでも、この命令を yes に設定する必要があります。 yes に設定すると、Samba は interfaces 命令で指定されていないサブネットに対するリクエストは処理せず、サーバ上で動作している別の Samba インスタンスに対するリクエストも処理しません。

# 6.15.4.3. 設定ファイルの [homes] セクション

[homes] セクションは、Samba サーバ上のユーザのホーム・ディレクトリが明示的に共用として定義されていない場合に、ホーム・ディレクトリに対する接続を処理する特別なセクションです。LifeKeeper では、この特別な共用を使用してユーザのホーム・ディレクトリを 保護することはありません 。したがって、 この共用は削除するかコメント・アウトしてください 。 LifeKeeper Samba Recovery Kit で Samba の共用を保護するには、path 命令を指定する必要があります。path 命令は LifeKeeper Samba 階層が保護するファイル・システムを決定するために使用されます。ユーザが Samba サーバに接続したときにパスが決定されるため、[homes] セクションのデフォルトではパスは指定されていません。この特別な共用を削除するかコメント・アウトする必要があるのは、このためです。

# 6.15.4.4. 設定ファイルの [printers] セクション

[printers] セクションは、Samba サーバ上のプリンタが具体的に共用として定義されていない場合に、プリンタへの接続を処理します。LifeKeeper はこの特別なセクションやグローバル命令 load printers を使用してプリンタの共用を保護することはありません。LifeKeeper が保護するプリンタの共用はそれぞれ設定ファイルの独自の共用セクションで定義する必要があります。

# 6.15.4.5. 設定ファイルの共用定義セクション

設定ファイルの残りの全セクションでは、このインスタンスに対してクライアントがアクセスできるファイルとプリンタの共用を定義します。設定ファイルでは共用を 1 つ以上定義する必要があります。Samba 設定ファイルでは、ファイルの共用のみ、プリンタの共用のみ、ファイルとプリンタの共用の両方の組み合わせのいずれも定義可能です。LifeKeeper の場合、定義可能な共用数に制限はありませんが、1 つの共用に関して障害が発生しただけで、階層全体がバックアップ・サーバに切り替えられることに注意する必要があります。各共用に対して次の命令を定義する必要があります。

- **path** - この命令はファイルまたはプリンタの共用のルート・パス名を指定します。この値によって、Samba 階層の一部として保護するファイル・システム・リソースが決まり ます。Samba リソース作成時に LifeKeeper ファイル・システム・リソースがすでに存在していない場合は、 LifeKeeper によって自動的に作成されます。

  注記： この命令は directory と呼ばれることもあります。Samba Recovery Kit はどちらの名前でも命令を処理します。

- **printable** - 値が Yes の場合、Samba の共用は Linux プリンタに印刷するためのプリント・スプール・レポジトリとして使用されます。共用を通常のファイルの共用として使用する場合は、この命令を No に設定するか、命令自体を指定しません。[global] セクションで Yes に設定されていない限り、この値のデフォルトは No です。この命令を Yes に設定した場合、下記の printer name 命令で定義したプリンタを保護する LifeKeeper プリント・サービス・リソースが Samba 階層作成時に存在している必要があります。

  注記： この命令は **print ok** と呼ばれることもあります。Samba Recovery Kit はどちらの名前でも命令を処理します。

- **printer name** - この命令は共用が使用するプリンタ名を定義し、名前付きプリンタを保護するプリント・サービス・インスタンスを探すときに使用されます。プリント・サービス・インスタンスは Samba 階層の子リソースになります。プリンタの共用に対してこ の命令が定義されていない場合、Samba Recovery Kit は共用名をプリンタ名と見なします。

  注記： この命令は **printer** と呼ばれることもあります。Samba Recovery Kit はどちらの名前 でも命令を処理します。

注記： Samba 設定ファイルでは、いくつかの命令で変数置換を使用できます。Samba ユーティリティ testparm で変数を解決できない限り、前述の命令では、変数置換を使用しないでください。

# 6.15.4.6. Samba の複数インスタンスの実行

LifeKeeper クラスタで Samba の複数インスタンスを実行する場合は、追加の設定要件と制限があります。次の Samba 設定の場合、Samba の複数インスタンスが関係してきます。

- アクティブ／スタンバイ設定で複数の LifeKeeper Samba インスタンスが 1 つのサーバ上にある場合
- アクティブ／アクティブ設定で複数の LifeKeeper Samba インスタンスが 1 つ以上のサーバ上にある場合

この設定のどちらの場合でも、LifeKeeper で保護されていない Samba インスタンスを含めることができます。

「LifeKeeperでのSambaの設定」で述べたように、Sambaの複数インスタンスを実行するときには、各インスタンスの設定ファイル名が異なっているか、異なるディレクトリに存在している必要があります。各設定ファイルにいくつか命令を記述する必要があり、その値は異なっていなくてはいけません。具体的には、netbios name、lock directory、pid directory、interfaces、およびlog fileです。これらの命令の値が一意でない場合、Sambaが起動せず、したがってクライアントが接続できなくなります。また、各インスタンスで指定したlock、log file、pidの各directoryはクラスタ内の全サーバに存在している必要があります。

## smbpasswd ユーティリティと Samba の複数インスタンス

LifeKeeper では必須ではありませんが、Samba Recovery Kit で使用する一部の Samba ユーティリティには、smb.conf がデフォルトの場所にあることを前提としているものがあります。Samba Recovery Kit は **smbclient** ユーティリティと **nmblookup** ユーティリティを使用してそれぞれ **smbd** と **nmbd** に接続し、LifeKeeper が保護しているデーモン・プロセスの状態を判断します。この 2 つのユーティリティは、デフォルトの場所に smb.conf が存在しなくてもエラー終了することはありません。しかし、**smbpasswd** ユーティリティの場合は、smb.conf がデフォルトの場所に存在する必要があります。

**smbpasswd** は、セキュリティ・レベルが share または user に設定されているときに、クライアント接続リクエストに対するユーザ認証を行う **smbpasswd** ファイルを管理するのに使用します。デフォルト設定ファイルが存在しない場合、Samba のパスワードの変更はすべて失敗します。この問題を回避するには、セキュリティ・レベルが share または user に設定されている場合、または、Samba セキュリティ・レベルが server に設定されているシステムに対する **smbpasswd** サーバとしてサーバが動作している場合には、Samba インスタンスの 1 つはデフォルト設定ファイルを使用する必要があります。これは、**smbpasswd** がデフォルト設定ファイルを使用して **smbpasswd** ファイルの場所を取得するためです。このため、LifeKeeper クラスタ内で **smbpasswd** ファイルが存在する場所は 1 か所だけに限られます。クラスタ内の Samba の全インスタンスの設定ファイルで、smb passwd file 命令を同じ値に設定する必要があります。また、**smbpasswd** ファイルはクラスタ内の全サーバで同期を保つ必要があります。

**smbpasswd** ユーティリティは、LifeKeeper Samba Recovery Kitでは必須のbind interfaces only命令の影響も受けます。bind interfaces only命令をYesに設定すると、Sambaパスワードを変更する通常ユーザはlocalhostアドレス 127.0.0.1 を使用して **smbd** デーモン・プロセスに接続しようとします。**smbd** デーモンが使用する設定ファイルのinterfaces命令にそのアドレスが追加されていて、**smbd** がそのアドレスに接続してリッスン中の場合、パスワードの変更は成功します。デーモンの設定ファイルのinterfaces命令

にそのアドレスが存在しない場合、パスワードの変更は失敗します。複数インスタンス環境でlocalhostが複数の設定ファイルで指定されている場合、起動して実行できるのは 1 つのインスタンスだ けです。**smbpasswd** で「 -r netbios_name」オプションを使用すると、インタフェース・リストにlocalhostアドレスを追加しなくても済みます(たとえば、 **smbpasswd -r server1 print1** …)。

注記：前述のように、保護されたSambaインスタンスが存在するLifeKeeperサーバ上で実行される保護されていないSambaインスタンスも、bind interfaces only命令をYesに設定する必要があります。

## Samba およびユーザ認証の考慮事項

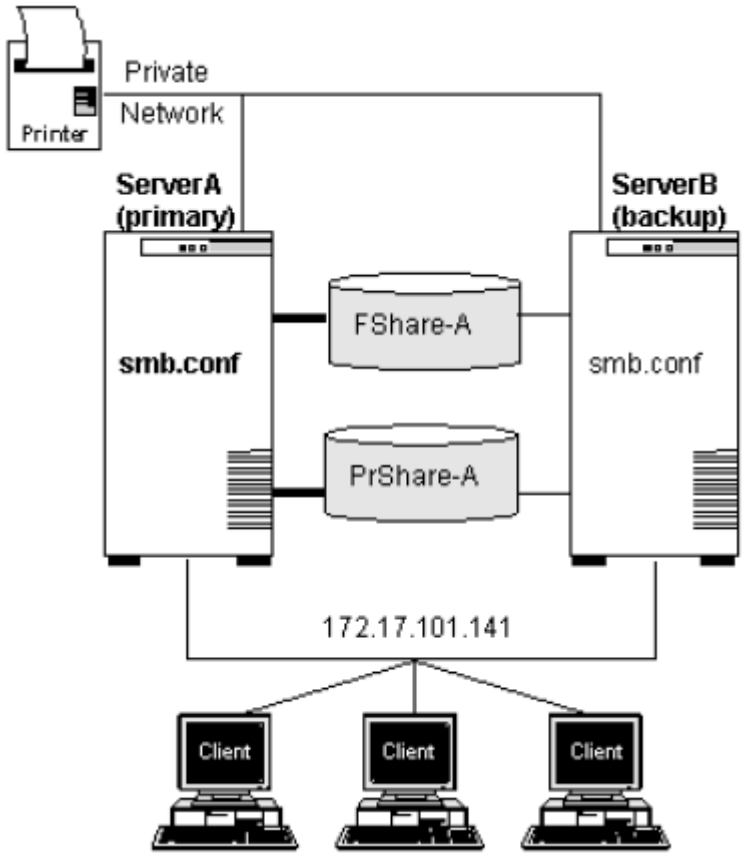
Samba は、ユーザ認証のために、クラスタ内の全 ての サーバ上で /etc/samba/smbpasswd または /etc/samba/secrets.tdb のようなデータファイルが一致しているのを保障するために、LifeKeeperwp 通じて Samba を保護する場合に考慮しなければならないセキュリティ・パラメータを通していくつかのメソッドをサポートします。(例えば、共有, ユーザ, ドメインなど…)。ユーザとしてセキュリティ・メソッドを使用する場合は、クラスタ内の全てのサーバで smbpasswd ファイルが一致していることを確認する必要があります。加えて、ドメインのようなセキュリティ・メソッドは secrets.tdb ファイルの一致が要求されます。 secrets.tdb ファイルの使用かつ LifeKeeper のアクティブ/アクティブ構成は、ファイルの配置場所を指定するために、プライベート dir パラメータの使用が必要となります。このパラメータの値はそれぞれの LifeKeeper Samba インスタンスにおいてユニークでなければなりません。

# 6.15.4.7. 構成例

このセクションでは一般的な Samba 設定の定義と例を示します。各例には、LifeKeeper に当てはまる設定ファイルの項目が含まれています。

## 構成 1：アクティブ／スタンバイ設定の例

アクティブ／スタンバイ設定では、ServerA がプライマリ LifeKeeper サーバです。プライ マリ・サーバは共有ストレージ・デバイスにあるファイルとプリンタの共用をエクスポートします。ServerB は別のアプリケーションやサービスを処理していてもかまいませんが、LifeKeeper に対しては Samba リソースのバックアップとしてだけ機能します。




設定に関する注記：

- クライアントは NetBIOS 名 LKServerA を使用して、保護された IP アドレス（172.17.101.141）経由で Samba サーバに接続します。 IP アドレスは設定ファイルの interfaces 命令で定義されています。

- 設定ファイル smb.conf は Samba リソース階層の拡張時に ServerB にコピーされます。設定ファイルには次の命令が記述されています。

  [global] netbios name = LKServerA

  bind interfaces only = yes

lock directory = /var/lock/samba interfaces = 172.17.101.141 127.0.0.1

log file = /var/log/sambaServA/log

[FShare-A]

path = /FShare-A

read only = no

public = yes

valid users =

printable = no create mode = 0664

directory mode = 0775

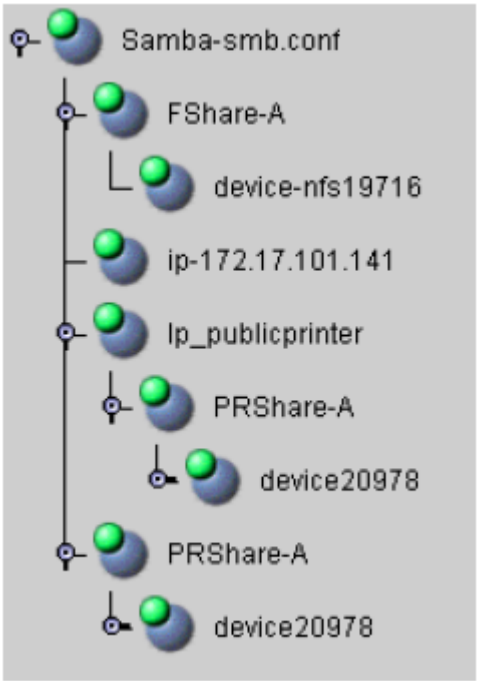
[PRShare-A]

path = /PRShare-A printer = publicprinter

printable = yes

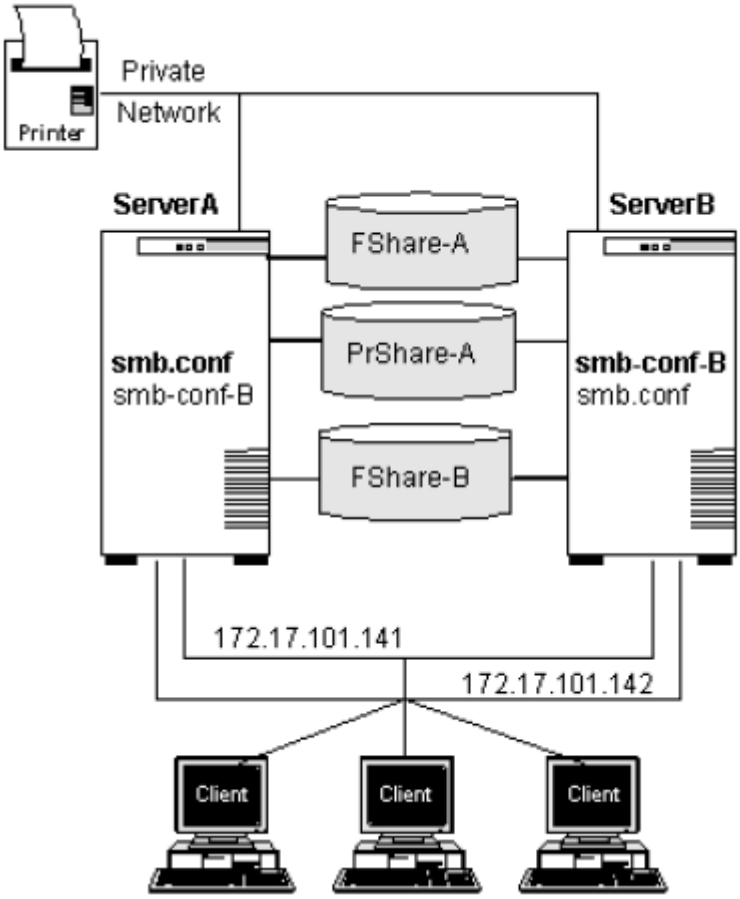
browseable = no

- Samba リソース階層は次のようになります。



# 構成 2：アクティブ／アクティブ設定の例

下図のアクティブ／アクティブ設定の場合、ServerA と ServerB の両方が Samba リソース のプライマリ LifeKeeper サーバです。各サーバは互いのバックアップ･サーバにもなります。



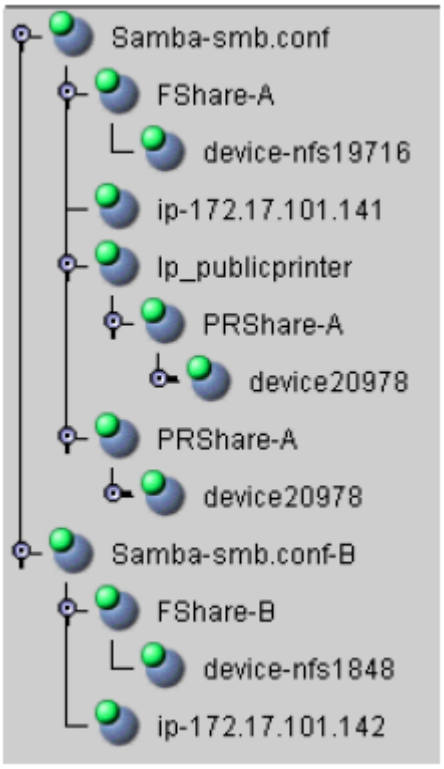

設定に関する注記：

- クライアントは NetBIOS 名 LKServerA と LKServerB を使用して、保護された IP アドレス (172.17.101.141 および 172.17.101.142) 経由で Samba サーバに接続します。IP アドレスは設定ファイルの interfaces 命令で定義されています。
- 設定ファイル smb.conf は Samba リソース階層の拡張時に ServerB にコピーされます。 同様に、設定ファイル smb.conf-B は Samba リソース階層の拡張時に ServerA にコピーされます。
- ServerA はファイルの共用/Fshare-A を保護します。ServerB はファイルの共用/Fshare- B を保護します。
- ServerA はプリンタの共用/publicprinter を保護します。ServerB はプリンタの共用を保護しません。
- 2 つの設定ファイルには次の命令が記述されます。

| smb.conf | smb-conf-B |
|---|---|

| | |
|---|---|
| [global]<br>netbios name = LKServerA bind<br>interfaces only = yes<br>lock directory = /var/lock/sambaServA<br>pid directory = /var/run/sambaServA<br>interfaces = 172.17.101.141 127.0.0.1<br>log file = /var/log/sambaServA/log<br>[FShare-A]<br>path = /FShare-A<br>read only = no<br>public = yes<br>valid users =<br>printable = no create mode = 0664 directory mode = 0775<br>[PRShare-A]<br>path = /PRShare-A<br>printer = publicprinter<br>printable = yes browseable = no | [global]<br>netbios name = LKServerB bind<br>interfaces only = yes<br>lock directory = /var/lock/<br>pid directory = /var/run/sambaServB<br>sambaServB interfaces = 172.17.101.142<br>log file = /var/log/sambaServB/log<br>[FShare-B]<br>path = /FShare-B<br>read only = no<br>public = yes<br>valid users =<br>printable = no create mode = 0664 directory mode = 0775 |

- Samba リソース階層は次のようになります。


Samba-smb.conf
FShare-A
device-nfs19716
ip-172.17.101.141
lp_publicprinter
PRShare-A
device20978
PRShare-A
device20978
Samba-smb.conf-B
FShare-B
device-nfs1848
ip-172.17.101.142

# 6.15.5. Samba の設定手順

このセクションでは Samba リソースの設定に必要な手順について説明します。

1. Samba の設定案を作成します。次のような項目を決定します。

   - 使用する NetBIOS 名
   - 保護して共用へのアクセスを許可するインタフェース
   - Samba の共用で使用し、したがって保護するファイル・システム
   - アクセスし保護するプリンタ(存在する場合)

   ロック・ディレクトリとログ・ディレクトリの場所 使用する設定ファイルの数と設定の種類 (アクティブ／スタンバイかアクティブ／アクティブか)について検討します。たとえば、保護する Samba の共用が 4 つある場合、1 つの設定ファイルに 4 つの共用をすべて記述することもできますが、1 つのファイル・システムに障害が発生するだけで 4 つのファイルの共用すべてを含めて Samba 階層全体のフェイルオーバが発生するというデメリットがあります。別の方法として、それぞれ 1 つのファイルの共用を保護する独立した設定ファイルを 4 つ作成することもできますが、この場合、4 つの NetBIOS 名を定義して管理する必要があります。

2. Samba 設定ファイルをステップ 1 で作成した設定案に従ってセットアップします。これには、[global] セクションに必要な命令や、使用するファイルとプリンタの共有などがあります。LifeKeeper Samba 階層に必要なグローバル命令と共用命令については、「LifeKeeper での Samba の設定 」を参照してください。

3. LifeKeeper で保護する IP アドレスを作成します。 この IP アドレスは、NetBIOS 名を使用してクライアントが Samba サーバーに接続するのに使用されます。保護された IP アドレスは、設定ファイルの interface 命令に一致する必要があります。IP リソースのセットアップの詳細については、『LifeKeeper for Linux IP Recovery Kit 管理ガイド 』を参照してください。すべてのクライアントおよびその他のクラスタ・サーバー から ping を実行して、保護する IP アドレスをテストします。ローカル・ホスト(127.0.0.1) 用の保護される IP リソースは不要です。

4. Samba デーモンを起動して、クライアント接続をテストします。

   a. デーモンを起動するコマンドは次のとおりです。
      ```
      s/nmbd -D -s ConfigurationFile
      ```

   b. Samba ユーティリティ smbclient を使用して、次のように smbd デーモンへの接続をテストします。これは interfaces 命令で定義された各アドレスに対して実行する必要があります。
      ```
      smbclient -L netbios_name -U% -I Protected_IP_Address
      ```

   c. **nmblookup** を使用して **nmbd** デーモン・プロセスへの接続をテストします。これは各ブロードキャスト・アドレスに対して実行する必要があります。interfaces 命令に定義された各アドレスごとに、関連するブロードキャスト・アドレスを使用します。(ブロードキャスト・アドレスは、 **ip addr show** を実行すると取得できます。)
      ```
      nmblookup -B broadcast_address netbios_name
      ```

5. 1 つ前の手順で起動した Samba デーモンを停止します。これは **kill** コマンドを使用して行います。ps コマンドを使用して実行中のデーモン・プロセスを見つけ、 **kill pid** を発行してデーモン・プロセスを終了します。

6. LifeKeeper 下で保護されるファイル・システム(Samba ファイルを管理する) を作成し、上記のステップで定義した共有を印刷します。(SPS for Linux テクニカルドキュメンテーションの「ファイルシステムリソース階層の作成 」を参照してください。) ファイル・システム・リソースは、プリント・サービス・リソースまたは Samba リソース作成時に自動的に作成されるため、この手順は省略することもできます。

7. 1 つのファイル・システムを複数の Samba の共用で使用する場合は、保護するファイル・システム上に共用のためのディレクトリを作成します。

8. LifeKeeper 下で保護される印刷サービス階層 (設定ファイルにプリンタ共有が定義さ れている場合に、クライアントの印刷に使用される) を作成します。

9. LifeKeeper で Samba リソース階層を作成し、少なくとも 1 つのバックアップ・サーバーに拡張します(後述の「LifeKeeper設定作業 」を参照してください)。extend スクリプトによってSamba設定ファイルがテンプレート・サーバーからターゲット・サーバーの同じ場所にコピーされます。

10. プライマリ・サーバー側で、サービス中のSamba階層が保護する共用へのクライアント接続をテストします。たとえば、Windows クライアントから共用ディレクトリにマッ ピングして、共用ファイルにアクセスできるか確認します。クラスタ内の全サーバーに対してテストを繰り返します。Samba リソースは、手動でバックアップ・サーバーへスイッチオーバする方法でテストすることもできます。(「リソース階層のテスト 」を参照してください。)

11. LifeKeeper で保護する Samba デーモンは LifeKeeper が制御するので、システム起動時の自動起動を無効化する必要があります。

# 6.15.6. LifeKeeper の設定作業

LifeKeeper for Linux Samba Recovery Kit の管理作業はすべて、LifeKeeper グラフィカル・ユーザ・インタフェース(GUI) で実行できます。LifeKeeper GUI では、Samba リソースの設定、管理、監視作業をガイド付きで行えます。

LifeKeeper for Linux Samba Recovery Kit の設定に関しては、次の作業が可能です。

- リソース階層の作成 。Samba リソース階層を作成します。
- リソース階層の削除 。Samba リソース階層を削除します。
- リソース階層の拡張 。Samba リソース階層をプライマリ・サーバからバックアップ・サーバに拡張します。
- リソース階層の拡張解除 。LifeKeeper クラスタの 1 つのサーバの Samba リソース階層を拡張解除 (削除) します。
- 従属関係の作成 。既存のリソース階層と別のリソース・インスタンスとの間に子従属関係を作成し、クラスタ内のすべての対象サーバに従属関係の変化を通知します。
- 従属関係の削除 。リソースの従属関係を削除して、クラスタ内のすべての対象サーバ に従属関係の変化を通知します。
- In Service 。リソース階層をアクティブにします。
- Out of Service 。リソース階層を非アクティブにします。
- 表示 / 編集 。 リソース階層のプロパティを表示または編集します。

**注記：** このセクションの残りの部分では、設定作業はすべて **[編集]** メニューから実行することになっています。この設定作業の大半は次の方法でも実行できます。

1. ツールバーから状態表示の左ペインにあるグローバル・リソースを右クリック。
2. 状態表示の右ペインにあるリソース・インスタンスを右クリック。

※右クリックの方法を使用すると、**[編集]** メニューを使用したときに必要な情報を入力しないですみます。

# 6.15.6.1. リソース階層の作成

必要なセットアップ作業を完了したあと、Samba リソース階層を作成するために、以下の手順を行います。

1. **[LifeKeeper GUI]** メニューから **[編集]** を選択し、次に **[サーバ]** を選択します。ドロップダウンメニューから、**[リソース階層の作成]** を選択します。

   **[リソース作成ウィザード]** ダイアログ･ボックスが表示され、クラスタにインストールされている全ての Recovery Kit がリストボックスに表示されます。

2. Samba　の共用を選択して、**[次へ]**　をクリックします。

3. 次の表で説明している情報を入力します。どのダイアログ･ボックスでも、**[戻る]** がアクティブになっているときには、直前のダイアログ･ボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも **[キャンセル]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| スイッチバック･タイプ | **[インテリジェント]** または **[自動]** を選択します。これによって、バックアップサーバへのフェイルオーバ後、Samba インスタンスがこのサーバでサービスに戻ったときにどのようにスイッチバックされるかが決まります。スイッチバック･タイプは、必要な場合、**[リソース･プロパティ]** ダイアログ･ボックスの **[一般]** タブで後から変更できます。<br><br>**注記：** スイッチバックの方法は、Samba リソースで使用するプリントサーバ、IP リソース、ファイル･システムリソースのスイッチバック方法と一致している必要があります。一致していない場合、Samba リソース作成時に、Samba リソースに対して選択された設定に一致するように設定がリセットされます。 |
| サーバ | 階層を作成するサーバを選択します。 |
| 設定ファイルのロケーション | Samba 設定ファイルを配置するディレクトリを選択します。 |
| 設定ファイル名 | このリソースの作成に使用する Samba 設定ファイルの名前を入力します。デフォルトは smb.conf です。<br><br>**注記：** LifeKeeper は選択された設定ファイルを読み込み、必要な命令が指定されていない場合、エラー･メッセージを表示します。設定ファイルの最小限のチェックも行います(共用が存在 |

| | |
|---|---|
| | するか、path 命令があるか、ロック・ディレクトリが指定されているか、そのディレクトリが存在するか検証)。その他のチェックも作成処理の途中で実行されます。 |
| ルート・タグ | LifeKeeper が示すデフォルト・ルート・タグを選択するか、このサーバ上のリソース・インスタンスの重複しない名前を入力します。デフォルトは Samba-configfilename です。 ここで、configfilename は対応する設定ファイルの名前です。英字、数字、および特殊文字 (- _ . /) を使用できます。 |

4. **[作成]** をクリックします。**[リソース作成ウィザード]** が表示され、Samba リソース階層が作成されます。LifeKeeper によって入力データが検証されます。問題が検出された場合、情報ボックスにエラーメッセージが表示されます。

5. 情報ボックスが表示され、Samba リソース階層の作成に成功したこと、フェイルオーバによって保護するには、クラスタ内の別のサーバにリソース階層を拡張する必要があることを示すメッセージが表示されます。**[次へ]** をクリックします。

6. **[続行]** をクリックします。**[事前拡張ウィザード]** が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、次の「Samba リソース階層の拡張 」の手順 2 を参照してください。

**注記：** Samba 階層の作成に失敗する場合は 、「トラブルシューティング」の「Samba 階層の復元の失敗 」を参照してください。

# 6.15.6.2. リソース階層の拡張

この操作は **[編集]** メニューから開始できます。また、**[リソース階層の作成]** の処理を完了すると自動的に開始されます。その場合は、手順2を参照してください。

1. **[編集]** メニューの **[リソース]** から **[リソース階層の拡張]** を選択します。**[事前拡張ウィザード]** が表示されます。拡張操作に慣れていない場合は、**[次へ]** をクリックします。LifeKeeper の[リソース階層の拡張]作業のデフォルト値が分かっていて、入力と確認 を省略する場合は、**[デフォルトを受け入れる]** をクリックします。

2. **[事前拡張ウィザード]** で次の情報を入力します。

   **注記：** 最初の 2 つのフィールドは、**[編集]** メニューの **[拡張]** から操作を開始した場合にだけ表示されます。

| フィールド | ヒント |
|---|---|
| テンプレート・サーバ | Samba リソースが現在サービス中のサーバを入力します。 |
| 拡張するタグ | 拡張するSamba リソースを選択します。 |
| ターゲット・サーバ | 拡張先のサーバを入力または選択します。 |
| スイッチバック・タイプ | **[インテリジェント]** または **[自動]** を選択します。スイッチバック・タイプは、必要な場合、**[リソース・プロパティ]** ダイアログ・ボックスの **[一般]** タブで後から変更できます。<br><br>**注記：** スイッチバック方法は、Samba リソースが使用する従属リソースのスイッチバック方法と一致する必要があることを忘れないでください。 |
| テンプレートの優先順位 | テンプレート階層の優先順位を選択または入力します。優先順位は1～999 の範囲の未使用の値が有効で、数値が小さいほど優先順位は高くなります(1は最高の優先順位を表わします)。 拡張操作時には、別のシステムで既に使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。 |

| | |
|---|---|
| | **注記** ：これは階層を最初に拡張するときだけ表示されます。 |
| ターゲットの優先順位 | ターゲット・サーバの階層の優先順位を選択、または入力します。 |
| ルート・タグ | ターゲット・サーバの新しい Samba リソース・インスタンスのデフォルト・タグ名が表示されます。デフォルト・タグ名はテンプレート・サーバ上のこのリソースのタグ名と同じです。新しいタグ名を入力する場合は、ターゲット・サーバ上で重複しない名前にします。英字、数字、および特殊文字（- _ . /）を使用できます。 |

3. 拡張前のチェックが成功したというメッセージが表示されたら、**[次へ]** をクリックします。

4. 拡張する階層に応じて、拡張するリソース・タグを表示する情報ボックスが表示されます。リソース・タグは編集できません。**[拡張]** をクリックします。

5. 「Hierarchy extend operations completed (階層の拡張操作が完了しました)」というメッセージが表示されたら、階層を別のサーバに拡張する場合は、**[次のサーバ]** をクリックし、拡張作業が必要ない場合は、**[完了]** をクリックします。

6. 「Hierarchy Verification Finished (階層の検査が完了しました)」というメッセージが表示されたら、**[完了]** をクリックします。

# 6.15.6.3. リソース階層の拡張解除

LifeKeeper クラスタの1 つのサーバのリソース階層を削除するには、次の手順を実行します。

1. **[編集]** メニューの **[リソース]** から **[リソース階層の拡張解除]** を選択します。

2. Samba リソースを拡張解除する **[ターゲット・サーバ]** を選択します。リソースが現在サービス中の Samba サーバは選択できません。(このダイアログ・ボックスは、右側のペ インでリソース・インスタンスを右クリックして、 **[拡張解除]** を選択した場合には、表示されません。) **[次へ]** をクリックします。

3. 拡張解除するSamba 階層を選択し、 **[次へ]** をクリックします。(このダイアログ・ボックスは、いずれかのペインでリソース・インスタンスを右クリックして、 **[拡張解除]** を選択した場合には、表示されません。)

4. 拡張解除するために選択したターゲット・サーバとSamba リソース階層を確認する情報ボックスが表示されます。 **[拡張解除]** をクリックします。

5. Samba リソースの拡張解除に成功したことを示す別の情報ボックスが表示されます。 **[完了]** をクリックして、 **[リソース階層の拡張解除]** メニューを終了します。

# 6.15.6.4. リソース階層の削除

Samba 階層を削除すると、従属関係および保護されているサービスに何が起きるかを理解しておくことが重要です。

- **従属関係:** リソース階層を削除する前に、従属関係を削除したほうがよい場合があ ります。別の階層で使用されていない場合、従属しているファイル・システムは削 除されます。従属している IP リソースとプリント・サービス・リソースは、 LifeKeeper GUI または Samba 削除スクリプトを使用して削除する限り、削除されません。

- **保護されたサービス:** Sambaリソースを削除する前に Samba リソースを停止した場合、リソースを削除した後 Samba デーモンは停止します。Samba リソースの動作中にリソースを削除した場合、Samba デーモンは動き続けます。リソース階層の削除をした後は LifeKeeper の保護対象となりません。

LifeKeeper 環境内のすべてのサーバからリソース階層を削除するには、以下の手順を実行します。

1. **[編集]** メニューの **[リソース]** から **[リソース階層の削除]** を選択します。
2. Samba リソース階層を削除する **[ター ゲット・サーバ]** を選択し、 **[次へ]** をクリックします。 （このダイアログ・ボックスは、いずれかのペインでリソース・インスタンスを右クリックして、 **[リソースの削除]** を選択した場合には、表示されません。）
3. **[削除する階層]** を選択します。このダイアログ・ボックスは、左側または右側のペインでリソース・インスタンスを右クリックして、 **[リソースの削除]** を選択した場合には、表示されません。） **[次へ]** をクリックします。
4. 選択したターゲット・サーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[削除]** をクリックします。
5. Samba リソースの削除に成功したことを示す別の情報ボックスが表示されます。
6. **[完了]** クリックして終了します。

# 6.15.6.5. リソース階層のテスト

手動で切り替えを行って、Samba リソース階層のテストを行うことができます。これは、プライマリ・サーバからバックアップ・サーバへのリソース・インスタンスのフェイルオーバをシミュレートします。

**[編集]** メニューの **[リソース]** から **[サービス中]** を選択します。たとえば、バックアップ・サーバ側で **[サービス中]** を実行すると、アプリケーション階層がプライマリ・サーバ側ではサービス休止になり、バックアップ・サーバ側でサービス中になります。この時点で、元のバックアップ・サーバがプライマリ・サーバになり、元のプライマリ・サーバがバックアップ・サーバになります。

**[サービス休止]** 要求を実行すると、アプリケーションは他のサーバでサービス中にならずに、サービスが停止されます。

# 6.15.7. Samba 階層の管理

Samba リソース階層の作成後、以後の Samba リソースの管理では以下の指針に従ってく ださい。

Samba 設定ファイルの修正

smbpasswd ファイルの管理

# 6.15.7.1. Samba 設定ファイルの修正

LifeKeeper Samba インスタンスで使用する Samba 設定ファイルを変更する必要があるときには、保護サービス中 (ISP) のサーバ上で、ここで説明する手順を実行します。設定ファイルの変更には次の 3 種類があります。

- Samba 階層に直接影響を与えない変更
- 階層に直接影響を与え、階層の削除と再作成は必要ない変更
- 階層に直接影響を与え、階層の削除と再作成が必要な変更必要な変更の種類を特定する方法と種類毎の手順については、以下のセクションを参照してください。

## Samba 階層に直接影響を与えない変更

Samba 階層に直接影響を与えない変更LifeKeeper によって使用されない設定ファイルの命令に対する変更は、このカテゴリに属します。(キットで使用する命令のリストについては、「LifeKeeper での Samba の設定 」を参照してください。) LifeKeeper が使用しない命令の例としては、security、hosts allow、hosts deny、valid users などがあります。手順は次のとおりです。

1. 設定ファイルを使用する Samba リソースをサービス休止にします。この手順は Samba デーモンを停止するために必要です。
2. Samba 設定ファイルに必要な変更を加えます。
3. クラスタ内の設定を同期します。この作業は、次のようにユーティリティ **synccfg** を使用して実行します。

   **LKROOT/lkadm/subsys/gen/samba/bin/synccfg -t TargetSys -c ConfigFile**

   ここで LKROOT は LifeKeeper のインストール場所(デフォルトは/opt/LifeKeeper)、TargetSys は更新するノード、ConfigFile はコピーする設定ファイルのフルパスです。
4. クラスタ内の全サーバに対して 1 つ前の手順を繰り返します。
5. 階層をサービス中に戻して、Samba デーモンを再起動します。

## Samba 階層に直接影響を与える変更

LifeKeeperが使用する設定ファイル命令（一覧は「LifeKeeper での Samba の設定 」を参照）の変更がこの種類に該当します。 ただし、netbios nameまたは設定ファイルの物理的移動を除きます。変更の大きさによっては、3 種類目と同様に階層を再作成するほうが簡単で作業が早く済むこともあります。この種類に該当する典型的な変更としては、新 しいファイルとプリンタの共用の追加、ファイルとプリンタの共用の削除、IPインタフ ェースの追加または削除などがあります。

1. 設定ファイルを使用する Samba リソースをサービス休止にします。この手順は Samba デーモンを停止するために必要です。

2. Samba 設定ファイルに必要な変更を加えます。

3. Samba 階層に必要な変更を加えます。これは設定ファイルに加える変更の種類によって異なります。たとえば、以下のようになります。

   - º interfaces 命令に IP アドレスを追加した場合は、新しい IP リソースを作成して拡 張し、従属した子として Samba リソース階層に追加する必要があります。SPS for Linux Technical Documentation の

リソース依存関係の作成 を参照してください。

   - º 設定ファイルにファイルの共用を追加した場合は、ファイル･システム･リソー スを作成して拡張し、従属した子として Samba リソース階層に追加しなくては ならない場合があります。ファイル･システム･リソースがすでに子として階層 に存在している場合は(たとえば、新しい共用のために定義された path 命令が 別のファイルの共用またはプリンタの共用と同じファイル･システム･マウン ト･ポイントである場合)、従属した子として作成し追加する必要はありません。

   - º プリンタの共用を追加する場合、ファイル･システム･リソースとプリント･サ ービス･リソースを作成して拡張し、従属した子として Samba 階層に追加する 必要があります。print share name 命令または printer/printer name 命令で定義され たプリンタを保護するプリント･サービス･リソースが存在しない場合は、それ を作成する必要があります。ファイル･システム･リソースを追加する必要があ るかどうかは、前述のファイルの共用の場合を参照してください。

   - º ファイルの共用またはプリンタの共用を削除した場合、または、interfaces 命令か ら IP アドレスを削除した場合は、Samba 階層の従属関係を削除してから、個々 のリソースを削除します。

   - º プリンタの共用名を変更する場合は、プリンタの共用を削除してから新しいプリ ンタの共用を追加します。

4. クラスタ内の設定を同期します。この作業は、次のようにユーティリティ **synccfg** を使用して実行します。

   **LKROOT/lkadm/subsys/gen/samba/bin/synccfg -t TargetSys -c ConfigFile**

   ここで LKROOT は LifeKeeper のインストール場所 (デフォルトは/opt/LifeKeeper)、TargetSys は更新するサーバ、ConfigFile はコピーする設定ファイルのフルパスです。

5. クラスタ内の全サーバに対して 1 つ前の手順を繰り返します。

6. 階層をサービス中に戻して、Samba デーモンを再起動します。

   **注記：** 多数のリソースの作成と従属関係の追加または削除が必要な変更を行う場合は、Samba 階層をサービス休止にする前に新しいリソースをすべて作成しておき、システムの停止時間を最小限に抑えたほうがよい場合があります。

## 階層に直接影響を与え、階層の削除と再作成が必要な変更

netbios name 命令を変更するか設定ファイルの物理的場所を変更する場合は、次の手順を 実行します。

1. 階層を削除します。（詳細は「Samba 階層の削除」を参照してください。）
2. NetBIOS 名を変更します。 または設定ファイルを移動します。
3. 新しい Samba 階層を作成して、すべてのバックアップ・サーバに拡張します。

# 6.15.7.2. smbpasswd ファイルの管理

Sambaでは、security命令で 4 つの異なる認証方法を指定できます。セキュリティ・レベ ルが share の場合もuserの場合も、ローカルの smbpasswd ファイルを参照してアクセスが許可されているか判断する必要があります。「Samba の複数インスタンスの実行 」で述べているように、smbpasswd ファイルは 1 つしか存在できず、そのためLifeKeeperクラスタでは管理上の問題が発生する可能性があります。セキュリティ・レベルとしてshare または userを選択した場合は、全サーバでファイルを同期して、フェイルオーバ後も認証が成功するようにしておく必要があります。

Samba 階層が 1 つしかないクラスタでは、セキュリティ・レベルを share または user にした場合、smbpasswd ファイルを設定ファイルで定義したファイルの共用に配置することができます。この共用に対しては、管理者だけがアクセスできるようにしておきます。複数インスタンスが存在する場合は、セキュリティ・レベルとして server または domain を使用することを推奨します。

注記： ファイアウォールを使用している場合は、ファイアウォールが **smbd** デーモンへの 接続を許可し、 **nmblookup** が動作することを確認してください。

# 6.15.8. トラブルシューティング

## Samba 階層の復元の失敗

Samba 階層の復元が失敗しても、デーモン・プロセス **smbd** および **nmbd** は実行されたままである場合があります。(復元操作は、作成、フェイルオーバ、または手動のスイッチオーバの完了によって開始されるか、または **smbd** および **nmbd** への接続障害が原因のローカル・リカバリーによって開始されます。) 問題を修正した後も、デーモンの停止と再起動が実行されない場合、復元をもう一度行おうとしても、復元は再び失敗します。

**推奨される対策：**

接続に失敗する原因を解消します (たとえば、interfaces 命令のマスク設定の間違い)。 次に、手動で **smbd** デーモンと **nmbd** デーモンを停止します。階層をサービス中に戻します。 作成時に復元に失敗した場合は、階層を再作成します。

デーモンを停止すると、復元時に確実に設定ファイルが再読み込みされます。

# 6.15.8.1. エラーメッセージ

## 共通エラーメッセージ

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 109009 | Error getting netbios name from the instance information field for tag "Samba-smb.conf" on server "ServerA". (サーバ"ServerA"上のタグ"Samba- smb.conf"用インスタンス情報からの netbios 名の取得時にエラー。) | インスタンス情報から NetBIOS 名を抽出するのに失敗しました。情報フィールドに設定ファイル名と NetBIOS 名が含まれている か確認してください。 |
| 109015 | The Samba utility testparm failed. Unable to parse Samba configuration file.(Samba ユーティリティ、testparm の実行に失敗しました。Samba 構成ファイルの構文解析ができません。) | 設定ファイルの解析に使用した Samba ユーティリティ **testparm** の実行に失敗しました。コマンド･ラインから、階層で使用する設定ファイルを指定して **testparm** を実行し、失敗の内容を確認してください。 |
| 109019 | Failed to initialize for reading of the Configuration File "/tmp/smb.ini.1234". (設定ファイル"/tmp/smb.ini.1234"の読み込み作業の初期化に失敗しました。) | 生成された設定ファイルの読み込みに失敗しました。ユーティリティ **testparm** が生成したファイルに間違いがあります。 |
| 109022 | Error getting configuration name from the instance information field for tag "Samba-smb.conf" on server "ServerA". (サーバ"ServerA"上のタグ "Samba- smb.conf" 用インスタンス情報からの設定名の取得時にエラー。) | インスタンス情報から NetBIOS 名を抽出するのに失敗しました。情報フィールドに設定ファイル名と NetBIOS 名が含まれているか確認してください。 |
| 109030 | Failure opening "/var/lock/samba/smbd.pid" on server "ServerA": "File Not Found". (サーバ"ServerA"上の "/var/lock/samba/smbd.pid"のオープンに 失敗しました："ファイルが見つかりません"。) | 表示している理由によってデーモン･プロセス ID ファイルのオープンに失敗しました。表示されたエラー･コードに基づいて問題を修正してください。 |
| 109050 | Open of the testparm output file failed. (testparm 出力ファイルのオープンに失敗しました。) | ファイルが存在しないか、ファイルにデータが存在しないため、 **testparm** を実行して作成した出力ファイルのオープンに失敗しました。コマンド･ラインから、階層で使用する設定ファイルを指定して **testparm** を実行し、失敗の内容を確認してください。 |

# 6.15.8.2. リソース階層作成時のエラーメッセージ

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 109001 | Usage: "valid_cf"CfgPath CfgName TemplateSys<br><br>(使用法："valid_cf" CfgPath CfgName TemplateSys) | **valid_cf** スクリプトは、設定ファイルのあるディレクトリ、設定ファイル名、設定ファイルを検証するテンプレート・システム名という 3 つの引数を必要とします。3 つすべてを指定する必要があります。 |
| 109002 | Must specify an absolute path to the "smb.conf" configuration file. ("smb.conf" 設定ファイルの絶対パスを指定してください。) | **choice_cf** スクリプトと **valid_cf** スクリプトを実行するときには、設定ファイルの絶対パスを指定する必要があります。 |
| 109003 | The file "smb.conf" does not exist in "/etc". (ファイル”smb.conf”が”/etc” にありません。) | **valid_cf** スクリプトを実行して指定の設定ファイルを検証するときには、設定ファイルの正しいパスを指定する必要があります。 |
| 109004 | The path "/export/fs" in "/etc/samba/smb.conf" does not reside on a shared file system. (/etc/samba/smb.conf" 内のパス "/export/fs" が共有ファイル・システム上にありません。) | Samba 設定の共有には LifeKeeper がファイル・システム・リソース経由で保護できる path 命令が含まれている必要があります。設定ファイルを編集して、LifeKeeper が保護できるファイル・システムのパスを変更してください。 |
| 109005 | Usage: "choices_cf" CfgFile (使用法：“choices_cf” CfgFile) | **choice_cf** スクリプトは既存の設定ファイルのフルパスを必要とします。正しいパスを指定してください。 |
| 109006 | Samba Configuration file not specified. (Samba 設定ファイルが指定されていません。) | Samba リソース階層の作成に使用する 設定ファイルが指定されていません。 |
| 109007 | Cannot bring hierarchy "Samba-smb.conf" in service on server "ServerA".(階層 "Samba-smb.conf" をサーバ "ServerA" でサービス中にできません。)<br><br>**処置：** 問題を修正した後、手動で階層をサービス中にします。 | 作成の最後に階層をサービス中にできませんでした。ログ・ファイルを参照して、失敗の原因を調べてください。 |
| 109010 | The Samba configuration file does not have the interfaces directive defined. This directive is required to create Samba | 設定ファイルに interfaces 命令が記述されていません。 または、 |

| | | |
|---|---|---|
| | File Share hierarchies.(Samba 設定ファイルに interfaces 命令が定義されていません。この命令は Samba のファイル共有階層を作成するには必須です。) | interfaces 命令で localhost (127.0.0.1) 以外の IP アドレスが指定されていません。 |
| 109011 | The Samba configuration file does not have a correctly formatted interfaces directive. The interfaces directive must be in full dotted decimal IP address format with or without the mask parameter.(Samba 設定ファイルの interfaces 命令の形式が正しくありません。interfaces 命令は、マスク・パラメータの有無にかかわらず、ドット形式 10 進 IP アドレスで記述する必要があります。) | interfaces 命令では、aaa.bbb.ccc.ddd または aaa.bbb.ccc.ddd/mask という形式で 1 つ以上の IP アドレスをスペースで区切って指定する必要があります。 |
| 109012 | The Samba configuration file section "FileShare1" does not have a path directive defined. All Samba shares must have a path directive.(Samba 設定ファイルのセクション "FileShare1"に path 命令が定義されていません。すべての Samba の共有に path 命令を記述する必要があります。) | 設定ファイルで指定された共有に path 命令が記述されていないか、path 命令 の値が指定されていません。 |
| 109014 | No IP resources defined on server "ServerA".(サーバ "ServerA" に IP リソースが定義されていません。)<br><br>**処置：** Samba 設定ファイルの interfaces 命令で定義された IP アドレス用の IP リソースを作成します。 | 指定されたサーバに、Samba リソース階層の作成に必要な LifeKeeper が保護する IP リソースが存在しません。 |
| 109016 | The IP(s) "100.25.104.25,100.35.104.26 defined in the interfaces directive are not under LifeKeeper protection.(interfaces 命令で定義された IP(s)、"100.25.104.25,100.35.104.26"が LifeKeeper によって保護されていません。)<br><br>**処置：** Samba 設定ファイルの interfaces命令で定義された保護されていないアドレス用の IP リソースを作成します。 | interfaces 命令で指定された IP アドレス には、LifeKeeper が保護する IP リソースが存在する必要があります。表示された IP アドレスは LifeKeeper で保護されていません。 |
| 109017 | Missing configuration file name.(設定ファイルが存在しません。) | Samba リソースの作成時に **testparm** を 実行しようとしたときに、設定ファイルが存在していませんでした。 |
| 109018 | No Samba shares found in "/etc/samba/smb.conf".("/etc/samba/smb.conf"に Samba の共有が見つかりませんでした。) | リソースに対して指定された Samba 設定ファイルには少なくとも 1 つファイルの共有またはプリンタの共有を含んでいる必要があります。 |
| 109020 | Bad configuration file. No section information found in the file "/tmp/smb.ini.1234".(設定ファイルが間違っています。ファイル"/tmp/smb.ini.1234" にセクション情報が見つかりません。) | リソースに対して指定された Samba 設定ファイルには少なくとも 1 つファイルの共有またはプリンタの共有を含んでいる必要があります。 |
| 109021 | Creation of Samba hierarchy with tag "Samba-smb.conf" on | Samba 階層の作成に失敗しまし |

| | | |
|---|---|---|
| | server "ServerA" failed.(サーバ "ServerA"上のタグ "Samba-smb.conf" の Samba 階層の作成に失敗しました。) | た。他のエラー・メッセージを参考にして、失敗の原因を判断してください。 |
| 109023 | The file system resource "filesys1328" is not in-service on server "ServerA".(ファイル・システム・リソース"filesys1328"がサーバ"ServerA"でサービス中ではありません。) | Samba 階層の従属している子として必要なファイル・システム・リソースステンプレート・サーバでサービス中ではありません。リソースをサービス中にして、リソースの作成をもう一度行ってください。 |
| 109024 | Selected IP resource "ip-100.25.104.26" does not exist on server "ServerA".(選択した IP リソース"ip-100.25.104.26" がサーバ "ServerA" に存在しません。)<br><br>**処置：** 操作をやり直します。 | 指定された IP リソース・タグはもう存在しませんが、Samba 階層の作成には必要です。IP リソースを再作成して、リソースの作成をもう一度行ってください。 |
| 109025 | LifeKeeper was unable to create a dependency between the Samba hierarchy "Samba-smb.conf" and the IP resource "ip-100.25.104.26" on server "ServerA".(Samba 階層 "Samba-smb.conf"とサーバ "ServerA"上の IP リソース"ip-100.25.104.26" との間に従属関係を作成できませんでした。) | Samba リソースと IP リソース間の従属関係の作成に失敗しました。他のメッセージを参考にして、失敗の原因を判断してください。 |
| 109026 | The Samba configuration file does not have a netbios name directive defined.(Samba 設定ファイルに netbios name 命令が定義されていません。)<br><br>**処置：** 設定ファイルの global セクションに netbios name 命令を追加してください。 | すべての設定ファイルに NetBIOS 名が記述されている必要があります。設定ファイルに NetBIOS name 命令を追加してください。 |
| 109029 | The Samba configuration file "%s" directive defines a directory that does not exist. The %s can contain pid directory,lock dir, or lock directory.(Samba 設定ファイルの "%s" 命令が存 在しないディレクトリを定義しています。%s は pid directory または lock directory (lock dir) である可能性があります。) | すべての設定ファイルには、この命令によって指定される既存のディレクトリが記述されている必要があります。存在しないディレクトリを追加してください。 |
| 109035 | The Samba directive "bind interfaces only" must be set to "Yes".(Samba 命令"bind interfaces only" を "Yes" に設定する必要があります。)<br><br>**処置：** "bind interfaces only"の値を "Yes"に変更して、階層を再作成します。 | すべての設定ファイルで bind interfacesonly 命令が Yes に設定されている必要があります。命令を正しく "Yes" に設定してください。 |
| 109036 | Selected Printer resource "lp-admin" does not exist on server "ServerA". (選択したプリンタ・リソース "lp-admin"がサーバ"ServerA"に存在しません。)<br><br>**処置：** 操作をやり直します。 | 指定されたプリント・サービス・リソース・タグはもう存在しませんが、Samba 階層の作成には必要です。プリント・サービス・リソースを再作成して、リソースの作成をも |

| | | |
|---|---|---|
| | | う一度行ってください。 |
| 109037 | LifeKeeper was unable to create a dependency between the Samba hierarchy "Samba-smb.conf" and the Printer resource "lp-admin" on server "ServerA".(Samba 階層 "Samba-smb.conf" とサーバ "ServerA"上のプリンタ・リソース"lp-admin" との間に従属関係を作成できませんでした。) | Samba リソースとプリント・サービス・リソース間の従属関係の作成に失敗しました。他のメッセージを参考にして、失敗の原因を判断してください。 |
| 109038 | The Printers(s) "lpadmin" defined in the configuration file are not under LifeKeeper protection.(設定ファイルで定義されているプリンタ "lpadmin" が LifeKeeper によって保 護されていません。)<br><br>**処置：** Samba 設定ファイルで定義された保護されていないプリンタ用のプリンタ・インスタンスを作成します。 | 設定ファイルで定義されたプリンタには、LifeKeeper が保護するプリント・サービス・リソースが存在する必要があります。表示されたプリンタは LifeKeeper で保護されていません。 |
| 109041 | The selected configuration file "/etc/samba/smb.conf" is in use by Samba resource "Samba-smb.conf".(選択した設定ファイル "/etc/samba/smb.conf" は Samba リソース "Samba-smb.conf" が使用中です。)<br><br>または<br><br>The selected netbios name "LKServer" is in use by Samba resource "Samba-smb.conf".(選択した netbios 名 “LKServer” は Samba リソース"Samba-smb.conf"が使用中です。) | Samba 設定ファイルまたは netbios 名はクラスタ内で 1 度だけしか保護できません。設定ファイル名を変更するか、新しい NetBIOS 名を選択します。 |

# 6.15.8.3. リソース拡張時のエラーメッセージ

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 109008 | Replication of config file to target server "ServerB" failed. The "mkdir" of "/etc/samba/config_files" failed. (ターゲット・サーバ"ServerB" に設定ファイルを複製するのに失敗しました。"/etc/samba/config_files"の”mkdir” に失敗しました。)<br><br>または<br><br>Replication of config file to target server "ServerB" failed. The "/etc/samba/config_files" failed. (ターゲット・サーバ"ServerB" に設定ファイルを複製するのに失敗しました。"/etc/samba/config_files" の "lcdrcp" に失敗しました。) | 拡張時に Samba 設定ファイルのコピーに失敗しました。 **mkdir** またはリモート・コピーに失敗しています。 |
| 109042 | WARNING: The configuration file "/etc/samba/smb.conf" currently exists on server "ServerB" and will be overwritten if this resource is extended. (警告：設定ファイル "/etc/samba/smb.conf" は現在サーバ "ServerB"上に存在し、このリソースを拡張すると上書きされます。) | リソース階層で使用する設定ファイルが既にバックアップ・サーバに存在しており、上書きされます。ファイルの上書きをしない場合は、拡張処理をキャンセルします。 |

# 6.15.8.4. リソース復元時のエラーメッセージ

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 109027 | Failed start of smbd as daemon process. (smbd をデーモン･プロセスとして起動するのに失敗しました。) | **smbd** をデーモンとして起動するのに失敗しました。Samba ログ･ファイルで 追加の情報を確認してください。 |
| 109028 | Failed start of nmbd as daemon process. (nmbd をデーモン･プロセスとして起動するのに失敗しました。) | **nmbd** をデーモンとして起動するのに失敗しました。Samba ログ･ファイルで追加の情報を確認してください。 |

# 6.15.8.5. リソース削除時のエラーメッセージ

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 109033 | Failed start of smbd as daemon process.Attempting to stop via SIGKILL.　(smbd をデーモン･プロセスとして起動するのに失敗しました。SIGKILL による停止を試みます。) | **smbd** デーモン･プロセスの通常終了に失敗したため、デーモンは強制終了されます。 |
| 109034 | Failed to stop nmbd daemon process. Attempting to stop via SIGKILL. (nmbd デーモン･プロセスの停止に失敗しました。SIGKILL による停止を試みます。) | **nmbd** デーモン･プロセスの通常終了に失敗したため、デーモンは強制終了されます。 |

# 6.15.8.6. リソース監視時のエラーメッセージ

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 109031 | Connection attempt to smbd daemon on address "100.25.104.26" failed. (アドレス"100.25.104.26" の smbd デーモンへの接続に失敗しました。) | 表示された IP アドレスへの接続に失敗したため、 **smbd** デーモン・プロセスのヘルス・チェックに失敗しました。ローカル・リカバリーが試みられます。 ローカル・リカバリーに失敗した場合、階層はバックアップ・サーバに切り替えられます。 |
| 109032 | Connection attempt to nmbd daemon failed for broadcast address "100.25.107.255" using netbios name "FILESERV1". (netbios 名"FILESERV1" を使用したブロードキャスト・アドレス "100.25.107.255" による nmbd デーモンへの接続に失敗しました。) | 表示されたブロードキャスト・アドレスへの接続に失敗したため、 **nmbd** デーモン・プロセスのヘルス・チェックに失敗しました。ローカル・リカバリーが試みられます。ローカル・リカバリーに失敗した場合、階層はバックアップ・サーバに切り替えられます。 |
| 109039 | No dependent IP resources were found for tag "Samba-smb.conf" on server "ServerA". (サーバ "ServerA" 上のタグ "Samba-smb.conf" に従属した IP リソースが見つかりませんでした。) | **nmbd** デーモンのヘルス・チェックを行おうとしたときに、Samba 階層に IP の子リソースが見つかりませんでした。ログを調べて、失敗の原因を判断してください。 |
| 109040 | Unable to determine IP address and/or mask for IP resource "ip-100.25.104.25". (IP リソース"ip-100.25.104.25" の IP アドレスまたはマスクを決定できません。) | IP リソース情報から IP アドレスとマスクを抽出するのに失敗しました。情報フィールドに IP アドレスとマスクが含まれているか確認してください。 |

# 6.15.8.7. 設定ファイル同期ユーティリティのエラーメッセージ

| エラー番号 | エラーメッセージ | 説明 |
|---|---|---|
| 109013 | The hierarchy for the specified configuration file does not have an equivalency with the target system. Select another target system or configuration file. (指定された設定ファイルの階層にターゲット・システムとのイクイバレンシがありません。別のターゲット・システムまたは設定ファイルを選択してください。) | **synccfg** ユーティリティの「-targetSys」引数で指定されたシステム にローカル・システム上の Samba 階層 に対するイクイバレンシがありません。 |
| 109044 | Usage: synccfg -t TargetSys -c ConfigFile(使用法：synccfg -t TargetSys -c ConfigFile) | **synccfg** ユーティリティは引数を 2 つ必要とします。「-t TargetSys」は設定ファイルのコピー先で、「-c ConfigFile」はコピーする設定ファイルの名前です。両方の引数を指定する必要があります。 |
| 109045 | Specified Configuration file does not exist on this server.(指定された設定ファイルはこのサーバに存在しません。) | **synccfg** ユーティリティの「-cConfigFile」引数で指定された設定ファイルがこのシステムには存在しません。 |
| 109046 | The target system specified for updating is the same as the template system. Select a new target system.(更新のために指定されたターゲット・システムがテンプレート・システムと同じです。別のターゲット・システムを選択してください。) | **synccfg** ユーティリティの「-TargetSys」引数で指定されたコピー先がローカル・サーバになっています。 ターゲット・サーバはローカル・システムであってはいけません。 |
| 109047 | The specified configuration file is not used in any Samba instances. Select another configuration file.(指定された設定ファイルは Samba インスタンスで使用されていません。別の設定ファイルを選択してください。) | **synccfg** ユーティリティの「-cConfigFile」引数で指定された設定ファイルがどの LifeKeeper Samba インスタンスでも使用されていません。このユーティリティでコピーできるのは、LikeKeeper で保護された設定ファイルだけです。 |
| 109048 | The syncronization of the Samba configuration file failed with a status of "%s".(Samba 設定ファイルの同期に失敗しました。ステータスは"%s"です。) | ローカル・システムとターゲット・システム間の設定ファイルの同期に失敗しました。ステータスは失敗した理由を示しています。 |
| 109049 | The hierarchy for the specified configuration file is ISP on the target node. Select another target system or configuration file.(ターゲット・ノートで ISP です。別のターゲット・システムまたは設定ファイルを選択してください。) | synccfg ユーティリティの「-tTargetSys」引数で指定されたシステムで Samba 階層が現在 ISP です。ターゲット・システムは ISP であってはなりません。 |

# 6.16. SAP Recovery Kit 管理ガイド

## SAP Recovery Kit のドキュメント

SIOS Protection Suite for Linux SAP Recovery Kit は、LifeKeeper 環境で障害の発生したプライマリサーバからバックアップサーバに SAP NetWeaver を復旧する仕組みを提供します。SAP Recovery Kit は、他の SIOS Protection Suite Recovery Kit (IP Recovery Kit 、NFS Server Recovery Kit 、NAS Recovery Kit 、および Oracle Recovery Kit などのデータベースリカバリキット) と連係して包括的なフェイルオーバ保護を提供します。

本ドキュメンテーションには、SAP リソースの設定と管理に不可欠な情報が記載されています。SPS を正しく実装させるには、設定手順を慎重に実行してください。また、関連するリカバリキット のドキュメンテーションも参照してください。

SAP Recovery Kit の概要

## ドキュメンテーション一覧

以下は、SIOS Technology Corp が発行している LifeKeeper 関連情報の一覧です。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション (LifeKeeper GUI の **[Help]** メニューからも参照できます)
- SIOS Technology Corp. Documentation and Support
- 略語と定義 - 本ドキュメンテーション全体で使用される略語と用語の一覧およびその意味。
- LifeKeeper/SAP のアイコン - 使用されるアイコンの一覧およびその意味。

## 参考ドキュメント

以下は、本ドキュメンテーション全体で参照した SAP 関連のドキュメントです。

- *NetWeaver 7.x:*
  - Software Provisioning Manager 1.0 – https://support.sap.com/en/tools/software-logistics-tools.html
  - System Provisioning -> System Provisioning Scenarios -> Install a System using Software Provisioning Manager -> Installation Option of Software Provisioning Manager 1.0 SP 00 (or higher) -> Installation Guides – Application Server Systems
- *S/4HANA 1809:*
  - Software Provisioning Manager 2.0 – https://support.sap.com/en/tools/software-logistics-tools.html
  - System Provisioning -> System Provisioning Scenarios -> Install a System using Software

Provisioning Manager -> Installation Option of Software Provisioning Manager 2.0 SP 00 (or higher) -> Installation Guides – Application Server Systems

- *SAP R/3 in Switchover Environments* (SAP ドキュメント 50020596)
- *R/3 Installation on UNIX:* (データベース固有)
- *SAP Web Application Server in Switchover Environments*
- *Component Installation Guide SAP Web Application Server* (データベース固有)
- *SAP ノート:*
  - 1662610 – Support details for SIOS Protection Suite for Linux
  - 2001838 – Maintenance Mode for SIOS Protection Suite for Linux SAP Recovery Kit
  - 1552925 – Linux: High Availability Cluster Solutions
  - 1693245 – SAP HA Script Connector Library
  - 201144 – INST: 4.6C R/3 Installation on UNIX
  - 7316 – SysLog: Module rslgsmcc(071) Error 003

# 6.16.1. 略語と定義

本マニュアル全体で、使用される略語は次のとおりです。

| 略語 | 意味 |
|---|---|
| **AS** | SAP アプリケーションサーバ。一般的には、AS は任意のアプリケーションサーバを指すが、本マニュアルにおいては、CI 以外の冗長化されたアプリケーションサーバを指す。そのため、アプリケーションサーバには LifeKeeper による保護は必要ない。 |
| **ASCS** | ABAP SAP セントラルサービスインスタンス。NetWeaver ABAP 環境向けのメッセージサービスと ENQEUE サービスで構成される SAP インスタンスである。このインスタンスは、単一障害点であり、LifeKeeper で保護する必要がある。 |
| **(ASCS)** | バックアップ ABAP SAP セントラルサービスインスタンスサーバ。プライマリ ASCS サーバに障害が発生した場合の ASCS ホストサーバである。 |
| **DB** | SAP データベースインスタンス。このデータベースは、Oracle またはその他の SAP がサポートするデータベースである。このインスタンスは、単一障害点であり、LifeKeeper で保護する必要がある。CI と DB は同一のサーバ上にあっても、異なるサーバ上にあってもよい。また、DB は、プライマリ DB サーバを指すこともある。 |
| **(DB)** | バックアップデータベースサーバ。プライマリ DB サーバに障害が発生した場合の DB ホストサーバである。1 つのサーバを データベース (DB) とセントラルインスタンス (CI) の両方のバックアップにすることも可能。 |
| **ENSAv1** | スタンドアローンエンキューサーバー バージョン1。これは、7.51以前のSAPカーネルバージョンで使用可能なエンキューサーバーのバージョンです。 |
| **ENSAv2** | スタンドアローンエンキューサーバー バージョン2。このバージョンのエンキューサーバーは、SAPカーネルバージョン7.51以降で使用できます。 |
| **ERS** | ENQUEUE レプリケーションサーバ。 |
| **ERSv1** | エンキューレプリケーションサーバー バージョン1。これは、7.51以前のSAPカーネルバージョンで使用可能なエンキューレプリケーションサーバーのバージョンです。 |
| **ERSv2** | エンキューレプリケーションサーバー バージョン2。このバージョンのエンキューレプリケーションサーバーは、SAPカーネルバージョン7.51以降で使用できます。 |
| **HA** | 高可用性。 |
| **ID** または **<ID>** | SAP インスタンスの数値識別子(2桁）。 |
| **< *INST* >** | SAP インスタンス用のディレクトリ。その名前はインスタンスに含まれるサービスとインスタンス番号に由来する。例えば、CI <INST> は、 *DVEBMGS00* など。 |
| **PAS** | プライマリアプリケーションサーバインスタンス。 |
| **SAP インスタンス** | 同時に開始および停止されるプロセスのグループ。 |
| **SAP シス** | SAP インスタンスのグループ。 |

| | |
|---|---|
| テム | |
| **<*sapmnt*>** | SAP のホームディレクトリ。デフォルトでは */sapmnt* だが、インストール時にユーザにより変更可能。 |
| **SCS** | SAP セントラルサービスインスタンス。NetWeaver Java 環境向けのメッセージサービスと ENQEUE サービスで構成される SAP インスタンスである。このインスタンスは、単一障害点であり、LifeKeeper で保護する必要がある。 |
| **(SCS)** | バックアップ SAP セントラルサービスインスタンスサーバ。プライマリ SCS サーバに障害が発生した場合の SCS ホストサーバである。 |
| **SID** または **<SID>** | システム ID。 |
| **sid** または **<sid>** | SID の小文字バージョン。 |
| **SPOF** | 単一障害点(Sinlge point of Failure) |

# 6.16.2. LifeKeeper – SAP アイコン

以下のアイコンは、LifeKeeper 環境における SAP リソースのステータスを解釈する上で重要です。これらのアイコンは、LifeKeeper UI 上に表示されます。

| | |
|---|---|
| | アクティブ - リソースはアクティブで In Service である (通常状態)。 |
| | スタンバイ - リソースはバックアップノードにあり、プライマリリソースに障害が発生したら引き継ぐ準備ができている (通常状態)。 |
| | 障害 - リソースに障害が発生した。リソースを In Service にする (リソース上で右クリックし、[In Service] を選択して Enter を押す) ことができる。リソースにまだ障害が残っている場合、リカバリは失敗する。 (障害状態)。 |
| | **要注意** - SAP リソースに障害が発生したか、警告状態である。障害が発生しており、自動リカバリが有効な場合 (保護レベルが「Full」または「Standard」)、LifeKeeper は自動的にリソースの復旧を試みる。<br>SAP リソースを右クリックして、 **[Properties]** を選択すると警告状態にあるリソースが表示される。 **黄色** の SAP 状態は通常だが、SAP リソースの実行が遅いか、パフォーマンスボトルネックがあることを示す。 |
| | **保護レベルの更新** - SAP リソースの保護レベルを変更するためのアクションアイコン。 |
| | **リカバリレベルの更新** - SAP リソースのリカバリレベルを変更するためのアクションアイコン。 |
| | **警告の処理** - 警告に障害を設定するためのアクションアイコン。 |
| | **開始** 、 **停止** 、 **移行** 、 **メンテナンスモード** の設定など、さまざまなアクションを選択するためのアクションアイコン。 |

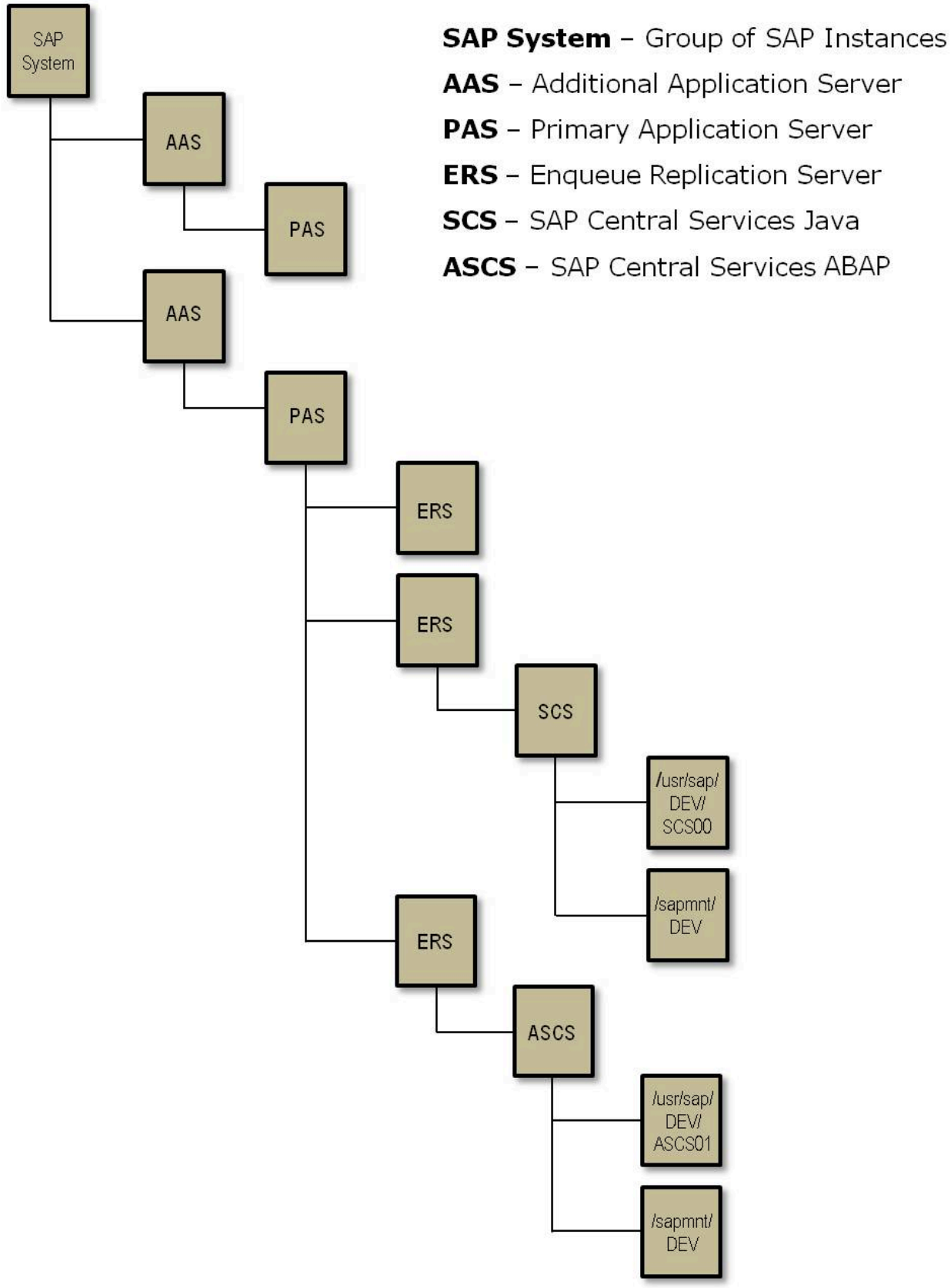
# 6.16.3. SAP Recovery Kit の概要

SAP NetWeaver フレームワークには、冗長化することのできないサービスが一部存在します。これらのサービスは、同一の SAP システム上に複数個存在することができないため、単一障害点になります。LifeKeeper SAP Recovery Kit は、これらの単一障害点に標準的な LifeKeeper 機能による保護を提供します。さらに、このキットでは、SAP インフラストラクチャのその他の部分についても、さまざまなレベルにおいて保護する機能を提供します。各インフラストラクチャコンポーネントの保護は、階層内の単一のリソースで表現されます。

SAP Recovery Kit は、さまざまな SAP インスタンス、すなわち SAP プライマリアプリケーションサーバ (PAS) インスタンス、ABAP SAP セントラルサービス (ASCS) インスタンス、SAP セントラルサービス (SCS) インスタンスに対して監視とスイッチオーバを提供します (セントラルサービスインスタンスは、ENQUEUE およびメッセージサーバを保護する)。SAP Recovery Kit は、適切なデータベースリカバリキットと連携してデータベースを保護し、Network File System (NFS) Server Recovery Kit と連携して NFS マウントを保護します。また、必要に応じてクラスタ内のネットワークカード間で移動できる仮想 IP アドレスを提供するため、IP Recovery Kit も使用されます。Network Attached Storage (NAS) Recovery Kit を使用して、ローカル NFS マウントを保護できます。これらさまざまな Recovery Kit を使用して、アプリケーション環境のすべてのコンポーネントを保護する SAP リソース階層を構築します。

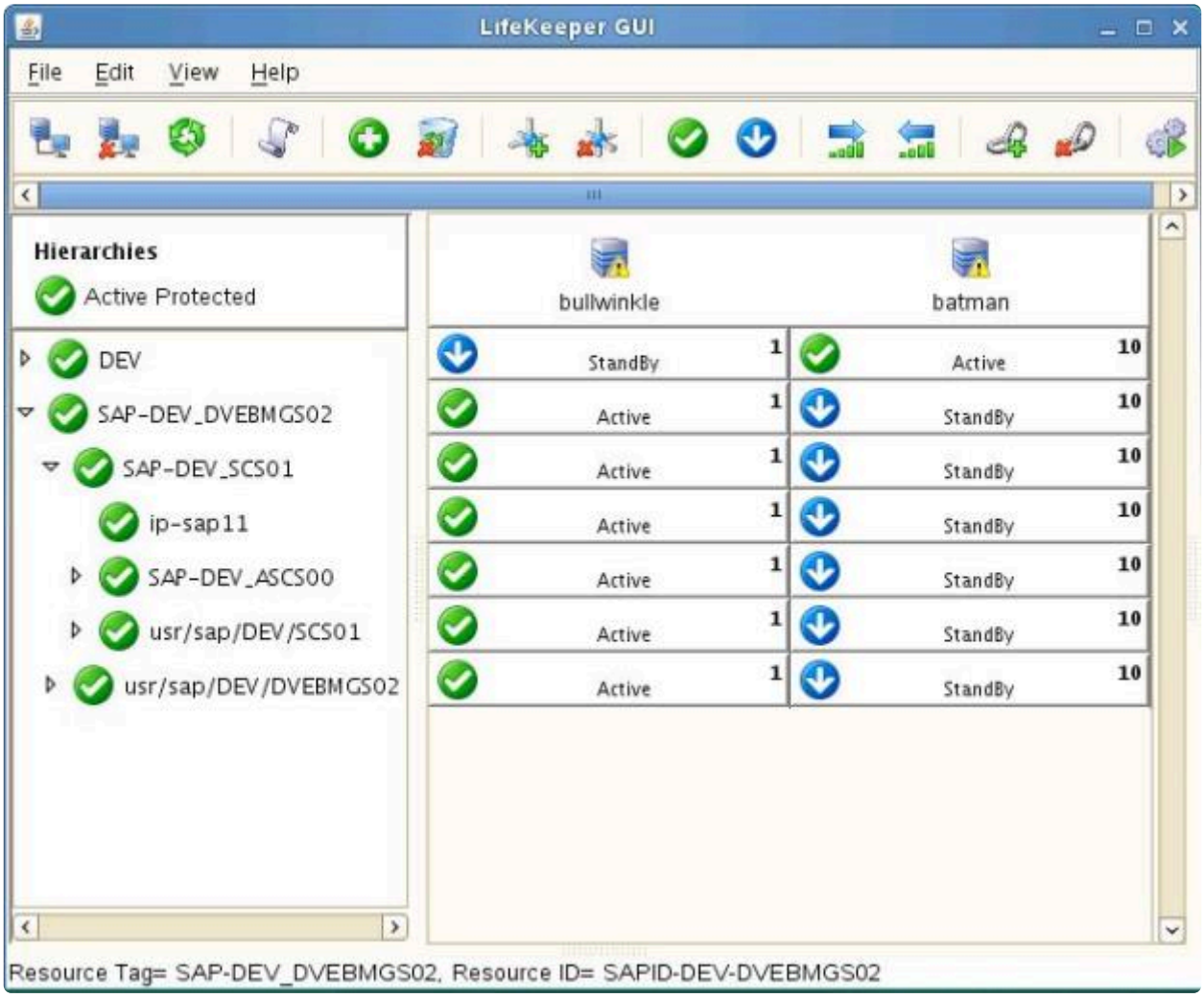
各 Recovery Kit は、保護下にあるアプリケーションの状態を監視し、ローカルおよび別のクラスタサーバ上のアプリケーションを停止し再開できます。

**SAP システム階層のマップ**




典型的な SAP リソース階層は、LifeKeeper GUI に次のように表示されます。

## SAP リソース階層




注記: ディレクトリ */usr/sap/trans* は、SAP 環境では必須ではありません。SAP NetWeaver Java のみの環境では、このディレクトリは存在しません。

注記: SPS-L v9.4.0以降で作成されたERSリソースは、独立した独自の階層で動作します。詳細については、LifeKeeperのERSリソースタイプ を参照してください。

# 6.16.4. SIOS Protection Suite for SAP ソリューションページ

## SAP High Availability Interface 7.73

| パッケージ | バージョン | OS / アプリケーションバージョンのサポート |
|---|---|---|
| LifeKeeper Core (SAP) | 9.5.2 | Red Hat Enterprise Linux 7 (7.0 から 7.9 まで)<br>Red Hat Enterprise Linux 8 (8.0 から 8.2 まで)<br>SUSE LINUX Enterprise Server (SLES) 12 (SP1 から SP5 まで)<br>SUSE LINUX Enterprise Server (SLES) 15 (15.0 から SP2 まで)<br>Oracle Linux 7 (7.0 から 7.6 まで、UEK R4 カーネルを含む) |
| LifeKeeper SAP Recovery Kit | 9.5.2 | SAP NetWeaver 7.3、Enhancement Package 1 を含む<br>SAP NetWeaver 7.4<br>SAP NetWeaver 7.5<br>SAP NetWeaver AS for ABAP 7.51 innovation package<br>SAP NetWeaver AS for ABAP 7.52 innovation package<br>SAP S/4HANA 1809 Platform<br>SAP S/4HANA 1909 Platform<br>SAP S/4HANA 2020 Platform |
| LifeKeeper NFS Server Recovery Kit | 9.5.2 | カーネルバージョンが 2.6 以降の Linux ディストリビューションのエクスポートされた NFS ファイルシステム |
| LifeKeeper Supported Databases for SAP | 9.5.2 | Oracle, SAP ASE (Sybase), SAP HANA, SAP DB (MaxDB), IBM DB2 |
| SAP HA Interface Connector | 7.73 | SAP High Availability Interface 7.73, 7.77, 7.81 |

✻ v9.5.0より、SIOSは新しいSAP HANA Application Recovery Kitをリリースしました。SIOSは **2022年3月31日** まで、9.4.xリリースでのSAP HANA gen/appベースのRecovery Kitのサポートを継続します。SIOS Protection Suite for Linux v9.5以降をご利用の場合は、新しい（ビルトインの）SAP HANA Application Recovery Kitを使用する必要があります。

! 既存のSAP HANA gen/appベースのRecovery Kitはv9.5.0では **サポートされていません** 。SIOS Protection Suite for Linux v9.5.0へのアップグレードを希望するユーザーは、既存のSAP HANA gen/appベースのRecovery Kitから新しい SAP HANA Recovery Kit に **移行する必要があります** 。詳細については、 SAP HANA Gen/AppからSAP HANA Recovery Kitへのアップグレード を参照してください。

**注意：** 拡張SAPサポート用に構築されたオペレーティングシステムのバージョン（Red Hat Enterprise Linux for SAP Business Applications、Red Hat Enterprise Linux for SAP Solutions、SUSE Linux Enterprise Server for SAP Applicationsなど）も、実行中のLinuxカーネルのバージョンが上記のサポート対象OSのバージョンと同じであればサポートされます。

# 6.16.5. ハードウェア / ソフトウェア要件

SPS SAP Recovery Kit をインストールして設定する前に、使用している設定が以下の要件を満たしていることを確認してください。

- **サーバ** 。この Recovery Kit には、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート で説明されている要件に従って設定した 2 台以上のコンピュータが必要です。

- **共有ストレージ** 。SAP プライマリアプリケーションサーバ (PAS) インスタンス、ABAP SAP セントラルサービス (ASCS) インスタンス、SAP セントラルサービス (SCS) インスタンス、およびプログラムファイルは、SPS 環境の共有ディスク上に存在する必要があります。ERSv2 構成の場合、ERS インスタンスのファイルシステムは共有ディスクに存在する場合もあります。

- **IP ネットワークインターフェース** 。各サーバは、イーサーネット TCP/IP をサポートするネットワーク・インターフェースを 1 つ以上必要とします。IP スイッチオーバが正しく動作するには、ローカルネットワークに接続されているユーザシステムが標準の TCP/IP 仕様に準拠している必要があります。

  **注記**: 各サーバが必要とするネットワークインターフェースが 1 つだけであっても、異種メディア要件、スループット要件、単一障害点の排除、ネットワークのセグメンテーションといった多くの理由のために、複数のインターフェースの使用を推奨します (詳細については、IP ローカルリカバリと設定に関する考慮事項 を参照してください)。

- **オペレーティングシステム** 。Linux オペレーティングシステム。(サポート対象のディストリビューションおよびカーネルのバージョンについては、SPS for Linux リリースノート を参照してください。)

- **SELinux** 。無効または permissive モード。 (詳細については、テクニカルノート を参照してください。)

- **TCP/IP ソフトウェア** 。各サーバは TCP/IP ソフトウェアを必要とします。

- **SAP ソフトウェア** 。各サーバは、SPS と SPS SAP Recovery Kit の設定に先だって SAP ソフトウェアをインストールして設定しておく必要があります。各サーバには同じバージョンをインストールする必要があります。最新のリリースの互換性および購入方法については、SPS for Linux リリースノート を参照するか、営業担当者にお問い合わせください。

- **SPS ソフトウェア** 。各サーバには同じバージョンの SPS ソフトウェアおよびパッチをインストールする必要があります。具体的な LifeKeeper の要件については、 SPS for Linux リリースノート を参照してください。

- **SPS IP Recovery Kit** 。この Recovery Kit は、リモートクライアントが SAP PAS、ASCS、または SCS インスタンスにアクセスする場合に必要です。各サーバで同じバージョンの Recovery Kit を使用する必要があります。

- **SPS for Linux NFS Server Recovery Kit** 。この Recovery Kit は、ほとんどの設定で必要です。各

サーバで同じバージョンの Recovery Kit を使用する必要があります。

- **SPS for Linux Network Attached Storage (NAS) Recovery Kit** 。この Recovery Kit は、一部の設定で必要です。各サーバで同じバージョンの Recovery Kit を使用する必要があります。

- **SPS for Linux Database Recovery Kit** 。SAP で利用されているデータベース用の SPS リカバリキットを各データベースサーバにインストールする必要があります。サポート対象のデータベースの詳細については、 SPS for Linux リリースノート を参照してください。SAP を設定する前に、SAP PAS、ASCS、または SCS インスタンスに対する LifeKeeper データベース階層を作成する必要があります。

**重要事項:**

- バージョン 7.3 より前の SAP を使用する場合、 **SAPHOSTAGENT** のダウンロードおよびインストール方法について、SAP ドキュメンテーションおよび注記を参照してください (「設定の計画」トピックの重要事項 を参照)。

- Core ソフトウェアおよび SAP Recovery Kit をインストールまたは削除する手順については、SPS for Linux インストールガイド を参照してください。

- インストール手順は、推奨する順序に従って行う必要があります。LifeKeeper を先にインストールした場合、SAP を正しくインストールできません。

- 必要な SPS Recovery Kit のそれぞれの設定方法については、各キットのドキュメンテーション (IP、NFS Server、NAS、およびデータベース Recovery Kit) を参照してください。

- スワップ領域やメモリ要件などの他のインストール要件については、SAP インストールドキュメンテーションを参照してください。

# 6.16.6. 設定上の考慮事項

このセクションでは、SAP の設定に着手する前に考慮すべき情報と、SAP の典型的な設定例を示します。また、SAP を LifeKeeper で設定および保護するための操作手順を示します。

SAP を LifeKeeper でサポートされているカーネル 2.4 または 2.6 を使用する Linux ディストリビューションにインストールする方法については、SAP が提供するデータベース固有の SAP インストールガイドを参照してください。

また、 SPS for Linux テクニカルドキュメンテーション を参照して、 SPS Core リソース (ファイルシステムリソースなど) を設定してください。

## サポートされる構成

SAP Highly Available (HA) 環境において取ることのできるデータベースサーバおよびアプリケーションサーバ構成は多数あります。LifeKeeper で保護するために SAP を設定する上で必要な具体的手順は、構成によって異なります。そのため、実際の要件に最も近い構成を見つけることが重要です。サポートされる構成例の一部を以下に示します。

ABAP+Java 構成 (ASCS およびSCS)

ABAP のみの構成 (ASCS)

Java のみの構成 (SCS)

上記の例に図示する構成は 2 つのサーバで構成されており、各サーバは、セントラルサービスインスタンスとともに、ERS インスタンス、データベースインスタンス、プライマリアプリケーションサーバインスタンス、0 個以上の追加の冗長性のあるアプリケーションサーバインスタンス (AS) をホストしています。冗長性のあるアプリケーションサーバなしで SAP を設定することは可能ですが、その場合、ASCS インスタンスまたは SCS インスタンスへのユーザによるログインが必要になります。これは、SAP によって推奨されていません。ASCS インスタンス、SCS インスタンス、およびデータベースサーバは、データベースファイルとアプリケーションファイルのために共有ファイルストレージにアクセスできます。

セントラルサービスは多くのリソースを使用するわけではなく、非常に高速でスイッチオーバ可能ですが、データベースはスイッチオーバ速度に大きな影響を与えます。そのため、データベースインスタンスとセントラルサービスインスタンス (ASCS および SCS) は、2 つの異なる LifeKeeper 階層によって保護することを推奨します。これらは、異なるサーバ上で実行することも、同一サーバ上で実行することも可能です。

## 設定に関する注記

以下は、HAクラスタ環境で実行するための SAP の設定に関する技術的な注記です。SAP を LifeKeeper で保護するための操作手順については、関連するトピックを参照してください。

ディレクトリ構造

仮想サーバ名

SAP ヘルスモニタリング

SAP ライセンス

自動スイッチバック

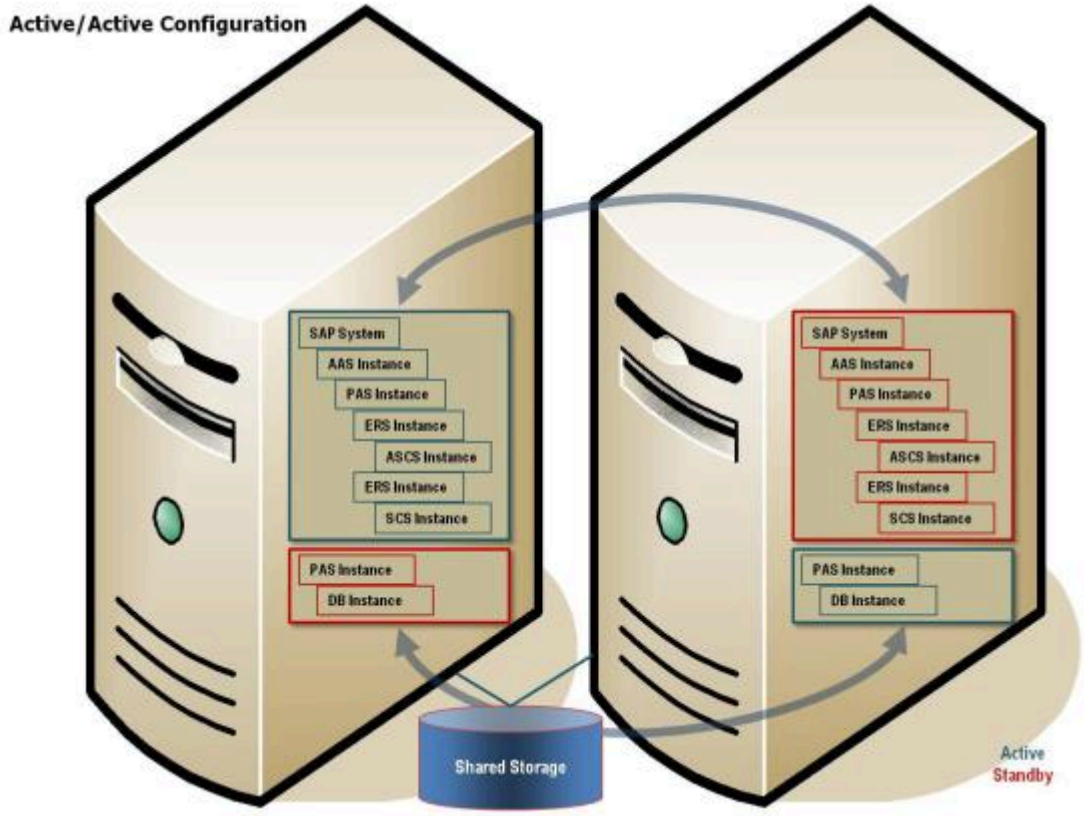
その他の注記

# 6.16.6.1. ABAP+Java の設定 (ASCS および SCS)

ABAP+Java の設定は、次のインストールで構成されます。

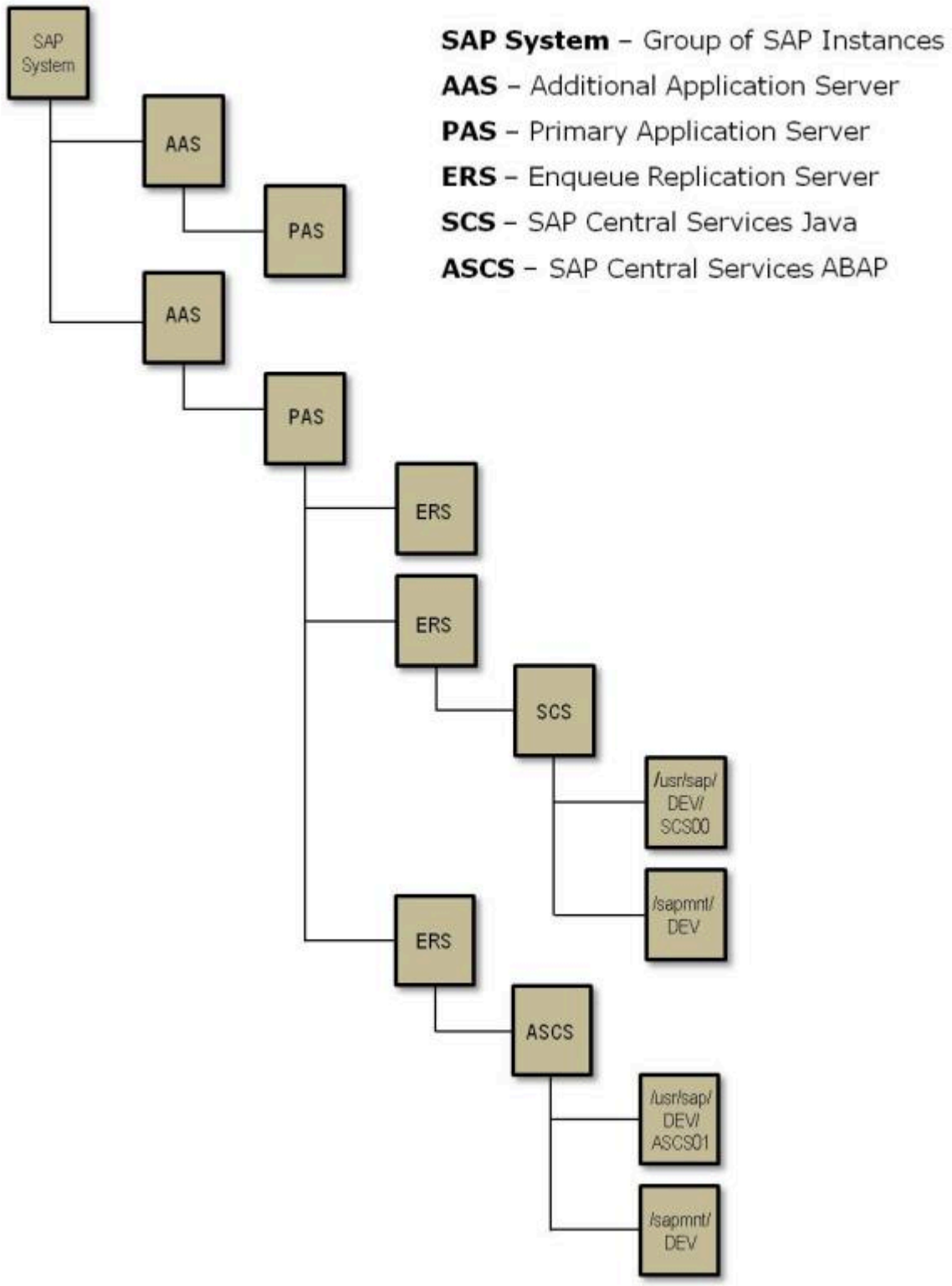
- ABAP セントラルサービスインスタンス (ASCS インスタンス)
- ASCS インスタンス用エンキューレプリケーションサーバインスタンス (ERS インスタンス) (オプション) (ASCS インスタンスと SCS インスタンスの両方は、自身の ERS インスタンスを持つ必要がある)
- Java 用セントラルサービスインスタンス (SCS インスタンス)
- SCS インスタンス用エンキューレプリケーションサーバインスタンス (ERS インスタンス) (オプション)
- データベースインスタンス (DB インスタンス) - ABAP スタックと Java スタックは、同一のデータベース内で独自のデータベーススキーマを使用します。
- プライマリアプリケーションサーバインスタンス (PAS)
- 追加のアプリケーションサーバインスタンス (AAS) - 追加のアプリケーションサーバインスタンス (AAS) を、プライマリアプリケーションサーバインスタンス (PAS) ホストとは別のホストにインストールすることを推奨します。

# SAP デュアルスタック (ABAP+Java) システム向けスイッチオーバクラスタ




上の例において、ASCS と SCS は、データベースとは別のリソース階層にあり、これらのセントラルサービスインスタンスは、データベースとは異なるサーバで起動しています。

# SAP 階層の例



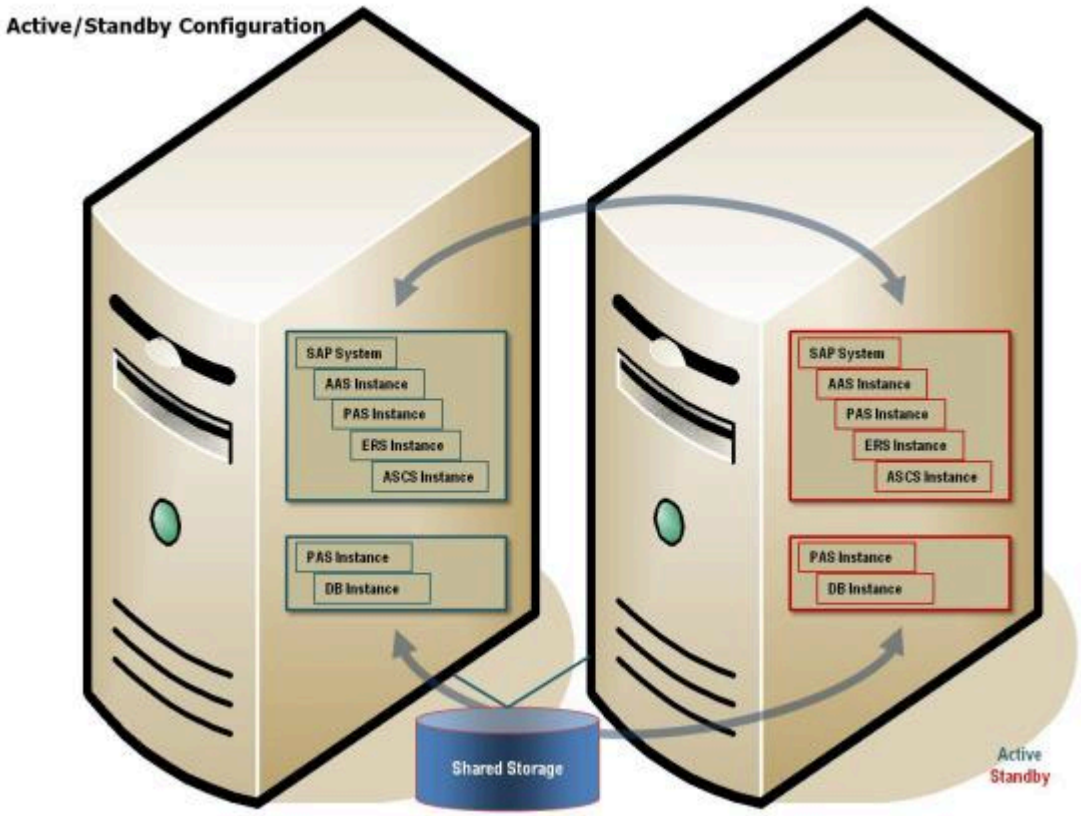

注記: SPS-L v9.4.0 以降で作成された ERS リソースは、独立した独自の階層で動作します。詳細については、LifeKeeper の ERS リソースタイプ を参照してください。

# 6.16.6.2. ABAP SCS (ASCS)

ABAP のみの設定は、次のインストールで構成されます。

- ABAP セントラルサービスインスタンス (ASCS インスタンス)
- ASCS インスタンス用エンキューレプリケーションサーバインスタンス (ERS インスタンス) (オプション)
- データベースインスタンス (DB インスタンス)
- プライマリアプリケーションサーバインスタンス (PAS)
- 追加のアプリケーションサーバインスタンス (AAS) - 追加のアプリケーションサーバインスタンス (AAS) を、プライマリアプリケーションサーバインスタンス (PAS) ホストとは別のホストにインストールすることを推奨します。

## SAP ABAP のみの (ASCS) システム向けスイッチオーバクラスタ




上の例において、ASCS は、データベースとは別のリソース階層にあります。同一サーバ上でデータベースとしてアクティブでも、別々にフェイルオーバできます。

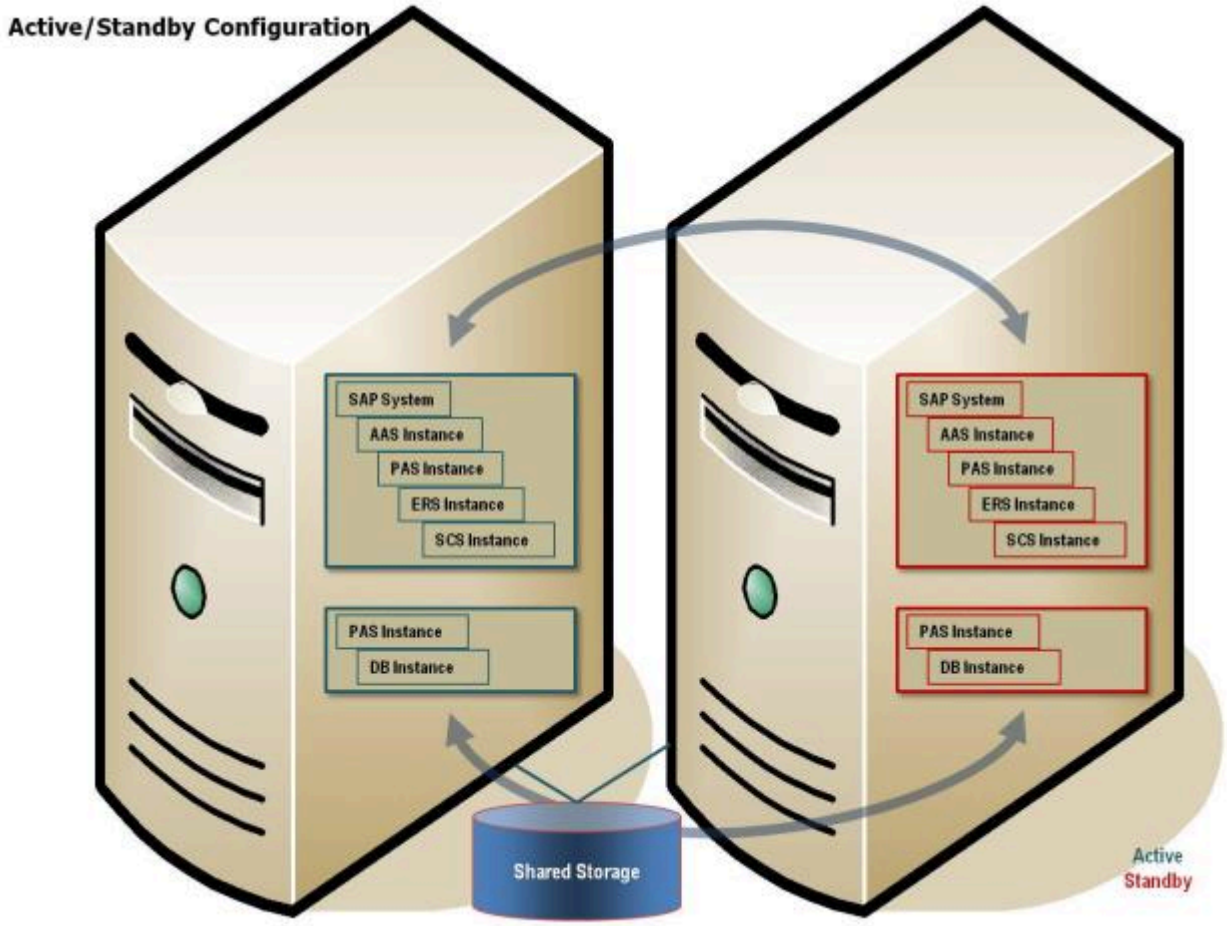
注記: SPS-L v9.4.0以降で作成されたERSリソースは、独立した独自の階層で動作します。詳細については、LifeKeeperのERSリソースタイプ を参照してください。

# 6.16.6.3. Java のみの構成 (SCS)

Java のみの構成は、次のインストールで構成されます。

- Java 用セントラルサービスインスタンス (SCS インスタンス)
- SCS インスタンス用エンキューレプリケーションサーバインスタンス (ERS インスタンス) (オプション)
- データベースインスタンス (DB インスタンス)
- プライマリアプリケーションサーバインスタンス (PAS)
- 追加のアプリケーションサーバインスタンス (AAS) - 追加のアプリケーションサーバインスタンス (AAS) を、プライマリアプリケーションサーバインスタンス (PAS) ホストとは別のホストにインストールすることを推奨します。

## Java のみのシステム (SCS) 向けスイッチオーバクラスタ




上の例において、SCS は、データベースとは別のリソース階層にあります。同一サーバ上でデータベースとしてアクティブでも、別々にフェイルオーバできます。

注記: SPS-L v9.4.0以降で作成された ERS リソースは、独立した独自の階層で動作します。詳細については、 LifeKeeper の ERS リソースタイプ を参照してください。

# 6.16.6.4. ディレクトリ構造

データベースのディレクトリ構造は、SAP システムと合わせて使用する各データベース管理システムによって異なります。使用するデータベースのディレクトリ構造の詳細については、データベース管理システム固有の SAP Installation Guide を参照してください。データベース向け LifeKeeper Recovery Kit で保護するには、すべてのデータベースファイルを共有ディスク上に配置する必要があります。データベースの保護に関する詳細については、データベースごとの Recovery Kit ドキュメンテーション を参照してください。

このセクションで説明する SAP ディレクトリの図解については、下のディレクトリ構造図 を参照してください。

インストール中に作成されるディレクトリの種類は、次の通りです。

**物理的に共有されたディレクトリ** (グローバルホスト上にあり、NFS で共有される)

> */<sapmnt>/<SAPSID>* - 1 つの SAP システム用のソフトウェアとデータ (同一の SAP システムに属するすべてのホストにマウントする必要がある)
>
> */usr/sap/trans* - グローバルトランスポートディレクトリ (エクスポートポイントを持つ必要がある)

以下のローカルディレクトリを持つ */usr/sap* などのノードにバインディングされている **論理的に共有されたディレクトリ** (ローカルホスト上にあり、グローバルホストへのシンボリックリンクを持つ)

> */usr/sap/<SAPSID>*
>
> */usr/sap/<SAPSID>/SYS*
>
> */usr/sap/hostctrl*

以下のような SAP インスタンスを含む **ローカルディレクトリ** (ローカルホスト上にあり、共有されている)

> */usr/sap/<**SAPSID**>/DVEBMGS<**No.**>* — プライマリアプリケーションサーバインスタンスのディレクトリ
>
> */usr/sap/<**SAPSID**>/D<**No.**>* — プライマリアプリケーションサーバインスタンスのディレクトリ
>
> */usr/sap/<**SAPSID**>/ASCS<**No.**>* — ABAP セントラルサービスインスタンス (ASCS) のディレクトリ
>
> */usr/sap/<**SAPSID**>/SCS<**No.**>* — Java セントラルサービスインスタンス (SCS) のディレクトリ
>
> */usr/sap/<**SAPSID**>/ERS<**No.**>* — ASCS および SCS 向けエンキューレプリケーションサーバインスタンス (ERS) のディレクトリ

SAP ディレクトリ */sapmnt/<**SAPSID**>* および */usr/sap/trans* は、NFS からマウントされます。ただし、SAP インスタンスディレクトリ (*/usr/sap/<**SAPSID**>/<**INSTTYPE**><**No.**>*) は、常に、現在インスタンスを実行しているクラスタノード上にマウントする必要があります。このようなディレクトリを NFS でマウントしないでください。必要なディレクトリ構造は、選択された構成によって変わります。必要なディレクトリ構造を決める問題はいくつかあります。

## NFS マウントポイントと inode

LifeKeeper は、inode を使用して NFS 共有情報を管理するため、各 NFS 共有は固有の inode を持つ必要があります。すべてのファイルシステムのルートディレクトリは同じ inode を持つため、NFS 共有を LifeKeeper で保護するためには、ルートから少なくとも 1つ下のディレクトリ階層にある必要があります。例えば、上記の情報に留意して、 */usr/sap/trans* ディレクトリを SAP サーバ上で NFS 共有する場合、共有ストレージデバイス上に */trans* ディレクトリが作成されます。これには、共有ストレージデバイスを */usr/sap* としてマウントする必要があります。しかしながら、この配置で要求されるように、すべてのファイルを共有ストレージ上の */usr/sap* 配下に置くことは必ずしも望ましくありません。この問題を回避するため、NFS 共有されているディレクトリを含むすべての共有ファイルシステムをマウントするための */exports* ディレクトリツリーを作成し、その後で SAP ディレクトリと */exports* ディレクトリ間にソフトリンクを作成するか、この NFS 共有ディレクトリをローカルに NFS マウントすることを推奨します。(注記: ここで */exports* と呼ぶディレクトリの名前は、ユーザの好みによって変更できますが、説明を分かりやすくするため、このドキュメンテーションではこのディレクトリを */exports* と呼びます。)例えば、この SAP プライマリサーバの例に関する次のディレクトリとリンク / マウントは、以下のようになります。

| ***/usr/sap/trans* 共有の場合** | |
|---|---|
| ディレクトリ | **備考** |
| */trans* | 共有ファイルシステム上に作成され、NFS で共有 |
| */exports/usr/sap* | / (共有ファイルシステム上) にマウント |
| */usr/sap/trans* | /exports/usr/sap/trans にソフトリンク |

同様に、 *<sapmnt>/<SAPSID>* 共有に対するディレクトリとリンクは、次のようになります。

| ***<sapmnt>/<SAPSID>* 共有の場合** | |
|---|---|
| ディレクトリ | **備考** |
| */<SAPSID>* | 共有ファイルシステム上に作成され、NFS で共有 |
| */exports/sapmnt* | / (共有ファイルシステム上) にマウント |
| *<sapmnt>/<SAPSID>* | *<virtual SAP server>:/exports/sapmnt/<SAPSID>* に *NFS* マウント |

すべてのディレクトリ構造とリンクを作成するための詳細な手順を、本ドキュメンテーションの設定手順で後述します。inode 競合の詳細と、NFSv4 の新機能を使用するための情報については、NFS Server Recovery Kit ドキュメンテーション を参照してください。

# ローカル NFS マウント

LifeKeeper 環境の SAP に推奨されるディレクトリ構造では、1 つまたは複数の SAP システムディレクトリについてローカルにマウントされた NFS 共有が要求されます。いずれかのローカルにマウントされた NFS 共有の NFS エクスポートポイントが使用不可になると、エクスポートポイントが使用可能になるのを待機する間、システムがハングすることがあります。システムの再起動を含め、多くのシステム操作が正しく動作しません。SAP クラスタの NFS サーバを LifeKeeper で保護する必要があり、ローカルマウントポイントが存在する間は手動でサービス休止状態にしてはならないことに注意してください。

不注意で NFS サーバを停止してクラスタをハングさせないように、NFS の考慮事項 トピックに記載した推奨事項に従ってください。

NFS 共有にアクセスできない場合、アンマウントが失敗することがあります。LifeKeeper は、ファイルシステムのアンマウントを複数回試行します。複数回試行することにより、通常、最終的にリソースを Out of service にすることに成功します。ただし、リソースを Out of service にする際に遅延が発生します。再試行を回避するには、マウントオプション「nfsvers=3, proto=udp 」を使用します。

> ! フェイルオーバとリカバリで重要になるので、udp の使用方法に注意してください。

# NFS マウントと su

LifeKeeper は、多くのデータベース作業および SAP 作業を、`su - <admin name> -c <command>` コマンド構文を使用してデータベース処理および SAP 処理を実行することで実現します。このような方法で呼び出された場合、su コマンドにより、管理者のホームディレクトリにあるログインスクリプトが実行されます。これらのログインスクリプトは、環境変数をさまざまな SAP パスに設定します。その一部は、NFS マウントされた共有に存在する可能性があります。これらの NFS 共有が何らかの理由で利用できない場合、su 呼び出しは、NFS 共有が利用可能になるまで待機してハングします。

ハングスクリプトは LifeKeeper の正常実行を妨げる可能性があるため、この潜在的な問題に対処するよう、サーバを設定することが望まれます。SAP リソースの削除、リストア、監視処理を行う LifeKeeper スクリプトは、これらのスクリプトがいつまでもハングし続けることを防止するタイマを内蔵しています。そのため、SAP Application Recovery Kit では、NFS ハングに対処するための設定作業は必要ありません。

それでも、利用不可能な NFS 共有によって影響を受ける手動の処理は数多く存在することに注意してください。手動で LifeKeeper 処理を実行する前に、必ず、すべての NFS 共有が利用可能であることを確認する必要があります。

> ✻ アクセスできない NFS 共有の処理遅延を回避するには、NFS の考慮事項 に記載されている推奨事項に従ってください。

# < *INST* > ディレクトリの場所

*/usr/sap/<SAPSID>* パスは NFS 共有ではないため、ファイルシステムのルートディレクトリにマウントできます。 */usr/sap/<SAPSID>* パスには、サーバ上で実行可能な SAP インスタンスごとに、 *SYS* サブディレクトリと < *INST* > サブディレクトリが含まれています。構成によっては、< *INST* > ディレクトリは 1 つしかないため、共有ファイルシステム上の */usr/sap/<SAPSID>* 配下に置くことができます。ただし、他の構成では、バックアップサーバもローカル AS インスタンスを含む可能性があり、その < *INST* > ディレクトリは常に利用できるわけではないため、共有ファイルシステム上に置くことはできません。この問題を解決するため、特定の構成では、PAS、ASCS、または SCS の */usr/sap/<SAPSID>/<INST>* 、*/usr/sap/<SAPSID>/<ASCS-INST>* 、または */usr/sap/<SAPSID>/<SCS-INST>* ディレクトリを */usr/sap/<SAPSID>* ではなく共有ファイルシステムにマウントし、AS の */usr/sap/<SAPSID>/SYS* および */usr/sap/<SAPSID>/<AS-INST>* をローカルサーバ上に置いてください。

> ERS ファイルシステムは、ERSv1 の場合はローカルにマウントし、ERSv2 の場合は共有ストレージにマウントする必要があります。

例えば、ABAP+Java 構成では、次のディレクトリとマウントポイントを作成する必要があります。

| ディレクトリ | 注意 |
|---|---|
| */usr/sap/<SAPSID>/DVEBMGS<**Instance #**>* | /(共有ファイルシステム上) にマウント |
| */usr/sap/<SAPSID>/SCS<**Instance #**>* | /(共有ファイルシステム上) にマウント |
| */usr/sap/<SAPSID>/ERS<**Instance #**> (SCS インスタンス用)* | すべてのクラスターノード上にローカルでマウントするか、NAS 共有からマウントする必要がある ( 共有ファイルシステム上にマウントしてはならない ) |
| */usr/sap/<SAPSID>/ASCS<**Instance #**>* | /(共有ファイルシステム上) にマウント |
| */usr/sap/<SAPSID>/ERS<**Instance #**>*<br><br>*(ASCS インスタンス用)* | すべてのクラスターノード上にローカルでマウントするか、NAS 共有からマウントする必要がある ( 共有ファイルシステム上にマウントしてはならない ) |
| */usr/sap/<SAPSID>/AS<**Instance #**>* | バックアップサーバ上の AS に作成 |

**注記:** エンキューレプリケーションサーバ (ERS) リソースは、クラスタのプライマリノードで In Service (ISP) になります。ただし、ERS のアーキテクチャと機能は、インスタンスの実際のプロセスがバックアップノード上で実行される必要があります。これによって、スタンバイサーバはプライマリサーバおよびプライマリエンキューサーバインスタンスのロックテーブル情報の完全なコピーを保持することができます。エンキューサーバを実行するプライマリサーバに障害が発生した場合、ERS プロセスが現在実行されているバックアップサーバ上の SIOS Protection Suite によって再起動されます。ERS 上に保存されてい

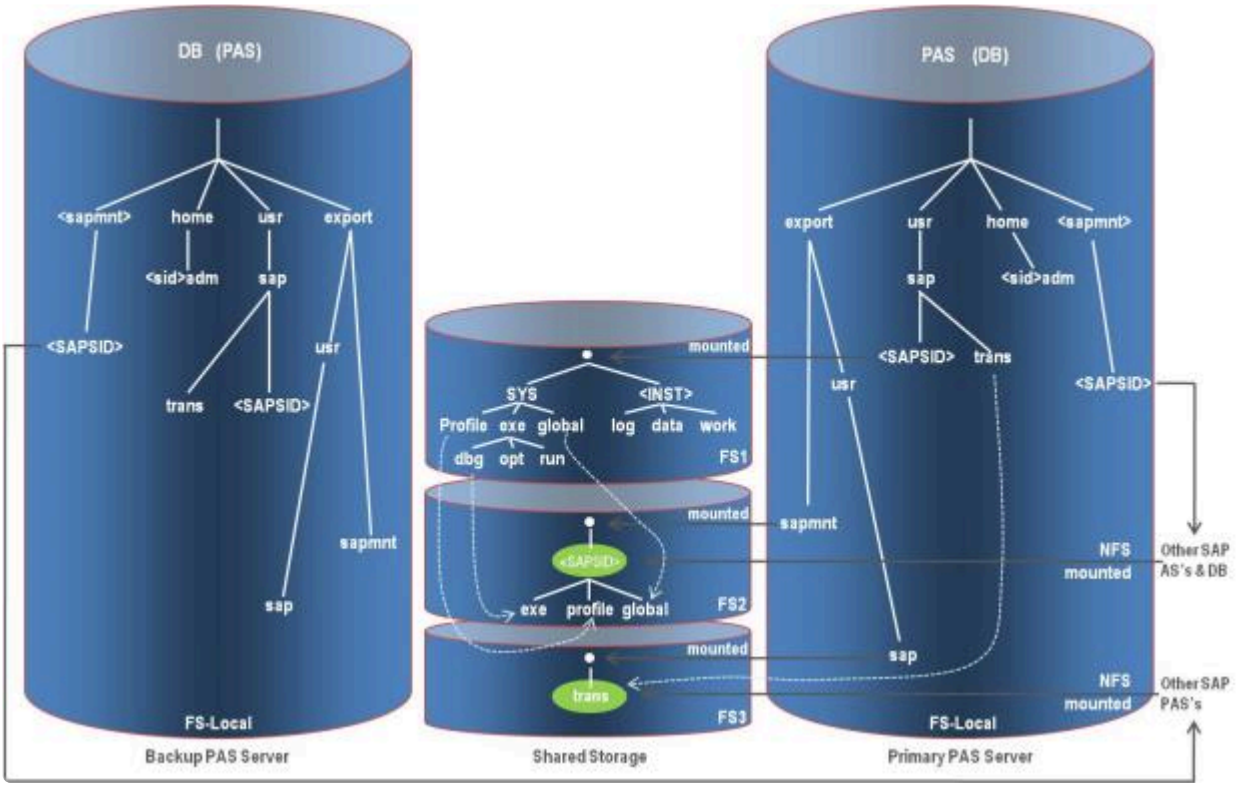
るロックテーブル (レプリケーションテーブル) が復旧中のエンキューサーバプロセスに転送され、それを基に新しいロックテーブルが作成されます。このプロセスが完了すると、アクティブなレプリケーションサーバは非アクティブになります (このサーバはエンキューサーバとの接続を閉じ、レプリケーションテーブルを削除します)。これまで非アクティブだった新しい現在のバックアップノード (元のプライマリ) 上で、SIOS Protection Suite によって ERS プロセスが再起動されます。ERS プロセスがアクティブになると、エンキューサーバに接続され、レプリケーションテーブルが作成されます。ERS プロセスと SAP アーキテクチャの機能の詳細については、http://help.sap.com にアクセスして、 **エンキューレプリケーションサービス** で検索してください。

レプリケーションサーバはバックアップノードで常にアクティブであるため、ファイルシステムはプライマリノードでアクティブになり、レプリケーションサーバプロセスはバックアップノードでアクティブになります。このため、レプリケーションサーバはSIOS Protection Suiteで保護されたファイルシステムには常駐できません。したがって、ERSが使用するファイルシステムは、すべてのクラスタノードでローカルにマウントするか、NAS共有からマウントする必要があります。

# ディレクトリ構造図

ABAP のみの環境を LifeKeeper で保護するために必要なディレクトリ構造を、下図に示します。図内で使用されている略語の説明については、略語と定義 セクションを参照してください。

## ディレクトリ構造の例



凡例

| | |
|---|---|
| | soft link |
| | mount point |
| | NFS shared file system |
| PAS | Primary PAS |
| (PAS) | Backup PAS |
| AS | Application Server |
| DB | Primary DB Server |
| (DB) | Backup DB Server |
| FSn | File System on shared storage |
| FS-local | File System on local disk |

## ディレクトリ構造のオプション

このドキュメンテーションに示す設定手順は、上に示したディレクトリ構造および図に基づきます。これは、SIOS Technology Corp によってテストおよび認証済みの、推奨ディレクトリ構造です。

すべてがテストされているわけではありませんが、SAP Recovery Kit が正しく動作する、その他のディレクトリ構造のバリエーションも存在します。ディレクトリ構造のバリエーションの設定については、以下の指針に従う必要があります。

- */usr/sap/trans* ディレクトリは、ネットワーク上でアクセス可能な任意のサーバ上にホストでき、PAS サーバである必要はない。 */usr/sap/trans* ディレクトリに PAS からリモートでアクセスする場合、このディレクトリへのアクセスがミッションクリティカルであるかどうか判断する必要がある。ミッションクリティカルであれば、LifeKeeper で保護する。これには、ディレクトリが共有または複製ファイルシステムにあり、NFS Server Recovery Kit によって保護されていることが必要である。NFS を使用せずに */usr/sap/trans* ディレクトリをすべての SAP インスタンスから利用可能にする方法があれば、それを使用してもよい。

- */usr/sap/trans* ディレクトリは、PAS サーバに配置されているかどうかに関わらず、NFS 共有されている必要はない。

- */usr/sap/trans* ディレクトリは、NFS 共有または LifeKeeper で保護されていない場合、共有ファイルシステム上に置く必要はない。

- 図示した NFS ファイルシステムをエクスポートするために使用されるディレクトリ構造とパス名は、単なる例である。パス */exports/usr/sap* は、 */exports/sap* または単に */sap* でよい。

- */usr/sap/<SAPSID>/<INST>* パスは、共有ファイルシステム上のいずれかのレベルにある必要がある（ERSv1 インスタンスの場合を除く）。このパスのどの部分がファイルシステムのマウントポイントであるかは関係ない。 */usr*、*/usr/sap*、*/usr/sap/<SAPSID>* 、 */usr/sap/<SAPSID>/<INST>* のいずれも可能である。

- */sapmnt/<SAPSID>* パスは、共有ファイルシステム上のいずれかのレベルにある必要がある。設定図では、このパスは NFS マウントとして図示されている。これは SAP の要件であるが、LifeKeeper の要件ではない。

# 6.16.6.5. 仮想サーバ名

SAP アプリケーションサーバと SAP クライアントは、PAS インスタンスが実行中のサーバの名前を使用して、SAP プライマリアプリケーションサーバ (PAS) と通信します。同様に、SAP PAS は、データベース (DB) と、DB サーバの名前を使用して通信します。HAクラスタでは、PAS は、常にプライマリサーバまたはバックアップサーバのいずれかで実行されます。どちらのサーバでアクティブになっているかに関わらず、その他のサーバとクライアントが PAS にシームレスに接続できるようにするため、PAS とのすべての通信に仮想サーバ名を使用します。また、仮想サーバ名は、PAS が実行されているどちらのサーバでもアクティブにできる切り替え可能 IP アドレスにもマッピングされます。

切り替え可能 IP アドレスは、LifeKeeper によって、IP Recovery Kit を使用して作成および処理されます。仮想サーバ名は、仮想サーバ名 / 切り替え可能 IP アドレスのマッピングを DNS に追加するか、すべてのサーバおよびクライアントのホストファイルに追加することにより、手動で設定されます。この仕組みの詳細については、IP ローカルリカバリ トピックを参照してください。

さらに、仮想サーバ名を物理サーバ名の代わりに使用するよう、SAP 設定ファイルを変更する必要があります。これについては、インストール セクションで詳しく説明されており、その他にも LifeKeeper で SAP を設定するための手順が記載されています。

**注記:** SAP アプリケーションサーバ階層と NFS サーバ階層では、別々の切り替え可能 IP アドレスを使用することを推奨します。これにより、NFS クライアントで使用される IP アドレスと SAP クライアントで使用される IP との分離を維持できます。

# 6.16.6.6. SAP ヘルスモニタリング

LifeKeeper は、プライマリアプリケーションサーバ (PAS) インスタンスの状態を監視し、SAP が正しく機能していないと判断した場合、リカバリ処理を開始します。

そのステータスは、GUI の [Properties Panel] と CLI を通じて、 **グレー** (未知/非アクティブ/オフライン)、 **赤** (障害発生)、 **黄** (問題)、または **緑** (良好) として、ユーザに返されます。

インスタンスのステータスが **グレー** の場合、状態は未知であるか、情報が利用できません。

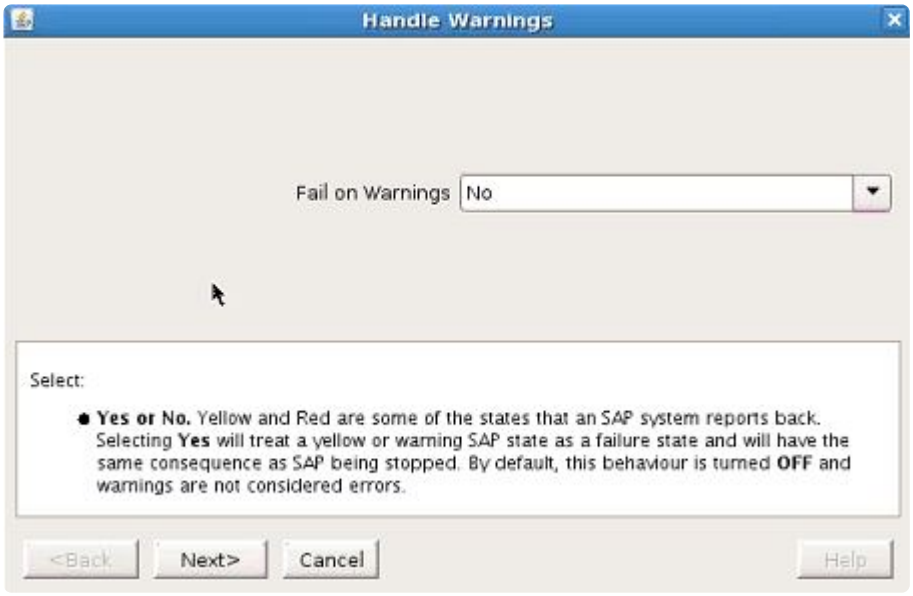
インスタンスのステータスが **赤** の場合、リソースは障害状態とみなされ、LifeKeeper は適切なリカバリ処理を開始します。

インスタンスのステータスが **黄** , の場合、定義されたインスタンスについて、SAP プロセスに問題がある可能性を示します。黄色のステータスのデフォルト動作は、リカバリを開始することなく監視を続行することです。

デフォルト動作は、このオプションを GUI リソースメニューで設定することで変更できます。

1. インスタンス を右クリックします。
2. **[Handle Warnings]** を選択します。

3. 以下の画面が表示され、 **[Fail on Warnings]** を選択できます。




**[Yes]** を選択すると、 **黄色の警告** はエラーとして扱われて、リカバリが開始されます。

**注記:** 黄は過渡的な状態であり、ほとんどの場合は障害を表すものではないため、この設定はデフォルト選択の **[No]** のままにすることを強くお勧めします。

# 6.16.6.7. SAP ライセンス

HAクラスタ環境では、SAP は、プライマリサーバとバックアップサーバの両方で実行できるよう設定されます。SAP ライセンススキームはハードウェア依存であるため、SAP が実行される各サーバについて個別のライセンスが必要になります。そのため、プライマリサーバとバックアップサーバの両方について、SAP ライセンスを取得しインストールする必要があります。

# 6.16.6.8. 自動スイッチバック

アクティブ / アクティブ構成において、SAP プライマリアプリケーションサーバインスタンス (PAS)、ABAP SAP セントラルサービスインスタンス (ASCS)、または SAP セントラルサービスインスタンス (SCS) の階層とデータベース (DB) 階層は独立しており、通常動作中は異なるサーバ上で In Service の状態になります。しかし、いずれかのサーバがメンテナンス中の場合など、両方の階層が同一のサーバ上で In Service になることもあります。いずれかのサーバ上で両方のリソース階層が In Service の場合に、両方のサーバがダウンした後、サーバが再起動するとき、SAP 階層の in-service 操作のタイムアウト前に、データベース階層が In Service になることが重要です。LifeKeeper は、起動中に階層の in-service 処理を段階的に行うため、LifeKeeper が SAP を先に起動した場合、データベースの in-service 処理は SAP の in-service 処理の終了を待機し、SAP の in-service 処理はデータベースが利用可能になるのを待機します。しかし、DB のリストア処理は PAS、ASCS、または SCS リストアが完了した後に初めて開始できるため、データベースが利用可能になることはありません。このデッドロック状態は、PAS、ASCS、または SCS のリストア処理がタイムアウトするまで持続します。( **注記** : SAP は、10 分後にタイムアウトして失敗します。)

デッドロックシナリオを回避するため、この設定では、両方のリソース階層に Automatic Switchback を設定しておくことが重要です。これで、LifeKeeper の起動中に、最も高い優先順位を持つサーバ上 (この場合、2 つの異なるサーバ) で強制的に各階層をリストアすることになります。異なるサーバ上での LifeKeeper の restore 処理は並列に実行できるため、デッドロック状態を回避できます。

# 6.16.6.9. その他の注記

次の項目については、HAクラスタ環境において特別な設定手順が必要になります。以下の項目それぞれの設定要件に関する情報については、 *SAP Web Application Server in Switchover Environments* ドキュメントを参照してください。

- ログイングループ
- SAP スプーラ
- バッチジョブ
- SAP ルータ
- SAP システムアップグレード

# 6.16.7. インストール

## 設定 / インストール

LifeKeeper を使用して SAP リソース階層を作成する前に、以下の作業を下記に **推奨する順序で** 実施してください。以下に記載されていない追加の非 HA 固有の設定作業を行う必要があることに注意してください。詳細については、SAP インストールガイドを参照してください。

以下の作業では、「**SAP プライマリサーバ**」および「**SAP バックアップサーバ**」を参照しています。 **SAP プライマリサーバ** は、通常動作時にセントラルサービスが実行されるサーバです。 **SAP バックアップサーバ** は、SAP プライマリサーバに障害が発生したときにセントラルサービスが実行されるサーバです。

必ずしも必須ではありませんが、以下の操作手順には、使用する前にすべての共有ファイルシステムを LifeKeeper で保護する推奨手順が含まれています。共有ファイルシステムは、LifeKeeper で保護する前は両方のサーバからアクセス可能であるため、データが破壊される可能性があります。LifeKeeper でファイルシステムを保護することで、データへのアクセスを 1 つのサーバに留めます。

## SAP をインストールする前に

以下のトピックの作業は、SAP ソフトウェアをインストールする前に行う必要があります。記載されている順序に従って、作業を行ってください。また、NetWeaver 環境でのインストールを予定している場合、SAP ドキュメントの *SAP Web Application Server in Switchover Environments* を参照してください。

設定の計画

## SAP ソフトウェアのインストール

以下の作業は、SAP ソフトウェアを高可用性環境にインストールするために必要です。記載されている順序に従って、作業を行ってください。詳細については、各作業をクリックしてください。SAP のインストール手順の詳細については、関連する SAP インストールガイドを参照してください。

### プライマリサーバのインストール

コアサービス、ABAP セントラルサービス (ASCS)、および Java セントラルサービス (SCS) のインストール

データベースのインストール

プライマリアプリケーションサーバインスタンスのインストール

追加のアプリケーションサーバインスタンスのインストール

### バックアップサーバのインストール

バックアップサーバへのインストール

# LifeKeeper のインストール

LifeKeeper のインストール

ファイルシステムとディレクトリ構造の作成

共有ディスクおよび LifeKeeper へのデータ移動

SAP Recovery Kit の旧バージョンからのアップグレード

# LifeKeeper での SAP の設定

## リソース設定作業

以下の作業では、LifeKeeper GUI の **[Edit]** ニューからオプションを選択することによって、Recovery Kit を設定する方法を説明します。それぞれの設定作業は、ツールバーから選択することもできます。または、状況表示ウィンドウの **リソース階層ツリー** (左側のペイン) のグローバルリソースを右クリックして、**[Edit]** メニューと同じドロップダウンメニューの選択項目を表示することもできます。言うまでもなく、これは階層がすでに存在している場合にだけ可能な方法です。

または、状況表示ウィンドウのリソース階層表 (右側のペイン) のリソースインスタンスを右クリックして、サーバおよび特定リソースの状況に応じて、リソース階層の作成を除くすべての設定作業を実行することもできます。

IP リソース

SAP リソース階層の作成

リソース階層の削除

リソース階層の拡張

リソース階層の拡張解除

共通の Recovery Kit 作業

コマンドラインによる SAP の設定

SAPインスタンスに対してSAP SIOS HA Cluster Connectorを有効にする方法については、 SAP SIOS HA Cluster Connector (SSHCC)のアクティブ化 を参照してください。

LifeKeeperでERSインスタンスを適切に管理するには、ERSプロファイルはERSプロセスの開始に、Restart_Programではなく Start_Programパラメーターを使用する必要があります。ERSインスタンスプロファイルでこのパラメーターを変更する方法の詳細については、ASCS + ERS Restart_Programパラメーター を参照してください。

# SAP リソース階層のテスト

SAP ソフトウェアに対して LifeKeeper 保護を構築した後、その SAP 階層を徹底的にテストする必要があります。記載されている順序に従って、作業を行ってください。

テストの準備

テストの実行

# 6.16.7.1. SAP構成を計画する

1. 使用する設定 を決定してください。必要な作業は、設定によって異なります。

2. SAP システム全体の */usr/sap/trans* ディレクトリを SAP プライマリアプリケーションサーバ上にホストするか、ファイルサーバ上にホストするかを決定してください。NFS で共有され完全にアクセス可能である限り、どちらにホストしても構いません。SAP プライマリアプリケーションサーバ上にホストして共有ファイルシステム上に置く場合、LifeKeeper で保護し、SAP 階層内に含める必要があります。

3. *SAP Installation Guide* に記載されている SAP およびデータベースのストレージ要件を考慮してください。大部分の SAP ファイルは、共有ストレージ上にインストールする必要があります。どのデータベースファイルが共有ストレージにインストールされ、どのデータベースファイルがローカルにインストールされるかについては、データベース固有リカバリキットの SPS for Linux テクニカルドキュメンテーション を参照してください。SAP 環境では、SAP がデータベースバイナリにローカルアクセスを必要とするため、これらはローカルにインストールする必要があることに注意してください。以上の要件を満たして、使用する共有ストレージに最適な使用方法を決定してください。

   また、共有ストレージリソースが LifeKeeper の保護下にある場合、そのリソースに一度にアクセス可能なサーバは 1 つだけになることに注意してください。共有デバイスがディスクアレイである場合、LUN 全体が保護されます。共有デバイスがディスクである場合、ディスク全体が保護されます。そのため、単一ボリュームに配置されているすべてのファイルシステムは、LifeKeeper によってまとめて制御されます。これは、データベース用に 1 つ、SAP 用に 1 つの、少なくとも 2 つの論理ボリューム (LUN) を用意する必要があることを意味します。

4. フェイルオーバ時にシステムを識別するため、仮想ホスト名が必要になります。使用する仮想ホスト名 1 つにつき 1 つの新しい IP アドレスが 必要です。この仮想ホスト名が DNS (ドメインネームシステム) 設定で正しく解決できることを確認したら、以下の手順に進みます。

   a. 以下のコマンドを使用して、新しい仮想 IP アドレスを作成してください。

   `ip addr add {IPADDRESS}/{NETMASK} dev eth0` (使用設定に合わせて、適切なネットマスクを使用してください)

   **注記:** これらの新しい仮想IPアドレスを確認するには、`ip addr show` コマンドを使用します。

   b. ERSv2 を使用する場合は、ERS インスタンス用に個別の仮想 IP も必要になります。

   切り替え可能な IP アドレスと仮想サーバー名を関連付けるために、/etc/hosts を編集し、新しい仮想 IP アドレスと関連する仮想ホスト名を追加します。

   **注記:** プライマリアプリケーションサーバとデータベースが常に同一のサーバ上で実行され、その通信が常にローカルである場合、この手順は必須ではありません。しかし、いつか異なるサーバで実行する場合に備えて、プライマリアプリケーションサーバとデータベースに対し、別々の切り替え可能 IP アドレスおよび仮想サーバ名を用意することを推奨します。

5. 両方マシンでキャッシングデーモンを停止してください。

```
rcnscd stop
```

6. ソフトウェアをマウントしてください。

```
mount //{path of software} (no password needed)
```

7. X セッション (ssh -X または VNC セッション、Microsoft Windows ユーザの場合、 Hummingbird Exceed X Windows を使用可能) を実行してください。

   **注記:** `sapinst` が実行されている場合、ディレクトリは */tmp* 配下に抽出されます。

8. SAP（具体的にはASCSまたはSCS）をインストールするときは、SAPINST_USE_HOSTNAME={仮想ホスト名} オプションを指定する必要があります。 これは、ERSv2を使用している場合はERSインスタンスにも必要ですが、ERSv1を使用している場合は必要ありません。

   **注記:** `sapinst SAPINST_USE_HOSTNAME = {仮想ホスト名}` を指定します。ここでは、 **仮想ホスト名** は、ノード間でフロートする仮想IPに解決されるホスト名です。

# 重要事項

LifeKeeper SAP Recovery Kit は、インストールされている SAP Host Agent に依存します。このソフトウェアがインストールされていない場合、LifeKeeper SAP Kit はインストールされません。SAP Netweaver Version 7.3 ではこの Host Agent が提供されていますが、それより前のバージョンでは SAP からダウンロードする必要があります。使用する特定のバージョンについては、SAP のヘルプを参照することをお勧めします。また、詳細なドキュメンテーションをヘルプフォーラム (help.sap.com) で入手できます。

- RPM または SAR フォーマットの saphostexec モジュールを、SAP からダウンロードできます。
- モジュールが正しくインストールされているか確認するには、いくつかのモジュール ( *saposcol*、*saphostexec*、*saphostctrl* ) を検索します。SAP がインストールされている場合、通常これらのモジュールが見つかります (一般的に */usr/sap* ディレクトリ)。

# 6.16.7.2. コアサービスのインストール

ソフトウェアをインストールする前に、すべてのサーバ上で日付と時刻が同期していることを確認してください。これは、LifeKeeper と SAP の両方にとって重要です。

コアサービス、ABAP セントラルサービス (ASCS) および Java セントラルサービス (SCS) は、単一障害点 (SPOF) であり、LifeKeeper で保護する必要があります。適切な SAP インストールガイドを使用して、コアサービスを SAP プライマリサーバ上にインストールしてください。

## インストール上の注記

- 設定の計画 トピックで作成した必要な仮想ホスト名を使用するには、SAPinst 開始前に SAPinst プロパティの `SAPINST_USE_HOSTNAME` を設定し、必要な仮想ホスト名を指定します。( **注記** :`SAPINST_USE_HOSTNAME` 仮想 IP アドレスは、後で LifeKeeper で SAP リソースを作成するときに使用するため、記録しておいてください。)

  次のコマンドを実行: `./sapinst SAPINST_USE_HOSTNAME={hostname}`

- 7 つの段階で、 **コアサービス** を作成して、起動する必要があります。`jdbcconnect.jar` でパーミッションエラーが発生する場合、 */sapmnt/STC/exe/uc/linuxx86_64* に移動し、ディレクトリとファイル `jdbcconnect.jar` を書き込み可能にしてください (chmod 777 ---)。

- 正常完了のメッセージが表示され、インストールが完了します。

# 6.16.7.3. データベースのインストール

1. dba と oinstall のグループ ID をメモしてください。これは、バックアップマシンで必要になります。

2. ソフトウェアのディレクトリに移動し、次のコマンドを実行してください。

```
./sapinst SAPINST_USE_HOSTNAME={database connectivity ip address}
```

3. 適切な SAP インストールガイドを使用して、SAPinst を実行し、データベースインスタンスをインストールしてください。

## インストール上の注記

- SIOS は、orarun パッケージがデータベースインスタンスのインストール前に既にインストールされている場合、orarun パッケージを削除することを推奨します ( **SAP ノート 1257556** を参照)。

- SAPinst ウィンドウのデータベースインストールオプションは、Oracle を除き、データベースソフトウェアが既にインストールされていることを前提としています。Oracle データベースの場合、SAPinst はインストールを停止し、データベースソフトウェアのインストールを要求します。

- <DBSID> は、データベースインスタンスを識別します。SAPinstは、データベースインスタンスをインストールスするときに <DBSID> の入力を要求します。<DBSID> は、<SAPSID> と同じにすることができます。

- SAP グローバルホスト以外のホストにデータベースをインストールする場合、SAP グローバルホストからグローバルディレクトリをマウントする必要があります。

- Listener が起動される時点で問題が発生した場合、以下のコマンドを使用して終了させてください。

```
(ps -ef | grep lsnrctl)
```

- SAPR3 と SAPR3DB のユーザ ID のパスワードをリセットするには、以下のコマンドを使用してください。

```
brtools
```

データベースのインストールが完了したら、下のダイアログを閉じ、SAP のインストール、すなわちアプリケーションサービスのインストール に進んでください。

# 6.16.7.4. プライマリアプリケーションサーバインスタンスのインストール

1. プライマリアプリケーションサーバインスタンスをインストールするには、先述したディレクトリから sapinst を再実行します。

   ./sapinst SAPINST_USE_HOSTNAME= < vip >

2. 要求された場合、**[Primary Application Server Instance]** を選択し、適切な SAP インストールガイドを使用してインストールを続行します。

## インストール上の注記

- プライマリアプリケーションサーバインスタンスは、既に単一障害点 (SPOF) ではなくなっているため、クラスタの一部にする必要はありません。この時点で、SPOF は、クラスタによって保護されているセントラルサービスインスタンス (SCS インスタンスおよび ASCS インスタンス) 内にあります。

- プライマリアプリケーションサーバインスタンスのディレクトリの名前は、DVEBMGS<*No*> です (<*No*> はインスタンス番号)。

- **[OK]** メッセージを受信すると、アプリケーションサーバのインストールが完了します。

- レプリケートされたエンキューを 7.1 上にインストールする場合、`sapinst` をそのまま実行します。

# 6.16.7.5. 追加のアプリケーションサーバインスタンスのインストール

冗長性を実現するため、追加のアプリケーションサーバインスタンスをインストールすることを推奨します。アプリケーションサーバインスタンスは SPOF ではないため、クラスタに含める必要はありません。

各追加アプリケーションサーバインスタンス上で、次の作業を実行してください。

1. `SAPinst` を実行し、追加のアプリケーションサーバインスタンスをインストールしてください。

2. 要求された場合、 **[Additional Application Server Instance]** を選択し、適切な SAP インストールガイドを使用してインストールを続行します。

# 6.16.7.6. バックアップサーバへのインストール

バックアップサーバでは、プライマリサーバ上で実行した以下のインストール手順をもう一度実行します。

1. コアサービス、ABAP セントラルサービス (ASCS)、および Java セントラルサービス (SCS) のインストール
2. データベースのインストール
3. アプリケーションサービスのインストール

# 6.16.7.7. SPS のインストール

SPS SAP Recovery Kit は、LifeKeeper バージョン 7.3 と 7.4 でテストされています。バージョン 7.3 を使用している場合、受注時に提供されるパッチを適用する必要があります。

**プライマリ** サーバと **バックアップ** サーバの両方において、LifeKeeper ソフトウェアを次のリカバリキットを含めてインストールしてください。

- SAP
- 適切なデータベース (Oracle、SAP DB など)
- IP
- NFS
- NAS

1. 両方のマシンで **Oracle Listener** と **SAP** を停止してください。

   例えば、Oracle ユーザが *orastc* で、Oracle Listener が *LISTENER_STC* で、SAP ユーザが *stcadm* の場合、次のようになります。

   a. `su` コマンドでユーザ `orastc` に切り替え、コマンド `lsnrctl stop LISTENER_STC` を実行してください。

   b. `su` コマンドでユーザ `stcadm` に切り替え、コマンド `stopsap sap{No.}` を実行してください。

   c. root ユーザとして、SAP または Oracle ユーザプロセスが実行されていないことを確認します。もし実行されている場合、「`killall sapstartsrv`」と入力してください。このコマンドを実行した後もプロセスが存在する場合は、「ps ?ef」を実行して各プロセスを終了してください。

2. 両方のマシンで */etc/hosts* を開き、ホストと DNS のエントリが正しく指定されていることを確認してください。

3. 現在のインターフェースから IP アドレスを **停止** および **削除** してください。 **注記**: この手順は、IP アドレスを LifeKeeper IP Recovery Kit で保護する前に行う必要があります。

   ```
   ip addr delete {A/SCS VIRTUAL IP ADDRESS}/{NETMASK} dev eth0

   ip addr delete {ERSv2 VIRTUAL IP ADDRESS}/{NETMASK} dev eth0
   ```

4. ping などを使用して接続試行を行い、IP アドレスが削除されていることを確認してください。

5. SPS インストールガイド に記載の手順に従って、プライマリサーバとバックアップサーバの両方に

SPS をインストールしてください (DE、コア、DataKeeper、LVM、およびライセンス)。Recovery Kit を選択するよう求められたら、以下を選択してください。

SAP、適切なデータベース (Oracle)、IP、NFS、NAS

インストールスクリプトが一定のチェックを実行します。以下の例に示すように、環境が正しく設定されない場合、失敗することがあります。

```
SAP Services file /usr/sap/sapservices not found

SAP Installation is not valid; please check environment and retry

error: %pre(steeleye-lkSAP-7.3.1-1.noarch) scriptlet failed, exit
status 2

error: install: %pre scriptlet failed (2), skipping steeleye-
lkSAP-7.3.1-1
```

上記の例では、期待される SAP ファイル、/usr/sap/sapservices が欠落しています。インストールを続行する前に、環境が正しい状態になることはとても重要です。

リカバリキットのインストールおよびリソースを保護するためのサーバの設定に関する詳細は、Recovery Kit ドキュメンテーション を参照してください。

# 6.16.7.8. ファイルシステムとディレクトリ構造の作成

使用するデータベース管理システムによって多くの異なる設定が可能ですが、準拠すべき基本的なレイアウトを以下に示します。

- プライマリサーバとセカンダリサーバ間にコミュニケーションパスを確立する
- 仮想 IP リソースを *etc/hosts* に追加する
- インスタンスおよびデータベースホストの仮想IPリソースを作成する
- 共有ディスクを構築する
- SAP 用のファイルシステムを (共有ディスク上に) 作成する
- データベース用のファイルシステムを (共有ディスク上に) 作成する
- メイン SAP ファイルシステムをマウントする
- マウントポイントを作成する
- PAS、ASCS/SCS、およびERSディレクトリ（該当する場合）および追加のアプリケーションサーバーをマウントする

データベースのディレクトリ構造の詳細については、データベース管理システム固有の SAP インストールガイドを参照してください。データベース用の LifeKeeper Recovery Kit の保護下に置くためには、すべてのデータベースファイルを共有ディスク上に配置する必要があります。データベースを保護するための追加情報については、データベース固有の Recovery Kit ドキュメンテーション を参照してください。

以下の例は可能な設定の例にすぎませんが、これらの多くの設定例を理解し、設定規則に従うことは、ユーザのコンピューティング環境で実現可能なソリューションを明確にし、設定する上で役立ちます。

1. プライマリサーバの UI から、プライマリサーバとセカンダリサーバ間にコミュニケーションパスを確立してください。
2. 実際のプライマリおよびセカンダリ仮想 IP アドレスのエントリを */etc/hosts* に追加してください。
3. プライマリサーバ上で LifeKeeper にログインし、ホストとデータベースの仮想 IP リソースを作成してください (例: *ip-db10* と *ip-sap10*)。
4. 2 台のマシン間に共有ディスクを構築してください。

   **注記**: 独立したフェイルオーバを可能にするため、データベース用に 1 つの LUN、SAP データ用に 1 つの LUN を用意することを推奨します。
5. 設定によっては、次の作業を完了させる必要があります。

- 物理デバイスを作成する
- ボリュームグループを作成する
- SAP 用の論理ボリュームを作成する
- データベース用の論理ボリュームを作成する

6. 共有ストレージ上に SAP 用のファイルシステム (すなわち、 *sapmnt* 、 *saptrans* 、*ASCS{No}*、*SCS{No}*、*DVEBMGS{No}* ) を作成してください。 **注記:** すべてを共有ストレージ上に配置するためには、SAP を停止する必要があります。

7. データベースに必要なすべてのファイルシステムを作成してください (例: *mirrlogA*、*mirrlogB*、*origlogA*、*origlogB*、*sapdata1*、*sapdata2*、*sapdata3*、*sapdata4*、*oraarch*、*saparch*、*sapreorg*、*saptrace*、*oraflash -- mkfs -t ext3 /dev/oracle/mirrlogA* )。

   **注記:** データベース固有のリカバリキットについては、 SPS for Linux テクニカルドキュメンテーション を参照し、LifeKeeper で作成および保護する必要のあるファイルシステムに関する追加情報については、 *Component Installation Guide SAP Web Application Server* を参照してください。

8. メイン SAP ファイルシステムにマウントポイントを作成し、マウントしてください (必要な場合)。追加情報については、NFS マウントポイントと inode トピックを参照してください。 **(注記:** */exports* ディレクトリは、ファイルシステムをマウントするために使用されました。)

   ```
   mount /dev/sap/sapmnt /exports/sapmnt

   mount /dev/sap/saptrans /exports/saptrans
   ```

9. 以下のコマンドを使用して、一時的なマウントポイントを作成してください。

   ```
   mkdir /tmp/m{No}
   ```

10. 3 つの SAP ディレクトリをマウントしてください (以下のマウントポイントは、外部 NFS を使用する場合もそうでない場合も存在するそれぞれのアプリケーションサーバで必要です)。

    ```
    mount /dev/sap/ASCS00 /tmp/m1

    mount /dev/sap/SCS01 /tmp/m2

    mount /dev/sap/DVEBMGS02 /tmp/m3
    ```

共有ディスクおよび LifeKeeper へのデータ移動 に進みます。

# 6.16.7.9. 共有ディスクおよび LifeKeeper へのデータ移動

次の手順は、Oracle を使用する場合の例です。 **注記** ：この例では、DB ID および SAP SID として「STC」が使用されています。これらのコマンド（たとえば、/usr/sap/STC またはユーザー orastc ）で使用されている「STC」または「stc」はすべて、ユーザーのクラスタ構成で使用されている実際の DB ID または SAP SID に置き換える必要があります。

**開始する前に:** プライマリサーバーとバックアップサーバーは、2 台のサーバーに指定されています。この手順の最後に、これらの役割が予約されます。まずは手順を最後まで通して読み、どのマシンをプライマリにし、どのマシンをバックアップにするのが良いかを計画することを推奨します。この手順を最後まで行うと、プライマリの役割とバックアップの役割が交換可能になります。特定の環境ではあるマシンがプライマリで、あるマシンがバックアップになることを考慮し、その構成方法を理解することが重要です。

1. ディレクトリを */usr/sap/STC* に変更し、各サブディレクトリに移動し、データをコピーしてください。

   - cd ASCS{No.}
   - cp –a * /tmp/m1
   - cd ../SCS{No.}
   - cp –a * /tmp/m2
   - cd ../DVEBMGS{No.}
   - cp –a * /tmp/m3

2. これらの一時ディレクトリを正しいユーザー権限に変更してください。

   ```
   chown stcadm:sapsys /tmp/m1 (repeat for m2 and m3)
   ```

3. `umount /tmp/m1` を使用して これらの 3 つの一時ディレクトリをアンマウントし、m2 と m3 についても同様に行ってください。

4. デバイスを古いディレクトリに再マウントしてください。

   ```
   mount /dev/sap/ASCS{No.} /usr/sap/STC/ASCS{No.}

   mount /dev/sap/SCS{No.} /usr/sap/STC/SCS{No.}

   mount /dev/sap/DVEBMGS{No.} /usr/sap/STC/DVEBMGS{No.}
   ```

5. Oracle 用の 13 個の一時ディレクトリをマウントしてください。

```
mount /dev/oracle/sapdata1 /tmp/m1

mount /dev/oracle/sapdata2 /tmp/m2

mount /dev/oracle/sapdata3 /tmp/m3

mount /dev/oracle/sapdata4 /tmp/m4

mount /dev/oracle/mirrlogA /tmp/m5

mount /dev/oracle/mirrlogB /tmp/m6

mount /dev/oracle/origlogA /tmp/m7

mount /dev/oracle/origlogB /tmp/m8

mount /dev/oracle/saparch /tmp/m9

mount /dev/oracle/sapreorg /tmp/m10

mount /dev/oracle/saptrace /tmp/m11

mount /dev/oracle/oraarch /tmp/m12

mount /dev/oracle/oraflash /tmp/m13
```

6. ディレクトリを */oracle/STC* に変更し、データをコピーしてください。

   a. 各サブディレクトリに移動してください (`cd sapdata1` を実行し、`cp -a * /tmp/m1` を実行する)

7. 前の手順を、上記の関係に示すように各サブディレクトリについて繰り返します。

8. これらの一時ディレクトリを正しいユーザー権限に変更してください。

   ```
   chown orastc:dba /tmp/m1 (repeat for m2 to m12)
   ```

9. すべての一時ディレクトリをアンマウントしてください。

   ```
   umount /tmp/m*
   ```

10. デバイスを古いディレクトリに再マウントしてください。

    ```
    mount /dev/oracle/sapdata1 /oracle/STC/sapdata1
    ```

11. リストされているすべてのディレクトリに対し、上記の手順を繰り返してください。

12. */etc/exports* ファイルを編集し、SAP のメインディレクトリ用のマウントポイントを挿入してくだ

さい。

```
/exports/sapmnt *(rw,sync,no_root_squash)

/exports/saptrans *(rw,sync,no_root_squash)
```

13. `systemctl start nfs-server.service` コマンドを使用して NFS サーバーを起動してください。NFS サーバーがすでにアクティブの場合、「`exportfs -va`」を使用してこれらのマウントポイントをエクスポートする必要があります。

14. 次のマウントコマンドを実行します ( フェイルオーバとリカバリで重要になるので、**udp** の使用方法に注意してください )。

```
mount { virtual ip }:/exports/sapmnt/< SID > /sapmnt/< SID > -o
rw,sync,bg,udp

mount { virtual ip }:/exports/saptrans /usr/sap/trans -o
rw,sync,bg,udp
```

15. Oracle にログインし、Oracle を起動します (su コマンドで orastc に切り替えた後)。

```
lsnrctl start LISTENER_STC

sqlplus / as sysdba

startup
```

16. SAP にログインし、SAP を起動します (su コマンドで stcadm に切り替えた後)。

```
startsap sap{No.}
```

17. すべてのプロセスが起動されていることを確認してください。

```
ps -ef | grep en.sap (2 processes)

ps -ef | grep ms.sap (2 processes)

ps -ef | grep dw.sap (17 processes)
```

SAP MCC ( SAP 用 Microsoft 管理コンソールスナップイン ) は、SAP インスタンスの管理に使用できる SAP 提供の Windows クライアントです。Unix/Linux 対応バージョンの SAP MC ( SAP 管理コンソール) も利用できます。

18. SAP と Oracle Listener を停止してください。( 注記 : ステップ c で、Oracle の SQL*Plus ユーティリティを使用して Oracle にログインし、データベースをシャットダウンしています。)

    a. su コマンドで stcadm に切り替え、コマンド「`stopsap sap{No.}`」を入力してくださ

い。

b. su コマンドで orastc に切り替え、コマンド「`lsnrctl stop LISTENER_STC`」を入力してください。

c. su コマンドで orastc に切り替え、コマンド「`sqlplus sys as SYSDBA`」を入力し、コマンドプロンプトに「`shutdown`」を入力してください。

d. コマンド「`stopsap sap{No.}`」を入力してください。

e. ルートユーザーとして、「`killall sapstartsrv`」を実行してください。

f. stcadm および orastc ユーザーにまだ関連付けられている残りのプロセスを強制終了してください。（例： `ps -u stcadm` および `ps -u orastc` ）

19. すべてのファイルシステムをアンマウントしてください。

```
umount /usr/sap/trans

umount /sapmnt/STC

umount /oracle/STC/*

umount /usr/sap/STC/DVEBMGS{No.}

umount /usr/sap/STC/SCS{No.}

umount /usr/sap/STC/ASCS{No.}
```

20. `systemctl stop nfs-server.service` コマンドを使用して NFS サーバーを停止し、アンマウントを実行してください。

```
umount /exports/sapmnt

umount /exports/saptrans
```

21. */etc/exports* をバックアップシステムにコピーしてください。

```
scp /etc/exports (backup ip):/etc/exports
```

22. 論理ボリュームをプライマリ上で非アクティブ化してください。

```
lvchange -an oracle

lvchange -an sap
```

23. 対応する SAP ディレクトリをバックアップシステム上に作成してください。

```
mkdir -p /exports/sapmnt

mkdir -p /exports/saptrans
```

24. 論理ボリュームをバックアップシステム上でアクティブ化してください。

```
lvchange -ay oracle

lvchange -ay sap
```

**注記:** ボリュームグループが構築されたとき、プライマリ上でストレージの再編成が発生した場合、この手順で問題が発生する場合があります。バックアップを再起動することで、この問題を解消できます。

25. バックアップマシンにディレクトリをマウントしてください。

```
mount /dev/sap/sapmnt /exports/sapmnt

mount /dev/sap/saptrans /export/saptrans

mount /dev/sap/ASCS00 /usr/sap/STC/ASCS{No.}

mount /dev/sap/SCS01 /usr/sap/STC/SCS{No.}

mount /dev/sap/DVEBMGS02 /usr/sap/STC/DVEBMGS{No.}

mount /dev/oracle/sapdata1 /oracle/STC/sapdata1

mount /dev/oracle/sapdata2 /oracle/STC/sapdata2

mount /dev/oracle/sapdata3 /oracle/STC/sapdata3

mount /dev/oracle/sapdata4 /oracle/STC/sapdata4

mount /dev/oracle/origlogA /oracle/STC/origlogA

mount /dev/oracle/origlogB /oracle/STC/origlogB

mount /dev/oracle/mirrlogA /oracle/STC/mirrlogA

mount /dev/oracle/mirrlogB /oracle/STC/mirrlogB

mount /dev/oracle/oraarch /oracle/STC/oraarch

mount /dev/oracle/saparch /oracle/STC/saparch

mount /dev/oracle/saptrace /oracle/STC/saptrace
```

```
mount /dev/oracle/sapreorg /oracle/STC/sapreorg
```

26. LifeKeeper を介して、IP アドレスをバックアップシステムにスイッチオーバしてください。
27. NFS エクスポートをバックアップ上にマウントしてください。

```
mount sap{No.}:/exports/sapmnt/STC /sapmnt/STC

mount sap{No.}:/exports/saptrans/trans /usr/sap/trans
```

27. Oracle にログインし、Oracle を起動します (su コマンドで orastc に切り替えた後)。

```
lsnrctl start LISTENER_STC

sqlplus / as sysdba

startup
```

28. SAP にログインし、SAP を起動します (su コマンドで stcadm に切り替えた後)。

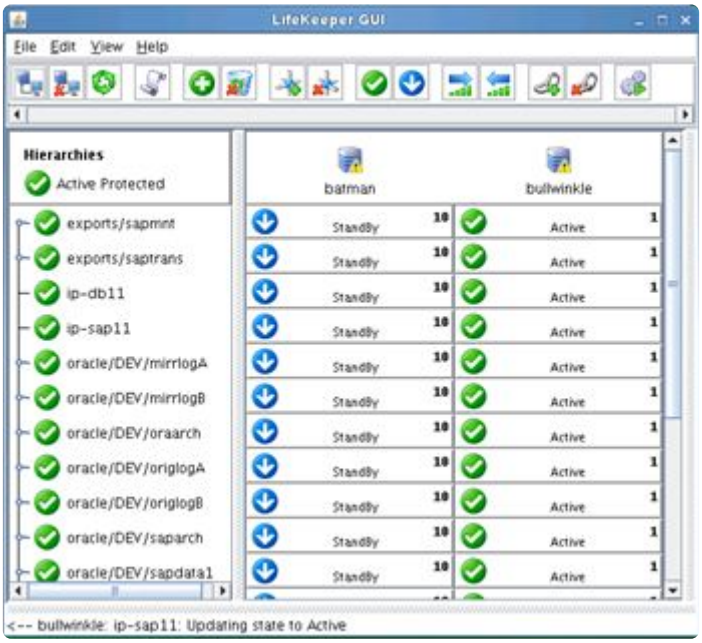
```
startsap sap{No.}
```

29. LifeKeeper にログインし、プライマリとバックアップの優先インスタンスを切り替えてください (バックアップを高い優先度にする)。
30. 元のプライマリ上で、元のディレクトリを以下のように保存します。

```
mv /exports /exports-save

mv /usr/sap/STC/DVEBMGS{No.} /usr/sap/STC/DVEBMGS{No.}-save (repeat
for SCS{No.} and ASCS{No.})

mv /oracle/STC/sapdata1 /oracle/STC/sapdata1-save (repeat for
sapdata2, sapdata3, sapdata4, mirrlogA, mirrlogB, origlogA,
origlogB, sapreorg, saptrace, saparch, oraarch)
```

31. 「ファイルシステム」リソース、17 個のマウントポイント (SAP 用に 5 個、Oracle 用に 12 個) を 1 つずつ作成してください。
32. 元のプライマリを拡張してください。

LifeKeeper リソース階層と SAP クラスタが構築されました。( 注記 : これは、DEV インスタンスからのスクリーンショットです。)


LifeKeeper GUI
File Edit View Help
Hierarchies
Active Protected
exports/sapmnt
exports/saptrans
ip-db11
ip-sap11
oracle/DEV/mirrlogA
oracle/DEV/mirrlogB
oracle/DEV/oraarch
oracle/DEV/origlogA
oracle/DEV/origlogB
oracle/DEV/saparch
oracle/DEV/sapdata1
batman
bullwinkle
Standby
Active
<-- bullwinkle: ip-sap11: Updating state to Active

# 6.16.7.10. ASCSおよびERSインスタンスプロファイル設定の変更

エンキューサーバーおよびエンキューレプリケータープロセス（それぞれASCSおよびERSインスタンスプロファイル内に存在）のRestart_ProgramパラメーターをStart_Programに変更して、sapstartユーティリティがそれらを自動的に再起動しないようにします。また、両方のインスタンスプロファイルのエントリ「Autostart = 0」を「Autostart = 1」に変更し、システムの再起動時に各インスタンスが自動的に再起動しないようにしてください。

次の手順を実行する前に、アクティブなASCSおよびERSインスタンスプロファイルの場所を特定します。これは、「pf=」パラメーターとしてsapstartsrvに提供されたファイルのパスを調べることにより、/usr/sap/sapservicesファイルで見つけることができます。

> 以下は例です。ここではASCSインスタンス番号として00、ERSインスタンス番号として10を使用していますが、ご利用の環境に合わせて番号を変更してください。また、<SID>を実際のシステムIDに置き換えてください。

```
LD_LIBRARY_PATH=/usr/sap/SID/ASCS00/exe:$LD_LIBRARY_PATH; export

LD_LIBRARY_PATH; /usr/sap/SID/ASCS00/exe/sapstartsrv

pf= /usr/sap/SID/SYS/profile/SID_ASCS00_sap1 -D -u SIDadm

LD_LIBRARY_PATH=/usr/sap/SID/ERS10/exe:$LD_LIBRARY_PATH; export

LD_LIBRARY_PATH; /usr/sap/SID/ERS10/exe/sapstartsrv

pf= /usr/sap/SID/SYS/profile/SID_ERS10_sap2 -D -u SIDadm
```

この例では、アクティブなASCSインスタンスプロファイルは/usr/sap/SID/SYS/profile/SID_ASCS00_sap1 に、アクティブなERSインスタンスプロファイルは /usr/sap/SID/SYS/profile/SID_ERS10_sap2 にあります。

## 手順

1. 以下のように、ASCSインスタンスプロファイルを編集します。

   a.「Autostart = 1」を「Autostart = 0」に変更するか、「Autostart = 0」という行を手動で追加します。

   b. エンキューサーバープロセスを開始するプロファイル内の行の正確な形式は、スタンドアロンエンキューサーバーフレームワークのバージョン1または2のどちらが使用されているかによって異なります。次の表に従ってプロファイルを変更してください。

| 変更前（この行が表示されたら、右の列の内容に変更してくだ | 変更後 |
|---|---|

| さい) | |
|---|---|
| Restart_Program_01 = local $(_EN) pf=$(_PF) | Start_Program_01 = local $(_EN) pf=$(_PF) |
| Restart_Program_01 = local $(_ES2) pf=$(_PF) | Start_Program_01 = local $(_ES2) pf=$(_PF) |
| Restart_Program_01 = local $(_ENQ) pf=$(_PF) | Start_Program_01 = local $(_ENQ) pf=$(_PF) |

Start_Program_xxまたはRestart_Program_xxエントリの番号xxは、お使いのシステムでは異なる場合があります。
上記の内容に一致するように変更する必要はありません。

2. すべてのエントリが正しいことを確認し、ASCSインスタンスプロファイルへの変更を保存します。

3. ERSインスタンスプロファイルを次のように編集します。

   a.「Autostart = 1」を「Autostart = 0」に変更するか、「Autostart = 0」という行を手動で追加します。

   b. エンキューレプリケータープロセスを開始するプロファイル内の行の正確な形式は、スタンドアロンエンキューサーバーフレームワークのバージョン1または2のどちらが使用されているかによって異なります。次の表に従ってプロファイルを変更してください。

| 変更前（この行が表示されたら、右の列の内容に変更してください) | 変更後 |
|---|---|
| Restart_Program_00 = local $(_ER) pf=$(_PFL) NR=$(SCSID) | Start_Program_00 = local $(_ER) pf=$(_PFL) NR=$(SCSID) |
| Restart_Program_00 = local $(_ER2) pf=$(_PF) | Start_Program_00 = local $(_ER2) pf=$(_PF) |
| Restart_Program_00 = local $(_ENQR) pf=$(_PF) | Start_Program_00 = local $(_ENQR) pf=$(_PF) |

Start_Program_xxまたはRestart_Program_xxエントリの番号xxは、お使いのシステムでは異なる場合があります。
上記の内容に一致するように変更する必要はありません。

4. すべてのエントリが正しいことを確認し、変更をERSインスタンスプロファイルに保存します。

5. システムを再起動して、更新されたプロファイルが読み込まれ、キャッシュが有効になっていないことを確認することを推奨します。

6. また、本番環境のワークロードに適用する前に、常にテスト環境での変更を検証することを推奨します。

autostart の値を変更した場合は、変更を反映させるため再起動が必要になります。

注記：

1. エントリ「Autostart = 0」と「Start_Program_xx ...」は、各インスタンスプロファイルの別々の行にある必要があります。

2. Start_Program_xxまたはRestart_Program_xxエントリの番号xxは、お使いのシステムでは異なる場合があります。この例で指定されている数字と一致するように変更する必要はありません。

3. Javaベースまたはデュアルスタックの Java+ABAP デプロイメントには、SCSセントラルサービスインスタンスがあります。この場合、SCSインスタンスに対しても手順1～3を実行する必要があります。

4. Start_ProgramとRestart_Programの違いの詳細については、SAP Note 768727（Automatic restart functions in sapstart for processes）を参照してください。

# 6.16.7.11. SAP Recovery Kit の旧バージョンからのアップグレード

旧バージョンの SAP Recovery Kit からアップグレードするには、次の手順を実行してください。

1. アップグレードの前に、設定の計画 トピックを読み直して、新しいソフトウェアによる影響をすべて理解しているか確認してください。

   注記: SAP Netweaver 7.3 より前のバージョンを実行している場合、SAPHOST エージェントをインストールする必要があります。詳細については、「設定の計画」トピックの重要事項 を参照してください。

   lkbackupユーティリティを使用して、現在の階層のスナップショットを取得することを推奨します。

2. *SPS for Linux インストールガイド* の SPS のアップグレード トピックの手順に従ってください。

   既存の階層のバックアップが実行されます。その後、アップグレードにより旧階層が破壊され、新しい階層が再作成されます。障害が発生した場合、以下の「障害が発生した場合 」を参照してください。

3. アップグレードが終了したら、LifeKeeper GUI を停止してから再起動し、更新された GUI クライアントをロードしてください。

   LifeKeeper GUI サーバーはページをキャッシュするため、新しいページを更新するために再起動が必要になります。root として、コマンド「lkGUIserver restart」を入力すると、GUI サーバーが停止し再起動されます。このような再起動を実行する前に、すべてのクライアントを終了させてください。

   **注記:** すべての LifeKeeper システムを再起動する必要はなく、テストでシステムのリサイクルは要求されませんが、実稼働環境では、一定時間のダウンタイムを計画し、正式なシステム準備時間に行うことをお勧めします。

4. LifeKeeper UI にログインし、階層をチェックし、階層が正しいことを確認してください。

## 障害が発生した場合

アップグレードを再試行できます。アップグレードスクリプトは、 */tmp* ディレクトリに完全な状態で保存されています (*lkcreatesaptmp*)。これは、アップグレード中に使用される一時ファイルです。ここにコマンドが記述され、階層を作成するために実行できます。

障害やエラーが発生した場合、または階層が正しくない可能性がある場合、次の手順を推奨します。

1. `$LKROOT/bin/lkcli stop` を実行して、LifeKeeper を停止してください。

2. `rpm -e steeleye-lkSAP` を実行して、新しい rpm を削除してください。

3. `rpm -i steeleye-lkSAP-<previous version>.noarch.rpm` を実行して、古い rpm をインストールしてください。

4. `lkbackup -x` を実行して、古い階層をリストアしてください。

5. LifeKeeper を再起動してください。

6. SIOS サポートにお問い合わせ ください。サポートへのお問い合わせ前に、ログ、階層の前のスナップショット、作成され障害が発生した階層、アップグレード中に表示されたエラーメッセージをお手元に用意してください。

# 6.16.7.12. IP リソース

LifeKeeper リソース階層の設定を続行する前に、SAP リソースがフェイルオーバまたはスイッチオーバに使用する IP アドレスを決定してください。これは、通常、SAP のインストール中にパラメータ SAPINST_USE_HOSTNAME で使用した仮想 IP アドレスです。この IP アドレスは、クラスタ内のノード間で共有される 1 つの仮想 IP アドレスで、一度に 1 つのノード上でアクティブになります。この IP アドレスは、データベース階層を保護するために使用される IP アドレスとは異なります。これらの IP アドレスは、SAP リソースを作成するときにも使用するので、メモしておいてください。

# 6.16.7.13. SAP リソース階層の作成

https://fast.wistia.net/embed/iframe/3j1g7ubr4a

参考動画「NFS を使用して SIOS SAP/EC2 リソース階層を作成する (英語コンテンツ)」

SAP システムを保護するには、SAP 階層が必要です。この SAP 階層は、コア (セントラルサービス) リソース、ERS リソース、プライマリリソース、セカンダリリソースで構成されます。この階層を作成するには、プライマリサーバから次の作業を実行します。 **注記**: 以下の例は、階層作成の指針として提供するものです。実際の構成によって、作業は少々異なります。

## コアリソースの作成

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択します。

   ダイアログボックスが表示され、クラスタ内にインストールされ認識されているすべての Recovery Kit が、ドロップダウンリストボックスに一覧で示されます。ドロップダウンメニューから **[SAP]** を選択します。

   **[Next]** をクリックします。

   ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、以前に入力した情報を訂正する必要がある場合に特に役立ちます。

階層作成の途中で **[Cancel]** ボタンをクリックすると、作成処理全体が取り消されます。

2. **[Switchback Type]** を選択します。この選択によって、バックアップサーバへのフェイルオーバの後、SAP インスタンスがサービス中に戻るときに、このサーバにどのようにスイッチバックされるのかが決まります。 **[intelligent]** または **[automatic]** を選択できます。 **[Intelligent switchback]** の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。 **[Automatic switchback]** の場合、プライマリサーバがオンラインに戻り、LifeKeeper コミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。

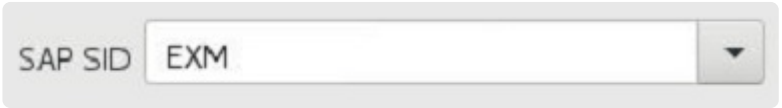
スイッチバックタイプは、必要に応じて **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。

**[Next]** をクリックします。

3. SAP PAS、ASCS、または SCS を配置するサーバ (通常これは、プライマリサーバまたはテンプレートサーバと呼ばれます) を選択します。クラスタ内のすべてのサーバがドロップダウンリストに表示されます。

**[Next]** をクリックします。

4. **[SAP SID]** を選択します。これは、保護対象の SAP PAS、ASCS、または SCS システムのシステム識別子です。




**[Next]** をクリックします。

5. 保護対象の SID に対する SAP インスタンス名 (例: *ASCS<No.>* ) (コアインスタンスが先) を選択します。

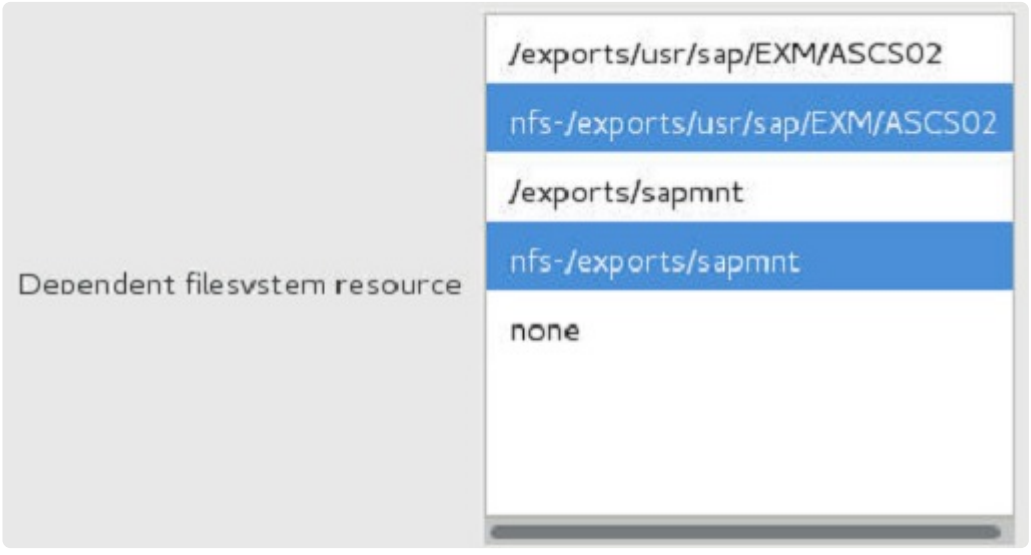
**[Next]** をクリックします。

**注記**: 保護およびリカバリレベルのカスタマイズに関して、その他の画面が表示されることがあります。

6. **[IP Child Resource]** を選択します。これは、一般的に、SAP インストール中にメモした仮想ホスト IP アドレス (SAPINST_USE_HOSTNAME) か、フェイルオーバに必要な IP アドレスのいずれかです。

a. LifeKeeperがインスタンスの **依存ファイルシステムの作成を自動化するかどうか** を選択します。 **yes** を選択すると、LifeKeeperは必要なファイルシステムリソースを作成し、それをSAPリソースの下に依存関係として追加します。（複製されたファイルシステムは自動では作成できません）。 **no** を選択すると次のダイアログボックスが表示され、依存関係として追加する既存のLifeKeeperファイルシステムリソースを選択するよう求められます。

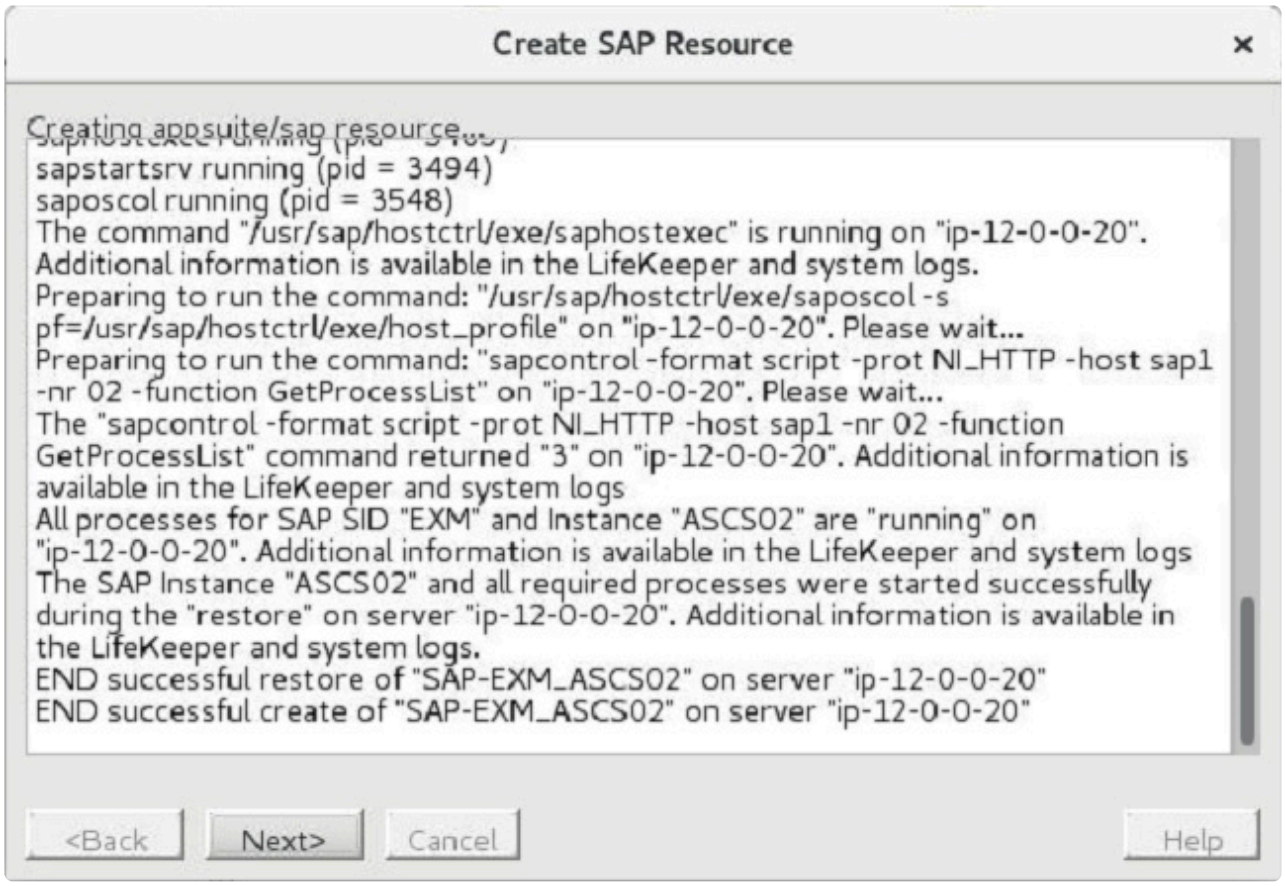
b. 前のダイアログで **no** を選択した場合、SAPリソースの下に依存関係として追加されるファイルシステムリソースを選択するオプションが提供されます。Ctrlキーを押しながら各リソースをクリックすると、複数のリソースを選択できます。 **注記：** このダイアログで選択肢として表示するには、LifeKeeperのこのサーバでファイルシステムリソースがIn Service（ISP）になっている必要があります。



7. **[SAP Tag]** を選択または入力します。これは、LifeKeeper が SAP リソースに与えるタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。デフォルトのタグは、*SAP-<SID>_<INST>* です。

**[Create]** をクリックすると、 ***[Create SAP Resource Wizard]*** によって SAP リソースが作成されます。

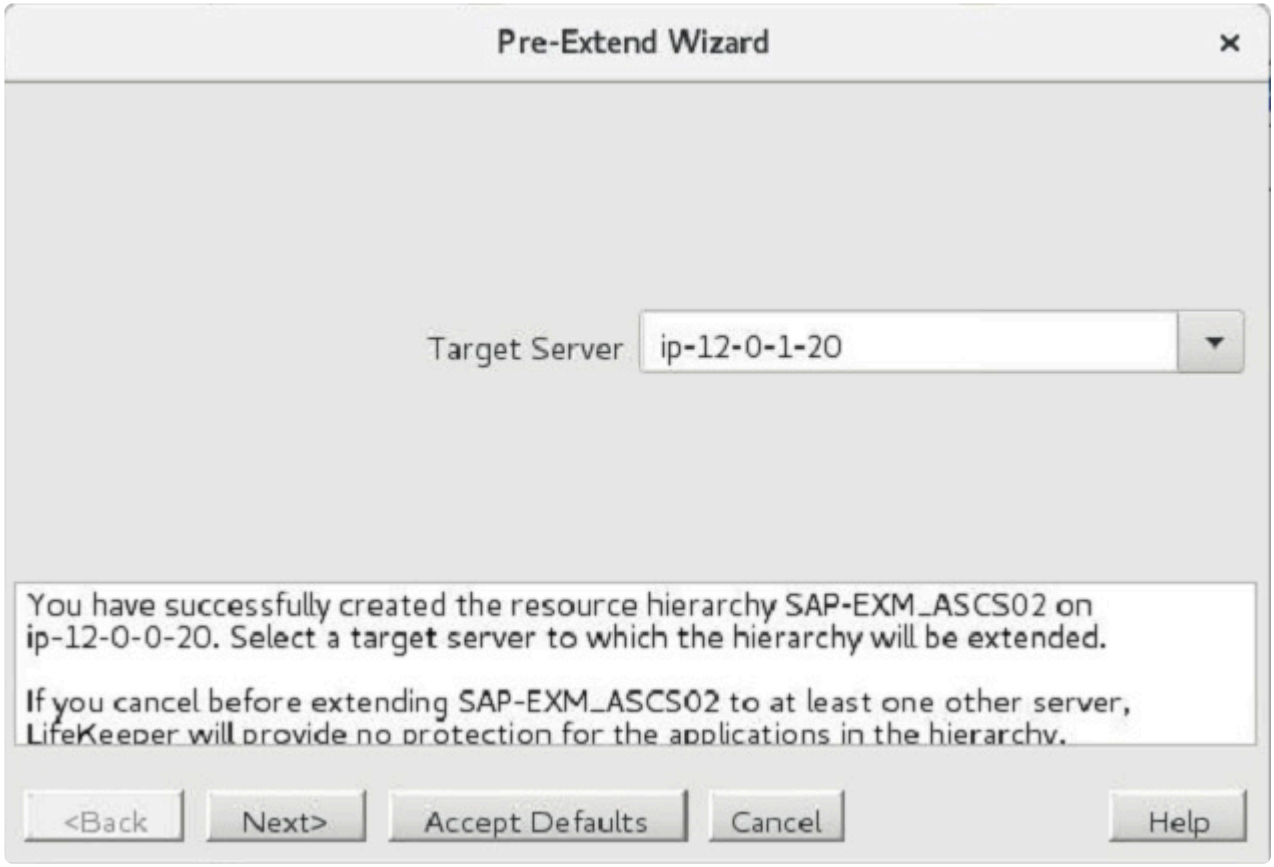
8. この時点で情報ボックスが表示され、LifeKeeper は、SAP リソース階層を作成するのに有効なデータが提供されているか検査します。LifeKeeper が問題を検知した場合は、情報ボックスにエラーが表示されます。検証が正常に完了すると、リソースが作成されます。SAP の起動スクリプトによって出力されたエラーやメッセージが、情報ボックスに表示されることもあります。




**[Next]** をクリックします。

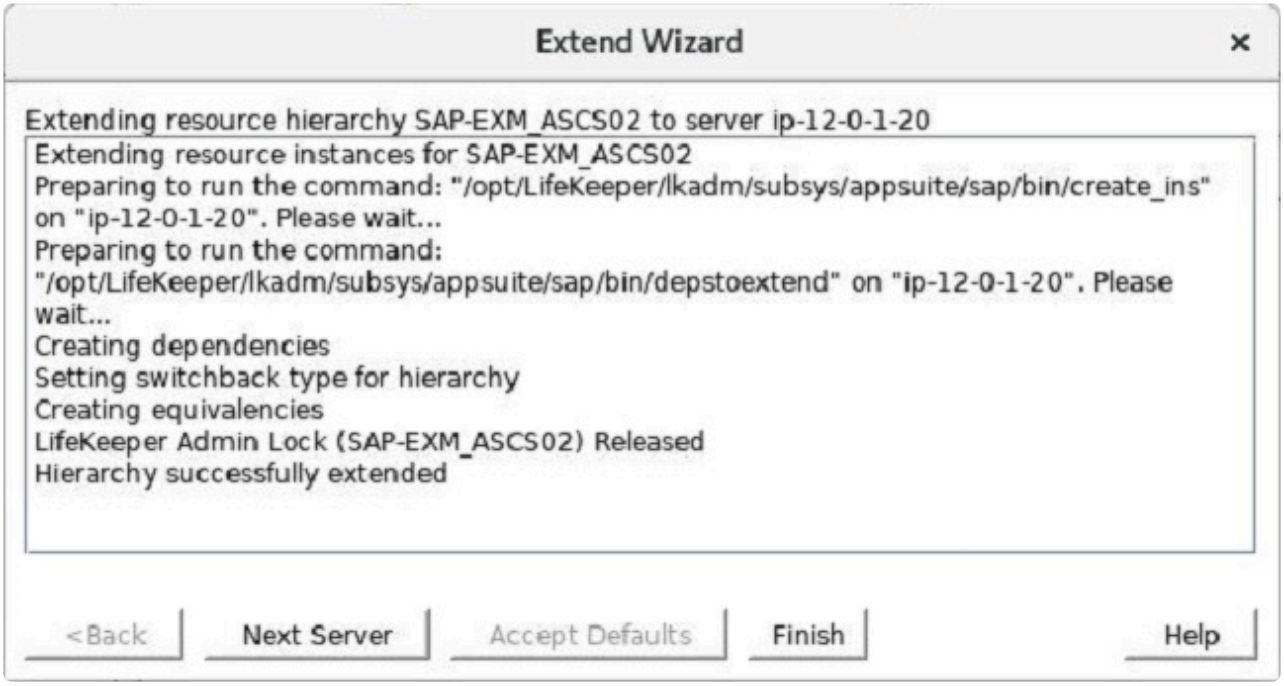
9. 別の情報ボックスが表示され、SAP リソース階層が正常に作成されたことが示されます。SAP リソース階層を LifeKeeper の保護下に置くには、その階層をクラスタ内の別のサーバに **拡張** する必要があります。

**[Next]** をクリックすると、 ***[Pre-Extend Wizard]*** が起動します。このウィザードについては、本セクションで後述します。




ここで **[Cancel]** をクリックすると、別のダイアログボックスが表示され、いずれかの時点でここに戻り、SAP リソース階層を別のサーバに拡張して、LifeKeeper の保護下に置く必要があることが警告されます。

10. ***[Extend Wizard]*** ダイアログに「 **Hierarchy successfully extended** 」と表示されます。 **[Finish]** をクリックします。




11. **[Hierarchy Integrity Verification]** ダイアログが表示されます。階層の検証が終了したら、 **[Done]** をクリックして、 **[Create Resource Hierarchy]** メニューを終了します。

コアが最上位レベルにある階層

# ERS リソースの作成

ERSリソースは、コアインスタンス（セントラルサービスインスタンス）またはエンキューサーバプロセスの単一障害点に対する追加の保護を提供します。コアインスタンス（セントラルサービスインスタンス）に障害が発生して再起動されると、コアインスタンスはエンキューレプリケーションサーバからエンキューロックテーブル（レプリケーションテーブル）のバックアップコピーを取得します。その結果、エンキューサーバに障害が発生してもトランザクションや更新は失われず、SAPシステムのサービスが継続されます。

SPS-Lのバージョン9.4.0以降と9.4.0より前でのERSリソースの実装の違いの説明については、LifeKeeperのERSリソースタイプ を参照してください。

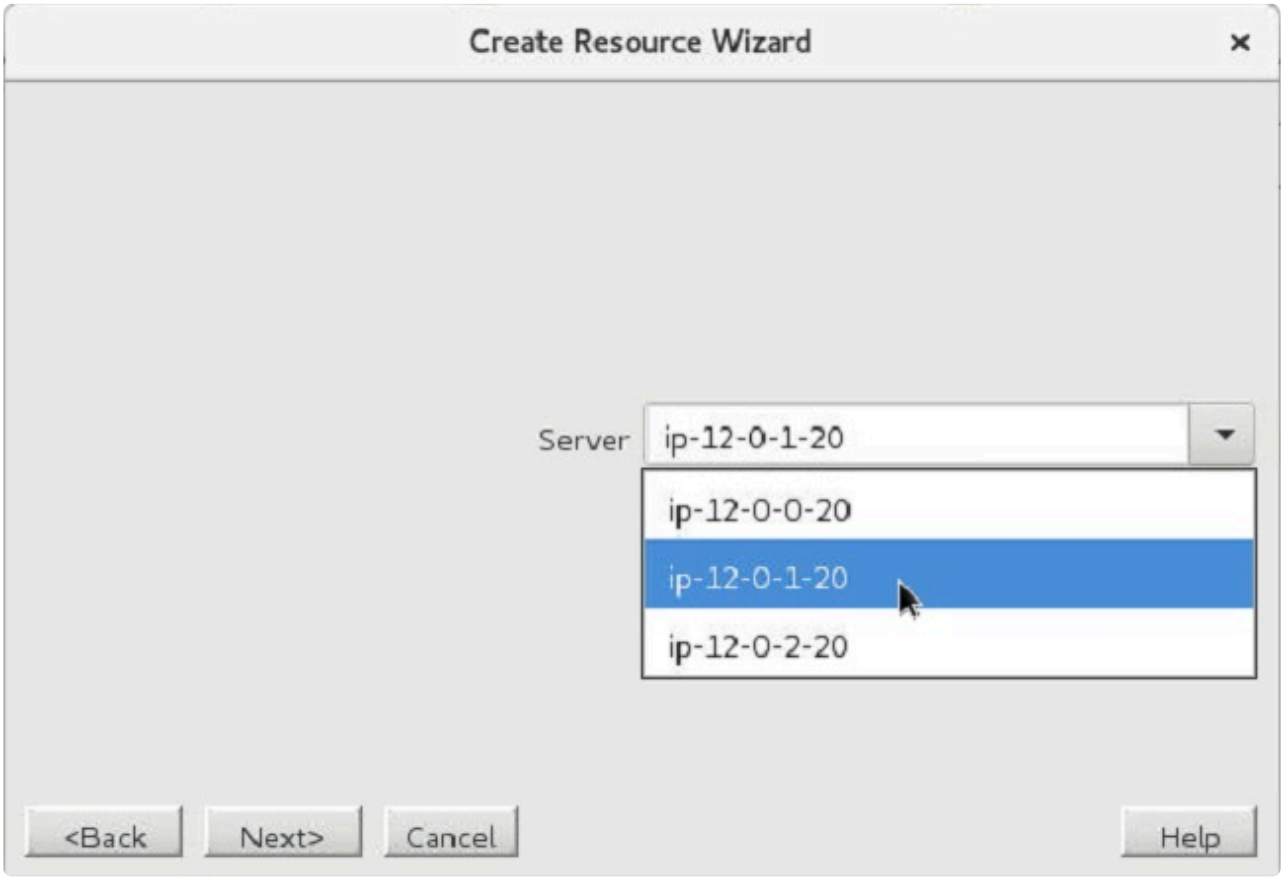
> **重要な注記:** ERS階層の作成と拡張は、 **対応するASCS/SCSインスタンスがLifeKeeperでIn Service (ISP)になってないサーバで実施する必要があります。** ERSv2の場合、ベースとなるERSファイルシステムリソースは、リソースが作成されるバックアップサーバで最初にIn Serviceにする必要があります。

次の手順を実行して、ERSリソースを作成します。

1. LifeKeeper GUIメニューから、 **[Edit]** 、 **[Server]** の順に選択します。ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択します。クラスタ内にインストールされ、認識されているすべてのリカバリーキットを含むドロップダウンリストボックスのあるダイアログボックスが表示されます。ドロップダウンリストから **SAP** を選択し、 **[Next]** をクリックします。

2. **Switchback Type** を選択し、 **[Next]** をクリックします。

3. **重要:** 対応するASCS/SCSインスタンスが **ISPではない** クラスタ内の **サーバ** を選択します。 **[Next]** をクリックします。

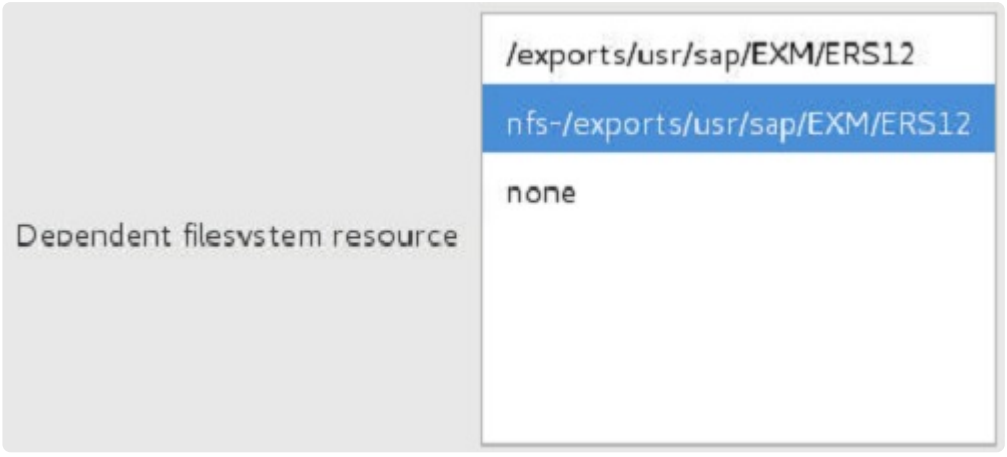


4. ERSインスタンスの **SAP SID** を選択し、 **[Next]** をクリックします。

5. **SAP**インスタンス名 (例. ERS<No.>) を選択し、 **[Next]** をクリックします。

6. ERSv2インスタンスを表すリソースを作成する場合は、 **IP Child Resource** を選択します。ERSv1インスタンスを表すリソースを作成する場合、この選択画面は表示されません。

7. ERSv2インスタンスを表すリソースを作成する場合、ERSリソースの下に依存関係として追加する **Dependent Filesystem Resource** を選択します。このリストに表示させるには、LifeKeeperのこのサーバでファイルシステムリソースがIn Service（ISP）になっている必要があります。ERSv1インスタンスを表すリソースを作成する場合、この選択画面は表示されません。




8. **SAP Tag** を選択するか、入力します。

9. プロンプトに従って、 **リソース階層を拡張します** 。 **注記:** ERSv1インスタンスを表すリソースは、1つのバックアップサーバにのみ拡張できます。ERSv1階層を3番目のクラスタノードに拡張しようとすると、階層の拡張は失敗します。

10. **[Hierarchy Successfully Extended]** と表示されたら、 **[Finish]** を選択します。

11. **[Done]** を選択します。

**ASCSおよびERSv1が別の階層にあります**

**ASCSおよびERSv2が別の階層にあります**

ASCSインスタンスがENSAv2を実行し、ERSインスタンスがERSv2を実行しており、3つ以上のノードを持つクラスタ内の同じシステムに階層が拡張されている場合、ASCSおよびERSリソース階層は、特別な「階層回避」Genericアプリケーションリソースを使用することにより、スイッチオーバーとフェールオーバーで互いを回避するように強制できます。
これらのリソースを作成し、ASCSおよびERS階層に依存関係として追加する方法の詳細については、ENSAv2/ERSv2使用時にASCS/ERSに回避動作を強制する を参照してください。 **注記：** これらの階層回避リソースは、ENSAv1/ERSv1構成または2ノードクラスタでは使用しないでください。

## プライマリアプリケーションサーバリソースの作成

1. ここでも同様に、同じ SAP SID について上記の手順を繰り返し、 **DVEBMGS{ *XX* }** (ここで {XX} はインスタンス番号) を選択して、プライマリアプリケーションサーバリソースを作成します。

2. 画面の指示に従って、 **[Level of Protection]** を選択します (デフォルトは **[FULL]** )。 **[Next]** をクリックします。

3. 画面の指示に従って、 **[Level of Recovery]** を選択します (デフォルトは **[FULL]** )。 **[Next]** をクリックします。

4. **[Dependent Instances]** の入力を要求されたら、「親」インスタンス、すなわち、上記で作成した **ERS** インスタンス を選択します。

5. **[IP Child Resource]** を選択します。

6. 画面の指示に従ってリソース階層を拡張します。

7. 「 **Hierarchy Successfully Extended** 」と表示されたら、 **[Finish]** をクリックします。

8. **[Done]** を選択します。

プライマリアプリケーションサーバが最上位レベルにある階層

# セカンダリアプリケーションサーバリソースの作成

必要な場合、セカンダリアプリケーションサーバリソースを同様の方法で作成します。

**注記:** コマンドラインでの手順については、コマンドラインによる SAP の設定 を参照してください。

# 6.16.7.14. リソース階層の削除

LifeKeeper 設定内の すべての サーバからリソースを削除するには、次の手順を実行します。

注記: 階層全体を削除するには、各リソースを個別に削除する必要があります。

1. **[Edit]** メニューの **[Resource]** から **[Delete Resource Hierarchy]** を選択します。
2. リソース階層を削除する **[Target Server]** の名前を選択します。

   注記: 左側ペインのグローバルリソースまたは右側ペインの個々のリソースインスタンスを右クリックして **[Delete Resource]** 作業を選択した場合、このダイアログボックスは表示されません。

   **[Next]** をクリックします。
3. **[Hierarchy to Delete]** を選択します。削除するリソース階層を特定して、強調表示にします

   注記: このダイアログは、右ペインまたは左ペインのリソースインスタンス上で右クリックして **[Delete Resource]** を選択した場合には表示されません。

   **[Next]** をクリックします。
4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。 **[Delete]** をクリックします。
5. リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。 **[Done]** をクリックして終了します。

# 6.16.7.15. 共通の Recovery Kit 作業

以下の作業は、すべての Recovery Kit を通じて同一の手順からなる共通の作業のため、SPS for Linux テクニカルドキュメンテーション の管理 セクションで説明しています。

- リソース依存関係の作成 。既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- リソース依存関係の削除 。リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service 。リソース階層を特定のサーバで in service にします。
- Out of Service 。リソース階層を特定のサーバで out of service にします。
- プロパティの表示 /編集 。特定のサーバでリソース階層のプロパティを表示または編集します。

# 6.16.7.16. コマンドラインによる SAP の設定

コマンドラインを使用して、SAP Recovery Kit を設定できます。

## コマンドラインによる SAP リソースの作成

プライマリサーバから、以下のコマンドを実行します。

```
$LKROOT/lkadm/subsys/appsuite/sap/bin/create <primary sys> <tag> <SAP SID> <SAP Instance> <switchback type> <IP Tag> <Protection Level> <Recovery Level> <Additional SAP Dependents>
```

**例:**
$LKROOT/lkadm/subsys/appsuite/sap/bin/create liono SAP-STC_SCS00 STC SCS00 intelligent ip-sap10 Full Full none

**注記:**

- **Switchback Type** - この選択によって、バックアップサーバへのフェイルオーバの後、SAP インスタンスが In Service に戻るときに、このサーバにどのようにスイッチバックされるのかが決まります。 **intelligent** または **automatic** を選択できます。intelligent の場合、インスタンスをプライマリ / オリジナルサーバにスイッチバックするときに管理者の介入が必要になります。automatic すなわち自動スイッチバック の場合、プライマリサーバがオンラインに戻り、LifeKeeper のコミュニケーションパスを再確立した直後に自動的にスイッチバックが行われます。
- **IP Tag** - SAP リソース階層の依存リソースになる IP リソースを表します。
- Protection Level - **Protection Level** は、各リソースに実行できるアクションを表します。
- Recovery Level - **Recovery Level** は、障害発生時のリソースに対する指示を提供します。
- **Additional SAP Dependents** - この値は、現在作成しようとしている SAP リソースの依存リソースになる LifeKeeper SAP リソースタグを表します。

## コマンドラインによる SAP リソースの拡張

SAP リソースの拡張では、既存の階層をあるサーバからコピーし、同様の階層を別の LifeKeeper サーバに作成します。コマンドラインでリソースを拡張するには、以下のコマンドを実行してください。

```
system "$LKROOT/lkadm/bin/extmgrDoExtend.pl -p 1 -f, \"$tag\" \"$backupnode\"
\"$priority\" \"$switchback\" \\\"$sapbundle\\\"";
```

例: 使いやすくするために、シンプルなスクリプトを使用できます。

```
#!/etc/default/LifeKeeper-perl
```

```
require "/etc/default/LifeKeeper.pl";
my $lkroot="$ENV{LKROOT}";
my $tag="SAP";
my $backupnode="snarf";
my $switchback="INTELLIGENT";
my $priority=10;
$sapbundle = "\"$tag\",\"$tag\"";
system "$lkroot/lkadm/bin/extmgrDoExtend.pl -p 1 -f, \"$tag\"
\"$backupnode\"
\"$priority\" \"$switchback\" \\\"$sapbundle\\\"";
```

# 6.16.7.17. SAP SIOS HA Cluster Connector（SHCC）のアクティブ化

SAP SIOS HA Cluster Connector（SSHCC）は、SAP Start Service（sapstartsrv）とLifeKeeper間のインターフェースを提供します。
SAPインスタンスに対してHA Cluster Connectorがアクティブになっている間、クラスター内のリソースのステータスを最新に保つために、インスタンスの状態に影響するsapcontrolを介した呼び出しはLifeKeeperによってルーティングされます。

SAPインスタンスのSAP SIOS HA Cluster Connectorをアクティブにするには、次の手順を実行します。

1. SAPインスタンスのアクティブなプロファイルの場所を特定します。これは、インスタンスに対応する行を検索することにより、/usr/sap/sapservicesファイルで見つけることができます。たとえば、ASCSインスタンスに対応する次の行を考えます。

```
LD_LIBRARY_PATH=/usr/sap/STC/ASCS00/exe:$LD_LIBRARY_PATH; export
LD_LIBRARY_PATH;
/usr/sap/STC/ASCS00/exe/sapstartsrv pf=/usr/sap/STC/SYS/profile/STC_ASCS00_sap1
-D -u stcadm
```

   この例では、アクティブなASCSインスタンスプロファイルは/usr/sap/STC/SYS/profile/STC_ASCS00_sap1にあります。

2. ステップ1. で見つかったインスタンスプロファイルを編集し、ファイルの下部に次の行を追加します。

```
#SIOS
#-------------------------------------------
service/halib = saphascriptco.so
service/halib_cluster_connector = /opt/LifeKeeper/lkadm/subsys/appsuite/sap/
bin/sap_sios_cluster_connector
service/halib_debug_level = 1
```

3. クラスター内の各システムの/etc/sudoersファイルに次の行を追加して、SAP管理者ユーザーがsap_sios_cluster_connectorスクリプトを実行できるようにします。次の行の<sid>をシステムの小文字のSAP SIDに置き換えます。

```
<sid>adm ALL=NOPASSWD:/opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/
sap_sios_cluster_connector-main
```

4. プロファイルの変更を反映するには、インスタンスのsapstartsrvプロセスを再起動する必要があります。これは、次のコマンドで実行できます（<sid>を小文字のSAP SIDに、<SID>を大文字のSAP SIDに置き換えます）。

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function RestartService <SID>"
```

5. SAPインスタンスに対してHA Cluster Connectorが正常にアクティブ化されたことを確認するに

は、次のコマンドを実行します。

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function HAGetFailoverConfig"
```

- インスタンスに対してHA Cluster Connectorが正常に構成されている場合、このコマンドの出力には、次のサンプル出力のようにHAActive: TRUEと表示されます。

```
HAGetFailoverConfig
OK
HAActive: TRUE
HAProductVersion: "SIOS Protection Suite for Linux" steeleye-lk 9.3.2-6863
HASAPInterfaceVersion: "SIOS Protection Suite for Linux" steeleye-lkHACONNECTO
R-for-SAP 7.5.0-6855
HADocumentation: docs.us.sios.com/Linux/current/LK4L/SAP/index.htm
HAActiveNode: ip-12-0-0-20
HANodes: ip-12-0-0-20, ip-12-0-1-20, ip-12-0-2-20
```

- HA Cluster Connectorが正常に初期化されなかった場合、関連するエラーメッセージは、通常/usr/sap/<

SID>/<INST>/work/sapstartsrv.log内にあるインスタンスのsapstartsrv.logファイルに出力されます。

# 6.16.7.18. テストの準備

1. 仮想 SAP サーバ名を使用して SAP にログインできるよう、SAP GUI を SAP クライアントに設定します。

2. リダンダント AS にログインできるよう、SAP GUI を SAP クライアントに設定します。

3. 必要な場合、追加の AS をクラスタ内の別のサーバにインストールして、PAS を除くすべてのアプリケーションサーバ間にログイングループを設定します。インストールされているすべての AS で、上述したようにプロファイルファイルを変更する必要があります。

# 6.16.7.19. テストの実行

以下の一連のテストを実行してください。テスト手順は、各設定で異なります。一部の手順では、SAP が正常に実行されていることを求めますが、実行すべき具体的なテスト内容は記載されていません。SAP が正常に設定され実行されていることを検証するために行うことのできるテストの一覧については、SAP ドキュメント、 *SAP R/3 in Switchover Environments* の付録を参照してください。

## アクティブ / アクティブ設定のテスト

1. SAP 階層の作成時、SAP と DB は、異なるサーバ上で In Service になります。SAP GUI から、SAP にログインします。ログインが成功し、SAP が正常に実行されていることを確認します。

2. ログアウトし、リダンダント AS からログインします。ログインが成功することを確認します。

3. ログイングループを設定した場合、そのグループを使用してログインが成功することを確認します。

4. LifeKeeper GUI を使用して、SAP バックアップサーバ上で SAP のリソース階層を In Service にします。これで、SAP と DB の両方が同一のサーバ上で In Service になります。

5. 同様に、SAP 仮想サーバ名、リダンダント AS、ログイングループを使用して SAP にログインできることを検証します。SAP が正常に実行されていることを確認します。

6. LifeKeeper GUI を使用して、DB バックアップサーバ上で DB 階層を In Service にします。これで、各階層はバックアップサーバ上で In Service になります。

7. 同様に、すべてのログイン方法を使って SAP にログインできることと、SAP が正常に実行されていることを確認します。トランザクション SM21 を実行すると、ログにおいて PAS が DB との通信を失った記録と再取得した記録を確認できます。

8. SAP にログイン中に、SAP が現在 In Service である SAP バックアップサーバの電源スイッチを押してシャットダウンします。SAP 階層が SAP プライマリサーバ上で In Service になること、そしてフェイルオーバ後、PAS に再度ログインできることと正常に実行されていることを確認します。

9. 切断したサーバの電源をリストアします。LifeKeeper GUI を使用して、DB プライマリ サーバ上で DB 階層をもう一度 In Service にします。同様に、SAP にログイン中に、DB が現在 In Service である DB プライマリサーバの電源スイッチを押してシャットダウンします。DB 階層が DB バックアップサーバ上で In Service になること、そしてフェイルオーバ後に SAP にログインでき、トランザクションを正常に実行できることを確認します。

10. 切断したサーバの電源をリストアします。LifeKeeper GUI を使用して、DB プライマリ サーバ上で DB のリソース階層をもう一度 In Service にします。

## アクティブ / スタンバイ設定のテスト

1. 階層作成時、SAP と DB の両方がプライマリサーバ上で In Service になります。リダンダント AS がバックアップサーバ上で開始されます。SAP GUI から、SAP にログインします。ログインが成功

し、SAP が正常に実行されていることを確認します。トランザクション SM51 を実行し、SAP サーバのリストを表示します。このリストに、PAS または ASCS と AS の両方が含まれている必要があります。

2. ログアウトし、バックアップサーバ上のリダンダント AS を介して再ログインします。ログインが成功することを確認します。

3. ログイングループを設定した場合、そのグループを使用してログインが成功することを確認します。

4. LifeKeeper GUI を使用して、バックアップサーバ上で SAP/DB のリソース階層を In Service にします。

5. 同様に、SAP 仮想サーバ名、リダンダント AS、ログイングループを使用して SAP にログインできることを検証します。SAP が正常に実行されていることを確認します。

6. SAP にログイン中に、階層が現在 In Service になっている SAP/DB バックアップサーバの電源スイッチを押してシャットダウンします。SAP/DB 階層がプライマリサーバ上で In Service になること、そしてフェイルオーバ後、PAS に再度ログインできることと PAS が正常に実行されていることを確認します (サーバがダウンするときに接続が切れるため、再ログインする必要があります)。

7. 切断したサーバの電源をリストアします。同様に、SAP にログイン中に、DB が現在 In Service である SAP/DB プライマリサーバの電源スイッチを押してシャットダウンします。SAP/DB 階層がバックアップサーバ上で In Service になること、そしてフェイルオーバ後に SAP に再度ログインでき、SAP が正常に実行されていることを確認します。

8. 同様に、切断したサーバの電源をリストアします。LifeKeeper GUI を使用して、プライマリサーバ上で SAP/DB のリソース階層を In Service にします。

# 6.16.8. 管理

## 管理のヒント

このセクションでは、特定の設定の管理およびメンテナンスに役立つヒントやその他の情報を提供します。

NFS の考慮事項

クライアントの再接続

LifeKeeper SAP の設定可能パラメータ

SAP 階層と NFS 階層の分離

保護レベルの更新

リカバリレベルの更新

プロパティの表示

## Oracle データベース

Oracle に関する特別な考慮事項

# 6.16.8.1. NFS の考慮事項

設定上の考慮事項 トピックで前述したように、SAP プライマリまたはバックアップサーバのいずれかにおいてファイルシステムが NFS 共有をローカルにマウントするよう設定されている場合、NFS 階層を Out of Service にするとシステムがハングし、再起動を正常に行えません。不注意で NFS サーバを停止してクラスタをハングさせないよう、次のようにすることを推奨します。

- 保護された NFS 共有に対するローカル NFS マウントポイントを含むサーバ上で、NFS リソース階層を Out of Service 状態にしないでください。NFS 子リソースが In Service である限り、SAP リソースは自由に In Service と Out of Service を切り替えることができます。また、サーバをシャットダウンする前に、NFS 階層を別のサーバで In Service にすることができます。

- ローカルでマウントされた NFS 共有を保護している NFS リソース階層が In Service であるサーバ上の LifeKeeper を停止する必要がある場合、必ず、 `-f` オプションを使用してください。LifeKeeper をコマンド `lkstop -f` を使用して停止すると、LifeKeeper は階層を Out of Service 状態にすることなく停止することができ、ローカル NFS マウントによるサーバハングを防ぐことができます。詳細については、 lkstop マニュアルページ を参照してください。

- ローカルでマウントされた NFS 共有を保護している NFS リソース階層が In Service であるサーバを再起動する必要がある場合、先に LifeKeeper を、上述の `-f` オプションを使用して停止してください。-f オプションを使用して停止せずにサーバを再起動すると、システムにより LifeKeeper が `-f` オプションなしで停止されるため、NFS リソース階層が Out of Service 状態になり、システムがハングします。

- SAP パッケージをアンインストールする必要がある場合、サーバ上で In Service であり保護された (ISP: In-Service Protected) NFS リソースを含む SAP リソース階層がある場合はアンインストールしてはなりません。SAP リソース階層を削除してから、パッケージをアンインストールしてください。

- SPS をアップグレードする場合、もしくは SPS インストール設定スクリプトを実行する必要がある場合、SPS for Linux インストールガイド に記載されているアップグレード手順に従うことを推奨します。には、 SPS インストールイメージファイル上の設定スクリプトを実行したり、 SPS パッケージをアップデートしたりする前に、アップグレード対象のサーバからすべてのアプリケーションを退避させることについて記載されています。特に、LifeKeeper インストールイメージファイル上の設定スクリプトは、LifeKeeper がアクティブな NFS 共有を保護しているサーバ上で実行してはなりません。なぜなら、nfsd カーネルモジュールをアップグレードするにはそのサーバ上の NFS を停止する必要があり、ローカルでマウントされた NFS ファイルシステムによりサーバがハングする可能性があるからです。その他の情報については、NFS Server Recovery Kit ドキュメンテーション を参照してください。

- TCP を使用すると、forceumount 呼び出し中の Out of Service 操作中にハングする可能性があります。NFS 共有にアクセスできない場合、アンマウントが失敗することがあります。LifeKeeper は、ファイルシステムのアンマウントを複数回試行します。複数回試行することにより、通常、最終的にリソースを Out of Service にすることに成功します。ただし、リソースを Out of Service にする際に遅延が発生します。再試行を回避するには、マウントオプション「nfsvers=3, proto=udp 」を使用します。

> **!** 注記：フェイルオーバとリカバリで重要になるので、udp の使用方法に注意してください。

- /sapmnt（または /sapmnt/<SID>）ファイルシステムが NFS 経由で共有されている場合、NFS 共有の喪失に起因する SAP リソース管理アクションのハングを防ぐために、クラスタ内の各ノードの /etc/default/LifeKeeper に「SAP_NFS_CHECK_DIRS=/sapmnt」を追加する必要があります。

> **!** ファイルシステムが AWS 上の EFS と共有されている場合、pingnfs チェックが適用されないため、SAP_NFS_CHECK_DIRS は使用しないでください。

# 6.16.8.2. クライアントの再接続

SAP クライアントは、特定の SAP インスタンスにログオンするよう設定することも、ログオングループにログオンするよう設定することもできます。ログオングループを介してログオンするよう設定した場合、SAP は、クライアントが実行中のどのインスタンスに実際に接続するかを決定します。

クライアントが接続されているインスタンスがダウンした場合、クライアント接続が失われ、クライアントは再ログインする必要があります。データベースが一時的に失われたものの、クライアントが接続しているインスタンスは動作中である場合、データベースが復帰するまでクライアントは一時的に使用不可になりますが、再ログオンする必要はありません。

パフォーマンス上の理由から、クライアントは、PAS、ASCS、または SCS ではなく、リダンダントアプリケーションインスタンスにログオンする必要があります。ただし、管理者は、ログの表示などのために PAS にログオンできます。SAP を LifeKeeper で保護した後、クライアントログインは仮想 SAP サーバ名を使用して設定でき、SAP インスタンスが SAP プライマリサーバまたはバックアップサーバ上でアクティブであるかに関わらずクライアントがログオンできるよう設定できます。

# 6.16.8.3. SAP Recovery Kit のチューニング

SAP スクリプトのいくつかはタイムアウト機能を備えており、スクリプトをハングさせて自動的に自分自身を終了させることができます。この機能は、使用不可の NFS 共有による問題に備えて要求されます。これについては、NFS マウントと su トピックでより詳細に説明します。この機能を備えた各スクリプトは、デフォルトのタイムアウト値 (秒単位) を含んでおり、必要に応じて、その値をオーバーライドできます。タイムアウトの待ち時間を短縮するには、必要な NFS ファイルシステムをカンマ区切りリストとして「SAP_NFS_CHECK_DIRS」に追加します。これにより、pin でファイルシステムがエクスポートされていることを確認できます。エクスポートされていない場合はすぐに戻ります。

SAP_CREATE_NAS の有効 / 無効を切り替えできます。SAP_CREATE_NAS のデフォルトは 1 (有効) です。これは、NAS マウントされたファイルシステムの NAS リソースを自動的に含めるために使用されます。無効にするには、このパラメータを 0 に設定してください。

チューニング可能な SAP_CONFIG_REFRESH のデフォルトは LKCHECKINTERVAL/2 です。ユーザーは呼び出し間隔の秒数を変更して、LifeKeeper GUI の SAP リソースのプロパティパネルを更新できます。

以下の表は、スクリプト名、変数名、およびデフォルト値を示しています。デフォルト値をオーバーライドするには、*/etc/default/LifeKeeper* ファイルに行を追加し、そのスクリプトに必要な値を設定してください。例えば、remove スクリプトが終了されるまで 1 分間実行されるようにするには、*/etc/default/LifeKeeper* に次の行を追加します。

```
SAP_REMOVE_TIMEOUT=60
```

注記: スクリプトは、実際には、終了前にタイムアウト値よりもわずかに長く実行されることがあります。

注記: これらの値を変更するとき、LifeKeeper を停止したり再起動する必要はありません。

| スクリプト名 | 変数名 | デフォルト値 |
|---|---|---|
| remove | SAP_REMOVE_TIMEOUT | 804秒 |
| restore | SAP_RESTORE_TIMEOUT | 516秒 |
| recover | SAP_RECOVER_TIMEOUT | 1320秒 |
| quickCheck | SAP_QUICKCHECK_TIMEOUT | 60秒 |
| create NAS | SAP_CREATE_NAS | 1 (無効にするには 0 に設定) |
| GUI Properties Panel refresh | SAP_CONFIG_REFRESH | LKCHECKINTERVAL の値の 1/2 |
| NFS shares to check | SAP_NFS_CHECK_DIRS | なし |

注記: NetWeaver Java のみの環境では、SCS インスタンスに加えて Java PAS を起動するよう設定した場合、SAP_RESTORE_TIMEOUT と SAP_RECOVER_TIMEOUT の値を増加させる必要がある場合があります。

# 6.16.8.4. SAP リソース階層と NFS リソース階層の分離

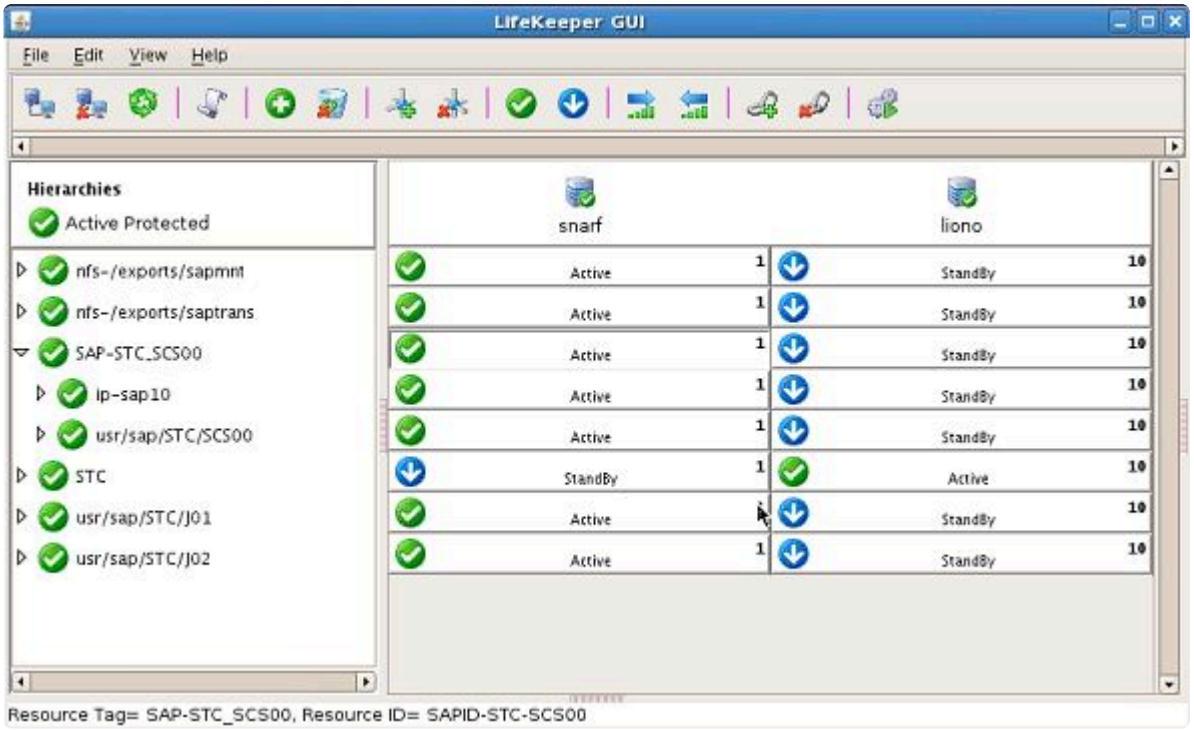
このセクションで説明する LifeKeeper SAP リソース階層では、NFS リソース階層を SAP リソースの子依存関係として実装しますが、SAP リソース階層を作成した後、NFS リソース階層を切り離して管理することは可能です。依存関係を解除する前に、下記で説明するように、これらを独立した階層として管理することによるメリットとデメリットを考慮する必要があります。これは、切り離す NFS 共有が他の SAP ファイルシステムとは独立した論理ボリューム (LUN) 上にホストされている場合のみ可能であることに注意してください。

これらの 2 つの階層を分離して管理するには、このドキュメンテーションに従って SAP リソース階層を作成した後、LifeKeeper GUI により、SAP リソースと NFS リソース間の依存関係を手動で解除します。

**SAP リソース階層と NFS リソース階層を分離して管理することのメリット:** NFS で問題が発生した場合、SAP とは独立して NFS リソース階層をフェイルオーバできます。この場合、SAP が NFS マウントされたディレクトリの一時的な欠落を透過的に処理する限り、SAP フェイルオーバは起こりません。NFS 共有が利用できない場合に LifeKeeper がハングするのを避けるためには、NFS マウントが /etc/mtab にあり、/etc/default/LifeKeeper のチューニング可能な値 SAP_NFS_CHECK_DIRS にリストされている必要があります。

**SAP リソース階層と NFS リソース階層を分離して管理することのデメリット:** NFS 共有において、PAS、ASCS、SCS、または ERS インスタンスが実行されているのと同じサーバーでホストすることは保証されていません。 **注記:** SAP の推奨事項については、 *SAP Installation Guide* を参照してください。

以下の図は、依存関係を解除した後の SAP リソース階層と NFS リソース階層を示しています。

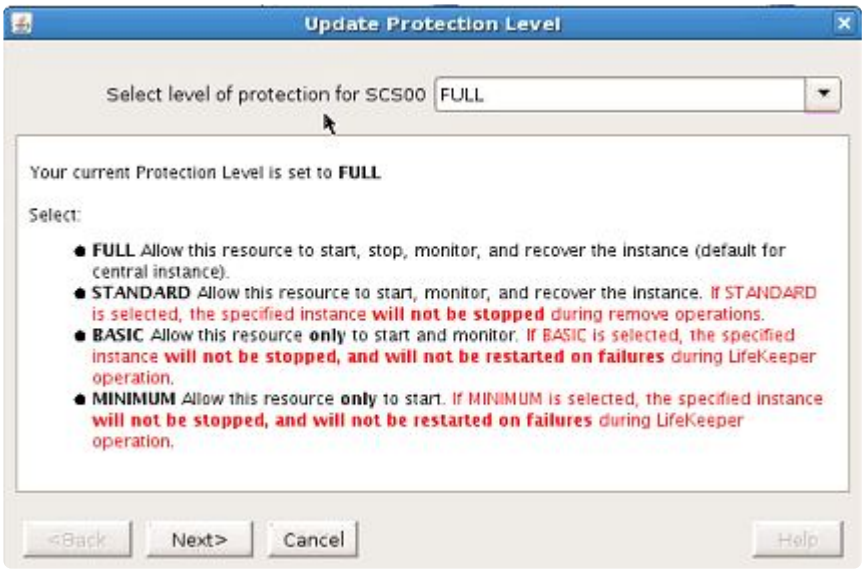


# 6.16.8.5. 保護レベルの更新

保護レベル は、各リソースに実行できるアクションを表します。現在の保護レベルを確認するか、このオプションを変更するには、 **[Update Protection Level]** を使用します。保護のレベルは、 **[FULL]** 、 **[STANDARD]** 、 **[BASIC]** 、または **[MINIMUM]** に設定できます。

1. インスタンス を右クリックします。
2. **[Update Protection Level]** をクリックします。

3. 以下の画面が表示され、保護レベルを選択できます。




**[FULL]**。 デフォルトのレベルです。完全な保護を提供し、インスタンスを起動、停止、監視、および復旧できます。

**[STANDARD]**。このレベルを選択すると、リソースによりインスタンスを起動、監視、および復旧できますが、リソースの停止中は、インスタンスの停止はできません。

**[BASIC]**。このレベルを選択すると、リソースによる起動と監視のみが行えます。障害発生時に停止したり再起動したりすることはできません。

**[MINIMUM]**。 このレベルを選択すると、リソースによる起動のみが行えます。障害発生時に停止し

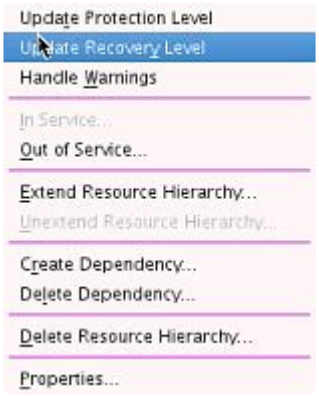
たり再起動したりすることはできません。

**注記:** 保護レベル **[BASIC]** と **[MINIMUM]** は、LifeKeeper 保護アプリケーションを一時的なメンテナンスモードにするために使用します。 **[BASIC]** または **[MINIMUM]** をリソースの保護レベルの継続的な状態として使用することは推奨しません。詳細については、「トラブルシューティング」セクションの階層削除時のエラー を参照してください。

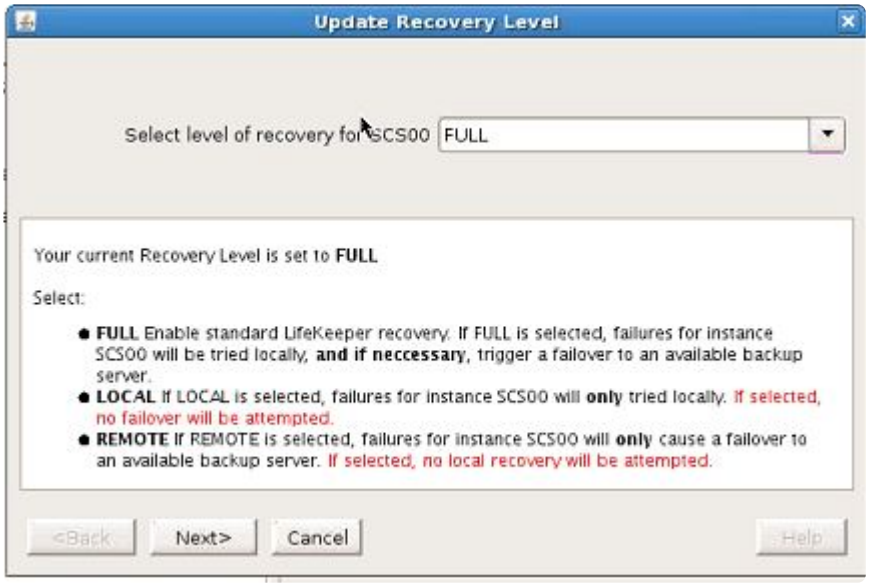
# 6.16.8.6. リカバリレベルの更新

**リカバリレベル** は、障害発生時のリソースに対する指示を提供します。現在のリカバリレベルを確認するか、このオプションを変更するには、 **[Update Recovery Level]** を使用します。リカバリレベルは、 **[FULL]** 、 **[LOCAL]** 、または **[REMOTE]** に設定できます。

1. インスタンス を右クリックします。
2. **[Update Recovery Level]** をクリックします。




3. 以下の画面が表示され、 **リカバリレベル** を選択できます。




**[FULL]**。 リカバリレベルが **[FULL]** に設定されている場合、リソースはローカルリカバリを試みます。それが失敗した場合、対向ノードでの復旧を、成功するまで試みます。

**[LOCAL]**。 リカバリレベルが **[LOCAL]** に設定されている場合、リソースはローカルリカバリのみを試みます。フェイルオーバは行いません。

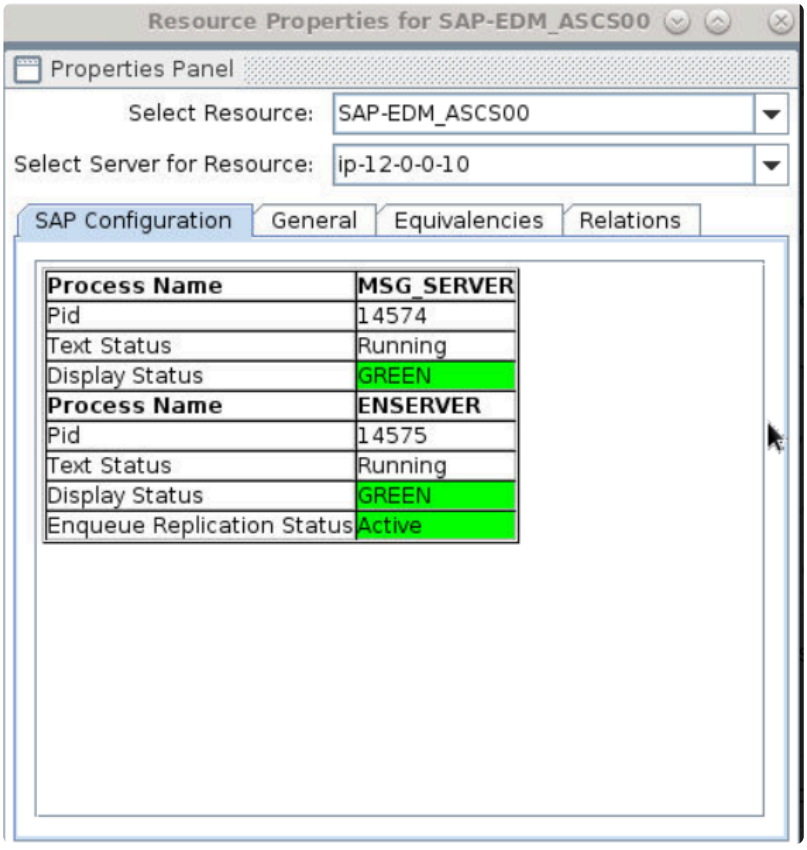
**[REMOTE]**。 リカバリレベルが **[REMOTE]** に設定されている場合、対向ノードでの起動を試みます。ローカルリカバリを試みることはありません。

# 6.16.8.7. プロパティの表示

**[Resource Properties]** ページでは、特定の SAP リソースの設定の詳細を確認できます。特定のサーバ上のリソースのプロパティを表示するには、または SAP プロセスの状態を表示するには、**[Properties]** 画面を開きます。

1. インスタンス を右クリックします。
2. **[Properties]** を選択します。

3. 以下のような **[Properties]** 画面が表示されます。




表示された **[Properties]** 画面にはタブが 4 つあります。1 番目のタブは [SAP Configuration] で, 特定の SAP リソースの設定情報が設定されています。残り 3 つのタブは、LifeKeeper のすべてのリソースタイプで共通の項目として使用できます。

# 6.16.8.8. Oracle に関する特別な考慮事項

一度 SAP プロセスが LifeKeeper クラスタ内のシステムで機能し始めると、主要な SAP 機能について、LifeKeeper でリソースを作成する必要が生じます。これには、ASCS システム、DVEBMGS システム、SCS システム、および Oracle データベースが含まれます。

このトピックでは、Oracle を LifeKeeper 環境で保護するためのいくつかの特別な考慮事項を説明します。

- LifeKeeper for Linux Oracle Application Recovery Kit がインストールされていることを確認してください。
- Oracle Recovery Kit ドキュメンテーション を参照してください。
- SAP のインストール中、SAPinst プロセスは、通常、データベースソフトウェアが既にインストールされ設定済みであることを前提としています。ただし、SAP で使用するデータベースが Oracle の場合、SAPinst プロセスはインストーラから Oracle インストールツール (RUNINSTALLER) を起動し、Oracle のインストールを完了させます。
- SAP のインストール中に Oracle をインストールすると、Oracle SID が作成されます。この SID は、Oracle Recovery Kit によって要求されます。そのため、Oracle リソースを LifeKeeper で作成するときに提供できるよう準備してください。
- Oracle を使用して 標準的な SAP インストールを作成する場合、Oracle インスタンスが使用する 13 個の個別のファイルシステムが作成されます。一般的に、これらのファイルシステムは、それぞれ LVM 論理ボリュームの最上位階層に構築され、それぞれに多くの別々の物理ボリュームが含まれることがあります。LifeKeeper が正しくこれらのファイルシステムを表現するため、各物理ボリュームと論理ボリュームおよびそれらのボリュームグループに対して個別のリソースが作成されます。この大規模なリソース集合を LifeKeeper 階層にまとめる必要があるため、Oracle 階層の作成と拡張を完了するにはある程度の時間がかかることがあります。作成処理に 1時間かかり、さらに拡張処理に 10～20分かかることは珍しくありません。
- 必要な Oracle (および SAP) ファイルシステムを LVM の最上位階層に構築することは必須ではなく、LifeKeeper 内の SAP と Oracle の Recovery Kit は、標準の Linux ファイルシステムで良好に動作します。
- LifeKeeper Oracle Recovery Kit は、Oracle SAP インストールが標準の Oracle 依存関係として使用する 13 個のファイルシステムのうち、10 個を識別することができ、これらのファイルシステムについて自動的に階層に依存関係を作成します。Oracle Recovery Kit は、 *saptrace* 、*sapreorg*、*saparch* ファイルシステムを自動的に認識しません。LifeKeeper を設定する管理者は、これらの追加のファイルシステムについて、手動で リソース依存関係の作成 を行う必要があります。

# 6.16.8.9. SSHCC HA アクション

SAP SIOS HA Cluster Connector (SSHCC) アクション は、SAP SIOS HA Cluster Connector と併用されるリソースに対して有効なアクションです。 次の SSCC HA Actions は、スタートインスタンス、ストップインスタンス、マイグレートインスタンスのそれぞれにリソースに対して有効です。

- **Start Instance：** Start Instance を選択することで、現在のノード上の特定のリソースタグに対してSAP SIOS HA Cluster Connectorのスタートアクションを実行します。
- **Stop Instance：** Stop Instance を選択することで、現在のノード上の特定のリソースタグに対してSAP SIOS HA Cluster Connectorのストップアクションを実行します。
- **Migrate Instance：** Migrate Instance を選択することで、現在のノード上の特定のリソースタグに対してSAP SIOS HA Cluster Connectorのマイグレートアクションを実行します。
- **Maintenance Mode：** Maintenance Mode を選択することで、すべてのクラスタノード上の指定されたリソースタグに対して、SAP SIOS HAクラスタコネクタのメンテナンスモード操作を実行します。

> **注記 ：**これらのアクションは高度な操作設定であり、稼働中、もしくは待機中のLifeKeeperの標準的な操作を置き換えるものとして使用されるものではありません。

この操作を行うには、次の手順を実行してください。

1. インスタンスを右クリックします。
2. 利用可能なメニューから SSHCC HA アクションを選択します。
3. 選択肢から目的の操作を選択します。
4. 指定されたタグに対して選択されたSSHCC HA操作を確認します。
5. [Update] を選択して、選択した操作を続行します。

# 6.16.8.10. LifeKeeperのERSリソースタイプ

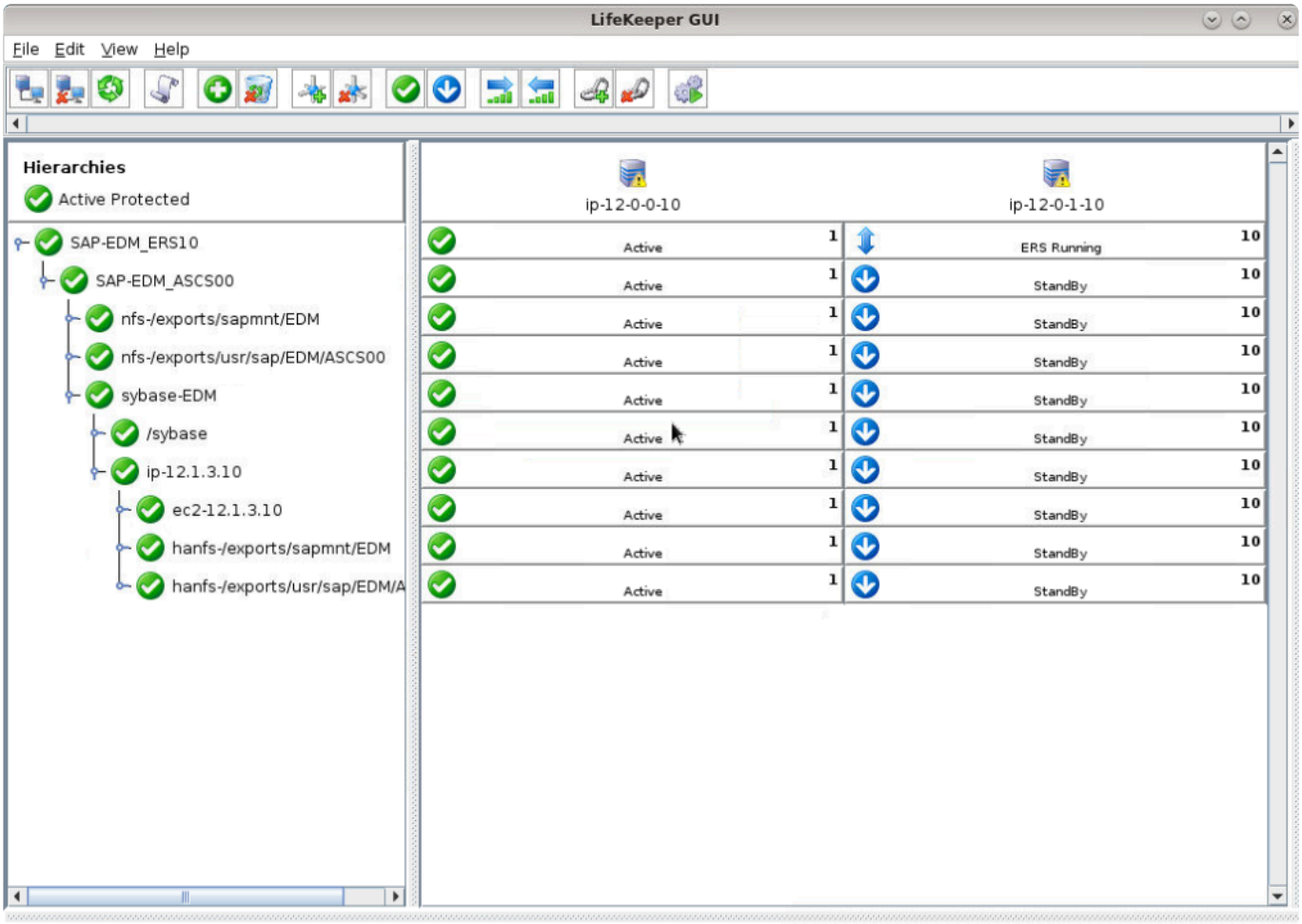
重要: SAP Recovery Kit – 既知の問題と制限 を参照してください。

ERSリソースタイプの設計と実装は、SPS-L 9.4.0で変更されました。このページでは、バージョン9.4.0より前の実装とバージョン9.4.0以降の実装の違い、システムに存在するバージョンの判別方法、および新しいリソースタイプへのアップグレード方法について説明します。

## SPS-L 9.4.0より前（9.3.2以前）のERSリソース

9.4.0より前（9.3.2以前）のバージョンのSPS-Lでは、ERSリソースは、ロックテーブルの冗長性を提供するセントラルサービス（ASCS/SCS）リソースに依存するSAP階層の最上位に位置するように設計されていました。

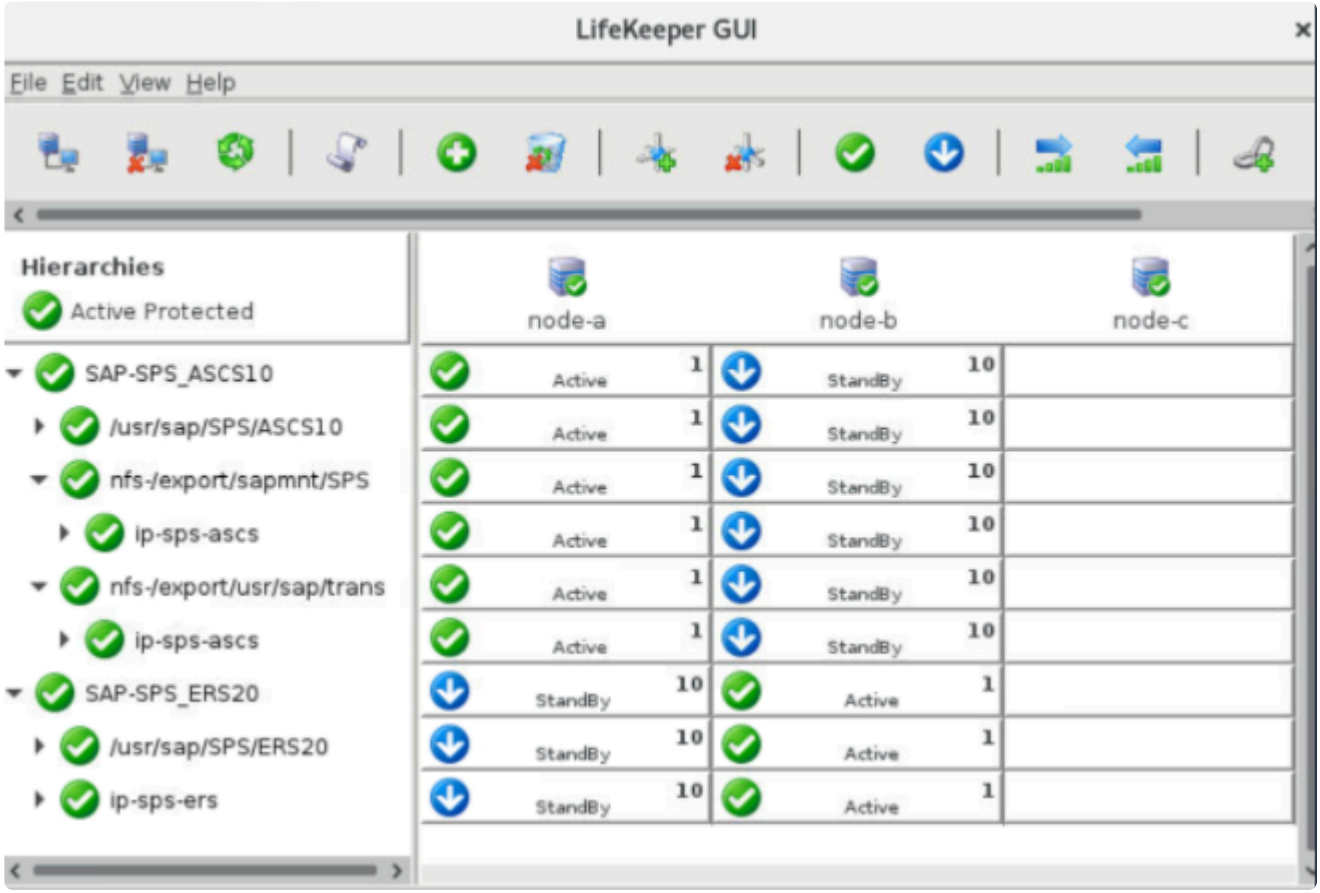
このリソースの動作は、バックアップノード（ERSリソースがLifeKeeper GUIでスタンバイとしてリストされていた場所）でERSインスタンスを開始するように設計されていました。SAP階層のスイッチオーバーまたはフェールオーバー時にASCS/SCSインスタンスがバックアップノードで起動され、そのシステムの共有メモリからロックテーブル（レプリケーションテーブル）のバックアップコピーを取得します。エンキューサーバーがロックテーブルを正常に取得すると、ERSインスタンスに自分自身を終了するよう通知する信号を送信します。
ERSリソースがバックアップノードのLifeKeeperでアクティブ（ISP）になると、ERSインスタンスは、利用可能になったときに元のプライマリノードで起動します。その時点で、レプリケーションサーバーはエンキューサーバーに再接続し、ロックテーブルのレプリケーションを再開します。ERSインスタンスがバックアップノードで実行されている場合、LifeKeeper GUIのステータスは「StandBy」から「ERS Running」に変わります。




## SPS-L 9.4.0以降のERSリソース

SPS-L 9.4.0以降では、ERSリソースは独自の独立した階層で動作するように再設計されました。

独立階層のERSv1リソース




仮想IPと高可用性依存ファイルシステムを備えた独立階層のERSv2リソース

この新しいERSリソース設計は、2ノードクラスターのERSv1インスタンスと、任意の数のノードを持つクラスターのERSv2インスタンスをサポートします。このリソースタイプを使用すると、ERSインスタンスは、LifeKeeperが現在Active（ISP）になっているノードと同じノードで開始されます。ERSv2階層（依存する仮想IPおよびファイルシステムリソースを含む）が対応するセントラルサービスリソース階層とは独立してフェールオーバーできるようにするために、設計を変更しました。

対応するセントラルサービスリソースと同じノードでERSリソースがActive（ISP）にならないようにする

ために、この新しいERSリソースタイプは、各quickCheck間隔（デフォルト：2分）で次のことを確認します。

1. ERSリソースが、対応するセントラルサービスリソースと同じノードでActiveになっている
2. ロックテーブルのレプリケーションは、エンキューサーバーとレプリケーションサーバーの間で同期している
3. ERS階層内のすべてのリソースを正常に再配置できるクラスター内に、使用可能な別のノードが存在する

これら3つの条件がすべて満たされている場合、クラスターノード間でエンキューサーバーのロックテーブルデータの冗長性を提供するために、LifeKeeperはERS階層を別のクラスターノードに自動的に再配置します。LifeKeeperでフラグ「sap_no_ers_relocation_<ERSタグ>」を設定すると、この自動再配置動作を無効にできます。これは、次のコマンドで実行できます（SAP-EXM_ERS12はERSリソースのタグの例です）。

```
/opt/LifeKeeper/bin/flg_create -f "sap_no_ers_relocation_SAP-EXM_ERS12"
```

このフラグを作成しても、ERSインスタンスのローカルリカバリーの失敗によるフェールオーバーは無効になりません。

**NFSを使用する場合の注意点：** この設計により、sapmntファイルシステム上のERSリソースのLifeKeeper依存関係が削除されるため、調整可能な値SAP_NFS_CHECK_DIRSを適切に使用して、sapmnt NFS共有にアクセスできないことに起因するアクションスクリプトのハングを防止する必要があります。詳細については、NFSの考慮事項へのリンク を参照してください。

## 階層に存在するERSリソースタイプの確認方法

既存のERSリソースがLifeKeeperのどのバージョンで作成されたかわからない場合は、次のコマンドを実行します（<ERSタグ>をERSリソースのタグ名に置き換えてください）。

```
/opt/LifeKeeper/bin/ins_list -f: -t <ERS Tag> | cut -d: -f6
```

このコマンドの出力は次のようになります。

EXMERS12sap2/sapmntFULLFULL5

- 出力の最後の桁が2の場合、このリソースはSPS-L 9.3.2以前で作成されています。このタイプのリソースは、2ノードクラスターのERSv1インスタンスを表すためにのみ使用でき、SAP階層の最上位に位置し、その下の対応する中央サービス（ASCS/SCS）インスタンスと依存関係を持つ必要があります。独立した階層で動作する新しい設計に切り替える方法については、以下の **ERSリソースタイプのアップグレード** セクションを参照してください。

- 出力の最後の桁が5の場合、このリソースはSPS-L 9.4.0以降で作成されています。このタイプのリソースを使用して、2ノードクラスターのERSv1インスタンスまたは任意の数のノードを持つクラスターのERSv2インスタンスを表すことができます。これは、対応するセントラルサービスリソースから独立した階層で動作する必要があります。

## ERSリソースタイプのアップグレード

SPS-L 9.4.0以降で新しいERSリソースタイプにアップグレードするには、次の手順を実行してください。

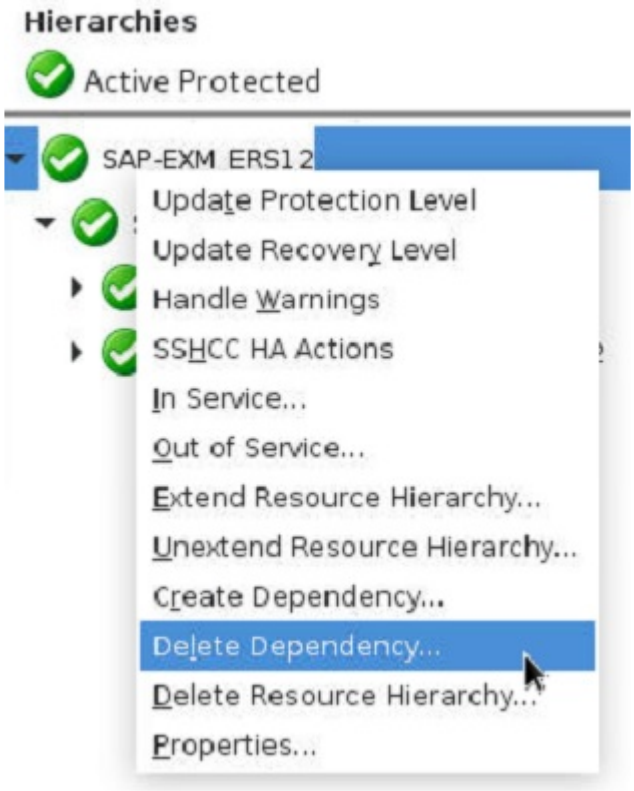
1. アップグレードプロセスを試行する前に、次のコマンドを実行して、すべてのクラスターノードでLifeKeeper階層のバックアップを作成します。

```
/opt/LifeKeeper/bin/lkbackup -c --cluster
```

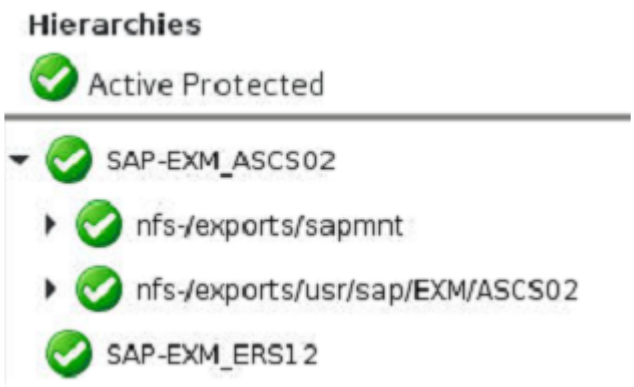
   - 作成中に何か間違えた場合は、すべてのノードでLifeKeeperを停止して次のコマンドを実行することにより、保存された階層構成を復元できます。

```
/opt/LifeKeeper/bin/lkbackup -x --cluster
```

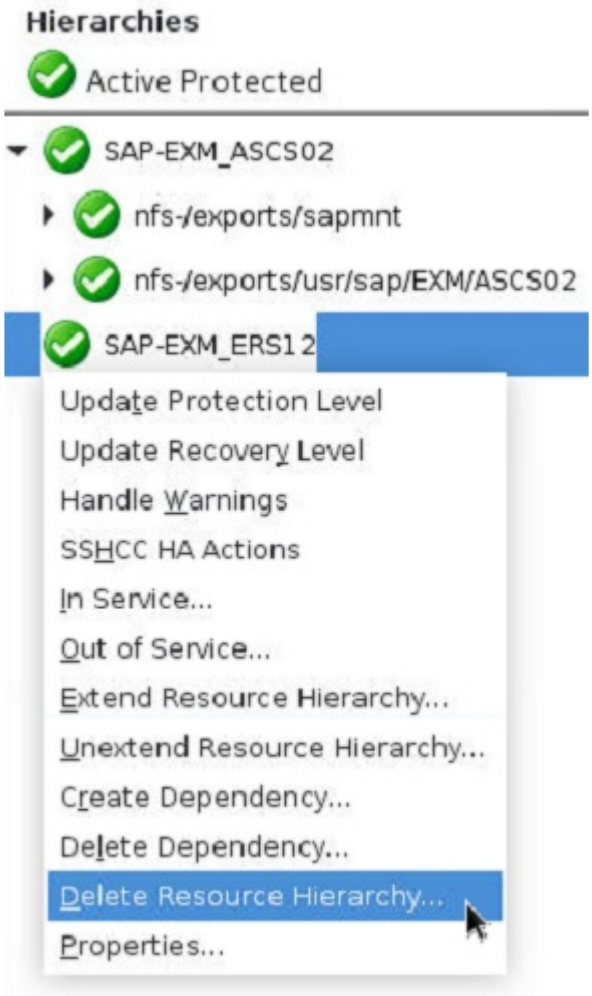
2. LifeKeeper階層パネルでERSリソースを右クリックし、 **[Delete Dependency...]** を選択します。




3. ERSリソースのすべての依存関係を削除します。すべての依存関係を削除するには、 **[Delete Dependency...]** を何度か選択する必要がある場合があります。依存関係を削除すると、次の図に示すように、ERSリソースは子の依存関係やそれに依存する他のリソースを持たず、単独の階層に存在するようになります。




4. LifeKeeperの階層パネルでERSリソースを右クリックし、 **[Delete Resource Hierarchy...]** を選択します。 **ターゲットサーバー** として任意のサーバーを選択し、 **[Next]** をクリックします。 **[Delete]** をクリックし、すべてのノードのERSリソースを削除します。 **警告:** 前のステップでERSリソースとSAP階層内の他のリソース間のすべての依存関係が正常に削除できなかった場合、 このステップによって**SAP階層全体が削除される可能性があります。**




5. ERSリソースをすべてのノードで正常に削除したら、「SAPのインストール → SAP階層の作成 」の「ERSリソースの作成」セクションの指示に従って、新しいERSリソースを作成し、目的のクラスターノードに拡張します。

# 6.16.8.11. ENSAv1からENSAv2へのアップグレード

スタンドアロンエンキューサーバー バージョン1からスタンドアロンエンキューサーバー バージョンバージョン2にアップグレードするには、まずSAPカーネルバージョンがENSAv2をサポートしていることを確認してから、次の手順を実行します。

1. デフォルトプロファイル（通常は /usr/sap/<SID>/SYS/profile/DEFAULT.PFL にあります）で以下のパラメーターを設定します。これらのパラメーターは、すべてのインスタンスで同じにする必要があります。

   - enq/enable=TRUE
     enq/serverhost=<
   ASCS instance host>
   enq/serverinst=<ASCS instance number>
   enque/process_location=REMOTESA

2. ASCSインスタンスプロファイル（通常 /usr/sap/<SID>/SYS/profile/<SID>_ASCS<No>_<VIP> にあります）で、次のパラメーターを設定します。

   - _ENQ = enq.sap$(
   SAPSYSTEMNAME)_$(INSTANCE_NAME)
   Execute_01 = local rm -f $(_ENQ)
   Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enq_server$(FT_EXE)
   Start_Program_01 = local $(_ENQ) pf=$(_PF)

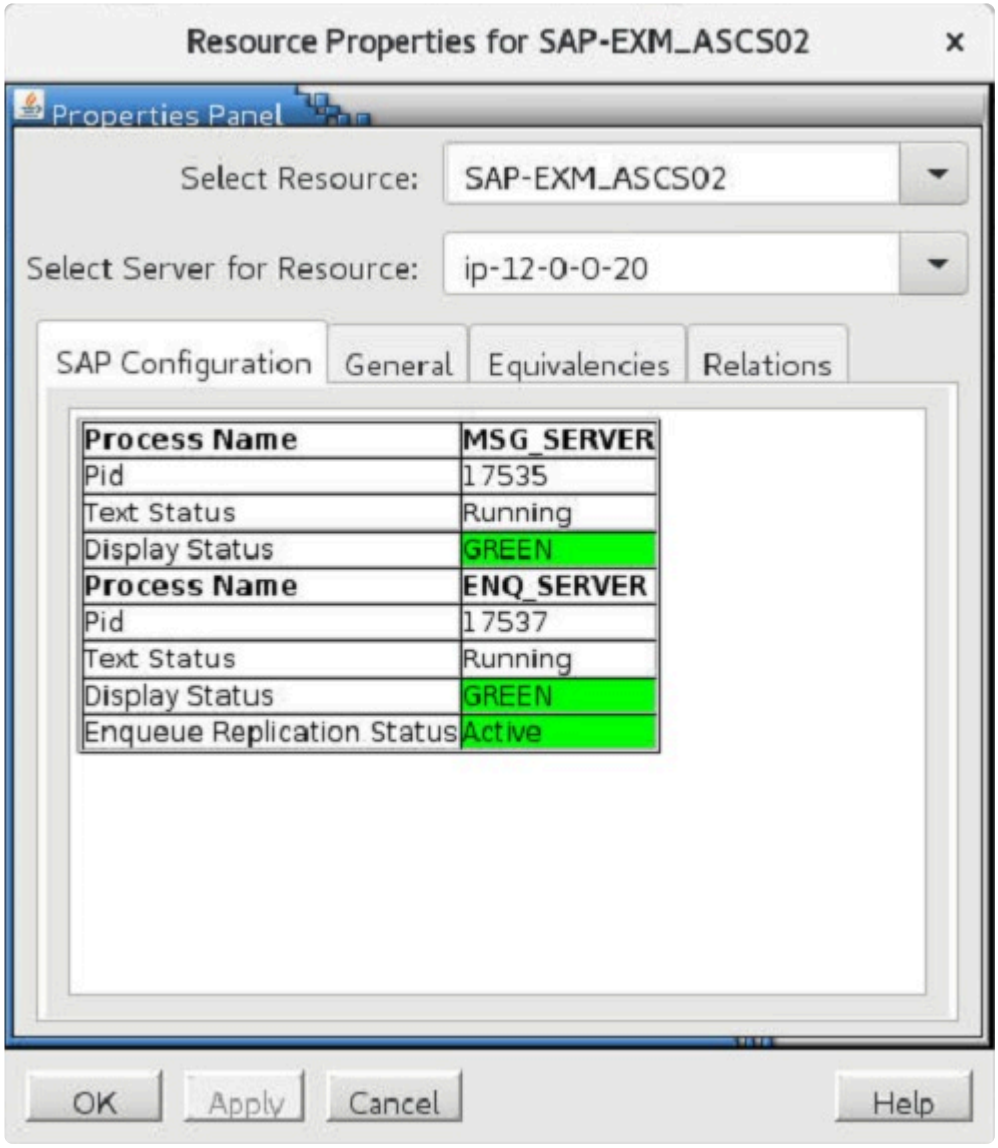
   **注記:** Execute_* および Start_Program_* パラメーターに使用される番号は、このプロファイルでそのパラメーターにまだ使用されていない最初の番号である必要があります。

3. デフォルトおよびASCSインスタンスプロファイルでパラメーターを設定した後、次のコマンドを実行してASCSインスタンスのSAP Start Serviceを再起動します（<sid>を小文字のSAP SIDに、<SID>を大文字のSAP SIDに置き換えます）。

   - su - <sid>adm -c "sapcontrol -nr <
   ASCS Inst#> -function RestartService <SID>"

4. **ASCS**リソースが拡張されているすべてのクラスターノードで、 ファイル/opt/LifeKeeper/subsys/appsuite/resources/sap/INFO_<ASCS Tag> を編集し、SAPENQ_VERSION=2であることを確認します。INFOファイルにSAPENQ_VERSIONの値が含まれていない場合、行「SAPENQ_VERSION=2」を追加する必要があります。

5. SAPシステムを再起動します。ASCSインスタンスが再起動すると、enserver（ENSAv1）プロセスではなくenq_server（ENSAv2）プロセスが使用されます。これは、LifeKeeperでASCSリソースを右クリックし、 **[Properties...]** を選択することで確認できます。

- 


注記： 対応するERSリソースのERSv1からERSv2へのアップグレードが完了するまで、システムのエンキューレプリケーションステータスにはActiveと表示されません。詳細については、ERSv1からERSv2へのアップグレード を参照してください。

# 6.16.8.12. ERSv1からERSv2へのアップグレード

エンキューレプリケーションサーバー バージョン1からエンキューレプリケーションサーバー バージョン2にアップグレードするには、まずSAPカーネルバージョンがERSv2をサポートしていることを確認してから、次の手順を実行します。

1. ENSAv1からENSAv2へのアップグレード の指示に従って、ENSAv2を使用するようにASCSインスタンスをアップグレードします。SAPでは混合バージョン構成はサポートしていないため、同じバージョンのスタンドアロンエンキューサーバーおよびエンキューレプリケーションサーバーを使用する必要があります。詳細については、ENSAv2/ERSv2のSAPドキュメントを参照してください。

2. ERSインスタンスの仮想IPと共有ファイルシステムをセットアップします。またLifeKeeperで、対応する仮想IPおよびファイルシステムのリソース階層を作成します。これらは、ERSリソースの再作成中にステップ9で使用します。 **注記：** これらのLifeKeeperリソースは、対応するASCSインスタンスが現在Standby（OSU）であるノード上でActive（ISP）である必要があります。

3. デフォルトプロファイル（通常 /usr/sap/<SID>/SYS/profile/DEFAULT.PFL にあります）で以下のパラメーターを設定します。これらのパラメーターは、すべてのインスタンスで同じにする必要があります。

   - enq/enable=TRUE
     enq/serverhost=<
   ASCS instance host>
   enq/serverinst=<ASCS instance number>
   enq/replicatorhost=<ERS instance host>
   enq/replicatorinst=<ERS instance number>
   enque/process_location=REMOTESA

4. ASCSインスタンスプロファイル（通常 /usr/sap/<SID>/SYS/profile/<SID>_ASCS<No>_<VIP> にあります）で、次のパラメーターを設定します。

   - enq/server/replication/enable = true
     _ENQ = enq.sap$(
   SAPSYSTEMNAME)_$(INSTANCE_NAME)
   Execute_01 = local rm -f $(_ENQ)
   Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enq_server$(FT_EXE) $(_ENQ)
   Start_Program_01 = local $(_ENQ) pf=$(_PF)

**注記：** Execute_* および Start_Program_* パラメーターに使用される番号は、このプロファイルでそのパラメーターにまだ使用されていない最初の番号である必要があります。

5. ASCSインスタンスプロファイル（通常 /usr/sap/<SID>/SYS/profile/<SID>_ERS<No>_<VIP> にあります）で、次のパラメーターを設定します。

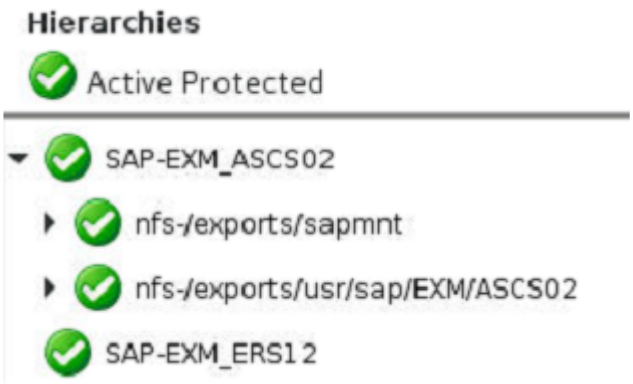
- _ENQR = enqr.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
  Execute_01 = local rm -f $(_ENQR)
  Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enq_replicator$(FT_EXE) $(_ENQR)
  Start_Program_00 = local $(_ENQR) pf=$(_PF)

注記： Execute_* および Start_Program_* パラメーターに使用される番号は、このプロファイルでそのパラメーターにまだ使用されていない最初の番号である必要があります。

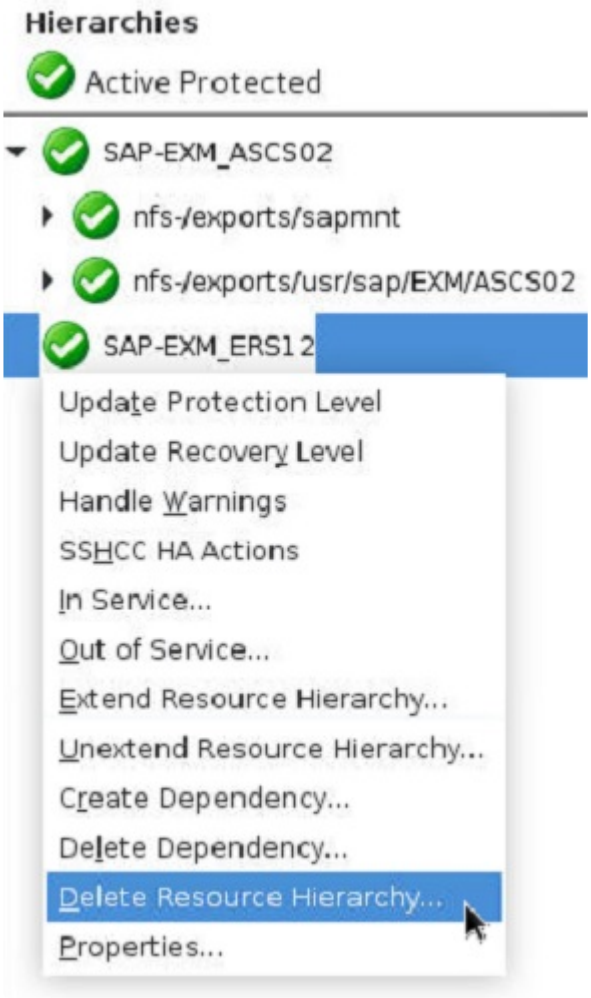
6. デフォルト、ASCS、およびERSインスタンスプロファイルのパラメーターを設定した後、次のコマンドを実行して、ASCSおよびERSインスタンスのSAP Start Serviceを再起動します（<sid>を小文字のSAP SIDに、<SID>を大文字のSAP SIDに置き換えます）。

- su – <sid >adm -c "sapcontrol -host <ASCS VIP> -nr <ASCS Inst#> -function RestartService <SID>"
  su – <sid >adm -c "sapcontrol -host <ERS VIP> -nr <ERS Inst#> -function RestartService <SID>"

7. アップグレードする必要のあるLifeKeeperに既存のERSリソースがない場合は、手順9に進んでください。それ以外の場合は、リソースを右クリックして **[Delete Dependency....]** を選択し、LifeKeeperのERSリソースのすべての依存関係を削除します。すべての依存関係を削除するには、**[Delete Dependency...]** を何度か選択する必要がある場合があります。依存関係を削除すると、次の図に示すように、ERSリソースは子の依存関係やそれに依存する他のリソースを持たず、単独の階層に存在するようになります。

- 

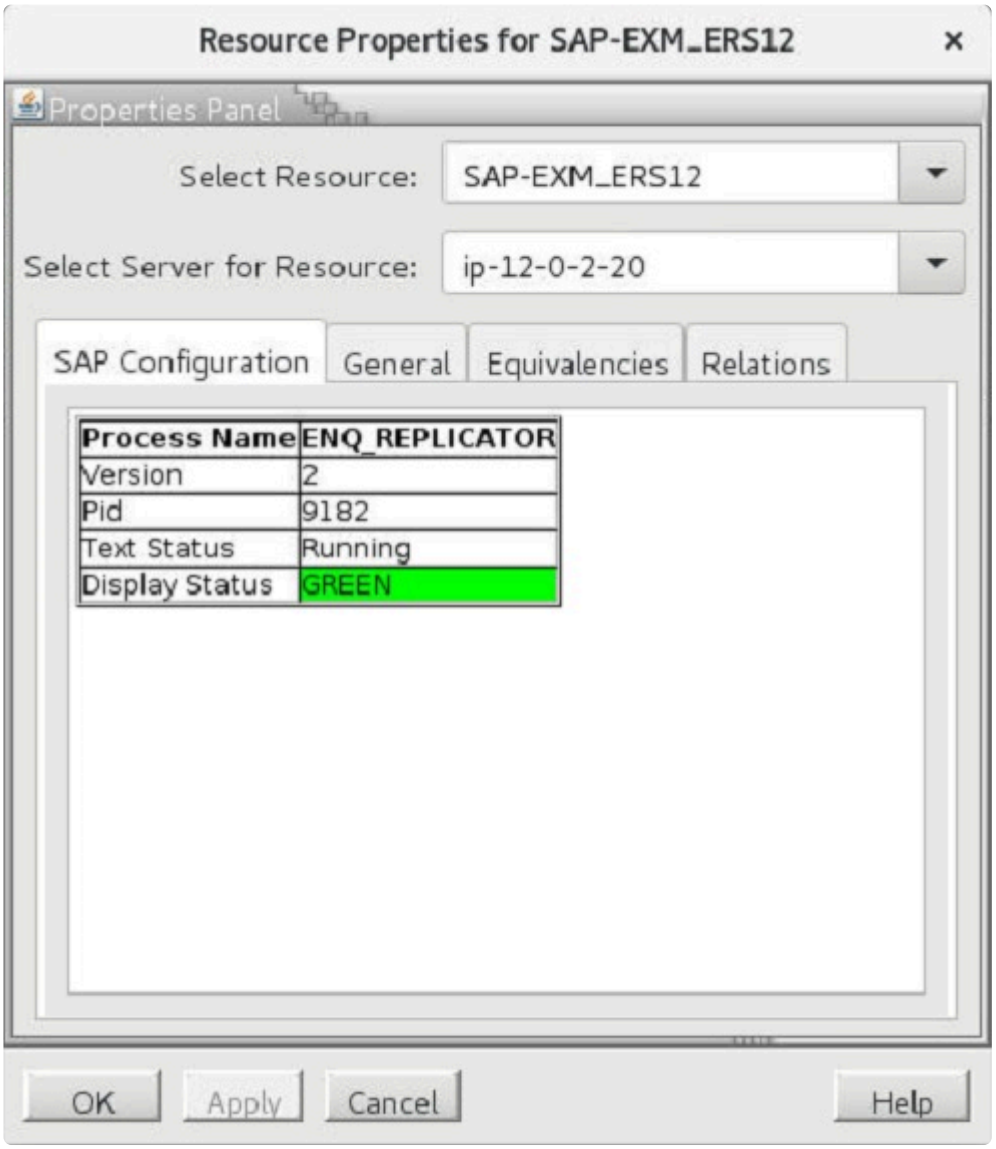

8. LifeKeeperの階層パネルでERSリソースを右クリックし、 **[Delete Resource Hierarchy...]** を選択します。 **ターゲットサーバー** として任意のサーバーを選択し、 **[Next]** をクリックします。 **[Delete]** をクリックして、すべてのノードのERSリソースを削除します。 **警告：** 前のステップでERSリソースとSAP階層内の他のリソース間のすべての依存関係が正常に削除できなかった場合、このステップによって**SAP階層全体が削除される可能性があります。**




9. すべてのノードでERSリソースを正常に削除したら、「 SAPインストール→SAP階層の作成 」の「ERSリソースの作成」セクションの指示に従って、新しいERSリソースを作成し、目的のクラスターノードに拡張します。ERSリソースの作成中に、手順2で作成した仮想IPおよび依存ファイルシステムリソースを選択します（該当する場合）。選択すると、これらのリソースはERS階層に子の依存関係として自動的に追加されます。

10. **ERS**リソースが拡張されたすべてのクラスターノードで、 ファイル/opt/LifeKeeper/subsys/appsuite/resources/sap/INFO_<ERS Tag> を編集し、SAPENQREP_VERSION=2であることを確認します。INFOファイルにSAPENQREP_VERSIONの値が含まれていない場合、行「SAPENQREP_VERSION=2」を追加する必要があります。

11. SAPシステムを再起動します。ERSインスタンスが再起動すると、enrepserver（ERSv1）プロセスではなく、enq_replicator（ERSv2）プロセスが使用されます。これは、LifeKeeperでERSリソースを右クリックし、 **[Properties…]** を選択することで確認できます。



# 6.16.9. トラブルシューティング

本セクションでは、SPS SAP リソース階層の作成 / 拡張中、およびリソースの削除 / リストア中に表示される可能性のあるメッセージの一覧を示します。また、必要に応じて、エラーの原因や、エラー状態を解決するために必要な処置について説明を付け加えます。

他の SPS コンポーネントからメッセージが出されることもあります。そのような場合は、メッセージカタログ を参照してください。メッセージカタログには、操作、管理、GUI など、SIOS Protection Suite for Linux を使用しているときに遭遇する可能性がある、すべてのエラーコードが列挙されています。また、エラーコードの原因に関する追加の説明や、問題解決のために必要な処置についても、必要に応じて記載されています。この完全なリストを検索すると、受信したエラーコードを見つけることができます。また、関連する SPS コンポーネントの個別のメッセージカタログに直接アクセスすることもできます。

このセクションのメッセージは、次のトピックに分類されています。

SPS SAP メッセージ

ERS インスタンスの変更

ASCS + ERS Restart_Program パラメーター

階層削除時のエラー

フェイルオーバ中または In-Service 処理中の SAP エラーメッセージ

SAP インストールエラー

sapinit のトラブルシューティング

LifeKeeper ログファイルに出現する「tset」エラー

# 6.16.9.1. ERSプロファイルでAutostartを無効にする

## 症状:

ERS インスタンスのステータスチェックを行うと、選択したインスタンスで sapstart が発生する。

ERS インスタンスは、両方のシステムで常に実行されている。

## 原因:

ERS インスタンスのプロファイルに *Autostart=1* が設定されている場合、特定の `sapcontrol` 呼び出しにより、実行中のコマンドの一部としてインスタンスが起動されます。

## 処置:

クラスタ内で実行中の ERS インスタンスを停止し、ERS インスタンスのプロファイルを変更して *Autostart=0* に設定します。

このプロファイルの変更を有効にするには、sapstartsrvの再起動が必要です。これを行うには、次のコマンドを実行します（を小文字のSAP SIDに置き換え、を大文字のSAP SIDに置き換えてください）。

```
su - <sid>adm -c “sapcontrol -nr <ERS Inst#> -function RestartService <SID>”
```

✻ autostart の値を変更した場合は、変更を反映させるため再起動が必要になります。

# 6.16.9.2. ASCS + ERS Restart_Program パラメーター

## 症状：

エンキューサーバーまたはエンキューレプリケータープロセスが、プライマリーサーバーとバックアップサーバーの両方で同時に実行されている。

## 原因：

ERSインスタンスプロファイルがStart_Programではなく Restart_Programでインスタンスを開始するように設定されている場合、ERSインスタンスが何らかの理由で終了した際、sapstartプロセスはERSインスタンスを自動的に再起動します。
これにより予期しない動作が発生し、フェールオーバーまたはスイッチオーバー中にエンキューサーバーのロックテーブルが失われる可能性があります。

## 処置：

エンキューサーバーおよびエンキューレプリケータープロセスの起動時に、Restart_Programではなく Start_Programを使用するよう、ASCSおよびERSインスタンスプロファイルのパラメーターを変更してください。詳細については、「 ASCSおよびERSインスタンスプロファイル設定の変更 」を参照してください。

ASCSおよびERSプロファイルが更新され、Restart_Programの代わりにStart_Programを使用し、SAP Start Serviceが再起動されたら、システムを再起動し、更新されたプロファイルが読み込まれ、キャッシュが有効になっていないことを確認することを推奨します。

成功した場合、ERSインスタンスは、ASCS/SCSインスタンスが現在実行されていないバックアップシステムでのみ実行する必要があります。

詳細については、 SAPのインストール を参照してください。

# 6.16.9.3. 階層削除時のエラー

## 症状:

使用中のファイルシステムの削除に失敗する。

## 原因:

1. 作成または拡張後に、 **[Resource Protection Level]** が **[Basic]** または **[Minimum]** に設定された場合。リソース保護レベルが Basic または Minimum に設定されている場合、削除処理時に SAP リソース階層が停止されません。これにより、削除が呼び出されても、そのインスタンスのプロセスは実行中のままになります。また、プロセスが保護されたファイルシステムにアクセス中の場合、LifeKeeper はファイルシステムをアンマウントできないことがあります。

2. レプリケートされていない enqueue リソースの **[Resource Protection Level]** が **[Standard]** に設定された場合。リソース保護レベルが Standard に設定されている場合、削除処理時に SAP リソース階層が停止されません。これにより、削除が呼び出されても、そのインスタンスのプロセスは実行中のままになります。また、プロセスが保護されたファイルシステムにアクセス中の場合、LifeKeeper はファイルシステムをアンマウントできないことがあります。

## 処置:

1. **[Basic]** と **[Minimum]** の設定は、リソースを一時的なメンテナンスモードにするときに使用します。継続的な保護レベルとして使用すべきではありません。問題のリソースが Basic または Minimum を継続的な保護レベルとして必要とする場合、そのインスタンスはローカルストレージを使用するように設定するか、リソース階層全体をローカル NFS マウント向けの LifeKeeper NAS Recovery Kit を使用せずに設定する必要があります。

2. **[Standard]** 設定は、レプリケートされた enqueue リソースでのみ使用します。

**注記:** [Standard] は、SPS-L 9.3.2以前で作成され、対応するセントラルサービスリソースに依存するSAP階層の最上位に存在するERSリソースに使用されます。SPS-L 9.4.0以降で作成され、セントラルサービスリソースから独立した階層に存在するERSインスタンスは、保護レベルを [Full] に設定する必要があります。

# 6.16.9.4. SAP 階層のリストアエラー

## 現象:

依存するファイルシステムリソースを含まない SAP リソース階層を以前のホストノードで In-Service に戻そうとすると、SAP リソースの In-Service（リストア）が失敗します。
SAP インスタンスのトレースログに、インスタンスプロセスの1つに必要なポートが既に使用されていることを示す以下のようなメッセージが表示されます。 (この例は、 /usr/sap/<SID>/ASCS<##>/work/dev_ms.new から取得したものです）。

```
[Thr 140225937864512] ***LOG Q0I=> NiIBindSocket: bind (98: Address already in use) [/bas/781_REL/src/base/ni/nixxi.cpp 3946
[Thr 140225937864512] *** ERROR => NiIBindSocket: SiBind failed for hdl 1/sock 6 (SI_EPORT_INUSE/98; I4; ST; 0.0.0.0:3610) [nixxi.cpp 3946]
[Thr 140225937864512] *** ERROR => MsSCommInit: NiBuf2Listen(sapmsSHC) (rc=NIESERV_USED) [msxxserv.c 12838]
[Thr 140225937864512] *** ERROR => MsSInit: MsSCommInit [msxxserv.c 2732]
[Thr 140225937864512] *** ERROR => MsSInit failed, see dev_ms.new for details [msxxserv.c 7363]
```

## 原因:

リソースの保護レベル が、作成または拡張後に、 **Basic** 、 **Minimum** 、または **Standard** に設定される。

リソースの保護レベルが Basic、Minimum、または Standard に設定されている場合、SAP リソースを Out-of-Service にする際に、保護されている SAP インスタンスは停止されません。これにより、remove スクリプトが呼び出された後、そのインスタンスのプロセスは実行されたままになります。SAP リソース階層には依存するファイルシステムリソースが含まれていないため、SAP インスタンスが依存するファイルシステムは、SAP 階層がサービスを停止した後もマウントされたままになります。これは、SAP インスタンスプロセスが引き続き実行されることを意味しますが、もし SAP リソース階層に依存するファイルシステムリソースが含まれていた場合は、LifeKeeper が基盤となるファイルシステムをアンマウントするときに `fuser -k` コールで強制終了させます。

SAP インスタンスプロセスが実行されたままになっているサーバー上で LifeKeeper がリソースを In-Service に戻そうとすると、必要なポートが既に実行中のプロセスによって使用されているため、保護されたインスタンスを起動できません。

## 対応:

1. **[Basic]** および **[Minimum]** 設定は、リソースを一時的にメンテナンスモードにする場合にのみ使用してください。これらの設定は、継続的な保護レベルの設定として使用しないでください。

2. **[Standard]** 設定は、 レプリケートされた enqueue リソースにのみ使用できます。特に、SAP リソースに LifeKeeper の保護下にある依存ファイルシステムがない場合は使用しないでください。

> **注意: [Standard]** は、SPS-L 9.3.2 以前で作成され、対応するセントラルサービスリソースに依存する SAP 階層の最上位に存在する ERS リソースにも使用されます。SPS-L 9.4.0以降で作成され、セントラルサービスリソースから独立した階層に存在する ERS インスタンスは、保護レベルを **[Full]** に設定する必要があります。

3. この問題を解決するには、以下の手順に従ってください。
   a. SAP リソースが拡張されたクラスター内のすべてのサーバーを再起動します。再起動後、保護された SAP インスタンスのすべてのプロセスが停止し、必要なポートが利用可能になります。
   b. サーバーの1つで SAP リソース階層を In-Service にします。LifeKeeper GUI でリソースを右クリックし、[Update Protection Level] を選択して、SAP リソースの保護レベルを **[Full]** に変更します。

1. SAP リソースが拡張されたクラスター内の各サーバーに対して、ステップ（b）を繰り返します。

# 6.16.9.5. フェイルオーバ中または In Service の SAP エラーメッセージ

SAP のフェイルオーバ後、SAP ログにエラーメッセージが記録されます。エラーメッセージの多くは正常であり、無視できます。

## DB 障害発生時

**BVx: Work Process is in reconnect status** – このエラーメッセージは、単に、作業プロセスがデータベースへの接続を失ったため、再接続を試みていることを示しています。

**BVx: Work Process has left reconnect status** – これは実際にはエラーではありませんが、データベースが復帰し、プロセスがデータベースに再接続したことを示しています。

**その他のエラー** – データベースがダウンしている間、いくつものエラーがログに記録される可能性があります。

## CI の起動時

**E15: Buffer SCSA Already Exists** – このエラーメッセージは、実際にはエラーではありません。単に、以前に作成された共有メモリ領域が SAP が使用するシステムに見つかったことを示しています。

**E07: Error 00000 : 3No such process in Module rslgsmcc (071)** – *SAP ノート 7316 を参照* – 前回のシャットダウン中に、ロックが正しく解放されませんでした。このエラーメッセージは無視できます。

## LifeKeeper In Service

In Service の動作時に、LifeKeeper の [In Service] ダイアログに以下のメッセージが表示されることがあります。

```
error: permission denied on key 'net.unix.max_dgram_qlen'

error: permission denied on key 'kernel.cap-bound'
```

これらのエラーは、saposcol が起動されたときに発生するもので、無視できます *(SAP ノート 201144 を参照)*。

# 6.16.9.6. SAP インストールエラー

## tnsnames.ora または listener.ora ファイルの名前の誤り

### 原因:

Oracle データベースを使用する場合で、PAS バックアップサーバのインストールを実行する際に、SAP インストールプログラムにより *tnsnames.ora* または *listener.ora* ファイルのサーバ名が誤っているというエラーが出る場合、Oracle バイナリがローカルファイルシステムにインストールされていない可能性があります。

### 処置:

設定を正しく実行するためには、 */oracle/<SID>/920_<32 または 64>* の Oracle バイナリが各サーバのローカルファイルシステム上にインストールされている必要があります。

# 6.16.9.7. sapinit のトラブルシューティング

## 症状:

init スクリプトにより開始された sapstartsrv プロセスまたは追加の SAP インスタンスプロセスが正しく動作しない、または LifeKeeper スタンバイノードでプロセスを実行しようとする。

## 原因:

SAP は、SAP インスタンスをローカルノードで自動的に起動するために init スクリプトを提供しています。リソースが LifeKeeper の保護に追加されているとき、init スクリプト (`sapinit`) が、現在のノードで実行すべきでない SAP インスタンスプロセスを起動しようとすることがあります。

## 処置:

`sapinit` スクリプトを無効化するか、`sapinit` を変更して LifeKeeper 保護下のインスタンスをスキップするように設定します。この動作を無効化するには、ユーザは `sapinit` を停止する必要があります *(例: `/etc/init.d/sapinit stop`)*。sapinit スクリプトも、chkconfig または同様のツール *(例: `chkconfig sapinit off`)* を使用して無効化する必要があります。

# 6.16.9.8. LifeKeeper ログファイルに出現する「tset」エラー

## 原因:

SAP と Database Recovery Kit によって使用された su コマンドにより、LK ログに次のような「tset」エラーメッセージが出力されます。

```
tset: standard error: Invalid argument
```

このエラーは、SAP 管理者およびデータベースユーザのホームディレクトリにあるプロファイルファイルの 1 つによって出力されるもので、非インタラクティブシェルのみに存在します。

## 処置:

データベースユーザと SAP 管理者に C シェルを使用している場合、これらのユーザのホームディレクトリの *.sapenv_<hostname>.csh* に次の行を追加してください。このコードは、「tset」を実行するかどうかを決定する次のコードを囲むように追加する必要があります。

```
if ( $?prompt ) then
    tty -s
    if ( $status == 0) then
        .
        .
        .
    endif
endif
```

注記:「tty -s」から内側の「endif」までのコードは、既にファイル内に存在しています。

データベースユーザと SAP 管理者に Bash シェルを使用している場合、これらのユーザのホームディレクトリの *.sapenv_<hostname>.sh* に次の行を追加してください。

「tset」を実行するかどうかを決定するコードの前に、以下を追加します。

```
case $- in
```

***i*) INTERACTIVE ="yes";;**

```
        *) INTERACTIVE =“no”;;

    esac
```

「tset」を実行するかどうかを決定するコードを囲むように、以下を追加します。

```
            if [ $INTERACTIVE == “yes” ]; then

            tty -s

            if [ $? -eq 0 ]; then

                    .

                    .

                    .

              fi

        fi
```

注記:「tty -s」から内側の「endif」までのコードは、既にファイル内に存在しています。

# 6.16.9.9. SAP SIOS HA Cluster Connector Errors

## Symptom:

When enabled, the SAP SIOS HA Cluster Connector fails to update the state of the SAP resource in LifeKeeper after the protected SAP instance is started or stopped outside of LifeKeeper (e.g., using sapcontrol, SAP Management Console, etc.). Errors similar to the following appear in the sapstartsrv.log file for the protected SAP instance when SAP HA Library debug logging is enabled (i.e., /service/halib_debug_level >= 1):

```
SAP HA Trace: Using tmp file /tmp/sapha-tmp-KKGWWV
SAP HA Trace: Fire system command /opt/LifeKeeper/lkadm/subsys/appsuite/sap/
bin/sap_sios_cluster_connector /opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/
sap_sios_cluster_connector lsr --out /tmp/sapha-tmp-KKGWWV --sid SPS --ino 10
SAP HA Trace: ClusterFileBuffer: <>SAP HA Trace: parseClusterFile Line: 0
SAP HA Trace: searchClusterFile: SPS:10 NOT found
SAP HA Trace: searchClusterFile: rc = 1
```

## Cause:

Due to a change made in the SAP HA Script Connector Library in April 2021 (see SAP Note 3018407), the SAP SIOS HA Cluster Connector is unable to successfully return information about the state of cluster resources to the HA Script Connector Library on servers running LifeKeeper v9.6.0 or earlier and the following affected SAP kernel versions:

| SAP Kernel | Patch Level |
|---|---|
| SAP Kernel 7.22 64-bit | PL >= 1021 |
| SAP Kernel 7.22 64-bit unicode | PL >= 1021 |
| SAP Kernel 7.22 EXT 64-bit | PL >= 1021 |
| SAP Kernel 7.22 EXT 64-bit unicode | PL >= 1021 |
| SAP Kernel 7.22 EX2 64-bit | PL >= 1021 |
| SAP Kernel 7.22 EX2 64-bit unicode | PL >= 1021 |
| SAP Kernel 7.49 64-bit | PL >= 942 |
| SAP Kernel 7.49 64-bit unicode | PL >= 942 |
| SAP Kernel 7.53 64-bit | PL >= 810 |
| SAP Kernel 7.53 64-bit unicode | PL >= 810 |
| SAP Kernel 7.77 64-bit unicode | PL >= 324 |

| | |
|---|---|
| SAP Kernel 7.81 64-bit unicode | PL >= 110 |
| SAP Kernel 7.85 64-bit unicode | All |

## Action:

On an affected server, the protected SAP instance should be started or stopped only through LifeKeeper (e.g., using the LifeKeeper GUI or LifeKeeper Command Line Interface). Otherwise, when LifeKeeper detects that the protected instance has been stopped it will take action to either restart the instance locally or initiate a resource failover to the standby server.

The issue can be resolved by either:
(i) Upgrading to a version of LifeKeeper later than v9.6.0, or
(ii) Installing the hotfix for LifeKeeper v9.5.2 or v9.6.0 located here:
http://ftp.us.sios.com/pickup/SAP-rpm-PL-13003-9.6.0

# 6.16.10. LifeKeeper が保護するシステムのメンテナンス

LifeKeeper が保護するサーバーをシャットダウンしてメンテナンスを行うときには、メンテナンスの前に、バックアップサーバーでシステムのリソース階層を in service にする必要があります。このプロセスにより、メンテナンスが必要なシステム上にある共有ディスクの動作がすべて停止します。

記載の順序で、以下の操作を実行してください。 *Server A* はメンテナンスが必要なプライマリシステム、*Server B* はバックアップサーバーです。

1. ***Server B* で階層を in service にしてください。** バックアップの *Server B* で、LifeKeeper の GUI を使用して、現在 *Server A* でサービス中のリソース階層を in service にします。これにより、LifeKeeper の保護下にある共有ディスクに存在している *Server A* のファイルシステムがアンマウントされます。詳細については、リソースを In Service にする を参照してください。

2. ***Server A* で LifeKeeper を停止してください。** コマンド `$LKROOT/bin/lkcli stop` を使用して、LifeKeeper を停止します。

3. **Linux をシャットダウンし、*Server A* の電源をオフにしてください。** *Server A* のLinux オペレーティングシステムをシャットダウンし、サーバーの電源をオフにします。

4. **メンテナンスを実行してください 。** *Server A* で必要なメンテナンスを実行します。

5. ***Server A* の電源をオンにし、Linux を再起動してください 。** *Server A* の電源をオンにし、次に Linux オペレーティングシステムを再起動します。

6. ***Server A* で LifeKeeper を開始してください。** コマンド `$LKROOT/bin/lkcli start` を使用して、LifeKeeper を開始します。

7. **必要に応じて、*Server A* で階層を in-service にしてください。** *Server A* で LifeKeeper の GUI を使用して、 *Server B* にスイッチオーバしていたすべてのリソース階層を in service にしてください。

# 6.16.10.1. SAP メンテナンスモード

## SAP ソフトウェアのアップデート中に LifeKeeper の保護下にあるリソースをメンテナンスモードにする

LifeKeeper には、SAP 階層内のリソースを メンテナンスモード にする機能があります。これにより、ユーザーはバックアップサーバ上で階層を In-service にしなくても、SAP ソフトウェアをアップグレードできます。

### SAP 階層でメンテナンスモードが有効になっている間

- 階層内のすべてのリソースに対して、リソースのヘルスモニタリング、ローカルリカバリ、およびフェイルオーバーが無効になります。

  **注記：** アップグレードプロセス中のデータ破損の可能性を最小限に抑えるために、LifeKeeper は SCSI リザベーションエラーを検出して対処します。これにより、リソース階層のフェイルオーバーが発生したり、SCSI リザベーションが失われた場合にシステムが停止することがあります。SCSI デバイスにアクセスできない場合の LifeKeeper の動作を変更するには、Coreパラメータ一覧 の SCSIERROR パラメータを参照してください。

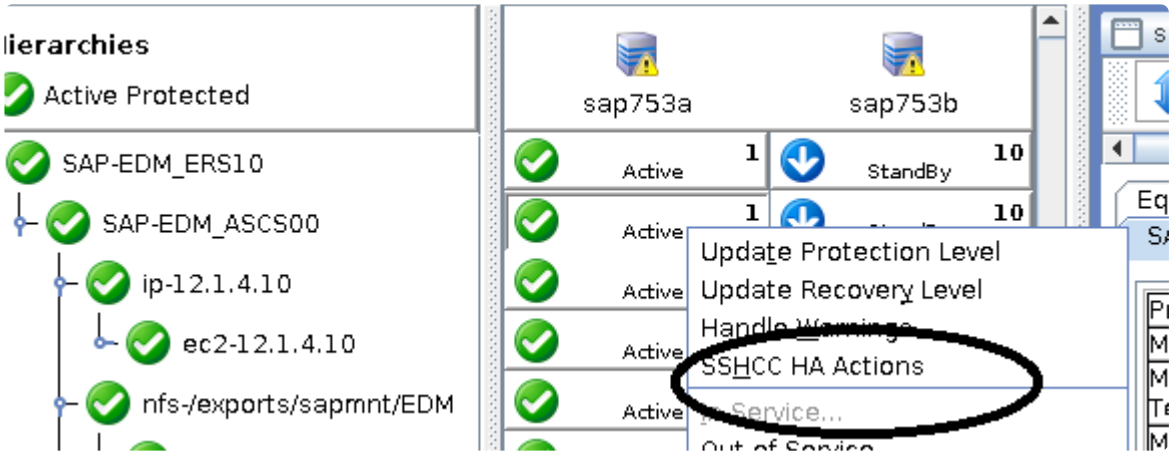
  **注記：** Quorum/Witness 機能を使用している場合、LifeKeeper はデフォルトでは、Quorum を失うとノードを停止します。これは、Quorum の喪失に対処する必要がある重要なアプリケーション階層がある場合に役立ちますが、ネットワーク通信エラーが発生した場合に保守作業を実行している間に予期しない停止を引き起こす可能性があります。手動でQuorum/Witness 機能を無効/有効にする手順は、マニュアルの Quorum / Witness のセクションを参照してください。

- メンテナンスモードが無効になるか、ユーザーが手動でミラーを再開するまで、階層内の DataKeeper ミラーはすべて一時停止されます。

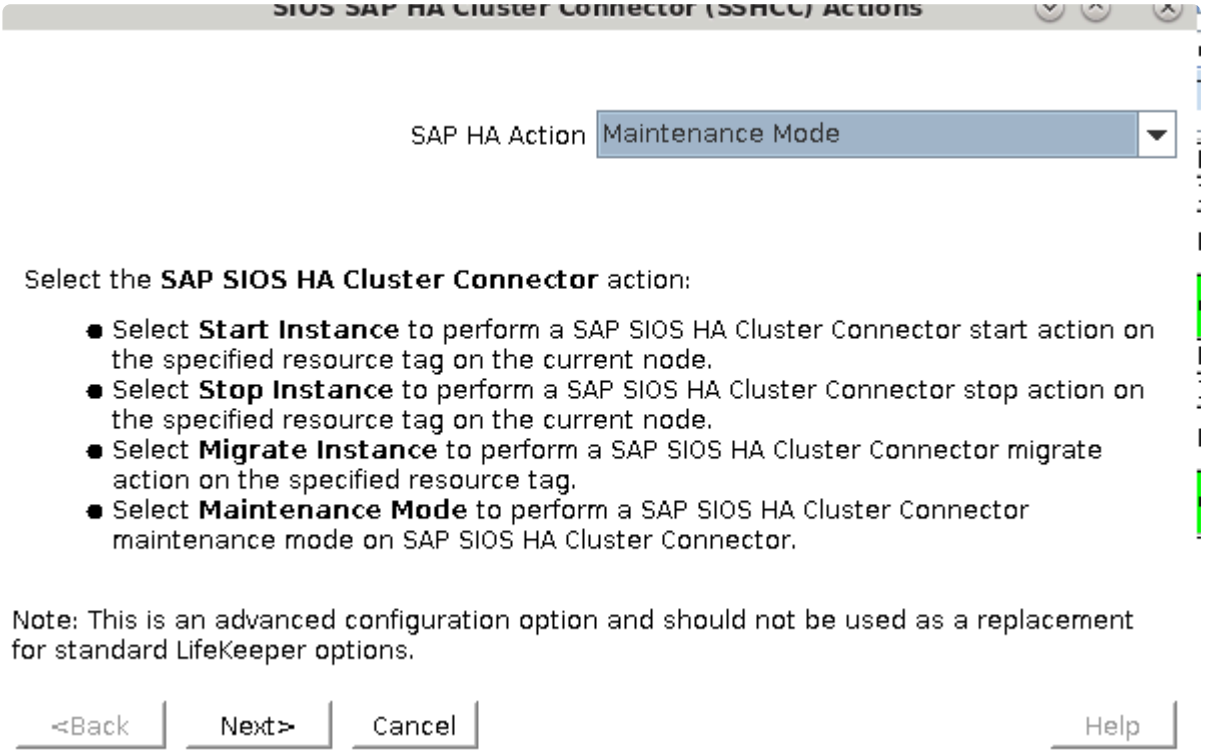
  **注記：** メンテナンスモードが階層に対して有効になっている間にミラーを手動で再開することは推奨しません。これは、ヘルスモニタリング、ローカルリカバリ、およびフェイルオーバーが無効になっているため、LifeKeeper はミラーの障害に対処できないからです。

SAP 階層をメンテナンスモードにするには、以下の手順を実行します。

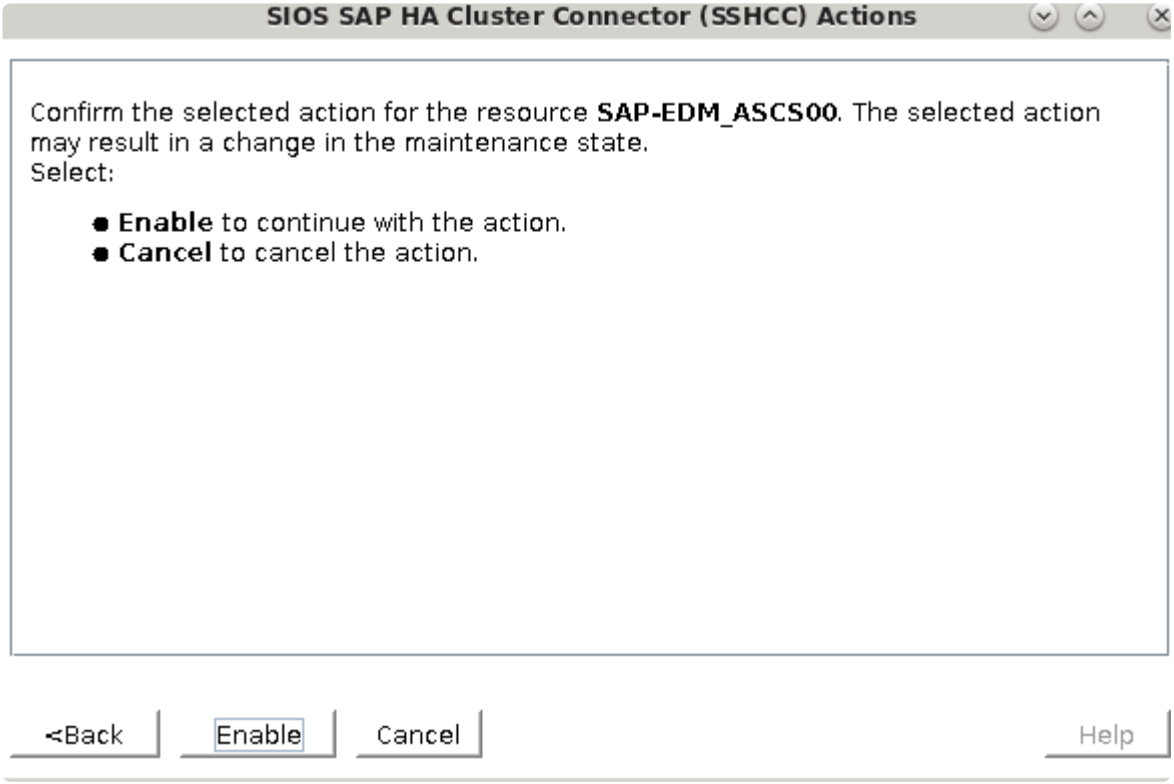
1. **LifeKeeper GUI** を **開きます。**
2. 現在In-serviceのノードの各階層で **SAP ASCS** または **SCS** リソースを **右クリック** し、 **[SSHCC HA Actions]** をクリックします。




3. ドロップダウンボックスで **[Maintenance Mode]** を選択します。




4. 選択した階層のすべてのリソースに対して、メンテナンスモード を **有効/無効** にするかどうかを選択します。この操作は、クラスタ内のすべてのノードで実行します。




注記： SAPカーネル7.49 PL 200以降では、SAP Update Managerを使用してSAPソフトウェアをメンテナンスモードにすると、対応するLifeKeeper SAP階層は、SAP SIOS HA Cluster ConnectorがSAPインスタンスに対してアクティブである場合、HAコネクタ関数HASetMaintenanceModeを介して自動的にメンテナンスモードになります。LifeKeeperのリソースをLifeKeeper GUI を使用して手動でメンテナンスモードにする場合、アップグレード前にSAP Update Manager のマニュアルを参照して、SAPソフトウェア自体をメンテナンスモードにする方法を決定する必要があります。

## コマンドラインから SAP メンテナンスモードを有効または無効にする

クラスター内の全てのサーバー上の全リソース階層に対して SAP メンテナンスモードを有効するには、LifeKeeper の保護下にある SAP リソースをホストしているサーバーのうち一つで以下のコマンドを実行してください。

```
sudo /opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/lk_maintenance_mode --mode=enable --cluster
```

オプションの --tag パラメーターはメンテナンスモードに設定する特定のリソース階層のみ指定するために使用されます。 例えば (A)SCS および ERS リソース階層のみメンテナンスモードにする場合は、以下のコマンドを LifeKeeper の保護下にある SAP リソースをホストするサーバーのうち一つで実行してください。

```
sudo /opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/lk_maintenance_mode --mode=enable --tag=<A/SCS Resource Tag>,<ERS Resource Tag> --cluster
```

クラスター内の全てのサーバー上の全リソース階層に対して SAP メンテナンスモードを無効にするには、以下のコマンドを LifeKeeper の保護下にある SAP リソースをホストするサーバーの一つで実行してください。

```
sudo /opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/lk_maintenance_mode --mode=disable --cluster
```

上に記載されているように、オプションの —tag パラメーターは、特定のリソース階層のメンテナンスモードを無効にするために使用されます。

```
sudo /opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/lk_maintenance_mode --mode=disable --tag=<A/SCS Resource Tag>,<ERS Resource Tag< --cluster
```

## LifeKeeper 保護下にあるリソースのメンテナンスモードステータスの確認

LifeKeeper GUI は現在、各リソース階層のメンテナンスモードステータスを表示しません。 LifeKeeper で SAP 階層のメンテナンスモードステータスを確認するには、コマンドラインから次のコマンドを実行します。

```
sudo /opt/LifeKeeper/lkadm/subsys/appsuite/sap/bin/lk_maintenance_mode --mode=check --tag=<A/SCS Resource Tag> --cluster
```

ここで、<A/SCS Resource Tag> は、SAP ASCS またはSCS リソースの LifeKeeper タグです（例：SAP-SID_ASCS00）。出力には、指定されたリソースを含む階層のメンテナンスモードが完全に有効になっているか、部分的に有効になっているか、または完全に無効になっているかが表示されます。

# 6.16.10.2. リソースポリシー管理

注記: SPS-L バージョン 9.3.2 以降では、LifeKeeper UI の SSHCC アクションメニューを介して、SAP 階層の HA メンテナンスモードを有効にできます。詳細については、SAP メンテナンスモード を参照してください。

## 概要

SIOS Protection Suite for Linux のリソースポリシー管理では、リソースのローカルリカバリとフェイルオーバの動作管理機能が提供されます。リソースポリシーは、 **lkpolicy** コマンドラインツール (CLI) を使用して管理できます。

## SIOS Protection Suite

SIOS Protection Suite には、個々のアプリケーションおよび関連し合うアプリケーションのグループを監視する機能があり、定期的にローカルリカバリを実行したり、保護下のアプリケーションに障害が発生したときに通知することができます。関連し合うアプリケーションの例としては、主アプリケーションが下位のストレージまたはネットワークリソースに依存する階層などがあります。アプリケーションまたはリソースに障害が発生した場合のデフォルトの動作は以下の通りです。

1. **ローカルリカバリ**: 最初に、リソースまたはアプリケーションの **ローカル** でリカバリを試みます。このときは、外部の介入なしにローカルサーバ上でリソースまたはアプリケーションをリストアしようとします。ローカルリカバリが成功した場合、SIOS Protection Suite は追加のアクションを実行しません。

2. **フェイルオーバ**: 次に、ローカルリカバリでリソースまたはアプリケーションのリストアに *失敗した* (またはリソースを監視する *リカバリキット* がローカルリカバリをサポートしていない) 場合、 *フェイルオーバ* は開始されます。フェイルオーバ処理では、クラスタ内の別のサーバ上で該当アプリケーション (および依存するすべてのリソース) を起動しようと試みます。

リカバリ動作の詳細については、SIOS Protection Suite 障害検出とリカバリのシナリオ を参照してください。

## ポリシーによるカスタム動作およびメンテナンスモード動作

SIOS Protection Suite Version 7.5 以降では、デフォルトのリカバリ動作を変更する追加ポリシーを設定する機能をサポートします。リソース単位 *または* サーバ単位で、4 つのポリシーが設定可能です (リソース単位のポリシーに関する注意については下のセクションを参照してください)。 ***サーバレベルでポリシーを変更する方法を推奨します。***

利用可能なポリシーは以下の通りです。

### 標準ポリシー

- **Failover** - このポリシー設定を使用すると、リソースフェイルオーバを有効 / 無効にできます。(**S**:

リザベーションが適切に処理されるには、 **フェイルオーバ** は、個々の SCSI リソースで無効にすることはできません。)

- **LocalRecovery** - SIOS Protection Suite は、デフォルトでは、フェイルオーバを実行する前に、個々のリソースまたは保護対象アプリケーション全体を再起動することにより、保護対象リソースのリカバリを試みます。このポリシー設定を使用すると、ローカルリカバリを有効 / 無効にできます。

- **TemporalRecovery** - 通常、SIOS Protection Suite は、障害リソースのローカルリカバリを実行します。ローカルリカバリに失敗すると、SIOS Protection Suite は、リソース階層を別ノードにフェイルオーバします。ローカルリカバリに成功した場合は、フェイルオーバは実行されません。

  ローカルリカバリに成功した場合でも、サーバの何らかの異常によって短時間の間にローカルリカバリが再試行される場合があり、結果として何度も連続してローカルリカバリが試行されることになります。これが発生すると、問題のアプリケーションは可用性が悪化します。

  この反復的なローカルリカバリ / 障害サイクルを回避するために、時間的リカバリポリシーを設定できます。時間的リカバリポリシーを使用すると、管理者は指定した時間内に試行するローカルリカバリの回数を (成功かどうかにかかわらず) 制限することができます。

  *例:* リソースが試行するローカルリカバリの回数を 30 分間で 3 回に限定するポリシー定義をユーザが設定した場合、30 分 *以内に* 3 回目のローカルリカバリが試行されると、SIOS Protection Suite はフェイルオーバを実行します。

  定義した時間的リカバリポリシーは *有効* または *無効* にできます。時間的リカバリポリシーが *無効* の場合、時間的リカバリ処理は継続して実行され、ポリシーが *適用されるはず* の時間に通知がログに表示されますが、実際のアクションは実行されません。

  **注記:** 時間的リカバリポリシーを設定した状態で、フェイルオーバとローカルリカバリの一方または両方を無効にすることは可能です。フェイルオーバまたはローカルリカバリを無効にした場合に、時間的リカバリポリシーは実行されることが **ない** ため、この状態は非論理的です。

### メタポリシー

「メタ」ポリシーは、他の複数のポリシーに影響を与える可能性があるポリシーです。通常、これらのポリシーは、標準ポリシーであれば複数個の設定が必要になるような特定のシステム動作を実現するためのショートカットとして使用します。

- **NotificationOnly** - このモードでは、管理者は SIOS Protection Suite を「監視専用」状態にすることができます。 ***1 つのリソース (または、サーバ単位のポリシーの場合はすべてのリソース) のローカルリカバリおよびフェイルオーバの両方が影響を受けます。*** 障害が検知されると、ユーザインターフェースには **Failure** 状態が表示されます。 *ただし、リカバリもフェイルオーバも実行されません。* **注記:** 管理者は、障害の原因となった問題を手動で修正し、障害が起きたリソースを復帰させて通常の SIOS Protection Suite の運用を継続する必要があります。

## リソースレベルのポリシーに関する重要な考慮事項

リソースレベルのポリシーとは、リソース階層全体またはサーバレベルのポリシーとは異なり、特定のリ

ソースにのみ適用されるポリシーです。

*例*:

アプリケーション
- IP
- file system

上記のリソース階層では、アプリケーションは IP とファイルシステムの両方に依存しています。ポリシーは、特定のリソースのローカルリカバリまたはフェイルオーバを無効にするように設定できます。これは、例えば、IP リソースのローカルリカバリが失敗し、IP リソースのフェイルオーバが *無効* に設定されていた場合、IP リソースはフェイルオーバを実行せず、他のリソースのフェイルオーバも発生させないことを意味します。ただし、ファイルシステムリソースのローカルリカバリが失敗し、ファイルシステムリソースのポリシーのフェイルオーバが無効化されていない場合、階層全体がフェイルオーバを実行します。

**注記:** 重要事項として、リソースレベルのポリシーは設定対象の特定のリソースに *のみ* 適用されることに注意してください。

上記は単純な例です。複雑な階層を構成することもできるため、リソースレベルのポリシーを設定するときは注意してください。

# lkpolicy ツール

**`lkpolicy`** ツールは、SIOS Protection Suite for Linux が稼働するサーバのポリシーを管理 (参照、設定、削除) するためのコマンドラインツールです。`lkpolicy` は、ポリシーの設定および修正、ポリシーの削除、利用可能なポリシーと現在の設定値の表示をサポートします。さらに、設定したポリシーは、有効または無効に設定できるため、リカバリ動作に影響を与えながらリソース / サーバ設定を保持できます。

全体的な使用方法は次の通りです。

```
lkpolicy [--list-policies | --get-policies | --set-policy | --remove-policy]
<name value pair data…>
```

`<name value pair data…>`は、運用方法 *および* 対象のポリシーによって異なります (特にポリシーを設定する場合)。 *例*: 有効 / 無効タイプのポリシーのほとんどでは、必要なのは `--on` または `--off` のスイッチのみですが、時間的ポリシーの場合は、しきい値を設定するための値も必要です。

## lkpolicy の使用方法の例

### ローカルおよびリモートサーバとの認証

**`lkpolicy`** ツールは、サーバが公開する API を通じて SIOS Protection Suite サーバと通信します。この API は、lkpolicy ツールなどのクライアントに対して認証を要求します。`lkpolicy` ツールで SIOS Protection Suite サーバに最初にアクセスしようとしたときに、そのサーバに対する認証情報がまだ保存されていない場合、ユーザは認証情報を求められます。認証情報はユーザ名とパスワードの形式であり、さ

らに以下の条件があります。

1. クライアントには SIOS Protection Suite の管理者権限が必要です。したがって、そのユーザ名は、( *PAM* による) オペレーティングシステムの認証設定によって *lkadmin* グループに属する必要があります。必ずしも ***root*** で実行する ***必要はありません*** が、root ユーザはデフォルトで適切なグループに属しているため、root を使用することもできます。

2. 認証情報は *認証情報ストア* に保存されるため、ツールを使用してこのサーバにアクセスするたびに手動で認証情報を入力する必要はありません。

認証情報ストアと credstore ユーティリティによる管理の詳細については、SIOS Protection Suite の認証情報の設定 を参照してください。

`lkpolicy` によるセッションの例は以下のようになります。

```
[root@thor49 ~]# lkpolicy -l -d v6test4
Please enter your credentials for the system 'v6test4'.
Username: root
Password:
Confirm password:
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]# lkpolicy -l -d v6test4
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]#
```

### ポリシーのリスト表示

lkpolicy --list-policy-types

### 現在のポリシーの表示

lkpolicy --get-policies

lkpolicy --get-policies tag=*

lkpolicy --get-policies --verbose tag=mysql* # all resources starting with mysql

lkpolicy --get-policies tag=mytagonly

### ポリシーの設定

```
lkpolicy --set-policy Failover --off

lkpolicy --set-policy Failover --on tag=myresource

lkpolicy --set-policy Failover --on tag=\*

lkpolicy --set-policy LocalRecovery --off tag=myresource

lkpolicy --set-policy NotificationOnly --on

lkpolicy --set-policy TemporalRecovery --on recoverylimit=5 period=15

lkpolicy --set-policy TemporalRecovery --on --force recoverylimit=5 period=10
```

### ポリシーの削除

```
lkpolicy --remove-policy Failover tag=steve
```

*注記: NotificationOnly はポリシーのエイリアスです。NotificationOnly を有効にすることは、対応する LocalRecovery および Failover ポリシーを無効にすることと等価です。*

# 6.17. SAP HANA Recovery Kit管理ガイド

✻ v9.5.0より、SIOSは新しいSAP HANA Application Recovery Kitをリリースしました。SIOSは **2022年3月31日** まで、9.4.xリリースでのSAP HANA gen/appベースのRecovery Kitのサポートを継続します。SIOS Protection Suite for Linux v9.5以降をご利用の場合は、新しい（ビルトインの）SAP HANA Application Recovery Kitを使用する必要があります。

! 既存のSAP HANA gen/appベースのRecovery Kitはv9.5.0では **サポートされていません** 。SIOS Protection Suite for Linux v9.5.0へのアップグレードを希望するユーザーは、既存のSAP HANA gen/appベースのRecovery Kitから新しいSAP HANA Recovery Kitに **移行する必要があります** 。詳細については、 SAP HANA Gen/AppからSAP HANA Recovery Kitへのアップグレード を参照してください。

SAP HANAは、幅広い用途に展開できるエンタープライズクラスのインメモリデータベースシステムで、SAP S/4HANA ERPプラットフォームの基盤です。
SAP HANA Recovery Kitは、SIOS Protection Suite for LinuxにおけるSAP HANAデータベースの耐障害性保護を提供します。

## 本ガイドの構成

このガイドには、SAP HANA階層の作成および管理に役立つ次のトピックが含まれています。

- SAP HANA Recovery Kitの要件 SAP HANA Recovery Kitの適切なセットアップ、インストール、操作に必要なハードウェアとソフトウェアをリストしています。
- 概要 SAP HANA Recovery Kitの機能について説明しています。
- SPSでのSAP HANAの設定 SAP HANAソフトウェアをインストールおよび構成する手順を提供します。
- リソース設定タスク LifeKeeper GUIを使用して階層で実行できるさまざまな機能（作成、拡張、削除、および拡張解除）について説明しています。
- 階層管理 SAP HANA階層の継続的な管理に関する重要な推奨事項を提供します。
- トラブルシューティング SAP HANA Recovery Kitに関連するエラーメッセージをリストして説明しています。

## SIOS Protection Suiteドキュメント

次のSIOS Protection Suite製品ドキュメントは、SIOS Technology Corp. のウェブサイトでご覧いただけます。

- SPS for Linux リリースノート
- SPS for Linuxテクニカルドキュメンテーション
- SPS for Linuxインストレーションガイド

- オプションのRecovery Kitドキュメント

本ドキュメントは、オプションのLifeKeeper Recovery Kitに関連するドキュメントとともに、SIOS Technology Corp.のウェブサイト http://jpdocs.us.sios.com/ またはLifeKeeper GUIのヘルプメニューで提供されています。

# SAP HANAドキュメント

Documentation for SAP HANAに関するドキュメントは、以下より入手できます。

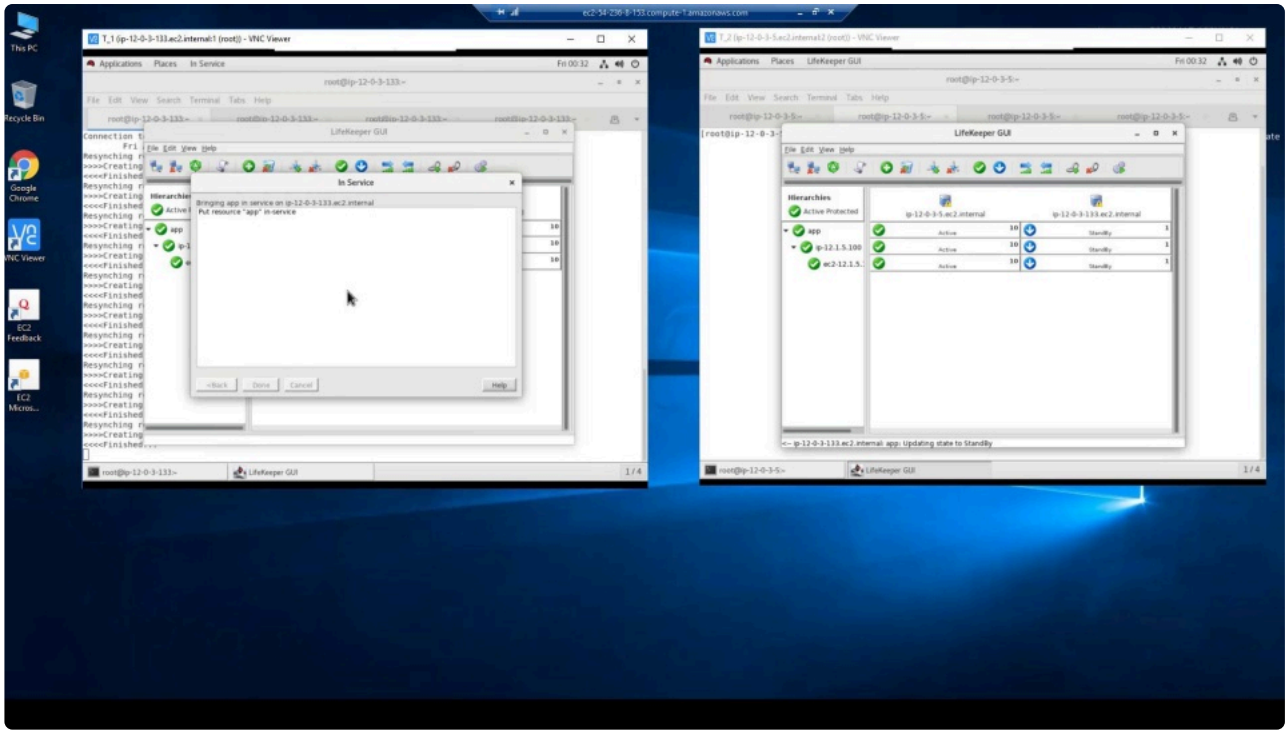
https://help.sap.com/viewer/ec06fadc50b64b6184f835e4f0e1f52f/4.2.9/ja-JP

# 6.17.1. SAP HANA Gen/AppからSAP HANA Recovery Kitへのアップグレード

✻ v9.5.0より、SIOSは新しいSAP HANA Application Recovery Kitをリリースしました。SIOSは **2022年3月31日** まで、9.4.xリリースでのSAP HANA gen/appベースのRecovery Kitのサポートを継続します。SIOS Protection Suite for Linux v9.5以降をご利用の場合は、新しい（ビルトインの）SAP HANA Application Recovery Kitを使用する必要があります。

! 既存のSAP HANA gen/appベースのRecovery Kitはv9.5.0ではサポートされていません。SIOS Protection Suite for Linux v9.5.0へのアップグレードを希望するユーザーは、既存のSAP HANA gen/appベースのRecovery Kitから新しい SAP HANA Recovery Kit に移行する必要があります。以下のアップグレード手順に従ってください。

✻ アップグレード前に、 サポートマトリックス を参照してください。



https://fast.wistia.net/embed/iframe/aggjjny6n1

**参考動画（英語）**

## SIOS Protection Suiteのアップグレード

1. クラスター内のすべてのノードで、SIOS Protection Suiteを最新バージョンにアップグレードします ( SPS のアップグレード を参照してください).

2. **SAP HANA ARK** と、対応する **必要な** ライセンスをクラスター内のすべてのノードにインストールします。

# SAP HANA Gen/App リソースの削除

1. LifeKeeperがすべてのノードで実行され、gen/appリソース階層がIn Serviceになっている必要があります。

2. リソース階層を以前の設定に復元する必要がある場合に備えて、LifeKeeperの設定をバックアップします。バックアップを取るには、次のコマンドを実行します。

   ```
   /opt/LifeKeeper/bin/lkbackup -c --cluster
   ```

   デフォルトでは、バックアップファイルは以下にあります。

   */opt/LifeKeeper/config/archive.<date – yyyymmddhhmm>.tar.gz*.

3. 左のパネルで **SAP HANA gen/app** リソース階層を右クリックします。

4. 依存関係がgen/appリソース（仮想IPまたはAWS EC2 EIPなど）にアタッチされている場合は、**[Delete Dependency...]** を選択します。

5. 依存関係が削除されたら、 **SAP HANA gen/app** リソースを右クリックします。

6. **[Delete Resource Hierarchy...]** を選択し、古いSAP HANA gen/appリソースを削除します。

> SAP HANA gen/appリソースが削除された後、新しいSAP HANAリソースが作成されるまで、SAP HANAインスタンスに対するSIOS Protection Suiteのフェイルオーバーまたは監視は行われないことに注意してください。SPS外部での変更によって終了される場合を除き、HANAシステムレプリケーションはアクティブのままになります。

# SAP HANAリソースの作成と拡張

新しくインストールされたSAP HANA ARKを使用して、SAP HANAリソースを作成/拡張します。

1. **SAP HANAリソース** を作成および拡張するには、 SAP HANAリソース階層の作成 および SAP HANAリソース階層の拡張 の手順に従ってください。

2. **SAP HANAリソース** 階層を作成して拡張した後、上記のステップ4で削除した依存関係を作成または再作成したい場合は、次の手順を実行します。

   a. 左のパネルで **SAP HANAリソース** 階層を右クリックします。

   b. **[Create Dependency...]** を選択し、SAP HANA gen/appリソースにアタッチされていた新しいSAP HANAリソース（仮想IPまたはAWS EC2 EIPなど）に依存関係を作成します。

3. ここ にあるテストを実行して、新しいSAP HANAリソース階層を確認/テストします。

## HANA Gen/Appファイルの削除

各システムからgen/appパッケージを削除するには、次のコマンドを実行してください。

```
rpm -e steeleye-lkHOTFIX-HANA-SP1-9.1.0-6538.noarch
```

# 6.17.2. SAP HANA Recovery Kitハードウェアおよびソフトウェアの要件

v9.5.0より、SIOSは新しいSAP HANA Application Recovery Kitをリリースしました。SIOSは **2022年3月31日** まで、9.4.xリリースでのSAP HANA gen/appベースのRecovery Kitのサポートを継続します。SIOS Protection Suite for Linux v9.5以降をご利用の場合は、新しい（ビルトインの）SAP HANA Application Recovery Kitを使用する必要があります。

既存のSAP HANA gen/appベースのRecovery Kitはv9.5.0では **サポートされていません** 。SIOS Protection Suite for Linux v9.5.0へのアップグレードを希望するユーザーは、既存のSAP HANA gen/appベースのRecovery Kitから新しい SAP HANA Recovery Kit に **移行する必要があります** 。詳細については、 SAP HANA Gen/AppからSAP HANA Recovery Kitへのアップグレード を参照してください。

## ハードウェアの要件

- **サーバー：** サーバーは、SAP HANAマスターガイド、本ガイドで言及しているSAPノート、SPS for Linuxドキュメント、SPS for Linuxリリースノートに記載されている要件に従って構成する必要があります。

- **ストレージ：** SAP HANAシステムレプリケーションを使用しているSAP HANAデータベースの場合、共有ストレージは不要です。特別なストレージ要件は、SAP HANAマスターガイドおよび前述のSAPノートに記載されています。

## ソフトウェア要件

- **SPSソフトウェア：** 必ずクラスター内の各サーバーに同じバージョンのSPSソフトウェアをインストールし、同じバージョンのSPSソフトウェアパッチを適用してください。

- **SAP HANAソフトウェア：** SAP HANA Recovery Kitを使用するには、SAP HANA Platform Edition 2.0 SP04以降が必要です。 SPSおよびSPS SAP HANA Recovery Kitを構成する前に、同じバージョンのSAP HANAソフトウェアを各サーバーにインストールして構成する必要があります。SAP HANAライセンスはハードウェアに関連付けられているため、各サーバーにはSAPの独自のライセンスが必要です。

  LifeKeeperでSAP HANAリソース階層を作成する前に、SAP HANAシステムレプリケーションを有効にして完全に構成し、データベースをすべてのクラスターノードで実行する必要があります。

- **SPS SAP HANA Recovery Kit：** SAP HANA Recovery Kitは、SIOS Protection Suite for Linux v9.5.0以降のCoreインストールにオプションのRecovery Kitとしてバンドルされています。

注記: SAP HANA Recovery Kitは2ノード構成のみがサポートされ、3ノード構成はサポー

トされていません。

- **Witness**サーバー： SIOSでは、2ノード構成に3番目のノードをWitnessサーバーとして追加することを推奨しています。

# 6.17.3. SAP HANA Recovery Kit概要

SAP HANA Recovery Kitは、SAP HANA Platform Edition 2（SP04以降）と互換性があります。

SAP HANAは、可用性を高めるための3つの異なるメカニズムを提供しています。

- **ホストの自動フェイルオーバー** – SAP HANAシステムに追加された少なくとも1つのスタンバイノード。これらのノードは、スタンバイモードで動作するように構成されています。必要なプロセスまたはデータベースがアクティブでない場合、LifeKeeperはそれらの再起動を試みます。
  プライマリノードでプロセスの再起動に失敗した場合、LifeKeeperはバックアップノードでデータベースをIn Serviceにし、そのノードをSAP HANAシステムレプリケーションのプライマリマスターとして登録し、以前のプライマリノードをセカンダリレプリケーションサイトとして登録しようとします。以前のプライマリノードをセカンダリSAP HANAシステムレプリケーションサイトとして構成できない場合、問題が修正されて正常に登録できるようになるまで、そのノードでリソースは障害（OSF）としてマークされます。以前のプライマリノードがセカンダリレプリケーションサイトとして正常に登録されると、LifeKeeperはノード上のSAP HANAリソースの状態をスタンバイ（OSU）に更新します。

- **ストレージレプリケーション** – SAP HANAプライマリノードで使用されるストレージは、すべてのデータを別のSAP HANAノードに複製します。この複製は、SAP HANA システムからの制御プロセスなしで実行されます。このストレージレプリケーション機能は、ハードウェアパートナーが提供します。

- **システムレプリケーション** – SAP HANAは、SAP独自の組み込みレプリケーションフレームワークを使用して、SAP HANAプライマリノードからバックアップノードにすべてのデータを複製します。データはSAP HANAセカンダリノードに常にプリロードされます。

SAP HANA Recovery Kitを使用すると、システムレプリケーションを使用しているSAP HANAシステムをSIOS LifeKeeperで保護および管理できます。

Recovery Kitは、すべてのノードでSAP HANAシステムを起動し、SAP HANAシステムレプリケーションのテイクオーバーおよびレプリケーションサイト登録プロセスを実行できます。SAP HANAシステムの機能を保証するために、以下のプロセスと状態が継続的に監視されます。

- すべてのノード上の SAP ホストエージェント
- すべてのノード上のHDBインスタンスのSAP Start Service（sapstartsrv）
- すべてのノード上のSAP HANAデータベースの状態
- SAP HANAシステムレプリケーションモードの状態（アクティブノードでプライマリ、セカンダリノードでsync|syncmem|async）

必要なプロセスまたはデータベースがアクティブでない場合、LifeKeeperはそれらの再起動を試みます。プライマリノードでプロセスの再起動に失敗した場合、LifeKeeperはバックアップノードでデータベースをIn Serviceにし、そのノードをSAP HANAシステムレプリケーションのプライマリマスターとして登録して、以前のプライマリノードをセカンダリレプリケーションサイトとして登録しようとします。以前のプライマリノードをセカンダリSAP HANAシステムレプリケーションサイトとして構成できない場合、問題が修正されて正常に登録できるまで、そのノードでリソースは障害（OSF）としてマークされます。以前のプライマリノードがセカンダリレプリケーションサイトとして正常に登録されると、LifeKeeperはノー

ド上のSAP HANAリソースの状態をスタンバイ（OSU）に更新します。

SAP HANAシステムレプリケーションの状態が無効の場合、SAP HANAリソースも「Out of Service – Faulty」（OSF）状態になります。テイクオーバーを実行するかどうか、またはSAP HANAシステムレプリケーションモードの修正方法については、データベース管理者に相談して決定する必要があります。

LifeKeeperでSAP HANAリソースに対して「Out of Service」アクションを実行すると、デフォルトではプライマリノード上のデータベースのみが停止します。セカンダリノードのデータベースは引き続きアクティブで、SAP HANAシステムレプリケーションモードを保持します。

# 6.17.3.1. SAP HANA GUI ステータス

アクティブ（ISP）リソースには、以下の表に示すステータスがあります。これらの警告および障害のステータスの一部は一時的なものであり、LifeKeeperが必要なプロセスを回復しようとしている間、またはLifeKeeperがHANAリソース階層のフェイルオーバーを開始する前に表示される場合があります。

| GUIのテキスト | GUIプロパティのステータス | 説明 | アイコン |
|---|---|---|---|
| Active | アクティブ | SapstartsrvおよびHDBインスタンスが正常に実行されています（ステータス名：ISP）。 | |
| Active – sapstartsrv Failure | Sapstartsrvが起動していない、HDBステータスが不明である | Sapstartsrvが起動していません。sapstartsrvはHDBのステータスを確認するために使用されるため、HDBインスタンスのステータスは不明です（ステータス名：ISPSapStartSrvNotRunning）。 | |
| Active – HSR Disabled | HSRはレプリケーションモード「None」で起動している | SapstartsrvおよびHDBは正常に実行されています。レプリケーションモードは「None」で、SAP HANAレプリケーションが無効になっていることを示しています（ステータス名：ISPHSRDisabled）。 | |
| Active – HDB Stopped | Sapstartsrvは起動しているが、HDBは正常に起動していない | Sapstartsrvは起動していますが、HDBインスタンスは正常に起動していません（ステータス名：ISPHDBNotRunning）。 | |
| Active – Secondary | HSRはセカンダリレプリケーションモードで実行されている | SapstartsrvとHDBインスタンスは正常に実行されていますが、アクティブノードがセカンダリレプリケーションサイトとして登録されています。通常の操作では、アクティブノードはプライマリレプリケーションサイトであることが期待されます（ステータス名：ISPSecondary） | |
| Active – Unknown Repl Mode | レプリケーションモードを特定できない | SapstartsrvおよびHDBインスタンスは正常に実行されています。「`hdbnsutil -sr_state`」を使用してレプリケーションモードを特定できないか、サポートされていないモードを返します。（ステータス名：ISPUnknownReplMode） | |
| Active – Suspended | データベースがプライマリおよび一時停止している。 | データベースは SAP HANA リソースが現在 In Service になっているサーバーで一時停止しています。これは予期しない状態で、通常、ユーザーが LifeKeeper の外部で“Takeover with handshake” を実行した時に発生します。(ステータス名: ISPSuspended) | |

スタンバイ（OSU）リソースのGUIに表示されるステータスは、アクティブ（ISP）リソースのリストアおよびquickCheckプロセスの影響を受けます。スタンバイリソースはIn Serviceの操作中に最終的なステー

タスになるまで数分かかる場合があり、その間に一時的なIntermidiateのステータスが表示される場合がありますが、これはSAP HANAリソースの正常な動作です。以下に示す警告および障害のステータスの一部は一時的なものであり、LifeKeeperが必要なプロセスの回復を試みている間、またはスタンバイサーバーをセカンダリレプリケーションサイトとして再登録しようとしているときに表示される場合があります。SAP HANAデータベースを正常に起動できない場合、またはバックアップサーバー上のセカンダリレプリケーションサイトとして登録できない場合は、バックアップサーバー上のリソースのステータスは失敗（Failed: OSF）に移行します。スタンバイリソースのステータスを以下の表に示します。

| GUIのテキスト | GUIプロパティのステータス | 説明 | アイコン |
|---|---|---|---|
| Standby | クラスターHANAリソースISPがない。HDBは実行されていない | これは、同等のリソースが別のノードでIn Serviceでない場合にOut of Serviceになっているリソースの通常のステータスです（ステータス名：OSUStopped）。 | |
| Standby – HDB Running | HDBは実行中、HSRモニタリングは非アクティブ | SapstartsrvおよびHDBインスタンスはスタンバイ（OSU）ノードで実行されています。アクティブ（ISP）ノードが存在しないか、quickCheckがアクティブノードのHSRステータスを判断していません（ステータス名：OSUHDBRunning）。 | |
| Standby – In Sync | HSRがアクティブで同期されている | HSRが構成され、レポートが同期された状態でSapstartsrvとHDBインスタンスが実行されています（ステータス名：SecondaryActive）。 | |
| Standby – sapstartsrv Failure | sapstartsrvが実行されておらず、HDBステータスが不明 | Sapstartsrvが実行されていません。アクティブ（ISP）ノードが存在します。sapstartsrvはHDBのステータス確認に使用されるため、HDBインスタンスのステータスは不明です（ステータス名：OSUSapStartSrvNotRunning）。 | |
| Standby – Unknown Repl. Mode | レプリケーションモードを判別できない | SapstartsrvとHDBインスタンスが実行されています。 アクティブ（ISP）ノードが存在します。 `hdbnsutil -sr_state` を使用してレプリケーションモードを決定できません（ステータス名：OSUUnknownReplMode）。 | |
| Standby – HSR Disabled | レプリケーションモードが「None」 | SapstartsrvとHDBインスタンスが実行されています。アクティブ（ISP）ノードが存在します。 `hdbnsutil -sr_state` に従ってレプリケーションモードが「none」であるか、HANAユーティリティsystemReplicationStatus.py が「10」を返しました（ステータス名：OSUHSRDisabled）。 | |
| Standby – HSR Error | アクティブノードHSRがエラー | SapstartsrvとHDBインスタンスが実行されています。アクティブ（ISP）ノードが存在します。HANAユーティリティsystemReplicationStatus.py は「11」を返しました（ステータス名： | |

| | | | |
|---|---|---|---|
| | （11）を報告 | OSUHSRError）。 | |
| Standby – Initializing | HSRの初期化中 | SapstartsrvとHDBインスタンスが実行されています。アクティブ（ISP）ノードが存在します。HANAユーティリティ systemReplicationStatus.py は「13」を返しました（ステータス名：SecondaryInitializing）。 | |
| Standby – Syncing | HSRの同期中 | SapstartsrvとHDBインスタンスが実行されています。アクティブ（ISP）ノードが存在します。HANAユーティリティ systemReplicationStatus.py は「14」を返しました（ステータス名：SecondarySyncing）。 | |
| Standby – HDB Stopped | クラスター内のHANAリソースISP、HDBが実行されていない | Sapstartsrvが実行されているが、HDBインスタンスは実行されていません。アクティブ（ISP）ノードが存在します（ステータス名：OSUHDBNotRunning）。 | |
| Standby – Primary | HSRはレプリケーションモード「Primary」で実行されている | SapstartsrvとHDBインスタンスは正常に実行されていますが、スタンバイノードがプライマリレプリケーションサイトとして登録されています。通常の運用では、スタンバイノードはセカンダリレプリケーションサイトであることが期待されます（ステータス名：OSUPrimary）。 | |
| Standby – Unknown HSR Status | アクティブノードHSRがエラー（12）を報告 | SapstartsrvとHDBインスタンスが実行されています。アクティブ（ISP）ノードが存在します。HANAユーティリティ systemReplicationStatus.py は「12」を返しました（ステータス名：SecondaryUnknownHSRStatus）。 | |
| Standby – Unexpected HSR state | ISPノードのHSRステータスが予想範囲10～15にない | SapstartsrvとHDBインスタンスが実行されています。アクティブ（ISP）ノードが存在します。HANAユーティリティ systemReplicationStatus.py は、10～15の予想範囲外の値を返しました（ステータス名：OSUUnexpectedHSRState）。 | |
| Standby – Suspended | データベースがプライマリおよび一時停止している。 | データベースは SAP HANA リソースが現在 Out of Service になっているサーバーで一時停止しています。 これは通常、一時停止されていたデータベースが停止され、セカンダリレプリケーションサイトとして再登録されている間の“Takeover with handshake”プロセス実行中に表示される予期された一時的なステータスとなります。(ステータス名: OSUSuspended) | |

# 6.17.3.2. SAP HANAリソース階層

次の例は、一般的なSAP HANAリソース階層を示しています。

この例での子リソース **vip-sps-hana** は、データベースへのクライアント接続に使用される切り替え可能なIPアドレスを保護しています。

フェイルオーバーが発生した場合、SPSはIPアドレスとデータベースリソースをバックアップサーバーでIn Serviceにします。まだコミットされていないデータベーストランザクションは、再度実行する必要があります。

# 6.17.3.3. マルチテナントデータベースのサポート

SAP HANA Recovery Kit は、マルチテナントデータベース環境でも動作することが可能です。ただし、下記の動作と特記事項についてご注意ください。

- SAP HANA システムレプリケーション (HSR) は、 リカバリーキットの HSR モニタリングを正しく機能させるために、システム上の **全ての** テナントデータベースに対するレプリケーションを正常に初期化できる必要があります。 つまり、例えば HSR を有効にする前に、全てのテナントデータベースのバックアップを取得する必要があるということです。HSR が保護されたインスタンス内に含まれるいずれかのテナントデータベースに対して非同期である場合、SAP HANA Recovery Kit は 、スタンバイシステムに !HANA_DATA_OUT_OF_SYNC フラグを残し、SAP HANA リソース階層のスイッチオーバーもしくはフェイルオーバーを回避します。

- SAP HANA Recovery Kit は現在 個々のテナントデータベースを手動で停止した場合 (つまり、HDBSQL の‘alter system stop database DB_NAME’ コマンドを使用して停止すること。)は、それらを再起動することはありません。この場合、SAP HANA ソフトウェアは HDB インスタンスのプロセスのリストから一致する indexserver プロセスを削除し、 SAP HANA Recovery Kit は、テナントデータベースの手動停止を意図的なものとして扱います。

- SAP HANA Recovery Kit は現在テナントデータベースを個別のレベルで監視していません。 SAP HANA Recovery Kit は(起動している各テナントデータベースのindexserver プロセスを含む) 全部の HDB インスタンスを全体として監視します。 HDB インスタンスのいずれかのプロセスが失敗すると、全てのインスタンスの再起動を試みます。つまり、例えば TENANT2 の indexserver プロセスが失敗するとTENANT1 の indexserver プロセスが予期せず再起動されてしまいます。

# 6.17.4. SPSでのSAP HANAの設定

SAP HANAデータベースおよびSPSソフトウェアのインストールと構成には、次の手順を実行することをお勧めします。これらの各手順は、後続の詳細なタスクにリンクしています。

SAP HANAソフトウェアのインストール

SAP HANAシステムレプリケーションの構成

SAP HANAインスタンスプロファイルの変更

SPSソフトウェアのインストール

これらのタスクを完了すると、SPSリソース階層を作成してSAP HANAデータベースを保護する準備が整います。

# 6.17.4.1. SAP HANAソフトウェアのインストール

同一のパラメーター/設定を使用して、クラスター内のすべてのサーバーにSAP HANAソフトウェアをインストールします。特に、すべてのシステムで同じSAPシステムID（SID）とインスタンス番号を使用する必要があります。インストールの詳細については、SAP HANAマスターガイドを参照してください。

# 6.17.4.2. SAP HANAシステムレプリケーションの構成

SAP HANAシステムレプリケーションガイド（http://help.sap.com で入手可能）の指示に従って、SAP HANAシステムレプリケーションを構成します。対象のプライマリレプリケーションサイトでSAP HANAシステムレプリケーションが有効になり、バックアップサーバーがセカンダリレプリケーションサイトとして登録されたら、本ガイドの次の手順に進んでください。

> **注記:** マルチテナントデータベース環境において、SAP HANA システムレプリケーションは、SAP HANA Recovery Kit を正しく機能させることができるよう全てのテナントデータベースに対するレプリケーションを正常に初期化できる必要があります。特に、システムのレプリケーションを有効にする前に保護対象のインスタンスに含まれる全てのテナントデータベースのバックアップを取得する必要があります。

# 6.17.4.3. SAP HANAインスタンスプロファイルの変更

## HDBインスタンスのAutostartを無効にする

フェイルオーバーまたはシステムの再起動中にLifeKeeperでSAP HANAリソースを管理するには、システム起動時にHDBインスタンスを自動的に起動するAutostart機能を無効にする必要があります。この機能を無効にするには、 すべてのクラスターノードで HDBインスタンスのインスタンスプロファイル (通常 /usr/sap/<SID>/SYS/profile/<SID>_HDB<##>_<HostName> に存在) を編集し、プロファイル内に

```
Autostart = 0
```

という記述があるようにします。必要に応じてこの行を追加または変更し、変更をプロファイルに保存してください。

**警告:** HDBインスタンスのAutostartが無効になっていない場合、SAP HANAプライマリシステムレプリケーションサイトのマシン障害により、元のプライマリノードがオンラインに戻った際に「システムレプリケーションのスプリットブレイン」シナリオが発生します。このシナリオでは、HDBインスタンスが実行されており、複数のクラスターノードで同時にプライマリマスターとして登録されています。その結果、LifeKeeperは、ユーザーがSAP HANAプライマリシステムレプリケーションサイトとして登録したいサイトを判断できません。SAP HANAリソースは元のプライマリノードでOSF（「Out of Service – 障害」）状態になり、データベース管理者が状況を手動で解決するまで、開いているすべてのコンソールに警告メッセージが表示されます。詳細については、 スプリットブレインシナリオの解決 を参照してください。

**注記:** SAP HANAリソースの正常な管理を継続するために、LifeKeeperはすべてのクラスターノード上のHDBインスタンスのAutostartパラメーターの値をquickCheckの間隔ごと（LKCHECKINTERVALパラメーター、デフォルト2分で定義）に監視し、必要に応じてAutostartを自動的に無効にします。

# 6.17.4.4. SPSソフトウェアのインストール

SAP HANAソフトウェアをインストールし、SAP HANAシステムレプリケーションの構成が完了したら、SPS Coreソフトウェアと必要なパッチをインストールし、その後SAP HANA Recovery Kitをインストールできます。

SPSパッケージのインストールの詳細については、 SPS for Linuxインストレーションガイド を参照してください。

# 6.17.5. SAP HANAリソースの設定タスク

前のセクションで説明したセットアップタスクを完了すると、SAP HANAリソース階層を作成して拡張できるようになります。

SIOS Protection Suite for Linux SAP HANA Recovery Kitを設定するには、次のタスクを実施します。

- **リソース階層の作成** – SAP HANAリソース階層を作成します。
- **リソース階層の削除** – SAP HANAリソース階層を削除します。
- **リソース階層の拡張** – SAP HANAリソース階層をプライマリサーバーからバックアップサーバーに拡張します。
- **リソース階層の拡張解除** – SPSクラスター内の単一サーバーからSAP HANAリソース階層の拡張を解除（削除）します。

LifeKeeperリソース階層を設定するための追加の手順については、 SPS for Linuxテクニカルドキュメンテーション を参照してください。

以下のタスクは、すべてのRecovery Kitで手順が同じ一般的なタスクなため、SPS for Linuxテクニカルドキュメント内の 管理 セクションの説明を参照してください。

- リソース依存関係の作成 既存のリソース階層と別のリソースインスタンスの間に親子の依存関係を作成し、依存関係の変更をクラスター内の該当するすべてのサーバーに通知します。
- リソース依存関係の削除 リソースの依存関係を削除し、依存関係の変更をクラスター内のすべての該当するサーバーに通知します。
- In Service 特定のサーバーでリソース階層をIn Serviceにします。
- Out of Service 特定のサーバーでリソース階層をOut of Serviceにします。
- プロパティの 表示 / 編集 特定のサーバー上のリソース階層のプロパティを表示または編集します。

**注記：** 本セクションの残りの部分では、 **Edit** メニューを使用して設定タスクを実施します。

また、ほとんどのタスクは以下によって実行できます。

- ツールバーから実行する
- ステータス表示の左ペインでグローバルリソースを右クリックする
- ステータス表示の右ペインでリソースを右クリックする

右クリック方式を使用すると、 **Edit** メニューの使用時に必要な特定の情報の入力をせずに済みます。

# 6.17.5.1. SAP HANAリソース階層の作成

> **重要：** SAP HANAリソース階層を作成する前に、SAP HANAシステムレプリケーションを有効にして完全に構成し、データベースをクラスター内のすべてのサーバーで実行する必要があります。 詳細については、 SAP HANAシステムレプリケーションの構成 をご覧ください。

プライマリサーバーで次の手順を実行します。

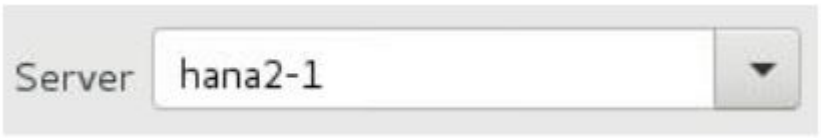
1. **Edit** メニューから、 **[Server]** 、 **[Create Resource Hierarchy]** を選択します。 ***Create Resource Wizard*** のダイアログが表示されます。

2. ドロップダウンリストから **SAP HANA** を選択し、 **[Next]** をクリックします。

3. SAP HANAリソースに使用する **スイッチバックタイプ** として、 *intelligent* を選択します。 **[Next]** をクリックします。

インテリジェントスイッチバックとは、バックアップサーバーへのフェイルオーバー後、管理者がSAP HANAリソースをプライマリサーバーに手動で切り替える必要があることを意味します。 注意： SPSのこのリリースは、SAP HANAリソースの自動スイッチバックをサポートしていません。さらに、自動スイッチバックの制限事項は、SAP HANAリソースに依存する他のLifeKeeperリソースにも適用されます。

4. SAP HANAリソースを作成する **サーバー** を選択します。最初のSAP HANAリソースを作成する場合、このステップで選択するサーバーは、そのサーバー上でデータベースが現在SAP HANAシステムレプリケーションのプライマリマスターとして登録されているサーバーでなければなりません。 **[Next]** をクリックします。




5. SAP HANAデータベースがインストールされている **SAP HANA SID** を選択し、 **[Next]** をクリックします。

6. LifeKeeperで保護する **SAP HANA** インスタンスを選択し、 **[Next]** をクリックします。

7. LifeKeeperで保護する、 **依存する仮想IPリソース** を選択します。 リストに表示するには、IPリソースはすでに存在し、選択したサーバーでIn Serviceになっている必要があります。LifeKeeper IPリソースを使用せずにフェイルオーバーまたはスイッチオーバーで仮想IP/ホスト名の切り替えを管理する場合は、「*none*」を選択します。 **[Next]** をクリックします。

8. 選択したサーバー上に作成されるSAP HANAリソースを識別するために使用される **SAP HANAリソースタグ** を入力します。 デフォルトのタグ名の形式は HANA-<SID>_<HDB インスタンス> ですが、現在別のLifeKeeperリソースの識別に使用されていないものであれば、別のタグ名を使用するように変更できます。タグ名で使用できる文字は、文字、数字、および次の特殊文字「-」、「_」、「.」、「/」です。

9. **[Create]** をクリックします。 ***Create Resource Wizard*** でSAP HANAリソース階層を作成します。SPSは、入力されたデータとSAP HANAシステムレプリケーションフレームワークから取得したデータを検証します。問題が検出されると、情報ボックスにエラーメッセージが表示されます。 **[Next]** をクリックします。

10. SAP HANAリソース階層が正常に作成されたことを示すメッセージが表示され、その階層をクラスター内の別のサーバーに拡張してフェイルオーバー保護を実現する必要があります。 **[Next]** をクリックします。

11. **[Continue]** をクリックします。その後、SPSは ***Pre-Extend Wizard*** を起動します。 リソース階層を別のサーバーに拡張する方法の詳細については、 SAP HANAリソース階層の拡張 を参照してください。

# 6.17.5.2. SAP HANAリソース階層の拡張

> **重要：** SAP HANAリソース階層を拡張する前に、SAP HANAシステムレプリケーションを有効にして完全に構成し、データベースをクラスター内のすべてのサーバーで実行する必要があります。詳細については、 SAP HANAシステムレプリケーションの構成 をご覧ください。

この操作は、 **[Edit]** メニューから実行するか、 **リソース階層の作成** オプションの完了時に自動的に開始できます。この場合、以下の手順2から開始する必要があります。

1. **[Edit]** メニューから、 **[Resource]** 、 **[Extend Resource Hierarchy]** の順に選択します。 ***Pre-Extend Wizard*** が表示されます。拡張操作に慣れていない場合は、 **[Next]** をクリックします。SPSのリソース階層拡張のデフォルト値に精通しており、入力/確認のプロンプトをスキップしたい場合は、 **[Accept Defaults]** をクリックしてください。

2. 既存のSAP HANAリソースを拡張する **テンプレートサーバー** を選択します（ *右ペインのリソースインスタンスを右クリックして **Extend** （タスクの拡張）を選択した場合、このダイアログボックスは表示されません* ）。 **[Next]** をクリックします。

3. 拡張したいSAP HANAリソースの **拡張するタグ** を選択します（ *いずれかのペインでリソースインスタンスを右クリックして **Extend** （タスクの拡張）を選択した場合、このダイアログボックスは表示されません* ）。 **[Next]** をクリックします。

4. SAP HANAリソースに使用する **スイッチバックタイプ** に *intelligent* を選択し、 **[Next]** をクリックします。

   インテリジェントスイッチバックとは、バックアップサーバーへのフェイルオーバー後、管理者がSAP HANAリソースをプライマリサーバーに手動で切り替える必要があることを意味します。 注意： SPSのこのリリースは、SAP HANAリソースの自動スイッチバックをサポートしていません。さらに、自動スイッチバックの制限事項は、SAP HANAリソースに依存する他のLifeKeeperリソースにも適用されます。

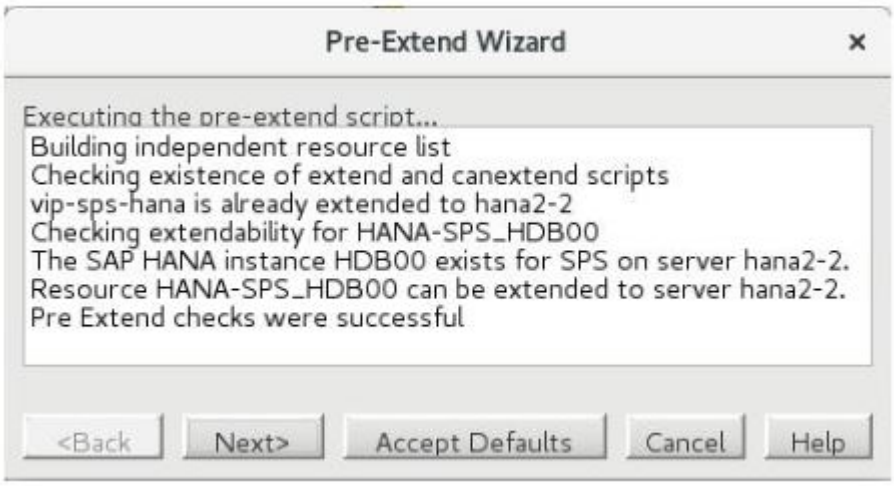
5. **テンプレートの優先順位** を選択または入力します。 **[Next]** をクリックします。

これは、現在稼働中のサーバー上のSAP HANA階層の優先順位です。1～999の未使用の優先順位はすべて有効で、数値が小さいほど優先順位が高くなります（1 =最高）。拡張プロセスは、別のシステムですでに使用されているこの階層の優先順位を拒否します。デフォルト値の使用を推奨します。 **注記：** ここで選択した値は、階層の最初の拡張に対してのみ表示されます。

6. **ターゲットの優先順位** を選択または入力し、 **[Next]** をクリックします。

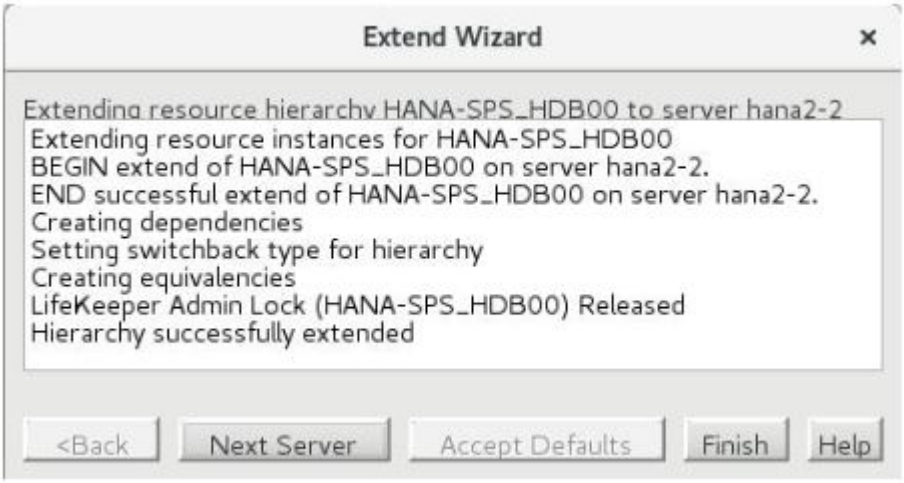
これは、新しく拡張されたSAP HANA階層の、他のサーバー上の同等の階層に対する優先順位です。1～999の未使用の優先順位はすべて有効で、リソースのカスケードフェイルオーバーのシーケンスにおけるサーバーの優先順位を示します。SPSは、デフォルトでは階層が作成されるサーバーに優先順位1を割り当てることに注意してください。優先順位は連続している必要はありませんが、2つのサーバーが特定のリソースに対して同じ優先順位を持つことはできません。

7. 事前拡張チェックが成功したというメッセージが表示されたら、 **[Next]** をクリックします。




8. ***Extend Wizard*** により、SAP HANAリソースの **ルートタグ** を入力するよう求められます。これは、LifeKeeperがターゲットサーバーで作成される同等のSAP HANAリソースを識別するために使用する一意の名前です。 **注記：** SAP HANAリソースタグ名はすべてのクラスターサーバーで同じである必要があるため、このダイアログボックスでは編集できません。

9. **[Extend]** をクリックします。 ***Extend Wizard*** で、SAP HANAリソース階層をターゲットサーバーに拡張します。問題が検出されると、情報ボックスにエラーメッセージが表示されます。



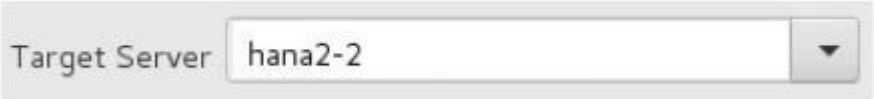

10. 「Hierarchy successfully extended」というメッセージが表示されたら、 **[Next Server]** をクリックして階層を別のサーバーに拡張するか、実行する拡張操作が他にない場合は **[Finish]** をクリックします。

11. 「Hierarchy Verification Finished」というメッセージが表示されたら、 **[Done]** をクリックします。

# 6.17.5.3. SAP HANAリソース階層の拡張解除

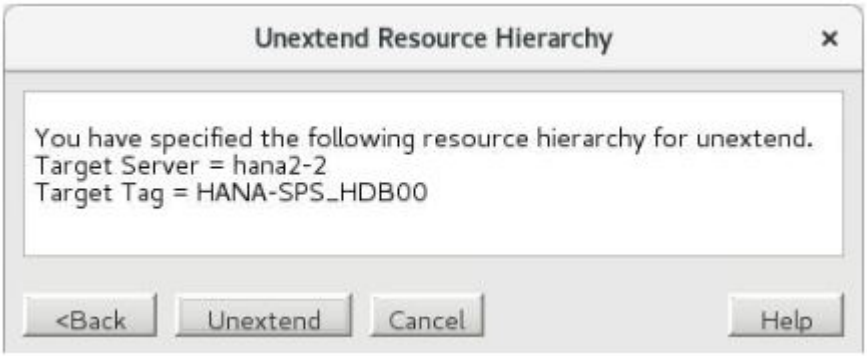
SPSクラスター内の単一サーバーからリソース階層を削除するには、次の手順を実行します。

1. **[Edit]** メニューから、 **[Resource]** 、 **[Unextend Resource Hierarchy]** の順に選択します。
2. SAP HANAリソースの拡張を解除する **ターゲットサーバー** を選択します。リソースが現在 In Service のサーバーは選択できません（ *右ペインでリソースインスタンスを右クリックして **Unextend** タスクを選択した場合、このダイアログボックスは表示されません* ）。 **[Next]** をクリックします。

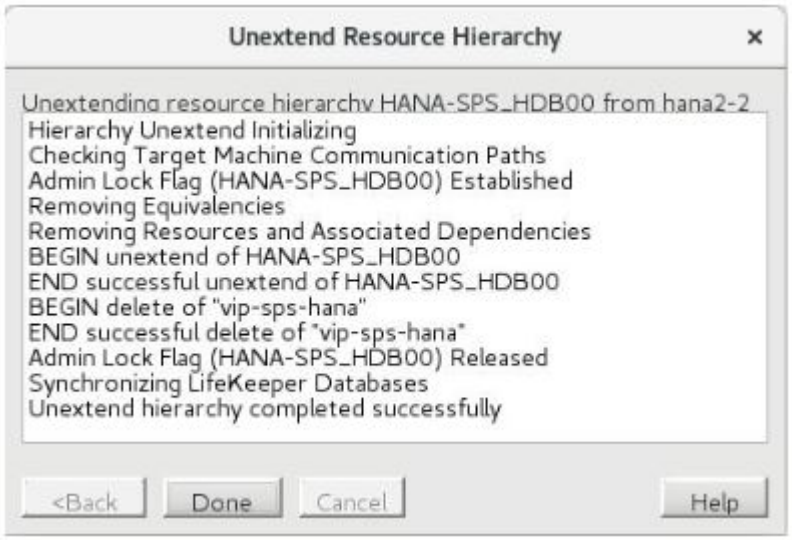



3. 拡張解除するSAP HANA階層を選択し、 **[Next]** をクリックします（ *いずれかのペインでリソースインスタンスを右クリックして **Unextend** タスクを選択した場合、このダイアログボックスは表示されません* ）。

4. 対象のサーバーと、拡張解除を選択したSAP HANAリソース階層を確認する情報ボックスが表示されます。 **[Unextend]** をクリックします。




5. SAP HANAリソースが正常に拡張解除されたことを確認する別の情報ボックスが表示されます。 **[Done]** をクリックして、 **リソース階層の拡張解除** メニューの選択を終了します。


Unextend Resource Hierarchy
Unextending resource hierarchy HANA-SPS_HDB00 from hana2-2
Hierarchy Unextend Initializing
Checking Target Machine Communication Paths
Admin Lock Flag (HANA-SPS_HDB00) Established
Removing Equivalencies
Removing Resources and Associated Dependencies
BEGIN unextend of HANA-SPS_HDB00
END successful unextend of HANA-SPS_HDB00
BEGIN delete of "vip-sps-hana"
END successful delete of "vip-sps-hana"
Admin Lock Flag (HANA-SPS_HDB00) Released
Synchronizing LifeKeeper Databases
Unextend hierarchy completed successfully
<Back
Done
Cancel
Help

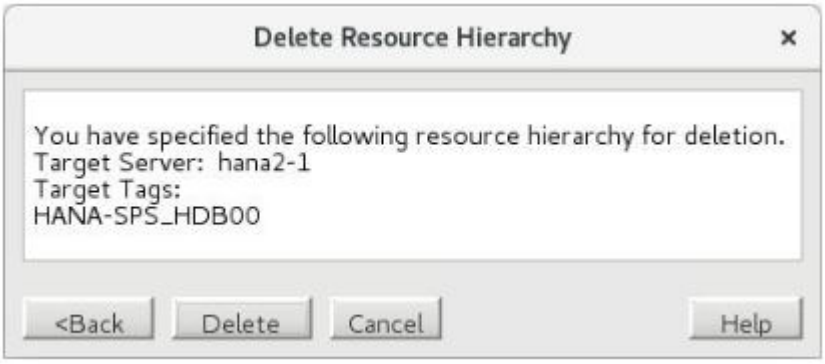
# 6.17.5.4. SAP HANAリソース階層の削除

SPSクラスター環境のすべてのサーバーからSAP HANAリソースを削除するには、次の手順を実行します。

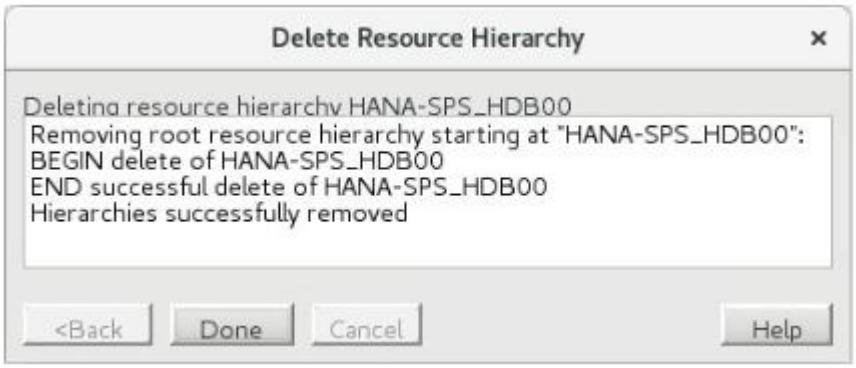
1. **[Edit]** メニューから、 **[Resource]** 、 **[Delete Resource Hierarchy]** の順に選択します。
2. SAP HANAリソース階層を削除する **ターゲットサーバー** の名前を選択します（ *グローバルリソースの左ペインまたは個々のリソースインスタンスの右ペインから右クリックして* ***Delete Resource*** *タスクを選択した場合、このダイアログは表示されません* ）。

3. **削除する階層** を選択します（ *左または右ペインのリソースインスタンスを右クリックして* ***Delete Resource*** *タスクを選択した場合、このダイアログは表示されません* ）。 **[Next]** をクリックします。

4. 削除する対象のサーバーと選択した階層の選択を確認する情報ボックスが表示されます。 **[Next]** をクリックします。




5. SAP HANAリソースが正常に削除されたことを確認する別の情報ボックスが表示されます。




6. **[Done]** をクリックして終了します。

# 6.17.5.5. SAP HANAリソース階層のテスト

## テストシナリオ

SAP HANA Recovery Kitの動作を理解するには、次のテストを実行してください。テストを実行する前に、次の前提条件を満たす必要があります。

- LifeKeeperおよびSAP HANAデータベースは、SIOSおよびSAPが提供するインストール手順に従ってインストールおよび設定する必要があります。
- 有効なレプリケーションモード（sync、syncmem、またはasync）およびオペレーションモード（delta_datashipping、logreplay、またはlogreplay_readaccess）のいずれかを使用してセカンダリレプリケーションサイトを登録し、クラスター内のすべてのサーバーでSAP HANAシステムレプリケーションを有効にしてアクティブにする必要があります。詳細については、 SAP HANAシステムレプリケーションの構成 を参照してください。
- SAP HANAデータベースに関連付けられた切り替え可能なIPアドレスをLifeKeeper IPリソースで管理する場合、IPリソースに対するSAP HANAリソースの依存関係が存在する必要があります。詳細については、 SAP HANAリソース階層の作成 のステップ7を参照してください。

### SAPホストエージェントのリカバリーのテスト

以下を使用して、SAPホストエージェントプロセスのステータスとプロセスIDを決定します。

```
# /usr/sap/hostctrl/exe/saphostexec -status

saphostexec running (pid = 3818)

sapstartsrv running (pid = 3867)

saposcol running (pid = 3965)
```

出力にリストされているプロセスの1つを手動で強制終了するか、

```
/usr/sap/hostctrl/exe/saphostexec -stop
```

を実行して、SAPホストエージェントの機能を低下させます。するとSAP HANA Recovery KitはSAPホストエージェントが正しく機能していないことを認識し、そのノードで再起動します。この動作は、次のコマンドでLifeKeeperログを監視することで確認できます。

```
tail -f /var/log/lifekeeper.log
```

このリカバリープロセス中、SAP HANAリソースの状態は変化しません。正常に回復すると、SAPホストエージェントは再び完全に機能するようになります。Recovery KitがSAPホストエージェントを再起動できない場合、HANAデータベースとリソースは現在の状態のままになります。するとSAPホストエージェントが再度チェックされ、可能であれば後で再起動されます。

## SAP HANAインスタンスのsapstartsrvのリカバリーのテスト

SAP HANAインスタンスのSAP Start Service（sapstartsrv）のリカバリーをテストするには、サービスを停止する必要があります。sapstartsrvを停止する方法の1つに、sapcontrol StopService webmethodの実行があります。

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function StopService"
```

<sid> はHANAインストール時に使用した小文字のSAPシステムID、<Inst#> はHDBインスタンス番号です。また、sapstartsrvプロセスを直接強制終了することもできます。いずれの場合も、sapstartsrvはSAP HANA Recovery Kitによって再起動されます。sapstartsrvが正常に再起動できる限り、リソースの状態は変わりません。

## セカンダリSAP HANA DB（レプリケーションターゲット）のリカバリーのテスト

セカンダリデータベースインスタンス（レプリケーションターゲット）に障害が発生した場合、またはセカンダリレプリケーションサイトがSAP HANAシステムレプリケーションで登録解除された場合、Recovery Kitは以前のレプリケーションモードとオペレーションードでセカンダリサイトを再登録し、セカンダリデータベースインスタンスを再起動します。

このような障害を引き起こすには、セカンダリレプリケーションサイトで次のコマンドのいずれかを実行します。

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function Stop"
```

```
su - <sid>adm -c "hdbnsutil -sr_unregister"
```

この動作は、ログファイル /var/log/lifekeeper.log を監視することで確認できます。リカバリー後、セカンダリレプリケーションサイトで次のコマンドを実行することにより、データベースインスタンスとSAP HANAシステムレプリケーションの状態をテストできます。

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function GetProcessList"
```

```
su - <sid>adm -c "hdbnsutil -sr_state"
```

セカンダリデータベースインスタンスをRecovery Kitで起動できない場合、SAP HANAリソースには、対応するノードで障害（OSF）のフラグが立てられます。

起動に失敗した原因を管理者が修正すると、SAP HANA Recovery Kitは後続のquickCheckサイクルでデータベースインスタンスを起動します。データベースインスタンスが正常に起動すると、対応するノードでリソースの状態がスタンバイに更新されます。

## プライマリSAP HANA DBのリカバリーのテスト

プライマリデータベースインスタンス（レプリケーションソース）に障害が発生した場合、プライマリノード上のデータベースインスタンスのレプリケーションモードが決定されます。レプリケーションモードがプライマリに設定されている場合、データベースインスタンスが再び起動されます。モードがプライマリに設定されていない場合、Recovery KitはLifeKeeperの外部でレプリケーションモードが変更されたことを示す警告をログに出力し、問題が解決するまでSAP HANAリソースの監視をすべて一時停止します。後者の場合、HANAリソース階層を正しいプライマリシステムでIn Serviceにするには、手動による操作が必要です。この場合の動作は、LifeKeeper GUIで確認できます。この場合の動作は、LifeKeeper GUIで確認できます。LifeKeeperGUIには、プライマリノードのリソースの状態が「Active – HSR Disabled」、「Active – Unknown Repl Mode」、または「Active – Secondar」と表示されます。あるいは、ログファイル `/var/log/lifekeeper.log` でも確認できます。

プライマリレプリケーションサイトで次のコマンドを実行すると、プライマリデータベースインスタンスの障害を引き起こすことができます。

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function Stop"
```

リカバリー後、データベースとレプリケーションの状態は次のコマンドを使用してテストできます。

```
su - <sid>adm -c "sapcontrol -nr <Inst#> -function GetProcessList"
```

```
su - <sid>adm -c "hdbnsutil -sr_state"
```

そのノードのプライマリデータベースインスタンスをRecovery Kitで起動できない場合、LifeKeeperはセカンダリノードへの階層全体のフェイルオーバーを開始します。このノードでHANA Recovery KitはSAP HANAシステムレプリケーションのテイクオーバーを実行し、以前のセカンダリノードがレプリケーションの新しいプライマリノードになります。LifeKeeperは、以前のレプリケーションモードとオペレーションモードを使用して、障害のあるノードをセカンダリレプリケーションサイトとして再登録しようとします。これが成功すると、セカンダリデータベースも再起動されます。セカンダリノードをセカンダリレプリケーションサイトとして正常に登録できない場合、またはセカンダリノードでデータベースを正常に再起動できない場合、対応するノードでHANAリソースに障害（OSF）のフラグが立てられます。障害の原因を除去するには、通常この時点で手動による操作が必要です。プライマリデータベースインスタンスのフェイルオーバーに失敗した場合、リソースに障害（OSF）のフラグが立てられ、管理者が手動でIn Service操作を実行するまでこの状態が継続します。

## セカンダリノードのマシン障害のテスト (reboot -f, power off)

エラーが原因でセカンダリノードに障害が発生した場合、リソースはプライマリノードでアクティブ（ISP）のままですが、SAP HANAシステムレプリケーションは中断されます。セカンダリノードが再起動され、LifeKeeperがアクティブになると、セカンダリデータベースインスタンスがレプリケーションターゲットとして自動的に再起動されます。

## プライマリノードのマシン障害のテスト (reboot -f, power off)

エラーが原因でプライマリノードに障害が発生した場合、HANAリソース階層のセカンダリノードへのフェイルオーバーが開始されます。SAP HANAシステムレプリケーションのテイクオーバーがセカンダリノー

ドで実行され、以前のセカンダリレプリケーションサイトが新しいプライマリレプリケーションサイトになります。障害が発生したノードが再起動され、LifeKeeperがアクティブになると、ノードはセカンダリレプリケーションサイトとして登録され、データベースインスタンスがレプリケーションターゲットとして自動的に再起動されます。

# 6.17.6. SAP HANAリソース階層の管理

**重要：** 本ガイドで特に明記されていない限り、LifeKeeperで保護されたSAP HANAデータベースのすべての管理タスクは、LifeKeeperを介して実行する必要があります。LifeKeeperでSAP HANAリソースがアクティブ/ISPな場合にデータベースの停止やLifeKeeperの外部でのSAP HANAシステムレプリケーションの無効化などの管理操作を実行すると、LifeKeeperはデータベースを望ましい実行状態に戻すアクションを実行します。

## SAP HANAリソースのスイッチオーバー

プライマリデータベースインスタンスのスイッチオーバーが開始されると、SAP HANA Recovery Kitは次の手順を実行します。

- データベースインスタンスを以前のプライマリノードで停止する
- 新しいプライマリノード（以前のセカンダリノード）でSAP HANAシステムレプリケーションのテイクオーバーを実行する
- 新しいセカンダリノード（以前のプライマリノード）を、セカンダリSAP HANAシステムレプリケーションサイトとして再登録する
- データベースインスタンスを新しいセカンダリノードで起動する

SAP HANAリソースをIn Serviceにする際に以下のようなメッセージが表示された場合は、プライマリデータベースインスタンスが停止したときにSAP HANAシステムレプリケーションが同期していなかったことを意味します。

```
ERROR:hana:restore:HANA-SPS_HDB00:136266:The resource HANA-SPS_HDB00
protecting SAP HANA database HDB00 is not in sync. To protect the data
LifeKeeper will not restore the resource on $me. Please restore the resource
on the previous source server to allow the resync to complete.
```

したがって、セカンダリデータベースサーバーにまだ複製されていないデータがプライマリデータベースサーバーに存在する可能性があります。このため、LifeKeeperはセカンダリサーバーがプライマリレプリケーションの役割を引き継ぐことを許可しません。この場合、SAP HANAリソース階層を以前のプライマリサーバーでIn Serviceに戻し、再同期を完了することを推奨します。

以前のプライマリサーバーがダウンして回復できない場合、データが同期されていないサーバーでSAP HANAリソースを強制的にオンラインにすることができます。しかし **これにより、以前のプライマリサーバーからまだ複製されていないすべてのデータが失われます** 。データベース管理者が、このデータの損失が許容可能または不可避であると判断した場合、次のコマンドを使用して非同期データフラグを手動で削除できます。

```
/opt/LifeKeeper/bin/flg_remove -f '!HANA_DATA_OUT_OF_SYNC_<Tag>'
```

<Tag> は、LifeKeeperのSAP HANAリソースタグ名です（例：HANA-SPS_HDB00）。非同期データフラグを削除した後、HANAリソースのIn Service操作を再試行します。In Serviceになると、データベースはプライマリSAP HANAシステムレプリケーションの役割を引き継ぎ、以前のプライマリサーバー上の複製されていないデータはすべて失われます。以前のプライマリサーバーが修復されてオンラインに戻るとセカンダリシステムのレプリケーションサイトとして登録され、データベースがレプリケーションターゲットとして再起動されます。

# SAP HANA データベースのTakeover with Handshake

“Takeover with handshake” は、SAP HANA 2.0 SPS04 以降に追加された機能で、新しいデータベースホスト上でSAP HANA システムレプリケーションの引き継ぎを行う前にプライマリデータベースを (完全に停止するのではなく)一時停止することで、スイッチオーバー時のプライマリデータベースのダウンタイムを抑えます。SAP HANA Recovery Kit で“Takeover with handshake” を実施する場合は、 Takeover with Handshake を参照してください。

# SAP HANAデータベースの停止

LifeKeeperでSAP HANAリソースがOut of Serviceになると、プライマリデータベースインスタンスのみが停止します。セカンダリデータベースインスタンスは、HANAリソース階層のスイッチオーバーまたはフェイルオーバー中のダウンタイムを最小限に抑えるために引き続き実行されます。

例外的な動作を引き起こす2つの特別なケースがあります。

1. SAP HANA リソースが Out of Service になっているシステム上において、LifeKeeper フラグ `!volatile!hana_leave_db_running_<HANA Tag>` が設定されている場合は、LifeKeeper は Out of Service の操作中にデータベースインスタンスを停止しません。このフラグが意図せず設定されている場合は、下記のコマンドを使用して削除することができます。:

   ```
   /opt/LifeKeeper/bin/flg_remove -f '!volatile!hana_leave_db_running_<HANA Tag>'
   ```

2. SAP HANA リソースが Out of Service になっているシステムにおいてデータベースが一時停止している場合、LifeKeeper は Out of Service の操作中にデータベースインスタンスを停止しません。一時停止されたデータベースインスタンスを起動し続けることで、必要に応じて再開するオプションを残しておくことができます。このシナリオは、LifeKeeper の外部でデータベース管理者による “Takeover with handshake”が手動で行われた際に最も発生しやくなります。必要に応じて一時停止されたデータベースインスタンスは下記のコマンドを使用して手動で停止することが可能です。:

   ```
   su - <sid>adm -c "sapcontrol -nr <InstNum> -function StopWait 600 5"
   ```

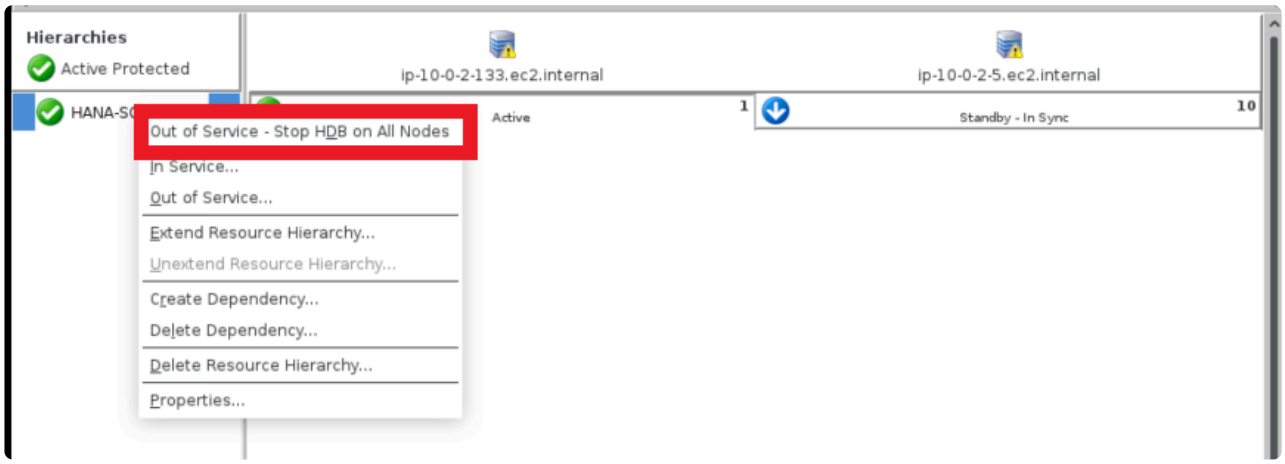
   <sid> はHANAインストール時に使用した小文字のSAPシステムID、<InstNum> はHDBインスタンス番号です。

# すべてのSAP HANAデータベースの停止（メンテナンスモード）

このオプションを選択すると、プライマリHANAリソースがOut of Serviceになり、HANAリソースクラスター内のすべてのHANAデータベースインスタンスが停止します。このオプションにより、バックアップマシンにすぐにフェイルオーバー/スイッチオーバーされる可能性があるため、細心の注意を払って実行する必要があります。 **注記：** このオプションは、メンテナンス期間中など、セカンダリデータベースインスタンスも停止する必要がある場合にのみ選択してください。

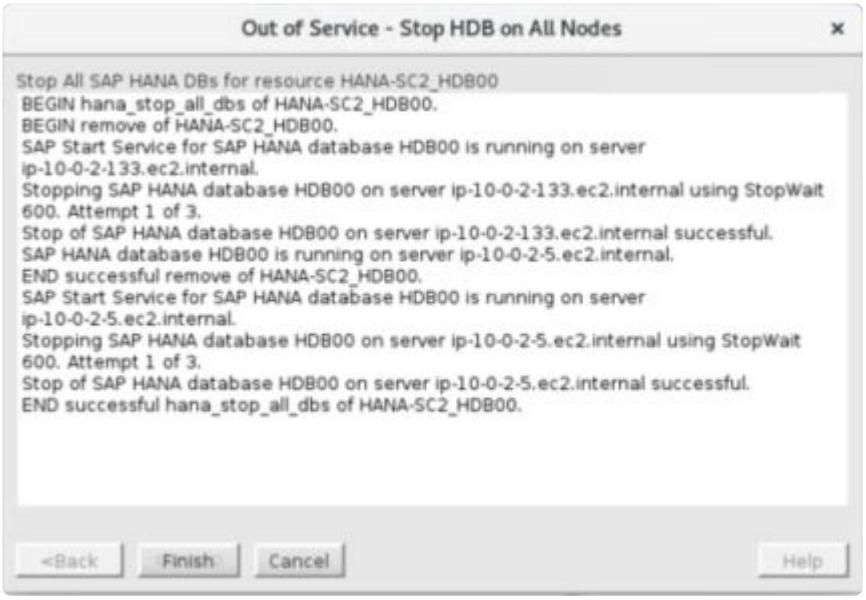
このオプションを使用するには、HANAリソース階層で次の手順を実行します

1. 左パネルの下のHANAリソースまたはIn Serviceのサーバーのいずれかを右クリックして、 **[Out of Service – Stop HDB on All Nodes]** オプションを選択します。




2. HANAリソースを確認し、ダイアログボックスに表示される指示に従います。 **[Stop All SAP HANA DBs]** をクリックしてプロセスを開始します。

3. プロセスが終了したら、 **[Finish]** をクリックしてプロセスを完了します。




4. SAP HANAリソースの最終状態は次のように表示されます。

5. すべてのメンテナンス作業が完了したら、最後のプライマリシステムでSAP HANAリソース階層をIn Serviceにしてください。

# 6.17.6.1. レプリケーションモードおよびオペレーションモードの変更

**警告：** セカンダリレプリケーションサイトのレプリケーションモードとオペレーションモードは変更できますが、LifeKeeperの外部でSAP HANAシステムレプリケーションの手動によるテイクオーバーまたはスイッチオーバーは実行しないでください。これを行うとエラー状態になり、SAP HANAシステムレプリケーションモードはLifeKeeperがアクティブ/スタンバイサーバーで期待しているモードと一致しなくなります。このエラー状態は、LifeKeeper GUIに次のリソース状態で表示されます。

この状態の間、SAP HANAリソースの監視はすべて、プライマリレプリケーションサイトとなるサーバー上のLifeKeeperでリソースがIn Serviceになるまで中断されます。

## レプリケーションモードの変更

データベースがセカンダリレプリケーションサイトで実行されている場合でも、 "hdbnsutil -sr_changemode" コマンドを使用してSAP HANAシステムレプリケーションモードを変更できます。

1. セカンダリレプリケーションサイトで次のコマンドを実行します。

```
su - <sid>adm -c "hdbnsutil -sr_changemode --mode=[sync|syncmem|async]"
```

<sid> は、SAP HANAインストール時に使用した小文字のSAPシステムIDで、希望するレプリケーションモード（sync、syncmem、またはasync）を --mode オプションで指定します。このコマンドを実行すると、セカンダリサイトでデータベースを実行または停止できます。

2. レプリケーションモードが正常に変更されたことを確認するには、セカンダリレプリケーションサイトで次のコマンドを実行し、"mode" パラメーターが新しいレプリケーションモードに更新されたことを確認します。

```
su - <sid>adm -c "hdbnsutil -sr_state"
```

次のquickCheck間隔（デフォルトは2分）後、LifeKeeperは変更を自動的に検出し、同等のすべてのSAP HANAリソースの情報フィールドを更新して新しいレプリケーションモードを含めます。これは、次のコマンドを使用してHANAリソースの情報フィールドを調べることで確認できます。

```
/opt/LifeKeeper/bin/ins_list -d <HostName> -t <Tag> -f: | cut -d: -f6 | tr '\002' ':'
```

<Hostname> は、情報を取得するサーバーのホスト名、 <Tag> は、指定されたサーバー上のLifeKeeperのSAP HANAリソースのタグ名（例：HANA-SPS_HDB00）です。

現在保存されているセカンダリレプリケーションモードは、このコマンドの出力の3番目のフィールドです。

```
SPS:HDB00:sync:shiba:logreplay
```

## オペレーションモードの変更

**注記：** オペレーションモードをlogreplayまたはlogreplay_readaccessからdelta_datashippingに変更するには、レプリケーションの再開時にプライマリサイトからセカンダリサイトへの完全なデータシッピングが必要になります。これらのオペレーションモードの変更の1つが検出されると、SAP HANAによってこの完全なデータシッピングが自動的に実行されます。

SAP HANAシステムレプリケーションのオペレーションモードを変更するには、通常、セカンダリサイトでデータベースを停止し、新しい操作モードで再登録し、セカンダリサイトでデータベースを再起動する必要があります。LifeKeeperがこのプロセス中にセカンダリサイトでデータベースを自動的に再起動しないようにするには、データベースが新しいオペレーションモードで正常に再登録されるまで、HANAリソースの監視を一時停止することをお勧めします。

1. リソースが現在アクティブ（ISP）になっているサーバーで対応する *nomonitor* フラグを作成して、HANAリソースの監視を一時停止します。

   ```
   /opt/LifeKeeper/bin/flg_create -f 'nomonitor_<Tag>'
   ```

   <Tag> は、そのサーバー上のLifeKeeperのHANAリソースタグです（例：HANA-SPS_HDB00）。このプロセスが完了したら、次のコマンドで *nomonitor* フラグを削除して、リソースの監視を再開することを忘れないようにしてください。

   ```
   /opt/LifeKeeper/bin/flg_remove -f 'nomonitor_<Tag>'
   ```

   **警告：** メンテナンス操作の実行後にこのフラグを削除しないと、SAP HANAデータベースの障害がLifeKeeperによって検出されなくなります。

2. 次のコマンドを実行して、バックアップサーバーでSAP HANAデータベースが停止していることを確認します。

   ```
   su - <sid>adm -c "sapcontrol -nr <HDB Inst#> -function StopSystem HDB"
   ```

   <HDB Inst#> は、SAP HANAデータベースインスタンスのインスタンス番号です（例：HDB00という名前のインスタンスの場合、インスタンス番号は00です）。

3. 使用したいレプリケーションモードとオペレーションモードを指定してバックアップサーバーで次のコマンドを実行し、セカンダリレプリケーションサイトとして再登録します。

   ```
   su - <sid>adm -c "hdbnsutil -sr_register --name=<SecondarySiteName> --
   ```

```
remoteHost=<PrimaryHost> --remoteInstance=<PrimaryInst#> --
replicationMode=[sync|syncmem|async] --
operationMode=[delta_datashipping|logreplay|logreplay_readaccess]"
```

<SecondarySiteName> は、セカンダリレプリケーションサイトを識別するためにSAP HANAシステムレプリケーションで使用されるエイリアス、 <PrimaryHost> は、プライマリレプリケーションサイトとして現在登録されているサーバーのホスト名、 <PrimaryInst#> はプライマリレプリケーションサイトのHDBインスタンスのインスタンス番号です。 **注記：** 新しいレプリケーションモードまたはオペレーションモードで再登録する前に、 "hdbnsutil -sr_unregister" コマンドでセカンダリサイトを登録解除する必要はありません。

4. バックアップサーバーが新しいオペレーションモードでセカンダリレプリケーションサイトとして正常に再登録されたことを確認するには、バックアップサーバーで次のコマンドを実行し、"is secondary/consumer system" が「true」であり、パラメーター"operation mode" が新しいオペレーションモードに更新されていることを確認します。

   ```
   su - <sid>adm -c "hdbnsutil -sr_state"
   ```

5. バックアップサーバーがセカンダリレプリケーションサイトとして正常に登録されたら、バックアップサーバーで次のコマンドを実行してデータベースを起動します。

   ```
   su - <sid>adm -c "sapcontrol -nr <HDB Inst#> -function StartSystem HDB"
   ```

   <HDB Inst#> は、SAP HANAデータベースインスタンスのインスタンス番号です。

6. プロセスが完了したら、ステップ1で作成した *nomonitor* フラグを削除して、LifeKeeperによるHANAリソースの監視を再開します。

   ```
   /opt/LifeKeeper/bin/flg_remove -f 'nomonitor_<Tag>'
   ```

   **警告：** このフラグを削除しないと、SAP HANAデータベースの障害がLifeKeeperによって検出されなくなります。

次のquickCheck間隔（デフォルトは2分）後、LifeKeeperは変更を自動的に検出し、同等のすべてのSAP HANAリソースの情報フィールドを更新して新しいオペレーションモードを含めます。これは、次のコマンドを使用してHANAリソースの情報フィールドを調べることで確認できます。

```
/opt/LifeKeeper/bin/ins_list -d <HostName> -t <Tag> -f: | cut -d: -f6 | tr
'\002' ':'
```

<HostName> は情報を取得するサーバーのホスト名で、 <Tag> は指定されたサーバー上のLifeKeeperのSAP HANAリソースのタグ名です（例：HANA-SPS_HDB00）。

現在保存されているセカンダリレプリケーションモードは、このコマンドの出力の5番目のフィールドです。

```
SPS:HDB00:sync:shiba:logreplay
```

# 6.17.6.2. スプリットブレインシナリオの解決

「スプリットブレイン」シナリオは、SAP HANAデータベースが実行され、複数のクラスターノードでSAP HANAのプライマリレプリケーションサイトとして構成されている場合に発生します。スプリットブレインが発生した場合、LifeKeeperは、データベース管理者が問題を手動で解決するまで、すべてのHANAデータベースの監視を一時停止します。

SAP HANAリソース階層で発生する可能性のある一般的なスプリットブレインシナリオは、2種類あります。

- **LifeKeeper HANAリソースのスプリットブレイン**: HANAリソースは、複数のクラスターノード上のLifeKeeperでアクティブ（ISP）です。この状態は通常、クラスターノード間のコミュニケーションパスに影響を与える一時的なネットワーク障害が原因です。

- **SAP HANAシステムレプリケーションのスプリットブレイン**: HANAリソースは、LifeKeeperのプライマリノードでアクティブ（ISP）、バックアップノードでスタンバイ（OSU）ですが、データベースは両方のノードでプライマリレプリケーションサイトとして実行および登録されています。この状態は通常、フェイルオーバー中に以前のプライマリノードでデータベースを停止できなかった、データベースのAutostartが有効になっている、またはLifeKeeperの外部のセカンダリレプリケーションサイトでデータベース管理者が "hdbnsutil -sr_takeover" を手動で実行したことが原因です。

各タイプのスプリットブレインシナリオを解決するための推奨事項を以下に示します。

## LifeKeeper HANAリソースのスプリットブレインの解決

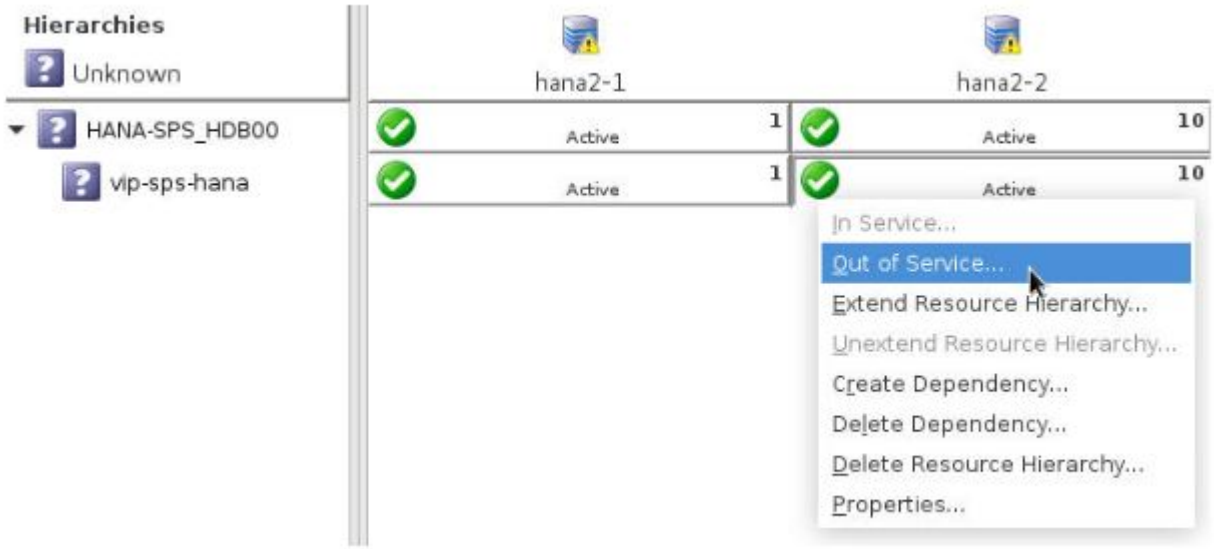
このスプリットブレインシナリオでは以下のようなメッセージがログに出力され、問題が解決されるまでquickCheckの間隔（デフォルトでは2分）ごとに、開いているすべてのコンソールに表示されます。

```
EMERG:hana:quickCheck:HANA-SPS_HDB00:136363:WARNING: A temporary communication
failure has occurred between servers hana2-1 and hana2-2. Manual intervention
is required in order to minimize the risk of data loss. To resolve this
situation, please take one of the following resource hierarchies out of
service: HANA-SPS_HDB00 on hana2-1 or HANA-SPS_HDB00 on hana2-2. The server
that the resource hierarchy is taken out of service on will become the
secondary SAP HANA System Replication site.
```

**推奨される解決方法：**

1. 各クラスターノードでデータベースを調査し、どのインスタンスに最新のデータまたは関連データが含まれているかを確認します。この確認は、データに精通している適任なデータベース管理者が行う必要があります。

2. 保持する必要があるデータを含むノードのHANAリソースはLifeKeeperでアクティブ（ISP）のままであり、セカンダリレプリケーションサイトとして再登録されるノードのHANAリソース階層は、LifeKeeperで完全にOut of Serviceになります。階層のサービスを停止するノード上のHANAリソース階層の各リーフリソースを右クリックし、 **[Out of Service ...]** をクリックします。




> このステップでは、SAP HANAリソース階層全体（ある場合は仮想IPリソースを含む）を、セカンダリレプリケーションサイトとして再登録されるノードでOut of Serviceにすることが重要です。

3. SAP HANAリソース階層のサービスがOut of Serviceになると、LifeKeeperは次のquickCheck間隔（デフォルトでは2分）の間にセカンダリノードをセカンダリレプリケーションサイトとして再登録します。レプリケーションが再開されると、アクティブノードに存在しないスタンバイノードのデータは失われます。スタンバイノードがセカンダリレプリケーションサイトとして再登録されると、SAP HANA階層は高い可用性が保たれた状態に戻ります。

## SAP HANAシステムレプリケーションのスプリットブレインの解決

このスプリットブレインシナリオでは、次のようなメッセージがログに出力され、問題が解決されるま

で、quickCheckの間隔（デフォルトでは2分）ごとに、開いているすべてのコンソールに表示されます。

```
EMERG:hana:quickCheck:HANA-SPS_HDB00:136364:WARNING: SAP HANA database HDB00
is running and registered as primary master on both hana2-1 and hana2-2.
Manual intervention is required in order to minimize the risk of data loss. To
resolve this situation, please stop database instance HDB00 on hana2-2 by
running the command 'su - spsadm -c "sapcontrol -nr 00 -function Stop"' on
that server. Once stopped, it will become the secondary SAP HANA System
Replication site.
```

**推奨される解決方法：**

1. 各クラスターノードでデータベースを調査して、スタンバイノードにアクティブノードには存在しない重要なデータが存在するかどうかを確認します。スプリットブレイン状態の間に重要なデータがスタンバイノードのデータベースにコミットされた場合、データをアクティブノードに手動でコピーする必要があります。これは、データに精通している適任なデータベース管理者が行う必要があります。

2. 不足しているデータがスタンバイノード上のデータベースからアクティブノードにコピーされたら、LifeKeeper警告メッセージに示されているコマンドを実行して、スタンバイノード上のデータベースを停止します。

   ```
   su - <sid>adm -c "sapcontrol -nr <Inst#> -function Stop"
   ```

   <sid> はHANAインストール時の小文字のSAPシステムIDで、 <Inst#> はHDBインスタンスのインスタンス番号です（例：インスタンスHDB00のインスタンス番号は00）。

3. データベースが正常に停止したら、LifeKeeperは次のquickCheck間隔（デフォルトでは2分）の間にスタンバイノードをセカンダリレプリケーションサイトとして再登録します。レプリケーションが再開されると、アクティブノードに存在しないスタンバイノードのデータは失われます。スタンバイノードがセカンダリレプリケーションサイトとして再登録されると、SAP HANA階層は高い可用性が保たれた状態に戻ります

# 6.17.6.3. Takeover with Handshake

## SAP HANA データベースの Takeover with Handshake

“Takeover with handshake” は、SAP HANA 2.0 SPS04 以降に追加された機能で、新しいデータベースホスト上でSAP HANA システムレプリケーションの引き継ぎを行う前にプライマリーデータベースを (完全に停止するのではなく)一時停止することで、スイッチオーバー時のプライマリーデータベースのダウンタイムを抑えます。

“Takeover with handshake” が開始されると、SAP HANA Recovery Kit は以下の処理を実行します。:

- データベースインスタンスを、旧プライマリーノードで起動したたままにする。
- 新しいプライマリーノード(旧セカンダリーノード)でSAP HANA システムレプリケーションの “Takeover with handshake” を実行する 。新しいプリマリーシステムレプリケーションサイトを登録する前に、旧プライマリーノードのデータベースを一時停止のステータスにする。
- 新しいセカンダリーノード (旧プライマリーノード) の一時停止されたデータベースを停止し、セカンダリーシステムレプリケーションサイトとして再登録する。
- データベースインスタンスを新しいセカンダリーノードで起動する。

## Takeover with Handshake の実行

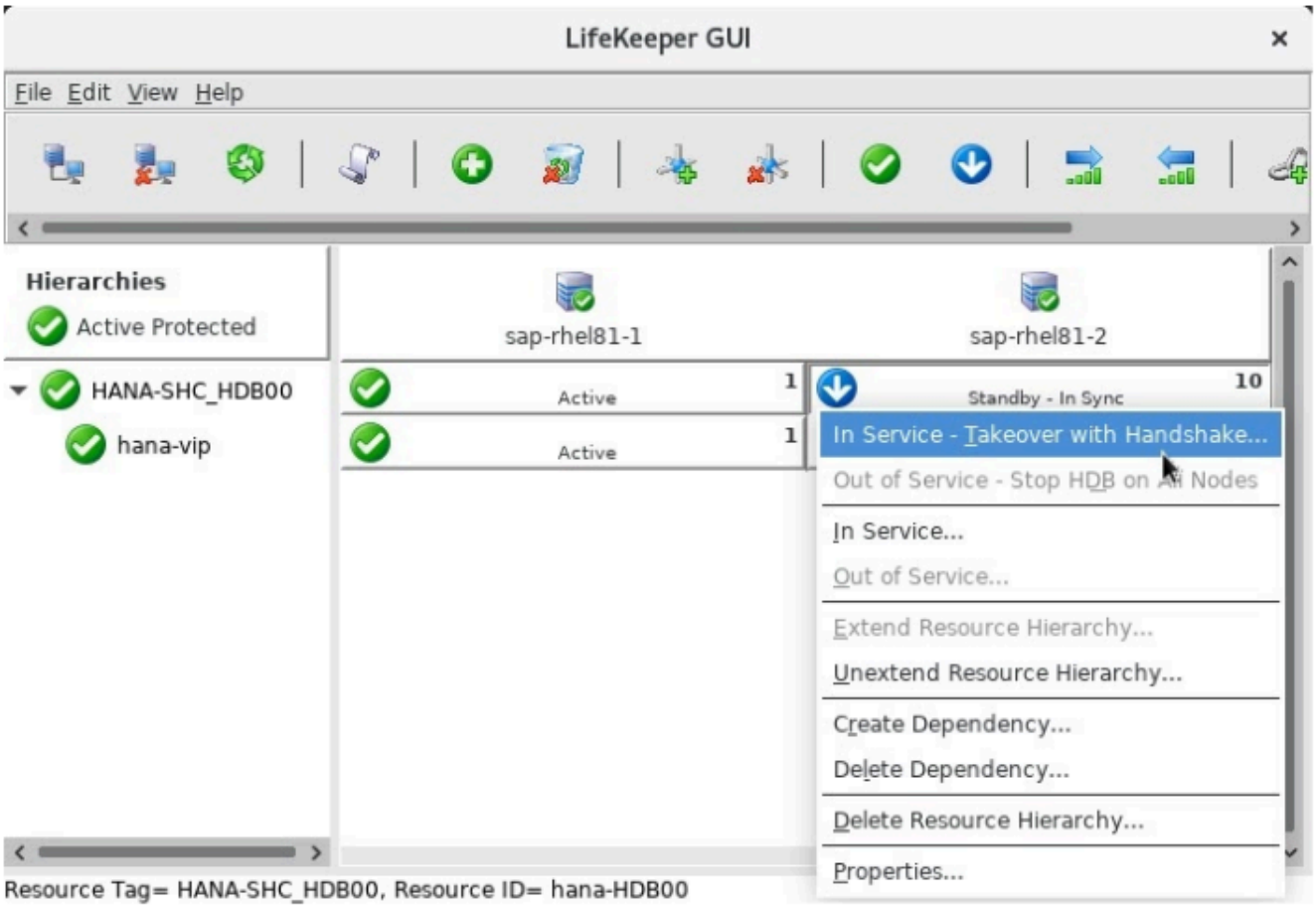
### 要件

Takeover with handshake を実行する前に、以下の要件を満たしている必要があります。:

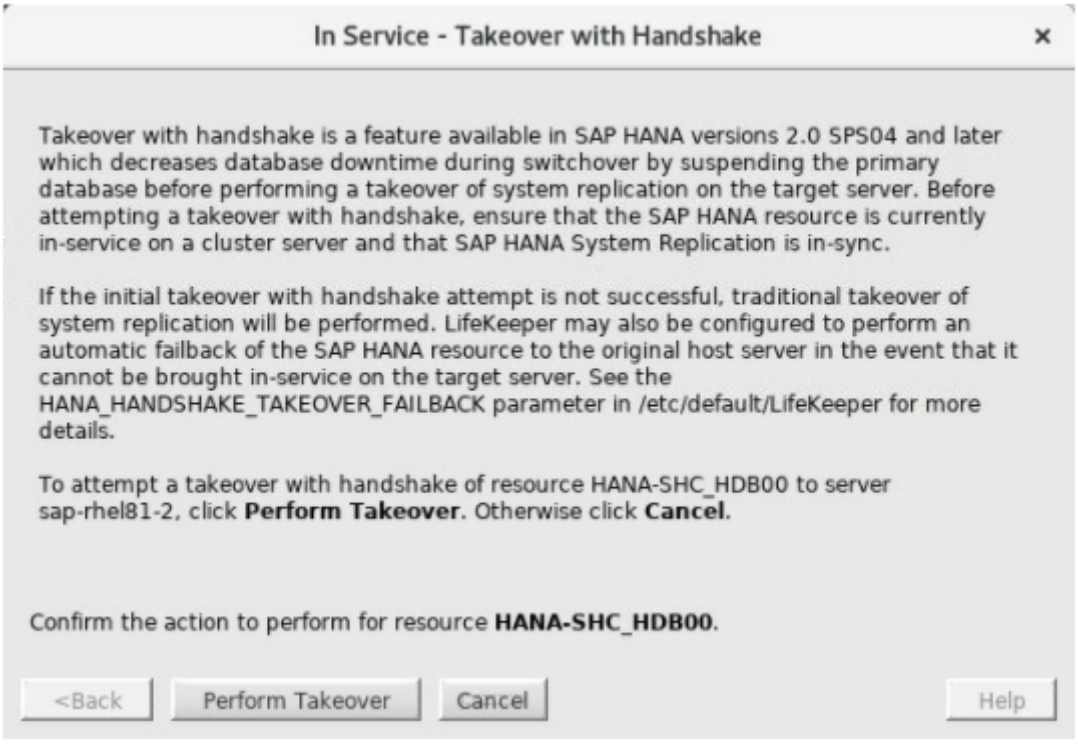
1. 各サーバーに SAP HANA 2.0 SPS04 以降がインストールされている。
2. クラスター内のサーバーで保護対象の SAP HANA データベースが In Service になっている。
3. SAP HANA システムレプリケーションが同期中になっている。

### LifeKeeper GUI でTakeover with Handshake を実行する

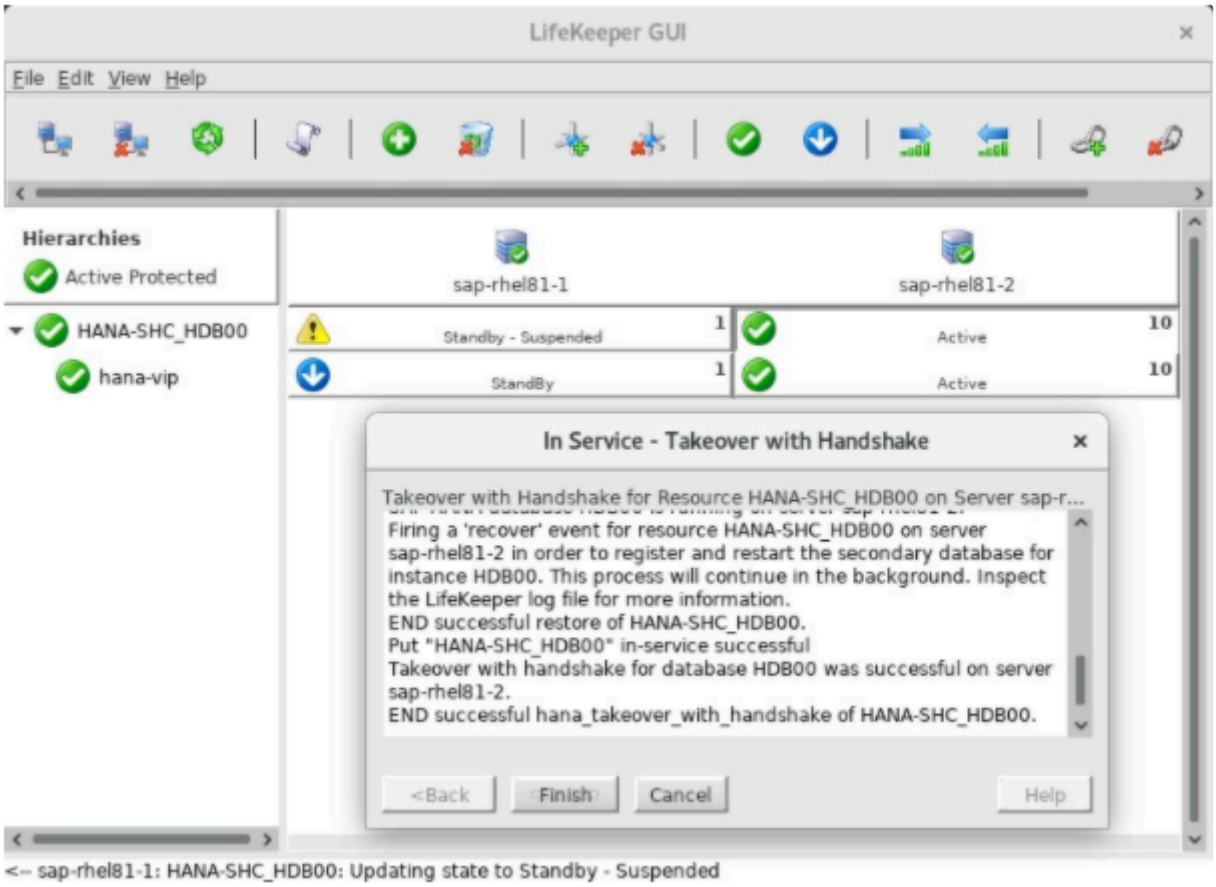
LifeKeeper GUI でTakeover with Handshake を実行するためには, スタンバイサーバーのリソース上で右クリックし、 “In Service – Takeover with Handshake…” を選択します。




Takeover with handshake の機能が説明された確認用ダイアログが表示されます。:




**Perform Takeover** をクリックしてTakeover with handshake を開始します。この処理が完了すると 各ノードの SAP HANA リソースのステータスが **Standby – Suspended** と **Active** へと切り替わります。




SAP HANA リソースがターゲットサーバーで In Service になったら、ローカルリカバリープロセスがバックグラウンドで自動的に実行され、セカンダリーデータベースインスタンスを停止、登録、再起動します。SAP HANA リソースを In Service にする間に、このセカンダリーデータベースの再起動のプロセスをフォアグラウンドで行いたい場合は、 /etc/default/LifeKeeper で、`HANA_REGISTER_SECONDARY_DURING_RESTORE=true` と設定してください。

ローカルリカバリープロセスが正常にセカンダリーデータベースインスタンスを再起動できたら、リソースステータスが **Standby – HDB Running** へ切り替わります。LifeKeeper がその後発生するquickCheck サイクル中にHRS 監視プロセスを開始し、HSR が In sync であることを判断したあと、リソースステータスは、 **Standby – In Sync** に切り替わります。この時点でSAP HANA データベースは高可用性を提供します。

## コマンドラインで Takeover with Handshake を実行する

以下のどちらかのコマンドを実行することによりTakeover with handshake をコマンドラインで実行することも可能です。:

- `/opt/LifeKeeper/bin/hana_takeover_with_handshake -t <HANA Tag> -s <Target Server> [-b]`
- `lkcli resource config hana –tag <HANA Tag> –takeover_with_handshake <Target Server>`

1番目のコマンドの -b パラメーターは、ターゲットサーバーでどの SAP HANA リソース階層を In Service にするかをコントロールします。-b オプションがない場合は、階層全体 (全ての親リソースと共有された

依存関係をもつリソース)が In Service になります。 -b オプションを付けると、指定された SAP HANA リソースとその依存関係が In Service になります。

### フェイルバックの動作を制御する

デフォルトでは SAP HANA リソースが、Takeover with handshake の実行中に In Service にならない場合はターゲットサーバーで **Out of Service – Failed (OSF)** の状態のままになり手動で In Service に戻すことが要求されます。このシナリオにおいて LifeKeeper は自動フェイルバックを行って、旧ホスト上で SAP HANA リソース階層を In Service に戻す設定をすることもできます。 自動フェイルバックは以下の設定を行うことで有効になります。/etc/default/LifeKeeper で `HANA_HANDSHAKE_TAKEOVER_FAILBACK=true` を設定してください。

## 一時停止したプライマリーデータベースを再開する

保護対象の SAP HANA データベースが、現在リソースが In Service になっているノードで一時停止される (例えば管理者が、LifeKeeper の外部で Takeover with handshake を手動で実施するためなどの理由で)と、 下記のようにリソースのステータスが、 **Active – Suspended** および **Standby – Primary** になります。

このステータスになっている間、すべての SAP HANA リソースの監視は一時停止され、下記のようなメッセージが記録され、問題が解決されるまでSAP HANA リソースが現在 In Service になっているシステムですべてのオープンになっているターミナルに対してブロードキャストされます。:

```
EMERG:hana:quickCheck:HANA-SHC_HDB00:136377:SAP HANA database HDB00
corresponding to resource HANA-SHC_HDB00 is currently suspended on server sap-
rhel81-1 due to actions performed outside of LifeKeeper. Please take the SAP
HANA resource out of service on server sap-rhel81-1 and bring it in-service on
the server where the database should be registered as primary master. Resource
monitoring for HANA-SHC_HDB00 will be suspended until the issue is resolved.
```

必要に応じて以下のコマンドを一時停止されているサーバー上で実行することでデータベースを再開することが可能です。:

```
su - <sid>adm -c "hdbnsutil -sr_resumeSuspendedPrimary"
```

データベース再開後、プライマリーとして登録されるスタンバイサーバーで以下のコマンドを実行して手動で停止する必要があります。:

```
su - <sid>adm -c "sapcontrol -nr <InstNum> -function StopWait 600 5"
```

スタンバイサーバーを新しいプライマリーサーバーにする場合は、SAP HANA リソースをSAP HANA リソースのステータスが **Standby – Primary** になっているサーバーへ切り替えることで解決できます。

# 6.17.7. SAP HANAトラブルシューティング

メッセージカタログ は、SIOS Protection Suite for Linuxの使用中に表示される可能性のあるすべてのメッセージのリストを掲載しています。また、エラーの原因の説明や、エラー状態を解決するために必要な処置も提供しています。このリストで、受信したエラーコードを検索することができます。または、SAP HANA Recovery Kitの使用中に表示される可能性のあるすべてのメッセージのリストが記載された SAP HANA Recovery Kit メッセージカタログ を直接確認しても構いません。

# 6.18. SAP MaxDB Recovery Kit 管理ガイド

SAP MaxDB は、幅広い用途に使用できる SQL ベースの非常に強力なデータベースシステムです。これは拡張性が高く、プラットフォームに依存しません。また、完全なトランザクションサポートを提供します。データベースシステムは元来 SAP が所有していましたが、現在はオープンソースコミュニティにリリースされています。

SAP MaxDB Recovery Kit は、SPS for Linux 環境における SAP MaxDB データベースの耐障害性保護を提供します。

## ドキュメントの内容

本マニュアルでは以下のトピックを取り上げ、SAP MaxDB の階層の定義および管理をサポートします。

- SAP MaxDB Recovery Kit の要件 。SAP MaxDB Recovery Kit の適切なセットアップ、インストールおよび操作を行うために必要なハードウェアとソフトウェアの一覧です。
- 概要 。SAP MaxDB Recovery Kitの特徴と機能について説明しています。
- 設定上の考慮事項 。SAP MaxDB Recovery Kit のインストールおよび設定前に考慮すべき情報を記載しています。
- SPS で SAP MaxDB を設定する 。SAP MaxDB ソフトウェアおよび SAP ソフトウェアのインストールおよび設定の手順を説明しています。
- リソース設定作業 。LifeKeeper GUI を使用して階層で実行できるさまざまな機能（作成、拡張、削除および拡張解除）について説明しています。
- リソース階層の管理 。SAP MaxDB 階層の継続的な管理に関する重要な推奨事項を記載しています。
- トラブルシューティング 。SAP MaxDB Recovery Kit に関連するエラーメッセージを一覧にして説明しています。
- 付録：Raw/IO によるデータベースデバイスの作成 。SAP MaxDB Recovery Kitで使用するためのRawデバイスの設定に関する要件と手順を説明しています。

## SIOS Protection Suite ドキュメンテーション

SIOS Technology Corp. の Web サイトでは、以下の SPS LifeKeeper 製品のドキュメンテーションを提供しています。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション

- SPS for Linux インストールガイド
- オプションの Recovery Kit ドキュメンテーション

このドキュメンテーション、およびオプションの LifeKeeper Recovery Kit 関連のドキュメンテーションは、次に示す SIOS Technology Corp の Web サイトから入手できます。

http://jpdocs.us.sios.com/

また、LifeKeeper GUI 内の [Help] メニューからも参照できます。

# SAP MaxDB ドキュメンテーション

インストールガイド、ユーザーマニュアル、リファレンスマニュアルなどの SAP MaxDB ドキュメンテーションは、以下のサイトから入手できます。

http://maxdb.sap.com/documentation/

# 6.18.1. SAP MaxDB Recovery Kit の要件

SPS の構成は、SPS for Linux SAP MaxDB Recovery Kit をインストールする前に、以下の要件を満たしている必要があります。SPS ハードウェアとソフトウェアのインストールと設定に関する具体的な手順については、SPS for Linux インストールガイド を参照してください。

## ハードウェア要件

- **サーバー** - サーバーは、SPS for Linux インストールガイド およびSPS for Linux リリースノート に記載されている要件に従って設定してください。

- **IP ネットワークインターフェースカード** - 各サーバーには、イーサネット TCP/IP 対応のネットワークインターフェースカードが少なくとも1つ必要です。また、SPS クラスターには最低2つのコミュニケーションパスが必要です。ハートビートには、独立した2つのサブネットを使用する2つの LAN ベースの通信パスが推奨され、少なくとも1つはプライベートネットワークとして構成する必要があります。TCP と TTY ハートビートの組み合わせもサポートしています。

## ソフトウェア要件

- **TCP/IP** ソフトウェア – SPS 構成の各サーバーにTCP/IP ソフトウェアが必要です。

- **SAP MaxDB** ソフトウェア –「SIOS Protection Suite for Linux サポートマトリックス」に記載されている、サポートされているバージョンの SAP MaxDB ソフトウェアをインストールする必要があります。

  **注記：** クラスター内のすべてのサーバーに同じバージョンの SAP MaxDB ソフトウェアをインストールする必要があります。

- **SPS** ソフトウェア – 同じバージョンの SPS ソフトウェアをインストールし、クラスター内の各サーバーに同じバージョンの SPS ソフトウェアパッチを適用してください。

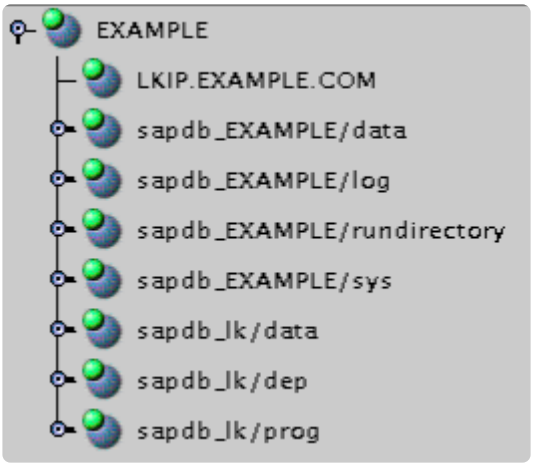
- **SPS for Linux SAP MaxDB Recovery Kit** – SAP MaxDB Recovery Kit は、SPS for Linux インストールイメージファイル（Steeleye-lkSAPDB）で提供しています。Red Hat Package Manager（RPM）を使用して、パッケージング、インストール、削除を行います。

# 6.18.2. 概要

SPS for Linux SAP MaxDB Recovery Kit は、SPS 内の SAP MaxDB インスタンスを保護するための仕組みを提供します。SAP MaxDB ソフトウェア、SPS Core、SAP MaxDB Recovery Kit は、クラスター内の複数のサーバーにインストールされています。SAP MaxDB データベースインスタンスが SPS の保護対象になると、クライアントは SPS で保護された IP アドレスを使用してデータベースに接続します。また、SPS で保護された IP アドレスを別途作成し、親 SAP MaxDB リソースインスタンスと子 IP アドレスリソース間に手動で依存関係を作成する必要があります。SAP MaxDBサーバーで障害が発生した場合、SPS はまずローカルサーバーで復旧を試みます。ローカルでの復旧に失敗した場合、SPS はバックアップサーバーにフェイルオーバーします。

# 6.18.2.1. SAP MaxDB リソース階層

以下に、一般的な SAP MaxDB リソース階層の例を示します。




上記の例での依存関係は、以下の保護されているリソースに対応しています。

| リソース | **SAP MaxDB** ソフトウェアコンポーネント |
|---|---|
| LKIP.EXAMPLE.COM | クライアント接続に使用される切り替え可能な IP アドレスを保護 |
| sapdb_EXAMPLE /data | EXAMPLE データベースのデータベース・データデバイス空間を保護 |
| sapdb_ EXAMPLE /log | EXAMPLE データベースのデータベース・ログデバイス空間を保護 |
| sapdb_ EXAMPLE /rundirectory | EXAMPLE データベースのデータベース RUNDIRECTORY を保護 |
| sapdb_ EXAMPLE /sys | EXAMPLE データベースのデータベース・システムデバイス空間を保護 |
| sapdb_lk/data | 独立したデータパスを保護 |
| sapdb_lk/dep | 従属したパスを保護 |
| sapdb_lk/prog | 独立したプログラムのパスを保護 |

フェイルオーバー時、SPS はファイルシステム、IP アドレスおよびデータベースリソース（すべてのリソースの依存関係を含む）をバックアップサーバー上で稼働させます。クライアントは切断されるため、サーバーに再接続する必要があります。コミットされていない SQL ステートメントはすべて再入力する必要があります。

# 6.18.3. 設定上の考慮事項

本セクションには、SAP MaxDB Recovery Kit の設定および管理を開始する前に考慮すべき情報を記載しています。

---

Raw I/O の使用

DataKeeper でのミラーファイルシステムの使用

アクティブ/スタンバイの場合の考慮事項

アクティブ/スタンバイ設定の例

アクティブ/アクティブの場合の考慮事項

アクティブ/アクティブ設定の例

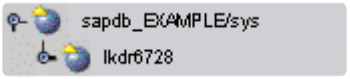
# 6.18.3.1. Raw I/O の使用

Raw デバイスで SAP MaxDB を使用する場合は、SPS のインストールイメージファイルから SPS Raw I/O Recovery Kit をインストールする必要があります。また、使用前に Raw I/O デバイスを適切に設定する必要があります。手順ついては、付録 を参照してください。

# 6.18.3.2. DataKeeper でのミラーファイルシステムの使用

SAP Max DB Recovery Kit は、デバイス領域としての DataKeeper の使用をサポートしています。さらに SAP MaxDB ソフトウェアは、ミラーリングされたファイルシステムにインストールすることができます。

例えば、SAP MaxDB リソースの依存ファイルシステムは次のようになりますが、これはシステムのデバイス領域とその依存関係のファイルシステム、DataKeeper リソースのミラーを示しています。



# 6.18.3.3. 内部向けロードバランサーの使用

SAP MaxDB Recovery Kit は Azure がサポートしているような内部向けロードバランサー (ILB) の使用をサポートします。ILB を有効にする場合は、LifeKeeper の設定ファイル (/etc/default/LifeKeeper) の MAXDB_ILB_ENABLED の値を 1 に設定してください。

ILBは、1つのサーバーのみで Xサーバーポートをアクティブにする必要があります。そのため、すべての SAP MaxDB リソースが同じ階層で、また同じノードでアクティブにすることになります。IndepDataPath および IndepProgPath が 共有ファイルシステム上にある場合、すべてのリソースは必然的に同一サーバー上に存在するはずです。IndepDataPath および IndepProgPath が共有ファイルシステムにない場合は、Generic Application リソースをリーフノードおよび/もしくはルートノードとして作成すると、同一サーバーで階層をアクティブにすることが可能です。

ILBの制限により、X サーバーはひとつのサーバー上でアクティブである必要があります。アクティブ/アクティブ構成は ILB でサポートされていません。

複数のSAP MaxDBリソースを単一の階層に配置するには、Generic Application リソースを最下位のリーフノードとして、および/または階層の最上位のルートノードとして別途作成してください。（Generic Application リソース階層の作成 に記述されている restore/remove スクリプトには、/bin/true を指定することができます。）

# 6.18.3.4. アクティブ/スタンバイの場合の考慮事項

アクティブ/スタンバイの構成では、バックアップサーバーは SAP MaxDB をアクティブに実行していませんが、プライマリサーバーの障害に備えて待機します。次のシナリオで、アクティブ/スタンバイ構成で SAP MaxDB リソースのインスタンスを保護する際に満たすべき具体的な要件を説明します。

## アクティブ/スタンバイのシナリオ

標準的なアクティブ/スタンバイ構成をシナリオ1およびシナリオ2で説明します。

## シナリオ1

SAP MaxDB の *IndepDataPath* 、 *IndepProgPath* および *DependPath* が **プライマリサーバー上の複数の共有ファイルシステムにインストールされている。**

- *IndepDataPath* 、 *IndepProgPath* および *DependPath* の3つのパスは、リソースインスタンスを保護するすべてのサーバー間で共有する必要があります。
- レジストリーファイル `/etc/opt/sdb` は、クラスター内の各サーバーに存在する必要があります。このファイルは、リソース拡張操作中にターゲットサーバーからアクセス可能でなければならないため、共有ストレージに置いてはいけません。
- データベースインスタンスのデータデバイス領域（`data devspaces`）、ログのデバイス領域（`log devspaces`）およびシステムのデバイス領域（`sys devspaces`）は、共有ディスク（共有ファイルシステムまたは共有 Raw デバイスのいずれか）上に存在する必要があります。

## シナリオ2

SAP MaxDB の *IndepDataPath* および *IndepProgPath* が両方のサーバーでローカルにインストールされている。SAP MaxDB *DependPath* はローカルまたはプライマリサーバー上の共有ファイルシステムにインストールできる。

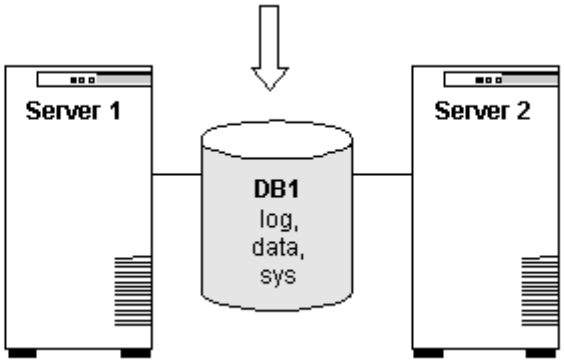
- レジストリーファイル `/etc/opt/sdb` は、クラスター内の各サーバーに存在する必要があります。このファイルは、リソース拡張操作中にターゲットサーバーからアクセス可能でなければならないため、共有ストレージに置いてはいけません。
- データベースインスタンスのデータデバイス領域（`data devspaces`）、ログのデバイス領域（`log devspaces`）およびシステムのデバイス領域（`sys devspaces`）は、共有ディスク（共有ファイルシステムまたは共有 Raw デバイスのいずれか）上に存在する必要があります。
- データベースインスタンス実行ディレクトリー(`RUNDIRECTORY`)は、共有ストレージに配置する必要があります。`RUNDIRECTORY` の値は、`DBMCLI` コマンドの `param_directput` を使用して変更できます。データベースの作成後に `RUNDIRECTORY` の値を変更した場合は、パラメーターの更新を反映するために、データベースインスタンスを停止して再起動する必要があります。

- データベースインスタンスの config (<IndepDataPath>/config) ディレクトリー構造は、データベースインスタンスが保護されるクラスター内のすべてのサーバー上の同じ場所に存在する必要があります。さらに、データベースインスタンスのパラメーターファイルは、テンプレート（またはプライマリ）サーバーからクラスター内のすべてのバックアップサーバーにコピーする必要があります。パラメーターの更新後には、パラメーターファイルをクラスター内のすべてのサーバーに再配布します。必要なファイルは以下の通りです。

```
config/Databases.ini

config/Installations.ini
```

# 6.18.3.4.1. アクティブ/スタンバイ設定の例

IndepData = /shr1/data
IndepPrograms = /shr1/programs
DependPath = /shr1/depend




**設定上の注意事項：**

- 両方のサーバーで共有ストレージ上の *IndepProgPath* 、 *DependPath* 、 *IndepDataPath* を使用します。
- データベースインスタンス DB1 は、共有ストレージにあります。これには、すべてのログデバイス領域、データデバイス領域およびシステムデバイス領域が含まれます。
- *Server 1* がアクティブな時は、 *Server 2* は共有ディスク上のファイルとディレクトリーにはアクセスできません。

# 6.18.3.5. アクティブ/アクティブの場合の考慮事項

アクティブ/アクティブ設定では、各サーバーは1つの SAP MaxDB インスタンスをアクティブに実行しており、障害発生時には他のサーバーのバックアップとして機能します。以下のシナリオでは、アクティブ/アクティブ設定で SAP MaxDB リソースインスタンスを保護する際に、順番に守る必要のある具体的な要件を示します。

内部ロードバランサーは、アクティブ/アクティブ構成ではサポートされていません。

## アクティ/アクティブのシナリオ

SAP MaxDB の *IndepDataPath* 、 *IndepProgPath* 、 *DependPath* が両方のサーバーにローカルにインストールされている。

- レジストリーファイル `/etc/opt/sdb` は、クラスター内の各サーバーに存在する必要があります。このファイルは、リソース拡張操作中にターゲットサーバーからアクセス可能でなければならないため、共有ストレージに置いてはいけません。

- データベースインスタンスのデータデバイス領域 (`data devspaces`)、ログのデバイス領域 (`log devspaces`) およびシステムのデバイス領域 (`sys devspaces`)は、共有ディスク（共有ファイルシステムまたは共有 Raw デバイスのいずれか）上に存在する必要があります。

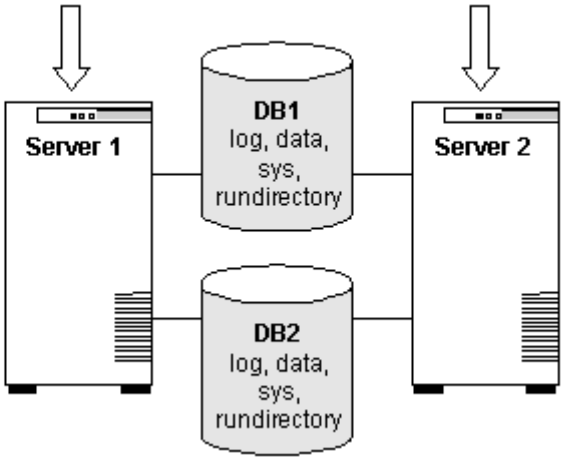
- データベースインスタンス実行ディレクトリー（`RUNDIRECTORY`）は、共有ストレージに配置する必要があります。`RUNDIRECTORY` の値は、`DBMCLI` コマンドの `param_directput` を使用して変更できます。データベースの作成後に `RUNDIRECTORY` の値を変更した場合は、パラメーターの更新を反映するために、データベースインスタンスを停止して再起動する必要があります。

- データベースインスタンスの`config(<IndepDataPath>/config)`ディレクトリー構造は、データベースインスタンスが保護されるクラスター内のすべてのサーバー上の同じ場所に存在する必要があります。さらに、データベースインスタンスのパラメーターファイルは、テンプレート（またはプライマリ）サーバーからクラスター内のすべてのバックアップサーバーにコピーする必要があります。パラメーターの更新後には、パラメーターファイルをクラスター内のすべてのサーバーに再配布します。必要なファイルは以下の通りです。

  ```
  config/Databases.ini

  config/Installations.ini
  ```

# 6.18.3.5.1. アクティブ/アクティブ設定の例

| | |
|---|---|
| IndepData = /usr/sapdb/data<br>IndepPrograms = /usr/sapdb/prog<br>DependPath = /usr/sapdb/dep | IndepData = /usr/sapdb/data<br>IndepPrograms = /usr/sapdb/prog<br>DependPath = /usr/sapdb/dep |




設定上の注意事項：

- *IndepDataPath* 、 *IndepProgPath* 、 *DependPath* は両方のサーバーにローカルにインストールします。
- 各データベースは別々の共有ディスクに構成されています。データベースインスタンスには、すべてのログデバイス領域、システムデバイス領域およびデータデバイス領域が含まれます。
- 各データベースの `RUNDIRECTORY` も共有ディスク上にあります。
- `DB1` のデータベース構成ファイルが *Server 2* にコピーされ、`DB2` のデータベース構成ファイルが *Server 1* にコピーされています。構成ファイルは `<IndepDataPath>/config/<db instance>` にあります。
- 最初に、 *Server 1* は `DB1` を実行し、 *Server 2* は `DB2`を実行します。スイッチオーバーの際には、1台のサーバーで両方のデータベースを実行できます。

# 6.18.4. SPS での SAP MaxDB の設定

SAP MaxDB データベースおよび SPS ソフトウェアのインストールと設定には、次の手順を推奨します。各ステップでは、詳細な作業を説明しています。

SAP MaxDB ソフトウェアのインストール

SAP MaxDB データベースの作成

User_Key の作成

SPS Core のインストールと SAP Max DB Recovery Kit

これらのタスクを実行すると、SPS リソース階層を作成して SAP MaxDB データベースを保護する準備が整います。

# 6.18.4.1. SAP MaxDB ソフトウェアのインストール

同じパラメーターと設定を使用して、クラスター内のすべてのサーバーに SAP MaxDB ソフトウェアをインストールします。詳細については、「SAP MaxDBインストールガイド」を参照してください。SAP MaxDB が SPS で確実に動作するようにするための推奨事項を以下に示します。

- root 以外 のシステムユーザー（OS ユーザー）は、次のようにすべてのサーバーに存在する必要があります。
- この OS ユーザーは、SAP MaxDB ソフトウェアのインストールとサブディレクトリの所有者として指定されているか、データベースインスタンスの管理に必要なソフトウェアのインストールパスとサブディレクトリに対して適切な権限を持っている必要があります。
- この OS ユーザーには、DBMCLI および x_server ユーティリティを使用する権限が必要です。また OS ユーザーは、x_server コマンドを使用して vserver の起動および停止を実行できる必要があります。
- OS ユーザー名に使用できるのは英数字のみです。
- この OS ユーザーのユーザー ID とグループ ID は、すべてのサーバーで同一でなければなりません。
- SAP MaxDB クライアントソフトウェアパッケージをインストールする必要があります。またこれらのパッケージには、SAP MaxDB DBMCLI クライアントユーティリティと SAP MaxDB x_server ユーティリティが含まれている必要があります。
- SAP MaxDB のリソース階層を含む各 SPS サーバーは、SAP MaxDB インスタンスの `/etc/services` ファイルに同じサービスエントリを持つ必要があります。

# 6.18.4.2. SAP MaxDB データベースの作成

「SAP MaxDB ユーザーマニュアル」の指示に従ってデータベースを作成してください。また、以下の推奨事項にも留意してください。

- クライアントユーティリティーを使用して、開始、停止、状態取得、データベースパラメーターの取得を実行する権限を持つ DBM オペレーターが必要です。
- データベースインスタンスのデータデバイス領域（`data devspaces`）、ログデバイス領域（`log devspaces`）およびシステムデバイス領域（`sys devspaces`）は、共有ディスク（共有ファイルシステムまたは共有 Raw デバイスのいずれか）上に存在している必要があります。
- SAP MaxDB データベース名には英数字のみ使用できます。
- User_Key は、DBMCLI ユーティリティーでの操作中に SAP MaxDB Recovery Kit で使用するために必要です。必要なパラメーターについては、 User_Key の作成 を参照してください。
- データベースの作成後、SAP MaxDB データベースインスタンスの自動起動を無効にする必要があります。 一旦 SPS の保護下に入ると、SPS がデータベースの開始と停止を処理します。
- `sdb` ファイルは、`/etc/opt` のすべてのサーバーに存在する必要があります。このファイルが存在しない場合、いくつかの SAP MaxDB ユーティリティーが誤った結果を返す可能性があります。これは、リソースの作成および拡張中の SPS の動作にも影響します。アクティブ/スタンバイ設定では、このファイルを手動でバックアップサーバーにコピーする必要があります。
- `databases.ini` ファイルが `IndepDataPath/config` ディレクトリーのすべてのサーバーに存在することを確認してください。
- LifeKeeper で設定した `databases.ini` ファイル内の各データベースエントリーをバックアップサーバにコピーしてください。

# 6.18.4.3. User_Key の作成

SAP MaxDB インスタンスには、ユーザーがデータベースインスタンスに正常にアクセスするためのいくつかのオプションが必要です。これらの必要な情報は、データベースインスタンスへのアクセスに使用されている SAP MaxDB ツールに渡す必要があります。SAP MaxDB ソフトウェアには、SAP MaxDB ツールに対する多くの必須オプションの指定を簡素化するための xuser ツールが含まれています。 xuser ツールを使用すると、ユーザーデータをあらかじめ定義して保存できます。このデータが保存されると、DBMCLI またはユーザーオプションを必要とする他のツールを呼び出すときにこのデータを使用できます。この事前定義されたユーザーデータは、ユーザーキー（User_Key）に保管されます。個々のユーザーは、同じデータベースまたは複数のデータベースの複数のユーザーキーを管理および保持できます。各キーには、ユーザー名/パスワード、データベース名、データベースサーバー名などのオプションの組み合わせが含まれています。

SAP MaxDB Recovery Kit では、保護されている各データベースインスタンスに対して有効なユーザーキーが必要です。この User_Key は、データベースインスタンスを所有する OS ユーザーによる作成およびアクセスが可能である必要があります。各User_Keyに対して指定されたユーザー情報は、次の権限を持つ DBM オペレーターのものでなければなりません。

- DBStart
- DBStop
- DBInfoRead
- ParamRead

User_Key は、<file name> が以下の表に記載する有効な XUSER エントリーを含む場合、次のコマンドで生成できます。

```
xuser -b <file name>
```

| パラメーター | パラメーターの定義 |
|---|---|
| USERKEY | User_Key のユニークな名前 |
| USERID | dbm オペレーターのユーザー名 |
| PASSWORD | ユーザーのパスワード |
| SERVERDB | このキーが参照するデータベースインスタンス名 |
| SERVERNODE | データベースを実行しているサーバー名（これは、DNS または SPS で保護されている IP のホストファイルエントリーである必要があります） |
| SQLMODE | 互換性のある SQL ダイアレクトを決定します |
| CACHELIMIT | 特定のセッションのキャッシュ制限を決定します |
| TIMEOUT | 非アクティブなセッションを終了するまでの時間（単位：秒）（デフォルトは-1） |
| ISOLATION | ユーザーに影響するロックに使用される分離レベルを決定します（デフォルトは-1） |

| | |
|---|---|
| DB_LOCAL | データベースのロケールを指定します |

パラメーターに関する詳細は、「SAP MaxDB ユーザーマニュアル」を参照してください。適切なエントリーが指定されたら、xuser ツールを使用して OS ユーザーのホームディレクトリに .XUSER.62 ファイルを生成します。2つのキーを含むサンプル XUSER ファイルは以下に含まれています（DEFAULT User_Key のエントリーが存在する必要があります）。

**DEFAULT**
**NULLDB**
**NULLDB**
**NULLDB**
**LKIP.example.com**
**INTERNAL**
**-1**
**-1**
**-1**
**my_locale**
**LK_USERKEY**
**LKDBMOPER**
**LKDBMPASSWD**
**DB1**
**LKIP.example.com**
**INTERNAL**
**-1**
**-1**
**-1**
**en_US**

この例の XUSER ファイルでは、 `DEFAULT` と `LK_USERKEY` の2つのユーザーキーを作成するように指定しています。 xuser ツールを実行して User_Key を生成したら、 xuser ツールで使用するよう指定されたファイルを削除する必要があります。

# 6.18.4.4. SPS ソフトウェアのインストール

SAP MaxDB ソフトウェアをインストールし、データベースを作成して User_Key を作成したら、SPS Core ソフトウェアと必要なパッチのインストール後に SAP MaxDB Recovery Kit をインストールします。また、Raw デバイスで SAP MaxDB を使用する場合は、SPS のインストールイメージファイルから SPS Raw I/O Recovery Kit をインストールする必要があります。Raw デバイスの設定に関する要件と手順については、付録 を参照してください。

SPS パッケージのインストールに関する詳細は、SPS for Linux インストールガイド を参照してください。

# 6.18.5. リソース設定作業

前のセクションで説明した設定作業を完了すると、SAP MaxDB リソース階層の作成および拡張の準備が整います。

SPS for Linux SAP MaxDB Recovery Kit の設定には、以下のタスクが利用可能です。

- **リソース階層の作成** - SAP MaxDB リソース階層を作成する
- **リソース階層の削除** - SAP MaxDB リソース階層を削除する
- **リソース階層の拡張** - SAP MaxDB リソース階層をプライマリサーバーからバックアップサーバーに拡張する
- **リソース階層の拡張解除** - SAP MaxDB リソース階層を SPS クラスター内の単一のサーバーから拡張を解除（削除）する

LifeKeeper Core のリソース階層の設定手順については、SIOS テクノロジーのウェブサイトに掲載している SPS for Linux テクニカルドキュメント を参照してください。

以下のタスクはすべての Recovery Kit で共通の手順であるため、SPS for Linux テクニカルドキュメント の 管理 セクションで説明しています。

- Create a Resource Dependency - 既存のリソース階層と別のリソースインスタンスとの間に親子関係を作成し、クラスター内のすべてのターゲットサーバーに依存関係の変更を伝播します。
- Delete a Resource Dependency - リソースの依存関係を削除し、クラスター内のすべてのターゲットサーバーに依存関係の変更を伝播します。
- In Service - リソース階層を特定のサーバー上でアクティブにします。
- Out of Service - 特定のサーバー上でリソース階層を非アクティブにします。
- View /Edit - 特定のサーバー上のリソース階層のプロパティの表示または編集を行います。

**注記：** このセクションの残りの部分では、設定は **[Edit]** メニューから実行します。また、次の操作によってほとんどのタスクを実行することができます。

- ツールバーから実行する
- **[Status]** 画面の左ペインにある **[global resource]** を右クリックする
- **[Status]** 画面の右ペインにある **[resource]** を右クリックする

右クリックを使用することにより、**[Edit]** メニューを使用して情報を入力しないで済みます。

# 6.18.5.1. SAP MaxDB リソース階層の作成

以下の手順をプライマリサーバーで実行してください。

1. **[Edit]** メニューの **[Server]** から **[Create Resource Hierarchy]** を選択します。

   *Create Resource Wizard* ダイアログボックスが表示されます。

2. ドロップダウンリストから **[SAP MaxDB Database]** を選択し、 **[Enter]** をクリックします。

3. 以下の情報の入力を要求されます。ダイアログボックスで **[Back]** ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生した際に、前に入力した情報を修正する必要がある場合に便利な機能です。また、いつでも **[Cancel]** をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| **Switchback Type** | インテリジェントまたは自動のいずれかを選択します。これにより、フェイルオーバー後にバックアップサーバー上で in-service（アクティブ）になった後の SAP MaxDB リソースのプライマリサーバーへのスイッチバック方法が決まります。インテリジェントスイッチバックでは、管理者がリソースをプライマリサーバーに戻す必要がありますが、プライマリサーバーがオンラインに戻って SPS のコミュニケーションパスが再確立されると、自動的にスイッチバックが行われます。<br><br>**注：** スイッチバック戦略は、SAP MaxDB リソースが使用する従属リソースのものと一致している必要があります。 |
| **SAP MaxDB Programs Directory** | このフィールドは、対応するサーバーの SAP `/etc/opt/sdb` ファイルにある SAP MaxDB プログラムパスがデフォルトで含まれています。別のディレクトリパスを入力することもできます。パス名に使用できる有効な文字は、文字、数字および特殊文字 - _ . / です。 |
| **SAP MaxDB Instance** | このフィールドには、デフォルトで、SPS 階層が存在しないシステム上にある最初の SAP MaxDB インスタンスの名前が含まれます。ドロップダウンリストには、システムで使用可能な他のデータベースインスタンスが表示されます。 |
| **SAP MaxDB System User** | SAP MaxDB コマンドを所有しているか、SAP MaxDB コマンドを実行する権限のあるシステムユーザーです。 このユーザーは、対応するサーバー上に存在する必要があります。選択画面で有効なユーザー名を入力します。 |
| **User_Key** | このフィールドには、XUSER User_Key のデフォルト値が含まれます。User_Key は、SAP MaxDB ツールで使用するデータベースユーザーのデータを保存するために使用されます。対応するサーバー、OS ユーザーおよびデータベースインスタンスの組み合わせに対して有効な User_Key を入力します。 |
| **SAP MaxDB** | これは、プライマリサーバー上の新しい SAP MaxDB データベースリソースの一意のタグ名です。デフォルトのタグ名は、SAP MaxDB インスタンスで構成されています。別の一意の |

| | |
|---|---|
| **Database Tag** | タグ名を入力することもできます。タグに使用できる有効な文字は、文字、数字および特殊文字 - _ . / です。 |

4. **[Create]** をクリックし、Create Resource Wizard で SAP MaxDB リソース階層を作成します。すると SPSは入力されたデータを検証します。SPS が問題を検出すると、情報ボックスにエラーメッセージが表示されます。

5. SAP MaxDB リソース階層を正常に作成したことを示すメッセージが表示され、フェイルオーバー保護を実施するためにクラスター内の別のサーバーにその階層を拡張する必要があります。 **[Next]** をクリックします。

6. **[Continue]** をクリックすると、SPS は *Pre-Extend Wizard* を表示します。リソース階層を他のサーバーに拡張する方法については、 SAP MaxDB リソース階層の拡張 のステップ2を参照してください。

# 6.18.5.2. SAP MaxDB リソース階層の拡張

この操作は、 **[Edit]** メニューから開始することも、 **[Create Resource Hierarchy]** オプションの完了時に自動的に開始することもできます。その場合は、以下の手順2を参照してください。

1. **[Edit]** メニューの **[Resource]** から **[Extend Resource Hierarchy]** を選択してください。 *Pre-Extend Wizard* が表示されます。拡張操作に慣れていない場合は、 **[Next]** をクリックしてください。SPS の **[Extend Resource Hierarchy]** のデフォルト値が分かっていて、入力と確認を省略する場合は **[Accept Defaults]** をクリックしてください。

2. *Pre-Extend Wizard* に以下の情報を入力します。

   注記： 最初の2つのフィールドは、 **[Edit]** メニューから拡張を開始した場合にのみ表示されます。

| フィールド | ヒント |
|---|---|
| **Template Server** | SAP MaxDB リソースが現在 in service のサーバーを選択します。 |
| **Tag to Extend** | 拡張する SAP MaxDB リソースを選択します。 |
| **Target Server** | 拡張先のサーバーを入力するか、選択してください。 |
| **Switchback Type** | これにより、フェイルオーバー後にバックアップサーバー上で in-service（アクティブ）になった後の SAP MaxDB リソースのプライマリサーバーへのスイッチバック方法が決まります。インテリジェントまたは自動のいずれかを選択できます。スイッチバックタイプは、後で必要に応じて **[Resource Properties]** ダイアログボックスの **[General]** タブで変更できます。<br><br>**注記:** スイッチバック戦略は、SAP MaxDB リソースが使用する従属リソースのものと一致している必要があります。 |
| **Template Priority** | **[Template Priority]** を選択または入力します。これはサーバーで現在 in service の SAP MaxDB 階層の優先順位です。優先順位は、1～999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。<br><br>**注記:** このフィールドは階層を最初に拡張するときだけ表示されます。 |
| **Target Priority** | これは、他のサーバーにある同等の階層に対する、新しく拡張する SAP MaxDB 階層の優先順位です。1～999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバーの優先順位を示します。SPS のデフォルトでは、階層が作成されたサーバーに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースに対して 2 つのサーバーで同じ優先順位を割り当てることはできません。 |

3. 拡張前のチェックが正常に終了したというメッセージが表示された後、**[Next]** をクリックしてください。

4. 拡張する階層に応じて、拡張されるリソースタグ (一部編集不可) を示す一連の情報ボックスが表示されます。

5. *Extend Wizard* に、次の情報を入力します。

| | |
|---|---|
| SAP MaxDB Programs Directory | このフィールドは、対応するサーバーの SAP `/etc/opt/sdb` ファイルにある SAP MaxDB プログラムパスがデフォルトで含まれています。パス名に使用できる有効な文字は、文字、数字および特殊文字 - _ . / です。 |
| User_Key | このフィールドには、XUSER User_Key のデフォルト値が含まれます。User_Key は、SAP MaxDB ツールで使用するデータベースユーザーのデータを保存するために使用されます。対応するサーバー、OS ユーザーおよびデータベースインスタンスの組み合わせに対して有効な User_Key を入力します。 |
| SAP MaxDB Database Tag | これは、ターゲットサーバー上の新しい SAP MaxDB データベースリソースの一意のタグ名です。デフォルトのタグ名は、SAP MaxDB インスタンスで構成されています。別の一意のタグ名を入力することもできます。タグに使用できる有効な文字は、文字、数字および特殊文字 - _ . / です。 |

6. 「階層の拡張が完了しました」というメッセージが表示されたら、[Next Server] をクリックして階層を別のサーバーに拡張するか、拡張しない場合は **[Finish]** をクリックします。

7. 「`Hierarchy Verification Finished`」というメッセージが表示されたら、**[Done]** をクリックします。

# 6.18.5.3. SAP MaxDB リソース階層の拡張解除

SPS クラスター内の単一サーバーからリソース階層を削除するには、次の手順を実行してください。

1. **[Edit]** メニューの **[Resource]** から **[Unextend Resource Hierarchy]** を選択します。
2. SAP MaxDB リソースの拡張を解除するターゲットサーバーを選択します。リソースが現在 in-service のサーバーを選択することはできません。（このダイアログボックスは、右ペインでリソースインスタンスを右クリックして **[Unextend]** タスクを選択した場合は表示されません）。 **[Next]** をクリックします。
3. 拡張を解除する SAP MaxDB 階層を選択し、 **[Next]** をクリックします。（このダイアログは、いずれかのペインでリソースインスタンスを右クリックして **[Unextend]** タスクを選択した場合は表示されません）。
4. ターゲットサーバーと拡張の解除を選択した SAP MaxDB リソース階層を確認する情報ボックスが表示されます。 **[Unextend]** をクリックします。
5. SAP MaxDB リソースの拡張解除が正常に完了したことを確認する別の情報ボックスが表示されます。 **[Done]** をクリックして、 **[Unextend Resource Hierarchy]** メニューを終了します。

# 6.18.5.4. SAP MaxDB リソース階層の削除

SPS 構成のすべてのサーバーから SAP MaxDB リソースを削除するには、次の手順を実行します。

1. **[Edit]** メニューの **[Resource]** から **[Delete Resource Hierarchy]** を選択します。
2. SAP MaxDB リソース階層を削除するターゲットサーバーを選択します。（グローバルリソースの左ペインまたは個別のリソースインスタンスの右ペインのいずれかを右クリックして **[Delete Resource]** タスクを選択した場合、このダイアログは表示されません。）
3. 削除する階層を選択します。（このダイアログは、左ペインまたは右ペインでリソースインスタンスを右クリックして **[Delete Resource]** タスクを選択した場合は表示されません。） **[Next]** をクリックします。
4. 削除するターゲットサーバーと階層を確認する情報ボックスが表示されます。 **[Next]** をクリックします。
5. SAP MaxDB リソースが正常に削除されたことを確認する別の情報ボックスが表示されます。
6. **[Done]** をクリックして終了します。

# 6.18.5.5. リソース階層のテスト

プライマリサーバーからバックアップサーバーへのリソースインスタンスのフェイルオーバーをシミュレートする手動のスイッチオーバーを開始することによって、SAP MaxDB リソース階層をテストすることができます。

## LifeKeeper GUI から手動スイッチオーバーを実行する

LifeKeeper GUI から手動でスイッチオーバーを開始するには、 **[Edit]** メニューの **[Resource]** から **[InService]** を選択します。例えば、バックアップサーバー上で in-service 要求が実行されると、SAP MaxDB のリソース階層はバックアップサーバー上で in-service 状態になり、プライマリサーバー上のサービスは停止されます。この時点で、元のバックアップサーバーはプライマリサーバーになり、元のプライマリサーバーがバックアップサーバーになります。

Out of Service 要求を実行すると、リソース階層は他のサーバー上でサービスを開始することなく、Out of Serviceとなります。

**重要：** リソース階層をバックアップサーバー上で In Service にした後、特に Raw デバイスをデバイス領域として使用する場合は、データベースへの接続を試みる必要があります。これは、すべてのディスクパーティションがバックアップサーバー上に表示され、Rawバインディングが正しく確立されていることを確認するために必要です。

Raw バインディングがバックアップサーバー上に確立されていない場合は、新しいパーティションがプライマリサーバーで作成されて構成に追加されているにもかかわらず、パーティションテーブルがバックアップサーバー上でまだ更新されていないことが原因である可能性が高いです。

これを解決するには、パーティションテーブルが正しく更新されるようにバックアップサーバーを再起動してください。

# 6.18.6. リソース階層の管理

リソース階層の作成後、次のタスクが必要になる場合があります。

---

User_Key の変更

OS User の変更

パラメーターの更新

# 6.18.6.1. User_Key の変更

既存の階層の User_Key を変更する必要がある場合は、その階層を削除して再作成する必要があります。

# 6.18.6.2. OS User の変更

データベースインスタンスを所有する OS User を変更する必要がある場合は、その階層を削除して再作成する必要があります。

# 6.18.6.3. パラメーターの更新

SAP MaxDB インスタンスのデータベースパラメーターが更新されたら、更新されたパラメーターファイルがインスタンスを保護しているすべてのサーバーに確実に再配布されるようにする必要があります。*IndepDataPath* が共有ディスク上にある場合、インスタンスを保護しているすべてのサーバーは自動的に更新されます。

# 6.18.7. トラブルシューティング

## 一般的なヒント

SAP MaxDB Recovery Kitの使用中に、次のエラーメッセージが表示される場合があります。

| エラーメッセージ | 解決策 |
| --- | --- |
| The tunable MAXDB_ILB_ENABLED is set in the LifeKeeper defaults file for Internal Load Balancer support. Protecting multiple SAP MaxDB resources in this configuration requires all SAP MaxDB resource hierarchies to share a common resource. | 各 SAP MaxDB リソースに対して共通するリソースに依存関係を作成、もしくは内部向けのロードバランサーのサポートを無効にしてください。Azure にもあるような内部向けロードバランサー (ILB) は、 1 つのサーバーのみで X サーバーポートをアクティブにする必要があります。LifeKeeper の SAP MaxDB リソースに対する in-service 操作により、X Server の処理が開始します。 通常の out-of-service の操作では、X サーバーの処理は停止しません。<br><br>LifeKeeperの設定ファイル（ */etc/default/LifeKeeper* ）で MAX_ILB_ENABLED を設定することで、ノード内最後の SAP MaxDB リソースが削除されたときに、X サーバープロセスも停止するようにサービス停止の動作を変更できます。この時、LifeKeeper が X サーバーを1つのサーバーのみでアクティブにしておくために、すべての SAP MaxDB リソースが同じ階層にある必要があります。<br><br>複数の SAP MaxDB リソースを単一の階層に配置するには、以下の手順で階層の最上位にルートノードとして Generic Application リソースを作成します。<br><br>• **[Edit]** メニューで、 **[Server]** 、 **[Create Resource Hierarchy]** の順に選択します。 **[Create Resource Wizard]** ダイアログが表示されます。<br>• ドロップダウンリストから **[Generic Application]** を選択し、 **[Enter]** をクリックします。<br>• Generic Application リソース階層の作成 の指示に従ってください。restore/ remove スクリプトは「/bin/true」を指定しても構いません。<br>• ルートリソースを作成したら、ルートリソースと各 SAP MaxDB リソースの間に依存関係を作成します。<br>◦ **[Edit]** メニューで、 **[Resource]** 、 **[Create Dependency]** の順に選択します。<br>◦ リソースが In-Service になっている サーバー を選択します。<br>◦ **[Parent Resource Tag]** にて、親リソースとして先ほど作成したルートリソースをドロップダウンリストから選択します。 |

| | |
|---|---|
| | ◦ **[Child Resource Tag]** にて、子リソースとして SAP MaxDB リソースを選択します。<br><br>ターミナルリーフノードを使用することもお勧めします。ユーティリティー上に単一のルートリソースを作成したら、create_terminal_leaf を使用して、階層の各ブランチに Generic Application リソースを作成できます。上記で作成したルートリソースのタグが「MaxDB_root」の場合、リーフノードを作成するには、「create_terminal_leaf MaxDB_root」を実行します。<br>ルートリソースとターミナルリーフリソースとの依存関係を作成して、新しい SAP MaxDB リソースを階層に追加する必要があります。 |
| Unable to create pipe `/usr/spool/sql/fifo/<db instance>` | `/usr/spool/sql` ディレクトリには、データベースインスタンスを所有するシステムユーザーのアクセスを許可するための適切な権限が必要です。 |
| Open device space *<dev>* permission denied | バックアップとプライマリ上のデバイス領域では、同じユーザーおよびグループの権限に加え、同じ所有者が必要です。 |
| Unable to set uid on startup | `<DependPath>/pgm/dbmsrv` の setuid ビットを設定する必要があります。ファイルの所有者は SAP MaxDB システムユーザーでなければなりません。 |
| runtime environment error | 考えられる原因に対し、それぞれ解決策が存在します。<br>• **データベースインスタンスのパラメーターと設定ファイルが存在しません。**<br>データベースパラメーターファイルを作成するか、テンプレートサーバーからファイルをコピーします。<br>• **データベースにライブラリの問題が発生しました。**<br>サーバーとソフトウェアインストールの組み合わせの場合、*libpthread-0.8.so* ライブラリを使用する必要があります。 手順については、SAP MaxDB のマニュアルを参照してください。<br>• **データベースインスタンスの環境が壊れています。**<br>プロセスを手動で強制終了する必要があります。その後、リソースを in-service 状態に復元してください。 |
| open Registry: Permission denied | `/usr/spool/sql/ini` は、SAP MaxDB ソフトウェアを所有しているユーザーおよびグループが所有する必要があります。さらに、ユーザーおよびグループはこのディレクトリで読み取り/書き込みのアクセス権限を持っている必要があります。 |
| **ERR_USRREAD**: could not read user data | `<IndepDataPath>/config/<db instance>` の設定ファイルがサーバーに存在しないか、適切なアクセス権限がありません。データベースインスタンスを所有しているシステムユーザーに対して適切なアクセス権限が設定されたファイルが存在することを確認してください。 |

# 6.18.7.1. SAP MaxDB Recovery Kit エラーメッセージ

| エラー番号 | メッセージ |
|---|---|
| 111000 | Usage: %s independent_program_path <validate:value_1:…:value_n:> |
| 111001 | Usage: %s %s %s |
| 111002 | No value specified to script %s for input argument %s. |
| 111003 | User %s with User_Key %s cannot access instance %s.<br><br>Action: Specify a User_Key for the given user with database access rights. |
| 111004 | The user %s does not exist on the server %s. |
| 111005 | The SAP DB instance %s is not running on server %s. |
| 111006 | Database Manager Utilities were not found in the specified path %s. |
| 111007 | An SPS internal error occurred in utility %s.<br><br>Action: Retry operation. |
| 111008 | Unable to obtain %s device space information for SAP DB instance %s for user %s and User_Key %s.<br><br>Action: Verify that the user and User_Key are valid for the corresponding database instance. |
| 111009 | Unable to create raw resource hierarchy for %s.<br><br>**Action: Verify that the underlying device is a shared device.** |
| 111010 | Unable to create filesystem resource hierarchy for %s.<br><br>Action: Verify that the underlying device is a shared device. |
| 111011 | Unable to determine the type of the dev space or install path %s. |

| | |
|---|---|
| | Action: Valid dev space types include file system and/or raw devices. |
| 111012 | The path %s is not on a shared filesystem . |
| 111013 | The SAP DB instance %s is already under SPS protection on server %s. |
| 111014 | The SAP DB instance %s has been successfully started on server %s. |
| 111015 | The SAP DB instance %s has been successfully stopped on server %s. |
| 111016 | Unable to start SAP DB instance %s on server %s. |
| 111017 | Unable to stop SAP DB instance %s on server %s. |
| 111018 | Attempting db_warm for database instance %s after db_start failure. |
| 111019 | The SAP DB x_server has been successfully started on server %s. |
| 111020 | The SAP DB x_server has been successfully stopped on server %s. |
| 111021 | The SAP DB x_server is not running on server %s |
| 111022 | Unable to start SAP DB x_server on server %s.<br><br>Action: A problem has occurred using the x_server utility, check the SAP DB logs and correct the problem. |
| 111023 | Unable to stop SAP DB x_server on server %s.<br><br>Action: A problem has occurred using the x_server utility; check the SAP DB logs and correct the problem. |
| 111024 | The SAB DB file SAP_DBTech.ini was not found on server %s.<br><br>Action: Verify that SAP DB is installed correctly on the specified server. |
| 111025 | The user id for user %s is not the same on server %s and %s. |
| 111026 | The group id for user %s is not the same on server %s and %s. |
| 111027 | The service file entries for are not the same on server %s and %s. |
| 111028 | One or more of the SAP DB service file entries do not exist on server %s. |
| 111029 | No dependents were found for resource %s on server %s. |
| 111064 | The tunable MAXDB_ILB_ENABLED is set in the LifeKeeper defaults file for Internal Load Balancer support. Protecting multiple SAP MaxDB resources in this configuration requires all |

| | |
|---|---|
| | SAP MaxDB resource hierarchies to share a common resource.<br><br>**Action: Please refer to SAP MaxDB Troubleshooting.** |

# 6.18.8. 付録：Raw I/O によるデバイス領域を作成する

## 要件

Raw I/O で SAP MaxDB Recovery Kit を使用するには、次の要件を満たす必要があります。

- Linux OS はRaw I/O デバイスをサポートしている必要があります。多くのディストリビューションでこのサポートはカーネル2.4に含まれていましたが、カーネル2.2で Raw I/Oをサポートするディストリビューションもあります。
- すべての Raw I/O デバイスは、共有ディスクパーティションにバインドされている必要があります。Raw I/O デバイスに配置されるデバイス領域 *（devspace）* の数によって、必要な Raw デバイスと共有ディスクパーティションの正確な数が決まります。Raw デバイス上で *devspaces* を作成する方法については、SAP MaxDB マニュアルを参照してください。
- SAP MaxDB ソフトウェアのバージョンは、Raw I/O デバイスの使用をサポートしている必要があります。

# 6.18.8.1. Raw I/O セットアップ手順

1. SAP MaxDB のデバイス領域に適したサイズの共有ディスクパーティションを選択します。

2. 未使用の Raw デバイスノードをこのパーティションにバインドします。これはマシンがリブートされてルートアクセスが必要な時に毎回実行する必要があるため、システム初期化ファイル（rc.local または boot.local）に Raw バインディングを追加しても構いません。階層が SPS 保護下に置かれたら、これらのバインディングはファイルから削除する必要があります。SPSは、SPS保護下にある Raw I/Oデバイスの Raw バインディングを再確立します。どの Raw デバイスノードがすでに使用されているかを確認するには、以下の例のように raw -qa コマンドを実行します。
   ```
   raw -qa
   raw /dev/raw/raw1 /dev/sda1
   ```

3. Rawデバイスコントローラ（/dev/raw/rawctl または /dev/raw）およびデータベースインスタンスを保護するすべてのサーバー上のディスクパーティションの両方に、グローバル読み取りアクセス権を設定します。
   ```
   chmod a+r /dev/raw/rawctl
   ```

4. データベースインスタンスを保護するすべてのサーバー上の Raw デバイスに対するグループおよびユーザーの読取り/書込み権限を設定します。
   ```
   chmod 664 /dev/raw/raw1
   ```

5. データベースインスタンスを保護するすべてのサーバー上で、指定されたデータベースインスタンスの Raw デバイスの所有者を SAP MaxDB OS Userに変更します。
   ```
   chown -R sapdb:sapdb /dev/raw/raw1
   ```

6. param_adddevspace または db_adddevspace を使用して、データベースにデバイス領域を追加します。SAP MaxDB ユーザーマニュアル、データベースマネージャー CLI マニュアルを参照してください。

# 6.18.8.2. 階層作成後のデバイス領域の追加

SPS で SAP MaxDB 階層を作成した後にRaw I/O デバイスまたは共有ファイルシステムにテーブルスペースを追加する場合は、LifeKeeper GUI を使用して Raw I/O デバイスまたはファイルシステムのリソース階層を手動で作成する必要があります。新たに作成されたリソース階層は、SAP MaxDB リソース階層に従属している（子になっている）必要があります。必要に応じて、更新されたパラメーターファイルを、データベースインスタンスを保護するすべてのサーバーに再配布する必要があります（*IndepDataPath* が共有ディスク上にある場合は必要ありません）。

# 6.19. Sybase Recovery Kit 管理ガイド

## Sybase ASE Recovery Kit ドキュメンテーション

Sybase Adaptive Server Enterprise (以下 Sybase ASE) は、高性能ビジネスアプリケーション向けの強力なデータ管理プラットフォームです。Sybase ASE は、OLTP 負荷の処理を特に得意とする汎用的なエンタープライズクラスの RDBMS です。金融、オンラインビジネス、その他の技術領域で幅広く使用されています。Sybase ASE プラットフォームは、Adaptive Server、Monitor Server、Backup Server、その他のプラグインコンポーネントを含む多くの標準コンポーネントで構成されています。Adaptive Server コンポーネントは、リレーショナルデータベースサーバです。Monitor Server は、Adaptive Server データベースサーバとは独立したサーバであり、Adaptive Server を監視します。Monitor Server は、リアルタイムデータまたはヒストリカルデータをクライアントアプリケーションに提供できます。Backup Server は、Open Server ベースのアプリケーションで、Adaptive Server でのデータベースのバックアップ (ダンプ) とリストア (ロード) をすべて管理します。

SIOS Protection Suite for Linux Sybase ASE Recovery Kit は、Sybase ASE コンポーネントの Adaptive Server、Monitor Server、Backup Server に対して SPS によるリソース保護を提供します。

## SIOS Protection Suite ドキュメンテーション

以下は、SIOS Technology Corp. のドキュメンテーション Web サイトで提供されている SIOS Protection Suite for Linux 関連ドキュメントの一覧です。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- Optional Recovery Kit ドキュメンテーション

## Sybase ASE のドキュメンテーション

*Installation Guide Adaptive Server for Linux*、*User Manual* 、 *Monitor Server User Manual* 、*Troubleshooting Guide* 、Reference Manual を含む Sybase ASE ドキュメンテーションは、Web サイトの次の場所にあります。

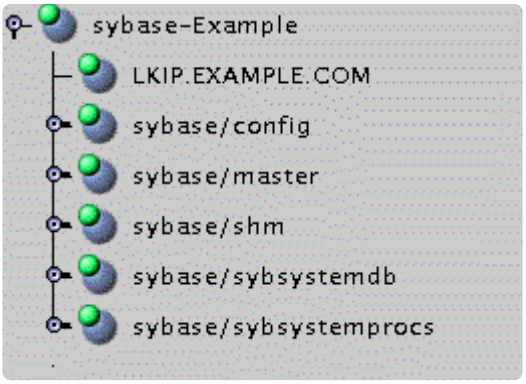
Sybase 製品のドキュメンテーション

# 6.19.1. 概要

LifeKeeper for Linux Sybase ASE Recovery Kit は、LifeKeeper 内の Sybase ASE Server インスタンスを保護する機能を提供します。Sybase ASE ソフトウェア、LifeKeeper Core、および Sybase ASE Recovery Kit は、クラスタ内の 2 台以上のサーバにインストールされます。Sybase ASE サーバインスタンスが LifeKeeper の保護下に入ったら、クライアントは LifeKeeper が保護する IP アドレスを使用してデータベースに接続します。LifeKeeper が保護する IP アドレスは、Sybase ASE リソース階層の作成前に別に作成する必要があります。Sybase ASE リソース階層の作成では、親 Sybase ASE リソースインスタンスと子 IP アドレスリソース間に依存関係を作成します。Sybase ASE サーバインスタンスに障害が発生した場合、LifeKeeper はまずローカルサーバ上でのリカバリを試みます。ローカルリカバリに失敗した場合、バックアップサーバにフェイルオーバします。

## Sybase ASE リソース階層

以下に示すのは、Sybase ASE リソース階層の一例です。




上記の例の依存関係は、以下の保護されるリソースに対応しています。

| リソース | **Sybase ASE** ソフトウェアコンポーネント |
|---|---|
| LKIP.EXAMPLE.COM | クライアント接続に使用される切り替え可能な IP アドレスを保護します。 |
| sybase/config | Sybase Adaptive Server、Monitor Server、Backup Serverの設定ファイルが格納されたファイルシステムを保護します。 |
| sybase/master | Sybase ASE マスターデバイスを保護します。 |
| sybase/shm | Sybase Adaptive Server と Monitor Server 共有メモリパスを保護します。 |
| sybase/sybsystemdb | Sybase ASE sybsystemdb デバイスを保護します。 |
| sybase/ sybsystemprocs | Sybase ASE sybsystemprocs デバイスを保護します。 |

フェイルオーバが発生した場合、LifeKeeper はバックアップサーバでファイルシステム、IP アドレス、およびデータベースリソース (すべてのリソース依存関係を含む) を In Service の状態にします。クライアントは切断され、サーバに再接続する必要があります。まだコミットされていない SQL ステートメントはす

べて再入力する必要があります

# 6.19.2. ハードウェア要件とソフトウェア要件

## Sybase ASE Recovery Kit の要件

以下に、Sybase ASE Recovery Kit を適切に設定、インストール、運用する上で必要なハードウェアとソフトウェアを挙げます。

LifeKeeper for Linux Sybase Recovery Kit をインストールする前に、LifeKeeper の構成が次の要件を満たしている必要があります。LifeKeeper のハードウェアとソフトウェアのインストールと設定に関する具体的な手順については、 SPS for Linux Installation Guide を参照してください。

### ハードウェア要件

- サーバ - サーバは、要件にしたがって設定する必要があります。要件は、 SPS for Linux テクニカルドキュメンテーション 、および SPS for Linux リリースノート に記載されています。

- **IP** ネットワークインターフェースカード - 各サーバは、イーサネット TCP/IP をサポートするネットワークインターフェースカードを 1 つ以上必要とします。ただし、LifeKeeper クラスタは、少なくとも 2 つのコミュニケーションパスを必要とすることに注意してください。LAN を使用した 2 つのコミュニケーションパスは、ハートビート用に別々のサブネットに接続することをお勧めします。また、少なくともどちらか 1 つのサブネットは、プライベートネットワークとして設定します。TCP と TTY のハートビートの組み合わせもサポートされています。

- ストレージ – サーバは、LifeKeeper でサポートされている共有ストレージ、もしくは LifeKeeper for Linux Data Replication ストレージを使用するように設定する必要があります。

### ソフトウェア要件

- **TCP/IP** ソフトウェア – LifeKeeper 環境の各サーバには TCP/IP ソフトウェアが必要です。

- **Sybase ASE** ソフトウェア – LifeKeeper は、 Sybase ASE ソフトウェアのバージョン 15.5 およびそれ以降をサポートしています。このバージョンは、Sybase Inc. の http://www.sap.com/japan/index.html から入手できます。 **注記:** クラスタ内の全サーバに同じバージョンの Sybase ASE ソフトウェアをインストールする必要があります。また、LifeKeeper の保護下にあるサーバ上にインストール可能な Sybase ASE ソフトウェアバージョンは 1 つのみです。

- **LifeKeeper** ソフトウェア – クラスタ内の各サーバにインストールする LifeKeeper ソフトウェアのバージョン、および、適用する LifeKeeper ソフトウェアパッチのバージョンが同一であることが絶対条件です。

- **LifeKeeper for Linux IP Recovery Kit** – LifeKeeper for Linux IP Recovery Kit は、LifeKeeper for Linux Sybase ASE Recovery Kit により必要とされます。LifeKeeper for Linux IP Recovery Kit は、FTP ダウンロードにより LifeKeeper for Linux Core イメージファイルで提供されます。

- **LifeKeeper for Linux Sybase ASE Recovery Kit** – Sybase ASE Recovery Kit は、FTP ダウンロードによりイメージファイルで提供されます。Red Hat Package Manager (rpm) でパッケージされ

ており、rpm を使用してインストール / アンインストールを行います。LifeKeeper for Linux Sybase ASE Recovery Kit の FTP ダウンロードによるイメージファイルには、次の rpm ファイルが格納されています。

**steeleye-lkSYBASE**

# 6.19.3. 設定上の考慮事項

以下は、Sybase ASE Recovery Kit のインストールと設定を行う前に考慮すべき情報です。

---

raw I/O の使用

DK4L でのミラーファイルシステムの使用

interfaces ファイルの考慮事項

Sybase Software Asset Manager (SySAM)

アクティブ / スタンバイの考慮事項

アクティブ / アクティブの考慮事項

Sybase Monitor Server と Backup Server の考慮事項

Network Attached Storage (NAS) を使用した LifeKeeper 保護下の Sybase

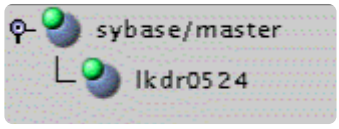
# 6.19.3.1. raw I/O の使用

raw デバイスで Sybase ASE を使用することを予定している場合は、LifeKeeper Raw I/O Recovery Kit を LifeKeeper Core イメージファイルからインストールする必要があります。また、使用前に raw I/O デバイスを適切に設定する必要もあります。手順については、raw I/O によるデータベースデバイスの作成 を参照してください。

# 6.19.3.2. DataKeeper でのミラーファイルシステムの使用

Sybase ASE Recovery Kit では、SIOS DataKeeper をデバイス空間として使用できます。また、Sybase ASE ソフトウェアをミラーファイルシステムにインストールできます。

例えば、Sybase ASE リソースの依存ファイルシステムは次のようになります。下図は、DataKeeper リソースミラーで、システムデバイス空間のファイルシステムとその依存関係を示しています。



# 6.19.3.3. interfaces ファイルの考慮事項

Sybase ASE Recovery Kit は、Sybase ASE interfaces ファイルを使用して、クライアントの IP アドレスとポートを検出します。このファイルは $SYBASE 配下にあり、通常、interfaces と呼ばれます。このファイルは、srvbuild または同様の設定ユーティリティを使用して Adaptive Server、Monitor Server、または Backup Server インスタンスが作成されるたびに更新されます。LifeKeeper for Linux Sybase ASE Recovery Kit では、このファイルに保護対象の各 Sybase ASE コンポーネントに対するエントリが含まれることを必要とします。コメント行は使用できません。interfaces ファイルに記述するすべてのサーバ名は、有効な仮想 IP アドレスに解決可能である必要があります。Sybase ASE リソース階層を保護するすべてのサーバは、interfaces ファイルに記述されているサーバ名を解決できる必要があります。また、仮想 IP アドレスをサーバ名の代わりに使用することを推奨します。

例

- master tcp ether example.com 4100
- query tcp ether example.com 4100

**Example_back**

- master tcp ether example.com 4200
- query tcp ether example.com 4200

**Example_mon**

- master tcp ether example.com 4200
- query tcp ether example.com 4200

*interfaces ファイルのサンプル*

# 6.19.3.4. Sybase Software Asset Manager (SySAM)

Sybase Software Asset Management (SySAM) を使用すると、ライセンスされた Sybase 製品を管理できます。Sybase ASE サーバの起動時に、各 ASE サーバコンポーネントは、特定の機能を実行するための権限があるかどうか、その環境のライセンスファイルを確認します。ASE サーバがこの作業を実行するためには、ライセンスマネージャとベンダモジュールが実行中である必要があります。LifeKeeper for Linux Sybase ASE Recovery Kit は、SySAM ライセンスマネージャに対する保護を提供していません。ライセンスマネージャは、冗長化されたサーバシステム内に設定することを推奨します。冗長化されたサーバシステムでは、冗長化されたライセンスにより、ローカルサーバをキューの 1 番目のライセンスサーバとして指定し、リモートサーバをバックアップライセンスサーバに指定することができます。SySAM アプリケーションは、まず 1 番目のサーバのライセンスファイルからライセンスをチェックアウトしようとします。何らかの理由でそのサーバに障害が発生した場合、リストの 2 番目にあるサーバにアクセスし、以下同様に実行されます。冗長化されたライセンスサーバ環境のユーザプロファイルで、LM_LICENSE_FILE 変数を正しく設定する必要があります。

# 6.19.3.5. アクティブ / スタンバイの考慮事項

アクティブ / スタンバイ設定では、バックアップサーバは Sybase ASE をアクティブに実行せず、プライマリサーバに障害が発生した場合に備えてスタンバイします。以下に、アクティブ / スタンバイ設定で Sybase ASE リソースインスタンスを保護する場合に従わなければならない具体的な要件を示します。

## シナリオ 1

Sybase ASE 製品は、クラスタ内のすべてのサーバにローカルでインストールされている。

- すべての Sybase Adaptive Server、Monitor Server、Backup Server デバイスは、共有ストレージ上に設定されている。
- Sybase Adaptive Server、Monitor Server、Backup Server の設定ファイルは、共有ファイルシステム上に格納されている。
- Sybase Adaptive Server と Monitor Server の共有メモリディレクトリは、共有ファイルシステム上に置かれている。
- interfaces ファイルは、保護対象の各インスタンスに対する共通のエントリを含むよう、すべてのサーバ上で手作業で更新しなければならない。
- すべての interfaces ファイルエントリは、リソースが保護されるすべてのサーバ上で解決可能でなければならない。
- 各 Adaptive Server、Monitor Server、Backup Server 用の RUN ファイルは、クラスタ内のすべてのサーバ上に存在しなければならない。
- 各 Adaptive Server、Monitor Server、Backup Server 用の RUN ファイルは、クラスタ内のすべてのサーバ上で実行可能でなければならない。
- 各 Adaptive Server、Monitor Server、Backup Server 用の RUN ファイルは、クラスタ内のすべてのサーバ上で同一のオプションを含んでいなければならない。

## シナリオ 2

Sybase ASE 製品は、プライマリサーバ上の 1 つまたは複数の共有ファイルシステムにインストールされている。

- すべての Sybase Adaptive Server、Monitor Server、Backup Server デバイスは、共有ストレージ上に設定されている。
- Sybase Adaptive Server、Monitor Server、Backup Server の設定ファイルは、共有ファイルシステム上に格納されている。
- Sybase Adaptive Server と Monitor Server の共有メモリディレクトリは、共有ファイルシステム上

に置かれている。

- interfaces ファイルは、ターゲットサーバで更新する必要はない。
- すべての interfaces ファイルエントリは、リソースが保護されるすべてのサーバ上で解決可能でなければならない。
- LifeKeeper Backup Server において、/etc/ld.so.conf にSybase 製品ライブラリのエントリを追加して更新する必要がある。
    - º $SYBASE/ASE/lib のエントリを追加する。
    - º $SYBASE/OCS/lib のエントリを追加する。
    - º Sybase

ASE インストール済み製品を含む共有ファイルシステムをマウントして、ldconfig を実行する。

# 6.19.3.6. アクティブ / アクティブの考慮事項

アクティブ / アクティブ設定では、障害時、他の LifeKeeper サーバに対するバックアップとして各サーバがアクティブに 1 つまたは複数の Sybase ASE サーバを実行します。以下に、アクティブ / アクティブ設定で Sybase ASE リソースインスタンスを保護する場合に、順を追って従わなければならない具体的な要件を示します。

## シナリオ 1

Sybase ASE 製品は、クラスタ内のすべてのサーバにローカルでインストールされている。

- すべての Sybase Adaptive Server、Monitor Server、Backup Server デバイスは、共有ストレージ上に設定されている。
- Sybase Adaptive Server、Monitor Server、Backup Server の設定ファイルは、共有ファイルシステム上に格納されている。
- Sybase Adaptive Server と Monitor Server の共有メモリディレクトリは、共有ファイルシステム上に置かれている。
- interfaces ファイルは、保護対象の各インスタンスに対する共通のエントリを含むよう、すべてのサーバ上で手動で更新しなければならない。
- すべての interfaces ファイルエントリは、リソースが保護されるすべてのサーバ上で解決可能でなければならない。
- 各 Adaptive Server、Monitor Server、Backup Server 用の RUN ファイルは、クラスタ内のすべてのサーバ上に存在しなければならない。
- 各 Adaptive Server、Monitor Server、Backup Server 用の RUN ファイルは、クラスタ内のすべてのサーバ上で実行可能でなければならない。
- 各 Adaptive Server、Monitor Server、Backup Server 用の RUN ファイルは、クラスタ内のすべてのサーバ上で同一のオプションを含んでいなければならない。

## シナリオ 2

Sybase ASE 製品は、プライマリサーバ上の 1 つまたは複数の共有ファイルシステムにインストールされている。

- すべての Sybase Adaptive Server、Monitor Server、Backup Server デバイスは、共有ストレージ上に設定されている。
- Sybase Adaptive Server、Monitor Server、Backup Server の設定ファイルは、共有ファイルシステム上に格納されている。

- Sybase Adaptive Server と Monitor Server の共有メモリディレクトリは、共有ファイルシステム上に置かれている。
- interfaces ファイルは、ターゲットサーバで更新する必要はない。
- すべての interfaces ファイルエントリは、リソースが保護されるすべてのサーバ上で解決可能でなければならない。
- LifeKeeper Backup Server において、/etc/ld.so.conf にSybase 製品ライブラリのエントリを追加して更新する必要がある。
    - º $SYBASE/ASE/lib のエントリを追加する。
    - º $SYBASE/OCS/lib のエントリを追加する。
    - º Sybase

ASE インストール済み製品を含む共有ファイルシステムをマウントして、ldconfig を実行する。

# 6.19.3.7. Sybase Monitor Server と Backup Server

注記 ：Sybase Monitor Server は Sybase ASE ARK v9.0.2 およびそれ以降のバージョンではサポートされません。

LifeKeeper for Linux Sybase ASE Recovery Kit は、Adaptive Server、Backup Server、Monitor Server コンポーネントにリソース保護を提供します。ただし、Backup Server コンポーネントと Monitor Server コンポーネントは、リソース階層の必須コンポーネントではありません。Sybase Backup Server と Sybase Monitor Server は、リソース保護から除外できます。Sybase Monitor Server または Sybase Backup Server の保護が必要ない場合、リソース階層の作成中に該当するコンポーネントの選択画面で「none」を選択できます。GUI によるリソース作成中に「none」を選択すると、選択されたコンポーネントがリソース階層の保護から除外されます。 **注記:**「none」は、Sybase ASE Recovery Kit の予約語です。そのため、Sybase Backup Server と Sybase Monitor Server のいずれも、「none」という名前を付けることはできません。

これらのコンポーネントを保護するかどうかを選択する際は、Adaptive Server の設定ファイル、デバイスパス、ログパス、または共有メモリディレクトリとファイルシステムを共有する設定ファイルが、LifeKeeper によって保護されることに注意することが重要です。1 つまたは複数のコンポーネントを LifeKeeper の保護から除外する場合は、保護するコンポーネントと保護しないコンポーネントが共有されないようにファイル配置を考慮する必要があります。

# 6.19.3.8. Network Attached Storage の使用

クラスタストレージとして NFS ファイルサーバ (Network Attached Storage) を使用する場合、LifeKeeper の設定においていくつかの特別な考慮事項があります。

## NAS Recovery Kit の使用

LifeKeeper for Linux で NFS サーバを共有ストレージアレイとして使用する場合、オプションの Network Attached Storage (NAS) Recovery Kit が必要になります。各クラスタノードに NAS Recovery Kit (およびライセンス) をインストールしてください。詳細については、NAS Recovery Kit のドキュメンテーションを参照してください。

### エラーメッセージ

Sybase ASE で Network Attached Storage (NAS) を使用する際、システムクラッシュによるフェイルオーバ後、Sybase が再開できないといった事象に遭遇する可能性があります。Sybase エラーログを確認すると、エラーの原因が分かります。

#### Sybase ASE 15.x

```
00:00:00000:00000:2011/05/09 16:08:51.66 kernel Adaptive Server
Enterprise(Developer Edition)
00:00:00000:00000:2011/05/09 16:08:51.66 kernel basis_dlock: file
'/s10/sybase-data155/data/master.dat' already in use by an ASE
00:00:00000:00000:2011/05/09 16:08:51.66 kernel kdconfig: unable to
read primary master device
00:00:00000:00000:2011/05/09 16:08:51.66 server kiconfig: read of
config block failed
```

これは、LifeKeeper によって制御されている NFS ファイルシステム上のファイル「*master.dat*」に、Sybase データサーバによって NFS ロックが設定されたことを示しています。システムクラッシュによりロックがクリアされなかったので、LifeKeeper はデータサーバを In Service の状態に戻すことができません。Sybase は何か別のプロセスが *master.dat* ファイルを使用しているとみなします。

### 解決方法

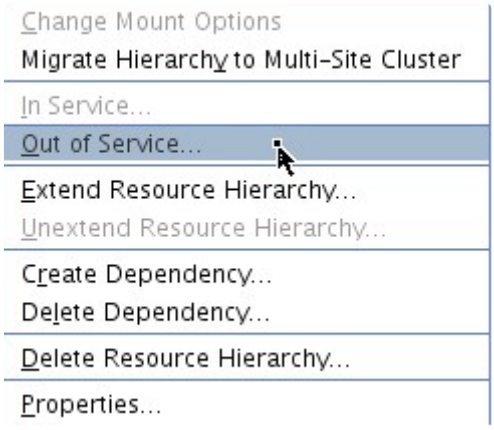
この事象を解決するためには、ファイルシステムリソースを作成する前に、NFS のオプション「nolock」を付けて *master.dat* を格納する予定の NFS ファイルシステムをマウントしてください。デフォルトでは、NFS はファイルのロックを有効にします。「nolock」オプションがリソース作成前に使用されると、LifeKeeper はこのオプションを認識し、ファイルシステムリソースを In Service の状態にするたびにこのオプションを使用します。LifeKeeper は (クラスタノードから) *master.dat* を含んだファイルシステムへのアクセスを制御するので、通常、ロックは重要ではありません。テスト中に使用される NFS マウントオプションは「`rw,sync,tcp,nfsvers=3,noac,nolock`」です。

Sybase ASE のバイナリが格納されているファイルシステムなど、Sybase リソース階層によって使用さ

れるその他のファイルシステム上では、「nolock」を使用する必要はありません。

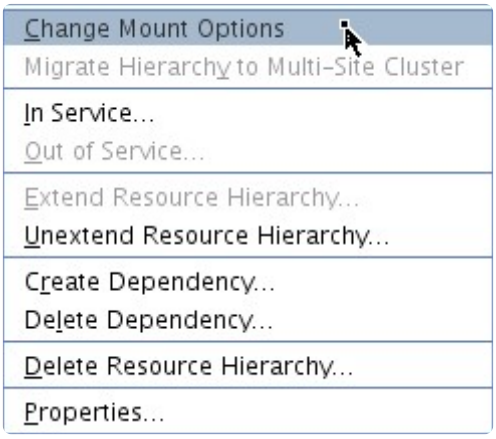
NAS ファイルシステムリソースがすでに「nolock」オプション設定なしに作成されている場合は、以下の手順に従ってマウントオプションを変更してください。

1. LifeKeeper GUI を使用して変更する必要があるファイルシステムリソースに対して Out of Service 操作を行ってください。この操作は LifeKeeper GUI でファイルシステムリソース上にポインタを置き、右クリックしてドロップダウンメニューから [Out of Service] を選択することで可能です。この操作は親リソースも同じく Out of Service 状態にします。

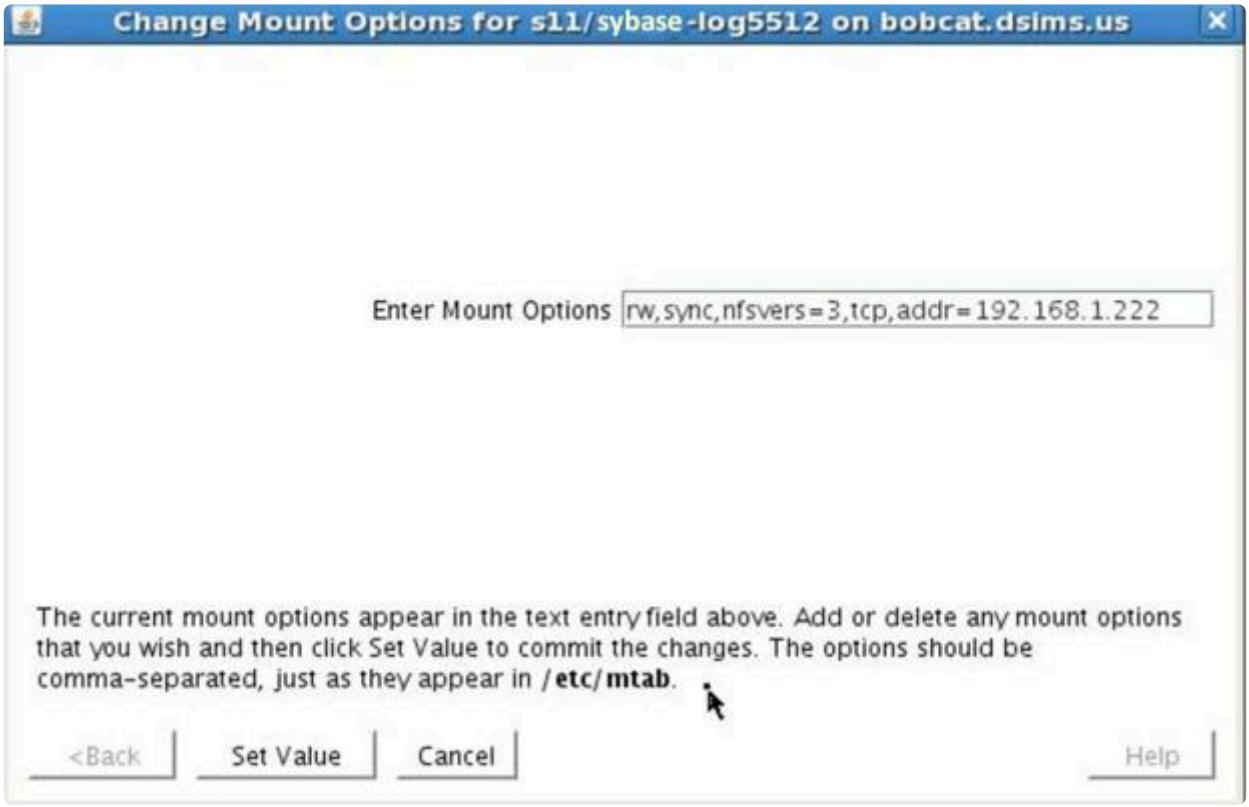



2. [Out of Service] 操作を実行し、プロセスを完了させてください。

3. 一旦ファイルシステムリソースが Out of Service 状態になったら、ポインタをリソース上に置き右クリックし、ドロップダウンメニューから [Change Mount Options] を選択してください。

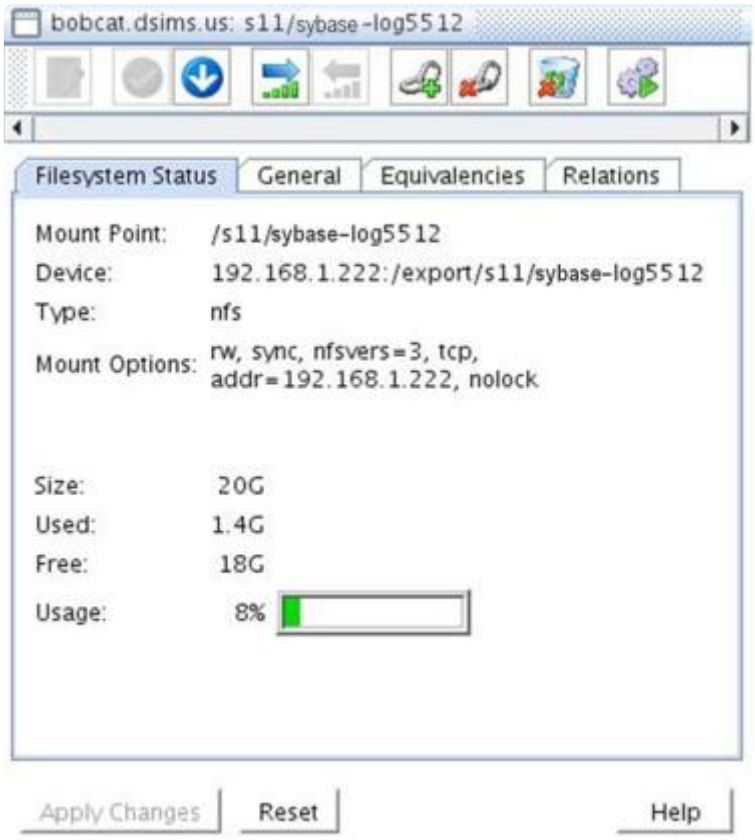



4. ポップアップウィンドウで、オプションの行に「nolock」を追加し、[Set Value] をクリックしてください。クラスタの各ノードで手順 3 および 4 を繰り返す必要があります。




5. マウスを右クリックし、[In Service] を選択し、NAS ファイルシステムリソースを In Service の状態に戻してください。

6. ファイルシステムリソースのプロパティパネルが、「nolock」が現在のマウントオプションであることを表示します。



# 6.19.4. インストール

## Sybase ASE と LifeKeeper のインストール / 設定

本セクションでは、Sybase ASE ソフトウェアと LifeKeeper ソフトウェアのインストールおよび設定方法を説明します。

Sybase ASE 製品と LifeKeeper ソフトウェアのインストールと設定には以下の手順をお勧めします。各ステップは、詳細作業にリンクしています。

Sybase ASE ソフトウェアのインストール

Sybase ASE サーバの作成

LifeKeeper ソフトウェアのインストール

これらの作業を実行すると、LifeKeeper リソース階層を作成して Sybase ASE サーバを保護できるようになります。

## リソース設定作業

階層の作成、拡張、削除、拡張解除など、LifeKeeper GUI で実行可能なさまざまな機能を説明します。

前のセクションで説明したセットアップ作業を終了したら、Sybase ASE リソース階層を作成して拡張できるようになります。

LifeKeeper for Linux Sybase ASE Recovery Kit の設定に関して、以下の作業を行うことができます。

- Create Resource Hierarchy - Sybase ASE リソース階層を作成します。
- Delete Resource Hierarchy - Sybase ASE リソース階層を削除します。
- Extend Resource Hierarchy - Sybase ASE リソース階層をプライマリサーバからバックアップサーバに拡張します。
- Unextend Resource Hierarchy - LifeKeeper クラスタ内の 1 つのサーバから Sybase ASE リソース階層を拡張解除 (削除) します。
- Testing Your Resource Hierarchy - Sybase ASE リソース階層をテストします。

LifeKeeper Core のリソース階層 (ファイルシステムや IP リソースなど) の設定方法については、SPS for Linux テクニカルドキュメンテーション の GUI Administrative Tasks セクションを参照してください。

以下の作業については、すべての Recovery Kit で同じ手順を使用する共通の作業であるため、 SPS for Linux テクニカルドキュメンテーション の 管理 セクションを参照してください。

- Create a Resource Dependency 。既存のリソース階層と別のリソースインスタンスとの間に親子の依存関係を作成し、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- Delete a Resource Dependency 。リソースの依存関係を削除して、クラスタ内のすべての対象サーバに依存関係の変更を反映します。
- In Service 。リソース階層を特定のサーバで In Service の状態にします。
- Out of Service 。リソース階層を特定のサーバで Out of Service の状態にします。
- View / Edit 特定のサーバでリソース階層のプロパティを表示または編集します。

**注記:** このセクションの残りの部分では、設定作業はすべて **[Edit]** メニューから実行されます。この設定作業の大半は次の方法でも実行できます。

- ツールバーから
- 状況表示の左側ペインにあるグローバルリソースを右クリックする
- 状況表示の右側ペインにあるリソースを右クリックする

右クリックの方法を使用すると、 **[Edit]** メニューを使用するときに要求される情報を入力せずにすみます。

# 6.19.4.1. Sybase ASE ソフトウェアのインストール

Sybase ASE ソフトウェアは、クラスタ内の全サーバに同一のパラメータ / 設定を使用してインストールする必要があります。詳細については、Installation Guide Adaptive Server for Linux を参照してください。以下は、LifeKeeper を Sybase ASE と連携させるための追加の推奨事項です。

- 非 root のシステムユーザ (Sybase OS ユーザ) がすべてのサーバ上に存在する必要があります。そのユーザは、リソースが保護されるすべてのサーバ上で同じユーザ ID、グループ ID、ホームディレクトリを持たなければなりません。

- Sybase ASE 共通ソフトウェアパッケージがインストールされている必要があります。このパッケージは、Sybase `srvbuild` と Sybase `isql` ユーティリティの両方を提供します。

- Sybase ASE リソース階層を含む各 LifeKeeper サーバは、 *$SYBASE/interfaces* ファイル内に Sybase ASE サーバについて同一のサービスエントリを持つ必要があります。

- *$SYBASE/ASE-<version>* と *$SYBASE/ASE* 間にリンクが存在することを確認してください。リンクが存在しない場合、手動で作成する必要があります。補足情報については、ASE と OCS のリンクの作成 を参照してください。

- $SYBASE/OCS-<version> と $SYBASE/OCS 間にリンクが存在することを確認してください。リンクが存在しない場合、手動で作成する必要があります。補足情報については、ASE と OCS のリンクの作成 を参照してください。

- Adaptive Server、Monitor Server、Backup Server に共有メモリパラメータを設定する方法の詳細については、 *Installation Guide Adaptive Server for Linux* を参照してください。

- データベースデバイスは LifeKeeper にて共有ストレージとして保護されている必要があります。また設定ファイルなども LifeKeeper により保護されている共有ストレージ上に置く必要があります。詳細は、Sybase ASE サーバの作成 も参照してください。

- Sybase ASE 共通ソフトウェアパッケージは、LifeKeeper により保護されている共有ストレージ上にインストールするか、クラスタ内の全サーバにてローカル領域の同一パスにインストールしてください。

# 6.19.4.2. Sybase ASE サーバの作成

注記 ：Sybase Monitor Server は Sybase ASE ARK v9.0.2 およびそれ以降のバージョンではサポートされません。

Sybase の Adaptive Server、Monitor Server、Backup Server を設定するには、お手元の Installation Guide Adaptive Server for Linux を参照してください。次の考慮事項に従う必要があります。

- srvbuild ユーティリティもしくはその他の Sybase ASE ユーティリティを使用して Sybase Adaptive Server インスタンスを作成する
  - º すべてのシステムデバイスを共有ストレージ上で設定する
  - º Adaptive Server 設定ファイルを共有ストレージ上で設定する
  - º Adaptive Server 共有メモリディレクトリを共有ストレージ上で設定する
  - º LifeKeeper の切り替え可能 IP アドレスを使用するためのインターフェースを設定する
  - º オプションで、共有ストレージ上にログを設定する
- 必要に応じて、Sybase Monitor Server インスタンスを作成する
  - º すべてのシステムデバイスを共有ストレージ上で設定する
  - º Monitor Server 設定ファイルを共有ストレージ上で設定する
  - º Monitor Server 共有メモリディレクトリを共有ストレージ上で設定する
  - º LifeKeeper の切り替え可能 IP アドレスを使用するためのインターフェースを設定する
  - º オプションで、共有ストレージ上にログを設定する
- 必要に応じて、Sybase Backup Server インスタンスを作成する
  - º すべてのシステムデバイスを共有ストレージ上で設定する

- º Monitor Server 設定ファイルを共有ストレージ上で設定する
- º Monitor Server 共有メモリディレクトリを共有ストレージ上で設定する
- º LifeKeeper の切り替え可能 IP アドレスを使用するためのインターフェースを設定する
- º オプションで、共有ストレージ上にログを設定する

# 6.19.4.3. LifeKeeper ソフトウェアのインストール

Sybase ASE ソフトウェアをインストールし、データベースサーバを作成したら、LifeKeeper Core ソフトウェアと LifeKeeper for Linux IP Recovery Kit および必要なパッチをインストールし、続いて Sybase ASE Recovery Kit をインストールできます。raw デバイスで Sybase ASE を使用する予定がある場合は、LifeKeeper Raw I/O Recovery Kit を LifeKeeper Core イメージファイルからインストールする必要があります。raw デバイスの設定の要件および方法については、「raw I/O によるデータベースデバイスの作成」を参照してください。

LifeKeeper パッケージのインストール方法については、SPS for Linux インストールガイド を参照してください。

# 6.19.4.4. Sybase ASE リソース階層の作成

注記: プライマリサーバ上でSybase ASE が起動していることを確認してください。

プライマリサーバで次の手順を実行します。

1. [Edit] メニューの [Server] から [Create Resource Hierarchy] を選択します。

   [Create Resource Wizard] ダイアログボックスが表示されます。

2. ドロップダウンリストから [Sybase ASE Database] を選択し、[Next] をクリックしてください。

3. 以下の情報を入力するように要求されます。ダイアログボックスで [Back] ボタンが有効な場合は、前のダイアログボックスに戻ることができます。これは、エラーが発生して、前に入力した情報を修正する必要がある場合に便利な機能です。いつでも [Cancel] をクリックして、作成処理全体を取り消すことができます。

| フィールド | ヒント |
|---|---|
| **Server** | Sybase ASE リソースを作成する LifeKeeper サーバを選択します。 |
| **Switchback Type** | intelligent または automatic を選択します。フェイルオーバ後、バックアップサーバで Sybase ASE リソースが In Service (アクティブ) になるとき、プライマリサーバにスイッチバックされる方法が決まります。インテリジェントスイッチバック (intelligent) の場合、リソースをプライマリサーバにスイッチバックするときに管理者の介入が必要ですが、自動スイッチバック (automatic) の場合は、プライマリサーバがオンラインに戻り、LifeKeeper のコミュニケーションパスが再確立されるとすぐにスイッチバックが実行されます。<br><br>注記: スイッチバック方法は、Sybase ASE リソースが使用する依存リソースのスイッチバック方法と一致する必要があります。 |
| **Sybase Install Directory** | Sybase ASE 製品のインストール場所を指定します。他のディレクトリパスを入力することもできます。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| **Sybase Instance** | このフィールドはSybaseデータディレクトリを含むディレクトリパスの特定に使われます。データディレクトリは通常、ASE-<version>/RUN_* ファイルをインスタンスとして含みま |

| | |
|---|---|
| **Directory** | す。 |
| **Sybase Instance** | システムで最初に検出された、LifeKeeper 階層が存在しない Sybase インスタンスがデフォルトで入ります。ドロップダウンリストには、LifeKeeper サーバ上で利用可能なその他の Sybase インスタンスが表示されます。このフィールドは LifeKeeper の保護下に置かれる Sybase ASE Database インスタンスの指定に使われます。指定対象となるインスタンスは存在し稼働していなくてはなりません。 |
| **Sybase Username** | このフィールドは Sybase System Administrator のユーザー名の入力に使われます。デーフォールトのユーザー名は sa です。この System Administrator はログインしており、保護される Sybase Adaptive Server 上のすべてのデータベースに対する特権を付与されている必要があります。 |
| **Sybase Login Password** | このフィールドは Sybase System Administrator 用のパスワードの指定に使われます。 |
| **Sybase Backup Server** | 指定された Adaptive Server インスタンスに対する Sybase Backup Server を指定します。この Sybase Backup Server は、LifeKeeper の保護下に置かれます。Sybase Backup Server を LifeKeeper の保護下に置く必要がない場合、「none」を選択できます。 |
| **Sybase ASE Database Tag** | プライマリサーバ上の新しい Sybase ASE データベースリソースに対する固有のタグ名です。デフォルトのタグ名は語「sybase」の後にAdaptive Server インスタンスの名前を付けたものです。別の固有のタグ名を入力することもできます。タグ名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |

4. **[Next]** をクリックしてください。 **[Create Resource Wizard]** が表示され、Sybase ASE リソース階層が作成されます。LifeKeeper によって入力データが検査されます。問題が検出された場合、情報ボックスにエラーメッセージが表示されます。

5. Sybase ASE リソース階層が正常に作成されたこと、フェイルオーバによる保護を実現するにはその階層をクラスタ内の別のサーバに拡張する必要があることを示すメッセージが表示されます。 **[Next]** をクリックします。

6. **[Continue]** をクリックします。 **[Pre-extend Wizard]** が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、「Sybase ASE リソース階層の拡張」の **手順 2** を参照してください。

# 6.19.4.5. Sybase ASE リソース階層の拡張

この操作は [Edit] メニューから開始できます。または、[Create Resource Hierarchy] オプションが完了すると自動的に開始されます。その場合は下の手順 2 を参照してください。

1. [Edit] メニューの [Resource] から [Extend Resource Hierarchy] を選択します。[Pre-Extend Wizard] が表示されます。拡張操作に慣れていない場合は、[Next] をクリックします。LifeKeeper のリソース階層の拡張作業のデフォルト値が分かっていて、入力と確認を省略する場合は、[Accept Defaults] をクリックします。

2. [Pre-Extend Wizard] で以下の情報を入力します。

   **注記:** 最初の 2 つのフィールドは [Edit] メニューの [Extend] から操作を開始した場合にだけ表示されます。

| フィールド | ヒント |
|---|---|
| **Template Server** | Sybase ASE リソースが現在 In Service のサーバを選択します。 |
| **Tag to Extend** | 拡張する Sybase ASE リソースを選択します。 |
| **Target Server** | 拡張先のサーバを入力または選択します。 |
| **Switchback Type** | フェイルオーバ後、バックアップサーバで Sybase ASE リソースが In Service (アクティブ) になるとき、プライマリサーバにスイッチバックされる方法が決まります。intelligent または automatic を選択できます。スイッチバックタイプは、必要な場合 [Resource Properties] ダイアログボックスの [General] タブで後から変更できます。<br><br>**注記**: スイッチバック方法は、Sybase ASE リソースが使用する依存リソースのスイッチバック方法と一致する必要があることに注意してください。 |
| **Template Priority** | **テンプレートの優先順位** を選択または入力します。これはサーバで現在 In Service の Sybase ASE 階層の優先順位です。優先順位は、1 ~ 999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当)。拡張処理時に、別のシステムですでに使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。<br><br>**注記**: このフィールドは階層を最初に拡張するときだけ表示されます。 |
| **Target Priority** | これは、新しく拡張する Sybase ASE 階層が、他のサーバに存在する同等の階層に対して持つ相対的な優先順位です。1 ~ 999 の範囲で、まだ優先順位として使用されていない値が有効で、リソースのカスケーディングフェイルオーバシーケンスにおけるサーバの優先順位を示します。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられる |

| | |
|---|---|
| | ことに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。 |

3. 拡張前のチェックが正常に終了したというメッセージが表示されたら、Next をクリックします。

4. 拡張する階層に応じて、拡張されるリソースタグ (一部編集不可) を示す一連の情報ボックスが表示されます。

5. [Extend Wizard] で以下の情報を入力します。

| | |
|---|---|
| **Sybase ASE Install Directory** | デフォルトでは、テンプレートリソースの Sybase ASE インストールパスが入ります。有効な Sybase ASE インストールパスを指定する必要があります。パス名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |
| **Sybase ASE Database Tag** | プライマリサーバ上の新しい Sybase ASE データベースリソースに対する固有のタグ名です。デフォルトのタグ名は語「sybase」の後にAdaptive Server インスタンスの名前を付けたものです。別の固有のタグ名を入力することもできます。タグ名に対する有効な文字は、アルファベット、数字および以下の特別な記号になります。- _ ./ |

6. 「Hierarchy extend operations completed (階層の拡張操作が完了しました)」というメッセージが表示されたら、階層を別のサーバに拡張する場合は、[Next Server] をクリックし、拡張作業が必要ない場合は、[Finish] をクリックします。

7. 「Hierarchy Verification Finished (階層の検査が完了しました)」というメッセージが表示されたら、[Done]をクリックします。

# 6.19.4.6. Sybase ASE リソース階層の拡張解除

LifeKeeper クラスタ内にある 1 つのサーバからリソース階層を削除するには、次の手順を実行します。

1. [Edit] メニューの [Resource] から [Unextend Resource Hierarchy] を選択します。
2. Sybase ASE リソースを拡張解除するターゲットサーバを選択します。リソースが現在 In Service のサーバは選択できません。(このダイアログボックスは、右側のペインでリソースインスタンスを右クリックして [Unextend] 作業を選択した場合には、表示されません。)[Next] をクリックします。
3. 拡張解除する Sybase ASE 階層を選択して、[Next] をクリックします。(このダイアログボックスは、いずれかのペインでリソースインスタンスを右クリックして [Unextend] 作業を選択した場合には、表示されません。)
4. 拡張解除のために選択したターゲットサーバと Sybase ASE リソース階層を確認する情報ボックスが表示されます。[Unextend] をクリックします。
5. Sybase ASE リソースが正常に拡張解除されたことを示す別の情報ボックスが表示されます。[Done] をクリックして、[Unextend Resource Hierarchy] メニューを終了します。

# 6.19.4.7. Sybase ASE リソース階層の削除

LifeKeeper 設定内のすべてのサーバから Sybase ASE リソースを削除するには、次の手順を実行します。

1. [Edit] メニューの [Resource] から [Delete Resource Hierarchy] を選択します。
2. Sybase ASE リソース階層を削除するターゲットサーバの名前を選択します。

   **注記:** 左側のペインからグローバルリソースを右クリックするか、右側のペインから個々のリソースインスタンスを右クリックして [Delete Resource] 作業を選択した場合、このダイアログボックスは表示されません。
3. [Hierarchy to Delete] を選択します。(このダイアログボックスは、左側または右側のペインでリソースインスタンスを右クリックして、[Delete Resource] 作業を選択した場合には、表示されません。)[Next] をクリックします。
4. 選択したターゲットサーバと、削除の対象として選択した階層を確認する情報ボックスが表示されます。[Next] をクリックします。
5. Sybase ASE リソースが正常に削除されたことを示す別の情報ボックスが表示されます。
6. [Done] をクリックして終了します。

# 6.19.4.8. リソース階層のテスト

Sybase ASE リソース階層のテストは、手動でスイッチオーバすることで、プライマリサーバからバックアップサーバへのリソースサーバインスタンスのフェイルオーバをシミュレートして実行できます。

## LifeKeeper GUI からの手動スイッチオーバの実行

LifeKeeper GUI で **[Edit] > [Resource] > [In Service]** を選択すると、スイッチオーバを開始できます。例えば、バックアップサーバで [In Service] 要求 が実行されると、Sybase ASE リソース階層がバックアップサーバ側で In Service になり、プライマリサーバ側では Out of Service になります。この時点で、元のバックアップサーバがプライマリサーバに、元のプライマリサーバがバックアップサーバに変わります。

**[Out of Service]** 要求を実行した場合、リソース階層は Out of Service になりますが、別のサーバで In Service にはなりません。

**重要:** バックアップサーバでリソース階層を In Service した後は、データベースに接続してみる必要があります。特に、raw デバイスをデバイス空間として使用している場合は接続を試みる必要があります。この作業は、バックアップサーバですべてのディスクパーティションが認識されており、raw バインディングが正しく確立されていることを確認するために必要です。

バックアップサーバで raw バインディングが確立されていない場合、その原因として最も可能性が高いのは、新しいパーティションをプライマリサーバに作成して設定に追加したにも関わらず、バックアップサーバ側のパーティションテーブルが更新されていないためです。

解決策は、パーティションテーブルが正しくアップデートされるように、バックアップサーバを再起動することです。

# 6.19.5. 管理

## リソース階層の管理

以下は、Sybase ASE 階層の継続的な管理に対する重要な推奨事項です。

リソース階層を作成後、以下の作業が必要になることがあります。

Sybase Backup Server の保護の変更

Sybase Monitor Server の保護の変更

パラメータの更新

# 6.19.5.1. Sybase Backup Server の保護の変更

Sybase Backup Server は、Open Server ベースのアプリケーションで、Adaptive Server でのデータベースのバックアップ (ダンプ) とリストア (ロード) をすべて管理します。Sybase Backup Server は、LifeKeeper Sybase ASE リソース階層の作成中に保護下に置くことも、リソース階層を作成した後に LifeKeeper の保護下に加えることもできます。また、リソース階層を作成した後に、Sybase Backup Server を LifeKeeper の保護下から除外することもできます。

## Sybase Backup Server の追加

Sybase Backup Server を既存の Sybase ASE リソース階層に追加するには、Sybase の srvbuild またはその他の設定ユーティリティによって Backup Server を作成済みである必要があります。

1. [Edit] メニューから [Resource] を選択し、次に [Properties] を選択してください。[Resource Properties] ウィザードが開きます。

2. [Select Resource] プルダウンメニューからリソースタグを選択してください。これは、LifeKeeper 保護下にある変更対象の Sybase ASE リソースのリソースタグです。

3. [Select Server for Resource] プルダウンメニューで [LifeKeeper Server] を選択してください。これは、Sybase ASE リソースインスタンスを更新するサーバです。複数の LifeKeeper サーバ上で変更が必要な場合、この手順をクラスタ内の各サーバについて繰り返す必要があります。

4. [Resource Properties] ページで [Resource Configuration] ボタンをクリックしてください。これで、手順 3 で選択した保護リソースに対する [Reconfiguration] ウィザードが開きます。ウィザードの最初の画面には、この LifeKeeper 保護リソースの現在の設定が表示されます。[Next] をクリックします。

5. 有効な Sybase Backup Server が指定されたサーバ上に存在する場合、追加または削除する Sybase Backup Server が次の画面のプルダウンリストに表示されます。追加する Sybase Backup Server をリストから選択します。[Next] をクリックします。 **注記:** Sybase ASE インストールにおいて、共有ストレージに Sybase ソフトウェアがインストールされている場合、そのインストールを含むファイルシステムが、再設定を実行するサーバ上で In-Service される必要があります。

6. 有効な Sybase Monitor Server が存在する場合、次の画面で設定できます。Monitor Server の保護を変更する際の考慮事項については、Sybase Monitor Server の保護の変更 を参照してください。

7. [Reconfigure] を選択してください。何らかのエラーが表示された場合、それを修正してから先に進んでください。エラーがない場合、[Done] をクリックしてください。

8. Sybase Backup Server 設定ファイルパスまたは関連するデータベースデバイスは、LifeKeeper ファイルシステムリソースで手動で保護し、親リソース階層の依存リソースにする必要があります。

9. Sybase Backup Server に関連する仮想 IP アドレスは、親リソース階層の依存リソースにする必要があります。関連する IP アドレスを特定するには、interfaces ファイル内で、Sybase Backup Server 名の後のマスターおよびクエリ行を参照してください。

# Sybase Backup Server の削除

以下に、Sybase Backup Server を既存の Sybase ASE リソース階層から削除する手順を説明します。

1. [Edit] メニューから [Resource] を選択し、次に [Properties] を選択してください。[Resource Properties] ウィザードが開きます。

2. [Select Resource] プルダウンメニューからリソースタグを選択してください。これは、LifeKeeper 保護下にある変更対象の Sybase ASE リソースのリソースタグです。

3. [Select Server for Resource] プルダウンメニューで [LifeKeeper Server] を選択してください。これは、Sybase ASE リソースインスタンスを更新するサーバです。複数の LifeKeeper サーバ上で変更が必要な場合、この手順をクラスタ内の各サーバについて繰り返す必要があります。

4. [Resource Properties] ページで [Resource Configuration] ボタンをクリックしてください。これで、手順 3 で選択された保護リソースに対する [Reconfiguration] ウィザードが開きます。ウィザードの最初の画面には、この LifeKeeper 保護リソースの現在の設定が表示されます。[Next] をクリックします。

5. 有効な Sybase Backup Server が指定されたサーバ上に存在する場合、追加または削除する Sybase Backup Server が次の画面のプルダウンリストに表示されます。Sybase Backup Server の保護を解除するには、リストで [none] を選択します。[Next] をクリックします。

6. 有効な Sybase Monitor Server が存在する場合、次の画面で設定できます。Monitor Server の保護を変更する際の考慮事項については、Sybase Monitor Server の保護の変更 を参照してください。

7. [Reconfigure] を選択してください。何らかのエラーが表示された場合、それを修正してから先に進んでください。エラーがない場合、[Done] をクリックしてください。

8. 不要になった Sybase Backup Server 設定ファイルパスまたは関連するデータベースデバイスは、Sybase ASE リソース依存関係を解除し、LifeKeeper から削除する必要があります。

9. 不要になった Sybase Backup Server 仮想 IP リソースは、Sybase ASE リソース依存関係を解除し、LifeKeeper から削除する必要があります。

# 6.19.5.2. Sybase Monitor Server の保護の変更

注記 ：Sybase Monitor Server は Sybase ASE ARK v9.0.2 およびそれ以降のバージョンではサポートされません。

Monitor Server は、Adaptive Server データベースサーバとは独立したサーバであり、Adaptive Server を監視します。Monitor Server は、リアルタイムデータまたはヒストリカルデータをクライアントアプリケーションに提供できます。Sybase Monitor Server は、LifeKeeper Sybase ASE リソース階層の作成中に保護下に置くことも、リソース階層を作成した後に LifeKeeper の保護下に加えることもできます。また、リソース階層を作成した後に、Sybase Monitor Server を LifeKeeper の保護下から除外することもできます。

## Sybase Monitor Server の追加

Sybase Monitor Server を既存の Sybase ASE リソース階層に追加するには、Sybase の srvbuild またはその他の設定ユーティリティによって Monitor Server を作成済みである必要があります。

1. [Edit] メニューから [Resource] を選択し、次に [Properties] を選択してください。[Resource Properties] ウィザードが開きます。

2. [Select Resource] プルダウンメニューからリソースタグを選択してください。これは、LifeKeeper 保護下にある変更対象の Sybase ASE リソースのリソースタグです。

3. [Select Server for Resource] プルダウンメニューで [LifeKeeper Server] を選択してください。これは、Sybase ASE リソースインスタンスを更新するサーバです。複数の LifeKeeper サーバ上で変更が必要な場合、この手順をクラスタ内の各サーバについて繰り返す必要があります。

4. [Resource Properties] ページで [Resource Configuration] ボタンをクリックしてください。これで、手順 3 で選択された保護リソースに対する [Reconfiguration] ウィザードが開きます。ウィザードの最初の画面には、この LifeKeeper 保護リソースの現在の設定が表示されます。[Next] をクリックします。

5. 有効な Sybase Backup Server が存在する場合、次の画面で設定できます。Backup Server の保護を変更する際の考慮事項については、Sybase Backup Server の保護の変更 を参照してください。

6. 有効な Sybase Monitor Server が指定されたサーバ上に存在する場合、追加または削除する Sybase Monitor Server が次の画面のプルダウンリストに表示されます。追加する Sybase Monitor Server をリストから選択します。[Next] をクリックします。 **注記:** Sybase ASE インストールにおいて、共有ストレージに Sybase ソフトウェアがインストールされている場合、再設定を実行するサーバ上で、そのインストールを含むファイルシステムが In-Service される必要があります。

7. [Reconfigure] を選択してください。何らかのエラーが表示された場合、それを修正してから先に進んでください。エラーがない場合、[Done] をクリックしてください。

8. Sybase Monitor Server 設定ファイルパスまたは関連するデータベースデバイスは、LifeKeeper フ

ファイルシステムリソースで手動で保護し、親 Sybase ASE リソース階層の依存リソースにする必要があります。

9. Sybase Monitor Server に関連する仮想 IP アドレスは、親 Sybase ASE リソース階層の依存リソースにする必要があります。関連する IP アドレスを特定するには、 interfaces ファイル内で、Sybase Monitor Server 名の後のマスターおよびクエリ行を参照してください。

# Sybase Monitor Server の削除

以下に、Sybase Monitor Server を既存の Sybase ASE リソース階層から削除する手順を説明します。

1. [Edit] メニューから [Resource] を選択し、次に [Properties] を選択してください。[Resource Properties] ウィザードが開きます。

2. [Select Resource] プルダウンメニューからリソースタグを選択してください。これは、LifeKeeper 保護下にある変更対象の Sybase ASE リソースのリソースタグです。

3. [Select Server for Resource] プルダウンメニューで [LifeKeeper Server] を選択してください。これは、Sybase ASE リソースインスタンスを更新するサーバです。複数の LifeKeeper サーバ上で変更が必要な場合、この手順をクラスタ内の各サーバについて繰り返す必要があります。

4. [Resource Properties] ページで [Resource Configuration] ボタンをクリックしてください。これで、手順 3 で選択された保護リソースに対する [Reconfiguration] ウィザードが開きます。ウィザードの最初の画面には、この LifeKeeper 保護リソースの現在の設定が表示されます。[Next] をクリックします。

5. 有効な Sybase Backup Server が存在する場合、次の画面で設定できます。Backup Server の保護を変更する際の考慮事項については、Sybase Backup Server の保護の変更 を参照してください。

6. 有効な Sybase Monitor Server が指定されたサーバ上に存在する場合、追加または削除する Sybase Monitor Server が次の画面のプルダウンリストに表示されます。Sybase Monitor Server の保護を解除するには、リストで [none] を選択します。[Next] をクリックします。

7. [Reconfigure] を選択してください。何らかのエラーが表示された場合、それを修正してから先に進んでください。エラーがない場合、[Done] をクリックしてください。

8. 不要になった Sybase Monitor Server 設定ファイルパスまたは関連するデータベースデバイスは、Sybase ASE リソース依存関係を解除し、LifeKeeper から削除する必要があります。

9. 不要になった Sybase Monitor Server 仮想 IP リソースは、Sybase ASE リソース依存関係を解除し、LifeKeeper から削除する必要があります。

# 6.19.5.3. パラメータの更新

Sybase ASE インスタンスのデータベースパラメータを更新するときは、すべての変更を行っても、クラスタ内の全 LifeKeeper サーバ上でそのインスタンスが正しく機能することを確認する必要があります。変更を行うことにより、LifeKeeper リソース (ファイルシステム、raw デバイス、仮想 IP アドレスなど) の追加や削除が必要になる場合、そのリソースを手動で追加し、親 Sybase ASE リソース階層の依存リソースにする必要があります。

# 6.19.6. トラブルシューティング

## Sybase ASE Recovery Kit のエラーメッセージ

Sybase ASE Recovery Kit に関連するエラーメッセージの一覧と説明です。

| | |
|---|---|
| 114000 | Usage: %s (使用法: %s) |
| 114001 | The Sybase Install Directory cannot be empty. (Sybase インストールディレクトリを空にすることができません)<br><br>処置: このフィールドに値を指定してください。 |
| 114002 | The path ”%s” is not a valid directory (パス「%s」は有効なディレクトリではありません) |
| 114003 | The Sybase Product was not found in the directory %s on server %s. (サーバ %s のディレクトリ %s に Sybase 製品が見つかりませんでした。)処置:指定の場所にサポート対象バージョンの Sybase がインストールされていることを確認してください。 |
| 114004 | The specified instance %s is not a valid Sybase ASE Server on %s. (指定されたインスタンス %s は、%s 上の有効な Sybase ASE サーバではありません。 |
| 114005 | Unable to verify that the Sybase ASE Server %s is running. (Sybase ASE サーバ %s が実行中であることを確認できません。) |
| 114006 | The Sybase Monitor Server %s will be protected. (Sybase Monitor Server %s を保護します。) |
| 114007 | The Sybase Backup Server %s will be protected. (Sybase Backup Server %s を保護します。) |
| 114008 | The Sybase ASE Server %s is already under LifeKeeper protection on %s. (Sybase ASE Server %s は、すでに %s 上で LifeKeeper に保護されています。) |
| 114009 | An unknown error has occurred in utility %s on server %s. (サーバ %s 上のユーティリティ %s で未知のエラーが発生しました。)<br><br>処置:LifeKeeper ログで詳細を確認し、操作をやり直してください。 |
| 114010 | Unable to get the version for the Sybase Server %s installed under %s on %s. (%s 上の %s にインストールされている Sybase サーバ %s のバージョンを取得できません。) |
| 114011 | The device %s for Sybase ASE Server %s is not a valid device. (Sybase ASE Server %s のデバイス %s は、有効なデバイスではありません。) |
| 114012 | Sybase ASE サーバ %s のデバイスを取得中にエラーが発生しました。 |
| 114013 | Unable to create raw resource hierarchy for %s. (%s の raw リソース階層を作成できません。) |
| 114014 | Unable to create file system resource hierarchy for %s. (%s のファイルシステムリソース階層を作成できません。) |

| | |
|---|---|
| 114015 | The path %s is not on a shared file system.(パス %s が共有ファイルシステムにありません。) |
| 114016 | Unable to create resource dependency for parent %s and child %s. (親 %s と子 %s にリソース依存関係を作成できません。) |
| 114017 | Information: LifeKeeper will not protect the path %s because it is not located on a shared file system. (情報: パス %s は共有ファイルシステム上にないため、LifeKeeper によって保護されません。) |
| 114018 | Unable to get the owner for the Sybase ASE Server%s installed under%s on %s. (%s 上の %s にインストールされている Sybase ASE サーバ %s の所有者を取得できません。) |
| 114019 | Unable to open file%s on server%s due to error %s. (エラー「%s」により、サーバ %s 上でファイル %s を開けません。) |
| 114020 | There are no hosts defined for the Sybase ASE Server %s in the file%s. (ファイル %s 内に Sybase ASE サーバ %s のホストが定義されていません。) |
| 114021 | There are no ports defined for the Sybase ASE Server %s in the file %s. (ファイル %s 内に Sybase ASE サーバ %s のポートが定義されていません。) |
| 114022 | The specified host name %s defined for the Sybase ASE Server %s in the file %s cannot be resolved. (ファイル %s 内に Sybase ASE サーバ %s に対して指定されているホスト名 %s を解決できません。) |
| 114023 | Unable to detect the host and ports for the Sybase ASE Server %s. (Sybase ASE サーバ %s のホストとポートを検出できません。) |
| 114024 | サーバ % 上の IP アドレス %s に LifeKeeper リソース階層が存在しません。<br><br>処置:指定された IP アドレスに LifeKeeper リソース階層を作成してください。 |
| 114025 | The values specified for the target and the template servers are the same.(ターゲットサーバとテンプレートサーバに指定された値が同一です。)<br><br>処置:ターゲットサーバとテンプレートサーバに正しい値を指定してください。 |
| 114026 | The system user %s does not exist on the server %s. (サーバ %s 上にシステムユーザ %s が存在しません。) |
| 114027 | The group id for user %s is not the same on template server %s and target server %s. (ユーザ %s のグループ ID がテンプレートサーバ %s とターゲットサーバ %s で同一ではありません。) |
| 114028 | The user id for user %s is not the same on template server %s and target server %s. (ユーザ %s のユーザ ID がテンプレートサーバ %s とターゲットサーバ %s で同一ではありません。) |
| 114029 | There are no IP dependent resources defined for the Sybase resource %s on %s. (%s 上に Sybase リソース %s に対して定義された IP 依存リソースがありません。) |

| | |
|---|---|
| | 処置:必要な依存 IP リソース階層を作成します。 |
| 114030 | The interfaces defined for Sybase ASE Server %s differ on template server %s and target server %s (Sybase ASE サーバ %s に定義されたインターフェースが、テンプレートサーバ %s とターゲットサーバ %s で異なっています) |
| 114031 | The ports defined for Sybase ASE Server %s differ on template server %s and target server %s (Sybase ASE サーバ %s に定義されたポートが、テンプレートサーバ %s とターゲットサーバ %s で異なっています) |
| 114032 | The port %s used by the Sybase resource hierarchy %s on the server %s is in use by another application on server%s. (サーバ %s 上で Sybase リソース階層 %s によって使用されるポート %s が、サーバ %s 上の別のアプリケーションによって使用されています。) |
| 114033 | The startup of the Sybase ASE Server(s) on %s failed for the following Sybase ASE Server(s): %s. (%s 上の Sybase ASE サーバの起動が、次の Sybase ASE サーバにおいて失敗しました: %s) |
| 114034 | Unable to stop the Sybase ASE Server(s) %s on %s. (%s 上で Sybase ASE サーバ %s を停止できません。) |
| 114035 | The Sybase ASE resource hierarchy %s does not contain any valid gen/filesys or scsi/raw resource dependents on server %s.(Sybase ASE リソース階層 %s に、サーバ %s 上で有効な gen/filesys または scsi/raw リソース依存関係がありません。)<br><br>処置:階層に有効な依存関係がないため、階層を削除して再作成する必要があります。 |
| 114036 | There are no Sybase ASE Servers available for protection with LifeKeeper.(LifeKeeper が保護できる Sybase ASE サーバが存在しません。) |
| 114037 | Unable to obtain the pid of the backupserver process corresponding to instance %s.(インスタンス %s に対応するBackup Server プロセスの pid を取得できません。) |
| 114038 | The pid detected for Sybase Backup Server %s in the LifeKeeper pidfile %s.LK on server %s exists in another LifeKeeper pidfile on this server.(サーバ %s 上の LifeKeeper pidfile %s.LK 内で 検出された Sybase Backup Server %s の pid が、このサーバ上の別の LifeKeeper pidfile に存在します。)<br><br>処置:pid ファイル内の重複する pid エントリを解決する必要があります。実行中でないインスタンスの pid ファイルを削除する必要があります。 |
| 114039 | Unable to update the resource instance %s on server %s. (サーバ %s 上でリソースインスタンス %s を更新できません。) |
| 114040 | The update of the resource instance %s failed on server %s. All attempts to rollback the instance information field have failed. (サーバ %s 上でリソースインスタンス %s の更新に失敗しました。インスタンス情報フィールドをロールバックする試みがすべて失敗しました。)処置:手動による介入が必要です。 |

| | |
|---|---|
| 114041 | The interfaces file %s on %s contains an invalid comment line. (%s 上のインターフェースファイル %s に無効なコメント行があります。)処置:インターフェースファイルを修正し、コメント行を削除してください。 |
| 114042 | One or more of the Sybase ASE Servers is missing from the file %s.(1 台または複数台の Sybase ASE サーバがファイル %s に欠落しています。) |
| 114043 | The file %s does not exist on server %s.(サーバ %s 上にファイル %s が存在しません。) |
| 114044 | The reconfiguration of the Sybase ASE resource hierarchy %s on server %s was successful (サーバ %s 上の Sybase ASE リソース階層 %s の再設定に成功しました。) |
| 114045 | The update of the resource instance %s failed on server %s. The instance information field has not been modified.(サーバ %s 上でリソースインスタンス %s の更新に失敗しました。インスタンス情報フィールドが変更されていません。)<br><br>処置:再設定の操作をもう一度行ってください。 |
| 114046 | The home directory for user %s is not the same on template server %s and target server %s (ユーザ %s のホームディレクトリがテンプレートサーバ %s とターゲットサーバ %s で同一ではありません。) |
| 114047 | The file %s on server %s is a link that does not resolve to a dependent shared resource on the template server %s.(サーバ %s 上のファイル %s はリンクですが、テンプレートサーバ %s 上の依存共有リソースに解決できません。) |
| 114048 | The link %s and its resolved path %s are not on a protected shared filesystem. (リンク %s とそのリンクを解決したパス %s は、保護された共有ファイルシステム上にありません。) |

# 6.19.7. Appendix

## raw I/O によるデータベースデバイスの作成

---

要件

命名規則

raw I/O の設定手順

階層作成後のデータベースデバイスの追加

ASE と OCS のリンクの作成

# 6.19.7.1. 要件

raw I/O で Sybase ASE Recovery Kit を使用する場合、次の要件を満たす必要があります。

- Linux OS が raw I/O デバイスをサポートしている必要があります。ほとんどのディストリビューションでは、カーネル 2.4 でサポートされましたが、カーネル 2.2 からサポートしているディストリビューションもあります。

- すべての raw I/O デバイスを共有ディスクパーティションにバインドする必要があります。raw I/O デバイス上に配置されるデータベースデバイスの数 (devspaces) により、必要な raw デバイスと共有ディスクパーティションの正確な数が決まります。raw デバイス上でデータベースデバイスを作成するための指針については、Installation Guide Adaptive Server for Linux を参照してください。

- Sybase ASE ソフトウェアのバージョンが raw I/Oデバイスの使用をサポートしている必要があります。

# 6.19.7.2. 命名規則

raw デバイスとコントローラの命名方法は、Linux ディストリビューションごとに異なります。

- Red Hat の場合、デバイス名は /dev/raw/raw<number> でコントローラ名は /dev/rawctl です。
- SuSE SLES 7 バージョンの場合、デバイス名は /dev/raw<number> で、コントローラ名は /dev/raw です。
- SuSE SLES 8 バージョンの場合、デバイス名は /dev/raw/raw<number> で、コントローラ名は /dev/rawctl です。

# 6.19.7.3. raw I/O の設定手順

1. Sybase ASE データベースデバイスに適したサイズの共有ディスクパーティションを選択します。

2. このパーティションに、未使用の raw デバイスノードをバインドします。これはマシンを再起動するたびに実行する必要があり、root アクセスを必要とするため、システム初期化ファイル (rc.local または boot.local) に raw バインディングを追加することもできます。このバインディングは、階層を LifeKeeper の保護下においた後、ファイルから削除する必要があります。LifeKeeper は、LifeKeeper の保護下にある raw I/O デバイスの raw バインディングを再確立します。コマンド raw -qa を使用して、どの raw デバイスノードが使用中か確認します。たとえば、以下のようになります。

   ```
   # raw -qa

   # raw /dev/raw/raw1 /dev/sda1
   ```

3. データベースインスタンスを保護するすべてのサーバ上で、raw デバイスコントローラ (/dev/rawctl または /dev/raw) とディスクパーティションの両方に、グローバル読み取り権限を設定してください。

   ```
   # chmod a+r /dev/rawctl (or chmod a+r /dev/raw )
   ```

4. データベースインスタンスを保護するすべてのサーバ上で、raw デバイスにグループおよびユーザの読み取り / 書き込み権限を設定してください。

   ```
   # chmod 664 /dev/raw/raw1
   ```

5. データベースインスタンスを保護するすべてのサーバ上で、raw デバイスの所有者を、指定されたデータベースインスタンスの Sybase ASE 所有者に変更してください。

   ```
   # chown -R sybase:sybase /dev/raw/raw1
   ```

6. raw デバイスのデータベースサーバへの追加については、Installation Guide Adaptive Server for Linux を参照してください。

# 6.19.7.4. 階層作成後のデータベースデバイスの追加

LifeKeeper で Sybase ASE 階層を作成後に、データベースデバイスを raw I/O デバイスまたは共有ファイルシステムに追加する場合、その raw デバイスまたはファイルシステムに対するリソース階層を LifeKeeper GUI を使って手動で作成する必要があります。そして、新しく作成したリソース階層を、Sybase ASE リソース階層の依存リソース階層 (子) にする必要があります。

# 6.19.7.5. ASE と OCS のリンクの作成

LifeKeeper for Linux Sybase ASE Recovery Kit では、パス $SYBASE/ASE-<version> が $SYBASE/ASE にシンボリックリンクされていることが要求されます。また、パス $SYBASE/OCS-<version> は $SYBASE/OCS にシンボリックリンクされている必要があります。LifeKeeper for Linux Sybase ASE Recovery Kit は、これらのリンクを使用してさまざまな Sybase ユーティリティやファイルにアクセスします。リンクを作成するには、次の手順を実行します。

1. コマンドライン上で $SYBASE ディレクトリに移動してください。

   例:

   ```
   server1 # cd $SYBASE

   server1 # pwd

        /opt/sybase-12.5
   ```

2. ASE-<version> ディレクトリを検索してください。

   例:

   ```
   server1 # ls -ld ASE*

   drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 ASE-12_5

   lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:35 ASEP -> ASEP-1_0

   drwxrwxr-x 4 sybase sybase 4096 Nov 17 11:35 ASEP-1_0
   ```

   注記: ASE-12_5 と ASE 間にリンクが既に存在する場合、手順 5 に進んでください。

3. ASE-<version> ディレクトリの中に bin/srvbuild ユーティリティがあることを確認してください。

   例:

   ```
   server1 # ls ASE-12_5/bin/srvbuild

        srvbuild
   ```

   注記:「No such file or directory」エラーが発生する場合、選択したパスが間違っています。

4. コマンドライン上で、特定された ASE-<version> ディレクトリと ASE 間にリンクを作成してください。

   例:

```
server1 # pwd

     /opt/sybase-12.5

server1 # ln -s ASE-12_5 ASE
```

5. リンクが正しく作成されたことを確認してください。

   例:

```
server1 # ls -ld ASE*

     lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:20 ASE -> ASE-12_5

     drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 ASE-12_5

     lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:35 ASEP -> ASEP-1_0

     drwxrwxr-x 4 sybase sybase 4096 Nov 17 11:35 ASEP-1_0

server1 # ls ASE/bin/srvbuild

     srvbuild
```

6. コマンドライン上で $SYBASE ディレクトリに移動してください。

   例:

```
server1 # cd $SYBASE

server1 # pwd

     /opt/sybase-12.5
```

7. OCS-<version> ディレクトリを検索してください。

   例:

```
server1 # ls -ld OCS*

     drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 OCS-12_5
```

   **注記:** OSC-12_5 と OSC 間にリンクが既に存在する場合、手順 5 に進んでください。

8. OCS-<version> ディレクトリの中に bin/isql ユーティリティがあることを確認してください。

   例:

```
server1 # ls OCS-12_5/bin/isql

     isql
```

**注記:**「No such file or directory」エラーが発生する場合、選択したパスが間違っています。

9. コマンドライン上で、特定された OCS-<version> ディレクトリと OCS 間にリンクを作成してください。

   例:

```
server1 # pwd

     /opt/sybase-12.5

server1 # ln -s OCS-12_5 OCS
```

10. リンクが正しく作成されたことを確認してください。

    例:

```
server1 # ls -ld OCS*

     lrwxrwxrwx 1 sybase sybase 8 Nov 17 11:20 OCS -> OCS-12_5

     drwxrwxr-x 16 sybase sybase 4096 Nov 18 09:08 OCS-12_5

server1 # ls OCS/bin/isql

     isql
```

# 6.20. VMDK as Shared Storage Recovery Kit 管理ガイド

## VMDK as Shared Storage Recovery Kit テクニカルドキュメンテーション

LifeKeeper for Linux VMDK as Shared Storage Recovery Kit（以下、VMDK Recovery Kit と呼ぶ）は、LifeKeeper 環境においてVMwareの仮想ハードディスクを共有ストレージとして使用するものです。VMDK Recovery Kit は、LifeKeeper ユーザに対し、 仮想ハードディスクを LifeKeeper 階層の基本ストレージとして使用する機会を提供します。

## ドキュメントの内容

本書は以下のトピックについて説明しています。

- ドキュメンテーションとリファレンス 。LifeKeeper for Linux のドキュメントと所在の一覧です。
- 要件 。VMDK Recovery Kit を適切に設定、導入、運用するうえで必要なハードウェアとソフトウェアについて説明しています。LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SIOS Protection Suite インストールガイド を参照してください。
- 概要 。VMDK Recovery Kit の特徴と機能について説明しています。
- LifeKeeper for Linux VMDK Recovery Kit の設定 。VMDK Recovery Kit を適切に設定するうえで必要な手順について説明しています。
- LifeKeeper 設定作業 。LifeKeeper GUI を使用して VMDK リソース階層を作成し管理する作業について説明しています。
- トラブルシューティング 。LifeKeeper for Linux のエラー･メッセージの一覧とその説明です。

# 6.20.1. ドキュメンテーションとリファレンス

以下に、SIOS Technology Corp が発行している SPS 関連情報の一覧を示します。

- SPS for Linux リリースノート
- SPS for Linux テクニカルドキュメンテーション
- SIOS Protection Suite インストールガイド
- Optional Recovery Kit ドキュメンテーション

このドキュメンテーション、および他の SPS Recovery Kit に関連するドキュメンテーションは、以下の Web サイトで入手できます。

http://jpdocs.us.sios.com

# 6.20.2. ハードウェア要件とソフトウェア要件

SPS for Linux VMDK Recovery Kit をインストールする前に、SPS の構成が次の要件を満たしている必要があります。SPS for Linux ハードウェアおよびソフトウェアの設定に関する具体的な手順については、SIOS Protection Suite インストールガイド を参照してください。

## ハードウェア要件

- **サーバ** 。製品媒体に付属する SPS for Linux リリースノート および SPS for Linux インストールガイド (製品メディアに収録) に記載の要件に従って設定されたLifeKeeper for Linux 対応の VMware ゲスト。

- **IP ネットワーク・インターフェース・カード** 。各サーバは、イーサネット TCP/IP をサポートするネットワーク・インタフェースを 1 つ以上必要とします。ただし、LifeKeeper クラスタには 2 つの通信パスが必要で、独立した 2 つのサブネットを使用する 2 つの分離した LAN ベース通信パスが推奨され、これらの 1 つ以上をプライベート・ネットワークとして構成する必要があることを忘れないでください。TCP と TTY のハートビートの組み合わせもサポートされています。

## ソフトウェア要件

- **TCP/IP ソフトウェア** 。LifeKeeper 環境の各サーバには TCP/IP ソフトウェアが必須です。

- **LifeKeeper ソフトウェア** 。クラスタの各サーバにインストールした LifeKeeper for Linux ソフトウェアのバージョン、および適用した LifeKeeper for Linux ソフトウェア・パッチのバージョンが同一であることが絶対条件です。

- **LifeKeeper for Linux VMDK Recovery Kit** - VMDK Recovery Kit は LifeKeeper インストールイメージに含まれています。セットアップスクリプトのリカバリキット選択画面で選択するとインストールされます。

- **Linux ソフトウェア** 。VMDK Recovery Kit の実行に必要な追加ソフトウエアは LifeKeeper インストールイメージに含まれています。セットアップスクリプトは VMDK Recovery Kit と同時にインストールします。

  LifeKeeper for Linux ソフトウェアの具体的なインストール / アンインストール方法については、SPS for Linux インストールガイド を参照してください。

# 6.20.3. 概要

## LifeKeeper for Linux VMDK Recovery Kit

LifeKeeper for Linux VMDK Recovery Kit の第 1 の目的は、LifeKeeper ユーザに対し、共有ストレージとデータ・レプリケーションに代わる代替ストレージ方式を提供することです。

VMDK Recovery Kit を使用すると、LifeKeeper 保護下のサーバまたはクライアント上に LifeKeeper リソース階層を作成できます。このリソース階層には、VMwareハイパーバイザーの提供する仮想ハードディスクが接続され、そのディスク上に作成されたファイルシステムがマウントされます。

仮想ハードディスクを接続したノードで障害が検出されると、VMDK Recovery Kit は、事前判別されたバックアップ・ノードへのフェイルオーバを開始し、バックアップ・ノードに同じ仮想ハードディスクを再接続します。そのため、仮想ディスク上に構成されたファイル・システムが LifeKeeper サーバまたはクライアントにマウントされたら、それを LifeKeeper 階層のストレージの基礎として完全に使用できます。

VMDK Recovery Kit のリソース階層は、LifeKeeper Core 製品（steeleye-lk パッケージ）から 入手可能な現存の File System Recovery Kit を使用して作成されます。

## VMDK Recovery Kit の制限事項

- このバージョンの VMDK Recovery Kit では、仮想ハードディスクへのアクセスが失敗したときのローカル・リカバリをサポートしていません。障害が検出されると、デフォルト・アクションによって、バックアップ・サーバへの階層の転送が開始されます。 リソース階層の構成に応じて、このアクションはハング・プロセスにつながることがあります。ハング・プロセスを回避するには、サーバを停止してバックアップ・ サーバへのフェイルオーバを強制するように、デフォルト・アクションを変更します。デフォルトの切り替え動作を変更するには、LifeKeeper デフォルト・ファイルで LKVMDKERROR の設定を変更します。LKVMDKERROR の詳細については、このドキュメントのセクション「 LifeKeeper for VMDK Recovery Kit の設定 」を参照してください。

- クラスタに参加するすべてのゲストはSCSIコントローラの構成を同一にする必要があります。VMDK Recovery Kit はリソースの作成時に仮想ハードディスクが接続されていたSCSIコントローラに対して仮想ハードディスクの再接続を行います。

- リソース起動時にはスナップショットの作成・復元が出来ません。 仮想ハードディスクがノード間を移動して変更された場合にスナップショットとの整合性が保証できなくなるため、明示的に禁止されます。

# 6.20.4. LifeKeeper for Linux VMDK Recovery Kit の設定

本セクションでは、LifeKeeper for Linux VMDK Recovery Kit の設定の詳細について説明します。VMDK Recovery Kit の設定と管理を開始する前に考慮すべき事項についても記載しています。LifeKeeper Core リソース階層の設定方法については、SPS for Linux テクニカルドキュメンテーション を参照してください。

# 6.20.4.1. 設定上の考慮事項

LifeKeeper for Linux VMDK Recovery Kit を操作する前に、次の事項を検討する必要があります。

1. 仮想ハードディスクを共有し、かつ VMDK リソース階層を拡張するクラスタ構成内のサーバへ、VMDK Recovery Kit をインストールします。仮想ハードディスクは vmdk 形式で作成します。

2. 仮想ハードディスクは、クラスタを構成するゲスト間で共有されているデータストア上に作成する必要があります。

3. 仮想ハードディスクの排他制御はハイパーバイザーに依存しているため、接続する SCSI コントローラの共有設定を「なし」に設定する必要があります。

4. このキットは VMware の提供する API を使用して仮想ハードディスクを操作するため、クラスタに参加するゲストを実行しているすべての VMware ESXi ホスト、もしくは管理している vCenter Server へ https でアクセスできる必要があります。

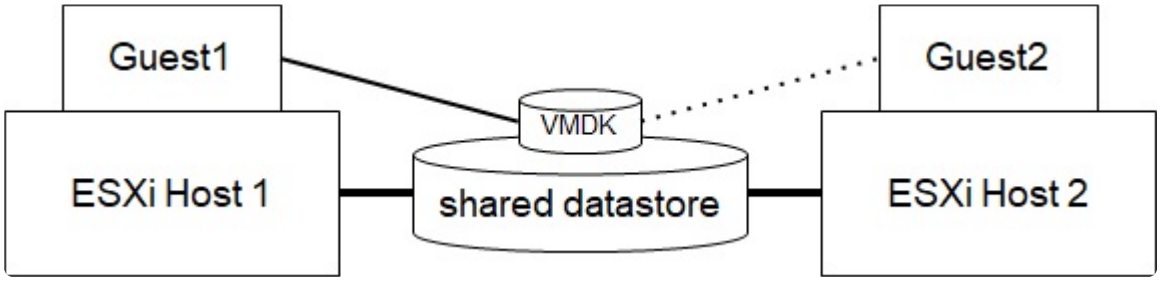
5. VMDK 階層をビルドするために使用するビルトイン・ファイル・システムのリカバリー・キットでは、フェイルオーバ条件でマウントされたファイル・システムを使用 している LifeKeeper 保護下に入っていないプロセスを検出したり、除去したりすることはできません。そのため、LifeKeeper 保護下に入れるプロセスのみが、VMDK 保護下のファイル・システムを使用するように設定することを強くお勧めします。

6. 調整可能パラメータ LKVMDKERROR は、 仮想ハードディスクへのアクセス権を失った際に VMDK Recovery Kit が実行するアクションを制御します。調整可能パラメータには、2 つの 値 ( **halt** 及び **event** ) があり、 **halt** がデフォルトです。値が **halt** に設定され、アクセス権を失ったことを検出すると、VMDK Recovery Kit は即座にシステムを停止し、バックアップ・サーバへのフェイルオーバを発生させます。 **event** に設定された場合、アクセス権を失うと LifeKeeper core にディスクが異常であることを通知し、通知を受けた LifeKeeper core は対応する処理を起動します。この時行われる処理は上位リソースに依存して決定されます。

# 6.20.4.2. 構成例

フェイルオーバが発生したときに、 **LifeKeeper for Linux VMDK Recovery Kit** を使用しているとどのような処理が行われるか、その例をいくつか示します。

## 構成 1：アクティブ／スタンバイ設定の例

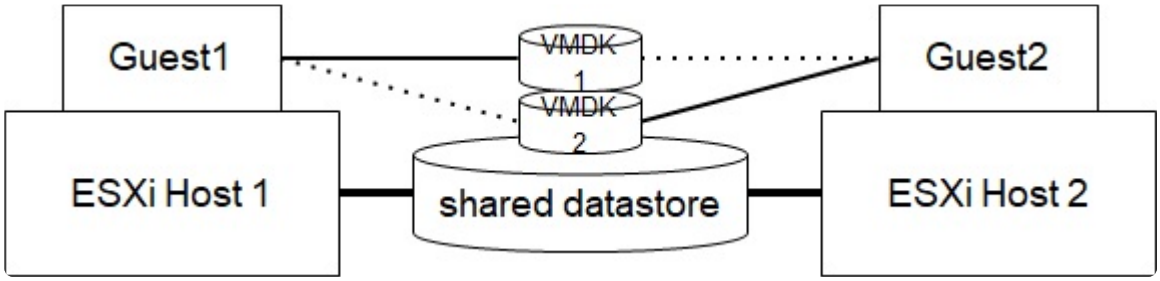



この構成では、ESXi Host 1 にある Guest 1がVMDK Recovery Kit ソフトウェアにより、仮想ハードディスクへアクセスできるため、アクティブとみなされます。Guest 2 は別の処理を実行します。 Guest 1 に障害が発生した場合、Guest 2 が VMDKへのアクセス権を獲得しファイル･システムへアクセスできるようにします。

**設定の注意点：**

- VMDK Recovery Kit を両方のゲストOSにインストールしておく必要があります。
- ファイル･システムは共有したVMDK仮想ハードディスク上に作成します。
- Guest 1 がアクティブである間、Guest 2 は 共有仮想ディスク上のファイルおよびディレクトリにアクセスできません。

## 構成 2：アクティブ／アクティブ設定の例




アクティブ／アクティブ設定は、VMDK Recovery Kit ソフトウェアをアクティブに実行し、 かつ個別の仮想ハードディスク接続した複数のシステムで構成されます。

**設定の注意点：**

- VMDK Recovery Kit を両方のゲストOSにインストールしておく必要があります。
- 最初に、Guest 1 が一つの仮想ハードディスクへ接続して、Guest 2 が別の仮想ハードディスクに接

続します。切り替え時には、1 つのシステムで両方の仮想ハードディスクへ接続します。

# 6.20.5. LifeKeeper 設定作業

LifeKeeper for Linux VMDK Recovery Kit の管理作業はすべて、LifeKeeper グラフィカル・ユーザ・インタフェース（GUI）で実行できます。LifeKeeper GUI では、VMDK リソースの設定、管理、監視作業をガイド付きで行えます。

LifeKeeper for Linux VMDK Recovery Kit の設定に関しては、次の作業が可能です。

- ESXiホストの登録 。仮想ハードディスクを管理している ESXi ホストの情報を登録します。
- VMオプションの変更 。VMDK Recovery Kit で必要なオプションを設定します。
- リソース階層の作成 。VMDK リソース階層を作成します。
- リソース階層の削除 。VMDK リソース階層を削除します。
- リソース階層の拡張 。VMDK リソース階層をプライマリ・サーバからバックアップ・サーバに拡張します。
- リソース階層の拡張解除 。LifeKeeper クラスタの 1 つのサーバの VMDK リソース階層を拡張解除（削除）します。
- 従属関係の作成 。既存のリソース階層と別のリソース・インスタンスとの間に子従属関係を作成し、クラスタ内のすべての対象サーバに従属関係の変化を通知します。
- 従属関係の削除 。リソースの従属関係を削除して、クラスタ内のすべての対象サーバ に従属関係の変化を通知します。
- In Service 。リソース階層をアクティブにします。
- Out of Service 。リソース階層を非アクティブにします。
- 表示 / 編集 。 リソース階層のプロパティを表示または編集します。

**注記：** このセクションの残りの部分では、設定作業はすべて **[Edit]** メニューから実行することになっています。この設定作業の大半は次の方法でも実行できます。

1. ツールバーから状態表示の左ペインにあるグローバル・リソースを右クリック。
2. 状態表示の右ペインにあるリソース・インスタンスを右クリック。

※右クリックの方法を使用すると、 **[Edit]** メニューを使用したときに必要な情報を入力しないですみます。

# 6.20.5.1. ESXiホストの登録

VMDK リソースを作成する前に、ESXi ホストの情報を登録する必要があります。以下の手順で登録してください。

1. コンソール画面で以下のコマンドを実行してください。

   ```
   # /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -a <ESXi ホスト名>
   ```

   コマンドを実行すると、対話的にユーザ名およびパスワードを問い合わせてくるので、ESXi ホストにログインすることのできるユーザ・パスワードを入力してください。
   正しくログインできなかった場合はエラーになり登録できません。

2. クラスタを構成する各 ESXi ホストを同じように登録して下さい。

3. すべての ESXi ホストが登録できたら、以下のコマンドで正しく登録できていることを確認してください。

   ```
   # /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -l
   ```

   登録済みのホスト一覧が出力されます。

4. クラスタ内のすべてのノードで同様に ESXi ホストを登録して下さい。

`esxi_register` コマンドの詳細は「 メンテナンス 」を参照して下さい。

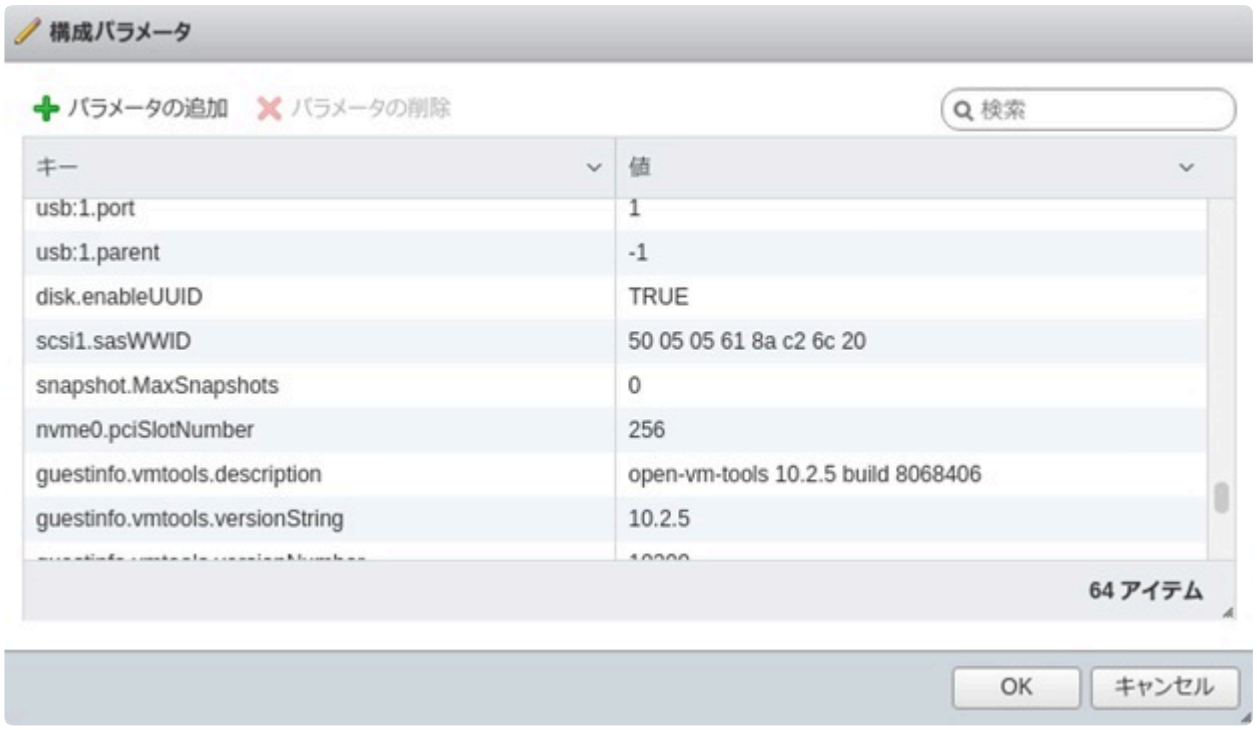
# 6.20.5.2. 仮想マシンオプションの設定変更

VMDK Recovery Kitでは、以下のオプションを設定する必要があります。

| キー | 値 |
| --- | --- |
| disk.enableUUID | TRUE |

VMの「設定の編集」ー「仮想マシン　オプション」ー「詳細」ー「構成パラメータ」ー「構成の編集」で、編集ダイアログを開いて上記パラメータを追加して下さい。

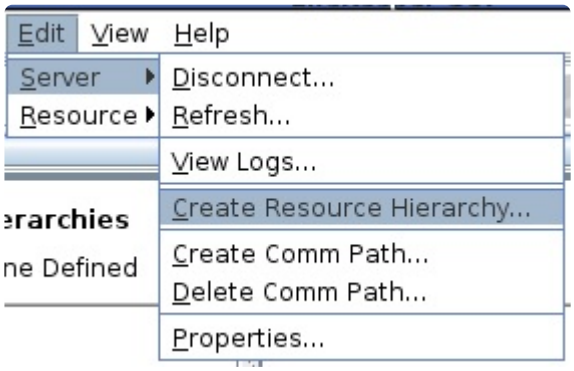
設定の編集: RHEL7.6 (ESXi 6.7 仮想マシン)
詳細
設定
アクセラレーション機能の無効化
ログの有効化
デバッグと統計
正常に実行
スワップ ファイルの場所
デフォルト
仮想マシンを含むクラスタまたはホストの設定を使用します。
仮想マシン ディレクトリ
仮想マシンと同じディレクトリにスワップ ファイルを格納します。
ホストによって指定されたデータストア
構成パラメータ
構成の編集...
遅延感度
正常


構成パラメータ
パラメータの追加
パラメータの削除
検索
キー
値
usb:1.port
1
usb:1.parent
-1
disk.enableUUID
TRUE
scsi1.sasWWID
50 05 05 61 8a c2 6c 20
snapshot.MaxSnapshots
0
nvme0.pciSlotNumber
256
guestinfo.vmtools.description
open-vm-tools 10.2.5 build 8068406
guestinfo.vmtools.versionString
10.2.5
64 アイテム
OK
キャンセル

# 6.20.5.3. リソース階層の作成

プライマリ・サーバで以下を実行して、**[リソース作成ウィザード]** を開始します。

1. **[Edit] > [Server] > [Create Resource Hierarchy]** を選択します。




2. **[リカバリー・キットの選択]** ダイアログが表示されます。ドロップダウン・リストから **[ファイル・システム]** オプションを選択します。VMDK Resource Hierarchy を、仮想ハードディスク上に作成したファイル・システム階層にします。

**[Next]** をクリックして次に進みます。

**注記：** 階層作成手順のどの段階でも、**[Cancel]** ボタンをクリックすると、作成プロセス全体がキャンセルされます。

3. **[スイッチバック・タイプ]** ダイアログが表示されます。スイッチバック・タイプによって、フェイルオーバ後、バックアップ・サーバで VMDK リソースがサービス中（アクティブ）になったとき、VMDK リソースをプライマリ・サーバにスイッチバックする方法が決まります。スイッチバック・タイプは intelligent または automatic です。インテリジェント・スイッチバックの場合、リソースをプライマリ・サーバに戻すために管理者の操作が必要ですが、自動スイッチバックの場合は、プライマリ・サーバがオンラインに戻り、LifeKeeper との通信パスが再確立されるとすぐにスイッチバックが実行されます。

**[Next]** をクリックして次に進みます。

4. **[サーバ]** ダイアログが表示されます。VMDK リソースを作成するサーバの名前を選択します（通常は、これがプライマリ・サーバです）。クラスタ内のすべてのサーバがドロップダウン・リスト・ボックスに表示されます。

**[Next]** をクリックして次に進みます。

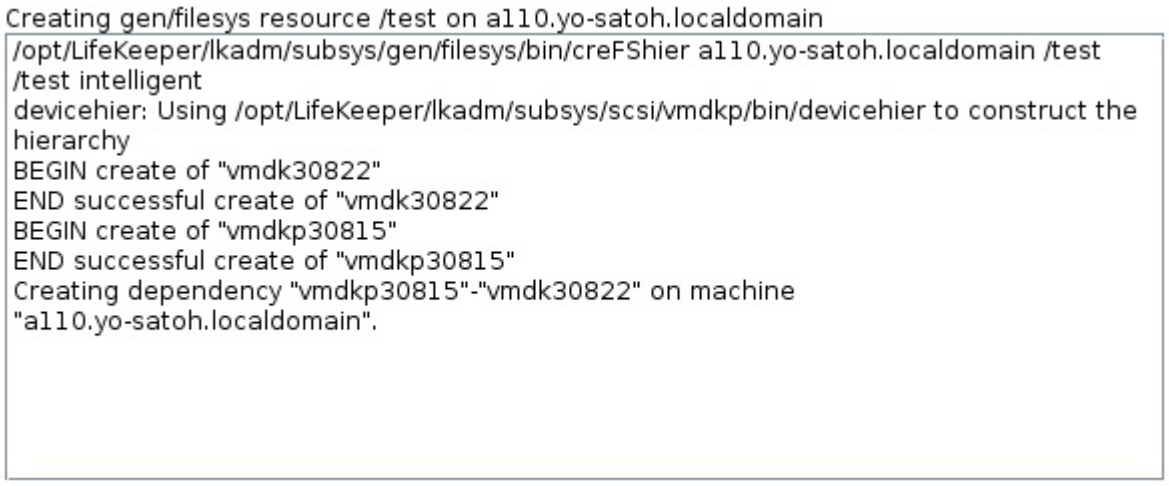
5. VMDK (File System) Resource Hierarchy で保護する **[マウント・ポイント]** パスを選択します。すべての「ローカル」・ファイル・システム（つまり、共有ストレージを使用するファイル・システム） および VMDK Recovery Kit で扱える仮想ハードディスクのマウントポイントがリストされます。ドロップダウン・リストから目的のマウントポイントを選択します。

**[Next]** をクリックして次に進みます。

6. **[ルート・タグ]** ダイアログに、ターゲット・サーバ（つまり、上記で選択したサーバ）上のリソース・インスタンスの固有名が自動的に表示されます。デフォルトを 受け入れることもできますし、文字、数字、および特殊文字-、_、.、または/で構成 された固有のタグ名を入力することもできます。

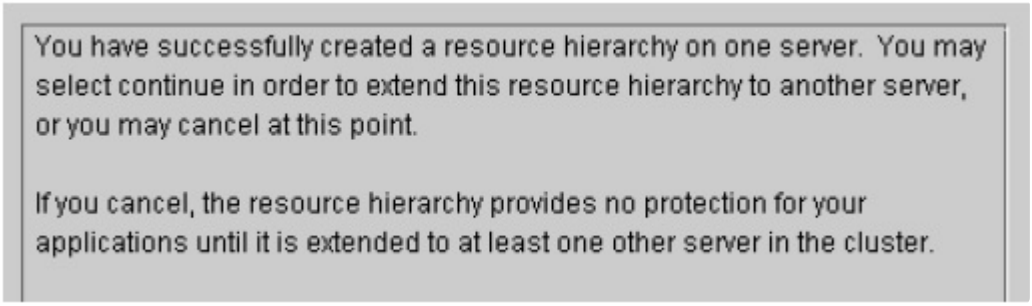
**[Create Instance]** をクリックします。

7. 階層の作成を開始することを示す情報ボックスが表示されます。

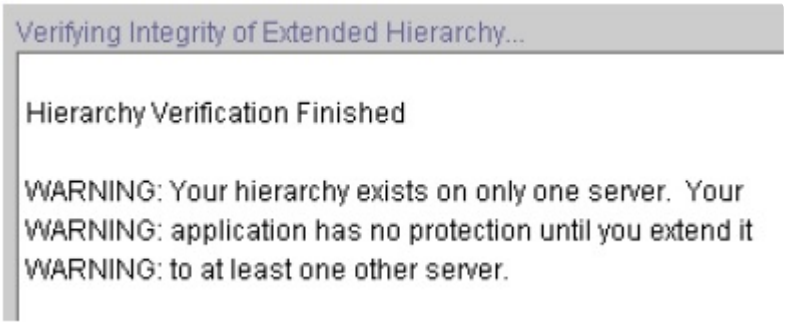



**[Next]** をクリックして次に進みます。

8. VMDK リソース階層の作成に成功したことを示す情報ボックスが表示されます。リソース階層を LifeKeeper で保護するには、クラスタ内の別のサーバにリソース階層を 拡張する必要があります。




リソースを拡張する場合は、 **[Continue]** をクリックします。 後でリソースを拡張する場合は、 **[Cancel]** をクリックします。




9. **[Done]** をクリックして、[リソース階層の作成]メニューを終了します。

# 6.20.5.4. リソース階層の削除

プライマリ・サーバで次の作業を行います。LifeKeeper 構成内のすべてのサーバから VMDK リソースを削除するには、次の手順を実行します。

1. **[LifeKeeper GUI]** メニューから **[Edit]** を選択し、次に **[Resource]** を選択します。ドロップダウン・メニューから、 **[Delete Resource Hierarchy]** を選択します。

2. VMDK リソース階層を削除する **ターゲット・サーバ** の名前を選択します。

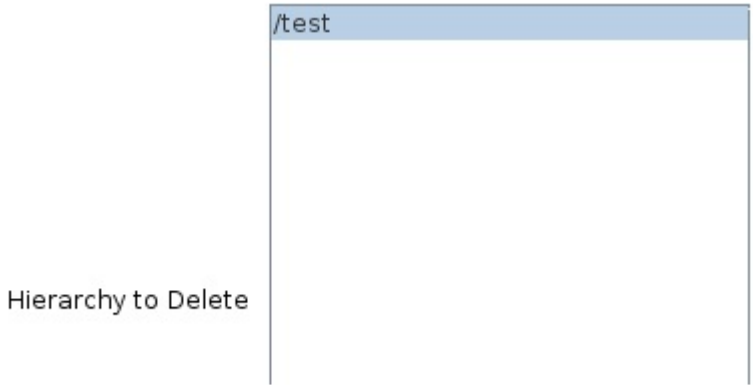
   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインから個々のリソース・インスタンスを右クリックして **[リソース階層の削除]** を選択した場合、このダイアログ・ボックスは表示されません。

   **[Next]** をクリックして次に進みます。

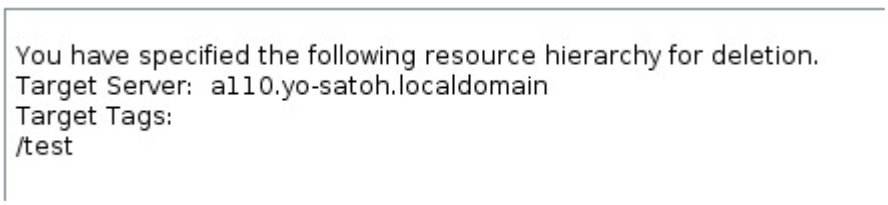
3. **[削除する階層]** を選択します。削除するリソース階層を特定して、強調表示にします。

   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインから個々のリソース・インスタンスを右クリックして **[リソース階層の削除]** を選択した場合、このダイアログ・ボックスは表示されません。

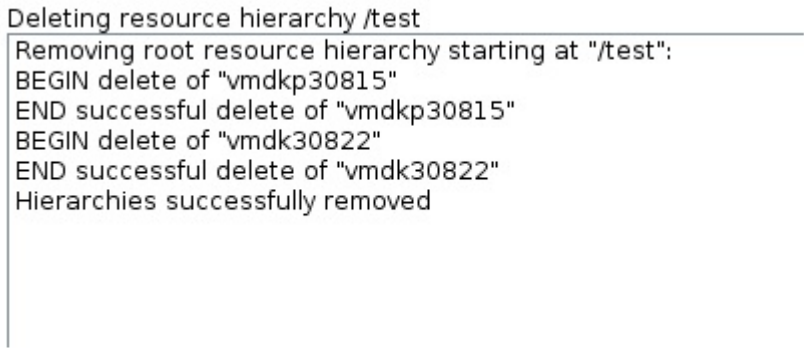
   **[Next]** をクリックして次に進みます。

4. 選択したターゲット・サーバと、削除の対象として選択した階層を確認する情報ボ ックスが表示されます。

**[Delete]** をクリックして次に進みます。

5. File System VMDK リソース・インスタンスの削除に成功したことを示す情報ボックス が表示されます。




6. **[Done]** をクリックして、 **[リソース階層の削除]** メニューを終了します。

# 6.20.5.5. 階層の拡張

階層は、作成後に、クラスタ内の別のサーバに拡張する必要があります。リソース・インスタンスを拡張するには、次の 3 つの方法が可能です。

1. VMDK リソース階層の作成に成功したときに、**[Continue]** を選択して、バックアップ・サーバにリソース階層を拡張する処理に進むことができます。

2. LifeKeeper GUI の左ペインまたは右ペインで拡張されていない階層を右クリックします。

3. LifeKeeper GUI で、ドロップダウン・メニューから **[Edit] > [Resource] > [Extend Resource Hierarchy]** を選択します。選択すると、**[リソース階層の拡張]** ウィザードが起動されます。**[Extend Resource Hierarchy]** オプションの **[Accept Defaults]** ボタンは、LifeKeeperの **[リソース階層の拡張]** の既定値をよく知っていて、値の入力や確認をしないで素早く LifeKeeper リソース階層を拡張したいユーザ向けです。GUI ダイアログを使用して対話的に段階を追って LifeKeeper リソース階層を拡張する場合は、**[Next]** ボタンを選択します。

   a. 最初に表示されるダイアログ・ボックスでは、VMDK リソース階層が現在サービス中であるテンプレート・サーバを選択します。**[テンプレート・サーバ]** で選択するテンプレート・サーバと、次のダイアログ・ボックスの **[拡張するタグ]** で選択する拡張するタグが、サービス中（アクティブ）のリソース階層を表すことを忘れないでください。選択したテンプレート・サーバでサービス中になっていない リソース・タグを選択すると、エラー・メッセージが表示されます。このダイアログのドロップダウン・ボックスには、クラスタ内の全サーバの名前が示されます。

   **注記：** VMDK リソース階層作成後、引き続きリソース階層の拡張作業に入った場合は、作成段階でテンプレート・サーバが特定されているため、このダイアログ・ボックスは表示されません。これは、GUI ウィンドウの左ペインの VMDK リソース・アイコンまたは右ペインの VMDK (File System) リソース・ボックスを右クリックして、**[リソース階層の拡張]** を選択した場合も同様です。

   **注記：** 階層拡張の途中で **[Cancel]** ボタンをクリックすると、拡張処理が取り消されます。ただし、すでにリソースを別のサーバに拡張している場合は、明示的に拡張解除するまで、そのインスタンスの拡張は有効です。

   **[Next]** をクリックして次に進みます。

   b. **[拡張するタグ]** を選択します。これは、テンプレート・サーバからターゲット・サーバに拡張する VMDK インスタンスの名前です。ウィザードのドロップダウン・リスト・ボックスには、テンプレート・サーバ上に作成されているすべてのリソースが表示されます。

   **注記：** ここでも、VMDK リソース階層の作成後すぐにリソース階層の拡張作業に進んだ場合は、作成段階ですでにリソースのタグ名が特定されているため、このダイアログ・ボックスは表示されません。これは、GUI ウィンドウの左ペインの VMDK (File System) リソース・アイ

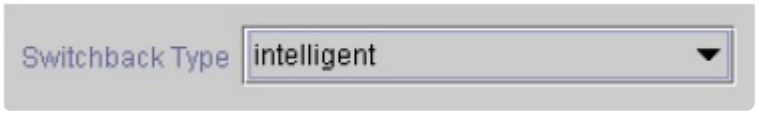
コンまたは右ペインの VMDK (File System) リソース・ボックスを右クリックして、[リソース階層の拡張]を選択した場合も同様です。

**[Next]** をクリックして次に進みます。

c. VMDK リソース階層を拡張するターゲット・サーバを選択します。

**[Next]** をクリックして次に進みます。

d. **[スイッチバック・タイプ]** ダイアログが表示されます。スイッチバック・タイプによって、フェイルオーバ後、バックアップ・サーバで VMDK リソースがサービス中（アクティブ）になったとき、VMDK リソースをプライマリ・サーバにスイッ チバックする方法が決まります。スイッチバック・タイプは intelligent または automatic です。インテリジェント・スイッチバックの場合、リソースをプライマリ・サーバに戻すために管理者の操作が必要ですが、自動スイッチバックの場合 は、プライマリ・サーバがオンラインに戻り、LifeKeeper との通信パスが再確立されるとすぐにスイッチバックが実行されます。



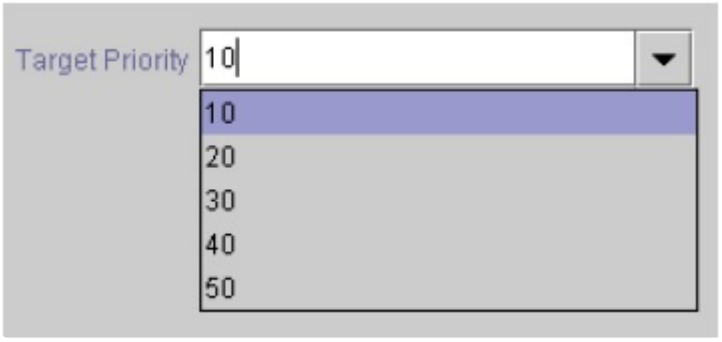

**[Next]** をクリックして次に進みます。

e. テンプレートの優先順位を選択または入力します。これはサーバで現在サービス中の VMDK 階層の優先順位です。優先順位は、1～999 の範囲で未使用の値が有効で、小さい数字ほど優先順位が高くなります（数字 1 が最高の優先順位に相当します）。拡張操作時には、別のシステムで既に使用中の優先順位をこの階層に対して指定することはできません。デフォルト値を推奨します。

**注記：** このフィールドは階層を最初に拡張するときだけ表示されます。

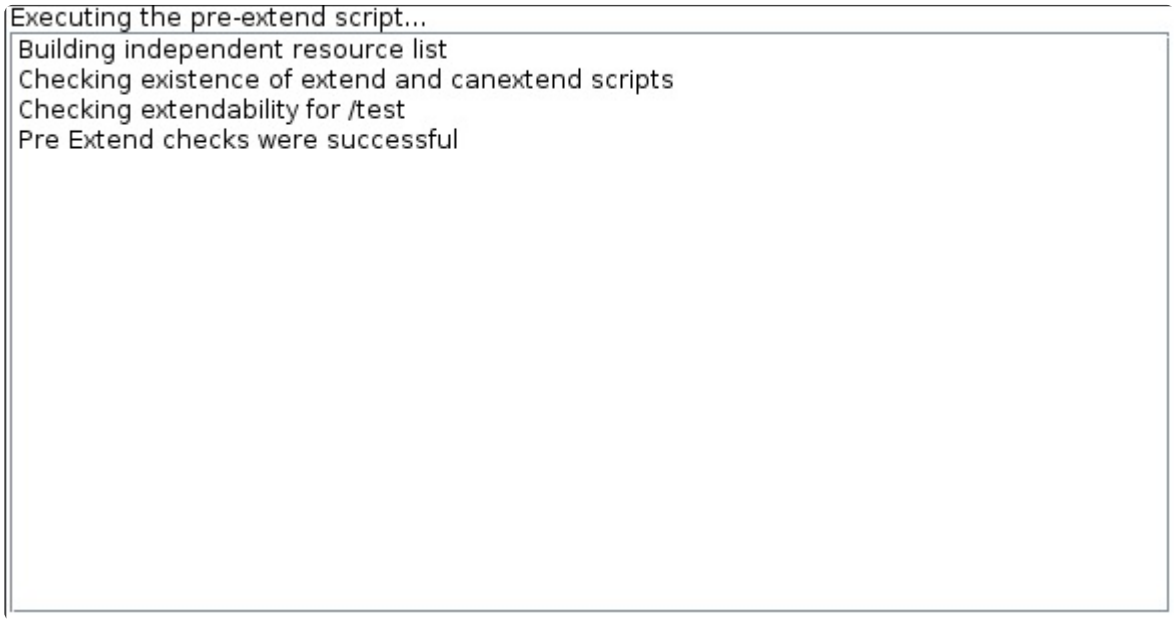
**[Next]** をクリックして次に進みます。

f. ターゲットの優先順位を選択または入力します。これは、別サーバにある同等の階層に対する新しく拡張する VMDK 階層の優先順位です。1～999 の範囲の未使用の値が有効で、リソースの連鎖フェイルオーバ・シーケンスにおけるサーバの優先順位を示します。数値が小さいほど優先順位は高くなります（1 は最高の優先順位を表わします）。LifeKeeper のデフォルトでは、階層が作成されたサーバに「1」が割り当てられることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。




**[Next]** をクリックして次に進みます。

g. 環境がチェックされ、このリソースを拡張するのに必要な要件がすべて満たされていることを示す情報ボックスが表示されます。満たされていない要件がある場合、 **[Next]** ボタンが使用できなくなり、 **[Back]** ボタンが有効になります。




リソースの拡張の設定を変更する場合は、 **[Back]** ボタンをクリックします。 後でリソースを拡張する場合は、[キャンセル]をクリックします。リソース階層の拡張を実行する場合は、 **[Next]** をクリックします。

**[Finish]** をクリックして、VMDK リソース・インスタンスの拡張に成功したことを確認します。

4. **[Done]** をクリックして、 **[リソース階層の拡張]** メニューを終了します。

**注記：** 必ず両方のサーバで新しいインスタンスの機能をテストしてください。

# 6.20.5.6. 階層の拡張解除

1. LifeKeeper GUI メニューで、 **[Edit] > [Resource] > [Unextending Resource Hierarchy]** を選択します。

2. VMDK リソースを拡張解除する **[ターゲット・サーバ]** を選択します。リソースが現在サービス中（アクティブ）のサーバは選択できません。

   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインから個々のリソース・インスタンスを右クリックして拡張解除作業を選択した場合、 このダイアログ・ボックスは表示されません。

   **[Next]** をクリックして次に進みます。

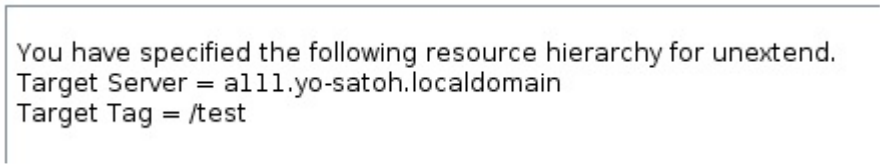
3. **[拡張解除する階層]** を選択します。

   **注記：** 左側のペインからグローバル・リソースを右クリックするか、右側のペインから個々のリソース・インスタンスを右クリックして拡張解除作業を選択した場合、 このダイアログ・ボックスは表示されません。

   **[Next]** をクリックして次に進みます。

4. 拡張解除するために選択したターゲット・サーバと VMDK リソース階層を確認する情報ボックスが表示されます。

   **[Unextend]** をクリックします。

5. VMDK リソースの拡張解除に成功したことを示す別の情報ボックスが表示されます。

6. **[Done]** をクリックして、 **[リソース階層の拡張解除]** メニューを終了します。

# 6.20.5.7. リソース階層のテスト

VMDK リソース階層のテストは、手動でサーバを切り替えて、プライマリ・サーバからバックアップ・サーバへのリソース・インスタンスのフェイルオーバをシミュレートすることによって行えます。

## LifeKeeper GUI からの手動切り替えの実行

LifeKeeper GUI で **[Edit] > [Resource] > [In Service]** を選択すると、手動切り替えを開始できます。たとえば、バックアップ・サーバでサービス中リクエストが実行されると、VMDK リソース階層がバックアップ・サーバ側でサービス中になり、プライマリ・サーバ側ではサービス休止になります。この時点で、元のバックアップ・サーバがプライマリ・サーバになり、元のプライマリ・サーバがバックアップ・サーバになります。

**[Our of Service]** を実行した場合、リソース階層はサービス休止になりますが、別のサ ーバでサービス中にはなりません。

# 6.20.5.8. メンテナンス

## ノード障害によるフェイルオーバーからの復旧

ESXi ホストの障害によりフェイルオーバーが発生した場合、仮想ハードディスクの切り離しが行えないため、停止したゲスト側にも仮想ハードディスクが接続されたままになります。
このため、復帰時に複数のゲストが仮想ハードディスクへアクセスを試みますが、その操作はハイパーバイザにより制限されるため、ゲストは起動できない状態になります。
このような状態が発生した場合には、 v Sphere client を使用して、ゲストから仮想ハードディスクを切り離すことで復帰できます。

## ESXi ログイン情報の変更

ESXi ホストのユーザー名･パスワードを変更する場合は以下の手順を実行してください。

1. LifeKeeper もしくはすべての VMDK リソースを停止します。
2. コマンドラインから以下のコマンドを実行して下さい。
   # /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -u <ESXi ホスト名>
   対話的に新しいユーザー名およびパスワードを入力してください。正常にログインできた場合には、新しい情報に更新されます。
   VMDK リソースが起動している場合やログインできない場合はエラーになり、情報は更新されません。
3. 停止した LifeKeeper もしくは VMDK リソースを起動します。
4. すべてのノードに対して同様の操作をおこない、ログイン情報を更新します。

## ESXi ホスト情報の削除

登録している ESXi ホストの情報を削除する場合には、以下の手順を実行して下さい。

1. LifeKeeper もしくはすべての VMDK リソースを停止します。
2. コマンドラインから以下のコマンドを実行して下さい。
   # /opt/LifeKeeper/lkadm/subsys/scsi/vmdk/bin/esxi_register -d <ESXi ホスト名>
3. 停止した LifeKeeper もしくは VMDK リソースを起動します。
4. すべてのノードに対して同様の操作をおこない、ホスト情報を削除してください。

## esxi_register詳細

| | |
|---|---|
| ホストの登録 | esxi_register -a <ESXi ホスト名> |
| ホストの削除 | esxi_register -d <ESXi ホスト名> |
| ログイン情報の更新 | esxi_register -u <ESXi ホスト名> |
| 登録済みホストの一覧 | esxi_register -l |

# 6.20.6. トラブルシューティング

| 症状 | 考えられる要因 |
| --- | --- |
| リソース作成時にマウントポイントが選択肢にでない。 | 以下の要因が考えられます。<br>• PowerShell / PowerCLIがインストールされていない<br>• ESXiホストが登録されていない<br>• disk.enableUUIDパラメータが設定されていない<br>• 仮想ハードディスクが共有されていないデータストア上にある<br>• SCSIコントローラの共有設定が「仮想」もしくは「物理」になっている<br>/var/log/lifekeeper.logに詳細なエラーが記録されているので、内容を確認して設定を見直して下さい。 |
| LifeKeeperを停止するとシステムが停止する | サービスの停止を伴わないLifeKeeperの停止を実施した場合、システム全体が停止する可能性があります。詳細については こちら をご確認ください。 |

# 6.20.6.1. エラーメッセージ

本セクションでは、LifeKeeper VMDK リソース階層の作成、拡張時、またはリソースの削除、復元時に表示される可能性のあるメッセージの一覧を示します。必要に応じて、エラーの原因およびエラー状態を解消するために必要な処置についても説明しています。

その他の LifeKeeper コンポーネントからもメッセージが表示される可能性があります。 そのような場合は、適切な LifeKeeper コンポーネントのマニュアルを参照してください。

このセクションのメッセージは、次のトピックに分類されています。

- 共通エラー・メッセージ
- 階層の作成
- 階層の拡張
- 階層の削除、復元、リカバリー

## 共通エラーメッセージ

| エラー番号 | エラーメッセージ |
|---|---|
| 000002 | Usage error （使用法のエラー） |
| 000010 | Error getting resource information （リソース情報取得時のエラー） |
| 000011 | Both Tag and ID name not specified （タグ名および ID 名が指定されていません） |
| 000019 | Resource not found on local server （ローカル・サーバにリソースが見つかりません） |
| 000022 | END failed hierarchy <tag name> in service on server <server name> （サーバ <サーバ名> でサービス中の階層 <タグ名> の END が失敗しました） |
| 000026 | END failed ACTION for <tag name> on server <server name> due to <signal> signal （<シグナル> シグナルのためにサーバ <サーバ名> の <タ グ名> ACTION の END が失敗しました） |

## 階層の作成

| エラー番号 | エラーメッセージ |
|---|---|
| 000012 | Usage error（使用法のエラー） |
| 000013 | Error getting resource information（リソース情報取得時のエラー） |
| 000014 | Resource with either matching tag <tag name> or ID exists （タグ <タグ名> または ID が一致するリソースが存在します） |
| 000015 | ins_create failed on server <server name> （サーバ <サーバ名> で ins_create に失敗しました） |

| | |
|---|---|
| 000018 | Error creating resource <tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の作成エラー） |
| 000021 | Removing resource instance < tag name> from server <server name> due to an error during creation （作成時のエラーにより、リソース・インスタンス <タグ 名> をサーバ <サーバ名> から削除しています） |
| 000023 | Error bringing resource < tag name> in service on server <server name> （サーバ <サーバ名> でリソース <タグ名> をサービス中 にするときにエラー） |
| 000024 | Failed resource creation of resource < tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の作成に失敗しました） |
| 000027 | Removing file system dependency from <parent tag> to <child tag> on server <server name> due to an error during creation （作成時のエラーにより、サーバ <サーバ名> の <親タグ> から <子タグ> へのファイル・システム従属関係を削除し ています） |
| 000028 | Removing file system hierarchy <filesys tag> created by <parent tag> on server <server name> due to an error during creation （作成時のエラーにより、サーバ <サーバ名> の <親タグ> によって作成されたファイル・システム階層 <ファイルシステム・タグ> を削除しています） |
| 000029 | Switchback type mismatch between parent <parent tag> and child <child tag> on server <server name> （サーバ <サーバ名> の親 <親タグ> と子 <子タグ> のスイ ッチバック・タイプが一致していません）<br><br>**処置：** スイッチバック・タイプの不一致は予想外の動作を 引き起こすことがあります。ins_setas コマンドを使用して スイッチバック・タイプを手動で変更することによって、 この不一致をなくすことができます。 |
| 000030 | create: tag name not specified （作成時：タグ名が指定されていません）<br><br>または<br><br>extend: tag name not specified （拡張時：タグ名が指定されていません） |

## 階層の拡張

| エラー番号 | エラーメッセージ |
|---|---|
| 000003 | Template resource < tag name> on server <server name> does not exist （サーバ <サーバ名> にテンプレート・リソース <タグ名> が存在しません） |
| 000004 | Template resource < tag name> cannot be extended to server <server name> because it already exists there （テンプレート・リソース <タグ名> がサーバ <サーバ名> に すでに存在するため拡張できません） |
| 000005 | Cannot access canextend script on server <server name> （サーバ <サーバ名> の canextend スクリプトにアクセスできません） |
| 000006 | Cannot access extend script <path to extend> on server <server name> （サーバ <サーバ名> の extend スクリプト <extend へのパス> にアクセスできません） |

| | |
|---|---|
| 000007 | Cannot access depstoextend script <path to depstoexend> on server <server name> （サーバ <サーバ名> の depstoextend スクリプト <depstoextend へのパス> にアクセスできません） |
| 000008 | Cannot extend resource < tag name> to server <server name> （サーバ <サーバ名> にリソース <タグ名> を拡張できません） |
| 000009 | Either <templatesys> or <templatetag> argument missing （<テンプレート・システム> または <テンプレート・タグ> 引数が指定されていません） |
| 000014 | Resource with either matching tag <tag name> or ID exists （タグ <タグ名> または ID が一致するリソースが存在します） |
| 000015 | ins_create failed on server <server name> （サーバ <サーバ名> で ins_create に失敗しました） |
| 000018 | Error creating resource <tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の作成エラー） |
| 000025 | END failed resource extension of < tag name> on server <server name> due to a "<signal>" signal - backing out changes made to server （"<シグナル>" シグナルのためにサーバ <サーバ名> の <タグ名> リソース拡張の END が失敗しました - サーバに加えた変更を元に戻しています） |
| 000030 | create: tag name not specified （作成時：タグ名が指定されていません）<br><br>または<br><br>extend: tag name not specified （拡張時：タグ名が指定されていません） |

### 復元

| エラー番号 | エラーメッセージ |
|---|---|
| 000023 | Error bringing resource < tag name> in service on server <server name> （サーバ <サーバ名> でリソース <タグ名> をサービス中にするときにエラー） |

### リソース監視

| エラー番号 | エラーメッセージ |
|---|---|
| 000001 | Calling sendevent for resource < tag name> on server <server name> （サーバ <サーバ名> でリソース <タグ名> の sendevent を呼び出 しています） |

## VMDK Recovery Kit のエラーメッセージ

| エラー番号 | エラーメッセージ |
|---|---|

| | |
|---|---|
| 137000 | PowerShell is not installed. (PowerShell がインストールされていません) |
| 137001 | PowerCLI is not installed. (PowerCLI がインストールされていません) |
| 137002 | A valid network interface was not found. (ネットワークインタフェースが見つかりません) |
| 137003 | Attaching VMDK $vmdkfile. ($vmdkfile をアタッチします) |
| 137004 | VMDK already attached. (VMDKファイルがすでにアタッチ済みです) |
| 137005 | Failed to attach VMDK. (VMDKファイルのアタッチに失敗しました) |
| 137006 | Attach success. (VMDK をアタッチしました) |
| 137008 | Detaching VMDK $vmdkfile. ($vmdkfile をデタッチします) |
| 137009 | VMDK Already detached. (VMDKファイルがすでにデタッチ済みです) |
| 137010 | Failed to detach VMDK. (VMDKファイルのデタッチに失敗しました) |
| 137011 | Detach success. (VMDK をデタッチしました) |
| 137012 | Restarting VMDK status checker daemon. (監視デーモンを再起動します) |
| 137016 | Stopping VMDK status checker daemon. (監視デーモンを停止します) |
| 137020 | Failed to execute VMDK status checker daemon. (監視デーモンの起動に失敗しました) |
| 137026 | Flushing $dev ($dev のバッファをフラッシュします) |
| 137027 | Skipping flush for $dev. ($dev のバッファはフラッシュしません) |
| 137030 | Disk not specified. (ディスクが指定されていません) |
| 137031 | Cannot get disk uuid for $Disk. Please check your ESXi settings. (ディスクのIDが取得できないので、ESXiの設定を確認してください) |
| 137032 | PowerCLI failed. %s (PowerCLI のエラーです) |
| 137034 | Cannot bring VMDK resource \"%s\" in service on server \"%s\". (VMDK リソースの起動に失敗しました) |
| 137035 | Error detected conflict in expected tag name \"%s\" on target machine \"%s\". (タグ名が重複しています) |
| 137036 | Template resource \"%s\" on server \"%s\" does not exist. (テンプレートのタグ名が存在しません) |
| 137037 | This system is not a VMware guest. (VMwareのゲストではありません) |
| 137038 | Unable to find shared device on \"%s\" for \"%s\". (共有可能な VMDK がありません) |
| 137039 | Unable to find SCSI controller on \"%s\" for \"%s\". (SCSI コントローラが見つかりません) |
| 137050 | Failed to connect to ESXi server $addr. (ESXi ホストの接続に失敗しました) |
| 137051 | There is no ESXi server connected. (ESXi ホストに接続していません) |
| 137055 | Cannot determine ESXi VM ID because multiple network interfaces were found with the MAC address $MAC_ADDR. ($MAC_ADDR のネットワークインタフェースが複数見つかったため、自分のIDを取得できません) |

| | |
|---|---|
| 137056 | Failed to $action VMDK $VMDK_FILENAME. Retrying $action in $wait_sec milliseconds. ($VMDK_FILENAME の操作に失敗したため、$wait_sec 秒間待機します) |
| 137057 | Usable SCSI controller not found. (使用可能な SCSI コントローラが見つかりません) |
| 137058 | Cannot find VMDK with ID $UUID. ($UUID の VMDK が見つかりません) |
| 137059 | This guest has snapshots present. (スナップショットがあります) |
| 137060 | The VMDK with ID $UUID cannot be attached to this guest. ($UUID の VMDK はこのゲストにアタッチできません) |
| 137061 | The virtual storage controller has an incompatible sharing mode configured. (SCSI コントローラの共有モードが正しく設定されていません) |
| 137062 | VMDK_TIMER too short. Using default value. (VMDK_TIMER の設定値が短すぎるため、初期値を使用します) |
| 137063 | Calling sendevent for resource `"$Tag`" on server `"$me`"“ (リソース $Tag でイベントが発生しました) |
| 137064 | skip quickcheck for `"$Tag`" on server `"$me`", sendevent pending. (イベント発生中のため q監視処理を中止します) |
| 137065 | sendevent issued for tag `"$Tag`" has not finished, halt server `"$me`" (イベントの通知に失敗しました) |
| 137066 | skip quickcheck for `"$Tag`" on server `"$me`", sendevent pending. (イベント発生中のため q監視処理を中止します) |
| 137068 | The VMDK detection failed. Retry count exceeded. (規定時間内にディスクが認識されません) |
| 137070 | Connect failed. (ESXi に接続できません) |
| 137071 | Get-LocalVM failed. (自VMの情報が取得できません) |
| 137072 | The VMDK is remote detached. This server has lost ownership. (リモートでデタッチ要求が発行されたため、VMDK のアクセス権を失いました) |
| 137073 | The VMDK quickCheck daemon has been stopped. (監視デーモンが停止しています) |
| 137074 | VMDK_RETRY too small. Using default value. (VMDK_RETRY が小さすぎるため、初期値を使用します) |
| 137075 | Cannot find virtual SCSI controller $CONTROLLER. (SCSIコントローラーがありません) |
| 137076 | The virtual storage controller has an incompatible sharing mode configured. (SCSIコントローラーの設定が正しくありません) |
| 137077 | Cannot find VM with MAC address $MAC_ADDR. ($MAC_ADDR のネットワークを使用するVMがありません) |
| 137078 | Cannot find VM with MAC address $MAC_ADDR. ($MAC_ADDR のネットワークを使用するVMがありません) |
| 137100 | Re-reading partition table on %s. (パーティション情報を再読込します) |

| | |
|---|---|
| 137101 | Partition information not defined for %s on %s. Retry. (パーティション情報を読み込めないので再試行します) |
| 137102 | Partition information not defined for %s on %s. (パーティションがありません) |
| 137103 | Resource %s is OSF, Skip flushing buffers. (リソースが OSF 状態なので、バッファのフラッシュを行いません) |
| 137104 | Flushing buffers on %s. (バッファをフラッシュします) |
| 137105 | Device not specified. (デバイス名の指定がありません) |
| 137106 | Cannot get device uuid for $Device. Please check your ESXi settings. (デバイスのUUIDが取得できないため、ESXiの設定を確認してください) |
| 137107 | %s is not shareable with any machine. (VMDK が共有されていません) |
| 137109 | Creating dependency \”%s\”-\”%s\” on machine \”%s\”. (依存関係を作成します) |
| 137110 | Dependency \”%s\”-\”%s\” on machine \”%s\” exists. (すでに依存関係が存在します) |
| 137111 | Failed to create dependency \”%s\”-\”%s\” on machine \”%s\”. (依存関係の作成に失敗しました) |
| 137112 | Cannot bring VMDKP resource \”%s\” in service on server \”%s\”. (VMDKP リソースの起動に失敗しました) |
| 137113 | detected conflict in expected tag name \”%s\” on target machine \”%s\”. (タグ名が重複しています) |

# 7. パラメーター一覧

## Coreパラメーターの一覧

下記の表は、Coreパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| FS_KERNEL_RETRIES | SIGTERMやSIGKILLシグナルによりファイルシステムにアクセスしているプロセスが停止できなかった場合に、forceumountスクリプトがファイルシステムのアンマウントを試行する最大回数。アンマウントは最大回数を超えるまで3秒毎に試行されます。 | 整数値 | 60 | 適宜(すぐに反映されます) | |
| FS_UMOUNT_RETRIES | アンマウントしたいファイルシステムにアクセスしている全プロセスを停止するためにforceumountスクリプトがSIGTERMシグナルを発行する最大回数。試行回数が最大回数を超えた場合、forceumountスクリプトはSIGKILLシグナルによるプロセス停止を最大3回試行します。 | 整数値 | 1 | 適宜(すぐに反映されます) | |
| REMOTETIMEOUT | 他のマシンから予測される応答がくる前に”lcdsendremote”機能により送信要求処理をするための時間です。この時間を経過しても応答がない場合は、代替の経路(利用可能な場合)での処理を試みます。この値によってタイムアウトと判定された場合には、リソース切り替えに失敗します。常にタイムアウトが発生するような環境である場合に | 整数値 | 900 | LifeKeeper起動時(**LifeKeeper再起動時**に反映さ | |

| | | | | | |
|---|---|---|---|---|---|
| | は、この値を伸ばすことによって対応する場合があります。 | | | れます) | |
| CONFIRMSODEF | フェイルオーバ確認を設定している時、マシンのフェイルオーバ処理で行われるデフォルト動作を指定します。デフォルト動作は、タイムアウトの間 （CONFIRMSOTO を参照してください）に管理者から手動で応答がこない場合のみ、行われます。 | 0: フェイルオーバーを続行します。<br>1: フェイルオーバーをブロックします。 | 0 | 適宜(すぐに反映されます) | ［Confirm Failover］と［[Block Resource Failover］の設定 を参照してください。 |
| CONFIRMSOTO | フェイルオーバ確認を設定している時、管理者からの指示が行われるまでの待機時間を秒単位で指定します。タイムアウト時間が経過すると、CONFIMRSODEF で指定したデフォルト動作が行われます。そうでなければ、管理者が指示した動作を実行します。 | 整数値 | 600 | 適宜(すぐに反映されます) | マシンのフェイルオーバの処理は管理者からの確認を受け取るか、タイムアウト時間が終了するまで遅延します。<br>［Confirm Failover］と［Block Resource Failover］の設定 を参照してください。 |
| FAILFASTTIMER | ローカルシステムによってSCSI RESERVEコマンドによりデバイスにリザーブをかける間隔です。リザーブを失った場合は、システムは停止、再起動します。 | 整数値 | 5 | LifeKeeper 起動時(**LifeKeeper 再起動時に反映されます**) | |
| SCSIERROR | SCSIデバイスをオープンできない、接続できない、あるいは他のSCSIエラー(タイムアウト等)が発生したときの動作を決 | event: スイッチオー | event | LifeKeeper 起動時 | SCSIリザベーションが失われた場合は、本パラメーターの設定にかかわらず常に停止にな |

| | | | | | |
|---|---|---|---|---|---|
| | 定します。 | バ<br>halt:<br>データの破損を避けるため即時強制停止、強制再起動 | | (**LifeKeeper再起動時に反映されます**) | ります。 |
| LKCHECKINTERVAL | アプリケーションの監視をする間隔を秒単位で指定します。値を0にした場合、監視が無効になります。 | 0または1以上の整数値 | 120 | LifeKeeper起動時(**LifeKeeper再起動時に反映されます**) | LifeKeeperをアップグレードすると、デフォルト値に上書きされます。設定を変更している場合は、再設定してください。 |
| FILESYSFULLWARN | LifeKeeperログに警告メッセージを出力し始める、ファイルシステム使用量の閾値を指定します。 0 を設定した場合、監視が無効になります。 | 整数値 | 90 | 適宜(すぐに反映されます) | |
| FILESYSFULLERROR | LifeKeeperログにエラーメッセージを出力し始める、ファイルシステム使用量の閾値を指定します。<br>さらに、この閾値に達した時にはLKROOT/events/filesys/diskfull/notify スクリプトが呼び出されます。<br>0 を設定した場合、監視が無効になります。 | 整数値 | 95 | 適宜(すぐに反映されます) | |
| LK_TRAP_MGR | SNMPトラップを受けるネットワークを１つまたは複数（カンマ区切り）指定します。この値を指定していない場合は、SN | 文字列 | （未設定） | 適宜(すぐに反映さ | /opt/LifeKeeper/bin/lk_configsnmp を使用して、この値を設定できます。設定値にスペー |

| | | | | | |
|---|---|---|---|---|---|
| | MPトラップの送信は行いません。 | | | れます) | スを含めないでください。特に複数指定する場合、カンマの前後にスペースを含めないでください。 |
| LK_NOTIFY_ALIAS | LifeKeeperクラスターが特定のイベントが発生したときに通知するメッセージを受けとるために使用するEmailアドレスまたはアドレスリストです。値を指定していない場合は、通知を行いません。使用例は次の通りです。<br>LK_NOTIFY_ALIAS=<br>– 通知を行いません。<br>LK_NOTIFY_ALIAS=user1@domain1<br>– domain1のuser1に通知を行います。<br>LK_NOTIFY_ALIAS=user1@domain1,user2@domain1<br>– domain1のuser1、user2に通知を行います。 | 文字列 | （未設定） | 適宜(すぐに反映されます) | /opt/LifeKeeper/bin/ lk_confignotifyalias を使用して、この値を設定できます。 |
| LKSYSLOGTAG | syslog出力時に使用するタグ名です。 | 文字列 | LifeKeeper | LifeKeeper 起動時(**LifeKeeper 再起動時に反映されます**) | |
| LKSYSLOGSELECTOR | syslogのファシリティです。 | user, daemon, local0, local1, …または local7 | local6 | LifeKeeper 起動時(**LifeKeeper 再起** | |

| | | | | | |
|---|---|---|---|---|---|
| | | | | 動時に反映されます) | |
| LCMHBEATTIME | ハートビート信号を送信する間隔を秒単位で指定します。またこの時間内に対向サーバからハートビート信号を含むLCM信号を受信できなかった場合、ハートビート停止と判断されます。 | 整数値 | 5 | LifeKeeper 起動時(**LifeKeeper 再起動時に反映されます**) | 0を設定すると、デフォルト値が設定されます。 詳細はテクニカルドキュメンテーション「LifeKeeper ハートビートの調整 」を参照してください。 |
| LCMNUMHBEATS | コミュニケーションパスが切断していると判断するハートビートの連続失敗回数を指定します。<br>実装上は回数ではなく、LCMHBEATTIME x LCMNUMHBEATS秒間LCM通信がない場合、コミュニケーションパス切断と判断されます。 | 整数値 | 3 | LifeKeeper 起動時(**LifeKeeper 再起動時に反映されます**) | 0を設定すると、デフォルト値が設定されます。 詳細はテクニカルドキュメンテーション「LifeKeeper ハートビートの調整 」を参照してください。 |
| LC_MESSAGES | 言語環境を変更します。 | 文字列 | C | LifeKeeper 起動時(**LifeKeeper 再起動時に反映されます**) | 値を変更する場合、LifeKeeper の動作に悪影響を及ぼす可能性があることに注意してください。悪影響は、メッセージカタログがさまざまな言語とユーティリティに対応してインストールされているかどうか、および LifeKeeper が予期していないテキスト出力をそれらが生成するかどうかによって異なります。 |

| | | | | | |
|---|---|---|---|---|---|
| GUI_WEB_PORT | LifeKeeper Management Web servers (lkGUI) に使用されるポートを指定します。 | 整数値 | 81 | steeleye-lighttpd再起動 | steeleye-lighttpdの再起動方法： /opt/LifeKeeper/sbin/sv restart /opt/LifeKeeper/etc/service/steeleye-lighttpd |
| API_SSL_PORT | LifeKeeper APIで使用するポートを指定します。 | 整数値 | 778 | steeleye-lighttpd再起動 | steeleye-lighttpdの再起動方法： /opt/LifeKeeper/sbin/sv restart /opt/LifeKeeper/etc/service/steeleye-lighttpd |
| LOGMGR_LOGLEVEL | Generic Applicationのログレベルを指定します。 | LK_INFO または LK_ERROR | LK_ERROR | LK起動時もしくはlk_logmgrプロセスの再起動(**LifeKeeper再起動時に反映されます**) | |

# 7.1. EC2パラメーター一覧

下記の表は、EC2パラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。
Recovery Kit for EC2 のコンポーネントはメモリに常駐していないため、/etc/default/LifeKeeper の値を変更すると即時に反映されます。LifeKeeper を再起動する必要はありません。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| EC2_RESTORE_TIMEOUT | リソースのリストアタイムアウト時間を秒単位で指定します。 | 整数値 | 300 | 適宜（すぐに反映されます） | |
| EC2_REMOVE_TIMEOUT | リソースを停止するタイムアウト時間を秒単位で指定します。 | 整数値 | 300 | 適宜（すぐに反映されます） | |
| EC2_RECOVER_TIMEOUT | ローカルリカバリのタイムアウト時間を秒単位で指定します。 | 整数値 | 300 | 適宜（すぐに反映されます） | |
| EC2_QUICKCHECK_TIMEOUT | quickCheck のタイムアウト時間を秒単位で指定します。 | 整数値 | 100 | 適宜（すぐに反映されます） | |
| EC2_AWS_REGION | EC2 リソースが動作するリージョンを | 文 | （未 | 適宜 | |

| | | | | | |
|---|---|---|---|---|---|
| | 指定します。 | 字列 | 設定) | (すぐに反映されます) | |
| IP_NOLINKCHECK | 保護対象のネットワークインターフェースのリンクチェックを無効に設定します。 | 0: 有効 1: 無効 | 0 | 適宜 (すぐに反映されます) | 本パラメーターは Elastic IP を保護しているときにのみ適用されます。 |
| IP_WAIT_LINKDOWN | 保護対象のネットワークインターフェースをダウン状態にしてからアップ状態に戻すまで待機する秒数を指定します。一部の環境では、この 2 つの動作の間に待機時間が必要です。 | 整数値 | 5 | 適宜 (すぐに反映されます) | 本パラメーターは Elastic IP を保護しているときにのみ適用されます。 |
| IP_MAX_LINKCHK | リンクが修復されてからアップ状態に戻るまで待機する最大時間を秒単位で指定します。一部の環境では、この値を増加する必要があります。 | 整数値 | 5 | 適宜 (すぐに反映されます) | 本パラメーターは Elastic IP を保護しているときにのみ適用されます。 |
| AWSCLI_CONNECT_TIMEOUT | awsコマンドのオプション —cli–connect–timeout の値を秒単位で設定します。 | 整数値 | 10 | 適宜 (すぐに反映されます) | 本パラメーターは Route53 と共通のパラメーターです。 |
| AWSCLI_READ_TIMEOUT | awsコマンドのオプション —cli–read–timeout の値を秒単位で設定します。 | 整数値 | 5 | 適宜 (すぐに反映されます) | 本パラメーターは Route53 と共通のパラメーターです。 |
| HTTP_PROXY HTTPS_PROXY NO_PROXY | サービスエンドポイントへのアクセスで HTTP プロキシを使用する場合に設定してください。ここで設定した値が | 文字列 | (未設定) | 適宜 (すぐに | 本パラメーターは Route53 や Quorum と |

| | | | | | |
|---|---|---|---|---|---|
| | そのまま AWS CLI へ渡されます。<br>詳しくはAWSのドキュメントを参照してください。<br>docs.aws.amazon.com/ja_jp/cli/latest/userguide/cli-http-proxy.html | | | 反映されます) | 共通のパラメーターです。 |

# 7.2. IPパラメーター一覧

下記の表は、IPパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| IP_PINGTRIES | IPヘルスチェック実行時のpingのリトライ数を指定します。 | 整数値 | 3 | 適宜(すぐに反映されます) | |
| IP_PINGTIME | LifeKeeperがIPヘルスチェック時に、pingの パケット１個の 応答を待つ時間を秒単位で指定します。 | 整数値 | 1 | 適宜(すぐに反映されます) | ブロードキャスト ping の替わりに手動で設定した Ping List を使用する場合、この値を 3 秒以上に設定しても効果は得られません。Linux の TCP/IP の実装では、Ping List コマンドにタイムアウト値を設定したとしても、3 秒以上応答を受け取れない場合は常に、「Destination Host Unreachable」エラーを返すためです。 |
| IP_QUICKCHECK_TIMEOUT | クイックチェックが誤判定することがあれば/etc/default/LifeKeeperの設定値を延ばしてください。 | 整数値 | 12 | 適宜(すぐ | この値を設定するために LifeKeeper のプロセスの停止もしくは再起動を行う必要はありません |

| | | | | | |
|---|---|---|---|---|---|
| | IPクイックチェックが完了するまでのデフォルトの時間は12秒です。 | | | に反映されます) | |
| NOIPUNIQUE | IPリソースをIn Serviceにする時、同じIPアドレスがネットワーク上に存在していないかチェックする機能を無効に設定します。デフォルトでは、LifeKeeperはIPアドレスが他のネットワーク上で使用されていないかどうか確認します。 | 0: 有効 1: 無効 | 0 | 適宜(すぐに反映されます) | |
| NOBCASTPING | IPリソースのヘルスチェック時のブロードキャストpingメカニズムを無効に設定します。 | 0: 有効 1: 無効 | 0 | 適宜(すぐに反映されます) | |
| IP_NOLINKCHECK | IPヘルスチェック実行時にリンク状態を確認する設定を無効に設定します。 | 0: 有効 1: 無効 | 0 | 適宜(すぐに反映されます) | この設定は、特に「仮想IPのリンクチェックに失敗しました」というメッセージがログに出力された場合、仮想環境で無効にする必要がある場合があります。 |
| IP_MAX_LINKCHK | リンクが修復されてからアップ状態に戻るまで待機する最大時間を秒単位で指定します。一部の環境では、この値を増加す | 整数値 | 5 | 適宜(すぐに | |

| | | | | | |
|---|---|---|---|---|---|
| | る必要があります。 | | | 反映されます) | |
| IP_WAIT_LINKDOWN | 保護対象のネットワークインターフェースをダウン状態にしてからアップ状態に戻すまで待機する時間を秒単位で指定します。一部の環境では、この 2 つの動作の間に待機時間が必要です。 | 整数値 | 5 | 適宜(すぐに反映されます) | |

# 7.3. MQパラメーター一覧

下記の表は、MQパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| MQS_QUICKCHECK_TIMEOUT_SC | サーバ接続チェックのタイムアウト時間を秒単位で指定します。 | 整数値 | 10 | 適宜(すぐに反映されます) | |
| MQS_QUICKCHECK_TIMEOUT_CC | クライアント接続チェックのタイムアウト時間を秒単位で指定します。 | 整数値 | 10 | 適宜(すぐに反映されます) | |
| MQS_QUICKCHECK_TIMEOUT_PUTGET | PUT/GETチェックのタイムアウト時間を秒単位で指定します。 | 整数値 | 10 | 適宜(すぐ | |

| | | | | | |
|---|---|---|---|---|---|
| | | | | に反映されます) | |
| MQS_QUICKCHECK_TIMEOUT_PS | publish/subscribe の使用有無チェックのタイムアウト時間を秒単位で指定します。 | 整数値 | 5 | 適宜(すぐに反映されます) | |
| MQS_QUICKCHECK_TIMEOUT_CLUSTER | キューマネージャが WebSphere MQ クラスタの一部であるかどうかチェックするタイムアウト時間を秒単位で指定します。 | 整数値 | 5 | 適宜(すぐに反映されます) | |
| MQS_QUICKCHECK_TIMEOUT | quickCheck スクリプトのタイムアウト時間を秒単位で指定します。 | 整数値 | 40 | 適宜(すぐに反映されます) | 10以下を設定した場合はデフォルト値が設定されます。 |
| MQS_QMGR_START_TIMEOUT | キューマネージャの起動コマンド完了のタイムアウト時間を秒単位で指定します。 | 整数値 | 60 | 適宜(すぐに | |

| | | | | | |
|---|---|---|---|---|---|
| | | | | 反映されます) | |
| MQS_CMDS_START_TIMEOUT | コマンドサーバの起動コマンド完了のタイムアウト時間を秒単位で指定します。 | 整数値 | 30 | 適宜(すぐに反映されます) | |
| MQS_LISTENER_START_TIMEOUT | リスナーの起動 コマンド完了のタイムアウト時間を秒単位で指定します。 | 整数値 | 30 | 適宜(すぐに反映されます) | |
| MQS_LISTENER_LIST_TIMEOUT | リスナーの list コマンド完了のタイムアウト時間を秒単位で指定します。 | 整数値 | 10 | 適宜(すぐに反映されます) | |
| MQS_CHECK_TIMEOUT_ACTION | サーバ接続チェック、またはクライアント接続チェックがタイムアウトになった場合のアクションを指定し | ignore: ログに記録するが、リカバリは行 | ignore | 適宜(すぐに反 | |

| | | | | | |
|---|---|---|---|---|---|
| | ます。 | わない<br>sendevent:<br>ローカル<br>リカバリ<br>を行う | | 映されます) | |
| MQS_LISTENER_CHECK_DELAY | リスナーの起動から、リスナー正常起動チェックを行うまでの時間を秒単位で指定します。デフォルト値で、使用状態のポートを検出するのに十分です。 | 整数値 | 2 | 適宜(すぐに反映されます) | |
| NO_AUTO_STORAGE_DEPS | MQ リソース階層作成の間、キューマネージャとログストレージディレクトリに、共有ストレージの確認とファイルシステムリソースの作成を行うかどうか決定します。<br>本パラメーターを 0 に設定すると、これらのタスクが実行されます。1 に設定すると、これらのタスクを回避します。 | 0 または 1 | 0 | 適宜(すぐに反映されます) | |
| MQS_DSPMQVER_TIMEOUT | dspmqver コマンド（WebSphere MQ のバージョンを知るために必要）のタイムアウト時間を秒単位で指定します。 | 整数値 | 5 | 適宜(すぐに反映されます) | 2以下を設定した場合はデフォルト値が設定されます。 |

| | | | | | |
|---|---|---|---|---|---|
| MQS_SKIP_CRT_MISSING_Q | 欠落したテストキューを自動で作成するかどうかを指定します。<br>本パラメーターを 0 に設定すると、欠落したテストキューは自動で作成されます。1 に設定すると、このプロセスはスキップされます。 | 0 または 1 | 0 | 適宜(すぐに反映されます) | |
| MQS_ALT_USER_NAME | すべての WebSphere MQコマンドに別のユーザーネームを使用。デフォールトでは、ユーザー”mqm”が使用される。もし設定されている場合には、ユーザーはそのプライマリーグループをグループ”mqm”に設定するか、そのグループにセカンダリーメンバーシップを持たなくてはならない。 | 文字列 | 設定されていない、もしくはユーザーが”mqm”グループにメンバーシップを持っていない場合、mqm | 適宜(すぐに反映されます) | もし、WebSphere MQのアドオンパッケージが”mqm”以外のユーザーを必要とする場合にのみ設定 |

# 7.4. NFSパラメーター一覧

下記の表は、NFSパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| RESTARTMOUNTD | NFSリストア時にrpc.mountの停止と再起動を行います。 | true: 有効<br>false: 無効 | true | 適宜 (すぐに反映されます) | |

# 7.5. Oracleパラメーター一覧

下記の表は、Oracleパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| ORACLE_ORATABLOC | デフォルトでは、/etc/oratab が使用されます。<br>代替oratabファイルのあるディレクトリを指定します。<br>通常は /etc/oratab を参照しますが、そのファイルが存在しない場合には、このパラメーターで指定されたディレクトリにある oratab を読み込みます。 | 文字列 | /var/opt/oracle | 適宜(すぐに反映されます) | |
| LK_ORA_NICE | 最大許容接続に達した時に、原因となるデータベース接続障害に対してリカバリを試みるかどうかを指定します。<br>最大許容接続に達した時のリカバリの実行は、スタンバイノードへのフェイルーバの原因となることがあります。 | 0:リカバリ実行<br>1:リカバリ回避 | 0 | 適宜(すぐに反映されます) | |
| ORACLE_RESTORE_TIMEOUT | Oracleリソース起動時のタイムアウト時間を秒単位で指定します。 | 整数値 | 300 | 適宜(すぐに反映されます) | |
| ORACLE_REMOVE_TIMEOUT | Oracleリソース停止時のタイムアウト時間を秒単位で指定します。 | 整数値 | 300 | 適宜(すぐに反映されます) | |

# 7.6. PostgreSQLパラメーター一覧

下記の表は、PostgresSQLパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| LKPGSQL_KILLPID_TIME | プロセス ID が停止した後、そのプロセスに対する再チェックを行うまでの時間を秒単位で指定します。 | 整数値 | 3 | 適宜 (すぐに反映されます) | デフォルト値より小さい場合は、デフォルト値が設定されます。 |
| LKPGSQL_CONN_RETRIES | 旧 LKPGSQLMAXCOUNT – 操作 （開始もしくは停止） を行った後、クライアント接続を試みる回数を指定します。 | 整数値 | 12 | 適宜 (すぐに反映されます) | デフォルト値より小さい場合は、デフォルト値が設定されます。 |
| LKPGSQL_ACTION_RETRIES | アクションコマンドに失敗するまで、開始と停止を試行する回数を指定します。 | 整数値 | 4 | 適宜 (すぐに反映されます) | デフォルト値より小さい場合は、デフォルト値が設定されます。 |
| LKPGSQL_STATUS_TIME | status コマンドのタイムアウト時間を秒単位で指定します。 | 整数値 | 17 + (3 * LKPGSQL_KILLPID_TIME) | 適宜 (すぐに反映されます) | デフォルト値より小さい場合は、デフォルト値が設定されます。 |
| LKPGSQL_QCKHANG_MAX | データベースインスタンスがフェイルオーバ / sendevent を発生させるまでに許容される、quickCheckがハングする回数を指定します。 | 整数値 | 2 | リソース作成時 | 1より小さい場合は、デフォルト値が設定されます。 |
| LKPGSQL_CUSTOM_DAEMON | postgres デーモン（postgres.bin,postmaster,postgres,edb-postgres） に対する追加の別名の | 文字列 | （未設定） | リソース作成時 | |

| | | | | | |
|---|---|---|---|---|---|
| | 指定を許可します。 | | | | |
| LKPGSQL_IDIRS | 旧 LKPGSQL_IPORTS – immediate オプションのみを使用してシャットダウンするインスタンスに対する datadir のエントリを含みます。 | 文字列 | （未設定） | 適宜 (すぐに反映されます) | |
| LKPGSQL_SDIRS | Smart オプションを使用してシャットダウンするインスタンスに対する datadir のエントリを含みます。 | 文字列 | （未設定） | 適宜 (すぐに反映されます) | |
| LKPGSQL_DISCONNECT_CLIENT | データベースに障害が発生している間の PostgreSQL リソース階層の振る舞いを制御します。本パラメーターを有効にすると、クライアントプロセスに SIGTERM シグナルが送信され、データベースから強制的に切断されます。この処置は、postmasterプロセスがローカルリカバリ中に稼働していない場合のみに取ることができます。 | 0: 無効<br>1: 有効 | 1 | 適宜 (すぐに反映されます) | PostgreSQL 8.2 以降では、本パラメーターをご利用いただけません。 |
| LKPGSQL_DISCONNECT_CLIENT_BYTAG | LKPGSQL_DISCONNECT_CLIENT と類似していますが、この設定は、処置をこの設定項目で指定したカンマで区切られたタグのリストに限定します。 | 文字列 | （未設定） | 適宜 (すぐに反映されます) | PostgreSQL 8.2 以降では、本パラメーターをご利用いただけません。 |
| LKPGSQL_RESUME_PROC | プロセスの停止状態の検出時　（プロセス状態が T ）、再開するか無視するかを決定します。 | 0: 無視<br>1: 再開 | 1 | 適宜 (すぐに反映されます) | |
| LKPGSQLDEBUG | PostgreSQL database kit および postgres データベースのデバッグを有効にします。0 ～ 5 の値が有効です。<br>この設定項目は、オプション -d <LKPGSQLDEBUG> を使用して postmaster データベースへと渡されます。 | 0～5 の整数値 | 0 | 適宜 (すぐに反映されます) | 設定可能な値でない場合は、デフォルト値が設定されます。 |

# 7.7. Quorumパラメーター一覧

下記の表は、Quorumパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| QUORUM_MODE | Quorum モードを指定します。 | majority tcp_remote storage none または off | majority | 適宜 (すぐに反映されます) | 詳細はテクニカルドキュメンテーション「Quorum/Witness」を参照。 |
| QUORUM_HOSTS | Quorum チェックで接続確認を行うホストを “host:port” の形式で指定します。複数のホストを指定する場合、カンマ区切りで指定してください（スペースは含めないこと）。<br><br>（設定例）<br><br>QUORUM_HOSTS=myhost:80,router1:443,router2:22 | 文字列 | （未設定） | 適宜 (すぐに反映されます) | このパラメーターは、QUORUM_MODE が *tcp_remote* の場合のみ、適用されます。 |
| WITNESS_MODE | Witness モードを指定します。 | remote_verify storage none または off | remote_verify | 適宜 (すぐに反映されます) | 詳細はテクニカルドキュメンテーション「Quorum/Witness」を参照。 |
| QUORUM_TIMEOUT_SECS | Quorum チェックで tcp/ip 接続を完了するための タイムアウト時間を秒単位で指定します。この時間内に接続が完了しない場合は、失敗、または使用不可となります | 整数値 | 20 | 適宜 (すぐに反映されます) | このパラメーターは、QUORUM_MODE が *tcp_remote* の場合のみ、適用されます。 |
| QUORUM_LOSS_ACTION | Quorum が失われた場合の動作を指定します。 | fastkill fastboot osu | fastkill | 適宜 (すぐに反映されます) | 詳細はテクニカルドキュメンテーション「Quorum/Witness」を参照。 |

| | | | | | |
|---|---|---|---|---|---|
| QWK_STORAGE_TYPE | 共有ストレージのタイプを指定します。<br>storage モードで使用する場合は、必ず指定してください。 | block<br>file<br>aws_s3 | (未設定) | qwk_storage_init 実行時 | このパラメーターは、QUORUM_MODE および WITNESS_MODE に storage を指定した場合のみ適用されます。<br>詳細はテクニカルドキュメンテーション「storage モード」を参照。 |
| QWK_STORAGE_HBEATTIME | QWK オブジェクトを読み書きする間隔を秒単位で指定します。この設定はLCMHBEATTIME のデフォルトの設定より大きいまたは同じである必要があります。 | 5以上10以下の整数値 | 6 | qwk_storage_init 実行時 | このパラメーターは、QUORUM_MODE および WITNESS_MODE に storage を指定した場合のみ適用されます。特に QWK_STORAGE_TYPE=aws_s3 の場合は、この値を7 以上の値に増やすことを推奨します。<br>詳細はテクニカルドキュメンテーション「storage モード」を参照。 |
| QWK_STORAGE_NUMHBEATS | Witness チェックで対象ノードに障害が発生していると判断するための値を指定します。QWK オブジェクトの読み込みにおいて、このパラメーターに指定した回数以上更新が停止していると、対象ノードに障害が発生していると判断します。この設定はLCMNUMHBEATS のデフォルトの設定より大きいまたは同じである必要があります。 | 3以上の整数値 | 4 | qwk_storage_init 実行時 | このパラメーターは、QUORUM_MODE および WITNESS_MODE に storage を指定した場合のみ適用されます。<br>詳細はテクニカルドキュメンテーション「storage モード」を参照。 |
| QWK_STORAGE_OBJECT_<ホスト名><br>**注意:** ホスト名に”-”ま | QWK オブジェクトのパスを指定します。<br>*storage* モードで使用する場 | 文字列(最大長は256文 | (未設定) | qwk_storage_init 実行 | このパラメーターは、QUORUM_MODE および |

| | | | | | |
|---|---|---|---|---|---|
| | 合は、必ずクラスターを構成するすべてのノード分のパラメーターを指定してください。<br><br>**[QWK_STORAGE_TYPEが block の場合]**<br>デバイスファイルのパスを指定してください。<br>**注意:** WWID (/dev/disk/by-id/) を使用して永続的なパスを指定してください。<br>（設定例）<br>QWK_STORAGE_OBJECT_nodeA=/dev/disk/by-id/xxxxx<br>QWK_STORAGE_OBJECT_nodeB=/dev/disk/by-id/yyyyy<br><br>**[QWK_STORAGE_TYPEが file の場合]**<br>レギュラーファイルのパスを指定してください。<br>（設定例）<br>QWK_STORAGE_OBJECT_nodeA=/quorum/nodeA<br>QWK_STORAGE_OBJECT_nodeB=/quorum/nodeB<br><br>**注意**: nodeA および nodeB は、ディレクトリーではなく通常のファイルである必要があります。<br><br>**[QWK_STORAGE_TYPEが aws_s3 の場合]**<br>Amazon S3 オブジェクトの s3uri を指定してください。LifeKeeper が動作しているリージョンと異なったリージョンの S3 を使用してください。また、異なった 2 つのリージョンの S3 の使用を推奨します。 2 リージョン指定する場合は、カンマ区切り（スペースは含まない）で列挙してください。<br>（設定例1） | | | | |
| たは”.”を含む場合、アンダースコア “_” に置き換えてください。 （例：lksios-1 → lksios_1）<br>**注意:** LifeKeeperで使用するホスト名は lcd uname コマンドで調べることができます。 | | 字です） | | 時 | WITNESS_MODE に storage を指定した場合のみ適用されます。<br>詳細はテクニカルドキュメンテーション「storage モード 」を参照。 |

| | | | | | |
|---|---|---|---|---|---|
| | QWK_STORAGE_OBJECT_nodeA=s3://bucket1/nodeA,s3://bucket2/nodeA<br>QWK_STORAGE_OBJECT_nodeB=s3://bucket1/nodeB,s3://bucket2/nodeB<br>（設定例2）<br>QWK_STORAGE_OBJECT_nodeA=s3://bucket/quorum/nodeA<br>QWK_STORAGE_OBJECT_nodeB=s3://bucket/quorum/nodeB | | | | |
| HTTP_PROXY<br>HTTPS_PROXY<br>NO_PROXY | サービスエンドポイントへのアクセスでHTTPプロキシを使用する場合に設定してください。<br>ここで設定した値がそのままAWS CLI へ渡されます。<br>詳しくは AWS のドキュメント を参照してください。 | 文字列 | （未設定） | `qwk_storage_init` 実行時 | このパラメーターは EC2 や Route53 と共通のパラメーターです。 |
| QUORUM_DEBUG | デバッグモードを指定します。 | 0: 無効<br>1: 有効 | 0 | 適宜 (すぐに反映されます) | 詳細はテクニカルドキュメンテーション「Quorum/Witness 」を参照。 |

# 7.8. Route53 パラメーター一覧

下記の表に Route53 Recovery Kit の動作のチューニングパラメーターとその説明を示します。チューニングを行う場合には、 両ノードの */etc/default/LifeKeeper* 設定ファイルに追加することで設定されます。 Route53 Recovery kit のコンポーネントはメモリに常駐していないため、 */etc/default/LifeKeeper* の値を変更すると即時に反映されます。LifeKeeperやOSの再起動は必要ありません。

| パラメーター名 | 意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 注記 |
|---|---|---|---|---|---|
| ROUTE53_TTL | TTL (Time To Live) の設定値を秒単位で指定します。 | 整数値 | 10 | スイッチオーバー後 | |
| ROUTE53_CHANGEID_INTERVAL | Route 53 API通信のステータス確認時のインターバルを秒単位で指定します。 | 整数値 | 20 | 適宜(すぐに反映されます) | |
| ROUTE53_CHANGEID_TRY_COUNT | Route 53 API通信のステータス確認時のトライ回数を指定します。 | 整数値 | 5 | 適宜(すぐに反映されます) | |
| AWSCLI_CONNECT_TIMEOUT | awsコマンドのオプション—cli–connect–timeout に与える値を秒単位で指定します。 | 整数値 | 10 | 適宜(すぐに反映されます) | 本パラメーターは EC2 と共通のパラメーターです。 |
| AWSCLI_READ_TIMEOUT | awsコマンドのオプション—cli–read–timeout に与える値を秒単位で指定します。 | 整数値 | 5 | 適宜(すぐに反映されます) | 本パラメーターは EC2 と共通のパラメーターです。 |

| | | | | | |
|---|---|---|---|---|---|
| HTTP_PROXY<br>HTTPS_PROXY<br>NO_PROXY | サービスエンドポイントへのアクセスで HTTP プロキシを使用する場合に設定してください。ここで設定した値がそのまま AWS CLI へ渡されます。<br>詳しくはAWSのドキュメントを参照してください。<br>docs.aws.amazon.com/ja_jp/cli/latest/userguide/cli-http-proxy.html | 文字列 | （未設定） | 適宜(すぐに反映されます) | 本パラメーターは EC2 や Quorum と共通のパラメーターです。 |

# 7.9. SAPパラメーター一覧

下記の表は、SAPパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| SAP_CONFIG_REFRESH | Configuration Propertiesページの更新頻度を秒単位で指定します。 | 整数値 | LKCHECKINTERVAL/2 | 適宜 (すぐに反映されます) | 5より小さい場合は、デフォルト値が設定されます。 |
| SAP_CREATE_NAS | NAS マウントされたファイルシステムの NAS リソースを自動的に生成するかどうかを指定します。 | 0: 無効 1: 有効 | 1 | 適宜 (すぐに反映されます) | |
| SAP_NFS_CHECK_DIRS | チェックするNFSマウントポイントのリストをカンマ区切りで指定します。 | 文字列 | 空欄 | NFS共有ファイルシステムを使用する際に必要な時 | Amazon EFSマウントポイントには使用しないでください。 |
| SAP_QUICKCHECK_TIMEOUT | quickCheck のタイムアウト時間を秒単位で指定します。 | 整数値 | 60 | 適宜 (すぐに反映されます) | |
| SAP_RESTORE_TIMEOUT | restore のタイムアウト時間を秒単位で指定します。 | 整数値 | 516 | 適宜 (すぐに反映されます) | |
| SAP_REMOVE_TIMEOUT | remove のタイムアウト時間を秒単位で指定します。 | 整数値 | 804 | 適宜 (すぐに反映されます) | |
| SAP_RECOVER_TIMEOUT | recover のタイムアウト時間を秒単位で指定します。 | 整数値 | 1320 | 適宜 (すぐに反映されます) | デフォルト値より小さい場合は、デフォルト |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | 値が設定されます。 |

# 7.10. DataKeeperパラメーター一覧

下記の表は、DataKeeperパラメーター名とその意味を説明しています。これらの値は */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| LKDR_CHUNK_SIZE | bitmapのチャンクサイズをKB単位で指定します。 | 整数値 | 256 | リソース作成時 | このパラメーターには、0からミラーサイズまでの任意の値を設定できます。値は2の累乗である必要があります。この値が2の累乗でない場合、あるいはミラーのサイズより大きい場合、ミラーの作成に失敗します。0に設定すると、デフォルトで256kになります。値が大きすぎると、1つのダーティブロックに対しても巨大なデータのチャンクが再同期されるため、ビットマップにはあまりメリットがありません。値が小さすぎる場合、ビットマップのオーバーヘッドが顕著になり、ミラーのスループットが制限されます。 |
| LKDR_CONNECT_NBD_DURING_RESTORE | ミラーの復元時（In | True または False | True | 複数の | デフォルト設定（「true」）では、ミラーが復元されると利用 |

| | | | | | |
|---|---|---|---|---|---|
| | | | | ミラーと複数のターゲットを持つクラスターでスイッチオーバーとフェールオーバーのパフォーマンスを | 可能なすべてのターゲットにデータがすぐに複製されるため、データの整合性が最も高くなります。In Serviceのパフォーマンス（「false」設定）を改善するために、最初のquickCheck 間隔（LKCHECKINTERVALで定義、デフォルトでは120秒）の間に NBD 接続を確立できます。これにより、ミラーを In Service にするのに必要な時間が短縮されますが、データがターゲットに複製されない時間も生じます。複数のミラーと複数のターゲットを使用している場合に、最もパフォーマンスが改善されます。 |
| | Service状態になった時）にNBD接続を確立する必要があるかどうかを指定します。 | | | | |

| | | | | | |
|---|---|---|---|---|---|
| | | | | 向上させたい時 | |
| LKDR_SPEED_LIMIT | 同期に使用する最大帯域幅を指定します。 | 整数値 | 500000 | リソース再同期 | 可能な最大速度で再同期が実行されるように、この値を十分高い値に設定する必要があります。 |
| LKDR_SPEED_LIMIT_MIN | 同時に他の I/O が実行されているときに許可する再同期の速度を指定します。 | 整数値 | 20000 | リソース再同期 | このパラメーターは、LKDR_SPEED_LIMIT の設定よりも小さくする必要があります。 また、同期の実行時に通常の I/O 動作が妨げられないようにするには、経験則として、この値をドライブの最大書き込みスループットの半分以下に設定する必要があります。 |
| LKDR_ASYNC_LIMIT | ターゲットへの同期を遅らせることのできる書き込み要求の数を指定します。非同期ミラーを使用してパフォーマンスを向上さ | 整数値 | 4096 | リソース作成時 | このパラメーターの有効範囲は、0〜16383です。16383を超える値に設定すると、ミラーの作成に失敗します。0に設定された場合、非同期制限はデフォルトで4096になります。1に設定された場合、デフォルトで256になります。2～16383の間の値を指定すると、非同期制限はその値に設定されます。注：4k以下の値は推奨されません。 |

| | | | | | |
|---|---|---|---|---|---|
| | せるには、この値を増やします。 | | | | |
| LKDR_NO_FULL_SYNC | 新しく追加されたターゲットの全同期を抑制します。 | 0: 抑制しない<br>1: 抑制する | 0 | 適宜(すぐに反映されます) | 詳細はテクニカルドキュメンテーション「全同期の回避」を参照。 |
| LKDR_WAIT_TO_RESYNC | ミラーの再同期前に、どの親リソースを In Service にする必要があるかを指定します。 | False, hierarchy, <resource type> | filesys | 複製の再開前に複製されたデーターの整合性を検証する時 | False は、9.5.1 以前のバージョンでの通常の動作となります。“Hierarchy” は全階層が In Service になるまで待ちます。‘filesys’ のような “<resource type>” は、そのタイプの全ての親リソースがIn Service になるまで待ちます。 |
| LKDR_WAIT_FOR_PREVIOUS_SOURCE_TIMEOUT | ビットマップをマー | -1： 前のソース | -1 | 適宜(す | ターゲットをクラスターに追加する際の完全再同期を回避するに |

| | | | | | |
|---|---|---|---|---|---|
| | ジできるように、前のソースがクラスターに参加するのを待つ時間を指定します。 | がクラスターに再参加するまで待機する<br>0： 待機しない<br>> 0： 待機する時間（秒） | | ぐに反映されます) | は、前のソースのビットマップを最初にマージする必要があります。この設定はすべての netraid デバイスに適用され、個々のデバイスに異なる値を設定することはできません。詳細はテクニカルドキュメンテーション「SIOS DataKeeper の仕組み 」を参照。 |
| NBD_NR_REQUESTS | 所定の時間処理を保留できるI/O要求の数を指定します。ミラーリングのパフォーマンスを高めるには、この値を増やします。 | 整数値 | 2048 | 適宜(すぐに反映されます) | |

# 7.11. Standby Node Health Check パラメータ一覧

Standby Node Health Check では、各機能の有効/無効や値を設定する必要があります。
これらの設定は、/etc/default/LifeKeeper 設定ファイルでカスタマイズすることが可能です。

| パラメータ名 | 説明 | 設定値 | デフォルト値 | 適応タイミング | 備考 |
|---|---|---|---|---|---|
| SNHC | Standby Node Health Check の全体の機能を有効にします。有効にする場合は 1 に設定してください。デフォルトでは無効となっています。さらに個々の機能の SNHC_XX を有効にする必要があります。 | 0 : 無効<br>1 : 有効 | 0 | lkcheck 起動時 | 設定後は lkcheck の再起動をしてください。詳細は「Standby Node Health Check 」を参照してください。 |
| SNHC_CPUCHECK | Node 監視 機能における、CPU 監視を有効にします。有効にする場合は 1 に設定してください。デフォルトでは無効となっています。 | 0 : 無効<br>1 : 有効 | 0 | 適宜 | |
| SNHC_CPUCHECK_THRESHOLD | CPU 監視において、異常と判断するCPU使用率の閾値を指定します。指定されていない場合、99 で実行されます。 | 10 以上 99 以下の整数値 | 99 | 適宜 | 設定がされていない場合 ERROR ログにて設定を促します。 |
| SNHC_CPUCHECK_TIME | CPU 監視において SNHC_CPUCHECK_THRESHOLD で設定した使用率を連続で超えた場合に異常と判断する回数を指定します。設定されていない場合、1 で実行されます。 | 1 以上 100 以下の整数値 | 1 | 適宜 | 設定がされていない場合 ERROR ログにて設定を促します。 |

| | | | | | |
|---|---|---|---|---|---|
| SNHC_MEMCHECK | Node監視 機能における、Memory 監視を有効にします。有効にする場合は 1 に設定してください。デフォルトでは無効となっています。 | 0 : 無効<br>1 : 有効 | 0 | 適宜 | |
| SNHC_MEMCHECK_THRESHOLD | Memory 監視において、異常と判断する Memory 使用率の閾値を指定します。設定されていない場合、99 で実行されます。 | 10 以上 99 以下の整数値 | 99 | 適宜 | 設定がされていない場合 ERROR ログにて設定を促します。 |
| SNHC_MEMCHECK_TIME | Memory 監視において SNHC_MEMCHECK_THRESHOLD で設定した使用率を連続で超えた場合に異常と判断する回数を指定します。設定されていない場合、1 で実行されます。 | 1 以上 100 以下の整数値 | 1 | 適宜 | 設定がされていない場合 ERROR ログにて設定を促します。 |
| SNHC_IPCHECK | IP リソースの OSU リソース監視 を有効にします。デフォルトでは無効となっています。 | 0 : 無効<br>1 : 有効 | 0 | 適宜 | |
| SNHC_IPCHECK_SLEEPTIME | IP リソースの OSU リソース監視で対象となる NIC が down している場合、NIC を起動してからリンクの確認を実行するまでの待機時間を設定します。NIC が down していた場合、監視処理後に再度 NIC を down します。 | 待機時間（秒） | 1 | 適宜 | |
| SNHC_DISKCHECK | Disk または DMMP リソースの OSU リソース監視 を有効にします。デフォルトでは無効となっています。 | 0 : 無効<br>1 : 有効 | 0 | 適宜 | |
| SNHC_DISKCHECK_<Tag名> | 指定した Tag 名の Disk または DMMP リソースのみの OSU リソース監視 を有効にします。デフォルトでは無効となっています。 | 0 : 無効<br>1 : 有効 | 0 | 適宜 | |
| SNHC_NASCHECK | NAS リソースの OSU リソース監視 を有効にします。 | 0 : 無効<br>1 : 有効 | 0 | 適宜 | |

| | | | | | |
|---|---|---|---|---|---|
| SNHC_NASCHECK_<Tag名> | 指定された Tag 名の NAS リソースのみ OSU リソース監視を有効にします。 | 0：無効<br>1：有効 | 0 | 適宜 | Tag 名にハイフンが入っている場合はアンダーバーに置き換えて設定してください。 |

# 7.12. SAP HANA パラメーターリスト

次の表には、SAP HANAパラメーターの名称と意味を記載しています。これらの値は、 /etc/default/LifeKeeper 構成ファイルを編集することによってチューニングできます。

| パラメーター名 | 意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 注記 |
|---|---|---|---|---|---|
| HANA_HANDSHAKE_TAKEOVER_FAILBACK | Takeover with Handshakeに失敗した時に SAP HANA リソースの自動フェイルバックを実行するかどうかを指定します。‘true’が設定されるとLifeKeeper は Takeover with Handshake に失敗した場合に旧ホストで SAP HANA リソースを In Service にもどします。‘false’ (デフォルト)が設定されると、Takeover with Handshake が失敗した後Out of Service Failed (OSF) のステータスのままになり、手動でIn Service にする必要があります。 | true もしくはfalse | false | 適宜(すぐに反映されます) | |
| HANA_HSR_POLL_INTERVAL | LifeKeeper が SAP HANA システムレプリケーションのステータスをポーリングする頻度（秒）を指定します。 | 正の整数 | 10 | 適宜(すぐに反映されます) | |
| HANA_QUICKCHECK_TIMEOUT | SAP HANA quickCheck スクリプトのタイムアウト値（秒）を指定します。 | 正の整数、最低値 30 | LKCHECKINTERVAL – 10 | 必要に応じて | 30 未満に設定すると、自動的にデフォルトで 30 になります。 |
| HANA_REGISTER_SECONDARY_DURING_RESTORE | プライマリーデータベースインスタンスに対する In Service 処理が完了する前に SAP HANA Recovery Kit がセカンダリーデータベースを登録し、再起動するかどうかを指定します。 ‘true’が設定されるとSAP HANA リソースは、SAP HANA Recovery Kit がセカンダリーデータベースを登 | true または false | false | 適宜(すぐに反映されま | |

| | | | | | |
|---|---|---|---|---|---|
| | 録し、再起動するまで In Service になりません。 'false' (デフォルト)が設定されるとSAP HANA リソースは、プライマリーデータベースインスタンスが起動するとすぐに In Service になり、セカンダリーデータベースの登録、再起動のため、ローカルリカバリーイベントがその後すぐに開始されます。 | | | す) | |
| HANA_RECOVER_TIMEOUT | SAP HANA リカバリースクリプトのタイムアウト値（秒）を指定します。 | 正の整数 | 1800 | 適宜(すぐに反映されます) | |
| HANA_START_WAIT | SAP HANA データベースインスタンスを停止するときに待機する時間（秒）を指定します。 | 正の整数 | 2700 | 適宜(すぐに反映されます) | |
| HANA_STOP_WAIT | SAP HANA データベースインスタンスを停止するときに待機する時間（秒）を指定します。 | 正の整数 | 600 | 適宜(すぐに反映されます) | |
| HANA_STOP_COUNT | SAP HANA データベースインスタンスを停止するときに sapcontrol StopWait コマンドを発行する試行回数を指定します。 | 正の整数 | 3 | 適宜(すぐに反映されます) | |
| HANA_STOP_FINAL_WAIT | HANA_STOP_COUNT が SAP HANA データベースの停止に失敗した後、データベースが完全に停止するまで待機する追加の時間（秒）を指定します。 | 正の整数 | 60 | 適宜(すぐに反映されます) | |

# 7.13. SAP MaxDB パラメーター一覧

以下の表は、SAP MaxDB リカバリーキットのパラメーターの名前と意味を説明しています。これらの値は、 */etc/default/LifeKeeper* 設定ファイルを編集することにより設定可能です。

| パラメーター名 | パラメーターの意味 | 設定値 | デフォルト値 | パラメーター適用タイミング | 備考 |
|---|---|---|---|---|---|
| MAXDB_START_TIMEOUT | ハングしているIn Serviceのアクションを中止するまで待機する秒数 | 整数 | 300 | 適宜 (すぐに反映されます) | |
| MAXDB_PID_CLEANUP | ハングしたIn Serviceのアクションの子プロセスをクリーンアップ（強制終了）するかどうかを指定します | y：クリーンアップする<br>n：クリーンアップしない | n | 適宜 (すぐに反映されます) | |
| MAXDB_STOP_COUNT | LifeKeeperがデータベースのオフラインを試行する回数 | 整数 | 5 | 適宜 (すぐに反映されます) | |
| MAXDB_WAIT | データベースのオフライン試行間の待機秒数 | 整数 | 5 | 適宜 (すぐに反映されます) | |
| MAXDB_DEBUG | デバッグロギング | 0: | 0 | 適宜 (すぐに反映されます) | |

| | | | | | |
|---|---|---|---|---|---|
| | を有効または無効にします | 無効化 1: 有効化 | | | |
| MAXDB_OFFLINE_ENABLED | LifeKeeper MaxDBリソースをOut of Serviceにしたときにデータベースをオフラインにするかどうかを指定します | 0: オフラインにしない 1: オフラインにする | 1 | データベースを停止せずにLifeKeeperを一時的に停止する必要がある場合（例：アップグレードやメンテナンスの実施など）に、注意して使用してください | メンテナンスが完了したら、データベースをオフラインに戻してください |

# 8. エラーコード一覧

総合メッセージカタログ をご参照ください。

# 8.1. 総合メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 000200 | ERROR | pam_start() failed | |
| 000201 | ERROR | pam_authenticate failed (user %s, retval %d) | |
| 000202 | ERROR | pam_end() failed?!?! | |
| 000203 | ERROR | Did not find expected group ‘lkguest’ | |
| 000204 | ERROR | Did not find expected group ‘lkoper’ | |
| 000205 | ERROR | Did not find expected group ‘lkadmin’ | |
| 000208 | ERROR | pam_setcred establish credentials failed (user %s, retval %d) | **原因**: ユーザー {user} の有効なログイン資格情報を確立できません。pam_setcred 呼び出しで返された値は次の通りです：{retval}<br><br>**処置**: 詳細については、/var/log/security および /var/log/messages を確認して下さい。 |
| 000209 | ERROR | pam_setcred delete credentials failed (user %s, retval %d) | **原因**: ユーザー {user} の有効なログイン資格情報を確立できません。pam_setcred 呼び出しで返された値は次の通りです：{retval}<br><br>**処置**: 詳細については、/var/log/security および /var/log/messages を確認して下さい。 |
| 000902 | ERROR | Error removing system name from loopback address line in /etc/hosts file. You must do this manually before starting the GUI server. | **原因**: システム名が /etc/hosts ファイルから削除されていません。<br><br>**処置**: システム名を手動で削除し、GUI サーバを再起動して、次のコマンドを入力します: run <action name> |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 000918 | ERROR | LifeKeeper GUI Server error during Startup | **原因**: 異常な状態により、GUI サーバが終了しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 001052 | FATAL | Template resource "%s" on server "%s" does not exist | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 001053 | ERROR | Cannot access canextend script "%s" on server "%s" | **原因**: {server} でスクリプト CANEXTEND が見つからなかったので、LifeKeeper は拡張前のチェックを実行できませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 001054 | ERROR | Cannot extend resource "%s" to server "%s" | **原因**: LifeKeeper は、{server} でリソース {resource} を拡張できませんでした。 |
| 001055 | ERROR | Cannot access extend script "%s" on server "%s" | **原因**: LifeKeeper は、{server} でスクリプト EXTEND が見つからなかったので、リソース階層を拡張できませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 001057 | ERROR | Cannot extend resource "%s" to server "%s" | **原因**: LifeKeeper は、{server} でリソース {resource} を拡張できませんでした。 |
| 001059 | ERROR | Resource with tag "%s" already exists | **原因**: リソースで指定した名前はすでに使用中です。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: リソースに別の名前を選択するか、既存のリソースを使用してください。 |
| 001060 | ERROR | Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s" | **原因**: リソースで指定した名前または ID はすでに使用中です。<br><br>**処置**: リソースに別の名前または ID を選択するか、既存のリソースを使用してください。 |
| 001061 | ERROR | Error creating resource "%s" on server "%s" | **原因**: リソースの作成中に予期しない障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 001081 | WARN | IP address \"$ip\" is neither v4 nor v6 | **原因**: 指定した IP アドレスは IPv4 アドレスでも IPv6 アドレスでもありません。<br><br>**処置**: 指定した名前またはアドレスを確認し、再試行してください。名前を指定している場合は、名前の解決が有効な IP アドレスを返していることを確認してください。 |
| 004024 | ERROR | | **原因**: LCD はリソースのリカバリ中にリソース ID {id} のリソース情報のフェッチに失敗しました。<br><br>**処置**: 入力リソース ID を確認し、リカバリ操作を再試行してください。 |
| 004028 | ERROR | %s occurred to resource \"%s\" | **原因**: リソース {resource} のローカルリカバリが失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004055 | ERROR | attempt to remote-remove resource \"%s\" that can’t be found | **原因**: タグ名 {tag} によるリソースの検索中に、リモートによるサービスからのリソース削除に失敗しました。<br><br>**処置**: 入力タグ名を確認し、リカバリ操作を再試行してください。 |
| 004056 | ERROR | attempt to remote-remove resource \"%s\" that is not a shared resource | **原因**: タグ名 {tag} が共有リソースでない場合に、リモートによるサービスからのリソース削除に失敗しました。<br><br>**処置**: 入力タグ名を確認し、リカバリ操作を再試行してください。 |
| 004060 | ERROR | attempt to transfer-restore resource \"%s\" that can’t be found | **原因**: タグ名 {tag} を指定したサービスリソースのリモート転送に失敗しました。<br><br>**処置**: 入力タグ名を確認し、リカバリ操作を再試行してください。 |
| 004061 | ERROR | attempt to transfer-restore resource \"%s\" that is not a shared resource with machine \"%s\" | **原因**: LifeKeeper は、リモート {machine} からサービスのリソースをリモート転送する際に、{tag} 名による共有リソースの検出に失敗しました。<br><br>**処置**: 入力タグ名を確認し、リカバリ操作を再試行してください。 |
| 004089 | ERROR | ERROR: Parallel recovery initialization failed.\n | **原因**: 並行リカバリは、階層内にあるリソースのリストの初期化に失敗しました。<br><br>**処置**: 関連するエラーをログで確認 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | し、レポートされている問題を解決してください。 |
| 004091 | ERROR | ERROR: fork failed. continuing to next resource\n | **原因**: パラレルリカバリーで、単一のリソースをリストアしようとする新しいプロセスを分岐できませんでした。 |
| 004093 | ERROR | ERROR: reserve failed. continuing to next resource\n | **原因**:並行リカバリは、集合階層から単一のリソースのリザーブに失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004096 | ERROR | ERROR: clone %d is hung, attempting to kill it\n | **原因**: リソースリカバリの単一のサブプロセスが、リソース階層全体の並行リカバリの際にハングしました。<br><br>**処置**: ハングしているサブプロセスが自動的に停止します。 |
| 004097 | ERROR | ERROR: Could not kill clone %d\n | **原因**: ハングしているサブプロセスの停止に失敗しました。 |
| 004116 | ERROR | %s | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、中間フォルダを作成しようとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、中間フォルダが作成されていない理由を判別してください。 |
| 004117 | ERROR | open(%s) | **原因**: メモリ内データオブジェクトのオ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ンディスクバージョンを書き込む際に、一時ファイルを開こうとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルを開くことができない理由を判別してください。 |
| 004118 | ERROR | write(%s) | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、一時ファイルに書き込もうして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルの書き込みに失敗した理由を判別してください。 |
| 004119 | ERROR | fsync(%s) | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、一時ファイルに fsync を実行しようとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、「fsync」が失敗した理由を特定してください。 |
| 004120 | ERROR | close(%s) | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、一時ファイルを閉じようとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルのクローズに失敗した理由を判別してください。 |
| 004121 | ERROR | rename(%s, %s) | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、一時ファイル {file} の名前を元のフ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ファイル {file} に変更しようとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルの名前変更に失敗した理由を判別してください。 |
| 004122 | ERROR | open(%s) | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、中間ディレクトリ {directory} を開こうとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ディレクトリのオープンに失敗した理由を判別してください。 |
| 004123 | ERROR | fsync(%s) | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、中間ディレクトリ {directory} に fsync を実行しようとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ディレクトリの「fsync」に失敗した理由を特定してください。 |
| 004124 | ERROR | close(%s) | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際に、中間ディレクトリ {directory} を閉じようとして失敗しました。これはシステムエラーです。<br><br>**処置**:ログで詳細なエラー情報を確認し、ディレクトリのクローズに失敗した理由を判別してください。 |
| 004125 | ERROR | wrote only %d bytes of requested %d\n | **原因**: メモリ内データオブジェクトのオンディスクバージョンを書き込む際 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | に、書き込まれるデータの最終的なサイズ {size} バイトが要求バイト数 {number} を下回っているため、書き込み操作が失敗しました。<br><br>**処置**: ログで関連エラー情報を確認し、データの書き込みに失敗した理由を判別してください。 |
| 004126 | ERROR | open(%s) | **原因**: データオブジェクトのオンディスクバージョンをバッファに読み取っている間に、データファイルを開こうとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルのオープンに失敗した理由を判別してください。 |
| 004127 | ERROR | open(%s) | **原因**: データオブジェクトのオンディスクバージョンをバッファに読み取っている間に、一時データファイルを開こうとして失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルのオープンに失敗した理由を判別してください。 |
| 004128 | ERROR | read(%s) | **原因**: データオブジェクトのオンディスクバージョンをバッファにロードしている間に、データファイルを読み取ろうとして失敗しました。これはシステムエラーです。<br><br>**処置**:ログで詳細なエラー情報を確認し、ファイルの読み取りに失敗した理由を判別してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004129 | ERROR | read buffer overflow (MAX=%d)\n | **原因**: データオブジェクトのオンディスクバージョンをバッファに読み取ろうとしている間に、読み取りバッファの制限 {max} に達しました。<br><br>**処置**: LifeKeeper 設定を確認し、LifeKeeper を再起動してください。 |
| 004130 | ERROR | close(%s) | **原因**: データオブジェクトのオンディスクバージョンをバッファに読み取っている間に、データファイルのクローズに失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルのクローズに失敗した理由を判別してください。 |
| 004131 | ERROR | rename(%s, %s) | **原因**: データオブジェクトのオンディスクバージョンをバッファに読み取っている間に、一時データファイルの名前変更に失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ファイルの名前変更に失敗した理由を判別してください。 |
| 004132 | ERROR | Can’t open %s : %s | **原因**: アプリケーションとリソースタイプ情報のオンディスクバージョンをバッファに読み取っている間にエラー {error} でディレクトリ {directory} のオープンに失敗しました。これはシステムエラーです。<br><br>**処置**: ログで詳細なエラー情報を確認し、ディレクトリのオープンに失敗した理由を判別してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004133 | ERROR | path argument may not be NULL | **原因**: 入力ソースパスが見つからないので、ファイルコピー中にコマンド「lcdrcp」が失敗しました。<br><br>**処置**: 入力ソースパスを確認し、「lcdrcp」を再試行してください。 |
| 004134 | ERROR | destination path argument may not be NULL | **原因**:入力ディスティネーションパスが見つからないので、ファイルコピー中にコマンド「lcdrcp」が失敗しました。<br><br>**処置**: 入力ディスティネーションパスを確認し、「lcdrcp」を再試行してください。 |
| 004135 | ERROR | destination path can’t be zero length string | **原因**:「lcdrcp」によるファイルコピー中に、入力ディスティネーションパスが空でした。<br><br>**処置**: 入力ディスティネーションパスを確認し、「lcdrcp」を再試行してください。 |
| 004136 | ERROR | open(%s) | **原因**:「lcdrcp」によるファイルコピー中に、ソースファイルパスのオープンに失敗しました。これはシステムエラーです。<br><br>**処置**: 入力ソースパスがあるかどうか、および使用可能かどうかを確認し、「lcdrcp」を再試行します。また、関連ログで詳細なエラー情報を確認してください。 |
| 004137 | ERROR | fstat(%s) | **原因**:「lcdrcp」によるファイルコピー中に、「fstat」を使用したファイル属性のフェッチに失敗しました。これはシステムエラーです。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**:ログで詳細なエラー情報を確認してください。 |
| 004138 | ERROR | file \"%s\" is not an ordinary file (mode=0%o) | **原因**:「lcdrcp」によるファイルコピー中に、通常のものではないファイルとしてソースファイルが検出されました。<br><br>**処置**: 入力ソースファイルパスを確認し、「lcdrcp」を再試行してください。 |
| 004151 | FATAL | lcdMalloc failure | **原因**: 共有メモリに、要求されたサイズのメモリを割り当てることができませんでした。<br><br>**処置**: 致命的なエラーが生成されます。 |
| 004152 | ERROR | having \"%s\" depend on \"%s\" would produce a loop | **原因**: 要求された依存関係を追加すると、依存関係がループします。<br><br>**処置**: 要求された依存関係を修正し、依存関係の作成を再試行してください。 |
| 004164 | ERROR | Priority mismatch between resources %s and %s. Dependency creation failed. | **原因**: {resource1} と {resource2} のプロパティが一致しません。<br><br>**処置**: リソースプロパティは一致している必要があります。どちらか一方または両方のプロパティを同じ値に変更し、依存関係の作成を再試行してください。 |
| 004176 | ERROR | %s | **原因**: コマンド「doabort」がコアファイルを書き込むための {directory} の作成に失敗しました。これはシステムエラーです。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004182 | ERROR | received signal %d\n | **原因**: 信号 {signum} を受信しました。 |
| 004186 | ERROR | %s: ::receive(%d) protocol error on incoming_mailbox %s | **原因**: {function} で、タイムアウト時間（{timeout}秒）以内にメッセージを受信しようとしましたが、受信メールボックス {mailbox} が非アイドル状態のため受信できませんでした。<br><br>**処置**: クラスター内の接続の状態を確認し、プロセスを再試行してください。 |
| 004190 | ERROR | %s: ::receive(%d) did not receive message within %d seconds on incoming_mailbox %s | **原因**: 関数 {function} で、受信メールボックス {mailbox} でタイムアウト {timeout} 秒以内にメッセージを受信しようとしたところ、失敗しました。<br><br>**処置**: クラスタ内の接続の状態を確認し、プロセスを再試行してください。 |
| 004204 | ERROR | attempt to send illegal message | **原因**: 不正なメッセージを送信しようとしたため、メッセージの送信に失敗しました。 |
| 004205 | ERROR | destination system \"%s\" is unknown | **原因**: 不明なディスティネーションシステム名 {system} のため、メッセージの送信に失敗しました。<br><br>**処置**: システムの設定と状態を確認し、関連エラーをログで確認してください。システムが完全に初期化されたら、同じプロセスを再試行してください。 |
| 004206 | ERROR | destination mailbox \"%s\" at system \"%s\" is unknown | **原因**: 不明なディスティネーションシステム名 {system} の不明なメールボックス {mailbox} のため、メッセージの送信に失敗しました。このエラーは、LCD が完全に初期化される前にメッセージを |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 送信することで発生します。<br><br>**処置**:システムの設定と状態を確認し、関連エラーをログで確認してください。システムが完全に初期化されたら、同じプロセスを再試行してください。 |
| 004208 | ERROR | destination system \"%s\" is alive but the \"%s\" mailbox process is not listening. | **原因**: メッセージの送信に失敗しました。ディスティネーションシステム {system} へのネットワーク接続は稼働中ですが、ディスティネーションメールボックスとの通信は失われました。<br><br>**処置**: システムの設定と状態を確認し、関連エラーをログで確認してください。システムが完全に初期化されたら、同じプロセスを再試行してください。 |
| 004209 | ERROR | destination system \"%s\" is dead. | **原因**: ディスティネーションシステム {system} との接続が失われたため、メッセージの送信に失敗しました。<br><br>**処置**: システムの設定と状態を確認し、関連エラーをログで確認してください。システムが完全に初期化されたら、同じプロセスを再試行してください。 |
| 004211 | ERROR | can’t send to destination \"%s\" error=%d | **原因**: 内部エラー {error} のため、ディスティネーションシステム {system} へのメッセージ送信に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004217 | ERROR | destination system \"%s\" is out of service. | **原因**: ディスティネーションシステム {sy |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | stem} との接続が失われたため、メッセージの送信に失敗しました。<br><br>**処置**: システムの設定と状態を確認し、関連エラーをログで確認してください。システムが完全に初期化されたら、同じプロセスを再試行してください。 |
| 004221 | ERROR | destination system \"%s\" went out of service. | **原因**: ディスティネーションシステム {system} との接続が失われたため、メッセージの送信に失敗しました。<br><br>**処置**: システムの設定と状態を確認し、関連エラーをログで確認してください。システムが完全に初期化されたら、同じプロセスを再試行してください。 |
| 004228 | ERROR | Can’t get host name from getaddrinfo() | **原因**:「getaddrinfo()」を使用してホスト名を取得しているときに障害が発生したため、ネットワークオブジェクトの作成に失敗しました。<br><br>**処置**: システムの設定と状態を確認し、同じプロセスを再実行してください。 |
| 004234 | ERROR | IP address pair %s already in use | **原因**: TCP コミュニケーションパスで IP アドレスペア {pair} がすでに使用中のため、ネットワークオブジェクトの作成に失敗しました。<br><br>**処置**: 入力 IP アドレスペアを確認し、ネットワークの作成を再実行してください。 |
| 004258 | WARN | Communication to %s by %s FAILED | **原因**: コミュニケーションパス {path} によるシステム {system} への通信に失敗 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | しました。<br><br>**処置**: システム設定とネットワーク接続を確認してください。 |
| 004261 | WARN | COMMUNICATIONS failover from system \"%s\" will be started. | **原因**: すべてのコミュニケーションパスがダウンしたため、システム {system} からのフェイルオーバが開始されます。<br><br>**処置**: システム設定とネットワーク接続の状態を確認してください。フェイルオーバが完了したら、システムの状態を確認してください。 |
| 004292 | ERROR | resource \"%s\" %s | **原因**: 現在の状態が不明なため、リソースを in service にできませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004293 | ERROR | resource \"%s\" %s | **原因**: 現在の状態により許可されていないため、リソースを in service にできませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004294 | ERROR | resource \"%s\" requires a license (for Kit %s/%s) but none is installed | **原因**: リソースの関連リカバリキットにはライセンスが必要です。<br><br>**処置**: リソースを in service にするサーバに、リカバリキットのライセンスをインストールしてください。 |
| 004297 | ERROR | secondary remote resource \"%s\" on machine \"%s\" is already in-service, so resource \"%s\" on machine \"%s\" can’t be br | **原因**:セカンダリリモートリソース {resource} がマシン {machine} ですでに in se |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | ought in-service. | rvice になっているため、リソース {resource} をマシン {machine} で in service にできませんでした。<br><br>**処置**: リモートリソースを手動で out of service に変更し、ローカルリソースで再び in service にしてください |
| 004298 | ERROR | remote resource \"%s\" on machine \"%s\" is still in-service, restore of resource \"%s\" will not be attempted!\n | |
| 004300 | ERROR | restore of resource \"%s\" has failed | **原因**: リソースを in service にできませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004311 | ERROR | can’t perform \"remove\" action on resources in state \"%s\" | **原因**: 現在の状態が {state} のため、リソースを out of service にできませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004313 | ERROR | remove of resource \"%s\" has failed | **原因**: リソース {resource} をサービス休止にできませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004318 | ERROR | %s,priv_globact(%d,%s): script %s FAILED returning %d | **原因**: 指定したエラーコードでグローバルアクションスクリプトが失敗しました。<br><br>**処置**: 関連するエラーをログで確認 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | し、レポートされている問題を解決してください。 |
| 004332 | ERROR | action \"%s\" has failed on resource \"%s\" | **原因**: リソースアクションが失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004351 | ERROR | a \"%s\" equivalency must have one remote resource | **原因**:同じシステムに 2 つの入力タグ名が存在しているため、イクイバレンシ {eqvtype} の作成に失敗しました。<br><br>**処置**: 入力リソースタグ名を修正し、同じプロセスを再実行してください。 |
| 004356 | WARN | Use unsupported option %s for remove. This option may be removed in future upgrades. | |
| 004376 | FATAL | wait period of %u seconds for LCM to become available has been exceeded (lock file \"%s\" not removed) | **原因**: 妥当な時間内に LCM デーモンが使用可能になりませんでした。LCD は LCM なしでは操作できません。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004386 | ERROR | initlcdMalloc;shmget | **原因**: 共有メモリセグメントを初期化できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。製品ドキュメントを参照し、サーバが最低要件を満たしていること、およびオペレーティングシステムが適切に設定されていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004439 | WARN | intermachine recovery skipped for %s. Failed to obtain resource_state_change lock.\n | |
| 004444 | WARN | License key (for Kit %s/%s) has EXPIRED | **原因**: ライセンスが期限切れです。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 004445 | WARN | License key (for Kit %s/%s) will expire at midnight in %ld days | **原因**: ライセンスが間もなく期限切れです。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 004466 | ERROR | system \"%s\" not defined on machine \"%s\". | **原因**: 指定したシステム名が不明です。<br><br>**処置**: システム名を確認し、操作を再試行してください。 |
| 004467 | ERROR | system \"%s\" unknown on machine \"%s\" | **原因**: 指定したシステム名が認識されません。<br><br>**処置**: システム名を確認し、操作を再試行してください。 |
| 004494 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004495 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください。 |
| 004496 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004497 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004498 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004499 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004500 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004501 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004502 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004503 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004504 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004505 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004506 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004507 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004508 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004509 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004510 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004511 | ERROR | COMMAND OUTPUT: %s | **原因**: アクションまたはイベントスクリプトが予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004512 | ERROR | %s | **原因**: リモートマシンでエラーが発生しました。<br><br>**処置**: リモートマシンのログを確認し、詳細を確認してください。 |
| 004565 | ERROR | can’t set resource state type to ILLSTATE | **原因**: リソースを不正な状態に配置しようとしました。<br><br>**処置**: リソースを不正な状態で配置しないでください。 |
| 004567 | ERROR | split brain detected while setting resource \"%s\" to \"%s\" state (SHARED equivalency to resource \"%s\" on machine \"%s\" which is in state \"%s\"). Setting local resource ISP but aborting the operation. | **原因**: マシン {machine} の SHARED イクイバレンシの プライマリリソース {resource} が {state} という状態にあるため、リソース {resource} を {state} という状態に変更できませんでした。<br><br>**処置**: スプリットブレインが発生しました。フェールオーバー操作は中止されました。スプリットブレインリソース（複数のシステム上の ISP）を手動で適切な状態にする必要があります。 |
| 004575 | ERROR | COMMAND OUTPUT: %s | |
| 004607 | ERROR | no resource instance has tag \"%s\" | **原因**: 指定したタグのリソースが存在しません。<br><br>**処置**: 有効なタグを指定するか、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004608 | ERROR | no resource instance has identifier \"%s\" | **原因**: 指定した識別子のリソースが存在しません。<br><br>**処置**: 有効な識別子を指定するか、関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004619 | ERROR | resource with tag \"%s\" already exists with identifier \"%s\" | **原因**: 指定したタグ名はすでに存在します。<br><br>**処置**: 別のタグ名を選択してください。 |
| 004620 | ERROR | resource with identifier \"%s\" already exists with tag \"%s\" | **原因**: 指定した識別子はすでに存在します。<br><br>**処置**: このリソースで使用する別の識別子を選択してください。 |
| 004643 | ERROR | Instance tag name is too long. It must be shorter than %d characters. | **原因**: タグ名が長すぎます。<br><br>**処置**: 256 文字未満のタグ名を指定してください。 |
| 004646 | ERROR | Tag name contains illegal characters | **原因**: タグ名に不正な文字が含まれています。<br><br>**処置**: こちら を参照してください。 |
| 004691 | ERROR | can’t set both tag and identifier at same time | **原因**: タグと識別子が両方指定されました。<br><br>**処置**: タグまたは識別子の一方のみを指定してください。 |
| 004745 | ERROR | failed to access lkexterrlog path=%s | **原因**: システム情報を収集するユーティリティ「lkexterrlog」にアクセスできません。<br><br>**処置**: パッケージ「steeleye-lk」がインストールされていること、およびユーティリティ「lkexterrlog」がアクセス可能であることを確認してください。 |
| 004746 | ERROR | lkexterrlog failed runret=%d cmdline=%s | **原因**: システム情報の収集時に、ユーテ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ィリティ「lkexterrlog」が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004782 | ERROR | Resource \"%s\" was in state \"%s\" before event occurred – recovery will not be attempted | **原因**: リソースはすでに in service です。リカバリは実行されません。 |
| 004783 | ERROR | Resource \"%s\" was already in state \"%s\" before event occurred | **原因**: リソースはリカバリを許可するために適切な状態にありませんでした。<br><br>**処置**: リカバリが必要な場合は、リソースを ISP 状態にしてください。 |
| 004786 | ERROR | %s on failing resource \"%s\" | **原因**: リソースを復旧しようとしているときにエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004788 | EMERG | failed to remove resource ‘%s’. SYSTEM HALTED. | **原因**: エラーが発生し、リカバリ中にリソースをサービス休止にできませんでした。リソースが 2 つのシステムでアクティブにならないように、システムが再起動されました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004793 | ERROR | lcdsendremote transfer resource \"%s\" to \"%s\" on machine \"%s\" failed (rt=%d) | **原因**: リソースとその依存関係を別のシステムへ転送中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。関連エラーがないかどう |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | か、もう一方のシステムのログを確認してください。 |
| 004797 | ERROR | Restore of SHARED resource \"%s\" has failed | **原因**: リソースのリストア中にエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 004806 | ERROR | Restore in parallel of resource \"%s\" has failed; will re-try serially | **原因**: 並行リカバリが失敗しました。関連するエラーをログで確認し、レポートされている問題を解決してください。<br><br>**処置**: 処置は必要ありません。システムはシリアル形式でリカバリを続行します。リカバリが失敗した場合は、復旧できなかったリソースに関連するエラーメッセージを確認し、対応策を探してください |
| 004819 | ERROR | read_temporal_recovery_log(): failed to fopen file: %s. fopen() %s. | **原因**: メモリにロードするために一時リカバリログファイル {file} を開こうとしたところ、エラー {error} で失敗しました。<br><br>**処置**: システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004820 | ERROR | read_temporal_recovery_log(): failed to malloc initial buf for temporal_recovery_stamp. | **原因**: ログ情報を格納するためのメモリを取得しようとしている際に、一時リカバリログ情報のメモリへのロードに失敗しました。<br><br>**処置**: システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004821 | ERROR | read_temporal_recovery_log(): failed to reallocate buffer for temporal_recovery_stamp. | **原因**: ログ情報の格納に必要なメモリサイズを増やそうとしている際に、一時リカバリログ情報のメモリへのロードに失敗しました。<br><br>**処置**: システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004822 | ERROR | write_temporal_recovery_log(): failed to open file: %s. | **原因**: 一時ファイル {temporary name} のオープンに失敗したため、一時リカバリログファイルの更新が終了しました。<br><br>**処置**: システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004823 | ERROR | rename(%s, %s) failed. | **原因**: 一時ファイル {temporary name} を実際のログファイル {real name} に名称変更できなかったため、一時リカバリログファイルの更新が終了しました。<br><br>**処置**: システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004829 | FATAL | err=%s line=%d Semid=%d numops=%zd perror=%s | **原因**: セマフォ ID {semaphore} の変更がエラー {err} およびエラーメッセージの説明 {perror} で失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。また、システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004860 | ERROR | restore ftok failed for resource %s with path %s | **原因**: パス {path} を使用するリソース {tag} のセマフォ操作で使用する IPC キーの生成に失敗しました。これはシステム |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | エラーです。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。また、システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004861 | ERROR | semget failed with error %d | **原因**: インスタンスファイルに関連付けられたセマフォ識別の取得に失敗しました。これはシステムエラーです。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。また、システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004862 | ERROR | semctl SEMSET failed with error %d | **原因**: リカバリプロセス中に使用されるセマフォの作成と初期化がエラー {error number} で失敗しました。これはシステムエラーです。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。また、システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004863 | ERROR | semop failed with error %d | **原因**: リカバリプロセス中に使用されるセマフォの設定がエラー {error number} で失敗しました。これはシステムエラーです。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。また、システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004864 | ERROR | semctl SEMSET failed with error %d | **原因**: リカバリプロセス中に使用されるセマフォの解放がエラー {error number} で失敗しました。これはシステムエラーです。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。また、システムログファイルを確認し、レポートされているエラーを修正してから、操作を再試行してください。 |
| 004865 | ERROR | restore action failed for resource %s (exit: %d) | **原因**: リソース {tag} を In Service にする操作が失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。レポートされているエラーを修正し、操作を再試行してください。 |
| 004872 | ERROR | Remote remove of resource \"%s\" on machine \"%s\" failed (rt=%d) | **原因**: ローカルシステムに転送するために {server} でリソース {tag} をサービス休止にする要求が失敗しました。<br><br>**処置**: ローカルシステムで、隣接するログメッセージから詳細を確認してください。また、リソースを削除するには、{server} のログメッセージで障害に関する詳細を確認してください。 |
| 004875 | ERROR | remote remove of resource \"%s\" on machine \"%s\" failed | **原因**: ローカルシステムに転送するために {server} でリソース {tag} をサービス休止にする要求が失敗しました。<br><br>**処置**: ローカルシステムで、隣接するログメッセージから詳細を確認してください。また、リソースを削除するには、{server} のログメッセージで障害に関する詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 004876 | ERROR | remote remove of resource \"%s\" on machine \"%s\" failed | **原因**: ローカルシステムに転送するために {server} でリソース {tag} をサービス休止にする要求が失敗しました。<br><br>**処置**: ローカルシステムで、隣接するログメッセージから詳細を確認してください。また、リソースを削除するには、{server} のログメッセージで障害に関する詳細を確認してください。 |
| 005045 | ERROR | tli_fdget_i::execute unable to establish a listener port | **原因**: ネットワーク接続を適切に設定できませんでした。<br><br>**処置**: すべてのネットワークハードウェアおよびドライバが適切に設定されていることを確認してください。それでもこのメッセージが表示され、リソースを in service にできない場合は、サポートに問い合わせてください。 |
| 005055 | ERROR | tli_fdget_o::execute – async connect failure | **原因**: ネットワーク接続を適切に設定できませんでした。<br><br>**処置**: すべてのネットワークハードウェアおよびドライバが適切に設定されていることを確認してください。それでもこのメッセージが表示され、リソースを in service にできない場合は、サポートに問い合わせてください。 |
| 005061 | ERROR | tli_fdget_o::execute – bind socket | **原因**: ネットワーク接続を適切に設定できませんでした。<br><br>**処置**: すべてのネットワークハードウェアおよびドライバが適切に設定されていることを確認してください。それでもこのメッセージが表示され、リソースを in service にできない場合は、サポートに問い合わせてください。 |
| 005090 | WARN | system_driver::add_driver: cmd=%s\n | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 005108 | WARN | system_driver::rm_driver: cmd=%s\n | |
| 005145 | ERROR | opening the file | **原因**: パイプを開く、または作成することができませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 005164 | ERROR | tli_handler::handle-error:sending/receiving data message | **原因**: メッセージの送信または受信に失敗しました。<br><br>**処置**: 詳細については、隣接するログメッセージを確認してください。これは一時的なエラーの可能性がありますが、このエラーが引き続き発生し、サーバーが通信できない場合は、サーバーのネットワーク構成を確認してください。 |
| 005165 | WARN | errno %d\n | **原因**: メッセージの送信または受信に失敗しました。<br><br>**処置**: 詳細については、隣接するログメッセージを確認してください。これは一時的なエラーの可能性がありますが、このエラーが引き続き発生し、サーバーが通信できない場合は、サーバーのネットワーク構成を確認してください。 |
| 005166 | WARN | poll 0x%hx\n | **原因**: メッセージの送信または受信に失敗しました。<br><br>**処置**: 詳細については、隣接するログメッセージを確認してください。これは一時的なエラーの可能性がありますが、このエラーが引き続き発生し、サーバーが通信できない場合は、サーバーのネットワーク構成を確認してください。 |
| 005167 | WARN | handler for sys %s\n | **原因**: メッセージの送信または受信に失 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 敗しました。<br><br>**処置**: 詳細については、隣接するログメッセージを確認してください。これは一時的なエラーの可能性がありますが、このエラーが引き続き発生し、サーバーが通信できない場合は、サーバーのネットワーク構成を確認してください。 |
| 005225 | WARN | so_driver::handle_error: sending/receiving data message errno %d: %s | **原因**: メッセージの送受信に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。これは一時的なエラーの可能性もありますが、このエラーが続き、サーバが通信できない場合は、サーバのネットワーク設定を確認してください。 |
| 005235 | WARN | found tcp connection to iwstp\n | |
| 005236 | WARN | didn’t find tcp connection to iwstp\n | |
| 005237 | WARN | lcm_handler retry send from %s:%s to %s:%s (%d)\n | |
| 005238 | WARN | detected duplicate request from %s:%s to %s:%s\n | |
| 005239 | WARN | lcm_handler retry timer set to %d based on lcd remote timeout of %d s (%d s) | |
| 005240 | WARN | lcm_handler retry count/time (%d/%zu) has exceeded the maximum. Giving up…\n | |
| 005241 | WARN | dup_list: %s %s %s %s %d %d (%d) | |
| 005242 | WARN | clean up stale dup_list entry (%d s): %s:%s to %s:%s last: %s | |
| 005243 | WARN | add new dup_list entry %s:%s to %s:%s %d %d (%d | |
| 005244 | WARN | closing fd %d\n | |
| 005245 | WARN | openpoll fd %d\n | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 006012 | ERROR | quickCheck script '%s' (%d) failed to exit after %u seconds. Forcibly terminated. Please examine the script or adjust the LKCHECKINTERVAL parameter in %s. | 原因: quickCheck スクリプトの処理に時間がかかりすぎているか、スクリプトがハングしています。<br><br>処置: メッセージテキストに記載されている手順を実行してください。 |
| 006014 | ERROR | LKCHECKINTERVAL parameter is too short. It is currently set to %d seconds. It should be at least %ld seconds. Please adjust this parameter in %s and execute 'kill %d' to restart the lkcheck daemon. | |
| 006102 | ERROR | COMMAND OUTPUT: $LKROOT/bin/sendevent | 原因: これは、「sendevent」(イベント生成) コマンドの結果です。<br><br>処置: 隣接するログメッセージで詳細を確認してください。 |
| 006103 | ERROR | COMMAND OUTPUT: $LKROOT/bin/sendevent | 原因: これは、「sendevent」(イベント生成) コマンドの結果です。<br><br>処置: 隣接するログメッセージで詳細を確認してください。 |
| 006104 | ERROR | COMMAND OUTPUT: $LKROOT/bin/sendevent | 原因: これは、「sendevent」(イベント生成) コマンドの結果です。<br><br>処置: 隣接するログメッセージで詳細を確認してください。 |
| 006502 | ERROR | CPU usage has exceeded the threshold ($threshold%) for $count check cycles. | |
| 006504 | ERROR | Could not open /proc/meminfo | |
| 006505 | ERROR | Memory usage has exceeded the threshold ($threshold%) for $count check cycles. | |
| 006508 | ERROR | [$SUBJECT event] mail returned $err | |
| 006509 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 006511 | ERROR | snmptrap returned $err for Trap 190 | |
| 006512 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 006514 | ERROR | [$SUBJECT event] mail returned $err | |
| 006515 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 006517 | ERROR | snmptrap returned $err for Trap 200 | |
| 006518 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 006520 | ERROR | Failed to update error count in $cpu_file: $! | |
| 006521 | ERROR | Failed to update error count in $mem_file: $! | |
| 006523 | ERROR | The SNHC_CPUCHECK_THRESHOLD setting is not valid. Please set SNHC_CPUCHECK_THRESHOLD to a value between 10 and 99 in /etc/default/LifeKeeper. If not set, the value will default to 99. | |
| 006524 | ERROR | The SNHC_CPUCHECK_TIME setting is not valid. Please set SNHC_CPUCHECK_TIME to a value between 1 and 100 in /etc/default/LifeKeeper. If not set, the value will default to 1. | |
| 006525 | ERROR | The SNHC_MEMCHECK_THRESHOLD setting is not valid. Please set SNHC_MEMCHECK_THRESHOLD to a value between 10 and 99 in /etc/default/LifeKeeper. If not set, the value will default to 99. | |
| 006526 | ERROR | The SNHC_MEMCHECK_TIME setting is not valid. Please set SNHC_MEMCHECK_TIME to a value between 1 and 100 in /etc/default/LifeKeeper. If not set, the value will default to 1. | |
| 006528 | ERROR | Could not open /proc/stat | |
| 006529 | ERROR | Could not open /proc/stat | |
| 006530 | ERROR | Could not use $tmp_path | |
| 007053 | ERROR | malloc failed. Assume that it is a monitoring target device. | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 007058 | ERROR | %s: %s failed on '%s', result:%d, Sense Key = %d. | **原因**: SCSI デバイスをリザーブできませんでした。またはステータスを確認できませんでした。これは、ストレージが誤動作しているか、ディスクが別のサーバによってリザーブされているためです。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、リソースが適切に処理されていることを確認してください。 |
| 007059 | ERROR | %s: %s failed on '%s', result:%d. | **原因**: SCSI デバイスをリザーブできませんでした。またはステータスを確認できませんでした。これは、ストレージが誤動作しているか、ディスクが別のサーバによってリザーブされているためです。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、リソースが適切に処理されていることを確認してください。 |
| 007060 | EMERG | %s: failure on device '%s'. SYSTEM HALTED. | **原因**: SCSI デバイスをリザーブできませんでした。またはステータスを確認できませんでした。これは、ストレージが誤動作しているか、ディスクが別のサーバによってリザーブされているためです。サーバはリブート/停止されます。<br><br>**処置**: ストレージが適切に動作していることを確認し、適切に動作している場合は、リソースが適切に処理されており、別のサーバで in service になっていることを確認してください。 |
| 007072 | ERROR | %s: failed to open SCSI device '%s', initiate recovery. errno=0x%x, retry count=%d. | **原因**: 保護された SCSI デバイスを開くことができませんでした。デバイスに障害が発生しているか、システムからデバイスが削除されている可能性があります。<br><br>**処置**: システムが停止されるか、バック |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | アップノードへのフェイルオーバが開始されます。この場合のデフォルトアクションはフェイルオーバですが、SCSIERROR の設定で変更可能です。 |
| 007073 | ERROR | %s: failed to open SCSI device '%s', RETRY. errno=%d, retry count=%d. | **原因**: 保護された SCSI デバイスを開くことができませんでした。デバイスに障害が発生しているか、システムからデバイスが削除されている可能性があります。<br><br>**処置**: このエラーは致命的なものではありません。操作は 5 秒以内で再試行されます。問題が続行する場合、システムは停止またはリソースのフェイルオーバを実行します。 |
| 007075 | ERROR | %s: RESERVATION CONFLICT on SCSI device '%s'. ret=%d, errno=0x%x, retry count=%d. | **原因**: 別のサーバと競合しているため、SCSI デバイスをリザーブできませんでした。これは、ストレージが誤動作しているか、ディスクが別のサーバによってリザーブされているためです。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、リソースが適切に処理されていることを確認してください。 |
| 007077 | ERROR | %s: DEVICE FAILURE on SCSI device '%s', initiate recovery. ret=%d, errno=0x%x, retry count=%d. | **原因**: SCSI デバイスをリザーブできませんでした。またはステータスを確認できませんでした。これは、ストレージが誤動作しているか、ディスクが別のサーバによってリザーブされているためです。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、リソースが適切に処理されていることを確認してください。 |
| 007078 | ERROR | %s: DEVICE FAILURE on SCSI device '%s', RETRY. ret=%d, errno=0x%x, retry count=%d. | **原因**: SCSI デバイスをリザーブできませんでした。またはステータスを確認でき |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ませんでした。これは、ストレージが誤動作しているか、ディスクが別のサーバによってリザーブされているためです。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、リソースが適切に処理されていることを確認してください。 |
| 010002 | WARN | flag $flag not present, send message again. | **原因**: このメッセージは、再試行される未完のプロセスを示します。<br><br>**処置**: 隣接するログメッセージで、繰り返されているエラーを確認してください。 |
| 010003 | ERROR | COMMAND OUTPUT: $LKBIN/ins_remove | **原因**: このメッセージは「ins_remove」コマンドの結果の一部です。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。これは実際にはエラーではない可能性もあります。 |
| 010006 | WARN | flg_list -d $i took more than $pswait seconds to complete… | **原因**: サーバのフラグリスト処理の時間が予想以上にかかりました。別のサーバとの接続に問題が発生している可能性もあります。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 010007 | ERROR | flag $flag not present, switchovers may occur. | **原因**: クラスタ内のいずれかのサーバに、現在のサーバからのフェイルオーバ操作の不許可を指示できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、クラスタで予期しない動作を監視してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 010008 | WARN | flag $flag not present, send message again. | **原因**: プロセスは不完全ですが、再試行されます。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、繰り返されている警告/エラーをチェックしてください。 |
| 010023 | FATAL | LifeKeeper failed to initialize properly. | **原因**: LifeKeeper を起動しようとしている間に、致命的なエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 010025 | ERROR | Unable to get a unique tag name on server “%s” for template resource “%s” | **原因**: ストレージリソースの作成プロセスで適合するタグを自動的に生成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。他のエラーがない場合は、操作を再試行してください。 |
| 010034 | FATAL | Unable to start lcm. | **原因**: ソフトウェアのコアコンポーネントを開始できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、レポートされている問題を解決してください。 |
| 010038 | WARN | Waiting for LifeKeeper core components to initialize has exceeded 10 seconds. Continuing anyway, check logs for further details. | **原因**: ソフトウェアの一部が、予想よりも起動に時間がかかっています。<br><br>**処置**: メッセージテキストに記載されている手順を実行してください。 |
| 010039 | WARN | Waiting for LifeKeeper core components to initialize has exceeded 10 seconds. Continuing anyway, check logs for further details. | **原因**: ソフトウェアの一部が、予想よりも起動に時間がかかっています。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: メッセージテキストに記載されている手順を実行してください。 |
| 010046 | ERROR | The dependency creation failed on server $SERVER:" `cat $TEMP_FILE` | **原因**: 所定のサーバで依存関係を作成できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010063 | ERROR | $REMSH error | **原因**: 別のサーバからデータをバックアップする要求が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010085 | ERROR | lkswitchback($MACH): Automatic switchback of \"$loctag\" failed | **原因**: リソースが予想どおりにスイッチバックされませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010102 | ERROR | admin machine not specified | **原因**:「getlocks」操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 010107 | WARN | Lock for $m is ignored because system is OOS | **原因**: ロックが作成されたシステムがアクティブではないため、ロックは無視されました。<br><br>**処置**: 関連するエラーをログで確認してください。これは無害なエラーの可能性 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | もあります。 |
| 010108 | ERROR | lock acquisition timeout | **原因**: ロックの取得に予想よりも時間がかかりました。 または許容時間を超えました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010109 | ERROR | could not get admin locks." `cat /tmp/ER$$` | **原因**: ソフトウェアは、リソースの管理に必要なロックの取得に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010112 | ERROR | lcdrcp failed with error no: $LCDRCPRES | **原因**: ファイルを別のサーバにコピーできませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010116 | ERROR | unable to set !lkstop flag | **原因**: サーバがユーザの要求によって停止されていることを示すためのフラグを設定できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010121 | ERROR | Extended logs aborted due to a failure in opening $destination. ($syserrmsg) | **原因**: 拡張ログファイルを開くときに、ユーティリティ「lkexterrlog」が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください。 |
| 010132 | ERROR | Unable to retrieve reservation id from "%s". Error: "%s". Attempting to regenerate. | 原因: SCSI 3 Persistent Reservations に使用する一意の ID の取得元のファイル /opt/LifeKeeper/config/.reservation_id を開くことができません。<br><br>処置: ありません。ID の再生成とファイルの更新が試行されます。 |
| 010135 | ERROR | The current reservation ID of "%s" is not unique within the cluster. A new ID must be generated by running "%s/bin/genresid -g" on "%s". | 原因: システムに定義されたリザベーション ID は、クラスタ内で一意でないため、使用できません。<br><br>処置: このノードのすべてのリソースを out of service にしてから、「/opt/LifeKeeper/bin/genresid -g」を実行して一意のリザベーション ID を生成してください。 |
| 010136 | ERROR | Unable to store reservation id in "%s". Error: "%s" | 原因: SCSI 3 Persistent Reservations に使用する一意の ID の保存先のファイル /opt/LifeKeeper/config.reservation_id を開くことができません。<br><br>処置: ファイルを開くことに失敗した理由としてリストされているエラーを修正し、このノードのすべてのリソースを out of service にしてから、「/opt/LifeKeeper/bin/genresid -g」を実行して新しい一意のリザベーション ID を生成してください。 |
| 010137 | ERROR | Failed to generate a reservation ID that is unique within the cluster. | 原因: 生成したリザベーション ID はすでにクラスタ内の別のノードに定義されています。リザベーション ID はクラスタ内で一意である必要があります。<br><br>処置: このノードのすべてのリソースを |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | out of service にしてから、「/opt/LifeKeeper/bin/genresid -g」を実行して新しい一意のリザベーション ID を生成してください。 |
| 010138 | ERROR | %s | |
| 010139 | WARN | %s | |
| 010140 | ERROR | $COMMAND_SNMPTRAP returned $exit code for Trap $oid:$result | |
| 010141 | ERROR | LK_TRAP_MGR is specified in /etc/default/LifeKeeper, but $COMMAND_SNMPTRAP command is not in PATH. | |
| 010142 | ERROR | $COMMAND_EMAIL returned $exitcode:$result | |
| 010143 | ERROR | LK_NOTIFY_ALIAS is specified in /etc/default/LifeKeeper, but $COMMAND_EMAIL command is not in PATH. | |
| 010144 | ERROR | can’t opendir $LICENSE_DIR: $! | |
| 010145 | ERROR | lktest failed | |
| 010146 | ERROR | lkcheck failed | |
| 010147 | ERROR | ins_list failed: exit code = $exit_code | |
| 010159 | ERROR | Maintenance mode disable currently in progress, can’t enable maintenance mode. If this problem persists, consider using the —force option. | |
| 010160 | ERROR | Maintenance mode enable currently in progress, can’t disable maintenance mode. If this problem persists, consider using the —force option. | |
| 010161 | ERROR | $tag is not a valid resource tag on the local machine, aborting. Please check the spelling and try again. | |
| 010163 | ERROR | $cmd script not found or not executable on system $sys. | |
| 010165 | ERROR | An error occurred while running \’$LKROOT/lkadm/subsys/appsuite/sap/bin/$cmd -m=${opt_mode}${tag_cmd}${force_cm | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | d}\' on system $sys (exit code: $remexec_ret). Please inspect the logs on that system for more information. | |
| 010168 | WARN | Maintenance mode was not fully ${opt_mode}d for at least one resource on system $sys. | |
| 010172 | ERROR | Maintenance mode was not fully ${opt_mode}d for the requested resources on at least one system in the cluster. | |
| 010173 | ERROR | LifeKeeper is not running on system $me. Unable to ${opt_mode} maintenance mode. Aborting. | |
| 010179 | ERROR | Maintenance mode action \'${opt_mode}\' did not complete successfully for resource $tag on system $me. | |
| 010181 | ERROR | An error occurred while attempting maintenance mode action \'${opt_mode}\' on system $me for resources: @{local_hier_tags}. | |
| 010187 | ERROR | Resource $tag has not been extended to system $sys. | |
| 010222 | ERROR | scsifree(%s): LKSCSI_Release(%s) unsuccessful | **原因**: リザーブされていると思われる SCSI デバイスは予想どおりに解放されませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。システムが正常に機能している場合、このエラーは無害な可能性があります。 |
| 010231 | ERROR | scsiplock(%s): reserve failed. | **原因**: SCSI デバイスでリザーブを取得できませんでした<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 010250 | ERROR | Failed to exec command ‘%s’ | 原因:「lklogmsg」ツールがサブコマンド {command} の実行に失敗しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。サブコマンドが存在しており、有効なコマンドまたはプログラムであることを確認してください。通常の操作でこのメッセージが表示された場合は、サポートに問い合わせてください。 |
| 010256 | ERROR | scsi_tur(%s): open failed. | |
| 010260 | ERROR | scsi_tur(%s): test unit ready failed. | |
| 010402 | EMERG | local recovery failure on resource $opts{‘N’}, trigger VMware HA… | 原因: LifeKeeper Single Server Protection の動作中に、リソースを復旧できませんでした。(VMware-HA が有効になっている場合は) 障害を処理するために VMware-HA が間もなくトリガされます。<br><br>処置: 処置は必要ありません。VMware が障害を処理します。 |
| 010413 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | 原因: これは、失敗した可能性がある「snmptrap」コマンドの結果です。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010420 | EMERG | local recovery failure on resource $opts{‘N’}, trigger reboot… | 原因: LifeKeeper Single Server Protection の動作中に、リソースを復旧できませんでした。障害を処理するために再起動が間もなくトリガされます。<br><br>処置: 処置は必要ありません。 |
| 010440 | ERROR | [$SUBJECT event] mail returned $err | 原因: これは、通知電子メールを「mail」コマンドで送信できなかったことを示 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | します。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010443 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | **原因**: これは、失敗した可能性がある「mail」コマンドの結果です。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010445 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | **原因**: これは、失敗した可能性がある「mail」コマンドの結果です。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010463 | ERROR | LifeKeeper: name of machine is not specified, ARGS=$ARGS | **原因**:「comm_down」イベントで無効な引数が指定されました。<br><br>**処置**: LifeKeeper 設定を確認し、操作を再試行してください。 |
| 010471 | ERROR | COMM_DOWN: Attempt to obtain local comm_down lock flag failed | **原因**: 別のノードとの通信障害の処理中に、ローカルロックを取得できませんでした。これにより、フェイルオーバの処理が中断されます。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。フェイルオーバが正常に実行されない場合は、サポートに問い合わせてください。 |
| 010482 | ERROR | LifeKeeper: name of machine is not specified, ARGS=$ARGS | **原因**:「comm_up」イベントで無効な引数 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | が指定されました。<br><br>**処置**: LifeKeeper 設定を確認し、操作を再試行してください。 |
| 010484 | WARN | flg_list -d $MACH check timed-out ($delay seconds). | **原因**:「flg_list」コマンドがタイムアウト値 {delay} 秒に達しました。 |
| 010487 | WARN | flg_list -d $MACH check timed-out, unintended switchovers may occur. | **原因**:「flg_list」コマンドがタイムアウト値に達しました。<br><br>**処置**: 意図しないスイッチオーバが発生した場合は、リソースツリーをスイッチバックします。 |
| 010492 | WARN | $m | **原因**: 他のサーバのいずれかが、このサーバ {server} には稼働していないように見えますが、Witness サーバが承諾していません。<br><br>**処置**: 他のサーバが稼働していないことを確認し、リソースを手動でスイッチオーバしてください。 |
| 010494 | ERROR | LifeKeeper: COMM_UP to machine $MACH completed with errors. | **原因**:「COMM_UP」イベントの間に、予期しない障害が発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 010503 | ERROR | lcdrecover hung or returned error, attempting kill of process $FPID | **原因**:「lcdrecover」の処理に時間がかかりすぎているか、エラーアウトしています。 |
| 010506 | ERROR | Intelligent Switchback Check Failed | **原因**:「lcdrecover」を 5 回実行し、いずれも失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: リソースツリーを手動でスイッチオーバしてください。 |
| 010535 | ERROR | LifeKeeper: name of machine is not specified, ARGS=$ARGS | |
| 010600 | ERROR | removing hierarchy remnants | |
| 010627 | WARN | Equivalency Trim: does not have a full complement of equivalencies. Hierarchy will be unextended from | |
| 010629 | WARN | Your hierarchy exists on only one server. Your application has no protection until you extend it to at least one other server. | |
| 010712 | ERROR | Unextend hierarchy failed | **原因**: サーバからリソース階層を拡張できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 010746 | ERROR | $ERRMSG Target machine \"$TARGET_MACH\" does not have an active LifeKeeper communication path to machine \"$aMach\" in the hierarchy." >&2 | **原因**: ターゲットサーバがクラスタ内の他のサーバと適切に通信していないため、階層を拡張できません。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。すべてのサーバが相互にコミュニケーションパスを持っていることを確認してください。 |
| 010763 | ERROR | lock failed | |
| 011000 | ERROR | appremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011001 | ERROR | depremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 製品を再起動してください。 |
| 011002 | ERROR | eqvremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011003 | ERROR | flgremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011004 | WARN | Illegal creation of resource | **原因**: これは、通常の状態では発生しません。 |
| 011009 | ERROR | insremote: unknown change field command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011010 | ERROR | insremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011011 | FATAL | %s | **原因**: LifeKeeper は IPC キーを取得できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 011012 | FATAL | semget(%s,%c) | **原因**: LifeKeeper はセマフォ設定 ID を取得できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 011013 | FATAL | shmget(%s,%c) | **原因**: システムは共有メモリセグメント |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | を割り当てることができませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 011014 | FATAL | prefix_lkroot("out")); | **原因**: /opt/LifeKeeper/out にアクセス中に、システムエラーが発生しました。<br><br>**処置**: /opt/LifeKeeper/out にアクセスできない理由を特定してください。 |
| 011015 | ERROR | Please contact the original supplier of your evaluation software, or visit http://www.us.sios.com for information about purchasing a permanent version of the LifeKeeper product. | **原因**: デモライセンスを実行しています。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011016 | ERROR | The product license installed on this system indicates single node operation but the LifeKeeper Single Server Protection package does not appear to be installed. Please install the LifeKeeper SSP package to ensure proper operation | **原因**: LifeKeeper Single Server Protection のライセンスがありますが、LifeKeeper Single Server Protection がインストールされていません。<br><br>**処置**: LifeKeeper Single Server Protection をインストールするか、実行している製品に適合するライセンスを入手してください。 |
| 011018 | ERROR | License manager initialization failed. | **原因**: ライセンスマネージャの初期化に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 011020 | EMERG | License manager initialization failed. | **原因**: ライセンスマネージャの初期化に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 011021 | EMERG | Error in obtaining LifeKeeper license key. | **原因**: ライセンスに問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011022 | EMERG | Error in obtaining LifeKeeper license key. | **原因**: ライセンスに問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011023 | EMERG | A recovery kit Protection Suite WAS NOT FOUND with the restricted LifeKeeper license. | **原因**: ライセンスに問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011024 | EMERG | Error in obtaining LifeKeeper license key. | **原因**: ライセンスに問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011025 | EMERG | A recovery kit Protection Suite WAS NOT FOUND with the restricted LifeKeeper license. | **原因**: LifeKeeper は有効なライセンスキーを見つけられませんでした。<br><br>**処置**: ライセンスキーがサーバに対して有効であることを確認し、操作を再試行してください。 |
| 011026 | EMERG | Error in obtaining LifeKeeper license key. | **原因**: ライセンスに不明な問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011027 | EMERG | A recovery kit Protection Suite WAS NOT FOUND with the restricted LifeKeeper license. | **原因**: ライセンスに問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 011028 | ERROR | Error, This version of LifeKeeper must be running in the %c%c locale. The environment variable %s is not set correctly. Please consult your documentation for how to set this. | **原因**: 設定されているロケールに問題があります。<br><br>**処置**: マニュアルなどを参照の上、適切なロケールを設定してください。 |
| 011029 | FATAL | can’t set reply system | **原因**: メッセージの送信に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。これは一時的なエラーの可能性もあります。 |
| 011030 | FATAL | can’t set reply mailbox | **原因**: メッセージの送信に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。これは一時的なエラーの可能性もあります。 |
| 011031 | ERROR | Failure reading output of ‘%s’ on behalf of %s | **原因**: 一時ファイル /tmp/OUT.{pid} にアクセスしている際にシステムエラーが発生しました。<br><br>**処置**: /tmp/OUT.{pid} にアクセスできない理由を特定してください。 |
| 011032 | ERROR | Failure reading output of ‘%s’ | **原因**: 一時ファイル /tmp/ERR.{pid} にアクセスしている際にシステムエラーが発生しました。<br><br>**処置**: /tmp/ERR.{pid} にアクセスできない理由を特定してください。 |
| 011033 | ERROR | event \"%s,%s\" already posted for resource with id \"%s\" | **原因**: このメッセージは情報提供の目的でのみ表示されます。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 011034 | ERROR | no resource has id of \"%s\" | 原因: LifeKeeper は {id} リソースを見つけられませんでした。<br><br>処置: パラメータを確認し、「sendevent」操作を再試行してください。 |
| 011044 | ERROR | flagcleanup:fopen(%s) | 原因: /opt/LifeKeeper/config/flg を読み取っている際にシステムエラーが発生しました。<br><br>処置: /opt/LifeKeeper/config/flg を読み取ることができない理由を特定してください。 |
| 011045 | ERROR | flagcleanup:fopen(%s) | 原因: /opt/LifeKeeper/config/flg を書き込んでいる際にシステムエラーが発生しました。<br><br>処置: /opt/LifeKeeper/config/flg を書き込むことができない理由を特定してください。 |
| 011046 | ERROR | flagcleanup:fputs(%s) | 原因: /opt/LifeKeeper/config/flg を書き込んでいる際にシステムエラーが発生しました。<br><br>処置: /opt/LifeKeeper/config/flg を書き込むことができない理由を特定してください。 |
| 011047 | ERROR | flagcleanup:rename(%s,%s) | 原因: /opt/LifeKeeper/config/.flg to /opt/LifeKeeper/config/flg の名前を変更している際にシステムエラーが発生しました。<br><br>処置: /opt/LifeKeeper/config/flg の名前を変更できない理由を特定してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 011048 | ERROR | flagcleanup:chmod(%s) | 原因: /opt/LifeKeeper/config/flg の権限を変更している際にシステムエラーが発生しました。<br><br>処置: LifeKeeper が /opt/LifeKeeper/config/flg の権限を変更できない理由を特定してください。 |
| 011049 | ERROR | License check failed with error code %d | 原因: ライセンスに問題があります。<br><br>処置: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011051 | ERROR | lcdinit: clearing Disk Reserve file failed | 原因: /opt/LifeKeeper/subsys/scsi/resources/disk/disk.reserve を書き込んでいる際にシステムエラーが発生しました<br><br>処置: /opt/LifeKeeper/subsys/scsi/resources/disk/disk.reserve を書き込むことができない理由を特定してください。 |
| 011052 | FATAL | malloc() failed | 原因: システムは LifeKeeper 用にメモリを割り当てられませんでした。<br><br>処置: データセグメントのプロセスの上限を増やしてください。 |
| 011053 | FATAL | /tmp/LK_IS_UNAVAIL | 原因: /tmp/LK_IS_UNAVAIL を書き込んでいる際にシステムエラーが発生しました。<br><br>処置: /tmp/LK_IS_UNAVAIL を書き込むことができない理由を特定してください。 |
| 011054 | FATAL | /tmp/LK_IS_UNAVAIL | 原因: /tmp/LK_IS_UNAVAIL を書き込んでいる際にシステムエラーが発生しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: /tmp/LK_IS_UNAVAIL を書き込むことができない理由を特定してください。 |
| 011055 | FATAL | config/LK_IS_ON | **原因**: config/LK_IS_ON を書き込んでいる際にシステムエラーが発生しました。<br><br>**処置**: config/LK_IS_ON を書き込むことができない理由を特定してください。 |
| 011056 | ERROR | License check failed with error code %d | **原因**: ライセンスに問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011057 | ERROR | lcdremote: unknown command type %d(‘%c’)\n | **原因**: メッセージの受信に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。これは一時的なエラーの可能性もありますが、このエラーが続き、サーバが通信できない場合は、サーバのネットワーク設定を確認してください。 |
| 011059 | FATAL | Could not write to: %s | **原因**: /opt/LifeKeeper/config/LK_START_TIME にアクセスしている際にシステムエラーが発生しました。<br><br>**処置**: /opt/LifeKeeper/config/LK_START_TIME にアクセスできない理由を特定してください |
| 011060 | FATAL | received NULL message | **原因**: メッセージの受信に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。これは一時的なエラ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ーの可能性もありますが、このエラーが続き、サーバが通信できない場合は、サーバのネットワーク設定を確認してください。 |
| 011061 | ERROR | unknown data type %d(‘%c’) on machine \"%s\"\n | **原因**: メッセージの受信に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。これは一時的なエラーの可能性もありますが、このエラーが続き、サーバが通信できない場合は、サーバのネットワーク設定を確認してください。 |
| 011062 | WARN | LifeKeeper shutdown in progress. Unable to perform failover recovery processing for %s\n | **原因**: LifeKeeper はシャットダウン中に所定のリソースをフェイルオーバできませんでした。<br><br>**処置**: リソースツリーを別のサーバに手動でスイッチオーバしてください。 |
| 011063 | WARN | LifeKeeper resource initialization in progress. Unable to perform failover recovery processing for %s\n | **原因**: LifeKeeper は起動中に所定のリソースをフェイルオーバできませんでした。<br><br>**処置**: LifeKeeper が起動したら、リソースツリーを手動でスイッチオーバしてください。 |
| 011068 | ERROR | ERROR on command %s | **原因**:「rlslocks」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するメッセージで詳細を確認してください。 |
| 011070 | ERROR | ERROR on command %s | **原因**:「getlocks」コマンドの実行中にエラーが発生しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細を確認してください。 |
| 011080 | FATAL | out of memory | **原因**: 内部エラー。<br><br>**処置**: データセグメントのプロセスの上限値を増やしてください。 |
| 011081 | FATAL | Failed to ask ksh to run: %s | **原因**: ksh の呼び出し中にシステムエラーが発生しました。<br><br>**処置**: pdksh (v8.0 以前) または steeley e-pdksh (v81 以降) のパッケージがインストールされていることを確認してください。 |
| 011082 | ERROR | Failed to remove: %s | **原因**: /tmp/LCM_IS_UNAVAIL を削除している際にシステムエラーが発生しました。<br><br>**処置**: /tmp/LCM_IS_UNAVAIL を削除できない理由を特定してください。 |
| 011083 | ERROR | Failed to remove: %s | **原因**: /tmp/LK_IS_UNAVAIL をリンク解除している際にシステムエラーが発生しました。<br><br>**処置**: /tmp/LK_IS_UNAVAIL をリンク解除できない理由を特定してください。 |
| 011084 | FATAL | Failed to generate an IPC key based on: %s | **原因**: /opt/LifeKeeper にアクセス中に、システムエラーが発生しました。<br><br>**処置**: /opt/LifeKeeper にアクセスできない理由を特定してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 011085 | ERROR | semget(%s,%c) failed | **原因**: セマフォを削除している際にシステムエラーが発生しました。<br><br>**処置**: セマフォを手動で削除してください。 |
| 011086 | ERROR | shmget(%s,%c) failed | **原因**: 共有メモリセグメントを削除している際にシステムエラーが発生しました。<br><br>**処置**: 共有メモリセグメントを手動で削除してください。 |
| 011087 | ERROR | semctl(IPC_RMID) failed | **原因**: セマフォを削除している際にシステムエラーが発生しました。<br><br>**処置**: セマフォを手動で削除してください。 |
| 011088 | ERROR | shmctl(IPC_RMID) failed | **原因**: 共有メモリセグメントを削除している際にシステムエラーが発生しました。<br><br>**処置**: 共有メモリセグメントを手動で削除してください。 |
| 011089 | FATAL | Execution of lcdstatus on remote system <%s> failed\n | **原因**: リモート {node} がダウンし、ネットワークからアクセスできません。または、リモートノードで別のシステムに問題が発生しています。<br><br>**処置**: リモートノードをオンラインにするか、隣接するメッセージで詳細を確認するか、リモートノードのログで詳細を確認してください。 |
| 011111 | ERROR | action \"%s\" on resource with tag \"%s\" has failed | **原因**: リソース {tag} のアクション {action} に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するメッセージで詳細を確認してください。 |
| 011112 | ERROR | netremote: network “%s” with device “%s” not found on “%s” | **原因**: LifeKeeper はネットワークデバイスを見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 011113 | ERROR | netremote: unknown subcommand type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011114 | ERROR | netremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011117 | ERROR | sysremote: system \"%s\" not found on \"%s\" | **原因**: 無効なシステム名が指定されました。<br><br>**処置**: システム名を再確認し、コマンドを再実行してください。 |
| 011118 | ERROR | sysremote: unknown subcommand type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011119 | ERROR | sysremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |
| 011120 | ERROR | typremote: unknown command type %d(‘%c’)\n | **原因**: 内部エラー。<br><br>**処置**: 製品を再起動してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 011129 | ERROR | Failure during run of ‘%s’ on behalf of %s | **原因**: コマンドの実行に失敗しました。<br><br>**処置**: メッセージで詳細を確認し、問題を特定してください。 |
| 011130 | ERROR | %s | **原因**: コマンド {command} が予期しない結果になりました。<br><br>**処置**: 隣接するエラーメッセージの内容に従って処置を実行してください。 |
| 011131 | EMERG | \nERROR!\nThe update of the expiration timestamp for the evaluation version of the LifeKeeper product on this system has failed. The evaluation product will no longer execute.\n\nPlease contact the original supplier of your evaluation software for assistance.\n\n | **原因**: デモライセンスに問題があります。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011132 | EMERG | \nERROR!\nThe expiration timestamp associated with the evaluation version of the LifeKeeper product on this system has been improperly altered. The evaluation software expires immediately.\n\nPlease contact the original supplier of your evaluation software, or visit http://www.us.sios.com for information about purchasing a permanent version of the LifeKeeper product.\n\n | **原因**: デモライセンスで、時間の改ざんが検出されました。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011133 | EMERG | \nERROR!\nThe expiration timestamp associated with the evaluation version of the LifeKeeper product on this system has been improperly altered. The evaluation software expires immediately.\n\nPlease contact the original supplier of your evaluation software, or visit http://www.us.sios.com for information about purchasing a permanent version of the LifeKeeper product.\n\n | **原因**: デモライセンスで、時間の改ざんが検出されました。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011134 | EMERG | \nERROR!\nThe evaluation version of the LifeKeeper product on this system has expired.\n\nPlease contact the original supplier of your evaluation software, or visit http://www.us.sios.com for information about | **原因**: この製品のデモライセンスが期限切れです。<br><br>**処置**: サポートに問い合わせ、新しいラ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | purchasing a permanent version of the LifeKeeper product.\n\n | イセンスを入手してください。 |
| 011135 | EMERG | \nERROR!\nThe expiration timestamp associated with the evaluation version of the LifeKeeper product on this system has been improperly altered. The evaluation software expires immediately.\n\nPlease contact the original supplier of your evaluation software, or visit http://www.us.sios.com for information about purchasing a permanent version of the LifeKeeper product.\n\n | **原因**: デモライセンスで、時間の改ざんが検出されました。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011136 | EMERG | \nIMPORTANT NOTICE!\nThe LifeKeeper product running on this system is an evaluation version which will expire on %s\nTo continue functioning beyond that time, the software must be upgraded to the permanent version.\n\nPlease contact the original supplier of your evaluation software, or visit http://www.us.sios.com for information about purchasing a permanent version of the LifeKeeper product.\n\n | **原因**: デモライセンスを実行しています。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011138 | EMERG | The LifeKeeper product on this system is using an evaluation license key which will expire at midnight on %s. To continue functioning beyond that time, a permanent license key must be obtained. | **原因**: デモライセンスを実行しています。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011142 | WARN | LifeKeeper Recovery Kit %s license key NOT FOUND | **原因**: {kit} の Application Recovery Kit のライセンスが見つかりませんでした。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011150 | ERROR | COMMAND OUTPUT: %s | **原因**: コマンド「eventslcm」が予期しない結果になりました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 011151 | EMERG | %s | **原因**: このバージョンの LifeKeeper Core パッケージは、中華人民共和国内または日本国内での使用に限られています。 |
| 011152 | EMERG | Localized license failure | **原因**: お客様のロケールと製品ライセンスが作成されたロケールが一致していません。<br><br>**処置**: サポートに問い合わせ、お客様のロケールに合った新しいライセンスを入手してください。 |
| 011154 | EMERG | Single Node flag check failed. | **原因**: LifeKeeper Single Server Protection のライセンスがありますが、LifeKeeper Single Server Protection がインストールされていません。<br><br>**処置**:LifeKeeper Single Server Protection をインストールするか、実行している製品に適合するライセンスを入手してください。 |
| 011155 | EMERG | %s | **原因**: この製品のライセンスキーが期限切れです。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011162 | EMERG | %s | **原因**: この製品のライセンスキーが期限切れです。<br><br>**処置**: サポートに問い合わせ、新しいライセンスを入手してください。 |
| 011163 | EMERG | Single Node license check failed | **原因**: LifeKeeper Single Server Protection のライセンスがありますが、LifeKeeper Single Server Protection がインストールされていません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: LifeKeeper Single Server Protection をインストールするか、実行している製品に適合するライセンスを入手してください。 |
| 011164 | EMERG | %s | **原因**: この製品のライセンスキーが期限切れです。<br><br>**処置**: サポートに問い合わせ、製品のパーマネントライセンスキーを入手してください。 |
| 011165 | ERROR | LifeKeeper initialize timed out in tag \"%s\" | |
| 015000 | ERROR | COMMAND OUTPUT: /opt/LifeKeeper/sbin/chpst | **原因**:「steeleye-lighttpd」プロセスでエラーが発生しました。エラーの詳細は実際のログメッセージに記載されています。<br><br>**処置**: 設定を修正すると、「steeleye-lighttpd」が自動的に再起動します。 |
| 103001 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | **原因**: db2nodes.cfg にサーバ名が含まれていません。<br><br>**処置**: db2nodes.cfg が有効であることを確認してください。 |
| 103002 | ERROR | LifeKeeper was unable to get the version for the requested instance "%s" | **原因**:「db2level」コマンドが DB2 のバージョンを返しませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103003 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103004 | ERROR | Unable to get the information for resource "%s" | **原因**: リソース情報の取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103005 | ERROR | Unable to get the information for resource "%s" | **原因**: リソース情報の取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103006 | ERROR | Unable to get the instance information for resource "%s" | **原因**: インスタンス情報の取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103007 | ERROR | Unable to get the instance home directory information for resource "%s" | **原因**: インスタンスのホームディレクトリのパスの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103008 | ERROR | Unable to get the instance type information for resource "%s" | **原因**: DB2 Application Recovery Kit は、無効なインスタンスタイプを見つけました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103009 | ERROR | LifeKeeper has encountered an error while trying to get the database configuration parameters for database "$DB" | **原因**:「db2 get db cfg for $DB」コマンドの実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103012 | ERROR | LifeKeeper was unable to start the database server for instance "%s" | **原因**: 要求された DB2 インスタンスの起動が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「restore」操作を再試行してください。 |
| 103013 | ERROR | LifeKeeper was unable to start the database server for instance "%s" | **原因**: 要求された DB2 インスタンスの起動が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「restore」操作を再試行してください。 |
| 103015 | ERROR | An entry for the home directory "%s" of instance "%s" does not exist in "/etc/fstab" | **原因**: マルチパーティションのデータベースのインスタンスのホームディレクトリは、「/etc/fstab」に存在する必要があります。<br><br>**処置**: ホームディレクトリが「/etc/fstab」に存在することを確認してください。 |
| 103016 | ERROR | LifeKeeper was unable to mount the home directory for the DB2 instance "%s" | **原因**: マルチパーティションのデータベースのインスタンスのホームディレクトリをマウントできませんでした。<br><br>**処置**: ホームディレクトリがマウントされていることを確認し、操作を再試行してください。 |
| 103017 | ERROR | Unable to get the instance nodes information for resource "%s" | **原因**: インスタンスノードの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103018 | ERROR | LifeKeeper was unable to start database partition server "%s" for instance "%s" | **原因**: 要求された DB2 インスタンスの起動が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「restore」操作を再試行してください。 |
| 103020 | ERROR | LifeKeeper was unable to stop the database server for instance "%s" | **原因**: 要求された DB2 インスタンスのシャットダウンが失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「remove」操作を再試行してください。 |
| 103021 | ERROR | LifeKeeper was unable to stop the database server for instance "%s" | **原因**: 要求された DB2 インスタンスのシャットダウンが失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「remove」操作を再試行してください。 |
| 103023 | ERROR | Unable to get the instance nodes information for resource "%s" | **原因**: インスタンスノードの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103024 | ERROR | LifeKeeper was unable to stop database partition server "%s" for instance "%s" | **原因**: 要求された DB2 インスタンスのシャットダウンが失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 修正してから、「remove」操作を再試行してください。 |
| 103026 | ERROR | Unable to get the instance nodes information for resource "%s" | **原因**: インスタンスノードの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103027 | FATAL | The argument for the DB2 instance is empty | **原因**: 作成操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 103028 | FATAL | Unable to determine the DB2 instance home directory | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>**処置**: インスタンスの所有者の名前がインスタンス名と同じであることを確認し、操作を再試行してください。 |
| 103029 | FATAL | Unable to determine the DB2 instance type | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103030 | FATAL | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103031 | ERROR | The path "%s" is not on a shared filesystem | **原因**: インスタンスのホームディレクト |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | リは、共有ファイルシステム上になければなりません。<br><br>**処置**: パスが共有ファイルシステム上にあることを確認し、作成操作を再試行してください。 |
| 103032 | ERROR | LifeKeeper was unable to get the DB tablespace containers for instance "%s" or the log path for one of its databases | **原因**: LifeKeeper はデータベースのテーブルスペースコンテナの場所を特定できなかったか、コンテナがマウント済みのファイルシステム上のパスにあることを確認できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「create」操作を再試行してください。 |
| 103033 | ERROR | The path "%s" is not on a shared filesystem | **原因**: データベースのテーブルスペースコンテナのパスは、共有ファイルシステム上になければなりません。<br><br>**処置**: データベースのテーブルスペースコンテナが共有ファイルシステム上にあることを確認し、操作を再試行してください。 |
| 103034 | ERROR | A DB2 Hierarchy already exists for instance "%s" | **原因**: LifeKeeper によってすでに保護されている DB2 インスタンスを保護しようとしました。<br><br>**処置**: 別の DB2 インスタンスを LifeKeeper の保護用に選択する必要があります。 |
| 103035 | ERROR | The file system resource "%s" is not in-service | **原因**: DB2 リソースが依存するファイルシステムは in service である必要があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: ファイルシステムリソースが in service であることを確認して、「create」操作を再試行してください。 |
| 103036 | ERROR | Unable to create the hierarchy for raw device "%s" | **原因**: LifeKeeper は、リソース {raw device} を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから「create」操作を再試行する必要があります。 |
| 103037 | ERROR | A RAW hierarchy does not exist for the tag "%s" | **原因**: LifeKeeper は、raw リソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103038 | ERROR | LifeKeeper was unable to create a dependency between the DB2 hierarchy "%s" and the Raw hierarchy "%s" | **原因**: 要求された親 DB2 リソースと子 Raw リソースとの間の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、「create」操作を再試行してください。 |
| 103039 | ERROR | LifeKeeper could not disable the automatic startup feature of DB2 instance "%s" | **原因**: DB2 の設定を更新しようとして、予期しないエラーが発生しました。<br><br>**処置**: システム起動時にインスタンスが自動で起動されないようにするために、DB2AUTOSTART を手動で更新する必要があります。 |
| 103040 | ERROR | DB2 version "%s" is not installed on server "%s" | **原因**: LifeKeeper は DB2 のインスト一 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ル場所を見つけられませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103041 | ERROR | The instance owner "%s" does not exist on target server "%s" | **原因**:「canextend」操作または「extend」操作の間に、テンプレートサーバから DB2 インスタンスの所有者を取得できませんでした。<br><br>**処置**: 指定したサーバに DB2 インスタンスの所有者が存在していることを確認してください。所有者が存在しない場合、クラスタ内のすべてのサーバで、同じ uid および gid を使用して作成する必要があります。 |
| 103042 | ERROR | The instance owner "%s" uids are different on target server "%s" and template server "%s" | **原因**: DB2 インスタンスの所有者 {user} のターゲットサーバ {target server} のユーザ ID が、テンプレートサーバ {template server} のユーザ {user} の値と一致しません。<br><br>**処置**: DB2 インスタンスの所有者 {user} のユーザ ID はクラスタ内のすべてのサーバで一致している必要があります。ユーザ ID の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103043 | ERROR | The instance owner "%s" gids are different on target server "%s" and template server "%s" | **原因**:DB2 インスタンスの所有者 {user} のターゲットサーバ {target server} のグループ ID が、テンプレートサーバ {template server} のユーザ {user} の値と一致しません。<br><br>**処置**: DB2 インスタンスの所有者 {user} のグループ ID はクラスタ内のすべてのサーバで一致している必要があります。グループ ID の不一致をすべてのサーバで手動で修正し、「canextend」操作 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | を再試行する必要があります。 |
| 103044 | ERROR | The instance owner "%s" home directories are different on target server "%s" and template server "%s" | **原因**: ターゲットサーバ {target server} のユーザ {user} のホームディレクトリが、テンプレートサーバ {template server} の DB2 インスタンスの所有者のホームディレクトリと一致しません。<br><br>**処置**: DB2 インスタンスの所有者 {user} のホームディレクトリは、クラスタ内のすべてのサーバで一致している必要があります。ディレクトリの不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103045 | ERROR | LifeKeeper was unable to get the DB2 "SVCENAME" parameter for the DB2 instance | **原因**:「db2 get dbm cfg」コマンドの実行中に予期しないエラーが発生しました。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103046 | ERROR | Unable to get the value of the DB2 "SVCENAME" parameter for the DB2 instance %s. | **原因**: DB2 の「SVCENAME」パラメータが null に設定されています。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103047 | ERROR | LifeKeeper was unable to get the contents of the "/etc/services" file on the server "%s" | **原因**: テンプレートサーバの「/etc/services」に、DB2 インスタンスのサービス名が含まれていません。<br><br>**処置**:「/etc/services」内にある DB2 インスタンスのサービス名は、クラスタ内のすべてのサーバで一致している必要があります。サービス名の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103048 | ERROR | LifeKeeper was unable to get the contents of the "/etc/services" file on the server "%s" | **原因**: ターゲットサーバの「/etc/services」に、DB2 インスタンスのサービス名が |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 含まれていません。<br><br>**処置**:「/etc/services」内にある DB2 インスタンスのサービス名は、クラスタ内のすべてのサーバで一致している必要があります。サービス名の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103049 | ERROR | The "/etc/services" entries for the instance "%s" are different on target server "%s" and template server "%s" | **原因**: インスタンスの「/etc/services」エントリに不一致があります。<br><br>**処置**:「/etc/services」内にある DB2 インスタンスのサービス名は、クラスタ内のすべてのサーバで一致している必要があります。サービス名の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103050 | ERROR | The home directory "%s" for instance "%s" is not mounted on server "%s" | **原因**: LifeKeeper はマルチパーティションインスタンス用の db2nodes.cfg を見つけられませんでした。<br><br>**処置**: ホームディレクトリがマウントされていることを確認し、操作を再試行してください。 |
| 103051 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: テンプレートサーバからのリソース情報の取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103052 | ERROR | LifeKeeper was unable to add instance "%s" and/or its variables to the DB2 registry | **原因**:「db2iset」コマンドの実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103053 | ERROR | Usage: %s instance | |
| 103054 | ERROR | Unable to determine the DB2 instance type | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103055 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103056 | ERROR | Usage: %s instance | |
| 103058 | ERROR | Usage: %s instance | |
| 103059 | ERROR | Usage: %s instance | |
| 103060 | ERROR | Unable to determine the DB2 instance home directory | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>処置: インスタンスの所有者の名前がインスタンス名と同じであることを確認し、操作を再試行してください。 |
| 103061 | ERROR | Unable to determine the DB2 instance type | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103062 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | 原因: DB2 Application Recovery Kit は、DB2 インスタンスの該当するノードを見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103063 | ERROR | Unable to determine the DB2 install path | 原因: DB2 Application Recovery Kit は、そのインスタンスの DB2 を見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103064 | ERROR | Usage: nodes -t tag -a add_nodenum \| nodes -t tag -d delete_nodenum \| nodes -t tag -p | |
| 103065 | ERROR | Invalid input provided for "%s" utility operation, characters are not allowed. | 原因:「nodes」コマンドで無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 103066 | ERROR | Unable to get the information for resource "%s" | 原因: LifeKeeper は、リソース {tag} を見つけられませんでした。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 103067 | ERROR | The DB2 instance "%s" is not a EEE or Multiple Partition instance | 原因: リソースタグ {tag} はシングルパーティションインスタンスです。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 103069 | ERROR | Node "%s" is already protected by this hierarchy | 原因:「nodes」コマンドで無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 103070 | ERROR | Node number "%s" is the last remaining node protected by resource "%s". Deleting all nodes is not allowed. | 原因:「nodes」コマンドで無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 行してください。 |
| 103071 | ERROR | LifeKeeper is unable to get the equivalent instance for resource "%s" | 原因:「nodes」コマンドの実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 103072 | ERROR | Unable to set NodesInfo for resource "%s" on "%s" | 原因:「nodes」コマンドの実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 103073 | ERROR | Unable to set NodesInfo for resource "%s" on "%s" | 原因:「nodes」コマンドの実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 103074 | ERROR | Usage: %s instance | |
| 103075 | ERROR | Usage: %s instance | |
| 103076 | ERROR | Unable to determine the DB2 instance type | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103077 | ERROR | Unable to determine the DB2 instance home directory | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>処置: インスタンスの所有者の名前がインスタンス名と同じであることを確認 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | し、操作を再試行してください。 |
| 103078 | ERROR | The database server is not running for instance "%s" | 原因: DB2 インスタンスのプロセスチェックで、実行中のプロセスが見つかりませんでした。<br><br>処置: DB2 インスタンスを開始してください。 |
| 103079 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103080 | ERROR | One or more of the database partition servers for instance "%s" is down | 原因: すべてのデータベースパーティションサーバが実行されている必要があります。<br><br>処置: すべてのデータベースパーティションサーバが実行されていることを確認し、操作を再試行してください。 |
| 103081 | ERROR | DB2 local recovery detected another recovery process in progress for "%s" and will exit. | |
| 103082 | ERROR | Failed to create flag "%s" | 原因: DB2 ローカルリカバリ処理を制御するためのフラグを作成しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 103083 | ERROR | Failed to remove flag "%s" | 原因: DB2 ローカルリカバリ処理を制御するためのフラグを削除しようとして、予期しないエラーが発生しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 103084 | ERROR | Unable to determine the DB2 instance \"$Instance\" home directory | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>**処置**: インスタンスの所有者の名前がインスタンス名と同じであることを確認し、操作を再試行してください。 |
| 104002 | FATAL | $msg | **原因**: このメッセージは内部ソフトウェアエラーを示します。<br><br>**処置**: スタックトレースにエラーの原因が示されます。 |
| 104003 | FATAL | $self->Val('Tag') . " is not an SDR resource"); | **原因**: 非データレプリケーションリソースに対してデータレプリケーションアクションが実行されました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104010 | ERROR | $self->{'md'}: bitmap merge failed, $action | **原因**: ビットマップのマージ操作に失敗しました。<br><br>**処置**: ターゲットサーバにミラーや保護されたファイルシステムがマウントされている可能性があります。または、ビットマップファイルがターゲットに見つからない可能性があります。ターゲットサーバを確認してください。 |
| 104022 | ERROR | $argv[1]: mdadm failed ($ret) | **原因**:「mdadm」コマンドによるミラーへのデバイスの追加に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: これは通常、一時的な状態です。 |
| 104023 | ERROR | $_); | **原因**: メッセージに「mdadm」コマンドの結果が含まれています。 |
| 104025 | ERROR | failed to spawn monitor | **原因**: システムは、「mdadm -F」監視プロセスの開始に失敗しました。これは、通常の状態では発生しません。<br><br>**処置**: システムを再起動し、競合の可能性があるものが解決されていることを確認してください。 |
| 104026 | ERROR | cannot create $md | **原因**: ミラーデバイスを作成できませんした。<br><br>**処置**: デバイスがまだ使用中ではないこと、およびミラー作成用の他の全パラメータが正しいことを確認してください。 |
| 104027 | ERROR | $_ | **原因**: このメッセージに「mdadm」コマンドの結果が含まれています。 |
| 104035 | ERROR | Too many failures. Aborting resync of $md | **原因**: 正常範囲とは思われないほど長い時間、デバイスがビジー状態です。<br><br>**処置**: システムを再起動し、デバイスがビジーではなくなっていることを確認してください。 |
| 104036 | ERROR | Failed to start nbd-server on $target (error $port) | **原因**: ターゲットサーバで nbd-server プロセスを開始できませんでした。<br><br>**処置**: ターゲットディスクデバイスが使用可能であること、およびデバイス ID が変更されていないことを確認してくだ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | さい。 |
| 104037 | ERROR | Failed to start compression (error $port) | **原因**: システムは、「バランス」トンネルプロセスを開始できませんでした。または、ネットワークに問題が発生しました。<br><br>**処置**: ネットワークが正常に実行されていること、および 10000 ~ 10512 の範囲内にある TCP ポートがオープンで未使用の状態にあることを確認してください。ソフトウェアがすべてのシステムに正常にインストールされていることを確認してください。 |
| 104038 | ERROR | Failed to start nbd-client on $source (error $ret) | **原因**: nbd-client プロセスをソースサーバで開始できませんでした。<br><br>**処置**: レポートされている errno 値を調べ、レポートされている問題を解決してください。たとえば、errno 値 110 は「Connection timed out」(接続のタイムアウト) を示します。これは、ネットワークかファイアウォールに問題がある可能性があります。 |
| 104039 | ERROR | Failed to add $nbd to $md on $source | **原因**: これは通常、一時的な状態です。<br><br>**処置**: このエラーが続く場合、システムを再起動し、競合の可能性があるものを解決してください。 |
| 104045 | ERROR | failed to stop $self->{'md'} | **原因**: ミラーデバイスを停止できませんした。<br><br>**処置**: デバイスがビジーでないこと、またはマウントされていないことを確認してください。「mdadm —stop」を手動で実行し、デバイスを停止してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104048 | WARN | failed to kill $proc, pid $pid | **原因**: プロセスをシグナル化できませんでした。これは、プロセスが終了していることを示している可能性があります。<br><br>**処置**: 問題のプロセスが実行されていないことを確認し、実行されている場合は、システムを再起動して、終了できないプロセスを消去してください。 |
| 104050 | ERROR | Setting $name on $dest failed: $ret. Please try again. | **原因**: システムは「mirrorinfo」ファイル設定を設定できませんでした。<br><br>**処置**: ネットワークとシステムを確認し、ミラー設定操作を再試行してください。 |
| 104052 | FATAL | Specified existing mount point "%s" is not mounted | **原因**: マウントポイントがアンマウント状態になりました。<br><br>**処置**: マウントポイントがマウントされていることを確認し、操作を再試行してください。 |
| 104055 | ERROR | Failed to set up temporary $type access to data for $self->{'tag'}. Error: $ret | **原因**: ターゲットサーバでファイルシステムまたはデバイスを使用できません。ミラーされたデータは、ミラーが一時停止してから再開されるまでターゲットサーバでは使用できません。<br><br>**処置**: ターゲットサーバを再起動し、競合の可能性があるものを解決してください。 |
| 104057 | ERROR | Failed to undo temporary access for $self->{'tag'} on $self->{'sys'}. Error: $ret. Please verify that $fsid is not mounted on server $self->{'sys'}. | **原因**: ターゲットサーバでファイルシステムをアンマウントできませんでした。<br><br>**処置**: ファイルシステムおよびデバイスがターゲットサーバでビジーでないことを確認してください。ターゲットサーバ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | を再起動し、競合の可能性があるものを解決してください。 |
| 104062 | FATAL | Cannot find a device with unique ID "%s" | **原因**: ターゲットディスクを識別できませんでした。<br><br>**処置**: 適切なストレージリカバリキットがターゲットサーバにインストールされていることを確認してください。ターゲットディスクのデバイス ID が変更されていないことを確認してください。 |
| 104066 | FATAL | Cannot get the hardware ID of device "%s" | **原因**: ターゲットディスクデバイスの一意の ID が見つかりませんでした。<br><br>**処置**: 適切なストレージリカバリキットがターゲットサーバにインストールされていることを確認してください。ターゲットディスクのデバイス ID が変更されていないことを確認してください。 |
| 104067 | FATAL | Asynchronous writes cannot be enabled without a bitmap file | **原因**: 無効なパラメータでミラーを作成しようとしました。<br><br>**処置**: ビットマップファイルパラメータを指定するか、同期書き込みを指定する必要があります。 |
| 104068 | FATAL | Failed to extend dependent resource %s to system %s. Error %s | **原因**: 階層の拡張操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104070 | FATAL | Unable to extend the mirror "%s" to system "%s" | **原因**: 階層の拡張操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください。 |
| 104071 | ERROR | Failed to restore target device resources on $target->{'sys'} : $err | **原因**: ターゲットサーバで in service 操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104074 | FATAL | Cannot get the hardware ID of device "%s" | **原因**: ミラーに使用しようとしている基本のディスクデバイスを認識するストレージリカバリキットがありません。<br><br>**処置**: 適切なストレージリカバリキットがインストールされていることを確認してください。必要に応じ、/opt/LifeKeeper/subsys/scsi/resources/DEVNAME/device_pattern ファイルにデバイス名を入れてください。 |
| 104081 | FATAL | Cannot make the %s filesystem on "%s" (%d): | **原因**:「mkfs」コマンドが失敗しました。<br><br>**処置**:ディスクデバイスが書き込み可能で、エラーがないこと、および選択したファイルシステムのファイルシステムツールがインストールされていることを確認してください。 |
| 104082 | FATAL | %s | **原因**: このメッセージに「mkfs」コマンドの結果が含まれています。 |
| 104083 | FATAL | Cannot create filesys hierarchy "%s" | **原因**: リソースの作成に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104086 | ERROR | The "%s_data_corrupt" flag is set in "%s/subsys/scsi/resources/netraid/" on system "%s". To avoid data corruption, LifeKeeper will not restore the resource. | **原因**: 予期しないデータ破損を防ぐための警告データとして data corrupt フラグファイルが設定されています。ファイルが削除されるまで、このサーバ上のミラーをリストアできません。<br><br>**処置**: 問題のサーバでデータが有効な場合、以下のいずれかを実行できます。1) ファイルを削除してミラーをリストアします。2) LifeKeeper GUI または「mirror_action force」コマンドでミラーを強制的にオンラインにします。 |
| 104092 | ERROR | Mirror target resource movement to system %s : status %s | **原因**: 階層のスイッチオーバ操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104099 | ERROR | Unable to unextend the mirror for resource "%s" from system "%s" | **原因**: 階層の拡張解除に失敗しました。<br><br>**処置**: ターゲットサーバを再起動し、競合の可能性があるものを解決して、操作を再試行します。 |
| 104106 | ERROR | remote 'bitmap -m' command failed on $target->{'sys'}: $ranges | **原因**: ターゲットサーバでビットマップマージコマンドが失敗しました。これには、次の 2 つの原因が考えられます。1) ビットマップファイルが見つからないか、破損している。2) ミラー（md）デバイスがターゲットでアクティブになっている。<br><br>**処置**: ミラーおよび保護されているファイルシステムがターゲットでアクティブになっていることを確認してください。ターゲットのビットマップファイルが見つからない場合、ミラーを一時停止して再開し、ビットマップファイルを再 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 作成してください。 |
| 104107 | ERROR | Asynchronous writes cannot be enabled without a bitmap file | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104108 | ERROR | Local Partition not available | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104109 | ERROR | Cannot get the hardware ID of device "%s" | **原因**: ディスクデバイスの一意の ID を特定できませんでした。<br><br>**処置**: 適切なストレージリカバリキットがサーバにインストールされていることを確認してください。ディスクのデバイス ID が変更されていないことを確認してください。 |
| 104111 | FATAL | Insufficient input parameters for "%s" creation | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104112 | FATAL | Insufficient input parameters for "%s" creation | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104113 | FATAL | Insufficient input parameters for "%s" creation | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104114 | FATAL | Insufficient input parameters for "%s" creation | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104115 | FATAL | Insufficient input parameters for "%s" creation | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104117 | FATAL | Insufficient input parameters for "%s" creation | **原因**: ミラー作成操作で無効なパラメー |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | タが指定されました。 |
| 104118 | FATAL | Cannot unmount existing Mount Point "%s" | **原因**: マウントポイントがビジーです。<br><br>**処置**: ファイルシステムがビジーではないことを確認してください。ファイルシステムにアクセスする可能性があるプロセスまたはアプリケーションを停止してください。 |
| 104119 | FATAL | Invalid data replication resource type requested ("%s") | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |
| 104124 | EMERG | WARNING: A temporary communication failure has occurred between systems %s and %s. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should take one of the following resources out of service: %s on %s or %s on %s. The resource that is taken out of service will become the mirror target. | **原因**: ソースサーバとターゲットサーバの間で、一時的な通信障害 (スプリットブレインシナリオ) が発生しました。<br><br>**処置**: メッセージテキストに記載されている手順を実行してください。 |
| 104125 | ERROR | failed to start '$cmd $_[2] $user_args' on '$_[3]': | **原因**: 指定したコマンドが失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104126 | ERROR | $_ | **原因**: このメッセージに、メッセージ 104125 で失敗したとレポートされているコマンドの結果が含まれています。 |
| 104128 | FATAL | comm path/server not specified | **原因**: netraid.down スクリプトが、通信パスまたはサーバー名を指定せずに呼び出されました。このスクリプトは内部で |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 呼び出されるため、常に適切なパラメーターを使用する必要があります。<br><br>**処置**: このエラーを SIOS サポートに報告してください。 |
| 104129 | WARN | The replication connection for mirror “%s” (resource: “%s”) is down (reason: %s) | **原因**: ミラーへのレプリケーションの接続がダウン状態です。<br><br>**処置**: ネットワークを確認してください。 |
| 104130 | ERROR | Mirror resize failed on %s (%s). You must successfully complete this operation before using the mirror. Please try again. | **原因**: ミラーのサイズ変更操作で、リストされたシステム上にあるミラーのメタデータの更新に失敗しました。<br><br>**処置**: ミラーを使用するには、サイズヘンコウを正常に完了する必要があります。「mirror_resize」を再実行してください (必要に応じて、強制実行する -f を使用)。 |
| 104132 | ERROR | The partition "%s" has an odd number of sectors and system "%s&qquot; is running kernel >= 4.12. Mirrors with this configuration will not work correctly with DataKeeper. Please see the SIOS product documentation for information on how to resize the mirror. | **原因**: ミラーの作成用に選択されたパーティションまたはディスクには奇数のディスクセクターがあるため、DataKeeperで使用するにはサイズを変更する必要があります。<br><br>**処置**:「parted」コマンドを使用してパーティションのサイズを変更するか、可能な場合はプラットフォーム（VMware、AWS）ツールを使用してディスクのサイズを変更します。注記：データが失われる可能性があるため、この操作は慎重に行ってください。 |
| 104136 | ERROR | Extend failed. | **原因**: 階層の拡張操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | し、レポートされている問題を解決してください。 |
| 104143 | ERROR | Mirror resume was unsuccessful ($ret) | **原因**:ミラーを確立できませんした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104144 | ERROR | Unable to stop the mirror access for $self->{'md'} on system $self->{'sys'}. Error: $ret. Use the "mdadm —stop $self->{'md'}" command to manually stop the mirror. | **原因**: ミラーの一時停止時にターゲットノード上に作成されたミラーデバイスを停止できませんでした。<br><br>**処置**: デバイスがビジーでないこと、またはマウントされていないことを確認してください。「mdadm —stop」を手動で実行し、デバイスを停止してください。 |
| 104145 | WARN | Unable to dirty full bitmap.  Setting fullsync flag. | **原因**: ビットマップ全体をダーティにすることによる完全な再同期を実行できませんでした。代わりに fullsync フラグが使用されます。完全な同期は実行されるので、これは致命的なエラーではありません。<br><br>**処置**: ありません |
| 104146 | EMERG | WARNING: The target system %s for mirror %s has the target mirror %s currently active. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should reboot system %s. | **原因**: ミラーがターゲットシステムで構成されています。<br><br>**処置**: ターゲットシステムを再起動する必要があります。 その後、DataKeeper はミラーを再同期できるようになります。 |
| 104147 | EMERG | WARNING: The target system %s for mirror %s has the target disk %s currently mounted. In order to avoid data corruption, data resynchronization will not occur. MAN | **原因**: ミラーディスクがターゲットシステムにマウントされています。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | UAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should unmount %s on %s. A full resync will occur. | **処置**: 完全なミラー再同期を開始するには、ミラーディスクをターゲットシステムでアンマウントする必要があります。ディスクで追跡されていない変更が発生したため、完全な再同期が必要です。 |
| 104148 | EMERG | The storage configuration for mirror "%s (%s)" does not have a unique identifier and may have potential for data corruption in some environments in certain circumstances. Please refer to the SIOS Product Documentation for details on DataKeeper storage configuration options. | **原因**: ミラーリング用に選択されたディスクが、オペレーティングシステムにUUIDを提供していません。DataKeeperはUUIDがないとディスクをミラーできません。<br><br>**処置**: ディスク上にGPTパーティションテーブルを作成して、ディスクパーティションにUUIDを提供できます。 DataKeeper for Linux トラブルシューティング を参照してください。 |
| 104149 | ERROR | snmptrap returned $err for Trap 142 | |
| 104150 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 104152 | ERROR | [$SUBJECT event] mail returned $err | |
| 104153 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 104155 | EMERG | The mirror %s cannot be forced online at this time. The underlying disk %s is mounted, indicating possible data corruption. MANUAL INTERVENTION IS REQUIRED. You must unmount %s on %s and restore the mirror to the last known mirror source system. A full resync will need to be performed from the source system to %s. | **原因**: ターゲットシステムでミラーディスクがマウントされた。<br><br>**処置**: データ破損を防ぐため、すぐにディスクをアンマウントしてください。 |
| 104156 | WARN | Resynchronization of "%s" is in PENDING state. Current sync_action is: "%s" | **原因**: md デバイスの再同期がペンディング状態であることが検出されました。<br><br>**処置**: LifeKeeper は再同期を強制実行して問題の修正を試行します。関連するエラーをログで確認してください。エラーを解決した後、/proc/mdstat でペンディング状態がクリアされたこと、および d |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ataレプ リソースの再同期が進行中であるか、完了したことを確認してください。 |
| 104157 | WARN | /etc/sysconfig/raid-check update failed. Please %s "md%d" to SKIP_DEVS. | **原因**: /etc/sysconfig/raid-check を変更して、スキップする MD デバイスのリスト（SKIP_DEVS）にエントリを追加または削除することができません。<br><br>**処置**: raid-check または SKIP_DEVS に関連するエラーがないか、システムログを確認してください。リストされたmdを手動で追加または削除してください。 |
| 104158 | EMERG | WARNING: The local disk partition $self->{ part } for data replication device\n$self->{ md } has failed. MANUAL INTERVENTION IS REQUIRED. | **原因**: ミラーのローカルデバイスで問題が発生しました。LKDR_FAILURE でリカバリーアクションが設定されていないため、復旧するには手動による介入が必要です。<br><br>**処置**: システムログと LifeKeeper ログで、ローカルディスクに関連するエラーを確認してください。 |
| 104163 | WARN | The "%s_data_corrupt" flag is set in "%s/subsys/scsi/resources/netraid/" on system "%s". The mirror is being forced online. | **原因**: ミラーは強制的にオンラインになり、data_corrupt フラグを上書きします。指定されたシステム上のデータは、最新のデータとして扱われます。これが正しくない場合、データ破損またはデータ損失につながる可能性があります。<br><br>**処置**: ありません |
| 104164 | ERROR | The "%s_data_corrupt" flag for related mirror resource "%s" is set in "%s/subsys/scsi/resources/netraid/" on system "%s". To avoid data corruption, LifeKeeper will not restore this mirror or any related mirrors in the hierarchy. | **原因**: data_corrupt フラグは、階層内の1つ以上のミラーに存在します。追加データの破損を防ぐため、すべての data_corrupt フラグが解決されるまで、どのミラーも In Service になりません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: LifeKeeperのログを確認して、各ミラーが最後に In Service になっていた場所、つまり各ミラーの最新データが存在する場所を特定します。すべての階層が In Service であった以前のソースでミラーを In Service にし、ミラーがすべてのターゲットと同期できるようにする必要があります。 |
| 104165 | ERROR | The "%s_data_corrupt" flag for related mirror resource "%s" is set in "%s/subsys/scsi/resources/netraid/" on system "%s". The mirror resource "%s" is being forced online. | **原因**: ミラーは強制的にオンラインになっており、data_corrupt フラグを上書きします。指定されたシステム上のデータは、すべてのターゲットと同期される正しいデータとして扱われます。これが最新のデータでない場合、データ破損またはデータ損失につながる可能性があります。<br><br>**処置**: ありません |
| 104170 | ERROR | Failed to create "source" flag file on shared source %s to track mirror source. This may result in a full resync. | **原因**:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104171 | ERROR | Failed to create "source" flag file on %s to track mirror source. Target %s will not be added to mirror. | **原因**:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104172 | ERROR | The "source" flag file on %s does not contain a valid target (%s). Full resync to remaining targets is required. | **原因**:「ソース」フラグファイルには以前 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | のソースのシステム名が含まれている必要がありますが、リストされている名前は構成されたシステムのリストにありませんでした。<br><br>**処置**: この問題を SIOS サポートに報告してください。 |
| 104173 | ERROR | Failed to create "source" flag file on %s to track mirror source. | **原因**:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104174 | ERROR | Failed to create “previous_source” flag file to track time waiting on source. Will not be able to timeout. | **原因**:「previous_source」フラグファイルは、ミラーの以前のソースを追跡するためのミラーソース上に作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104175 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | **原因**: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104176 | ERROR | The "source" flag file on %s to track mirror source does not exist. Full resync is required. | **原因**:「ソース」フラグファイルはシステム上に存在する必要があり、これがないとミラーの整合性を検証できません。デ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ータの信頼性を確保するには、完全な再同期が必要です。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104177 | ERROR | Failed to determine amount of time waiting on %s. | **原因**: 以前のソースを待機する時間を決定できませんでした。以前のソースが見つからない場合、ターゲットは完全に再同期されて追加されます。<br><br>**処置**: ありません |
| 104178 | ERROR | Failed to update "source" flag file on target "%s", previous source must be merged first. | **原因**: ターゲットのソースフラグファイルは、同期されて停止されると更新されるため、次のIn Service時には以前のソースは必要ありません。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104180 | ERROR | Internal Error: "previous_source" has the local system name (%s). | **原因**: ローカルシステム名は、previous_source フラグファイルに含めることはできません。<br><br>**処置**: このエラーを SIOS サポートに報告してください。 |
| 104181 | ERROR | Internal Error: There are no targets waiting on %s to be merged. | **原因**: 以前のソースがマージされるのを待機しているターゲットがありません。<br><br>**処置**: このエラーを SIOS サポートに報告してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104182 | ERROR | Failed to create \"source\" flag file on %s to track mirror source. This may result in a full resync. | **原因**: リストされたターゲットに「ソース」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104186 | ERROR | Failed to create "last_owner" flag file on %s to track mirror source. This may allow in-service of mirror on old data. | **原因**:「last_owner」フラグファイルがソースに作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104187 | WARN | $REM_MACH has ${REM_TAG}_last_owner file, create flag ${FLAGTAG}_data_corrupt. | **原因**: リストされているシステムでは、ミラーが最後にIn Serviceになりました。<br><br>**処置**: リストされているシステムには last_owner ファイルがあります。これは、このシステムには最新のデータがあり、データの損失を防ぐためにミラーをIn Serviceにするのに最適なシステムである可能性が高いことを示しています。 |
| 104188 | WARN | $REM_MACH is not alive, create flag ${FLAGTAG}_data_corrupt. | **原因**: リストされているシステムは稼働していません。<br><br>**処置**: リストされているシステムは稼働していないため、そのシステムがローカルシステムよりも新しいミラーソースであったかどうかを判断できません。このため、ローカルシステムがミラーをIn Serviceにすることを自動的に許可しないでください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104200 | EMERG | Continue to wait for %s to merge bitmap and do partial resyncs to all targets, no timeout set. | **原因**: マルチターゲット構成では、すべてのターゲットが部分的に再同期できるように、以前のソースがビットマップをマージできるようになるまでターゲットは構成されません。/etc/defaults/LifeKeeper の LKDR_WAIT_ON_PREVIOUS_SOURCE_TIMEOUT エントリは、無期限に待機するために「-1」に設定されます。<br><br>**処置**: メッセージにリストされている以前のソースのステータスを確認し、クラスターへの再参加を妨げている問題を解決してください。 |
| 104201 | EMERG | To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: "%s/bin/mirror_action %s fullresync %s %s" on %s. | **原因**: マルチターゲット構成では、ターゲットは構成されず、以前のソースがクラスターに再参加するのを待機します。<br><br>**処置**: メッセージにリストされているコマンドを実行して、このターゲットおよび再同期を待機している残りのターゲットに対して即時の完全再同期を強制してください。 |
| 104202 | EMERG | Continue to wait for %s to merge bitmap and do partial resyncs to all targets. Continue to wait %d more seconds. | **原因**: マルチターゲット構成では、すべてのターゲットが部分的に再同期できるように、前のソースがビットマップをマージできるようになるまでターゲットは構成されません。 /etc/defaults/LifeKeeper の LKDR_WAIT_ON_PREVIOUS_SOURCE_TIMEOUT エントリは、待機する秒数に設定されます。以前のソースがその時間内にクラスターに参加しない場合、ターゲットは完全に再同期されて追加されます。<br><br>**処置**: メッセージにリストされている以前のソースのステータスを確認し、クラスターへの再参加を妨げている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104203 | EMERG | To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: \"%s/bin/mirror_action %s fullresync %s %s\" on %s. | **原因**: マルチターゲット構成では、ターゲットは構成されず、以前のソースがクラスターに再参加するのを待機します。<br><br>**処置**: メッセージにリストされているコマンドを実行して、このターゲットおよび再同期を待機している残りのターゲットに対して即時の完全再同期を強制してください。<br><br>**注記**: リストされたターゲットが削除されて戻らない場合でも、このコマンドを実行して待機を停止してください。 |
| 104207 | ERROR | Failed to create "data_corrupt" flag file on "%s". | **原因**: リストされたソースに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104208 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | **原因**: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104209 | ERROR | Failed to create "data_corrupt" flag file on "%s". | **原因**: リストされたソースに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104210 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104211 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104212 | ERROR | The "source" flag file on %s to track mirror source does not exist. Full resync to remaining targets is required. | 原因:「ソース」フラグファイルはシステム上に存在する必要があり、これがないとミラーの整合性を検証できません。データの信頼性を確保するには、完全な再同期が必要です。まだミラーされていないすべてのターゲットでは、完全な再同期が必要です。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください. |
| 104214 | ERROR | Failed to create "source" flag file on %s to track mirror source. | 原因:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104216 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104217 | ERROR | Failed to create "source" flag file on shared source %s to track mirror source. This may result in a full resync. | 原因:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104218 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104221 | ERROR | Failed to create "last_owner" flag file on %s to track mirror source. This may allow in-service of mirror on old data. | 原因: リストされたターゲットに「last_owner」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104222 | ERROR | Failed to create "last_owner" flag file to track mirror source". This may allow in-service of mirror on old data. | 原因:「last_owner」フラグファイルは、ミラーが最後にIn Serviceになった場所を特定するために使用されていま |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | す。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104223 | ERROR | Failed to create "last_owner" flag file to track mirror source". This may allow in-service of mirror on old data. | **原因**:「last_owner」フラグファイルは、ミラーが最後にIn Serviceになった場所を特定するために使用されています。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104224 | ERROR | Failed to create "previous_source" flag file. | **原因**:「previous_source」フラグファイルが作成されませんでした。これは、以前のソースのビットマップをマージして完全な同期を回避するために必要です。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104227 | ERROR | Failed to set %s to %s. | **原因**: このメッセージは、nbdドライバー（/sys/block/nbdX）のsysfsパラメーターの設定に失敗したことを示しています。<br><br>**処置**: このエラーを回避するには、/etc/default/LifeKeeper で次のうち1つ以上を調整する必要がある場合があります。<br>NBD_NR_REQUESTS<br>NBD_SCHEDULER<br>LKDR_ASYNC_LIMIT |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104232 | ERROR | Mirror resize failed on %s (%s). Could not set size to %d. | 原因: ミラーのサイズ変更操作に失敗しました。 |
| 104233 | ERROR | Mirror resize failed on %s (%s). Could not set bitmap to %s and bitmap-chunk to %d. | 原因: ミラーのサイズ変更操作に失敗しました。 |
| 104234 | ERROR | The mirror %s failed to resize. You must successfully complete this operation before using the mirror. Please try again. | 原因: ミラーのサイズ変更操作に失敗しました。 |
| 104235 | ERROR | mirror_resize of mirror %s failed due to signal "%s". | 原因: ミラーのサイズ変更操作に失敗しました。 |
| 104236 | EMERG | Resource "%s" is "OSF". The mirror "%s" will wait to replicate data until all resources in the hierarchy%s are in-service. This may indicate inconsistent data. Verify the data is correct before replicating data; replication will continue when all resources in the hierarchy%s are in-service. A full resync may be necessary (see "LKDR_WAIT_TO_RESYNC" in /etc/default/LifeKeeper). | |
| 104237 | WARN | Resource "%s" is "OSU". The mirror "%s" will wait to replicate data until all resources in the hierarchy%s are in-service. To replicate data immediately run: "%s/bin/mirror_action %s resume" on "%s" (see "LKDR_WAIT_TO_RESYNC" in /etc/default/LifeKeeper). | |
| 104238 | ERROR | Unable to read /sys/module/nbd/taint. Assuming that the SIOS nbd module is not loaded. | |
| 104239 | EMERG | $nbd_failure_msg | |
| 104242 | ERROR | Unable to read /proc/modules. Assuming that the SIOS nbd module is not loaded. | |
| 104251 | ERROR | There is no LifeKeeper protected resource with tag $tag on system $me. | 原因: 指定されたタグは、指定されたシ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ステム上のLifeKeeperで保護されているリソースに対応していません。<br><br>**処置**: リソースタグとシステム名が正しいことを確認してください。 |
| 104252 | ERROR | Resource $tag is not a $app/$typ resource. Please use the $ins_app/$ins_typ resource-specific canfailover script instead. | **原因**: scsi/netraid 固有の canfailover スクリプトが、非 scsi/netraid リソースに対して呼び出されました。<br><br>**処置**: 指定されたリソースの適切なアプリケーションとタイプに対応する canfailover スクリプトを使用してください（存在する場合）。 |
| 107015 | ERROR | Exported file system $opt_t cannot be accessed on $me. | |
| 112000 | FATAL | Usage: "%s %s". Specify the correct usage for the requested command. | **原因**: SAP HANA createスクリプトに無効なパラメーターが指定された。<br><br>**処置**: 適切なパラメーターをSAP HANA createスクリプトに指定してください。 |
| 112004 | ERROR | Neither the SID/instance pair nor the tag parameter were specified for the internal "%s" routine on %s. If this was a command line operation, specify the correct parameters. Otherwise, consult the troubleshooting documentation. | **原因**: 内部SAPオブジェクトの作成において無効なパラメーターが指定された。<br><br>**処置**: SIDおよびインスタンスまたはLifeKeeper のリソースタグ名のいずれかを指定してください。 |
| 112013 | ERROR | The file "%s" does not exist or was not readable on %s. Verify that the specified file exists and/or is readable. | **原因**: sapservices ファイルが存在しないか、読み込み不可。<br><br>**処置**: sapservices ファイルが存在し、読み込み可能であることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112017 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: 指定したサーバー上で指定したリソースに関連する情報を見つけることができない。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していて、すべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112019 | ERROR | The attempt to update the resource information field for resource %s has failed on %s. View the resource properties manually using "ins_list -t <tag>" to verify that the resource is functional. | **原因**: 指定したサーバーで指定したリソースの LifeKeeper リソース情報フィールドを更新できない。<br><br>**処置**: 指定したサーバーにリソースが存在しているか、もしくはLifeKeeper が起動しているかを確認してください。 |
| 112022 | ERROR | An error occurred while trying to find the IP address corresponding to "%s" on %s. Verify the IP address or host name exists in DNS or the hosts file. | **原因**: 指定したサーバーにIPアドレスもしくはホスト名が見つからない。<br><br>**処置**: IPアドレスもしくはDNS名がDNSもしくはローカルホストファイルに存在しているかを確認してください。 |
| 112024 | FATAL | There was an error verifying the NFS connections for SAP related mount points on $me. One or more NFS servers is not operational and needs to be restarted. | **原因**: SAP_NFS_CHECK_DIRS パラメーター に掲載されている少なくとも一つの NFS 共有ファイルシステムを現在使用できない。<br><br>**処置**: NFSサーバーの起動、すべての必要とされるNFS関連のサービスの起動、すべての必要とされるファイルシステムがエクスポートされていることを確認してください。 |
| 112028 | ERROR | Unable to determine the user name for the SAP administrative user for resource %s on %s. | **原因**: 指定したサーバー上に指定したリソース情報を確認することができない。<br><br>**処置**: 指定したサーバーに指定したリソ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ースが存在していて、すべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112029 | ERROR | The canextend script "%s" either does not exist or is not executable on %s. | 原因: SAP canextend スクリプトが存在していないか、実行できない。<br><br>処置: SAP canextend スクリプトが存在していて、実行可能か確認してください。 |
| 112037 | ERROR | Unable to create an internal object for the SAP instance using SID %s, instance %s, and tag %s on server %s. Verify that all necessary SAP file systems are mounted and accessible before reattempting the operation. | 原因: 指定した SAP インスタンスを表すための内部の SAP オブジェクトを作成することができない。<br><br>処置: SAP インスタンスが正常にインストール、設定されていて、すべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112040 | ERROR | The SAP Directory "%s" ("%s") does not exist on %s. Verify that the directory exists and that the SAP software is properly installed. | 原因: 指定したSAP インストールディレクトリが指定したサーバーに存在していない。<br><br>処置: SAP ソフトウェアが正常にインストールされすべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112041 | ERROR | The required utility "%s" was not found or was not executable on %s. Verify the SAP installation and location of the required utility. | 原因: saphostexec または saposcol が見つからないか実行できない。<br><br>処置: SAP Host Agent package が正しくインストールされ、すべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112042 | ERROR | One or more SAP or LifeKeeper validation checks has failed on %s. Please update | 原因: saphostexec または saposcol が見 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | the SAP software on this host to include the SAPHOST and SAPCONTROL packages. | つからないか実行できない。<br><br>**処置**: SAP Host Agent package が正しくインストールされ、すべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112048 | ERROR | The SAP instance %s is already under LifeKeeper protection on server %s. Choose another SAP instance to protect or specify the correct instance. | **原因**: 指定した SAP インスタンスはすでに指定したサーバーでLifeKeeper により保護されている。<br><br>**処置**: まだLifeKeeper により保護されていない SAP インスタンスを選択してください。 |
| 112049 | ERROR | Unable to locate the SAP Mount directory on %s. Verify that all SAP file systems are mounted and accessible before reattempting the operation. | **原因**: SAP のマウント (sapmnt) ディレクトリの場所を特定できない。<br><br>**処置**: すべての必要となるファイルシステムがマウントされ、すべての必要となる SAP インスタンスプロファイルがアクセス可能になっていることを確認してください。 |
| 112050 | ERROR | Detected multiple virtual IP addresses/host names for instance %s on %s. Verify that the instance is configured correctly. | **原因**: 指定したサーバー上に指定した SAP インスタンスに対する複数の仮想 IP もしくはホスト名が検出された。<br><br>**処置**: インスタンスに関連する仮想 IP もしくはホスト名が正しく構成されているかを確認してください。 |
| 112051 | ERROR | The "%s" or "%s" value in the default profile "%s" is still set to the physical host name on %s. The value(s) must be set to a virtual host name. | **原因**: 指定したサーバーで指定したホスト名のパラメーターが物理サーバーのホスト名に指定されている。<br><br>**処置**: ホスト名のパラメーターは仮想ホスト名に設定してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112053 | ERROR | Detected multiple instances under SID %s with the same instance number (%s) on %s. Each instance within a particular SID must have a unique instance number. | **原因**: 同じインスタンス番号のSAP インスタンスが、同じ SAP SID 下に検出された。<br><br>**処置**: SAP環境を再構築することで、指定した SAP SID 下の各インスタンスが一意のインスタンス番号をもつことができます。 |
| 112056 | ERROR | The NFS export for the path "%s" required by the instance %s for the "%s" directory does not have an NFS hierarchy protecting it on %s. You must create an NFS hierarchy to protect this NFS export before creating the SAP resource hierarchy. | **原因**: 指定したファイルシステムのNFS エクスポートが LifeKeeper により保護されていません。<br><br>**処置**: エクスポートしたファイルシステムの LifeKeeper NFS 階層を作成し、SAP リソースを作成してください。 |
| 112057 | ERROR | Unable to create a file system resource hierarchy for the file system “%s” on “%s”. | **原因**: 指定したサーバーで指定したファイルシステムを保護するための LifeKeeper ファイルシステムリソース階層を作成することができない。<br><br>**処置**: 詳細について LifeKeeper とシステムログを確認してください。 |
| 112058 | ERROR | Unable to create a dependency between parent tag “%s” and child tag “%s” on “%s”. | **原因**: 指定したサーバーで指定したリソース間での依存関係を作成できない。<br><br>**処置**: 詳細について LifeKeeper のログを確認してください。 |
| 112060 | ERROR | All attempts at local recovery for the SAP resource “%s” have failed on “%s”. A failover to the backup server will be attempted. | **原因**: 指定したサーバーで指定した SAP リソースを復旧できない。<br><br>**処置**: SAP リソース階層のフェイルオーバーは自動的に行われます。ユーザーの介入は必要ありません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112061 | ERROR | The values specified for the target and the template servers are the same. Please specify the correct values for the target and template servers. | **原因**: SAP リソース階層の拡張中に提供されたテンプレート/ターゲットサーバーが同一。<br><br>**処置**: テンプレート・ターゲットサーバーの正しい名前を提供し、拡張処理を再度実行してください。 |
| 112062 | ERROR | Unable to find the home directory "%s" for the SAP administrative user "%s" on %s. Verify that the SAP software is installed correctly. | **原因**: 指定したサーバーで指定した SAP ユーザーのホームディレクトリが見つからない。<br><br>**処置**: SAP ソフトウェアが正常にインストールされ、指定した SID に対する適切な SAP 管理ユーザーがサーバーに存在していることを確認してください。 |
| 112063 | ERROR | The SAP administrative user "%s" does not exist on %s. Verify that the SAP software is installed correctly or create the required SAP user on %s. | **原因**: 指定した SAP 管理ユーザーが指定したサーバー上に存在していない。<br><br>**処置**: SAP ソフトウェアが正常にインストールされているか確認し、必要に応じて SAP 管理ユーザーを作成してください。 |
| 112064 | ERROR | The group ID for user "%s" is not the same on template server "%s" and target server "%s". Please correct the group ID for the user so that it is the same on the template and target servers. | **原因**: 指定したSAP 管理ユーザーのグループ IDがテンプレート/ターゲットサーバーで一致していない。<br><br>**処置**: 指定した SAP 管理ユーザーのグループ ID を修正することで、テンプレート/ターゲットサーバーで同一となります。 |
| 112065 | ERROR | The user ID for user "%s" is not the same on template server "%s" and target server "%s". Please correct the user ID so that it is the same on the template and target servers. | **原因**: 指定した SAP 管理ユーザーの ID がテンプレート/ターゲットサーバーで一致していない。<br><br>**処置**: 指定した SAP 管理ユーザーの ID |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | を修正することで、テンプレート/ターゲットサーバーで同一となります。 |
| 112066 | ERROR | Required SAP utilities could not be found in "%s" on %s. Verify that the SAP software is installed correctly. | 原因: 必要な SAP 実行ファイルもしくは SAP インスタンスプロファイルを配置することができない。<br><br>処置: SAP ソフトウェアが正常にインストールされ構成されていて、すべての必要なファイルシステムがマウントされているか確認してください。 |
| 112069 | ERROR | The command “%s” is not found in the “%s” perl module (“%s”) on “%s”. Please check the command specified and retry the operation. | 原因: 指定したコマンドが指定したサーバーの sap perl modedule に見つからない。<br><br>処置: ユーザーから開始したコマンドアクションによりエラーとなった場合は、正しいルーチン名が remoteControl スクリプトに渡されているかを確認してください。通常のLifeKeeper 運用において発生したエラーの場合は、製品サポートへお問い合わせください。 |
| 112071 | ERROR | The file “%s” exists, but was not read and write enabled on “%s”. Enable read and write permissions on the specified file | 原因: 指定したファイルが指定したサーバーで読み込み/書き込み権限をもっていない。<br><br>処置: 指定したサーバーで読み込み/書き込み権限を有効にしてください。 |
| 112073 | ERROR | Unable to create an internal object for the SAP instance using either SID "%s" and instance "%s" or tag "%s" on server "%s". Either the values specified for the object initialization (SID/instance pair or tag, system) were not valid, or an error occurred while attempting to gather information about the SAP instance. If all specified parameters are correct, verify that all necessary SAP file systems are mounted and acces | 原因: 指定した SAP インスタンスを示すための内部 SAP オブジェクトを作成することができない。<br><br>処置: SAP インスタンスが正常にインストール、設定されていて、すべての必要なファイルシステムがマウントされているかを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | sible before reattempting the operation. | |
| 112074 | WARN | Warning: The profile file %s for SID %s and Instance %s has Autostart is enabled on %s. Disable Autostart for the specified instance by setting Autostart=0 in the profile file | 原因: Autostart パラメーターが指定したサーバーの指定したインスタンスプロファイルで有効になっている。<br><br>処置: 指定した SAP インスタンスがインスタンスプロファイルの中で 'Autostart = 0'に設定されており、Autostartが有効になっています。Autostartを無効にしてください。 |
| 112076 | FATAL | Unable to start the sapstartsrv service for SAP SID $sid and SAP instance $Instance on $me. Verify that the sapservices file is correct and the process can be started manually. | 原因: 指定した SAP インスタンスに対して SAP Start Service (sapstartsrv) のプロセスを開始できない。<br><br>処置: sapservices ファイルがsapstartsrv プロセスを開始するための適切なコマンドを含んでいて、そのプロセスを手動で開始できるかを確認してください。 |
| 112077 | ERROR | Unable to stop the sapstartsrv service for SAP SID %s and SAP instance %s on %s. Verify that the sapservices file is correct and the process can be stopped manually. | 原因: 指定した SAP インスタンスに対して SAP Start Service (sapstartsrv) プロセスを停止することができない。<br><br>処置: sapservices ファイルが sapstartsrv プロセスを開始するための適切なコマンドを含んでいて、そのプロセスを手動で開始できるかを確認してください。 |
| 112078 | ERROR | ERSv1 is only supported in two-node clusters. Resource %s is unable to be extended to system %s. Upgrade to ERSv2 in order to extend the hierarchy to three or more nodes. | 原因: ERSv1インスタンスを表す SAP リソースを 3つ以上のノードに拡張することができない。<br><br>処置: ERS インスタンスを表す SAP リソースを3つ以上のノードに拡張する場合は、ERSv2 へアップグレードしてください。 アップグレード手順は、オンラインプロダクトドキュメント内に記載されています。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112082 | WARN | Instance %s is running a different version of the enqueue server than its corresponding enqueue replication server. This configuration is not supported by SAP and will lead to unexpected resource behavior. See SAP Note 2711036 – "Usage of the Standalone Enqueue Server 2 in an HA Environment" for more details. Please review the online product documentation for instructions on how to modify the instance profiles for the enqueue server and enqueue replication server so that they use the same version. | 原因: エンキューサーバーとエンキューレプリケーションサーバーのバージョンが一致していない。<br><br>処置: オンラインプロダクトドキュメントに記載のインスタンスプロファイルの修正方法を確認の上、エンキューサーバーとエンキューレプリケーションサーバーのバージョンを一致させてください。 |
| 112086 | ERROR | The ERS resource corresponding to resource %s is in-service and maintaining backup locks on a remote system. Bringing resource %s in-service on %s would result in a loss of the backup lock table. Please bring resource %s in-service on the system where the corresponding ERS resource is currently in-service in order to maintain consistency of the lock table. In order to force resource %s in-service on %s, either (i) run the command \'/opt/LifeKeeper/bin/flg_create -f sap_cs_force_restore_%s\' as root on %s and reattempt the in-service operation or (ii) take the corresponding ERS resource out of service on the remote system. Both of these actions will result in a loss of the backup lock table. | 原因: ASCS/SCS リソースに対応する ERS インスタンスがクラスタ内の異なるサーバーで起動し、バックアップのエンキューテーブルを保存していたので、ASCS/SCS リソースに対して in-service 処理が試された。<br><br>処置: バックアップのエンキューテーブルを取得するためにERSインスタンスが起動しているサーバーで、ASCS/SCS リソースを in-sevice にしてください。ASCS/SCS リソースを強制的に in-service にする場合は、‘/opt/LifeKeeper/bin/flg_create -f sap_cs_force_restore_<ASCS/SCS Tag>’ コマンドを実行し in-service 処理を試みるか、またはリモートサーバーで対応する ERS リソースを out of service にしてください。これら両方の操作を実行すると、バックアップのエンキューテーブルが失われます。 |
| 112089 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance types TYPE_CS, TYPE_ERS, or TYPE_NEW_ERS (1, 2, or 5). | 原因: GetEnqVersion ルーチンがサポートされていない SAP インスタンスのタイプに対して呼び出された。インスタンスタイプ 1 (TYPE_CS), 2 (TYPE_ERS), 5 (TYPE_NEW_ERS) のみがサポート対象です。<br><br>処置: こちらは内部エラーです。SIOS |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | カスタマーセンターへお問い合わせください。 |
| 112092 | ERROR | The profile "%s" either does not exist or cannot be read on %s. Unable to determine whether enqueue replication is enabled for resource %s. Please verify that the file exists and can be read. | **原因**: 指定したサーバー上に指定したインスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>**処置**: ファイルが存在していて、読み取り可能かどうかを確認してください。 |
| 112095 | ERROR | Error creating resource “%s” on server “%s” | **原因**: 指定したサーバーに指定したリソースを作成することができない。<br><br>**処置**: 指定したサーバーにSAP ソフトウェアが正常にインストールされ、すべての必要なファイルシステムがマウントされアクセス可能になっていることを確認してください。 |
| 112096 | ERROR | Resource %s is not currently in-service on server %s. Manually bring the resource in-service and retry the operation. | **原因**: 依存関係の作成中に、指定したリソースがサーバーで In Service ではありませんでした。<br><br>**処置**: 指定したサーバーでリソースを In Service にし、再度操作を実施してください。 |
| 112101 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができない。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112102 | ERROR | Cannot extend resource “%s” to server “%s” | **原因**: 指定したサーバーへ拡張したリソースを作成することができない。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: SAP ソフトウェアがターゲットサーバーに正常にインストールされ、すべての必要なファイルシステムがターゲットシステムからアクセス可能になっていることを確認してください。 |
| 112103 | ERROR | Error creating resource “%s” on server “%s” | **原因**: 指定したサーバーに指定したリソースを作成することができない。<br><br>**処置**: 指定したサーバーにSAP ソフトウェアが正常にインストールされ、すべての必要なファイルシステムがマウントされアクセス可能になっていることを確認してください。 |
| 112104 | ERROR | The extend script "%s" either does not exist or is not executable on %s. | **原因**: 指定したサーバーに指定した extend スクリプトが存在しないか実行することができない。<br><br>**処置**: すべての必要なリカバリーキットがインストールされ、extend スクリプトが実行可能か確認してください。 |
| 112106 | ERROR | Unable to create an internal SAP object for resource "%s" on %s. If the tag is correct, verify that all necessary SAP file systems are mounted and accessible. | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112112 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 確認してください。 |
| 112115 | FATAL | Error getting resource information for resource \”$tag\” on server \”$sap::me\”. | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112119 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112123 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112124 | ERROR | Resource with either matching tag “%s” or id “%s” already exists on server “%s” for App “%s” and Type “%s” | **原因**: 指定したリソースタグ名もしくはID および同一のアプリケーション、タイプをもつリソースがすでに指定したサーバーに存在しています。<br><br>**処置**: SAP インスタンスが指定したサーバーでまだLifeKeeper に保護されていないことを確認してください。まだ保護されていない場合は、別のリソースタグ名を選択してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112125 | ERROR | Unable to create an SAP object for resource %s on system %s. | **原因**: 指定のサーバーで指定されたリソースを示すための SAP オブジェクトを作成することができない。<br><br>**処置**: 指定したサーバーにSAP ソフトウェアが正常にインストールされ、すべての必要なファイルシステムがマウントされアクセス可能になっていることを確認してください。 |
| 112126 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | **原因**: SAP canextend スクリプトの使用に誤りがあります。<br><br>**処置**: canextend スクリプトに対する正しいパラメーターを指定してください。: canextend <template server> <template tag> |
| 112127 | FATAL | Usage: \”$cmd $usage\” Specify the correct usage for the requested command. | **原因**: SAP delete スクリプトの使用に誤りがある。<br><br>**処置**: SAP delete スクリプトに対する正しいパラメーターを指定してください。: delete [-U] -t <tag> -i <id> |
| 112128 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | **原因**: SAP depstoextend スクリプトの使用に誤りがある。<br><br>**処置**: depstoextend スクリプトに対する正しいパラメーターを指定してください。: depstoextend <template server> <template tag> |
| 112129 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | **原因**: SAP extend スクリプトの使用に誤りがある。<br><br>**処置**: extend スクリプトに対する正しいパラメーターを指定してください。: extend <template server> <template tag> |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | <switchback> <target tag> |
| 112130 | FATAL | Usage: \"$cmd $usage\" Specify the correct usage for the requested command. | 原因: SAP quickCheck スクリプトの使用に誤りがある。<br><br>処置: quickCheck スクリプトに対する正しいパラメーターを指定してください。: quickCheck -t <tag> -i <id> |
| 112131 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | 原因: SAP recover スクリプトの使用に誤りがある。<br><br>処置: recover スクリプトに対する正しいパラメーターを指定してください。: recover -d <tag> |
| 112132 | FATAL | Usage: \"$cmd $usage\". Specify the correct usage for the requested command. | 原因: SAP remoteControl スクリプトの使用に誤りがある。<br><br>処置: remoteControl スクリプトに対する正しいパラメーターを指定してください。: remoteControl <tag> <remote instance> <remote cmd> <remote cmd option> <primary system> <primary tag> |
| 112133 | FATAL | Usage: “%s %s”. Specify the correct usage for the requested command. | 原因: SAP remove スクリプトの使用に誤りがある。<br><br>処置: remove スクリプトに対する正しいパラメーターを指定してください。: remove -t <tag> -i <id> |
| 112134 | FATAL | Usage: \"$cmd $usage\". Specify the correct usage for the requested command. | 原因: SAP restore スクリプトの使用に誤りがある。<br><br>処置: restore スクリプトに対する正しいパラメーターを指定してください。: restore -t <tag> -i <id> |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112137 | ERROR | The required parameter \"parent\" was either not provided or was invalid in the $func routine on $me. | 原因: CleanUpルーチンの使用に誤りがある。<br><br>処置: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112138 | ERROR | At least one required process for instance %s was not started successfully during "%s" on server %s. Please check the LifeKeeper and system logs for additional information. | 原因: 指定したサーバーで、指定した SAP インスタンスに対する少なくとも1つのプロセスが正常に開始されませんでした。<br><br>処置: LifeKeeper/システムログ, SAP トレースファイルなどから原因を調査し、操作を再度行ってください。 |
| 112140 | FATAL | The tag parameter was not specified for the internal \"$func\" routine on $me. If this was a command line operation, specify the correct parameters. Otherwise, consult the troubleshooting documentation. | 原因: 指定したサーバーでタグパラメーターが GetLK ルーチンに指定されていません。<br><br>処置: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112141 | ERROR | Either the SID ("%s") or instance ("%s") parameter was not specified for the "%s" routine on %s. | 原因: SID もしくは SID のどちらかが指定したサーバーの StatusSapServer ルーチン に指定されていません。<br><br>処置: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112142 | ERROR | Either the SID ("%s"), instance ("%s"), or instance number ("%s") parameter was not specified for the "%s" routine on %s. | 原因: SID、インスタンス、インスタンス番号のいずれかが指定のサーバーの StartSapServer ルーチンに提供されていない。<br><br>処置: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせくだ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | さい。 |
| 112143 | ERROR | The SID, instance, or instance number parameter was not specified for the "%s" routine on %s. | **原因**: SID、インスタンス、インスタンス番号のいずれかが指定のサーバーの StopSapServer ルーチンで指定されていない。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112173 | ERROR | The file “%s” does not exist or was not readable on “%s”. Verify that the specified file exists and/or is readable. | **原因**: 指定したファイルが指定したサーバーに存在しないか、読み込み不可になっている。<br><br>**処置**: ファイルが存在しているかを確認および/もしくは権限を修正して読み込み可能な状態にしてください。 |
| 112174 | ERROR | The file “%s” does not exist or was not readable on “%s”. Verify that the specified file exists and/or is readable. | **原因**: 指定したファイルが指定したサーバーに存在しないか、読み込み不可になっています。<br><br>**処置**: ファイルが存在しているかを確認および/もしくは権限を修正して読み込み可能な状態にしてください。 |
| 112175 | ERROR | The file “%s” does not exist or was not readable on “%s”. Verify that the specified file exists and/or is readable. | **原因**: 指定したファイルが指定したサーバーに存在しないか、読み込み不可になっている。<br><br>**処置**: ファイルが存在しているかを確認および/もしくは権限を修正して読み込み可能な状態にしてください。 |
| 112194 | ERROR | There was an error verifying the NFS connections for SAP related mount points on “%s”. One or more NFS servers is not operational and needs to be restarted. | **原因**: マウントポイントが /etc/default/LifeKeeper の SAP_NFS_CHECK_DIRS エントリに記述されている少なくとも一 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | つの 重要なNFS 共有システムが現在無効となっている。<br><br>**処置**: すべての必要な NFS 共有ファイルシステムがアクセス可能であることを確認し、現在操作不可となっている NFS サーバーを再起動してください。 |
| 112195 | FATAL | There was an error verifying the NFS connections for SAP related mount points on $me. One or more NFS servers is not operational and needs to be restarted. | **原因**: マウントポイントが /etc/default/LifeKeeper の SAP_NFS_CHECK_DIRS エントリに記述されている少なくとも一つの 重要なNFS 共有システムが現在無効となっている。<br><br>**処置**: すべての必要な NFS 共有ファイルシステムがアクセス可能であることを確認し、現在操作不可となっている NFS サーバーを再起動してください。 |
| 112196 | WARN | There was an error verifying the NFS connections for SAP related mount points on “%s”. One or more NFS servers is not operational and needs to be restarted. | **原因**: マウントポイントが /etc/default/LifeKeeper の SAP_NFS_CHECK_DIRS エントリに記述されている少なくとも一つの 重要なNFS 共有システムが現在無効となっている。<br><br>**処置**: すべての必要な NFS 共有ファイルシステムがアクセス可能であることを確認し、現在操作不可となっている NFS サーバーを再起動してください。 |
| 112201 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. | **原因**: リソースタグの引数が GetLKEquivルーチンに提供されていません。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112203 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. Additional information available in the LifeKeeper and system logs. | **原因**: SAP インスタンス番号が IsInstanceRunning ルーチンに提供されていませ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ん。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112204 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. | **原因**: saphostexec もしくは saposcol コマンドのどちらを使用するか判断できない。<br><br>**処置**: SAP kernel 7.3より前のSAP Net Weave に対して SAP_SRVHOST_CMD, SAP_HOSTCTL_CMD, SAP_OSCOL_CMD を使用している場合は、チューニングが正しく設定されていることを確認してください。 |
| 112205 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. | **原因**: SAP インスタンスが 指定したサーバーのSAPRemExec ルーチンに提供されなかった。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112206 | ERROR | The required action parameter was not provided. Unable to complete "%s" on %s. | **原因**: 指定したサーバーで SAPRemExec ルーチンに対するアクションが提供されませんでした。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112208 | ERROR | Error getting the value of "%s" or "%s" from the default profile "%s" on %s. Verify that the specified value exists. | **原因**: 指定したサーバーで提供されたプロファイルから仮想IPもしくはホスト名を取得できない。<br><br>**処置**: プロファイルに適切なエントリーが存在していることを確認してくださ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | い。 |
| 112209 | FATAL | Unable to gather required information fro m the SAP default profile for SID $sid ($D PFL) on $me. Verify that the default profil e exists and is accessible. | **原因**: 指定したサーバーで提供されたプロファイルから SAP インスタンスに関する情報を取得できない。<br><br>**処置**: SAP ソフトウェアが正常にインストールされ提供されたプロファイルが存在および読み込み可能となっていることを確認してください。 |
| 112214 | ERROR | Unable to determine the status of the path "%s" ("%s") on %s. The path on %s may r equire the execution of the command: "m ount <ip>:<export> %s". Verify that the S AP software is correctly installed and that all SAP file systems are mounted and acc essible. | **原因**: 指定したパスのファイルシステムのステータスを判断できません。<br><br>**処置**: SAP ソフトウェアが正常にインストールされすべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112219 | ERROR | [HACONNECTOR:%s] Unable to write to file "%s" on %s. If the file already exists, manually enable write permissions on it. | **原因**: 指定したファイルが指定したサーバーで読み込み/書き込み権限をもっていない。<br><br>**処置**: 指定したサーバーの読み込み/書き込み権限を有効にしてください。 |
| 112220 | ERROR | Unable to start the sapstartsrv service for SID %s and SAP instance %s on %s. Veri fy that the sapservices file is correct and t he process can be started manually. | **原因**: 指定した SAP インスタンスに対する SAP Start Service (sapstartsrv) プロセスを開始できない。<br><br>**処置**: sapservices ファイルがsapstartsrv プロセスを開始するための適切なコマンドを含んでいて、そのプロセスを手動で開始できるかを確認してください。 |
| 112221 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP insta nce types TYPE_CS, TYPE_ERS, or TYP | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出され |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | E_NEW_ERS (1, 2, or 5). | た。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112222 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance types TYPE_CS, TYPE_ERS, or TYPE_NEW_ERS (1, 2, or 5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112223 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_CS (1). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112224 | ERROR | The internal \"%s\" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_CS (1). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112225 | ERROR | The internal \"%s\" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_NEW_ERS (5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOS |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | カスタマーセンターへお問い合わせください。 |
| 112226 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance types TYPE_CS, TYPE_ERS, and TYPE_NEW_ERS (1, 2, and 5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112227 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This routine only supports SAP instance type TYPE_CS (1). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112228 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_NEW_ERS (5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112229 | ERROR | The profile \"%s\" either does not exist or cannot be read on %s. Unable to determine whether enqueue replication is enabled for resource %s. Please verify that the file exists and can be read. | **原因**: 指定したサーバー上に指定したインスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>**処置**: ファイルが存在していて、読み取り可能かどうかを確認してください。 |
| 112325 | ERROR | The $SAP_CONTROL utility cannot be located. Verify that all necessary file system | **原因**: SAP インスタンスの管理に必要な |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | s are mounted and that the $SAP_CONTROL utility can be located in the $sapadmin user's PATH. | sapcontrol ユーティリティが配置できない。<br><br>**処置**: 全ての必要なファイルシステムがマウントされ、sapcontrol ユーティリティがSAP 管理ユーザーの Path に配置されていることを確認してください。 |
| 112326 | ERROR | The \"which $SAP_CONTROL\" command for user $sapadmin returned $sapcmd as the location of the $SAP_CONTROL utility, but the utility could not be found or was not executable in this location. Verify that all necessary file systems are mounted and that the $SAP_CONTROL utility can be located in the $sapadmin user's PATH. | **原因**: SAP インスタンスの管理に必要な sapcontrol ユーティリティが配置できない。<br><br>**処置**: 全ての必要なファイルシステムがマウントされ、sapcontrol ユーティリティがSAP 管理ユーザーのパスに配置されていることを確認してください。 |
| 112433 | ERROR | Unsupported SAPENQ_VERSION ($enqversion) for resource $tag on $me. Unable to obtain enqueue replication status. | **原因**: 指定したサーバーで指定したリソースのSAPENQ_VERSIONパラメーターにサポートされていない値が検知された。<br><br>**処置**: 指定したリソースの info ファイルで、SAPENQ_VERSION に有効な値 (現在使用しているエンキューサーバーのバージョンを表す1 または 2) が設定されていることを確認してください。 |
| 112437 | ERROR | Profile \"$srvpf\" not found on $me. Unable to obtain enqueue replication status for instance $inst. | **原因**: 指定したサーバー上に指定したインスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>**処置**: ファイルが存在していて、読み取り可能かどうかを確認してください。 |
| 112438 | ERROR | Unsupported SAPENQ_VERSION ($enqversion) for resource $tag on $me. Unable to obtain enqueue replication status for instance $inst. | **原因**: 指定したサーバーで指定したリソースのSAPENQ_VERSION パラメーターにサポートされていない値が検知された。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 指定したリソースの info ファイルで、SAPENQ_VERSION に有効な値 (現在使用しているエンキューサーバーのバージョンを表す1 または 2) が設定されていることを確認してください。 |
| 112470 | ERROR | The instance profile %s could not be found on %s. Verify that the SAP software is installed correctly and that all necessary file systems are mounted and accessible. | **原因**: 指定したサーバー上に指定したインスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>**処置**: ファイルが存在していて、読み取り可能かどうかを確認してください。 |
| 112490 | ERROR | [HACONNECTOR] Unable to determine the corresponding tag for resource with ID \"$res\" on $me. | **原因**: 指定したサーバー上に指定したリソースID をもつ LifeKeeper SAP リソースを見つけることができない。<br><br>**処置**: fra コマンドの —res 引数として提供されるソース ID が有効な LifeKeeper SAP リソースに対応しているか確認してください。 |
| 112507 | ERROR | [HACONNECTOR] At least one required process for the instance was not killed successfully during the fra migrate action on $me. Aborting resource migration. | **原因**: fra マイグレーション実行中に SAP インスタンスが指定したサーバーで正常に停止しなかった。<br><br>**処置**: SAP インスタンスに対して起動中のプロセスを手動で停止し、マイグレーションを再度実行してください。 |
| 112539 | ERROR | [HACONNECTOR] Unable to find rpm information for one or more packages on $me. | **原因**: HA Connector gvi (“Get Version Information”)ルーチンは、LifeKeeper および/または the SAP Recovery Kit のバージョン番号を判別することができなかった。<br><br>**処置**: rpm -q コマンドでLifeKeeper Core および SAP Recovery Kit の rpm 情報を取得できるかを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112976 | ERROR | There is no LifeKeeper protected resource with tag $tag on system $me. | 原因: SAP canfailover スクリプトに渡されたリソースタグに対応する LifeKeeper リソースがない。<br><br>処置: リソースタグ名が正しいことを確認し再度コマンドを実行してください。 |
| 112977 | ERROR | Resource $tag is not a $app/$typ resource. Please use the $ins_app/$ins_typ resource-specific canfailover script instead. | 原因: SAP canfailover スクリプトに渡された LifeKeeper リソースが、SAP（app suite/sap）リソースではない。<br><br>処置: 指定したリソースに対して適切なタイプ固有の canfailover スクリプトを使用してください。 |
| 122005 | ERROR | Unable to "%s" on "%s" | 原因:「getlocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122007 | ERROR | Unable to "%s" on "%s" | 原因:「rlslocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122009 | ERROR | The path %s is not a valid file. | 原因: listener.ora ファイルがありません。<br><br>処置: ファイルが存在していることを確認し、操作を再試行してください。 |
| 122010 | ERROR | The listener user does not exist on the server %s. | 原因:「Stat」コマンドでユーザ ID を取得できませんでした。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 操作を再試行してください。 |
| 122011 | ERROR | The listener user does not exist on the server %s. | **原因**: UID が passwd ファイルにありません。<br><br>**処置**: UID が passwd ファイルに存在していることを確認し、操作を再試行してください。 |
| 122012 | ERROR | The listener user does not exist on the server %s. | **原因**:ユーザ名が passwd ファイルにありません。<br><br>**処置**: ユーザ名が passwd ファイルに存在していることを確認し、操作を再試行してください。 |
| 122023 | ERROR | The %s command failed (%d) | **原因**: このメッセージに「lsnrctl」コマンドの戻りコードが含まれています。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122024 | ERROR | $line | **原因**: メッセージに「lsnrctl」コマンドの結果が含まれています。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122039 | ERROR | Usage error | **原因**: リストア操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122040 | ERROR | Script $cmd has hung on the restore of \"$opt_t\". Forcibly terminating. | **原因**: リスナーの restore スクリプトがタイムアウト値に達しました。<br><br>**処置**: listener.ora が有効であること、および /etc/default/LifeKeeper の LSNR_START_TIME（デフォルトでは 35 秒）がリスナーの起動に必要な時間以上の値に設定されていることを確認してください。 |
| 122041 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**: LifeKeeper は、{server} でリソース {resource} をリストアできませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122045 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: リソース情報の取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122046 | ERROR | Usage error | **原因**: リストア操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122049 | ERROR | The script $cmd has hung on remove of \"$opt_t\".  Forcibly terminating. | **原因**: リスナーの remove スクリプトがタイムアウト値に達しました。<br><br>**処置**: listener.ora が有効であること、および /etc/default/LifeKeeper の LSNR_STOP_TIME (デフォルトでは 35 秒) がリスナーの停止に必要な時間以上の値に設定されていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122051 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122055 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: LifeKeeper は、{server} でリソース {resource} に対して quickCheck を実行できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122057 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122064 | WARN | The %s level is set to %s a %s will not occur. | 原因: リスナーの最低限の保護レベルは起動と監視です。<br><br>処置: リスナーを手動で起動してください。 |
| 122066 | ERROR | Script has hung checking \"$tag\". Forcibly terminating. | 原因: リスナーの quickCheck スクリプトがタイムアウト値に達しました。<br><br>処置: listener.ora が有効であること、および /etc/default/LifeKeeper の LSNR_STATUS_TIME (デフォルトでは 15 秒) がリスナーの確認に必要な時間以上の値に設定されていることを確認してください。 |
| 122067 | ERROR | Usage error | 原因: quickCheck 操作で無効なパラメータが指定されました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: パラメータを確認し、操作を再試行してください。 |
| 122069 | ERROR | Usage error | **原因**: 削除操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122072 | ERROR | %s: resource "%s" not found on local server | **原因**: 復旧操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122074 | WARN | The local recovery attempt has failed but %s level is set to %s preventing a failover to another node in the cluster. With %s recovery set all local recovery failures will exit successfully to prevent resource failovers. | **原因**: オプションのリスナーリカバリレベルがローカルリカバリにのみ設定されています。<br><br>**処置**: リソースツリーを手動でスイッチオーバしてください。 |
| 122078 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**:LifeKeeper は、{server} でリソース {resource} を復旧できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122082 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122083 | ERROR | $cmd has hung checking \"$tag\". Forcibly terminating | **原因**: recover スクリプトがシグナルによって停止されました。<br><br>**処置**: listener.ora が有効であることを確認してください。 |
| 122084 | ERROR | Cannot extend resource "%s" to server "%s" | **原因**: LifeKeeper は、{server} でリソース {resource} を拡張できませんでした。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122085 | ERROR | Usage: %s %s | **原因**: canextend 操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122086 | ERROR | The values specified for the target and the template servers are the same. Please specify the correct values for the target and template servers. | **原因**: ターゲットサーバに指定した値とテンプレートサーバに指定した値が同じです。<br><br>**処置**: メッセージテキストに記載されている手順を実行してください。 |
| 122087 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122088 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: リソース情報からのリスナーユーザ名の取得に失敗しました。<br><br>**処置**: リソース情報フィールドが有効であることを確認し、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122089 | ERROR | The listener user %s does not exist on the server %s. | **原因**: ユーザ名が passwd ファイルにありません。<br><br>**処置**: ユーザ名が passwd ファイルに存在していることを確認し、操作を再試行してください。 |
| 122090 | ERROR | The id for user %s is not the same on template server %s and target server %s. | **原因**: ユーザ ID はどちらのサーバでも同じものにする必要があります。<br><br>**処置**: ユーザ ID を同じものにしてください。 |
| 122091 | ERROR | The group id for user %s is not the same on template server %s and target server %s. | **原因**: グループ ID はどちらのサーバでも同じものにする必要があります。<br><br>**処置**: グループ ID を同じものにしてください。 |
| 122092 | ERROR | Cannot access canextend script "%s" on server "%s" | **原因**: {server} で「canextend」スクリプトを見つけられなかったので、LifeKeeper は拡張前のチェックを実行できませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122097 | ERROR | Usage: %s %s | **原因**:「configActions」操作で無効な引数が指定されました。<br><br>**処置**: 引数を確認し、操作を再試行してください。 |
| 122098 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122099 | ERROR | Unable to update the resource %s to change the %s to %s on %s. | 原因: LifeKeeper は情報フィールドに情報を入力できませんでした。<br><br>処置: LifeKeeper を再起動し、操作を再試行してください。 |
| 122100 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122101 | ERROR | Unable to update the resource %s to change the %s to %s on %s. | 原因: LifeKeeper は {server} で情報フィールドに情報を入力できませんでした。<br><br>処置: {server} で LifeKeeper を再起動し、操作を再試行してください。 |
| 122103 | ERROR | Usage: %s %s | 原因: 作成操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122124 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122126 | ERROR | Unable to "%s" on "%s | 原因:「rlslocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122127 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122129 | ERROR | Unable to "%s" on "%s | 原因:「getlocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122131 | ERROR | Error creating resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122133 | ERROR | Unable to create a file system resource hierarchy for the file system %s. | 原因:「filesyshier」の実行中に予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細を確認してください。 |
| 122135 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122140 | ERROR | Resource "%s" is not ISP on server "%s" Manually bring the resource in service and retry the operation | 原因: リスナーリソースが依存する IP リソース {tag} は ISP の必要があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: メッセージテキストに記載されている手順を実行してください。 |
| 122141 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | **原因**:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122144 | ERROR | Usage: %s %s | **原因**:「create_ins」操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122145 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | **原因**:「app_create」の実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122146 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | **原因**:「typ_create」の実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122147 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | **原因**:「newtag」の実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122148 | ERROR | Error creating resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122149 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | 原因:「ins_setstate」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122150 | ERROR | Error creating resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122151 | ERROR | Usage: %s %s | 原因:「depstoextend」操作で無効な引数が指定されました。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122152 | ERROR | Usage: %s %s | 原因:「extend」操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122153 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122154 | ERROR | Cannot extend resource "%s" to server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を拡張できませんでした。 |
| 122155 | ERROR | Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s" | 原因: リスナーリソースの拡張中に、同じ {tag} または {id} で異なるリソースアプリケーションおよびタイプを使用しているリソースインスタンスが見つかりました。<br><br>処置: リソース ID は一意のものにする必要があります。Oracle Listener リソースインスタンスに合致する ID のリソースインスタンスを削除してください。 |
| 122156 | ERROR | Cannot access extend script "%s" on server "%s" | 原因: LifeKeeper は、{server} でスクリプト EXTEND が見つからなかったので、リソース階層を拡張できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122157 | ERROR | Usage: %s %s | 原因:「getConfigIps」操作で無効な引数が指定されました。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122158 | ERROR | The file %s is not a valid listener file.  The file does not contain any listener definitions. | 原因: リスナー定義が見つかりませんでした。<br><br>処置: リスナー定義が listener.ora にあることを確認し、操作を再試行してください。 |
| 122159 | ERROR | Usage: %s %s | 原因:「getSidListeners」操作で無効な引数が指定されました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 引数を確認し、操作を再試行してください。 |
| 122160 | ERROR | The file %s is not a valid listener file. The file does not contain any listener definitions. | **原因**: リスナー定義が見つかりませんでした。<br><br>**処置**: リスナー定義が listener.ora にあることを確認し、操作を再試行してください。 |
| 122161 | ERROR | Usage: %s %s | **原因**:「lsn-display」操作で無効な引数が指定されました。<br><br>**処置**: 引数を確認し、操作を再試行してください。 |
| 122162 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122163 | ERROR | Usage: %s %s | **原因**: updateHelper 操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122164 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | **原因**: LifeKeeper は、{server} でリソース {resource} を更新できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122166 | ERROR | Usage: %s %s | 原因:「updateHelper」操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122170 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122171 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122172 | ERROR | Usage: %s %s | 原因:「updIPDeps」操作で無効な引数が指定されました。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122173 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | 原因: LifeKeeper は、{server} でリソース {resource} を更新できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122175 | ERROR | Unable to "%s" on "%s | 原因:「rlslocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | し、レポートされている問題を解決してください。 |
| 122177 | ERROR | Unable to "%s" on "%s | 原因:「getlocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122180 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122181 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122183 | ERROR | The path %s is not a valid file. | 原因: listener.ora ファイルがありません。<br><br>処置: ファイルが存在していることを確認し、操作を再試行してください。 |
| 122185 | ERROR | The file %s is not a valid listener file. The file does not contain any listener definitions. | 原因: LifeKeeper は有効なリスナー定義の検索に失敗しました。<br><br>処置: 有効なリスナー定義が listener.ora にあることを確認し、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122186 | ERROR | The value specified for %s cannot be empty. Please specify a value for this field. | 原因:設定ファイルまたは実行可能ファイルの {path} フィールドが空です。<br><br>処置: 空でない値を {path} に入力し、操作を再試行してください。 |
| 122187 | ERROR | The path %s is not a valid file or directory. | 原因: 定義済み {path} が無効です。<br><br>処置: {path} が存在していることを確認し、操作を再試行してください。 |
| 122188 | ERROR | The path %s is not a valid file or directory. | 原因: {path} がありません。<br><br>処置: {path} が存在していることを確認し、操作を再試行してください。 |
| 122189 | ERROR | The value specified for %s cannot be empty. Please specify a value for this field. | 原因: 設定および実行パスフィールドが空です。<br><br>処置: フィールドのパスを入力してください。 |
| 122190 | ERROR | Usage: %s %s | 原因:「valid_rpath」操作で無効な引数が指定されました。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122191 | ERROR | The values specified for the target and the template servers are the same. | 原因: valid_rpath の無効な引数。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122192 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあるこ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | と、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122193 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122194 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122195 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122196 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因:LifeKeeper は、{server} でリソース {resource} を削除できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122197 | ERROR | Unable to find the configuration file \"oratab\" in its default locations, /etc/oratab or $listener::oraTab on \"$me\" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122198 | ERROR | remove for $okListener failed. | |
| 122251 | ERROR | Update of pluggable database info field for "%s" on "%s" failed (%s). | |
| 122252 | ERROR | Initial connect with query buffer to database "%s" on "%s" failed, testing output. | |
| 122253 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | |
| 122261 | ERROR | The Oracle resource (%s) and dependency are not set on %s. | |
| 122262 | ERROR | Usage: %s %s | |
| 122263 | ERROR | The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper. | |
| 122264 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | |
| 122268 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122269 | ERROR | no dependency for Oracle on "%s". | |
| 122270 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122271 | ERROR | Usage: %s %s | |
| 122272 | ERROR | The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper. | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122273 | ERROR | END failed %s of "%s" on server "%s" du e to a "%s" signal | |
| 122277 | ERROR | Usage: %s %s | |
| 122278 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122279 | ERROR | no dependency for Oracle on "%s". | |
| 122280 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122281 | ERROR | The quickCheck of %s has timed out on s erver %s. The default TIMEOUT is 45 sec onds. To increase the TIMEOUT, set OR ACLE_QUICKCHECK_TIMEOUT in /etc/ default/LifeKeeper. | |
| 122282 | ERROR | END failed %s of "%s" on server "%s" du e to a "%s" signal | |
| 122284 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122285 | ERROR | no dependency for Oracle on "%s". | |
| 122287 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122288 | ERROR | Usage: %s %s | |
| 122291 | ERROR | Cannot extend resource "%s" to server "%s" | |
| 122292 | ERROR | The values specified for the target and th e template servers are the same: "%s". | |
| 122294 | ERROR | Cannot access canextend script "%s" on server "%s" | |
| 122295 | ERROR | Usage: %s %s | |
| 122296 | ERROR | DB instance "%s" is not protected on "%s". | |
| 122297 | ERROR | Failed to create object instance for Oracle PDB on "%s". | |
| 122298 | ERROR | Unable to locate the oratab file "%s" on "%s". | |
| 122299 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | %d | |
| 122301 | ERROR | Unable to "%s" on "%s" during resource create. | |
| 122302 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | |
| 122304 | ERROR | Unable to "%s" on "%s" during resource create. | |
| 122305 | ERROR | Unable to determine Oracle user for "%s" on "%s". | |
| 122306 | ERROR | Error creating resource "%s" on server "%s" | |
| 122308 | ERROR | Dependency creation between Oracle pluggable database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason | |
| 122309 | ERROR | %s | |
| 122311 | ERROR | In-service attempted failed for tag "%s" on "%s". | |
| 122312 | ERROR | Usage: %s %s | |
| 122313 | ERROR | Create of app "%s" on "%s" failed with return code of "%d". | |
| 122314 | ERROR | Create of typ "%s" for app "%s" on "%s" failed with return code of "%d". | |
| 122316 | ERROR | Create of resource tag via "newtag" on "%s" failed. | |
| 122318 | ERROR | Error creating resource "%s" on server "%s" | |
| 122320 | ERROR | Setting "resstate" for resource "%s" on "%s" failed with return code of "%d". | |
| 122321 | ERROR | Error creating resource "%s" on server "%s" | |
| 122322 | ERROR | Usage: %s %s | |
| 122323 | ERROR | Usage: %s %s | |
| 122324 | ERROR | Usage: %s %s | |
| 122325 | ERROR | Cannot extend resource "%s" to server | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | "%s" | |
| 122326 | ERROR | Resource with either matching tag "%s" o r id "%s" already exists on server "%s" for App "%s" and Type "%s" | |
| 122327 | ERROR | Error creating resource "%s" on server "%s" | |
| 122328 | ERROR | Cannot access extend script "%s" on serv er "%s" | |
| 122329 | ERROR | Cannot extend resource "%s" to server "%s" | |
| 122330 | ERROR | Usage: %s %s | |
| 122331 | ERROR | Failed to create object instance for Oracle PDB on "%s". | |
| 122332 | ERROR | Usage: %s %s | |
| 122334 | ERROR | Backup node %s is unreachable; abort pr otection PDB changes. | |
| 122336 | ERROR | Update of protection PDB failed for "%s" on "%s". | |
| 122339 | ERROR | Usage: %s %s | |
| 122340 | ERROR | Usage: %s %s | |
| 122341 | ERROR | Usage: %s %s | |
| 122342 | ERROR | Failed to create object instance for Oracle PDB on "%s". | |
| 122343 | ERROR | Usage: %s %s | |
| 122344 | ERROR | END failed %s of "%s" on server "%s" du e to a "%s" signal | |
| 122348 | ERROR | Failed to create flag "%s" on "%s". | |
| 122350 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122351 | ERROR | no dependency for Oracle on "%s". | |
| 122353 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122356 | ERROR | Usage: %s %s | |
| 122357 | ERROR | Failed to create object instance for Oracle | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | PDB on "%s". | |
| 122358 | ERROR | The selected oracle SID "%s" is not a CDB. | |
| 122359 | ERROR | No protectable PDB found for the selected SID "%s". | |
| 122360 | ERROR | No protected Oracle database found on "%s". | |
| 122500 | ERROR | Usage: %s %s | 原因: 作成操作で無効なパラメーターが指定されました。<br><br>処置: パラメーターが正しいことを確認し、操作を再実行してください。 |
| 122501 | ERROR | DB instance "%s" is already protected on "%s". | 原因: {server} で LifeKeeper によってすでに保護されている Oracle データベースインスタンス {sid} を保護しようとしました。<br><br>処置: 別のデータベースインスタンス {sid} を LifeKeeper の保護用に選択する必要があります。 |
| 122502 | ERROR | Failed to create object instance for Oracle on "%s". | 原因: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122503 | ERROR | Unable to locate the oratab file "%s" on "%s". | 原因: {server} のデフォルトまたは代替ロケーションで oratab ファイルが見つかりませんでした。<br><br>処置: oratab ファイルが存在しており、Oracle ユーザの適切な権限を持っていることを確認してください。「creat |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | e」操作を完了するには有効な oratab ファイルが必要です。 |
| 122504 | ERROR | Unable to determine Oracle user for "%s" on "%s". | **原因**: Oracle Application Recovery Kit は Oracle データベースインストールバイナリの所有権を特定できませんでした。<br><br>**処置**: Oracle バイナリの所有者は {server} の有効な非 root ユーザである必要があります。Oracle データベース所有権の権限と所有権を修正し、操作を再試行してください。 |
| 122505 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | **原因**: データベースインスタンス {sid} が実行されていません。またはデータベースへの接続が、指定された認証情報では使用できませんでした。<br><br>**処置**:「create」操作を完了するには、データベースインスタンス {sid} を {server} で起動し、正しい認証情報を指定する必要があります。 |
| 122506 | ERROR | Unable to determine Oracle dbspaces and logfiles for "%s" on "%s". | **原因**: 必要なテーブルスペース、ログファイル、関連データベースファイルの場所を特定するクエリーが失敗しました。これは、内部データベースエラーが原因の可能性もあります。<br><br>**処置**: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122507 | ERROR | Unknown chunk type found for "%s" on "%s". | **原因**: 指定したテーブルスペース、ログファイル、または必要なデータベースファイルが LifeKeeper でサポートされて |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | いるファイルまたはキャラクタデバイスタイプではありません。<br><br>**処置**: 指定したファイル {database_file} は既存のキャラクタデバイスまたはファイルを参照する必要があります。Oracle インストールドキュメントを参照し、指定したファイル {database_file} を、サポートされているファイルまたはキャラクタデバイスタイプとして再作成してください。 |
| 122508 | ERROR | DB Chunk "%s" for "%s" on "%s" does not reside on a shared file system. | **原因**: 指定したテーブルスペース、ログファイル、またはその他の必要なデータベースファイル {database_file} は、クラスタ内の他のシステムと共有されているファイルシステムに存在します。<br><br>**処置**: コミュニケーションパスが正常に作成されたことを確認するには、LifeKeeper UI または「lcdstatus (1M)」を使用してください。ストレージ保護に必要なアプリケーションリカバリキットがインストールされたことを確認するには、「rpm」を使用してください。ファイルが実際には共有ストレージにないことを確認し、ない場合は、共有ストレージデバイスに移動してください。 |
| 122510 | ERROR | File system create failed for "%s" on "%s". Reason: | **原因**: LifeKeeper は、指定したサーバ {server} でリソース {filesystem} を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから「create」操作を再試行する必要があります。 |
| 122511 | ERROR | %s | **原因**: メッセージに「filesyshier」コマンド |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | の結果が含まれています。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122513 | ERROR | Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed.  Reason: | **原因**: LifeKeeper は、データベースリソース {tag} と必要な子リソース {childtag} との間に依存関係を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。問題が修正されている場合は、{tag} と {childtag} との間の依存関係を手動で作成できる可能性があります。 |
| 122514 | ERROR | Unable to "%s" on "%s" during resource create. | **原因**: Oracle Application Recovery Kit は、「rlslocks」コマンドで管理ロックを解除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122516 | ERROR | Raw device resource created failed for "%s" on "%s".  Reason: | **原因**: LifeKeeper は、指定したサーバ {server} でリソース {raw device} を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから「create」操作を再試行する必要があります。 |
| 122519 | ERROR | In-service attempted failed for tag "%s" on "%s". | **原因**: {server} の {tag} に対する「perform_action」コマンドがデータベース {sid} の開始に失敗しました。in service 操作に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから「create」操作を再試行する必要があります。 |
| 122521 | ERROR | Create of app "%s" on "%s" failed with return code of "%d". | **原因**: 内部アプリケーションタイプを作成する「app_create」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122522 | ERROR | Create of typ "%s" for app "%s" on "%s" failed with return code of "%d". | **原因**: 内部リソースタイプを作成する「typ_create」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122524 | ERROR | Setting "resstate" for resource "%s" on "%s" failed with return code of "%d". | **原因**: リソースの状態を {state} に設定する「ins_setstate」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122525 | ERROR | The values specified for the target and the template servers are the same: "%s". | **原因**:「extend」操作でターゲットサーバとテンプレートサーバで指定した値が同じでした。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: {target server} と {template server} に対して正しいパラメータを指定する必要があります。{target server} は、{tag} が拡張されるサーバです。 |
| 122526 | ERROR | Unable to locate the oratab file in "/etc" or in "%s" on "%s". | **原因**: {server} のデフォルトまたは代替ロケーションで oratab ファイルが見つかりませんでした。<br><br>**処置**: oratab ファイルが存在しており、Oracle ユーザの適切な権限を持っていることを確認してください。「extend」操作を完了するには有効な oratab ファイルが必要です。 |
| 122527 | ERROR | Unable to retrieve the Oracle user on "%s". | **原因**:「canextend」操作または「extend」操作の間に {template server} から Oracle ユーザを取得できませんでした。<br><br>**処置**: Oracle バイナリの所有者は、{target server} と {template server} の有効なユーザである必要があります。Oracle データベース所有権の権限と所有権を修正し、操作を再試行してください。 |
| 122528 | ERROR | The Oracle user and/or group information for user "%s" does not exist on the server "%s". | **原因**: LifeKeeper は、サーバ {server} の Oracle ユーザ {user} に対して Oracle ユーザおよびグループ情報を検索できません。<br><br>**処置**: 指定した {server} に Oracle ユーザ {user} が存在していることを確認してください。ユーザ {user} が存在しない場合、クラスタ内のすべてのサーバで、同じ uid および gid を使用して作成する必要があります。 |
| 122529 | ERROR | The id for user "%s" is not the same on template server "%s" and target server "%s". | **原因**: ターゲットサーバ {target server} にある Oracle ユーザ {user} のユーザ ID |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | が、テンプレートサーバ {template server} にあるユーザ {user} の値と一致しません。<br><br>**処置**:Oracle ユーザ {user} のユーザ ID はクラスタ内のすべてのサーバで一致している必要があります。ユーザ ID の不一致をすべてのサーバで手動で修正し、「extend」操作を再試行する必要があります。 |
| 122530 | ERROR | The group id for user "%s" is not the same on template server "%s" and target server "%s". | **原因**: ターゲットサーバ {target server} にある Oracle ユーザ {user} のグループ ID が、テンプレートサーバ {template server} にあるユーザ {user} の値と一致しません。<br><br>**処置**: Oracle ユーザ {user} のグループ ID はクラスタ内のすべてのサーバで一致している必要があります。グループ ID の不一致をすべてのサーバで手動で修正し、「extend」操作を再試行する必要があります。 |
| 122532 | ERROR | No file system or raw devices found to extend for "%s" on "%s". | **原因**: サーバ {template server} で Oracle リソース {tag} の依存ファイルシステムまたは Raw デバイスリソースが見つかりません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122533 | WARN | A RAMDISK (%s) was detected in the ORACLE Database configuration for "%s" on "%s". LifeKeeper cannot protect RAMDISK. This RAMDISK resource will not be protected by LifeKeeper! ORACLE hierarchy creation will continue. | **原因**: 指定したテーブルスペース、ログファイル、またはその他のデータベースファイル {database_file} が ramdisk として検出されました。現在の LifeKeeper 製品では、このタイプのリソースに保護を適用できません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: ramdisk は保護されません。すべての Oracle データベース操作で必要なデータベースファイル {database_file} が使用可能になるように手動で確認する必要があります。 |
| 122534 | ERROR | Failed to initialize object instance for Oracle sid "%s" on "%s". | **原因**: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122537 | ERROR | Update of instance info field for "%s" on "%s" failed (%s). | **原因**: 内部リソース情報フィールドを更新する「ins_setinfo」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122538 | ERROR | Initial connect with query buffer to database "%s" on "%s" failed, testing output. | **原因**: データベースの状態を判別するために Oracle データベース {sid} に接続しようとしたところ、失敗しました。<br><br>**処置**: 指定した認証情報で接続が失敗しました。隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122542 | ERROR | The "%s [ %s ]" attempt of the database "%s" appears to have failed on "%s". | **原因**: データベースインスタンス {sid} に |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 対して メソッド {action_method} を使用した Oracle アクション {action} がサーバ {server} で失敗しました。<br><br>**処置**: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122543 | ERROR | All attempts to "%s" database "%s" on "%s" failed | **原因**: サーバ {server} の Oracle データベース {sid} に対するアクション {action} を実行するためのあらゆる取り組みが失敗しました。<br><br>**処置**: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122544 | ERROR | Update of "%s" sid "%s" on "%s" failed. Reason: "%s" "%s" failed: "%s". | **原因**: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。このエラーは oratab ファイルを開こうとしている間に発生しました。<br><br>**処置**: システム起動時にデータベースが自動で起動されないようにするために、{sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122545 | ERROR | Unable to locate the oratab file in "/etc" or in "%s" on "%s". | **原因**: {server} のデフォルトまたは代替ロケーションで oratab ファイルが見つかりませんでした。<br><br>**処置**: oratab ファイルが存在しており、Oracle ユーザの適切な権限を持っていることを確認してください。「extend」操作を完了するには有効な oratab フ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ァイルが必要です。 |
| 122546 | ERROR | Unable to open file "%s" on "%s" (%s). | **原因**: エラー {error} のため、サーバ {server} では、指定したファイル {file} を開くことも、ファイルにアクセスすることもできませんでした。<br><br>**処置**: 指定したファイル {file} の有無や権限を確認してください。隣接するログメッセージおよび関連エラーで詳細を確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122547 | ERROR | (cleanUpPids):Forcefully killing hung pid(s):pid(s)="%s" | **原因**: プロセス {pid} が正常終了するための要求に応答できませんでした。プロセス {pid} は強制終了します。<br><br>**処置**: コマンドラインで、プロセス {pid} が終了していることを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122548 | ERROR | Unable to locate the DB utility (%s/%s) on this host. | **原因**: {path/utility} にある Oracle バイナリおよび必要なデータベースユーティリティ {utility} がこのサーバ {server} では見つかりませんでした。<br><br>**処置**: Oracle バイナリおよび必要なソフトウェアユーティリティがインストールされ、サーバ {server} で適切に設定されていることを確認してください。Oracle バイナリを各ノードにローカルインストールするか、クラスタ内の全ノードで使用可能な共有ストレージに配置する必要があります。 |
| 122549 | ERROR | Oracle internal error or non-standard Oracle configuration detected. Oracle User and/or Group set to "root". | **原因**: 検出された Oracle データベースインストールの所有権が、ルートユーザま |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | たはルートグループに解決されています。ルートによる Oracle インストールの所有権は標準的ではない設定です。<br><br>**処置**: Oracle バイナリの所有者は {server} の有効な非 root ユーザである必要があります。Oracle データベース所有権の権限と所有権を修正し、操作を再試行してください。 |
| 122550 | ERROR | Initial inspection of "%s" failed, verifying failure or success of received output. | **原因**: 前の Oracle クエリー {query} またはコマンド {cmd} が成功を返すことができませんでした。<br><br>**処置**: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122551 | ERROR | Logon failed with "%s" for "%s" on "%s". Please check username/password and privileges. | **原因**: 認証情報 {credentials} を使用してサーバ {server} のデータベースインスタンス {sid} へのログオンに失敗しました。無効なユーザ {user} またはパスワードが指定されました。<br><br>**処置**: Oracle データベースユーザ {user} およびパスワード {password} が有効であることを確認してください。また、Oracle データベースユーザ {user} は、試行されるアクションに対して十分な権限を持っている必要があります。 |
| 122552 | ERROR | %s | **原因**: メッセージに「sqlplus」コマンドの結果が含まれています。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122553 | ERROR | Unable to open file "%s" on "%s" (%s). | 原因: エラー {error} のため、サーバ {server} では、指定したファイル {file} を開くことも、ファイルにアクセスすることもできませんでした。<br><br>処置: 指定したファイル {file} の有無や権限を確認してください。隣接するログメッセージおよび関連エラーで詳細を確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122554 | ERROR | The tag "%s" on "%s" is not an Oracle instance or it does not exist. | 原因: サーバ {server} の指定されたタグ {tag} は、既存の有効な Oracle リソースインスタンスを参照していません。<br><br>処置: UI または「lcdstatus (1M)」を使用して、リソースタグ {tag} の有無を確認してください。コマンド「ora-display」を使用するには、リソースタグ {tag} が Oracle リソースインスタンスである必要があります。 |
| 122555 | ERROR | Failed to create object instance for Oracle on "%s". | 原因: Oracle リソースインスタンスの認定ユーザ、パスワード、データベースロールを更新している際に、保護されている Oracle インスタンスの内部表現を作成しようとしたところ、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122557 | ERROR | Update of user and password failed for "%s" on "%s". | 原因: リソースタグ {tag} のユーザおよびパスワードを更新する要求に失敗しました。指定した認証情報では、サーバ {server} に対する初期検証/接続ができませ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | んでした。<br><br>**処置**: 目的の操作に対して正しい認証情報 {user/password} が指定されていることを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122559 | ERROR | Update of user and password failed for "%s" on "%s". | **原因**: サーバ {server} のリソースタグ {tag} のユーザおよびパスワード情報の更新に失敗しました。<br><br>**処置**: 目的の操作に対して正しい認証情報 {user/password} が指定されていることを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122562 | ERROR | Unable to find the Oracle executable "%s" on "%s". | **原因**: 必要な Oracle 実行ファイル {exe} がこのサーバ {server} では見つかりませんでした。<br><br>**処置**: Oracle バイナリおよび必要なソフトウェアユーティリティがインストールされ、サーバ {server} で適切に設定されていることを確認してください。Oracle バイナリを各ノードにローカルインストールするか、クラスタ内の全ノードで使用可能な共有ストレージに配置する必要があります。 |
| 122566 | ERROR | Unable to find Oracle home for "%s" on "%s". | **原因**: Oracle ホームディレクトリ {Oracle home} には、Oracle インスタンス {sid} の正常な動作に必要なファイルが含まれていないようです。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: Oracle ホームディレクトリ {Oracle home} に Oracle バイナリ、有効な spfile{sid}.ora、または init{sid}.ora ファイルが含まれていることをコマンドラインで確認してください。 |
| 122567 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。指定した内部 ID {id} が、予想した SID {sid} と一致しません。<br><br>**処置**: パラメータが正しいことを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122568 | ERROR | DB Processes are not running on "%s". | **原因**: Oracle インスタンスのプロセスチェックで、サーバ {server} で実行されているプロセスは見つかりませんでした。<br><br>**処置**: ローカルリカバリが有効になっている場合、Oracle インスタンスはローカルで再起動されます。隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122572 | ERROR | Failed to create flag "%s" on "%s". | **原因**: スタンバイノードへのフェイルオーバを促す Oracle ローカルリカバリ処理を制御するためのフラグを作成しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122574 | ERROR | all attempts to shutdown the database %s failed on "%s". | **原因**: ローカルリカバリプロセス中に、おそらくデータベース接続数が上限に達したため、Oracle データベースのシャットダウンに失敗しました。<br><br>**処置**: Oracle ログで、許容接続数の上限に達したことによって起こった接続の障害の有無を確認してください。その障害が見つかった場合は、接続数を増やしてください。また、ローカルリカバリの実行による quickCheck の障害発生が、接続の失敗を引き起こさないように、LK_ORA_NICE を 1 に設定してください。 |
| 122597 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護されている Oracle インスタンスの内部表現を作成中、拡張前のチェックの際に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、pre-extend を再試行してください。 |
| 122598 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 作成中の Oracle インスタンスの内部表現を作成しているときに、Oracle ホームディレクトリの有効性を確認しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、「create」を再試行してください。 |
| 122599 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、テンプレ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ートシステムで Oracle ユーザをルックアップしようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、extend を再試行してください。 |
| 122600 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、リソースプロパティを表示しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、リソースプロパティの表示を再試行してください。 |
| 122601 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、有効なデータベース認証を確認しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、コマンドを再試行してください。 |
| 122603 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、Oracle リソースインスタンスに対してヘルスチェックを実行しようとして、予期しないエラーが発生しました。<br><br>**処置**: 正しい引数が quickCheck コマンドに渡されていることを確認し、隣接す |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | るログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、restore を再試行してください。 |
| 122604 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、Oracle リソースインスタンスに対してローカルリカバリを実行しようとして、予期しないエラーが発生しました。<br><br>**処置**: 正しい引数が「recover」コマンドに渡されていることを確認し、隣接するログメッセージと関連メッセージで詳細を確認してください。レポートされているエラーを修正してから、recover を再試行してください。 |
| 122606 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | **原因**: データベースインスタンス {sid} が実行されていません。またはデータベースへの接続が、指定された認証情報では使用できません。<br><br>**処置**: 選択した操作を完了するには、データベースインスタンス {sid} を {server} で起動し、正しい認証情報を指定する必要があります。 |
| 122607 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | **原因**: データベースインスタンス {sid} が実行されていません。またはデータベースへの接続が、指定された認証情報では使用できません。<br><br>**処置**: 選択した操作を完了するには、データベースインスタンス {sid} を {server} で起動し、正しい認証情報を指定する必要があります。 |
| 122608 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**:「remove」操作で、Oracle リソー |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | スを Out of Service にするために必要なリソースオブジェクトインスタンスを作成できませんでした。<br><br>**処置**: 正しい引数が「remove」コマンドに渡されていることを確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、restore を再試行してください。 |
| 122609 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**:「restore」操作で、Oracle リソースを In Service にするために必要なリソースオブジェクトインスタンスを作成できませんでした。<br><br>**処置**: 正しい引数が「restore」コマンドに渡されていることを確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、「restore」を再試行してください。 |
| 122610 | ERROR | Unable to "%s" on "%s" during resource create. | **原因**: Oracle Application Recovery Kit は、リソース作成中に「getlocks」コマンドで管理ロックを作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create を再試行してください。 |
| 122611 | ERROR | %s | **原因**: 要求された親 Oracle リソースと子 File System リソースとの間の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、creat |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | e 操作を再試行してください。 |
| 122612 | ERROR | %s | **原因**: 要求された親 Oracle リソースと子 Raw リソースとの間の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create 操作を再試行してください。 |
| 122613 | ERROR | %s | **原因**: 要求された親 Oracle リソースと子 Raw リソースとの間の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create 操作を再試行してください。 |
| 122614 | ERROR | %s | **原因**: 要求された親 Oracle リソースと子 Listener リソースとの間の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create 操作を再試行してください。 |
| 122616 | ERROR | %s | **原因**: Oracle データベースの要求された起動またはシャットダウンに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、「restore」または「remove」操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122618 | ERROR | Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed.  Reason: | 原因: LifeKeeper は、データベースリソース {tag} と必要な子リソース {childtag} との間に依存関係を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。問題が修正されている場合は、{tag} と {childtag} との間の依存関係を手動で作成できる可能性があります。 |
| 122619 | ERROR | Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed.  Reason: | 原因: LifeKeeper は、データベースリソース {tag} と必要な子リソース {childtag} との間に依存関係を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。問題が修正されている場合は、{tag} と {childtag} との間の依存関係を手動で作成できる可能性があります。 |
| 122625 | ERROR | Unable to find the Oracle executable "%s" on "%s". | 原因: quickCheck プロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>処置: Oracle 設定を確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |
| 122626 | ERROR | Unable to find the Oracle executable "%s" on "%s". | 原因: 削除プロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>処置: Oracle 設定を確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122627 | ERROR | Unable to find the Oracle executable "%s" on "%s". | **原因**: リストアプロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>**処置**: Oracle 設定を確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |
| 122628 | ERROR | Unable to find the Oracle executable "%s" on "%s". | **原因**: 復旧プロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>**処置**: Oracle 設定を確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |
| 122632 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: 削除の際に、削除プロセスに渡されるリソースインスタンス {sid} が、{sid} の内部リソースインスタンス情報と一致しません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122633 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: リストアの際に、restore に渡されるリソースインスタンス {sid} が、{sid} の内部リソースインスタンス情報と一致しません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122634 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: リソースリカバリの際に、recovery に渡されるリソースインスタンス {sid} が、{sid} の内部リソースインスタンス |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 情報と一致しません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122636 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | **原因**:{server} で Oracle リソース階層 {tag} の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122638 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**:サーバ {server} の Oracle データベースリソース {tag} で作成アクションに失敗しました。シグナル {sig} が作成プロセスによって受信されました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122640 | ERROR | Error creating resource "%s" on server "%s" | **原因**: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122641 | ERROR | Error creating resource "%s" on server "%s" | **原因**: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122642 | ERROR | Error creating resource "%s" on server "%s" | 原因: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122643 | ERROR | Cannot extend resource "%s" to server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を拡張できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122644 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: リソースインスタンス情報 {tag} を {server} で取得しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正し、extend を再試行してください。 |
| 122645 | ERROR | Cannot access canextend script "%s" on server "%s" | 原因: {server} で依存関係の子リソース用の「canextend」スクリプトが見つからなかったので、LifeKeeper は拡張前のチェックを実行できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122646 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: リソースインスタンス情報 {tag} を {server} で取得しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポート |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | されているエラーを修正し、extend を再試行してください。 |
| 122647 | ERROR | Cannot extend resource "%s" to server "%s" | **原因**: LifeKeeper は、{server} でリソース {resource} を拡張できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122648 | ERROR | Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s" | **原因**: データベースリソースの拡張中に、同じ {tag} または {id} で異なるリソースアプリケーションおよびタイプを使用しているリソースインスタンスが見つかりました。<br><br>**処置**: リソース ID は一意のものにする必要があります。Oracle リソースインスタンスに合致する ID のリソースインスタンスを削除してください。 |
| 122649 | ERROR | Error creating resource "%s" on server "%s" | **原因**: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122650 | ERROR | Cannot access extend script "%s" on server "%s" | **原因**: 依存関係にある子リソースのスクリプト {extend} がサーバ {server} で見つけられなかったので、サーバ {server} にデータベースリソース {resource} を拡張する要求が失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122651 | ERROR | Cannot extend resource "%s" to server "%s" | 原因: 依存関係にある子リソースの拡張でエラーが発生したため、サーバ {server} にデータベースリソース {resource} を拡張する要求が失敗しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122654 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: quickCheck プロセスがシグナルを受信したため、データベース {sid} のヘルスチェックが終了しました。これはおそらく、quickCheck プロセスの処理に、割り当てられているよりも長い時間が必要となっているためです。<br><br>処置: Oracle リソースのヘルスチェック時間は ORACLE_QUICKCHECK_TIMEOUT の設定で制御します。ヘルスチェックプロセスの完了に今より長い時間を割り当てる場合は、45 秒よりも大きな値に設定してください。 |
| 122655 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: 削除プロセスがシグナルを受信したため、データベース {sid} を「Out of Service」にする要求が終了しました。これはおそらく、削除プロセスの処理に、割り当てられているよりも長い時間が必要となっているためです。<br><br>処置: Oracle リソースの削除時間は ORACLE_REMOVE_TIMEOUT の設定で制御します。削除プロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122659 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: リストアプロセスがシグナルを受信したため、データベース {sid} を「In Service」にする要求が終了しました。これはおそらく、リストアプロセスの処理 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | に、割り当てられているよりも長い時間が必要となっているためです。<br><br>**処置**: Oracle リソースのリストア時間は ORACLE_RESTORE_TIMEOUT の設定で制御します。リストアプロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122663 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**: リカバリプロセスがシグナルを受信したため、失敗したデータベースのリカバリが終了しました。これはおそらく、リカバリプロセスの処理に、割り当てられているよりも長い時間が必要となっているためです。<br><br>**処置**: Oracle リソースのリカバリ時間は ORACLE_RESTORE_TIMEOUT と ORACLE_REMOVE_TIMEOUT の設定で制御します。リカバリの完了に今より長い時間を割り当てる場合は、一方または両方を 240 秒よりも大きな値に設定してください。 |
| 122670 | ERROR | Update of "%s" sid "%s" on "%s" failed.  Reason: "%s" "%s" failed: "%s". | **原因**: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。更新プロセスで使用する一時ファイルを開こうとして、エラーが発生しました。<br><br>**処置**: システム起動時にデータベースが自動で起動されないようにするために、{sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122671 | ERROR | Update of "%s" sid "%s" on "%s" failed.  Reason: "%s" "%s" failed: "%s". | **原因**: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。更新プロセスで使用する一時ファイルを閉じようと |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | して、エラーが発生しました。<br><br>**処置**: システム起動時にデータベースが自動で起動されないようにするために、{sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122672 | ERROR | Update of "%s" sid "%s" on "%s" failed.  Reason: "%s" "%s" failed: "%s". | **原因**: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。一時ファイルの名前を oratab に変更しようとして、エラーが発生しました。<br><br>**処置**: システム起動時にデータベースが自動で起動されないようにするために、{sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122673 | ERROR | Unable to log messages queued while running as oracle user %s on %s. Reason: $! | **原因**: メッセージをログファイルに追加しようとして、予期しないエラー {reason} が発生しました。これらのメッセージは、Oracle ユーザとして実行している間に生成されました。<br><br>**処置**: 障害に関する理由を参照し、修正処置を実行してください。 |
| 122674 | ERROR | Unable to open %s Reason: %s. | **原因**: Oracle データベースへの接続を開き、データベース {cmd} を実行しようとしたときに、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。また、Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122680 | ERROR | Unable to find Oracle home for “%s” on “%s”. | 原因: Oracle ホームディレクトリ {Oracle home} には、Oracle インスタンス {sid} の正常な動作に必要なファイルが含まれていないようです。<br><br>処置: Oracle ホームディレクトリ {Oracle home} に Oracle バイナリ、有効な spfile{sid}.ora、または init{sid}.ora ファイルが含まれていることをコマンドラインで確認してください。 |
| 122681 | ERROR | Failed to create object instance for Oracle on "%s". | 原因: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122682 | ERROR | Unable to find the Oracle executable "%s" on "%s". | 原因: 必要な Oracle 実行ファイル {exe} がこのサーバ {server} では見つかりませんでした。<br><br>処置: Oracle バイナリおよび必要なソフトウェアユーティリティがインストールされ、サーバ {server} で適切に設定されていることを確認してください。Oracle バイナリを各ノードにローカルインストールするか、クラスタ内の全ノードで使用可能な共有ストレージに配置する必要があります。 |
| 122683 | ERROR | Backup node %s is unreachable; abort username/password changes. | 原因: バックアップノード {node} に接続できません。保留中のユーザー名/パスワードに対する変更は中断されました。<br><br>処置: 指定したバックアップノード {node} 上でSIOS LifeKeeper が起動中で、す |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | べてのコミュニケーションパスが確立されていることが確認できたら、再度操作を行ってください。 |
| 122684 | ERROR | The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper. | 原因: サーバー {server} 上で Oracle リソース {resource} の restore 処理がタイムアウトしました。これは、In Service 処理が設定された時間を経過すると発生します。<br><br>処置: Oracle リソースのリストア時間は ORACLE_RESTORE_TIMEOUT の設定で制御します。リストアプロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122685 | ERROR | The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper. | 原因: サーバー {server} 上で Oracle リソース {resource} の remove 処理がタイムアウトしました。 これは、Out of Service 処理が設定された時間を経過すると発生します。<br><br>処置: Oracle リソースの削除時間は ORACLE_REMOVE_TIMEOUT の設定で制御します。削除プロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122686 | ERROR | The quickCheck of %s has timed out on server %s. The default TIMEOUT is 45 seconds. To increase the TIMEOUT, set ORACLE_QUICKCHECK_TIMEOUT in /etc/default/LifeKeeper. | 原因: サーバー {server} 上で Oracle リソース {resource} に対する quickCheck がタイムアウトしました。 これは、health check 処理が設定された時間を経過すると発生します。<br><br>処置: Oracle リソースのヘルスチェック時間は ORACLE_QUICKCHECK_TIMEOUT の設定で制御します。ヘルスチェックプロセスの完了に今より長い時間を割り当てる場合は、45 秒よりも大きな |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 値に設定してください。 |
| 122687 | ERROR | Usage: %s %s | **原因**: getalertlog スクリプトに対して無効なパラメーターが指定されました。<br><br>**処置**: Oracle リソースのタグ名が、最初のコマンドラインのパラメーターとして提供される必要があります。パラメーターが正しいかどうかを確認し、再度操作を行ってください。 |
| 123006 | FATAL | Unknown version %s of IP address | **原因**: IP アドレスが IPv4 でも IPv6 でも有効ではないようです。<br><br>**処置**: 有効な IP アドレスを指定してください。 |
| 123008 | ERROR | No pinglist found for %s. | **原因**: この IP アドレスの pinglist を開こうとして問題が発生しました。<br><br>**処置**: この IP アドレスの pinglist を指定していることを確認してください。 |
| 123009 | ERROR | List ping test failed for virtual IP %s | **原因**: ping リストのどのアドレスからも応答を受信しませんでした。<br><br>**処置**: このノードのネットワーク接続および ping リストの IP が存在するシステムを確認してください。 |
| 123013 | ERROR | Link check failed for virtual IP %s on interface %s. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、物理レイヤー接続にリンクが存在しないことを示します。<br><br>**処置**: インターフェースの物理接続を確認し、物理レイヤーリンクを有効にしてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 123015 | ERROR | Link check failed for virtual IP %s on interface %s. | **原因**: 要求したインターフェースは bonding インターフェースで、スレーブのいずれか 1 つが「NO-CARRIER」です。これは、物理レイヤー接続にリンクが存在しないことを示します。<br><br>**処置**: スレーブインターフェースの物理接続を確認し、物理レイヤーリンクを有効にしてください。 |
| 123024 | ERROR | IP address seems to still exist somewhere else. | **原因**: IP アドレスはネットワークの別の場所で使用中のようです。<br><br>**処置**: 別の IP アドレスを選択して使用するか、この IP アドレスを特定して、現在の使用を無効にしてください。 |
| 123037 | ERROR | must specify machine name containing primary hierarchy | **原因**: creIPhier に指定した引数が不十分です。<br><br>**処置**: 必要な引数をすべて creIPhier に指定してください。 |
| 123038 | ERROR | must specify IP resource name | **原因**: creIPhier に渡された引数が不十分です。<br><br>**処置**: 必要な引数をすべて creIPhier に指定してください。 |
| 123039 | ERROR | must specify primary IP Resource tag | **原因**:「creIPhier」コマンドにプライマリ IP リソースタグの引数が指定されていません。<br><br>**処置**: 必要な引数をすべて指定してください。 |
| 123042 | ERROR | An unknown error has occurred in utility validmask on machine %s. | **原因**:「validmask」ユーティリティの実行中に予期しないエラーが発生しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細を確認してください。 |
| 123045 | ERROR | An unknown error has occurred in utility getlocks. | **原因**:「getlocks」ユーティリティの実行中に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 123053 | ERROR | Cannot resolve hostname %s | **原因**: IP アドレスに対してホスト名が指定されましたが、システムはホスト名を IP アドレスに解決できませんでした。<br><br>**処置**: ホスト名が正確かどうかを確認してから、名前の解決（DIS または /etc/hosts）が正常に機能しており、ホスト名の IP を返すことを確認してください。 |
| 123055 | ERROR | An unknown error has occurred in utility %s on machine %s. | **原因**: IP リソースの作成中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123056 | ERROR | create ip hierarchy failure: perform_action failed | **原因**:作成中に IP アドレスをリストアしようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 123059 | ERROR | Resource already exists on machine %s | **原因**: すでに存在する IP アドレスを作成しようとしました。<br><br>**処置**: 既存のリソースを再利用するか、すでに存在している IP アドレスを手動で削除するか、別の IP アドレスを |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 使用してください。 |
| 123060 | ERROR | ins_create failed on machine %s | **原因**: IP リソースの作成中に予期しない障害が発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 123064 | ERROR | An unknown error has occurred in utility %s on machine %s. | **原因**: IP リソースの依存関係を作成中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123066 | ERROR | An error occurred during creation of LifeKeeper application=comm on %s. | **原因**:「app_create」の呼び出し中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123068 | ERROR | An error occurred during creation of LifeKeeper resource type=ip on %s. | **原因**:「typ_create」の呼び出し中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123089 | ERROR | Link check failed for virtual IP %s on interface %s. | |
| 123091 | ERROR | the link for interface %s is down | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、物理レイヤー接続にリンクが存在しないことを示します。<br><br>**処置**: インターフェースの物理接続を確 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 認し、物理レイヤーリンクを有効にしてください。 |
| 123093 | ERROR | the ping list check failed | **原因**: ping リストのどのアドレスからも応答を受信しませんでした。<br><br>**処置**: このノードのネットワーク接続および ping リストの IP が存在するシステムを確認してください。 |
| 123095 | ERROR | broadcast ping failed | **原因**: ブロードキャスト ping から応答を受信しませんでした。<br><br>**処置**: サブネット上の 1 つ以上のホストがブロードキャスト ping に応答することを確認してください。仮想 IP が正しいネットワークインターフェースにあることを確認してください。ブロードキャスト ping ではなく、pinglist の使用を検討してください。 |
| 123096 | ERROR | $msg | **原因**: 仮想 IP の実行可能性の判定に使用するブロードキャスト ping に失敗しました。<br><br>**処置**: このリソースの ping リストがプロパティパネルで正常に設定されていることを確認するか、/etc/default/LifeKeeper 設定ファイルに NOBCASTPING=1 を追加してブロードキャスト ping のチェックを確実に無効にしてください。 |
| 123097 | ERROR | exec_list_ping(): broadcast ping failed. | **原因**: 仮想 IP の実行可能性の判定に使用するブロードキャスト ping に失敗しました。<br><br>**処置**: このリソースの ping リストがプロパティパネルで正常に設定されていることを確認するか、/etc/default/LifeKeepe |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | r 設定ファイルに NOBCASTPING=1 を追加してブロードキャスト ping のチェックを確実に無効にしてください。 |
| 123299 | ERROR | Unable to open %s. Reason %s | |
| 123410 | ERROR | Usage error OSUquickCheck | |
| 123411 | ERROR | OSUquickCheck: both tag and id name not specified | |
| 123412 | ERROR | resource $Tag not found on local server | |
| 123414 | ERROR | The link for network interface $IPObj->{'device'} is down | |
| 123415 | ERROR | No pinglist found for $IPObj->{'ipaddr'} | |
| 123416 | ERROR | List ping test failed for virtual IP $IPObj->{'ipaddr'} | |
| 124004 | FATAL | resource tag name not specified | **原因**:「quickCheck」操作で無効な引数が指定されました。<br><br>**処置**: 正しい引数が渡されたことを確認してください。 |
| 124005 | FATAL | resource id not specified | **原因**:「quickCheck」操作で無効な引数が指定されました。<br><br>**処置**: 正しい引数が渡されたことを確認してください。 |
| 124007 | FATAL | Failed to get resource information | **原因**: ファイルシステムリソースの情報フィールドに正しい情報が含まれていません。<br><br>**処置**: リソースの情報フィールドに正しい情報を入力するか、最近の「lkbackup」からシステムをリストアして元の情報フィールドをリストアします。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124008 | ERROR | getId failed | **原因**: ファイルシステムリソースは基本ディスクデバイスを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。リソース階層が有効であること、および必要なストレージキットがすべてインストールされていることを確認してください。 |
| 124009 | ERROR | LifeKeeper protected filesystem is in service but quickCheck detects the following error: | **原因**: ファイルシステムキットはリソースに問題を発見しました。<br><br>**処置**: この直後に表示されるメッセージで詳細を確認してください。 |
| 124010 | ERROR | "$id" is not mounted | **原因**: ファイルシステムリソースがマウントされていません。<br><br>**処置**: 処置は必要ありません。ローカルリカバリでリソースの再マウントを許可してください。 |
| 124011 | ERROR | "$id" is mounted but with the incorrect mount options (current mount option list: $mntopts, expected mount option list: $infoopts) | **原因**: ファイルシステムリソースが不正にマウントされています。<br><br>**処置**: 処置は必要ありません。ローカルリカバリでリソースの再マウントを許可してください。 |
| 124012 | ERROR | "$id" is mounted but on the wrong device (current mount device: $tmpdev, expected mount device: $dev) | **原因**: ファイルシステムリソースに誤ったデバイスがマウントされています。<br><br>**処置**: 処置は必要ありません。ローカルリカバリでリソースの再マウントを許可してください。 |
| 124015 | ERROR | LifeKeeper protected filesystem "$tag" ($id) is $percent% full ($blocksfree free bloc | **原因**: ディスク使用量が FILESYSFULLE |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | ks). | RROR の値（%）を超過しました。<br><br>**処置**: ファイルシステムからデータを削除または移行してください。 |
| 124016 | WARN | LifeKeeper protected filesystem "$tag" ($id) is $percent% full ($blocksfree free blocks). | **原因**: ディスク使用量が FILESYSFULL WARNの値（%）を超過しました。<br><br>**処置**: ファイルシステムからデータを削除または移行してください。 |
| 124020 | FATAL | cannot find device information for filesystem $id | **原因**: ファイルシステムリソースは基本ディスクデバイスを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。リソース階層が有効であること、および必要なストレージキットがすべてインストールされていることを確認してください。 |
| 124029 | ERROR | Failed to find child resource. | **原因**: ファイルシステムリソースは基本ディスクリソースを特定できませんでした。<br><br>**処置**: リソース階層が正しいことを確認してください。 |
| 124032 | FATAL | Script has hung. Exiting. | **原因**: アンマウントする必要があるものの、マウントされているファイルシステムに開かれているファイルが複数のプロセスにありました。それらのプロセスの終了に時間がかかりすぎています。<br><br>**処置**: このエラーが続く場合は、マウントポイントを使用している可能性のあるすべてのソフトウェアを一時的に停止し、アンマウントできるようにしてください。ファイルシステムをそれでもアン |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | マウントできない場合は、サポートに問い合わせてください。 |
| 124042 | ERROR | file system $fs failed unmount; will try again | **原因**: アンマウントする必要があるものの、マウントされているファイルシステムに開かれているファイルが複数のプロセスにありました。場合によっては、それらのプロセスを消去するために、何度か操作を実行する必要があります。<br><br>**処置**: 処置は必要ありません。プロセスをそのまま続行してください。 |
| 124046 | ERROR | file system $fsname failed unmount | **原因**: ファイルシステムをアンマウントできませんでした。<br><br>**処置**: このエラーが続く場合は、マウントポイントを使用している可能性のあるすべてのソフトウェアを一時的に停止し、アンマウントできるようにしてください。ファイルシステムをそれでもアンマウントできない場合は、サポートに問い合わせてください。 |
| 124049 | ERROR | Local recovery of resource has failed (err=$err) | **原因**: ファイルシステムリソースに、ローカルでは修正できない問題があります。<br><br>**処置**: 処置は必要ありません。リソースを別のシステムにフェイルオーバできるようにしてください。 |
| 124051 | WARN | getId failed, try count : $cnt/$try | |
| 124052 | ERROR | \"$id\" is mounted but filesystem is shutdown state. | |
| 124054 | ERROR | Failed to change mount option from \”$old_opts\” to \”$new_opts\” for migration | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124103 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124104 | ERROR | $ERRMSG Required template machine name is null | **原因**: canextend 操作で無効な引数が指定されました。<br><br>**処置**: 引数が正しいことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124105 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: canextend 操作で無効な引数が指定されました。<br><br>**処置**: 引数が正しいことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124106 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: リソースの基本ディスク情報を特定できませんでした。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124107 | ERROR | $ERRMSG Resource "$TemplateTagName" must have one and only one device resource dependency | **原因**: リソースの階層にある基本デバイスが多すぎます。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124108 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: テンプレートシステムでリソースを見つけられません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124109 | ERROR | $ERRMSG Can not access canextend for scsi/$DeviceResType resources on machine "$TargetSysName" | **原因**: ターゲットシステムに、必須のコンポーネントがいくつか見つかりません。<br><br>**処置**: ターゲットシステムに正しいキットがすべてインストールされ、ライセンスされていることを確認してください。 |
| 124110 | ERROR | $ERRMSG Either filesystem "$TemplateLKId" is not mounted on "$TemplateSysName" or filesystem is not shareable with "$TargetSysName" | **原因**: ファイルシステムがテンプレートシステムのサービス内にないか、ターゲットシステムに拡張するための要件を満たしていません。<br><br>**処置**: リソースがテンプレートシステムのサービス内にあることを確認し、ファイルシステムを拡張するための要件に関する製品ドキュメントを参照してください。 |
| 124111 | ERROR | $ERRMSG File system type "${FSType}" is not supported by the kernel currently running on "${TargetSysName}" | **原因**: カーネルのサポートがないため、ファイルシステムのタイプをターゲットシステムにマウントできません。<br><br>**処置**: ターゲットシステムにそのカーネルモジュールがすべてインストールされており、正常に設定されていることを確認してから、リソースを拡張してください。 |
| 124112 | ERROR | must specify machine name containing primary hierarchy | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124113 | ERROR | must specify primary ROOT tag | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124114 | ERROR | must specify primary mount point | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124115 | ERROR | must specify primary switchback type | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124118 | ERROR | dep_remove failure on machine \""$PRIMACH"" for parent "$PRITAG" and child "$DEVTAG." | **原因**: 依存関係の作成後のクリーンアップに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124119 | ERROR | ins_remove failure on machine ""$PRIMACH"" for \"$PRITAG." | **原因**: インスタンスの作成後のクリーンアップに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124121 | ERROR | ins_remove failure on machine ""$PRIMACH" | **原因**: リソースの作成後のクリーンアップに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124122 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124123 | ERROR | $ERRMSG Required template machine name is null | **原因**: depstoextend 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124124 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: depstoextend 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124125 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: リソースが基本ディスクリソースを特定できませんでした。<br><br>**処置**: 階層とすべての依存関係が正しいことを確認してから拡張してください。 |
| 124126 | ERROR | unextmgr failure on machine ""$PRIMACH" | **原因**: 失敗したリソース拡張操作の後のクリーンアップに失敗しました。<br><br>**処置**: 残りのリソースを手動でクリーンアップし、隣接するログメッセージで詳細を確認してください。 |
| 124128 | ERROR | unextmgr failure on machine ""$PRIMACH"" for "$PRITAG." | **原因**: 失敗したリソース拡張操作の後のクリーンアップに失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 残りのリソースを手動でクリーンアップし、隣接するログメッセージで詳細を確認してください。 |
| 124129 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 追加のログメッセージを探して、詳細を確認してください。 |
| 124130 | ERROR | $ERRMSG Required template machine name is null | **原因**: 拡張操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124131 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: 拡張操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124132 | ERROR | $ERRMSG Required target mount point is null | **原因**: 拡張操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124133 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: 拡張中のタグがテンプレートシステムに存在しません。<br><br>**処置**: テンプレートシステムで階層が正 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | しいことを確認してから拡張してください。 |
| 124134 | ERROR | $ERRMSG Detected conflict in expected tag name "$TargetTagName" on target machine. | **原因**: 拡張するリソースと同じタグのリソースがすでにターゲットシステムにあります。<br><br>**処置**: 競合しているリソースのどちらか一方を別のタグで再作成してください。 |
| 124135 | ERROR | $ERRMSG Resource "$TemplateTagName" does not have required device resource dependency or unable to access this resource on template machine. | **原因**: リソースまたは基本ディスクリソースがテンプレートシステムで見つかりません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124136 | ERROR | $ERRMSG Resource "$TemplateTagName" must have one and only one device resource dependency | **原因**: リソースは、テンプレートシステムの階層内に複数の基本デバイスを持っています。<br><br>**処置**: 拡張する前に階層が正しいことを確認し、ファイルシステムリソースが単一のディスクリソースにのみ依存していることを確認してください。 |
| 124137 | ERROR | $ERRMSG Can not access extend for scsi/$DeviceResType resources on machine "$TargetSysName" | **原因**: 所定のストレージタイプのサポートに必要なファイルがターゲットシステムにありません。<br><br>**処置**: 必要なキットがターゲットシステムにインストールされ、ライセンスされていることを確認してください。 |
| 124138 | ERROR | $ERRMSG Unable to access target device resource "$DeviceTagName" on machine "$TargetSysName&quot | **原因**: 必要な基本ディスクリソースがターゲットシステムに存在していません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細を確認し、ターゲットシステムが拡張対象リソースをホストするために正常に設定されていることを確認してください。 |
| 124141 | ERROR | $ERRMSG Unable to find mount point "$TemplateLKId" mode on template machine | **原因**: テンプレートシステムのマウントポイント詳細を特定できません。<br><br>**処置**: リソースが in service で、テンプレートシステムでアクセス可能であることを確認してから拡張してください。 |
| 124142 | ERROR | $ERRMSG Unable to create or access mount point "$TargetLKId" on target machine | **原因**: ターゲットシステムでマウントポイントを作成できませんでした。<br><br>**処置**: マウントポイントの親ディレクトリが存在し、ターゲットシステムでアクセス可能であることを確認してください。 |
| 124143 | ERROR | $ERRMSG Two or more conflicting entries found in /etc/fstab on "$TargetSysName&quot | **原因**: デバイスまたはマウントポイントはターゲットシステムで複数回マウントされているようです。<br><br>**処置**: マウントポイントがターゲットシステムにマウントされていないことを確認してから拡張してください。 |
| 124144 | ERROR | $ERRMSG Failed to create resource instance on "$TargetSysName" | **原因**: ターゲットシステムでのリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。ターゲットサーバでログを必ず確認してください。 |
| 124145 | ERROR | $ERRMSG Failed to set resource instance state for "$TargetTagName" on \"$TargetSysName" | **原因**: ターゲットシステムでソースの状態を OSU に変更できませんでした。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124146 | ERROR | must specify machine name containing primary hierarchy | **原因**: filesyshier 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124147 | ERROR | must specify primary mount point | **原因**: filesyshier 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124149 | ERROR | create file system hierarchy failure | **原因**: リソースインスタンスを検索するプロセスが失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124150 | ERROR | create file system hierarchy failure | **原因**: システムが /etc/mtab ファイルの読み取りに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124152 | ERROR | create file system hierarchy failure | **原因**: 基本ディスクリソースを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124153 | ERROR | create file system hierarchy failure | **原因**: ファイルシステムリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124154 | ERROR | create file system hierarchy failure | **原因**: リソースの情報フィールドを更新できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124155 | ERROR | create file system hierarchy failure | **原因**: スイッチバック戦略をリソースで設定できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124157 | ERROR | create file system hierarchy failure (conflicting entries in /etc/fstab\) | **原因**: マウントポイントを /etc/fstab ファイルから削除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124160 | ERROR | Unknown error in script filesysins, err=$err | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124161 | ERROR | create filesys instance – existid – failure | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124163 | ERROR | create filesys instance – ins_list – failure | **原因**: 既存のリソースの確認に失敗しま |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | した。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124164 | ERROR | create filesys instance – newtag – failure | **原因**: システムはリソースの推奨タグの生成に失敗しました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124168 | ERROR | create filesys instance - ins_create - failure | **原因**: ファイルシステムリソースを作成できませんした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124169 | ERROR | filesys instance - ins_setstate - failure | **原因**:新しいファイルシステムリソースの状態を初期化できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124173 | ERROR | create filesys instance - dep_create - failure | **原因**: 基本ディスクに対するリソースの依存関係を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124174 | ERROR | machine not specified | **原因**: rmenu_mp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124175 | ERROR | mount point not specified | **原因**: rmenu_mp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124177 | ERROR | unexpected multiple matches found | **原因**: 1 つまたは複数のシステムが、複数回使用されたファイルシステムまたはマウントポイントを示しています。<br><br>**処置**: ファイルシステムデバイスとマウントポイントを確認し、ファイルシステムが 1 回だけマウントされていることを確認してください。追加のログメッセージを探して、詳細を確認してください。 |
| 124178 | ERROR | machine name not specified | **原因**: rmenump 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124180 | ERROR | must specify filesystem type | **原因**: validfstype 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124181 | ERROR | mount point not specified | **原因**: validmp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作で |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | このエラーが発生した場合は、サポートに問い合わせてください。 |
| 124182 | ERROR | The mount point $MP is not an absolute path | **原因**: 絶対パスではない（「/」で始まらない）マウントポイントが指定されました。<br><br>**処置**:「/」で始まる絶対パスでマウントポイントを指定してください。 |
| 124183 | ERROR | $MP is already mounted on $MACH | **原因**: 要求されたマウントポイントはシステムですでに使用中です。<br><br>**処置**: 使用中ではないマウントポイントを指定するか、操作を再試行する前にアンマウントしてください。 |
| 124184 | ERROR | The mount point $MP is already protected by LifeKeeper on $MACH | **原因**: システムは、指定したマウントポイントをすでに保護しています。<br><br>**処置**: まだ保護されていない別のマウントポイントを選択してください。 |
| 124185 | ERROR | The mount point $MP is not a directory on $MACH | **原因**: マウントポイントが通常のファイルなどの非ディレクトリを参照しています。<br><br>**処置**: ディレクトリを参照するマウントポイントを選択してください。 |
| 124186 | ERROR | The mount point directory $MP is not empty on $MACH | **原因**: 指定したマウントポイントは空でないディレクトリを参照します。<br><br>**処置**: 空のマウントポイントを選択するか、指定したディレクトリの内容を削除してから操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124187 | ERROR | server name not specified | **原因**: valuepmp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124188 | ERROR | There are no mount points on server $MACH | **原因**: サーバにファイルシステムリソースのマウントポイントがありません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124194 | WARN | Please correct conflicting "/etc/fstab" entries on server $UNAME for: $FSDEV, $FSNAME | **原因**: ファイルシステムリソースを削除したら、/etc/fstab の一部のエントリを手動でクリーンアップする必要があります。<br><br>**処置**: /etc/fstab ファイルを手動でクリーンアップします。 |
| 124195 | ERROR | getchildinfo found no $OKAPP child for $PTAG | **原因**: システムは子リソースを階層で見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、階層が正しいことを確認してから、操作を再試行してください。 |
| 124196 | ERROR | enablequotas - quotacheck may have failed for $FS_NAME | **原因**: 割り当て操作に失敗しました。<br><br>**処置**: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124198 | ERROR | enablequotas - quotaon failed to turn on quotas for $FS_NAME, reason | **原因**: 割り当て操作に失敗しました。<br><br>**処置**: lifekeeper ログおよび /var/log/mes |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | sages の隣接するログメッセージで詳細を確認してください。 |
| 124200 | ERROR | The device node $dev was not found or did not appear in the udev create time limit of $delay seconds | 原因: デバイスノード（/dev/...）が udev によって作成されませんでした。これは、ストレージまたはストレージへのサーバ接続の問題を示している可能性があります。<br><br>処置: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124201 | WARN | Device $device not found. Will retry wait to see if it appears. | 原因: これは、udev がストレージのデバイスノードエントリを作成している間、通常の条件下で発生する可能性もあります。このメッセージは繰り返し表示されないはずです。<br><br>処置: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124202 | ERROR | Command "$commandwithargs" failed. Retrying …. | 原因: 指定されたコマンドは失敗しましたが、一時的なものである可能性もあります。この障害は、通常の操作で発生しますが、継続はしないはずです。<br><br>処置: このメッセージが何度も表示される場合は、隣接するログメッセージで詳細を確認してください。 |
| 124204 | WARN | cannot make file system $FSNAME mount point | 原因: マウントポイントディレクトリを作成できませんでした。<br><br>処置: マウントポイントを作成できることを確認してください。これは、ファイルシステム権限やマウントオプションなどによる可能性があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124207 | ERROR | "fsck"ing file system $FSNAME failed, trying alternative superblock | **原因**: このメッセージは、通常のファイルシステムチェックに失敗したことを示します。このメッセージは、ext2 ファイルシステムまたは代替スーパーブロックロケーションが使用されているファイルシステムの場合は問題ないと思われます。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124209 | ERROR | "fsck"ing file system $FSNAME with alternative superblock failed | **原因**: これは、代替スーパーブロックロケーションで ext2 ファイルシステム (または代替スーパーブロックロケーションが使用されている他のファイルシステム) の確認が失敗したことを示します。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、実行方法の手順を確かめてください。 |
| 124210 | WARN | POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME) | **原因**: ミラーソースと同期していないファイルシステムが in service になったか、フェイルオーバされました。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、ファイルシステムを安全に in service にする方法を製品ドキュメントで調べてください。 |
| 124211 | ERROR | Reason for fsck failure ($retval): $ret | **原因**: このログメッセージは一連のメッセージの一部で、fsck プロセスからの実際の終了コードを提供します。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、実行方法の手順を確かめてください。 |
| 124212 | ERROR | "fsck" of file system $FSNAME failed | **原因**: ファイルシステムの確認に失敗し |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ました。これは通常、ファイルシステムが破損したことによるものです。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。製品ドキュメントを参照し、可能性のあるファイルシステムの破損を処理する方法を調べてください。 |
| 124213 | WARN | POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME) | **原因**: システムまたはユーザが、破損している可能性のあるファイルシステムを in service にしようとしました。これは、ミラーサイトと同期していないときにファイルシステムが切り替えられたか、フェイルオーバされた場合に起こることがあります。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、リソースを安全に in service にする方法を製品ドキュメントで調べてください。 |
| 124214 | ERROR | Reason for fsck failure ($retval) | **原因**: このメッセージは、ファイルシステムチェックの障害に関する前のログメッセージに続いて表示されるもので、fsck プロセスのプロセス終了コードを提供します。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124218 | ERROR | File system $FSNAME was found to be already | **原因**: このメッセージは、一連のメッセージの一部です。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124219 | ERROR | mounted after initial mount attempt failed. | **原因**: このメッセージは、一連のメッセージの一部です。通常の状態では表示さ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | れないはずですが、リソースが in service の場合は致命的なエラーの可能性もあります。<br><br>**処置**: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124220 | ERROR | File system $FSNAME failed to mount. | **原因**: ファイルシステムをマウントできませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124221 | WARN | Protected Filesystem $ID is full | **原因**: ファイルシステムの空き容量がありません。<br><br>**処置**: 使用していないデータをファイルシステムから削除するか、より大きなファイルシステムに移行してください。 |
| 124222 | WARN | Dependent Applications may be affected <> | **原因**: これは、あるリソースに対する操作が、リソースの階層に基づいて別のリソースにも実行される可能性が高いことを示します。<br><br>**処置**: 示されているリソースが影響を受けても許容できることを確認してから操作を続行してください。 |
| 124223 | ERROR | Put "$t" Out-Of-Service Failed By Signal | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124227 | ERROR | Put "$i" Out-Of-Service Failed | **原因**: 操作に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124230 | ERROR | Put "$t" In-Service Failed By Signal | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124231 | ERROR | Put "$t" In-Service Failed | **原因**: 操作に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124234 | ERROR | Put "$t" In-Service Failed | **原因**: 操作に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 125102 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125103 | ERROR | `printf '%s is not shareable with any machine.' $DEV` | **原因**: デバイスは他のシステムと共有されていないようです。<br><br>**処置**: デバイスがクラスタ内のすべてのサーバからアクセス可能であることを確認してください。すべての関連ストレージドライバおよびソフトウェアが正常にインストールおよび設定されていることを確認してください。 |
| 125104 | ERROR | `printf 'Failed to create disk hierarchy for "%s" on "%s"' $PRIMACH $DEV` | **原因**: 物理ディスクを保護するリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 確認し、レポートされている問題を解決してください。 |
| 125107 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125114 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125120 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125123 | ERROR | `printf 'Cannot access depstoextend script "%s" on server "%s"' $depstoextend $TargetSysName` | 原因: {server} でスクリプト{DEPSTOEXTEND}を見つけられなかったので、LifeKeeper はリソース階層に対して拡張前のチェックを実行できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 125126 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $ChildTag $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125129 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125155 | ERROR | SCSI $DEV failed to lock. | 原因: SCSI デバイスをロックする際に問題が発生しました。<br><br>処置: 隣接するログメッセージで詳細を確認し、レポートされている問題を解決してください。 |
| 125164 | ERROR | SCSI $INFO failed to unlock. | 原因: SCSI デバイスをロック解除する |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 際に問題が発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、レポートされている問題を解決してください。 |
| 125181 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTag  $TemplateSysName` | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125194 | ERROR | Failed to check disk.(tag=\"$opt_t\" | |
| 126105 | ERROR | script not specified - $PTH is a directory | **原因**: 指定したスクリプトパスはディレクトリです。<br><br>**処置**: スクリプトのパスを修正してください。 |
| 126110 | ERROR | script $PTH does not exist | **原因**: 指定したスクリプトパスは存在しません。<br><br>**処置**: スクリプトのパスを修正してください。 |
| 126115 | ERROR | script $PTH is a zero length file | **原因**: 指定したスクリプトは空のファイルです。<br><br>**処置**: スクリプトのファイルパスを修正し、スクリプトの内容を確認してください。 |
| 126117 | ERROR | script $PTH is not executable | **原因**: 指定したスクリプトは実行可能ではありません。<br><br>**処置**: スクリプトのファイルパスを修正し、スクリプトファイルの内容を確認したら、正常な実行権限があることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 126125 | ERROR | required template machine name is null | **原因**: 入力テンプレートマシン名が null です。<br><br>**処置**: 入力テンプレートマシン名を修正してください。 |
| 126130 | ERROR | required template resource tag name is null | **原因**: 入力テンプレートリソース {tag} が null です。<br><br>**処置**: 入力テンプレートリソースタグ名を修正してください。 |
| 126135 | ERROR | Unable to generate a new tag | **原因**:拡張の際に、「newtag」スクリプトを使用してターゲットノードでテンプレートタグ名と同じ新規タグを生成できませんでした。タグ名はすでに存在しています。<br><br>**処置**: 同じノードで重複するタグ名を避け、ログで詳細を確認してください。 |
| 126140 | ERROR | Unable to generate a new tag | **原因**: 拡張の際に、「newtag」スクリプトを使用してターゲットノードで入力ターゲットタグ名として新規タグを生成できませんでした。タグ名はすでに存在しています。<br><br>**処置**: 同じノードで重複するタグ名を避け、ログで詳細を確認してください。 |
| 126150 | ERROR | unable to remote copy template \"$_lscript\" script file | **原因**: テンプレートスクリプトファイルのリモートコピーに失敗しました。原因としては、テンプレートノードにテンプレートスクリプトファイルがない、またはスクリプトファイルを使用できない、あるいは「lcdrcp」プロセスで何らかのトランザクション障害が発生したことなどが考えられます。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: テンプレートスクリプトの有無/使用可能性を確認し、テンプレートノードへの接続を確かめてください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126155 | ERROR | failed to create resource instance on \"$TargetSysName\ | **原因**:「ins_create」によるリソースインスタンスの作成に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126160 | ERROR | failed to set resource instance state for \"$TargetTagName\" on \"$TargetSysName\ | **原因**: GenApp リソース拡張の際に、「ins_setstate」によるリソースインスタンスの状態の設定に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126170 | ERROR | getlocks failure | **原因**:リソース階層の作成時に管理ロックの取得に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126175 | ERROR | instance create failed | **原因**:「appins」による GenApp インスタンスの作成に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126180 | ERROR | unable to set state to OSU | **原因**: GenApp リソース作成の際に、「ins_setstate」によるリソースインスタンスの状態の設定に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126190 | ERROR | resource restore has failed | **原因**: GenApp リソースのリストアに失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126200 | ERROR | create application hierarchy rlslocks failure | **原因**: GenApp リソースの作成後にロックの解除に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126210 | ERROR | copy $ltype script $lscript failure | **原因**: リソース作成時に、ユーザ提供のスクリプトを適切な GenApp ディレクトリにコピーできませんでした。<br><br>**処置**: ユーザ提供のスクリプトの有無/使用可能性に加え、GenApp ディレクトリを確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126215 | ERROR | no $ltype script specified | **原因**: GenApp リソース作成時にユーザ定義スクリプトが見つかりません。<br><br>**処置**: 入力アクションスクリプトを確認し、リソース作成を再実行してください。 |
| 126220 | ERROR | no machine name specified | **原因**:GenApp リソース作成時に指定のマシン名が見つかりません。マシン名の入力内容が見つからないため、指定した |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ユーザスクリプトのコピーに失敗しました。<br><br>**処置**: マシン名の入力内容を確認し、リソース作成を再実行してください。 |
| 126225 | ERROR | no resource tag specified | **原因**:リソース作成時に指定のタグ名が見つかりません。<br><br>**処置**: ソースタグ名の入力内容を確認し、リソース作成を再実行してください。 |
| 126230 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: 不明な理由で GenApp リソースの拡張に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126235 | ERROR | $ERRMSG Required template machine name is null | **原因**: GenApp リソース拡張時にテンプレートマシン名の入力が見つかりません。<br><br>**処置**: テンプレートマシン名の入力内容を確認し、リソース拡張を再実行してください。 |
| 126240 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: GenApp リソース拡張時にテンプレートリソースタグ名の入力が見つかりません。<br><br>**処置**: テンプレートリソースタグ名の入力内容を確認し、リソース拡張を再実行してください。 |
| 126245 | ERROR | $ERRMSG Can not access extend for $AppType/$ResType resources on machine \"$TargetSysName\ | **原因**: ターゲットノードでの GenApp リソース拡張の際に、「extend」スクリプト |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | を見つけられませんでした。<br><br>**処置**:「extend」スクリプトの有無/使用可能性を確認し、GenApp リソース拡張を再実行してください。 |
| 126250 | ERROR | create application failure - ins_list failed | **原因**: GenApp リソース作成で「ins_list」を呼び出している際に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126255 | ERROR | create application failure - unable to generate a new tag | **原因**: GenApp リソース作成で新規タグの生成に失敗しました。<br><br>**処置**: 同じノードで重複するタグ名の使用を避けてください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126270 | ERROR | create application failure - ins_create failed | **原因**:「ins_create」による GenApp インスタンスの作成に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126275 | ERROR | create application failure - copy_actions failed | **原因**:「copy_actions」によるユーザ指定テンプレートスクリプトファイルのコピーに失敗しました。<br><br>**処置**: テンプレートスクリプトの有無/使用可能性を確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126290 | ERROR | Unable to obtain tag for resource with id $ID | **原因**:リカバリ時に入力 ID による GenA |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | pp リソースタグ名のフェッチに失敗しました。<br><br>**処置**: 入力 ID が正しいこと、および LCD の GenApp リソースの有無/使用可能性を確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126300 | ERROR | generic application recover script for $TAG was not found or was not executable | **原因**: リカバリ時に GenApp リソースのユーザ定義スクリプトの特定に失敗しました。<br><br>**処置**: ユーザ定義スクリプトの有無/使用可能性を確認し、GenApp リカバリプロセスを再実行してください。 |
| 126310 | ERROR | -t flag not specified | **原因**: GenApp リソースリストア時にリソースタグ名の入力が見つかりません。<br><br>**処置**: リソースタグ名の入力内容を確認し、リソースリストアを再実行してください。 |
| 126315 | ERROR | -i flag not specified | **原因**: GenApp リソースリストア時にリソース内部 ID の入力が見つかりません。<br><br>**処置**: リソース内部 ID の入力内容を確認し、リソースリストアを再実行してください。 |
| 126327 | ERROR | END timeout restore of \"$TAG\" (forcibly terminating) | |
| 126335 | ERROR | restore script \"$LCDAS/$APP_RESTOREDIR/$TAG\" was not found or is not executable | **原因**: リストア時に GenApp リソースのユーザ定義スクリプトの特定に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: ユーザ定義スクリプトの有無/使用可能性を確認し、GenApp リストアプロセスを再実行してください。 |
| 126340 | ERROR | -t flag not specified | **原因**: GenApp リソース削除時にリソースタグ名の入力が見つかりません。<br><br>**処置**: リソースタグ名の入力内容を確認し、リソース削除を再実行してください。 |
| 126345 | ERROR | -i flag not specified | **原因**: GenApp リソース削除時にリソース内部 ID の入力が見つかりません。<br><br>**処置**: リソース内部 ID の入力内容を確認し、リソース削除を再実行してください。 |
| 126357 | ERROR | END timeout remove of \"$TAG\" (forcibly terminating) | |
| 126365 | ERROR | remove script \"$LCDAS/$APP_REMOVEDIR/$TAG\" was not found or was not executable | **原因**: 削除時に GenApp リソースのユーザ定義スクリプトの特定に失敗しました。<br><br>**処置**: ユーザ定義スクリプトの有無/使用可能性を確認し、GenApp 削除プロセスを再実行してください。 |
| 126375 | ERROR | Script has hung checking \"$tag\". Forcibly terminating. | **原因**: 待機時間がユーザ定義のタイムアウトを超えたので、タグ名 {tag} の GenApp リソースの「quickCheck」スクリプトが強制的に終了しました。<br><br>**処置**: GenApp リソースのパフォーマンスを確認し、quickChecking を再起動してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 126380 | ERROR | Usage error: no tag specified | 原因: GenApp リソース quickCheck 時にリソースタグ名の入力が見つかりません。<br><br>処置: リソースタグ名の入力内容を確認し、リソース quickCheck を再試行してください。 |
| 126385 | ERROR | Internal error: ins_list failed on $tag. | 原因: quickCheck プロセスで、「ins_list」による入力タグ名別の GenApp リソース情報のフェッチに失敗しました。<br><br>処置: 入力タグ名を修正し、quickCheck プロセスを再実行してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126390 | FATAL | Failed to fork process to execute $userscript: $! | 原因: GenApp リソースの「quickCheck」プロセスで、ユーザ定義の「quickCheck」スクリプトを実行するためにプロセスを分岐させられませんでした。<br><br>処置: ユーザ定義の「quickCheck」スクリプトの有無/使用可能性を確認し、「quickCheck」プロセスを再実行してください。 |
| 126391 | ERROR | quickCheck has failed for \"$tag\". Starting recovery. | 原因: タグ名 {tag} の GenApp リソースが失敗することが、ユーザ定義の健全性監視スクリプトにより判定されました。「quickCheck」およびリカバリプロセスが開始されます。<br><br>処置: ローカルリカバリが完了したら、GenApp リソースのパフォーマンスを確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126392 | WARN | ${convtag}_TIMEOUT: This parameter is | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | old. This parameter will not be supported soon. | |
| 126400 | ERROR | -t flag not specified | 原因:GenApp リソース削除プロセスでリソースタグ名の入力が見つかりません。<br><br>処置: リソースタグ名の入力内容を確認し、リソース削除プロセスを再実行してください。 |
| 126405 | ERROR | -i flag not specified | 原因: GenApp リソース削除プロセスでリソース内部 ID の入力が見つかりません。<br><br>処置: リソース内部 ID の入力内容を確認し、リソース削除プロセスを再実行してください。 |
| 126478 | ERROR | Failed to create tag \'$new_leaf\'. | 原因:「creapphier」ユーティリティは、指定されたタグ {new_leaf} を作成できませんでした。<br><br>処置: /var/log/lifekeeper.log で「creapphier」からの追加メッセージを確認してください。 |
| 126479 | ERROR | Failed to extend tag \'$new_leaf\'. | 処置: /var/log/lifekeeper.log で拡張マネージャーからの追加エラーを確認してください。 |
| 126481 | ERROR | Failed to create dependency on \'$sys\' for \'$new_leaf\' to \'$hier{$leaf}{$sys}{‘Tag’}\'. | 処置: /var/log/lifekeeper.log で「dep_create」関数のエラーを確認してください。 |
| 126484 | ERROR | Tag \'$root_tag\' is not in-service. | 原因: 指定されたタグ{root_tag} は、クラスター内のどのノードでも In Service になっていません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 指定されたタグ{root_tag} をクラスター内の任意のノードで In Service にし、「create_terminal_leaf」を再実行してください。 |
| 126485 | ERROR | Tag \'$root_tag_1\' was not found, select the root tag for a hierarchy to add a terminal leaf resource. | **原因**:「create_terminal_leaf」に渡された最初のタグ {root_tag_1} が、ユーティリティが実行されたシステムで見つかりませんでした。<br><br>**処置**: リソースがクラスタ内のノードで In-service であり、すべてのノードへリソースが完全に拡張されていることを確認してください。 |
| 126486 | ERROR | Unable to create leaf tag from \'$tag\'. | **原因**: 一意の端末リーフタグを作成できませんでした。100回試行した後、一意の端末リーフタグを作成できませんでした。<br><br>**処置**: 複数のリーフタグと、問題を暗示している可能性のある /var/log/lifekeeper.log のエラーを確認してください。 |
| 126487 | ERROR | Tag \'$root_tag_2\' was not found, select the root tag for a hierarchy to add a terminal leaf resource. | **原因**:「create_terminal_leaf」に渡された2番目のタグ {root_tag_2} が、ユーティリティが実行されたシステムで見つかりませんでした。<br><br>**処置**: リソースがクラスタ内のノードで In-service であり、すべてのノードへリソースが完全に拡張されていることを確認してください。 |
| 126488 | ERROR | Tag \'$root_tag_1' is not extended to 3 or more systems. | **原因**: 指定されたタグ {root_tag_1} は、3つ以上のノードに拡張されていません。<br><br>**処置**: 指定したタグ {root_tag_1} を最低 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | でも3つのノードに拡張し、「create_terminal_leaf」を再度実行してください。 |
| 126489 | ERROR | $cmd does not support SDRS resources. | 原因: マルチサイト構成が検知されました。<br><br>処置: ありません |
| 126492 | ERROR | Remove resource $tag failed. | |
| 126494 | ERROR | Delete dependency failed on \'$sys\' for \'$tag\' to \'$parent\'. | |
| 126495 | ERROR | New tag \'$new_tag\' was modified during create, expected \'$new_leaf\'. | |
| 126496 | ERROR | Tag \'$root_tag_1\' is not in-service. | |
| 126497 | ERROR | Tag \'$root_tag_2\' is not in-service. | |
| 126498 | ERROR | Tag \'$root_tag_1\' and \'$root_tag_2' are not extended to same systems. | |
| 128005 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: オペレーティングシステムの信号 {signal} によって、{server} 上の {resource} の quickCheck が失敗しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128008 | ERROR | Usage: quickCheck -t <tag name> -i <id> | 原因: dmmp デバイスの quickCheck コマンドに不正な引数が指定されたため、コマンドが実行できませんでした。<br><br>処置: すべてのソフトウェアコンポーネントの正しいバージョンが適切にインストールされていることを確認してください。次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (quickChe |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ck 対象の dmmp デバイスリソースを特定する ID) |
| 128010 | ERROR | quickCheck for "%s" failed checks of underlying paths, initiate recovery. retry count=%s. | **原因**: dmmp キットが {count} 回の試行後、デバイスの quickCheck に失敗しました。保護されている dmmp リソースのリカバリが実行されます。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128021 | ERROR | unable to find device for uuid "%s". | **原因**: restore 操作時に一意の ID を使用してデバイスを見つけられませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。リストア対象の dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |
| 128025 | ERROR | Device "%s" failed to unlock. | **原因**: restore 操作時に正常に動作していない {device} が検出され、ロック解除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128026 | ERROR | Device "%s" failed to lock. | **原因**: リストア時に {device} をロックできませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要がありま |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | す。 |
| 128031 | ERROR | unable to find device for uuid "%s". | **原因**: remove 操作時に一意の ID を使用してデバイスを見つけられませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。削除対象の dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |
| 128034 | ERROR | Device "%s" failed to unlock. | **原因**: 削除時に {device} をロック解除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128036 | ERROR | unable to load existing information for device with uuid "%s". | **原因**: 一意の ID を使用してデバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |
| 128037 | ERROR | unable to load existing information for device "%s". | **原因**: 名前を使用してデバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。dmmp デバイスリソースを特定する正しいデバイス名でコマンドを再実行してください。 |
| 128038 | ERROR | unable to load existing information for device, no dev or uuid defined. | **原因**: 一意のデバイス ID もデバイス名も |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 指定されていないため、デバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。dmmp デバイスリソースを特定する正しいデバイス ID またはデバイス名でコマンドを再実行してください。 |
| 128041 | ERROR | unable to load existing information for device with uuid "%s". | **原因**: 一意の ID を使用してデバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |
| 128057 | ERROR | All paths are failed on "%s". | **原因**: LifeKeeper は、保護対象の dmmp デバイスへのすべてのパスが障害状態にあることを検出しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128058 | ERROR | could not determine registrations for "%s"! All paths failed. | **原因**: LifeKeeper は、保護対象の dmmp {device} の登録を特定できませんでした。dmmp {device} へのすべてのパスが障害状態にあります。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128059 | WARN | path "%s" no longer configured for "%s", remove from path list. | **原因**:LifeKeeper が検出した保護対象の {device} への {path} は、現在は有効ではなく、パスリストから削除されます。<br><br>**処置**: 隣接するログメッセージで詳細お |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | よび関連メッセージを確認してください。 |
| 128060 | WARN | registration failed on path "%s" for "%s". | 原因: LifeKeeper は、保護対象の dmmp {device} の {path} の登録を特定できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128062 | ERROR | all paths failed for "%s". | 原因: LifeKeeper は、保護対象の dmmp {device} の有効なパスを確認できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128072 | ERROR | The daemon "%s" does not appear to be running and could not be restarted. Path failures may not be correctly handled without this daemon. | 原因: LifeKeeper は dmmp デーモンが実行されていることを確認できなかったため、デーモンを再起動できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128078 | ERROR | "%s" resource type is not installed on "%s". | 原因: dmmp デバイスをサポートするための Device Mapper Multipath Recovery Kit がシステムにインストールされていません。<br><br>処置: steeleye-lkDMMP Device Mapper Multipath Recovery Kit rpm をシステムにインストールしてください。 |
| 128083 | ERROR | This script must be executed on "%s". | 原因: dmmp デバイスリソースを作成す |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | るための devicehier スクリプトの引数に不正なシステム名が指定されました。<br><br>**処置**: クラスタノードとコミュニケーションパスが適切に設定されていることを確認してください。devicehier スクリプトに正しいシステム名を指定してください。この名前は、コマンドを実行するシステムの名前に一致する必要があります。 |
| 128084 | ERROR | The device %s is not active. | **原因**: リソース作成時に LifeKeeper は、指定された {device} をシステム上の有効なデバイスとして見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (作成する dmmp デバイスリソースを特定する ID) |
| 128086 | ERROR | Failed to create "%s" hierarchy. | **原因**: LifeKeeper は、{device} のためのリソース階層を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128088 | ERROR | Error creating resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} で {tagname} を使用してリソースを作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要がありま |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | す。 |
| 128090 | ERROR | Failed to create dependency "%s"-"%s" on system "%s". | 原因: LifeKeeper は、{system} で依存関係 {resource tag name} – {resource tag name} を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128091 | ERROR | Error creating resource "%s" on server "%s" | 原因: LifeKeeper は、{system} で {resource} を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128101 | ERROR | "%s" constructor requires a valid argument. | 原因: LifeKeeper は、dmmp リソースのオブジェクトを作成できませんでした。<br><br>処置: 次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (dmmp デバイスリソースを特定する ID) |
| 128102 | ERROR | Invalid tag "%s". | 原因: 指定されたタグ名のリソースインスタンスが見つかりませんでした。<br><br>処置: リソースが正しく設定されていることを確認してください。次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (dmmp デバイスリソースを特定する ID) |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 128111 | ERROR | Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations. | 原因: LifeKeeper は {device} の登録の取得に失敗し、メッセージ「bad field in Persistent reservation in cdb」を出力しました。<br><br>処置: このストレージでパーシステントリザベーションがサポートされているか確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128112 | ERROR | Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations. | 原因: LifeKeeper は {device} の登録の取得に失敗し、メッセージ「illegal request」を出力しました。<br><br>処置: このストレージでパーシステントリザベーションがサポートされているか確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128136 | ERROR | A previous quickCheck with PID "%s" running for device "%s" has been terminated. | 原因: LifeKeeper は、dmmp リソースの restore 操作時に前の quickCheck がまだ実行中であることを検出しました。LifeKeeper は quickCheck を終了しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128137 | ERROR | SCSI reservation conflict on %s during LifeKeeper resource initialization. Manual intervention required. | 原因: LifeKeeper は、dmmp リソースの restore 操作時に {device} で SCSI リザベーションコンフリクトを検出しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。手動による介入および {device} でのリザベーションコンフリクトの解消が必要です。 |
| 128138 | ERROR | unable to clear registrations on %s. | **原因**: LifeKeeper は、{device} 上のすべての登録を消去できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128140 | WARN | registration failed on path %s for %s. | **原因**: LifeKeeper は、{device} の {path} の登録を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128143 | ERROR | reserve failed (%d) on %s. | **原因**: LifeKeeper は、{device} 上の {resource} の登録を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128145 | ERROR | The server ID "%s" returned by "%s" is not valid. | **原因**: LifeKeeperは有効なホスト {ID} を生成できませんでした。<br><br>**処置**: デバイスを登録するための ID は、クラスタ内のサーバを一意に特定する 1 ～ 12 桁の 16 進数で構成されま |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | す。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128146 | ERROR | device failure on %s. SYSTEM HALTED. | 原因:LifeKeeper は {device} で障害を検出しました。サーバを再起動します。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128148 | ERROR | device failure on %s. SYSTEM HALTED DISABLED. | 原因: LifeKeeper は {device} で障害を検出しました。LifeKeeper の設定によって、再起動はスキップされました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。「SCSIHALT」設定を有効にして、いかなるデバイス障害が検出されたときも再起動が行われるようにしてください。 |
| 128149 | ERROR | device failure or SCSI Error on %s. SEND EVENT DISABLED. | 原因: LifeKeeper は {device} で障害を検出しました。LifeKeeper の設定によって、イベント生成はスキップされました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。「SCSIEVENT」設定を有効にして、いかなるデバイス障害が検出されたときも sendevent を使用できるようにしてください。 |
| 128150 | ERROR | %s does not have EXCLUSIVE access to %s, halt system. | 原因: LifeKeeper は {server} の {device} でリザベーションコンフリクトを検出しました。サーバを再起動します。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128151 | ERROR | %s does not have EXCLUSIVE access to %s, halt system DISABLED. | **原因**: LifeKeeper は {server} の {device} でリザベーションコンフリクトを検出しました。LifeKeeper の設定によって、再起動はスキップされました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。「RESERVATIONCONFLICT」設定を有効にして、いかなるリザベーションコンフリクトが検出されたときも再起動が行われるようにしてください。 |
| 128154 | WARN | unable to flush buffers on %s. | **原因**: LifeKeeper は、dmmp リソースの削除時に {device} のバッファを消去できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128157 | WARN | %s utility not found, limited healthcheck for %s. | **原因**: LifeKeeper は、{device} の健全性チェックを行う「dd」ユーティリティを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128160 | ERROR | %s failed to read %s. | **原因**: LifeKeeper は、{utility} の使用中に {device} のディスク I/O テストに失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128163 | ERROR | Registration ID "%s" for "%s" is not valid. | **原因**: LifeKeeper は {device} の有効な登録 {ID} の生成に失敗しました。<br><br>**処置**: デバイスを登録するための ID は、デバイスへのパスに由来する 4 桁の 16 進数で構成されます。隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128170 | ERROR | Usage: canextend <Template system name> <Template tag name> | |
| 128500 | ERROR | Usage error | **原因**: dmmp デバイスの restore コマンドに不正な引数が指定されたため、コマンドが実行できませんでした。<br><br>**処置**: 次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (リストアする dmmp デバイスリソースを特定する ID) |
| 128504 | ERROR | "%s" resource type is not installed on "%s". | **原因**: dmmp デバイスをサポートするための Device Mapper Multipath Recovery Kit がシステムにインストールされていません。<br><br>**処置**: steeleye-lkDMMP Device Mapper Multipath Recovery Kit rpm をシステムにインストールしてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 128506 | ERROR | Usage error | **原因**: dmmp デバイスの devShared コマンドに不正な引数が指定されたため、コマンドが実行できませんでした。<br><br>**処置**: 次の正しい引数を指定してコマンドを再実行してください。<Template Resource System Name> および <Template Resource Tag> (作成する dmmp デバイスリソースを特定するタグ) |
| 128507 | FATAL | This script must be executed on "%s". | **原因**: dmmp デバイスリソースを作成するための devicehier スクリプトの引数に不正なシステム名が指定されました。<br><br>**処置**: devicehier スクリプトに正しいシステム名を指定してください。この名前は、コマンドを実行するシステムの名前に一致する必要があります。 |
| 128511 | ERROR | Failed to get the ID for the device "%s". Hierarchy create failed. | **原因**: dmmp デバイスリソースを作成するための devicehier スクリプトは、指定されたデバイスの SCSI ID を特定できませんでした。<br><br>**処置**: 指定されたデバイスパスが存在すること、およびそのパスがサポート対象の SCSI ストレージアレイのパスであることを確認してください。 |
| 128512 | ERROR | Failed to get the disk ID for the device "%s". Hierarchy create failed. | **原因**: dmmp ディスクリソースを作成するための devicehier スクリプトは、指定されたディスクの SCSI ID を特定できませんでした。<br><br>**処置**: 指定されたデバイスパスが存在すること、およびそのパスがサポート対象の SCSI ストレージアレイのパスであることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 128513 | ERROR | Failed to create the underlying resource for device "%s". Hierarchy create failed. | **原因**: ベースとなる dmmp ディスクリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128515 | ERROR | Error creating resource "%s" on server "%s" | **原因**: dmmp デバイスリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128517 | ERROR | Failed to create dependency "%s"-"%s" on system "%s". | **原因**: dmmp デバイスリソースと dmmp ディスクリソースとの間の親 / 子の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128519 | ERROR | Error creating resource "%s" on server "%s" | **原因**: 新しく作成した dmmp デバイスリソースを in service にする操作が失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128521 | ERROR | Either TEMPLATESYS or TEMPLATETAG argument missing | **原因**: dmmp デバイスリソースの extend |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | コマンドに不正な引数が指定されました。<br><br>**処置**:テンプレートシステムおよびタグの正しい名前を指定して dmmp デバイスリソースの extend を再実行してください。 |
| 128540 | ERROR | Usage error | **原因**: dmmp デバイスの SCSI ID を取得する getId コマンドに不正な引数が指定されました。<br><br>**処置**: 次の正しい引数を指定してコマンドを再実行してください。-i <device path> または -b <device ID>。 |
| 128541 | ERROR | Usage error | **原因**: dmmp デバイスリソースを削除するコマンドに不正な引数が指定されました。<br><br>**処置**: 次の正しい引数を指定してコマンドを再実行してください。-t <dmmp device resource tag>。 |
| 128543 | ERROR | device node "$dev" does not exist. | **原因**: dmmp デバイスリソースをリストアするために必要なデバイスノードが存在しません。リストア時に udev デバイスの作成に割り当てられている待機時間を越えました。<br><br>**処置**: udev がデバイスを作成した後に dmmp デバイスリソースの restore を再実行してください。 |
| 128544 | ERROR | Usage error | **原因**: dmmp デバイスリソースを out of service にする remove コマンドに不正な引数が指定されました。<br><br>**処置**: 次の正しい引数を指定してコマン |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ドを再実行してください。-t <dmmp device resource tag>。 |
| 128550 | ERROR | Failed to get path information.(tag=\"$opt_t\" | |
| 128551 | ERROR | Failed to get the status of path.(tag=\"$opt_t\" | |
| 128552 | ERROR | Failed to check the status of path.(tag=\"$opt_t\" | |
| 128553 | ERROR | Failed to get the resource information.(tag=\"$opt_t\" | |
| 129100 | FATAL | Failed to load instance from LifeKeeper. | **原因**: 無効なリソースタグまたは ID が指定されました。<br><br>**処置**: タグまたは ID が有効であることを確認し、コマンドを再実行してください。 |
| 129103 | FATAL | No resource matches tag "$self->{'tag'}". | **原因**: 無効なリソースタグが指定されました。<br><br>**処置**: タグが有効であることを確認し、コマンドを再実行してください。 |
| 129104 | FATAL | An error occurred setting LifeKeeper resource information | **原因**: LifeKeeper で内部エラーが発生しました。 |
| 129110 | ERROR | Could not get the Elastic Network Interface ID for $dev | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129111 | ERROR | Failed to get Allocation ID of Elastic IP "$elasticIp". | **原因**: EC2 API の呼び出しに失敗しまし |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | た。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129113 | ERROR | Failed to get my instance ID. | **原因**: EC2 インスタンスのメタデータのアクセスに失敗しました。<br><br>**処置**: Amazon コンソールを確認し、操作を再試行してください。 |
| 129114 | ERROR | Failed to get ENI ID. | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129116 | ERROR | Failed to associate Elastic IP "$self->{'EIP'}" on "$self->{'DEV'}". | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129118 | WARN | $self->{'EIP'} is not associated with any instance. | **原因**: Elastic IP がインスタンスに関連付けられていません。<br><br>**処置**: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129119 | WARN | $self->{'EIP'} is associated with another instance. | **原因**: Elastic IP が別のインスタンスに関 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 連付けられています。<br><br>**処置**: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129120 | ERROR | Failed to recover Elastic IP. | **原因**: EC2 API が Elastic IP の関連付けに失敗しました。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129121 | ERROR | Recovery process ended but Elastic IP is not associated with this instance. Please check AWS console. | **原因**: EC2 API が Elastic IP の関連付けに失敗しました。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129122 | ERROR | Error creating resource \"$target_tag\" with return code of \"$err\". | **原因**: LifeKeeper がサーバ上にリソースインスタンスを作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129123 | ERROR | Failed to get ENI ID. | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129124 | WARN | $self->{'EIP'} is associated with another network interface. | **原因**: Elastic IP は正しいインスタンスに関連付けられていますが、ENI が正しく |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ありません。<br><br>**処置**: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129125 | ERROR | Link check failed for interface '$dev'. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: ネットワークインターフェースを確認し、リンクをアップ状態にしてください。 |
| 129126 | ERROR | Link check failed for interface '$dev'. Reason: down slave. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: ネットワークインターフェースを確認し、リンクをアップ状態にしてください。 |
| 129129 | WARN | The link for network interface '$self->{'DEV'}' is down. Attempting to bring the link up. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: LifeKeeper は、リンクをアップ状態にして Elastic IP とインターフェースを関連付けることにより、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129130 | ERROR | Failed to modify "$opt_t" to end pint URL "$endpoint". | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 129137 | ERROR | The link for network interface '$self->{'DEV'}' is still down. | **原因**: LifeKeeper がリンクをアップ状態にできませんでした。<br><br>**処置**: インターフェースが有効であり、アップ状態であることを確認してください。隣接するログメッセージで詳細を確認してください。 |
| 129139 | WARN | The link for network interface '$self->{'DEV'}' is down. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: ネットワークインターフェースを確認し、リンクをアップ状態にしてください。 |
| 129140 | ERROR | Could not get ENI ID for $self->{IP}. | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129142 | ERROR | Failed to update route table | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129143 | ERROR | You must have exactly one IP address resource as the parent of the RouteTable EC2 resource. Please reconfigure your resource hierarchy. | **原因**: ルートテーブルの EC2 リソースは親として IP リソースを 1 つのみ持つ必要があります。<br><br>**処置**: 必要に応じてリソース階層を修復してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 129144 | ERROR | $func called with invalid timeout: $timeout | 原因: /etc/default/LifeKeeper ファイルに無効なタイムアウト値が指定されました。<br><br>処置: /etc/default/LifeKeeper 内にあるすべての EC2_*_TIMEOUT 設定が正しいことを確認してください。 |
| 129145 | ERROR | $func action timed out after $timeout seconds | 原因: タイムアウト期間中に動作が完了しませんでした。<br><br>処置: その動作の EC2_*_TIMEOUT 値 (/etc/default/LifeKeeper 内) を増加することを検討してください。 |
| 129146 | ERROR | failed to run $func with timeout: $@ | 原因: これは内部エラーです。 |
| 129148 | ERROR | Amazon describe-route-tables call failed (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129150 | ERROR | Elastic IP \"$elasticIp\" is associated with another instance. | 原因: Elastic IP が正しいインスタンスに関連付けられていません。<br><br>処置: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129151 | ERROR | Could not get the Association ID for Elastic IP "$elasticIp". | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソー |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ルを確認し、操作を再試行してください。 |
| 129152 | ERROR | Failed to disassociate Elastic IP "$self->{'EIP'}" on "$self->{'DEV'}". | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129153 | ERROR | Failed to disassociate Elastic IP "$elasticIp", (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129154 | ERROR | Amazon describe-addresses call failed (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129155 | ERROR | Amazon describe-address call failed (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129157 | ERROR | curl call failed (err=%s)(output=%s) | 原因: EC2 インスタンスのメタデータのアクセスに失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: Amazon コンソールを確認し、操作を再試行してください。 |
| 129159 | ERROR | Amazon associate-address call failed (err=%s)(output=%s) | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129160 | ERROR | Amazon describe-addresses call failed (err=%s)(output=%s) | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129161 | ERROR | Error deleting resource "$otherTag" on "$otherSys" with return code of "$err". | **原因**: LifeKeeper がサーバ上のリソースインスタンスを削除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129162 | ERROR | Could not getRouteTablesByIP | |
| 129163 | ERROR | Could not getRouteTablesByIP | |
| 129164 | ERROR | [$SUBJECT event] mail returned $err | |
| 129165 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 129167 | ERROR | snmptrap returned $err for Trap 180 | |
| 129168 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 129170 | ERROR | This resource is in the old format. Please update. | |
| 129403 | ERROR | END failed create of $TAG due to a $sig signal | **原因**: 作成プロセスがシグナルによって |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 中断されました。 |
| 129409 | ERROR | The IP resource $IP_RES is not \"ISP\". | **原因**: IP リソースが in service ではありません。<br><br>**処置**: リソースを in service にして、操作をもう一度行ってください。 |
| 129410 | ERROR | Could not find IP resource $IP_RES | **原因**: IP リソースが存在していることを確認し、操作を再試行してください。 |
| 129412 | ERROR | EC2 resource $ID is already protected | **原因**: 指定した ID を持つリソースはすでに存在しています。<br><br>**処置**: 残っている古いリソースをすべてクリーンアップしてから、新しいリソースを再作成してください。 |
| 129416 | ERROR | Error creating resource "$TAG" with return code of "$lcderror". | **原因**: LifeKeeper がリソースインスタンスを作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129418 | ERROR | Dependency creation between "$IP_RES" and "$TAG" failed with return code of "$lcderror". | **原因**: LifeKeeper がリソースの依存関係を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129420 | ERROR | In-service failed for tag "$TAG". | **原因**: LifeKeeper がリソースインスタンスを in service にできませんでした。<br><br>**処置**: 隣接するログメッセージで詳細お |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | よび関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129423 | ERROR | Could not get ENI ID for $dev. | |
| 129425 | ERROR | Failed to update route table | |
| 129426 | ERROR | In-service (dummy) failed for tag "$TAG". | |
| 129800 | ERROR | canextend checks failed for \"$self->{'tag'}\" (err=$ret) | **原因**: ターゲットサーバで拡張前のチェックに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129801 | ERROR | canextend checks failed for "$self->{'tag'}". EC2_HOME "$self->{EC2_HOME}" does not exist on $me. | |
| 133106 | ERROR | You must have exactly one IP address resource as the child of the route53 resource. Please reconfigure your resource hierarchy. | |
| 133111 | ERROR | Failed to init the object. | |
| 133114 | ERROR | Failed start Route53 resource $self->{Tag} | |
| 133115 | ERROR | Failed start Route53 resource $self->{Tag} | |
| 133116 | WARN | Could not Get A Record Value Address : $aAWSresult | |
| 133117 | WARN | aws call failed : $aAWSresult | |
| 133118 | WARN | aws call failed : $upsertResp | |
| 133119 | ERROR | Could not Update and Create A Record Set : $upsertResp | |
| 133120 | WARN | Could not get Route53 API batch request ID form UPSERT response XML data : $upsertResp | |
| 133121 | WARN | Could not Get A Record Value Address : $aAWSresult | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 133122 | WARN | aws call failed : $statusResult | |
| 133123 | ERROR | Faild check change batch request status : $statusResult | |
| 133126 | WARN | Failed to Route53 API response : $rc | |
| 133128 | WARN | Failed to Route53 API response | |
| 133129 | ERROR | Route53 resource $self->{Tag} is stopped | |
| 133130 | ERROR | $self->{Tag} is in the old format. Please update. | |
| 133601 | ERROR | $usage | |
| 133602 | ERROR | Template resource "$template_tag" on server \"$template_sys" does not exist | |
| 133608 | ERROR | Template resource "$template_tag" on server "$template_sys" does not exist | |
| 133614 | ERROR | Error creating resource "$Tag" on server "$me" | |
| 133617 | ERROR | END failed hierarchy extend of resource  $Tag with return value of $ecode. | |
| 133619 | ERROR | $usage | |
| 133620 | ERROR | END failed extend of "$Tag" due to a "$sig" signal | |
| 133700 | ERROR | $usage | |
| 133701 | ERROR | END failed create of "$Route53Tag" due to a "$sig" signal | |
| 133703 | ERROR | Unable to getlocks on $me during resource create. | |
| 133704 | ERROR | The route53 on $Route53HostName is already protected by LifeKeeper. | |
| 133706 | ERROR | Error creating resource $Route53Tag. Error ($rc) | |
| 133708 | ERROR | Dependency create failed between $Route53Tag and $Route53IPResTag. Error ($rc). | |
| 133710 | ERROR | In-service attempted failed for tag $Route53Tag. | |
| 133813 | ERROR | Usage: $usage | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 133815 | ERROR | The host name "$hostname" is too long. | |
| 133816 | ERROR | The host name "$hostname" is too short. | |
| 133817 | ERROR | The host name "$hostname" contains invalid character. | |
| 133818 | ERROR | The first character must be an alpha character. | |
| 133819 | ERROR | The last character must not be a minus sign or period. | |
| 133826 | ERROR | Host Name cannot be blank. | |
| 134003 | ERROR | catch a \"$sig\" signal | **原因**: create処理がシグナルによって中断されました。<br><br>**処置**: 前後のログメッセージを確認してください。 |
| 134004 | ERROR | Unable to getlocks on $server during resource create. Error ($rc) | **原因**: リソース階層の作成時に管理ロックの取得に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134005 | ERROR | The service \"$serviceName\" is not supported on $server. Error ($rc) | **原因**: サービスが存在しないか保護できないサービスです。<br><br>**処置**: 適切なサービス名を入力してください。 |
| 134006 | ERROR | The service \"$serviceName\" is already protected on $server. | **原因**: すでに保護されているサービスです。<br><br>**処置**: このサービスを保護するためのリソースを作成することはできません。 |
| 134007 | ERROR | Error creating resource $tag. Error ($rc) | **原因**: QSPリソースの作成に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 前後のログメッセージを確認して、エラーの原因を取り除いてください。 |
| 134011 | ERROR | In-service attempted failed for tag $tag. | **原因**: QSPリソースの起動に失敗しました。<br><br>**処置**: 保護したいサービスに関連したログを確認して、問題を解決してください。 |
| 134015 | ERROR | Unable to rlslocks on $server during resource create. Error ($rc) | **原因**: QSPリソースの作成後にロックの解除に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134103 | ERROR | Template resource \"$template_tag\" on server \"$template_sys\" does not exist | **原因**: テンプレートシステムでリソースを見つけられません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 134104 | ERROR | Template resource \”$template_tag\” on server \”$template_sys\” is not QSP resource (app=$ins[1], res=$ins[2]) | **原因**: テンプレートリソースはQSPリソースではありません。<br><br>**処置**: テンプレートリソースと同じタイプに拡張してください。 |
| 134105 | ERROR | The service \"$service\" is not supported on $me. Error ($check) | **原因**: 拡張先のサーバにサービスが存在しません。<br><br>**処置**: 拡張先サーバにサービスをインストールしてから拡張してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 134106 | ERROR | The service \"$service\" is already protected on $me. | **原因**: 拡張先サーバに同じIDのリソースが既にあります。<br><br>**処置**: 同じサービスのリソースを作成することはできません。 |
| 134203 | ERROR | catch a \"$sig\" signal | **原因**: リソース拡張処理がシグナルによって中断されました。<br><br>**処置**: 前後のログメッセージを確認してください。 |
| 134204 | ERROR | Template resource \"$template_tag\" on server \"$template_sys\" does not exist | **原因**: テンプレートシステムでリソースを見つけられません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 134208 | ERROR | Error creating resource \"$tag\" on server \"$me\" | **原因**: LifeKeeperはターゲットサーバーにリソースインスタンスを作成できませんでした。<br><br>**処置**: 前後のログメッセージを確認して、エラーの原因を取り除いてください。 |
| 134401 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | **原因**: サービスの起動処理が規定時間を超えても終了しません。<br><br>**処置**: 保護しているサービスについて確認し、restore操作を再試行してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134405 | FATAL | Failed to fork process to execute service command: $! | **原因**: forkが失敗しました。これはシステムエラーです。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: forkが失敗する理由を特定してください。 |
| 134407 | ERROR | service command has failed for \"$tag\" | **原因**: serviceコマンドの実行に失敗しました。<br><br>**処置**: startオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原因を取り除いてください。 |
| 134501 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | **原因**: サービスの停止処理が規定時間を超えても終了しません。<br><br>**処置**: 保護しているサービスについて確認し、remove操作を再試行してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134505 | FATAL | Failed to fork process to execute service command: $! | **原因**: forkが失敗しました。これはシステムエラーです。<br><br>**処置**: forkが失敗する理由を特定してください。 |
| 134507 | ERROR | service command has failed for \"$tag\" | **原因**: serviceコマンドの実行に失敗しました。<br><br>**処置**: stopオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原因を取り除いてください。 |
| 134601 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | **原因**: サービスの監視処理が規定時間を超えても終了しません。<br><br>**処置**: 保護しているサービスについて確 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134605 | FATAL | Failed to fork process to execute service command: $! | 原因: forkが失敗しました。これはシステムエラーです。<br><br>処置: forkが失敗する理由を特定してください。 |
| 134607 | ERROR | service command has failed for \"$tag\" | 原因: serviceコマンドの実行に失敗しました。<br><br>処置: statusオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原因を取り除いてください。 |
| 134701 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | 原因: サービスの回復処理が規定時間を超えても終了しません。<br><br>処置: 保護しているサービスについて確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134706 | FATAL | Failed to fork process to execute service command: $! | 原因: forkが失敗しました。これはシステムエラーです。<br><br>処置: forkが失敗する理由を特定してください。 |
| 134708 | ERROR | service command has failed for \"$tag\" | 原因: serviceコマンドの実行に失敗しました。<br><br>処置: startオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 因を取り除いてください。 |
| 134803 | ERROR | tag \"$tag\" does not exist on server \"$me\" | 原因: 指定されたタグが存在しません。これは内部エラーです。 |
| 134804 | ERROR | app type \”$ins[1]\” is not $app | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134805 | ERROR | res type \”$ins[2]\” is not $res | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134823 | ERROR | tag \"$tag\" does not exist on server \"$me\" | 原因: 指定されたタグが存在しません。これは内部エラーです。 |
| 134824 | ERROR | app type \”$ins[1]\” is not $app | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134825 | ERROR | res type \”$ins[2]\” is not $res | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134843 | ERROR | tag \"$tag\" does not exist | 原因: 指定されたタグが存在しません。これは内部エラーです。 |
| 134844 | ERROR | app type \”$ins[1]\” is not $app | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134845 | ERROR | res type \”$ins[2]\” is not $res | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 135002 | ERROR | Socket connection to $host failed (not a timeout): $@ | |
| 135802 | ERROR | object[%d] not iterated for %lld ms. | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 135806 | ERROR | qwk_config() failed. | |
| 135807 | ERROR | thread_initialize() failed. | |
| 135808 | ERROR | state_monitor_initialize() failed. | |
| 135809 | ERROR | state_monitor() failed. | |
| 135810 | ERROR | start_server() failed. | |
| 135812 | ERROR | ‘fopen’ for %s failed: %m | |
| 135813 | ERROR | ‘qwk_object_path=’ line cannot be found for %s. | |
| 135814 | ERROR | ‘%s’ is unknown config. | |
| 135815 | ERROR | configuration of hbeattime ‘%d’ is incorrect. | |
| 135816 | ERROR | configuration of numhbeats ‘%d’ is incorrect. | |
| 135817 | ERROR | configuration of timeout_multiplier ‘%d’ is incorrect. | |
| 135818 | ERROR | configuration of lcmhbeattime ‘%d’ is incorrect. | |
| 135819 | ERROR | configuration of lcmnumhbeats ‘%d’ is incorrect. | |
| 135820 | ERROR | configuration of qwk_object_type is incorrect. | |
| 135821 | ERROR | configuration of my_node is incorrect. | |
| 135822 | ERROR | configuration of number_of_node ‘%d’ is incorrect. | |
| 135823 | ERROR | configuration of number_of_object ‘%d’ is incorrect. | |
| 135824 | ERROR | configuration of object node is incorrect. | |
| 135825 | ERROR | configuration of object path is incorrect. | |
| 135826 | ERROR | my_node is not include in qwk_objects. | |
| 135827 | ERROR | ‘open’ for %s failed: %m | |
| 135828 | ERROR | ‘read’ for %s failed: %m | |
| 135829 | ERROR | ‘popen’ for %s failed: %m | |
| 135830 | ERROR | ‘open’ for %s failed: %m | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 135831 | ERROR | ‘write’ for %s failed: %m | |
| 135832 | ERROR | ‘popen’ for %s failed: %m | |
| 135833 | ERROR | (bug) buffer overflow | |
| 135834 | ERROR | (bug) data is corrupted | |
| 135835 | ERROR | ‘signature=’ line cannot be found. | |
| 135836 | ERROR | signature ‘%s’ does not match. | |
| 135837 | ERROR | ‘local_node=’ line cannot be found. | |
| 135838 | ERROR | local_node ‘%s’ does not match. | |
| 135839 | ERROR | ‘time=’ line cannot be found. | |
| 135840 | ERROR | ‘sequence=’ line cannot be found. | |
| 135841 | ERROR | sequence ‘%s’ scan failed. | |
| 135842 | WARN | ‘node=’ line cannot be found. index=%d | |
| 135843 | ERROR | ‘commstat=’ line cannot be found. inde x=%d | |
| 135844 | ERROR | ‘checksum=’ line cannot be found. | |
| 135845 | ERROR | checksum ‘%s’ scan failed. | |
| 135846 | ERROR | checksum does not match. | |
| 135849 | ERROR | qwk object was not found. | |
| 135851 | ERROR | failed to read qwk object. | |
| 135852 | ERROR | failed to decode node_info. | |
| 135854 | WARN | sequence backed down from %llu to %llu. | |
| 135855 | ERROR | ‘malloc’ for %zu failed: %m | |
| 135856 | ERROR | thread_create() failed. index=%d | |
| 135870 | ERROR | (bug) data is corrupted | |
| 135871 | ERROR | format error in request. | |
| 135872 | ERROR | format error in quorum_verify request. lke vent cannot be found. | |
| 135873 | ERROR | format error in witness_verify request. lke vent cannot be found. | |
| 135874 | ERROR | format error in witness_verify request. tar get_node cannot be found. | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 135875 | ERROR | ‘%s’ is unknown command. | |
| 135877 | ERROR | qwk_receive() did not receive full header. Close socket. | |
| 135878 | ERROR | Request is too long. Close socket. | |
| 135879 | ERROR | qwk_receive() did not receive full request. Close socket. | |
| 135880 | ERROR | do_request() failed. | |
| 135881 | ERROR | qwk_send() did not send full header. Clos e socket. | |
| 135882 | ERROR | qwk_send() did not send full response. Cl ose socket. | |
| 135884 | ERROR | qwk_accept() failed. | |
| 135885 | ERROR | thread_create() failed. | |
| 135886 | ERROR | cannot create socket. | |
| 135887 | ERROR | ‘bind’ failed: %m | |
| 135888 | ERROR | ‘listen’ failed: %m | |
| 135889 | ERROR | create_sockets() failed. | |
| 135890 | ERROR | thread_create() failed. | |
| 135891 | ERROR | ‘pthread_attr_init’ failed: %m | |
| 135892 | ERROR | ‘pthread_attr_setstacksize’ failed: %m | |
| 135893 | ERROR | ‘pthread_create’ failed: %m | |
| 135894 | ERROR | requst is too long. header.size=%zu | |
| 135895 | ERROR | qwk_send() failed for header. | |
| 135896 | ERROR | qwk_send() failed for request. | |
| 135897 | ERROR | qwk_send() failed for termination. | |
| 135898 | ERROR | qwk_receive() did not receive full header. | |
| 135899 | ERROR | Response buffer is not enough large. Ser ver sent %zu bytes. | |
| 135900 | ERROR | qwk_receive() did not receive full respons e. | |
| 135901 | ERROR | ‘%s’ is unknown command. | |
| 135903 | ERROR | Cannot create socket. | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 135904 | ERROR | ‘connect’ failed: %m | |
| 135905 | ERROR | request_send() failed. | |
| 135906 | ERROR | request_receive() failed. | |
| 135907 | ERROR | ‘%s’ is unknown lkevent. | |
| 135908 | ERROR | ‘%s’ is unknown qwktype. | |
| 135909 | ERROR | ‘%s’ is unknown node state. | |
| 135910 | ERROR | ‘%s’ is unknown quorum state. | |
| 135911 | ERROR | ‘accept’ failed: %m | |
| 135912 | ERROR | You must install the LifeKeeper license key for Storage Quorum Witness Kit. | |
| 135999 | ERROR | COMMAND OUTPUT: $LKROOT/lkadm/bin/qwk_storage_server $LKROOT/config/qwk-storage.cfg | |
| 136002 | ERROR | Usage: $usage | 原因: SAP HANA createスクリプトに無効な引数が指定された。<br><br>処置: 次の形式で適切な引数を指定してください：<br><リソースタグ> <SAP SID> <HDBインスタンス> [スイッチバックタイプ] [仮想IPリソースタグ] |
| 136003 | ERROR | END failed create of resource $tag on server $me with return value of $errcode. | 原因: SAP HANAリソースの作成中に障害が発生した。<br><br>処置: SAP HANAシステムレプリケーションがプライマリとセカンダリの両方のレプリケーションサイトで完全に構成および有効化されていることを確認し、リソース作成操作を再試行してください。 |
| 136005 | ERROR | An unknown error has occurred in utility rlslocks on server $me. View the LifeKeeper logs for details and retry the operation. | 原因: SAP HANAリソースの作成中にLifeKeeper rlslocksユーティリティで問題が発生する。<br><br>処置: LifeKeeperログファイルで見つか |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | った問題を解決し、リソース作成操作を再試行してください。 |
| 136006 | ERROR | END failed create of resource $tag on server $me with signal $sig. | 原因: SAP HANAリソース作成中のシグナルによる割込み。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136008 | ERROR | An unknown error has occurred in utility getlocks on server $me. View the LifeKeeper logs for details and retry the operation. | 原因: SAP HANAリソースの作成中にLifeKeeper getlocksユーティリティで問題が発生する。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース作成操作を再試行してください。 |
| 136009 | ERROR | The SAP HANA product was not found in the directory $obj->{'hana_util_path'} on server $me. Verify that SAP HANA is correctly installed and that all necessary file systems are mounted. | 原因: リソースの作成中に、必須のSAP HANAバイナリが見つからなかった。<br><br>処置: クラスター内のすべてのサーバーでSAP HANAが正しくインストールされ、構成されていることを確認してください。 |
| 136010 | ERROR | Failed to create resource as id $id already exists on system $me. | 原因: 指定されたIDのLifeKeeperリソースがすでにシステムに存在する。<br><br>処置: SAP HANAデータベースがLifeKeeperによってすでに保護されているかどうかを確認してください。 |
| 136011 | ERROR | Failed to create new tag $tag for SAP HANA resource on $me. | 原因: 入力されたSAP HANAリソースタグは別のLifeKeeperリソースによってすでに使用されており、LifeKeeper newtagユーティリティは新しいタグを作成できなかった。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: SAP HANAリソースに対し、別のタグ名を選択してください。 |
| 136012 | ERROR | Failed creation of resource with $tag on system $me. | **原因**: LifeKeeperで指定したSAP HANAリソースを作成できなかった。<br><br>**処置**: LifeKeeperログファイルで見つかった問題を解決し、リソース作成操作を再試行してください。 |
| 136014 | ERROR | Failed to create resource dependency for parent $tag and child $virtual_ip_tag. | **原因**: SAP HANAリソースとそれが依存する仮想IPリソース間の依存関係の作成に失敗した。<br><br>**処置**: LifeKeeperログファイルで見つかった問題を解決し、リソース作成操作を再試行してください。 |
| 136015 | ERROR | The info field for resource $tag could not be successfully generated using values [SID: $info_sid, Instance: $info_instance, Replication Mode: $info_repl_mode, Site Name: $info_site_name, Operation Mode: $info_oper_mode]. Please verify that SAP HANA System Replication is fully configured and enabled on both the primary and secondary systems before creating the SAP HANA resource. | **原因**: SAP HANAの「リソース情報」フィールドの作成時に無効な値が見つかった。<br><br>**処置**: SAP HANAが正しくインストールおよび構成されていること、SAP HANAシステムレプリケーションがクラスター内のすべてのサーバーで完全に構成され有効になっていることを確認してください。 |
| 136016 | ERROR | The selected server $me is not the primary/source system for SAP HANA System Replication for the selected SID $sid and HDB instance $instance. Please select ‘Cancel’ and start this action on the primary/source HANA System Replication system. | **原因**: リソースの作成が、HANAシステムレプリケーションのセカンダリシステムで開始されている。<br><br>**処置**: HANAシステムレプリケーションのプライマリシステムで作成操作を開始してください。 |
| 136031 | ERROR | END failed extend of resource $target_tag on server $me with return value of $er | **原因**: SAP HANAリソースの拡張中に問 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | r_code. | 題が発生した。<br><br>**処置**: LifeKeeperログファイルで見つかった問題を解決し、リソース拡張操作を再試行してください。 |
| 136033 | ERROR | Usage: $usage | **原因**: SAP HANA extendスクリプトに無効な引数が指定された。<br><br>**処置**: 次の形式で適切な引数を指定してください：<br><テンプレートシステム> <テンプレートタグ> <スイッチバックタイプ> <ターゲットタグ> |
| 136035 | ERROR | Template resource $template_tag on server $template_sys does not exist. | **原因**: 拡張するテンプレートSAP HANAリソースがテンプレートサーバー上に存在しない。<br><br>**処置**: テンプレートサーバー上に、拡張されているSAP HANAリソースが存在することを確認してください。 |
| 136036 | ERROR | Resource with matching id $target_id already exists on server $me for App $app_type and Type $res_type. | **原因**: 同じLifeKeeper IDのSAP HANAリソースがターゲットシステムにすでに存在する。<br><br>**処置**: SAP HANAデータベースが、ターゲットサーバー上のLifeKeeperによってすでに保護されているかどうかを確認してください。 |
| 136037 | ERROR | Resource with matching tag $target_tag already exists on server $me for App $app_type and Type $res_type | **原因**: 同じLifeKeeperリソースタグを持つSAPリソースがターゲットサーバーに既に存在する。<br><br>**処置**: SAP HANAデータベースが、ターゲットサーバー上のLifeKeeperによってすでに保護されているかどうかを確認し |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | てください。 |
| 136039 | ERROR | Error creating resource $target_tag on system $me. | 原因: ターゲットサーバーで同等のSAP HANAリソースを作成できなかった。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース拡張操作を再試行してください。 |
| 136040 | ERROR | The target tag ($target_tag) and template tag ($template_tag) must be the same. | 原因: プライマリシステムで使用されるリソースタグ名が、セカンダリシステムで使用されるリソースタグ名と異なる。両方の名前は同じでなければならない。<br><br>処置: リソースの作成では、プライマリシステムタグおよびセカンダリシステムタグと同じ名前を使用してください。 |
| 136045 | ERROR | Cannot extend resource $template_tag to server $me. | 原因: SAP HANA canextendスクリプトは、リソースを特定のターゲットシステムに拡張できないことを示している。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース拡張操作を再試行してください。 |
| 136047 | ERROR | Usage: $usage | 原因: SAP HANA canextendスクリプトに無効な引数が指定された。<br><br>処置: 次の形式で適切な引数を指定してください：<br><テンプレートシステム> <テンプレートタグ> |
| 136048 | ERROR | Resource $template_tag does not exist on server $template_sys. | 原因: 拡張するテンプレートSAP HANAリソースがテンプレートサーバー上に存在しない。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: テンプレートサーバー上に、拡張しているSAP HANAリソースが存在することを確認してください。 |
| 136049 | ERROR | The system user $hana_user does not exist on server $me. | **原因**: SAP HANAデータベースのSAP管理ユーザーが、指定されたサーバーに存在しない。<br><br>**処置**: SAP HANAが正しくインストールおよび構成されていることを確認してください。 |
| 136050 | ERROR | The user id for user $hana_user ($template_uid) on template server $template_sys is not the same as user id ($uid) on target server $me. | **原因**: SAP管理ユーザーのユーザーIDが、テンプレートサーバーとターゲットサーバーで異なっている。<br><br>**処置**: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |
| 136051 | ERROR | The group id for user $hana_user ($template_gid) on template server $template_sys is not the same as group id ($gid) on target server $me. | **原因**: SAP管理ユーザーのグループIDが、テンプレートサーバーとターゲットサーバーで異なっている。<br><br>**処置**: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |
| 136052 | ERROR | The home directory for user $hana_user ($template_home) on template server $template_sys is not the same as home directory ($user_home) on target server $me. | **原因**: SAP管理ユーザーのホームディレクトリが、テンプレートサーバーとターゲットサーバーで異なっている。<br><br>**処置**: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136053 | ERROR | The SAP HANA instance $instance does not exist for $sid on server $me. | 原因: 指定されたSAP HANAデータベースインスタンスのインストールディレクトリが見つからなかった。<br><br>処置: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |
| 136055 | ERROR | The SAP HANA site name $target_obj->{'site_name'} on server $me must be different from site name $template_obj->{'site_name'} on $template_obj->{'sys'}. | 原因: SAP HANAシステムレプリケーションのサイト名が、プライマリサーバーとセカンダリサーバーの両方で同じになっている。<br><br>処置: セカンダリサーバー上のSAP HANAデータベースを停止し、hdbnsutilユーティリティを使用して、別のサイト名でセカンダリレプリケーションサイトを再登録してください。 |
| 136056 | ERROR | Unable to obtain SAP HANA System Replication parameters for database $instance on server $me. Please verify that SAP HANA System Replication is enabled and properly configured and that the database instance is running on all servers in the cluster. | 原因: 指定されたシステム上のデータベースのSAP HANAシステムレプリケーションパラメーターを決定できなかった。<br><br>処置: SAP HANAシステムレプリケーションが有効で適切に構成されていること、およびデータベースがクラスター内のすべてのサーバーで実行されていることを確認してください。 |
| 136057 | ERROR | Unable to create a HANA object for database $instance on server $me. Please verify that database instance $instance is properly installed. | 原因: 指定したサーバー上の指定したインスタンスを表す内部HANAオブジェクトを作成できない。<br><br>処置: データベースインスタンスが正しくインストールされており、必要なすべてのファイルシステムがマウントされていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136083 | ERROR | Usage: $usage | 原因: SAP HANA deleteスクリプトの引数が無効。<br><br>処置: 有効なHANA LifeKeeperリソースタグ名とリソースIDの両方を入力してください。<br>入力方法：delete -t <tag> -i <id> [-U] |
| 136160 | ERROR | Unable to create SAP HANA object. The SID and instance for the SAP HANA data base must be provided. | 原因: SAP HANAオブジェクトの作成中に、SAP SIDまたはSAP HANAインスタンス名のいずれかが欠落していた。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136161 | ERROR | Unable to create SAP HANA object. Resource system name and tag name must be provided. | 原因: SAP HANAオブジェクトの作成中に、システム名またはリソースタグ名のいずれかが欠落していた。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136162 | ERROR | Could not find any information regarding resource $tag on $sys. | 原因: 指定されたタグを持つSAP HANAリソースに関する情報を取得できなかった。<br><br>処置: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136168 | ERROR | Unable to check status of SAP Host Agent on server $self->{‘sys’}. Command \"$curr_cmd\" returned exit code $ret. | 原因: 指定されたサーバーでSAP Host Agentプロセスのステータスを判別できなかった。<br><br>処置: 詳細については、SAP Host Agentトレースファイル（dev_saphostexecなど）を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136174 | ERROR | Unable to create SAP HANA object. The SID or instance value is missing. | 原因:SAP HANAオブジェクトの作成時に、SAP SID、SAP HANAインスタンス名、またはSAP HANAシステムレプリケーション値のいずれかが欠落していた。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136182 | ERROR | Failed to $flg_action flag \"${HANA_FLAG_DATA_OUT_OF_SYNC}_$eqv_tag{$sys}\" on server $sys. | 原因: 指定されたサーバーで !HANA_DATA_OUT_OF_SYNC_<tag> フラグを作成または削除できなかった。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136190 | ERROR | Failed to start SAP Host Agent processes on server $self->{‘sys’}. Command \"$hostagent_cmd\" returned $ret. | 原因: 指定されたサーバーでSAP Host Agentプロセスを開始できなかった。<br><br>処置: 詳細については、SAP Host Agentトレースファイル（dev_saphostexecなど）を確認してください。 |
| 136191 | ERROR | Failed to start SAP OS Collector process on server $self->{‘sys’}. Command \"$oscol_cmd\" returned $ret. | 原因: 指定されたサーバーでSAP OS Collectorプロセスを開始できなかった。<br><br>処置: 詳細については、SAP OSコレクタートレースファイル（dev_collなど）を確認してください。 |
| 136193 | ERROR | $takeover_text of SAP HANA System Replication for SAP HANA database $self->{'instance'} failed on server $node with exit code $ret. | 原因: SAP HANAシステムレプリケーションで、指定されたデータベースのプライマリマスターとして指定されたサーバーを登録できなかった。<br><br>処置: 詳細については、SAP HANAトレースファイル（例：nameserver_<hostname>.xxxxx.xxx.trc）を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136197 | ERROR | Update of resource info field for $eqv_tag{$sys} on $sys failed with exit code $setinfo_ret. Current info: [$info]. Attempted new info: [$new_info]. | **原因**: 指定されたサーバー上の指定されたリソースの情報フィールドを更新できなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136202 | EMERG | Failed to disable Autostart for SAP HANA instance $self->{'instance'} on server $sys with exit code $remexec_ret. Please manually set \"Autostart = 0\" in the instance profile $profile on $sys. | **原因**: Autostartパラメーターの値を、指定されたサーバー上の指定されたHDBインスタンスプロファイルで変更できなかった。<br><br>**処置**: HDBインスタンスプロファイルを手動で編集し、「Autostart = 0」に設定してください。 |
| 136205 | ERROR | Failed start of SAP Start Service for SAP HANA database $instance on server $sys. | **原因**: 指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗した。<br><br>**処置**: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136208 | ERROR | LifeKeeper was unable to determine the SAP HANA System Replication mode for database $rem_obj->{'instance'} on server $rem_obj->{'sys'} while attempting to identify the previous primary replication site. Please resolve the issue and bring the SAP HANA resource in-service on the system where the database should be registered as primary master. | **原因**: 指定されたサーバーでSAP HANAシステムレプリケーションモードを判別できず、以前のプライマリレプリケーションサイトを特定できなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136210 | ERROR | Failed start of SAP Start Service for SAP HANA database $rem_obj->{'instance'} on server $rem_obj->{'sys'}. Unable to stop the database on the server where it is currently registered as primary master. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗し、その結果現在のプライマリSAP HANAシステムレプリケーションサイトでデータベースを停止できなかった。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 詳細については、SAP Start Service トレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136212 | ERROR | Failed stop of SAP HANA database $rem_obj->{‘instance’} on server $rem_obj->{‘sys’} where it is currently registered as primary master. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースの停止に失敗した。<br><br>**処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |
| 136217 | ERROR | Failed start of SAP HANA database $instance on server $sys. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースの起動に失敗した。<br><br>**処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |
| 136220 | ERROR | Unable to register $sys as a secondary SAP HANA System Replication site for database $instance. The host name of the current primary replication site was not provided. | **原因**: セカンダリレプリケーションサイトの登録時に、現在のプライマリSAP HANAシステムレプリケーションサイトのホスト名を入力しなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136233 | EMERG | WARNING: A temporary communication failure has occurred between servers $self->{'sys'} and $sys. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please take one of the following resource hierarchies out of service: $self->{'tag'} on $self->{'sys'} or $eqv{$sys} on $sys. The server that the resource hierarchy is taken out of service on will become the standby server for SAP HANA database $self->{'instance'}. | **原因**: 一時的な通信障害により、指定された同等のHANAリソースがそれぞれのホストサーバーで同時にIn Serviceになった。<br><br>**処置**: セカンダリレプリケーションサイトになるサーバーで、HANAリソース階層全体をOut of Serviceにします。そのサーバーでデータベースが停止されると、LifeKeeperは次のquickCheckサイクル中にデータベースをセカンダリレプリ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ケーションサイトとして自動的に登録します。 |
| 136234 | EMERG | WARNING: SAP HANA database $self->{'instance'} is running and registered as primary master on the following servers: $primary_node_list. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please stop database $self->{'instance'} on the intended secondary replication site by running the command \'su – $self->{'sid_admin'} -c \"$SAP_CONTROL -nr $self->{'instance_number'} -function StopWait $HANA_STOP_WAIT 5\"\' on that server, then bring resource $self->{'tag'} in-service on the intended primary replication site. Once the resource is in-service, the stopped database will automatically be re-registered as a secondary replication site. | 原因: 指定されたSAP HANAデータベースが、2つのクラスターサーバーで同時にプライマリマスターとして実行および登録されている。<br><br>処置: メッセージに示されているコマンドを使用して、スタンバイサーバー上のデータベースを停止します。 データベースが停止すると、LifeKeeperは自動的にスタンバイサーバーをセカンダリレプリケーションサイトとして登録します。 |
| 136238 | ERROR | Unable to create SAP HANA object. Resource system name and tag name must be provided. | 原因: SAP HANAオブジェクトの作成時に、サーバー名またはリソースタグ名のいずれかが欠落していた。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136239 | ERROR | Failed start of SAP Start Service for SAP HANA database $obj->{‘instance’} on server $obj->{‘sys’}. Unable to determine status of the database on $obj->{‘sys’}. | 原因: 指定されたサーバー上の指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗し、その結果データベースのステータスを判別できなかった。<br><br>処置: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136242 | ERROR | Unable to create SAP HANA object. At least one of the SAP HANA System Replication values is missing. | 原因: 指定したシステム上のデータベースに対し、SAP HANAシステムレプリケーションパラメーターを特定できない。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: SAP HANAシステムレプリケーションが有効になっていて適切に構成されていること、データベースがクラスター内のすべてのサーバーで実行されていることを確認してください。 |
| 136243 | ERROR | Unable to locate the pingnfs utility ($pingnfs) on server $me. Please verify that this utility exists and is executable. | **原因**: エクスポートされたNFS共有で利用可能なテストに使用されるpingnfsユーティリティが見つからない。<br><br>**処置**: pingnfsユーティリティが所定の場所に存在し、実行可能であることを確認してください。 |
| 136245 | ERROR | Critical NFS shares being exported by server $sys ($export_list) are not currently available. Please verify that the NFS server is alive and that all NFS-related services are running. | **原因**: 指定した重要なNFS共有ファイルシステムが現在利用できない。<br><br>**処置**: ファイルシステムをエクスポートするNFSサーバーが稼働していること、および必要なNFS関連サービスがすべて実行されていることを確認してください。 |
| 136248 | ERROR | Unable to open file $crit_mount_file on server $me. Please verify that this file exists and is read-enabled. | **原因**: 指定したサーバー上の重要なNFS共有のマウント情報を含むファイルを開けない。<br><br>**処置**: ファイルが指定した場所に存在し、読み取りが有効になっていることを確認してください。 |
| 136253 | ERROR | The SAP HANA \"takeover with handshake\" feature is available only for SAP HANA versions 2.0 SPS04 and greater. Database $self->{'instance'} cannot be resumed. | **原因**: 一時停止されたデータベースの再開は、2.0 SPS04より前のSAP HANAバージョンではサポートされていない。<br><br>**処置**:「Takeover with Handshake」に関連する機能を使用するには、SAP HANA 2.0 SPS04以降にアップグレードしてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136254 | ERROR | Attempt to resume suspended primary database $self->{'instance'} on server $self->{'sys'} failed with exit code $ret. | 原因: 指定したサーバーで中断されたデータベースインスタンスを再開しようとしたら失敗した。<br><br>処置:LifeKeeperとSAPのログファイルを調べて、失敗の原因を特定します。操作を再試行するか、対応するLifeKeeperリソースを別のサーバーでIn-Serviceにしてください。 |
| 136258 | ERROR | Attempt to register server $node as the secondary SAP HANA System Replication site for database $self->{'instance'} failed with exit code $ret. | 原因: 指定したサーバーを、指定したデータベースインスタンスのセカンダリSAP HANAシステムレプリケーションサイトとして登録しようとしたが失敗した。<br><br>処置:LifeKeeperとSAPのログファイルを調べて障害の原因を特定し、見つかった問題を手動で修正してください。SAP HANAリソースがIn-Serviceになっている間、LifeKeeperは自動的に、セカンダリHSRロールへのバックアップサーバーの登録試行を継続します。 |
| 136263 | ERROR | Usage: $usage | 原因: SAP HANAリソースのリストアスクリプトの引数が無効。<br><br>処置:次の形式で適切な引数を指定してください：<br>-t <リソースタグ> -i <リソースID> |
| 136265 | ERROR | Error getting resource information for $tag on server $me. | 原因: 指定されたサーバーの指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>処置:指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136266 | ERROR | The resource $tag protecting SAP HANA database $instance is not in sync. To protect the data LifeKeeper will not restore th | 原因: SAP HANAシステムレプリケーシ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | e resource on $me. Please restore the resource on the previous source server to allow the resync to complete. | ョンは、バックアップサーバー上でデータベースリソースをIn Serviceにする前に同期されていなかった。<br><br>**処置**: 前のプライマリサーバーでSAP HANAリソースをIn Serviceにし、再同期を完了できるようにしてください。 |
| 136275 | ERROR | Failed to determine SAP HANA System Replication mode for database $instance on server $me. | **原因**: 指定されたサーバー上の指定されたデータベースのSAP HANAシステムレプリケーションモードを判定できなかった。<br><br>**処置**: 詳細については、SAP HANAトレースファイル（例：nameserver_<hostname>.xxxxx.xxx.trc）およびLifeKeeperログファイルを確認してください。 |
| 136351 | ERROR | Usage: $usage | **原因**: SAP HANA resource quickCheckスクリプトの引数が無効。<br><br>**処置**: 次の形式で適切な引数を指定してください：<br>-t <リソースタグ> -i <リソースID> |
| 136353 | ERROR | Error getting resource information for $tag. | **原因**: 指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>**処置**: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136354 | EMERG | The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKeeper and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issu | **原因**: 指定されたデータベースのSAP HANAシステムレプリケーションモードが、LifeKeeperの外部で変更されている。<br><br>**処置**: プライマリマスターとして登録する必要があるサーバーで、SAP HANAリ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | e is resolved. | ソースをIn Serviceにしてください。 |
| 136363 | EMERG | LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved. | **原因**: 指定されたサーバーでSAP HANAシステムレプリケーションモードを判定できなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136375 | EMERG | An NFS server exporting a critical shared file system for resource $tag is currently unavailable. Resource monitoring for $tag will be suspended until the issue is resolved. | **原因**: 重要なNFS共有ファイルシステムが現在利用不可である。<br><br>**処置**: NFSサーバーが稼働していること、および必要なNFS関連サービスがすべて実行されていることを確認してください。 |
| 136376 | EMERG | WARNING: LifeKeeper resource $tag is designed for use in situations where SAP HANA System Replication (HSR) is disabled, but HSR was found to be enabled on server $me. Please ensure that the correct LifeKeeper resource type is being used for your current SAP HANA configuration. | **原因**: 指定したLifeKeeperリソースは、HANAシステムレプリケーション（HSR）が無効になっているSAP HANAデータベース環境を保護するように設計されているが、指定したサーバーではHSRが有効になっている。<br><br>**処置**: SAP HANAの構成に応じて、正しいLifeKeeperリソースタイプが使用されていることを確認してください。HSRが無効になっているSAP HANA構成から、HSRが有効になっているSAP HANA構成に移行した場合は、対応するSAP HANAリソースをLifeKeeperで再作成する必要があります。 |
| 136377 | EMERG | SAP HANA database $instance corresponding to resource $tag is currently suspended on server $me due to actions performed outside of LifeKeeper. In order to resume the suspended database, execute the command 'su – $obj->{'sid_admin'} -c \"$obj->{'hana_util'} -sr_resumeSuspendedPrimary\"' on server $me. Once the suspend | **原因**: 指定したサーバー上のSAP HANAデータベースインスタンスが、LifeKeeperの外部で行われた操作により停止された。<br><br>**処置**: 対応するLifeKeeperリソースが現 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | ed database has been resumed, LifeKeeper will automatically register the other servers in the cluster as secondary replication sites. Otherwise, please bring resource $tag in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved. | 在In-Service（Active）になっているサーバー上で一時停止中のデータベースを再開する場合は、メッセージに記載されているコマンドを実行してください。それ以外の場合は、目的のプライマリーレプリケーションサイトでLifeKeeper SAP HANAリソースをIn-Serviceにしてください。 |
| 136450 | ERROR | Usage: $usage | 原因: SAP HANA removeスクリプトに無効な引数が指定されている。<br><br>処置: 次の形式で適切な引数を指定してください：<br><テンプレートタグ> <テンプレートID> |
| 136454 | ERROR | Error getting resource information for $tag. | 原因: 指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>処置: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136456 | ERROR | Failed start of SAP Start Service for SAP HANA database $instance on server $me. | 原因: 指定されたSAP HANAデータベースのSAP Start Serviceを開始できなかった。<br><br>処置: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136459 | EMERG | WARNING: LifeKeeper resource $tag is designed for use in situations where SAP HANA System Replication (HSR) is disabled, but HSR was found to be enabled on server $me. Please ensure that the correct LifeKeeper resource type is being used for your current SAP HANA configuration. | 原因: 指定したLifeKeeperリソースは、HANAシステムレプリケーション（HSR）が無効になっているSAP HANAデータベース環境を保護するように設計されているが、指定したサーバーではHSRが有効になっている。<br><br>処置: SAP HANAの構成に応じて、正し |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | いLifeKeeperリソースタイプが使用されていることを確認してください。HSRが無効になっているSAP HANA構成から、HSRが有効になっているSAP HANA構成に移行した場合は、対応するSAP HANAリソースをLifeKeeperで再作成する必要があります。 |
| 136462 | ERROR | Failed to remove flag \"${hana::HANA_FLAG_LEAVE_DB_RUNNING}_$tag\" on server $me. This may cause subsequent remove actions for resource $tag on server $me to unintentionally fail to stop the database. | 原因: 指定したサーバーでLifeKeeperの!volatile!hana_leave_db_running_<tag>フラグを削除できない。<br><br>処置: 指定したサーバーでコマンド「/opt/LifeKeeper/bin/flg_remove -f '!volatile!hana_leave_db_running_<tag>'」を実行して、フラグを手動で削除してください。フラグが存在する間は、指定したサーバー上のSAP HANAリソースのOut-of-service操作では、保護対象のデータベースインスタンスが実行されたままになります。 |
| 136550 | ERROR | Usage: $usage | 原因: HANAリソース回復スクリプトの引数が無効。<br><br>処置: 有効なHANA LifeKeeperリソースタグ名とリソースIDの両方を入力してください。<br>入力方法：recover -d <タグ> -n <id> |
| 136555 | ERROR | Error getting resource information for $tag on server $me. | 原因: 指定されたサーバー上の指定されたHANAリソースに関する情報を取得できなかった。<br><br>処置: サーバーがオンラインで、LifeKeeperが実行されており、HANAリソースが存在することを確認してください。 |
| 136556 | EMERG | The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKee | 原因: 指定されたデータベースのSAP HA |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | per and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved. | NAシステムレプリケーションモードが、LifeKeeperの外部で変更されている。<br><br>**処置**: プライマリマスターとして登録する必要があるサーバーで、SAP HANAリソースをIn Serviceにしてください。 |
| 136558 | EMERG | LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved. | **原因**: 指定されたサーバー上の指定されたデータベースに対し、SAP HANAシステムレプリケーションモードを決定でなかった。<br><br>**処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |
| 136559 | ERROR | Resource $tag is no longer ISP on server $me. Exiting $cmd for $tag. | **原因**: 指定されたSAP HANAリソースが、指定されたサーバー上でISPではなくなった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136650 | ERROR | Usage: $usage | **原因**: SAP HANA hana_stop_all_dbs スクリプトに無効な引数が指定されている。<br><br>**処置**: 次の形式で適切な引数を指定してください：<br>hana_stop_all_dbs -t <タグ> |
| 136654 | ERROR | Error getting resource information for $tag. | **原因**: 指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>**処置**: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136658 | ERROR | Failed start of SAP Start Service for SAP HANA database $x->{'instance'} on server $x->{'sys'}. Could not determine status of SAP HANA DB on $x->{'sys'}. | **原因**: 指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗した。<br><br>**処置**: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136661 | ERROR | Failed stop of SAP HANA database $x->{'instance'} on server $x->{'sys'}. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースの停止に失敗した。<br><br>**処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |
| 136673 | ERROR | Usage: $usage | **原因**: SAP HANAのhana_takeover_with_handshakeスクリプトで無効な引数が指定された。<br><br>**処置**: 次の形式で適切な引数を指定してください： -t <tag> [-s <target server>] [-b] |
| 136674 | ERROR | Unable to remove the \"${hana::HANA_FLAG_LEAVE_DB_RUNNING}_$tag\" flag on server $isp_node. This may cause subsequent remove actions for resource $tag on server $isp_node to unexpectedly fail to stop the database. | **原因**: 指定したサーバーでLifeKeeperの!volatile!hana_leave_db_running_<tag>フラグを削除できない。<br><br>**処置**: 指定したサーバーでコマンド「/opt/LifeKeeper/bin/flg_remove -f '!volatile!hana_leave_db_running_<tag>'」を実行して、フラグを手動で削除してください。フラグが存在する間は、指定したサーバー上のSAP HANAリソースのOut-of-service操作では、保護対象のデータベースインスタンスが実行されたままになります。 |
| 136675 | ERROR | Script $cmd exited unexpectedly due to signal \"$sig\" on server $me. This may leave the $tag resource hierarchy as well as | **原因**: hana_takeover_with_handshake |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | SAP HANA System Replication in an unexpected state. Please verify that the cluster resources are in the expected state. | スクリプトが、指定したサーバー上の指定したシグナルにより、予期せず終了した。<br><br>**処置**: SAP HANAクラスタリソースが期待通りの状態であることを確認します。そうでない場合は、見つかった問題を修正し、目的のプライマリーサーバーでSAP HANAリソース階層をIn-Serviceにします。 |
| 136677 | ERROR | Unable to find equivalent SAP HANA resource corresponding to $tag on server $target_node. | **原因**: 指定したサーバー上の指定したリソースに対応する同等のSAP HANAリソースが見つからない。<br><br>**処置**: 指定したリソースタグが正しく、リソースが指定したターゲットサーバーに拡張されており、LifeKeeper がターゲットサーバー上で実行されて完全に初期化されていることを確認してください。 |
| 136678 | ERROR | Unable to obtain information about equivalent SAP HANA resource $tag on server $target_node. Verify that LifeKeeper is running and fully initialized. | **原因**: 指定したサーバー上の指定したリソースに対応する同等のSAP HANAリソースが見つからない。<br><br>**処置**: 指定したリソースタグが正しく、リソースが指定したターゲットサーバーに拡張されており、LifeKeeper がターゲットサーバー上で実行されて完全に初期化されていることを確認してください。 |
| 136679 | ERROR | Resource $tag is not a SAP HANA resource. | **原因**: 指定したリソースがSAP HANA（データベース/hana）リソースではない。<br><br>**処置**: リソースタグが正しいことを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136680 | ERROR | Resource $tag is designed for use in an environment where SAP HANA System Replication is disabled. Takeover with handshake cannot be performed for this resource type. Please use the standard \"In Service…\" command instead. | **原因**:「Takeover with Handshake」に関連する機能は、SAP HANAシステムレプリケーションが有効な環境でのみ使用できる。<br><br>**処置**: 標準の「In Service...」オプションを使用して、SAP HANAリソースをIn-Serviceにしてください。 |
| 136681 | ERROR | SAP HANA resource $tag is not currently in-service on any server in the cluster. The resource must be in-service and SAP HANA System Replication must be in-sync before performing a takeover with handshake. | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAリソースがクラスター内のどのサーバーでもIn-Serviceになっていない。<br><br>**処置**: 目的のプライマリーサーバーでSAP HANAリソースをIn-Serviceにしてください。 |
| 136682 | ERROR | SAP HANA resource $tag is currently in-service on multiple servers: $isp_node_list. The resource must be in-service on only one server and SAP HANA System Replication must be in-sync before performing a takeover with handshake. | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAリソースがクラスター内の複数のサーバーでIn-Serviceになっている。<br><br>**処置**: プライマリーマスターとして登録する予定のサーバーを除くすべてのサーバーで、リソースをOut-of-serviceにしてください。 |
| 136684 | ERROR | Unable to create internal SAP HANA object for resource $tag on server $target_node. Verify that all necessary file systems are mounted and that LifeKeeper is running and fully initialized on $target_node. | **原因**: 指定したサーバー上の指定したインスタンスを表す内部HANAオブジェクトを作成できない。<br><br>**処置**: データベースインスタンスが正しくインストールされ、必要なファイルシステムがすべてマウントされていることを確認してください。 |
| 136685 | ERROR | Takeover with handshake is only supported in SAP HANA versions 2.0 SPS04 and greater. The SAP HANA software must be upgraded in order to use this feature. Pl | **原因**: 基礎となるSAP HANAデータベースのバージョンが2.0 SPS04未満の場 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | ease use the standard \"In Service…\" command instead. | 合、「Takeover with Handshake」機能は使用できない。<br><br>**処置**:「Takeover with Handshake」機能を使用するには、SAP HANA 2.0 SPS04以降にアップグレードしてください。 |
| 136686 | ERROR | Takeover with handshake cannot be performed for database $target_obj->{'instance'} on server $target_node because the database is not currently running and registered as primary on any other server in the cluster. | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAデータベースインスタンスがクラスター内のどのサーバーでも実行されておらず、プライマリーマスターとして登録されていない。<br><br>**処置**: 目的のプライマリーサーバーで対応するSAP HANAリソースをIn-Serviceにしてください。 |
| 136687 | ERROR | SAP HANA database $target_obj->{'instance'} is running and registered as primary on more than one server in the cluster. Please resolve this situation and reattempt the takeover. | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAデータベースインスタンスが複数のサーバーでプライマリーとして実行および登録されている。<br><br>**処置**: データベースインスタンスがすでに実行されており、対象のプライマリーサーバーにプライマリーとして登録されている場合は、対応するLifeKeeperリソースをそのサーバーでIn-Serviceにします。そうでない場合は、LifeKeeperで現在In-Serviceになっているサーバーを除くすべてのサーバーでデータベースインスタンスを停止し、LifeKeeperがシステムレプリケーションを再開できるようにしてから、テイクオーバーを再試行してください。 |
| 136688 | ERROR | The ${hana::HANA_FLAG_DATA_OUT_OF_SYNC}_$tag flag is set on server $target_node, indicating that SAP HANA System Replication is not currently in-sync. Please allow the replication to synchronize before reattempting the takeover. | **原因**: 指定したサーバーで「!HANA_DATA_OUT_OF_SYNC_<tag>」フラグが設定され、「Takeover with Handshake」の試行中にSAP HANAシステムレプリケー |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ションが同期していないことを示している。<br><br>**処置**: システムレプリケーションが完全に同期するのを待ってから、テイクオーバーを再試行してください。 |
| 136689 | ERROR | Unable to create internal SAP HANA object for resource $tag on server $isp_node. Verify that all necessary file systems are mounted and that LifeKeeper is running and fully initialized on $isp_node. | **原因**: 指定したサーバー上の指定したインスタンスを表す内部HANAオブジェクトを作成できない。<br><br>**処置**: データベースインスタンスが正しくインストールされ、必要なファイルシステムがすべてマウントされていることを確認してください。 |
| 136690 | ERROR | Unable to set the ${hana::HANA_FLAG_LEAVE_DB_RUNNING}_$tag flag on server $isp_node. Aborting takeover with handshake attempt for database $target_obj->{'instance'} on server $target_node. | **原因**:「Takeover with Handshake」の試行中に、指定したサーバーで、LifeKeeperフラグ「!volatile!hana_leave_db_running_<tag>」を設定できなかった。<br><br>**処置**: 詳細については、LifeKeeper のログファイルを確認してください。見つかった問題を修正し、テイクオーバーを再試行してください。 |
| 136692 | ERROR | Takeover with handshake for database $target_obj->{'instance'} failed on server $target_node. | **原因**: 指定したターゲットサーバー上の指定したSAP HANAデータベースインスタンスに対する「Takeover with Handshake」の試行が失敗した。<br><br>**処置**: /etc/default/LifeKeeper で HANA_HANDSHAKE_TAKEOVER_FAILBACK=true が指定されている場合、SAP HANAリソース階層は以前のデータベースホストサーバーで自動的にIn-Serviceに戻ります。そうでない場合は、SAP HANAリソース階層を対象のプライマリーサーバーで手動でIn-Serviceにする必要があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136695 | ERROR | Resource $tag does not exist on server $me. | **原因**: 指定したリソースが指定したサーバーに存在しない。<br><br>**処置**: リソースタグが正しいことを確認してください。 |
| 136696 | ERROR | Unable to verify the status of resource $tag on server $target_node. Assuming that it is not in-service. | **原因**:「Takeover with Handshake」の成否の確認中に、指定したサーバー上の指定したリソースのステータスを判別できなかった。<br><br>**処置**: 指定したサーバー上でLifeKeeperが実行され、完全に初期化されていること、およびローカルサーバーとターゲットサーバー間のコミュニケーションパスがアクティブであることを確認してください。/etc/default/LifeKeeper で HANA_HANDSHAKE_TAKEOVER_FAILBACK=true が指定されている場合、SAP HANAリソース階層は自動的に以前のデータベースホストサーバーでIn-Serviceになります。そうでない場合は、SAP HANAリソース階層を対象のプライマリーサーバー上で手動でIn-Serviceにする必要があります。 |
| 136697 | ERROR | Resource $res was not successfully brought in-service on server $target_node. | **原因**:「Takeover with Handshake」の試行中に、指定したリソースが指定したサーバーでIn-Serviceにならなかった。<br><br>**処置**: etc/default/LifeKeeper で HANA_HANDSHAKE_TAKEOVER_FAILBACK=true に設定されている場合、SAP HANAリソース階層は自動的に以前のデータベースホストサーバーでサービスをIn-Serviceにします。そうでない場合は、SAP HANAリソース階層を対象のプライマリーサーバーで手動でIn-Serviceにする必要があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136698 | ERROR | LifeKeeper is not running or is not fully initialized on server $me. | 原因:「Takeover with Handshake」の試行中に、LifeKeeperが指定したサーバーで実行されていないか、完全に初期化されていない。<br><br>処置: LifeKeeperを /opt/LifeKeeper/bin/lkstart で起動するか、完全に初期化してから、テイクオーバーを再試行してください。 |
| 136699 | ERROR | Unknown server $target_node. | 原因: 指定したサーバーのホスト名が認識されない。<br><br>処置: サーバーのホスト名が正しく、ローカルサーバーとターゲットサーバーの間でコミュニケーションパスが作成されていることを確認してください。 |
| 137000 | ERROR | PowerShell is not installed. | |
| 137001 | ERROR | PowerCLI is not installed. | |
| 137002 | ERROR | A valid network interface was not found. | |
| 137005 | ERROR | Failed to attach VMDK. | |
| 137010 | ERROR | Failed to detach VMDK. | |
| 137020 | ERROR | Failed to execute VMDK status checker daemon. | |
| 137030 | ERROR | Disk not specified. | |
| 137031 | ERROR | Cannot get disk uuid for $Disk. Please check your ESXi settings. | |
| 137032 | ERROR | PowerCLI failed. %s | |
| 137034 | ERROR | Cannot bring VMDK resource \"%s\" in service on server \"%s\". | |
| 137037 | ERROR | This system is not a VMware guest. | |
| 137050 | ERROR | Failed to connect to ESXi server $addr. | |
| 137051 | ERROR | There is no ESXi server connected. | |
| 137055 | ERROR | Cannot determine ESXi VM ID because multiple network interfaces were found wit | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | h the MAC address $MAC_ADDR. | |
| 137057 | ERROR | Usable SCSI controller not found. | |
| 137058 | ERROR | Cannot find VMDK with ID $UUID. | |
| 137059 | ERROR | This guest has snapshots present. | |
| 137060 | ERROR | The VMDK with ID $UUID cannot be attac hed to this guest. | |
| 137061 | ERROR | The virtual storage controller has an inco mpatible sharing mode configured. | |
| 137068 | ERROR | The VMDK detection failed. Retry count e xceeded. | |
| 137070 | ERROR | Connect failed. | |
| 137071 | ERROR | Get-LocalVM failed. | |
| 137072 | ERROR | The VMDK has been detached remotely. This server has lost ownership. | |
| 137075 | ERROR | Cannot find virtual SCSI controller $CON TROLLER. | |
| 137076 | ERROR | The virtual storage controller has an inco mpatible sharing mode configured. | |
| 137077 | ERROR | Cannot find VM with MAC address $MA C_ADDR. | |
| 137078 | ERROR | Cannot find VM with MAC address $MA C_ADDR. | |
| 137101 | WARN | Partition information not defined for %s on %s. Retry. | |
| 137102 | ERROR | Partition information not defined for %s on %s. | |
| 137105 | ERROR | Device not specified. | |
| 137106 | ERROR | Cannot get device uuid for $Device. Pleas e check your ESXi settings. | |
| 137107 | ERROR | %s is not shareable with any machine. | |
| 137111 | ERROR | Failed to create dependency \"%s\"-\"%s\" on machine \"%s\". | |
| 137112 | ERROR | Cannot bring VMDKP resource \"%s\" in s ervice on server \"%s\". | |

# 8.1.1. DataKeeper メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104002 | FATAL | $msg | **原因**: このメッセージは内部ソフトウェアエラーを示します。<br><br>**処置**: スタックトレースにエラーの原因が示されます。 |
| 104003 | FATAL | $self->Val('Tag') . " is not an SDR resource"); | **原因**: 非データレプリケーションリソースに対してデータレプリケーションアクションが実行されました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104010 | ERROR | $self->{'md'}: bitmap merge failed, $action | **原因**: ビットマップのマージ操作に失敗しました。<br><br>**処置**: ターゲットサーバにミラーや保護されたファイルシステムがマウントされている可能性があります。または、ビットマップファイルがターゲットに見つからない可能性があります。ターゲットサーバを確認してください。 |
| 104022 | ERROR | $argv[1]: mdadm failed ($ret) | **原因**:「mdadm」コマンドによるミラーへのデバイスの追加に失敗しました。<br><br>**処置**: これは通常、一時的な状態です。 |
| 104023 | ERROR | $_); | **原因**: メッセージに「mdadm」コマンドの結果が含まれています。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104025 | ERROR | failed to spawn monitor | **原因**: システムは、「mdadm -F」監視プロセスの開始に失敗しました。これは、通常の状態では発生しません。<br><br>**処置**: システムを再起動し、競合の可能性があるものが解決されていることを確認してください。 |
| 104026 | ERROR | cannot create $md | **原因**: ミラーデバイスを作成できませんした。<br><br>**処置**: デバイスがまだ使用中ではないこと、およびミラー作成用の他の全パラメータが正しいことを確認してください。 |
| 104027 | ERROR | $_ | **原因**: このメッセージに「mdadm」コマンドの結果が含まれています。 |
| 104035 | ERROR | Too many failures. Aborting resync of $md | **原因**: 正常範囲とは思われないほど長い時間、デバイスがビジー状態です。<br><br>**処置**: システムを再起動し、デバイスがビジーではなくなっていることを確認してください。 |
| 104036 | ERROR | Failed to start nbd-server on $target (error $port) | **原因**: ターゲットサーバで nbd-server プロセスを開始できませんでした。<br><br>**処置**: ターゲットディスクデバイスが使用可能であること、およびデバイス ID が変更されていないことを確認してください。 |
| 104037 | ERROR | Failed to start compression (error $port) | **原因**: システムは、「バランス」トンネルプロセスを開始できませんでした。または、ネットワークに問題が発生しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: ネットワークが正常に実行されていること、および 10000 ～ 10512 の範囲内にある TCP ポートがオープンで未使用の状態にあることを確認してください。ソフトウェアがすべてのシステムに正常にインストールされていることを確認してください。 |
| 104038 | ERROR | Failed to start nbd-client on $source (error $ret) | **原因**: nbd-client プロセスをソースサーバで開始できませんでした。<br><br>**処置**: レポートされている errno 値を調べ、レポートされている問題を解決してください。たとえば、errno 値 110 は「Connection timed out」(接続のタイムアウト) を示します。これは、ネットワークかファイアウォールに問題がある可能性があります。 |
| 104039 | ERROR | Failed to add $nbd to $md on $source | **原因**: これは通常、一時的な状態です。<br><br>**処置**: このエラーが続く場合、システムを再起動し、競合の可能性があるものを解決してください。 |
| 104045 | ERROR | failed to stop $self->{'md'} | **原因**: ミラーデバイスを停止できませんした。<br><br>**処置**: デバイスがビジーでないこと、またはマウントされていないことを確認してください。「mdadm —stop」を手動で実行し、デバイスを停止してください。 |
| 104048 | WARN | failed to kill $proc, pid $pid | **原因**: プロセスをシグナル化できませんでした。これは、プロセスが終了していることを示している可能性があります。<br><br>**処置**: 問題のプロセスが実行されていないことを確認し、実行されている場合 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | は、システムを再起動して、終了できないプロセスを消去してください。 |
| 104050 | ERROR | Setting $name on $dest failed: $ret. Please try again. | **原因**: システムは「mirrorinfo」ファイル設定を設定できませんでした。<br><br>**処置**: ネットワークとシステムを確認し、ミラー設定操作を再試行してください。 |
| 104052 | FATAL | Specified existing mount point "%s" is not mounted | **原因**: マウントポイントがアンマウント状態になりました。<br><br>**処置**: マウントポイントがマウントされていることを確認し、操作を再試行してください。 |
| 104055 | ERROR | Failed to set up temporary $type access to data for $self->{'tag'}. Error: $ret | **原因**: ターゲットサーバでファイルシステムまたはデバイスを使用できません。ミラーされたデータは、ミラーが一時停止してから再開されるまでターゲットサーバでは使用できません。<br><br>**処置**: ターゲットサーバを再起動し、競合の可能性があるものを解決してください。 |
| 104057 | ERROR | Failed to undo temporary access for $self->{'tag'} on $self->{'sys'}. Error: $ret. Please verify that $fsid is not mounted on server $self->{'sys'}. | **原因**: ターゲットサーバでファイルシステムをアンマウントできませんでした。<br><br>**処置**: ファイルシステムおよびデバイスがターゲットサーバでビジーでないことを確認してください。ターゲットサーバを再起動し、競合の可能性があるものを解決してください。 |
| 104062 | FATAL | Cannot find a device with unique ID "%s" | **原因**: ターゲットディスクを識別できませんでした。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 適切なストレージリカバリキットがターゲットサーバにインストールされていることを確認してください。ターゲットディスクのデバイス ID が変更されていないことを確認してください。 |
| 104066 | FATAL | Cannot get the hardware ID of device "%s" | **原因**: ターゲットディスクデバイスの一意の ID が見つかりませんでした。<br><br>**処置**: 適切なストレージリカバリキットがターゲットサーバにインストールされていることを確認してください。ターゲットディスクのデバイス ID が変更されていないことを確認してください。 |
| 104067 | FATAL | Asynchronous writes cannot be enabled without a bitmap file | **原因**: 無効なパラメータでミラーを作成しようとしました。<br><br>**処置**: ビットマップファイルパラメータを指定するか、同期書き込みを指定する必要があります。 |
| 104068 | FATAL | Failed to extend dependent resource %s to system %s. Error %s | **原因**: 階層の拡張操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104070 | FATAL | Unable to extend the mirror "%s" to system "%s" | **原因**: 階層の拡張操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104071 | ERROR | Failed to restore target device resources on $target->{'sys'} : $err | **原因**: ターゲットサーバで in service 操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | し、レポートされている問題を解決してください。 |
| 104074 | FATAL | Cannot get the hardware ID of device "%s" | **原因**: ミラーに使用しようとしている基本のディスクデバイスを認識するストレージリカバリキットがありません。<br><br>**処置**: 適切なストレージリカバリキットがインストールされていることを確認してください。必要に応じ、/opt/LifeKeeper/subsys/scsi/resources/DEVNAME/device_pattern ファイルにデバイス名を入れてください。 |
| 104081 | FATAL | Cannot make the %s filesystem on "%s" (%d): | **原因**:「mkfs」コマンドが失敗しました。<br><br>**処置**:ディスクデバイスが書き込み可能で、エラーがないこと、および選択したファイルシステムのファイルシステムツールがインストールされていることを確認してください。 |
| 104082 | FATAL | %s | **原因**: このメッセージに「mkfs」コマンドの結果が含まれています。 |
| 104083 | FATAL | Cannot create filesys hierarchy "%s" | **原因**: リソースの作成に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104086 | ERROR | The "%s_data_corrupt" flag is set in "%s/subsys/scsi/resources/netraid/" on system "%s". To avoid data corruption, LifeKeeper will not restore the resource. | **原因**: 予期しないデータ破損を防ぐための警告データとして data corrupt フラグファイルが設定されています。ファイルが削除されるまで、このサーバ上のミラーをリストアできません。<br><br>**処置**: 問題のサーバでデータが有効な場 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 合、以下のいずれかを実行できます。1) ファイルを削除してミラーをリストアします。2) LifeKeeper GUI または「mirror_action force」コマンドでミラーを強制的にオンラインにします。 |
| 104092 | ERROR | Mirror target resource movement to system %s : status %s | **原因**: 階層のスイッチオーバ操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104099 | ERROR | Unable to unextend the mirror for resource "%s" from system "%s" | **原因**: 階層の拡張解除に失敗しました。<br><br>**処置**: ターゲットサーバを再起動し、競合の可能性があるものを解決して、操作を再試行します。 |
| 104106 | ERROR | remote 'bitmap -m' command failed on $target->{'sys'}: $ranges | **原因**: ターゲットサーバでビットマップマージコマンドが失敗しました。これには、次の 2 つの原因が考えられます。1) ビットマップファイルが見つからないか、破損している。2) ミラー（md）デバイスがターゲットでアクティブになっている。<br><br>**処置**: ミラーおよび保護されているファイルシステムがターゲットでアクティブになっていることを確認してください。ターゲットのビットマップファイルが見つからない場合、ミラーを一時停止して再開し、ビットマップファイルを再作成してください。 |
| 104107 | ERROR | Asynchronous writes cannot be enabled without a bitmap file | **原因**: ミラー作成操作で無効なパラメータが指定されました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104108 | ERROR | Local Partition not available | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104109 | ERROR | Cannot get the hardware ID of device "%s" | 原因: ディスクデバイスの一意の ID を特定できませんでした。<br><br>処置: 適切なストレージリカバリキットがサーバにインストールされていることを確認してください。ディスクのデバイス ID が変更されていないことを確認してください。 |
| 104111 | FATAL | Insufficient input parameters for "%s" creation | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104112 | FATAL | Insufficient input parameters for "%s" creation | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104113 | FATAL | Insufficient input parameters for "%s" creation | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104114 | FATAL | Insufficient input parameters for "%s" creation | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104115 | FATAL | Insufficient input parameters for "%s" creation | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104117 | FATAL | Insufficient input parameters for "%s" creation | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104118 | FATAL | Cannot unmount existing Mount Point "%s" | 原因: マウントポイントがビジーです。<br><br>処置: ファイルシステムがビジーではないことを確認してください。ファイルシ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ステムにアクセスする可能性があるプロセスまたはアプリケーションを停止してください。 |
| 104119 | FATAL | Invalid data replication resource type requested ("%s") | 原因: ミラー作成操作で無効なパラメータが指定されました。 |
| 104124 | EMERG | WARNING: A temporary communication failure has occurred between systems %s and %s. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should take one of the following resources out of service: %s on %s or %s on %s. The resource that is taken out of service will become the mirror target. | 原因: ソースサーバとターゲットサーバの間で、一時的な通信障害 (スプリットブレインシナリオ) が発生しました。<br><br>処置: メッセージテキストに記載されている手順を実行してください。 |
| 104125 | ERROR | failed to start '$cmd $_[2] $user_args' on '$_[3]': | 原因: 指定したコマンドが失敗しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104126 | ERROR | $_ | 原因: このメッセージに、メッセージ 104125 で失敗したとレポートされているコマンドの結果が含まれています。 |
| 104128 | FATAL | comm path/server not specified | 原因: netraid.down スクリプトが、通信パスまたはサーバー名を指定せずに呼び出されました。このスクリプトは内部で呼び出されるため、常に適切なパラメーターを使用する必要があります。<br><br>処置: このエラーを SIOS サポートに報告してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104129 | WARN | The replication connection for mirror “%s” (resource: “%s”) is down (reason: %s) | **原因**: ミラーへのレプリケーションの接続がダウン状態です。<br><br>**処置**: ネットワークを確認してください。 |
| 104130 | ERROR | Mirror resize failed on %s (%s). You must successfully complete this operation before using the mirror. Please try again. | **原因**: ミラーのサイズ変更操作で、リストされたシステム上にあるミラーのメタデータの更新に失敗しました。<br><br>**処置**: ミラーを使用するには、サイズへンコウを正常に完了する必要があります。「mirror_resize」を再実行してください (必要に応じて、強制実行する -f を使用)。 |
| 104132 | ERROR | The partition "%s" has an odd number of sectors and system "%s&qquot; is running kernel >= 4.12. Mirrors with this configuration will not work correctly with DataKeeper. Please see the SIOS product documentation for information on how to resize the mirror. | **原因**: ミラーの作成用に選択されたパーティションまたはディスクには奇数のディスクセクターがあるため、DataKeeperで使用するにはサイズを変更する必要があります。<br><br>**処置**:「parted」コマンドを使用してパーティションのサイズを変更するか、可能な場合はプラットフォーム（VMware、AWS）ツールを使用してディスクのサイズを変更します。注記：データが失われる可能性があるため、この操作は慎重に行ってください。 |
| 104136 | ERROR | Extend failed. | **原因**: 階層の拡張操作に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 104143 | ERROR | Mirror resume was unsuccessful ($ret) | **原因**:ミラーを確立できませんした。<br><br>**処置**: 関連するエラーをログで確認 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | し、レポートされている問題を解決してください。 |
| 104144 | ERROR | Unable to stop the mirror access for $self->{'md'} on system $self->{'sys'}. Error: $ret. Use the "mdadm —stop $self->{'md'}" command to manually stop the mirror. | **原因**: ミラーの一時停止時にターゲットノード上に作成されたミラーデバイスを停止できませんでした。<br><br>**処置**: デバイスがビジーでないこと、またはマウントされていないことを確認してください。「mdadm —stop」を手動で実行し、デバイスを停止してください。 |
| 104145 | WARN | Unable to dirty full bitmap.  Setting fullsync flag. | **原因**: ビットマップ全体をダーティにすることによる完全な再同期を実行できませんでした。代わりに fullsync フラグが使用されます。完全な同期は実行されるので、これは致命的なエラーではありません。<br><br>**処置**: ありません |
| 104146 | EMERG | WARNING: The target system %s for mirror %s has the target mirror %s currently active. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should reboot system %s. | **原因**: ミラーがターゲットシステムで構成されています。<br><br>**処置**: ターゲットシステムを再起動する必要があります。 その後、DataKeeper はミラーを再同期できるようになります。 |
| 104147 | EMERG | WARNING: The target system %s for mirror %s has the target disk %s currently mounted. In order to avoid data corruption, data resynchronization will not occur. MANUAL INTERVENTION IS REQUIRED. In order to initiate data resynchronization, you should unmount %s on %s. A full resync will occur. | **原因**: ミラーディスクがターゲットシステムにマウントされています。<br><br>**処置**: 完全なミラー再同期を開始するには、ミラーディスクをターゲットシステムでアンマウントする必要があります。ディスクで追跡されていない変更が発生したため、完全な再同期が必要です。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104148 | EMERG | The storage configuration for mirror "%s (%s)" does not have a unique identifier and may have potential for data corruption in some environments in certain circumstances. Please refer to the SIOS Product Documentation for details on DataKeeper storage configuration options. | 原因: ミラーリング用に選択されたディスクが、オペレーティングシステムにUUIDを提供していません。DataKeeperはUUIDがないとディスクをミラーできません。<br><br>処置: ディスク上にGPTパーティションテーブルを作成して、ディスクパーティションにUUIDを提供できます。 DataKeeper for Linux トラブルシューティング を参照してください。 |
| 104149 | ERROR | snmptrap returned $err for Trap 142 | |
| 104150 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 104152 | ERROR | [$SUBJECT event] mail returned $err | |
| 104153 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 104155 | EMERG | The mirror %s cannot be forced online at this time. The underlying disk %s is mounted, indicating possible data corruption. MANUAL INTERVENTION IS REQUIRED. You must unmount %s on %s and restore the mirror to the last known mirror source system. A full resync will need to be performed from the source system to %s. | 原因: ターゲットシステムでミラーディスクがマウントされた。<br><br>処置: データ破損を防ぐため、すぐにディスクをアンマウントしてください。 |
| 104156 | WARN | Resynchronization of "%s" is in PENDING state. Current sync_action is: "%s" | 原因: md デバイスの再同期がペンディング状態であることが検出されました。<br><br>処置: LifeKeeper は再同期を強制実行して問題の修正を試行します。関連するエラーをログで確認してください。エラーを解決した後、/proc/mdstat でペンディング状態がクリアされたこと、および datarep リソースの再同期が進行中であるか、完了したことを確認してください。 |
| 104157 | WARN | /etc/sysconfig/raid-check update failed. Please %s "md%d" to SKIP_DEVS. | 原因: /etc/sysconfig/raid-check を変更して、スキップする MD デバイスのリスト（SKIP_DEVS）にエントリを追加または削除することができません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: raid-check または SKIP_DEVS に関連するエラーがないか、システムログを確認してください。リストされたmdを手動で追加または削除してください。 |
| 104158 | EMERG | WARNING: The local disk partition $self->{ part } for data replication device\n$self->{ md } has failed. MANUAL INTERVENTION IS REQUIRED. | **原因**: ミラーのローカルデバイスで問題が発生しました。LKDR_FAILURE でリカバリーアクションが設定されていないため、復旧するには手動による介入が必要です。<br><br>**処置**: システムログと LifeKeeper ログで、ローカルディスクに関連するエラーを確認してください。 |
| 104163 | WARN | The "%s_data_corrupt" flag is set in "%s/subsys/scsi/resources/netraid/" on system "%s". The mirror is being forced online. | **原因**: ミラーは強制的にオンラインになり、data_corrupt フラグを上書きします。指定されたシステム上のデータは、最新のデータとして扱われます。これが正しくない場合、データ破損またはデータ損失につながる可能性があります。<br><br>**処置**: ありません |
| 104164 | ERROR | The "%s_data_corrupt" flag for related mirror resource "%s" is set in "%s/subsys/scsi/resources/netraid/" on system "%s". To avoid data corruption, LifeKeeper will not restore this mirror or any related mirrors in the hierarchy. | **原因**: data_corrupt フラグは、階層内の1つ以上のミラーに存在します。追加データの破損を防ぐため、すべての data_corrupt フラグが解決されるまで、どのミラーも In Service になりません。<br><br>**処置**: LifeKeeperのログを確認して、各ミラーが最後に In Service になっていた場所、つまり各ミラーの最新データが存在する場所を特定します。すべての階層が In Service であった以前のソースでミラーを In Service にし、ミラーがすべてのターゲットと同期できるようにする必要があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104165 | ERROR | The "%s_data_corrupt" flag for related mirror resource "%s" is set in "%s/subsys/scsi/resources/netraid/" on system "%s". The mirror resource "%s" is being forced online. | 原因: ミラーは強制的にオンラインになっており、data_corrupt フラグを上書きします。指定されたシステム上のデータは、すべてのターゲットと同期される正しいデータとして扱われます。これが最新のデータでない場合、データ破損またはデータ損失につながる可能性があります。<br><br>処置: ありません |
| 104170 | ERROR | Failed to create "source" flag file on shared source %s to track mirror source. This may result in a full resync. | 原因:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104171 | ERROR | Failed to create "source" flag file on %s to track mirror source. Target %s will not be added to mirror. | 原因:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104172 | ERROR | The "source" flag file on %s does not contain a valid target (%s). Full resync to remaining targets is required. | 原因:「ソース」フラグファイルには以前のソースのシステム名が含まれている必要がありますが、リストされている名前は構成されたシステムのリストにありませんでした。<br><br>処置: この問題を SIOS サポートに報告してください。 |
| 104173 | ERROR | Failed to create "source" flag file on %s to track mirror source. | 原因:「ソース」フラグファイルが、ミラ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104174 | ERROR | Failed to create “previous_source” flag file to track time waiting on source. Will not be able to timeout. | 原因:「previous_source」フラグファイルは、ミラーの以前のソースを追跡するためのミラーソース上に作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104175 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104176 | ERROR | The "source" flag file on %s to track mirror source does not exist. Full resync is required. | 原因:「ソース」フラグファイルはシステム上に存在する必要があり、これがないとミラーの整合性を検証できません。データの信頼性を確保するには、完全な再同期が必要です。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104177 | ERROR | Failed to determine amount of time waiting on %s. | 原因: 以前のソースを待機する時間を決 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 定できませんでした。以前のソースが見つからない場合、ターゲットは完全に再同期されて追加されます。<br><br>**処置**: ありません |
| 104178 | ERROR | Failed to update "source" flag file on target "%s", previous source must be merged first. | **原因**: ターゲットのソースフラグファイルは、同期されて停止されると更新されるため、次のIn Service時には以前のソースは必要ありません。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104180 | ERROR | Internal Error: "previous_source" has the local system name (%s). | **原因**: ローカルシステム名は、previous_source フラグファイルに含めることはできません。<br><br>**処置**: このエラーを SIOS サポートに報告してください。 |
| 104181 | ERROR | Internal Error: There are no targets waiting on %s to be merged. | **原因**: 以前のソースがマージされるのを待機しているターゲットがありません。<br><br>**処置**: このエラーを SIOS サポートに報告してください。 |
| 104182 | ERROR | Failed to create \"source\" flag file on %s to track mirror source. This may result in a full resync. | **原因**: リストされたターゲットに「ソース」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 104186 | ERROR | Failed to create "last_owner" flag file on %s to track mirror source. This may allow in-service of mirror on old data. | 原因:「last_owner」フラグファイルがソースに作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104187 | WARN | $REM_MACH has ${REM_TAG}_last_owner file, create flag ${FLAGTAG}_data_corrupt. | 原因: リストされているシステムでは、ミラーが最後にIn Serviceになりました。<br><br>処置: リストされているシステムには last_owner ファイルがあります。これは、このシステムには最新のデータがあり、データの損失を防ぐためにミラーをIn Serviceにするのに最適なシステムである可能性が高いことを示しています。 |
| 104188 | WARN | $REM_MACH is not alive, create flag ${FLAGTAG}_data_corrupt. | 原因: リストされているシステムは稼働していません。<br><br>処置: リストされているシステムは稼働していないため、そのシステムがローカルシステムよりも新しいミラーソースであったかどうかを判断できません。このため、ローカルシステムがミラーをIn Serviceにすることを自動的に許可しないでください。 |
| 104200 | EMERG | Continue to wait for %s to merge bitmap and do partial resyncs to all targets, no timeout set. | 原因: マルチターゲット構成では、すべてのターゲットが部分的に再同期できるように、以前のソースがビットマップをマージできるようになるまでターゲットは構成されません。/etc/defaults/LifeKeeper の LKDR_WAIT_ON_PREVIOUS_SOURCE_TIMEOUT エントリは、無期限に待機するために「-1」に設定されます。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: メッセージにリストされている以前のソースのステータスを確認し、クラスターへの再参加を妨げている問題を解決してください。 |
| 104201 | EMERG | To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: "%s/bin/mirror_action %s fullresync %s %s" on %s. | **原因**: マルチターゲット構成では、ターゲットは構成されず、以前のソースがクラスターに再参加するのを待機します。<br><br>**処置**: メッセージにリストされているコマンドを実行して、このターゲットおよび再同期を待機している残りのターゲットに対して即時の完全再同期を強制してください。 |
| 104202 | EMERG | Continue to wait for %s to merge bitmap and do partial resyncs to all targets. Continue to wait %d more seconds. | **原因**: マルチターゲット構成では、すべてのターゲットが部分的に再同期できるように、前のソースがビットマップをマージできるようになるまでターゲットは構成されません。 /etc/defaults/LifeKeeper の LKDR_WAIT_ON_PREVIOUS_SOURCE_TIMEOUT エントリは、待機する秒数に設定されます。以前のソースがその時間内にクラスターに参加しない場合、ターゲットは完全に再同期されて追加されます。<br><br>**処置**: メッセージにリストされている以前のソースのステータスを確認し、クラスターへの再参加を妨げている問題を解決してください。 |
| 104203 | EMERG | To stop waiting for the previous source (forcing a full resync to remaining waiting targets) run: \"%s/bin/mirror_action %s fullresync %s %s\" on %s. | **原因**: マルチターゲット構成では、ターゲットは構成されず、以前のソースがクラスターに再参加するのを待機します。<br><br>**処置**: メッセージにリストされているコマンドを実行して、このターゲットおよび再同期を待機している残りのターゲットに対して即時の完全再同期を強制して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください。<br><br>**注記**: リストされたターゲットが削除されて戻らない場合でも、このコマンドを実行して待機を停止してください。 |
| 104207 | ERROR | Failed to create "data_corrupt" flag file on "%s". | **原因**: リストされたソースに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104208 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | **原因**: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104209 | ERROR | Failed to create "data_corrupt" flag file on "%s". | **原因**: リストされたソースに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104210 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | **原因**: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>**処置**: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | Full になっていないか確認してください。 |
| 104211 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104212 | ERROR | The "source" flag file on %s to track mirror source does not exist. Full resync to remaining targets is required. | 原因:「ソース」フラグファイルはシステム上に存在する必要があり、これがないとミラーの整合性を検証できません。データの信頼性を確保するには、完全な再同期が必要です。まだミラーされていないすべてのターゲットでは、完全な再同期が必要です。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください. |
| 104214 | ERROR | Failed to create "source" flag file on %s to track mirror source. | 原因:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104216 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | イルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104217 | ERROR | Failed to create "source" flag file on shared source %s to track mirror source. This may result in a full resync. | 原因:「ソース」フラグファイルが、ミラーソースを追跡するために指定されたシステムで作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104218 | ERROR | Failed to create "data_corrupt" flag file on target "%s". | 原因: リストされたターゲットに「data_corrupt」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104221 | ERROR | Failed to create "last_owner" flag file on %s to track mirror source. This may allow in-service of mirror on old data. | 原因: リストされたターゲットに「last_owner」フラグファイルが作成されませんでした。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してください。 |
| 104222 | ERROR | Failed to create "last_owner" flag file to track mirror source". This may allow in-service of mirror on old data. | 原因:「last_owner」フラグファイルは、ミラーが最後にIn Serviceになった場所を特定するために使用されています。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、または Full になっていないか確認してくださ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | い。 |
| 104223 | ERROR | Failed to create "last_owner" flag file to track mirror source". This may allow in-service of mirror on old data. | 原因:「last_owner」フラグファイルは、ミラーが最後にIn Serviceになった場所を特定するために使用されています。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、またはFull になっていないか確認してください。 |
| 104224 | ERROR | Failed to create "previous_source" flag file. | 原因:「previous_source」フラグファイルが作成されませんでした。これは、以前のソースのビットマップをマージして完全な同期を回避するために必要です。<br><br>処置: LKROOT（/opt/LifeKeeper）ファイルシステムにエラーがないか、またはFull になっていないか確認してください。 |
| 104227 | ERROR | Failed to set %s to %s. | 原因: このメッセージは、nbdドライバー（/sys/block/nbdX）のsysfsパラメーターの設定に失敗したことを示しています。<br><br>処置: このエラーを回避するには、/etc/default/LifeKeeper で次のうち1つ以上を調整する必要がある場合があります。<br>NBD_NR_REQUESTS<br>NBD_SCHEDULER<br>LKDR_ASYNC_LIMIT |
| 104232 | ERROR | Mirror resize failed on %s (%s). Could not set size to %d. | 原因: ミラーのサイズ変更操作に失敗しました。 |
| 104233 | ERROR | Mirror resize failed on %s (%s). Could not set bitmap to %s and bitmap-chunk to | 原因: ミラーのサイズ変更操作に失敗し |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | %d. | ました。 |
| 104234 | ERROR | The mirror %s failed to resize. You must successfully complete this operation before using the mirror. Please try again. | 原因: ミラーのサイズ変更操作に失敗しました。 |
| 104235 | ERROR | mirror_resize of mirror %s failed due to signal "%s". | 原因: ミラーのサイズ変更操作に失敗しました。 |
| 104236 | EMERG | Resource "%s" is "OSF". The mirror "%s" will wait to replicate data until all resources in the hierarchy%s are in-service. This may indicate inconsistent data. Verify the data is correct before replicating data; replication will continue when all resources in the hierarchy%s are in-service. A full resync may be necessary (see "LKDR_WAIT_TO_RESYNC" in /etc/default/LifeKeeper). | |
| 104237 | WARN | Resource "%s" is "OSU". The mirror "%s" will wait to replicate data until all resources in the hierarchy%s are in-service. To replicate data immediately run: "%s/bin/mirror_action %s resume" on "%s" (see "LKDR_WAIT_TO_RESYNC" in /etc/default/LifeKeeper). | |
| 104238 | ERROR | Unable to read /sys/module/nbd/taint. Assuming that the SIOS nbd module is not loaded. | |
| 104239 | EMERG | $nbd_failure_msg | |
| 104242 | ERROR | Unable to read /proc/modules. Assuming that the SIOS nbd module is not loaded. | |
| 104251 | ERROR | There is no LifeKeeper protected resource with tag $tag on system $me. | 原因: 指定されたタグは、指定されたシステム上のLifeKeeperで保護されているリソースに対応していません。<br><br>処置: リソースタグとシステム名が正しいことを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
| --- | --- | --- | --- |
| 104252 | ERROR | Resource $tag is not a $app/$typ resource. Please use the $ins_app/$ins_typ resource-specific canfailover script instead. | 原因: scsi/netraid 固有の canfailover スクリプトが、非 scsi/netraid リソースに対して呼び出されました。<br><br>処置: 指定されたリソースの適切なアプリケーションとタイプに対応する canfailover スクリプトを使用してください（存在する場合）。 |

# 8.1.2. DB2 Kit メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103001 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | 原因: db2nodes.cfg にサーバ名が含まれていません。<br><br>処置: db2nodes.cfg が有効であることを確認してください。 |
| 103002 | ERROR | LifeKeeper was unable to get the version for the requested instance "%s" | 原因:「db2level」コマンドが DB2 のバージョンを返しませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103003 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103004 | ERROR | Unable to get the information for resource "%s" | 原因: リソース情報の取得に失敗しました。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 103005 | ERROR | Unable to get the information for resource "%s" | 原因: リソース情報の取得に失敗しました。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 103006 | ERROR | Unable to get the instance information for resource "%s" | 原因: インスタンス情報の取得に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: LifeKeeper の設定を確認してください。 |
| 103007 | ERROR | Unable to get the instance home directory information for resource "%s" | **原因**: インスタンスのホームディレクトリのパスの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103008 | ERROR | Unable to get the instance type information for resource "%s" | **原因**: DB2 Application Recovery Kit は、無効なインスタンスタイプを見つけました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103009 | ERROR | LifeKeeper has encountered an error while trying to get the database configuration parameters for database "$DB" | **原因**:「db2 get db cfg for $DB」コマンドの実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 103012 | ERROR | LifeKeeper was unable to start the database server for instance "%s" | **原因**: 要求された DB2 インスタンスの起動が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「restore」操作を再試行してください。 |
| 103013 | ERROR | LifeKeeper was unable to start the database server for instance "%s" | **原因**: 要求された DB2 インスタンスの起動が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してくだ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | さい。レポートされているエラーを修正してから、「restore」操作を再試行してください。 |
| 103015 | ERROR | An entry for the home directory "%s" of instance "%s" does not exist in "/etc/fstab" | **原因**: マルチパーティションのデータベースのインスタンスのホームディレクトリは、「/etc/fstab」に存在する必要があります。<br><br>**処置**: ホームディレクトリが「/etc/fstab」に存在することを確認してください。 |
| 103016 | ERROR | LifeKeeper was unable to mount the home directory for the DB2 instance "%s" | **原因**: マルチパーティションのデータベースのインスタンスのホームディレクトリをマウントできませんでした。<br><br>**処置**: ホームディレクトリがマウントされていることを確認し、操作を再試行してください。 |
| 103017 | ERROR | Unable to get the instance nodes information for resource "%s" | **原因**: インスタンスノードの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103018 | ERROR | LifeKeeper was unable to start database partition server "%s" for instance "%s" | **原因**: 要求された DB2 インスタンスの起動が失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「restore」操作を再試行してください。 |
| 103020 | ERROR | LifeKeeper was unable to stop the database server for instance "%s" | **原因**: 要求された DB2 インスタンスのシャットダウンが失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「remove」操作を再試行してください。 |
| 103021 | ERROR | LifeKeeper was unable to stop the database server for instance "%s" | **原因**: 要求された DB2 インスタンスのシャットダウンが失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「remove」操作を再試行してください。 |
| 103023 | ERROR | Unable to get the instance nodes information for resource "%s" | **原因**: インスタンスノードの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103024 | ERROR | LifeKeeper was unable to stop database partition server "%s" for instance "%s" | **原因**: 要求された DB2 インスタンスのシャットダウンが失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。レポートされているエラーを修正してから、「remove」操作を再試行してください。 |
| 103026 | ERROR | Unable to get the instance nodes information for resource "%s" | **原因**: インスタンスノードの取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103027 | FATAL | The argument for the DB2 instance is empty | **原因**: 作成操作で無効なパラメータが指定されました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: パラメータを確認し、操作を再試行してください。 |
| 103028 | FATAL | Unable to determine the DB2 instance home directory | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>**処置**: インスタンスの所有者の名前がインスタンス名と同じであることを確認し、操作を再試行してください。 |
| 103029 | FATAL | Unable to determine the DB2 instance type | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103030 | FATAL | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103031 | ERROR | The path "%s" is not on a shared filesystem | **原因**: インスタンスのホームディレクトリは、共有ファイルシステム上になければなりません。<br><br>**処置**: パスが共有ファイルシステム上にあることを確認し、作成操作を再試行してください。 |
| 103032 | ERROR | LifeKeeper was unable to get the DB tablespace containers for instance "%s" or the log path for one of its databases | **原因**: LifeKeeper はデータベースのテーブルスペースコンテナの場所を特定できなかったか、コンテナがマウント済みのファイルシステム上のパスにあることを確認できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | レポートされている問題を解決してください。レポートされているエラーを修正してから、「create」操作を再試行してください。 |
| 103033 | ERROR | The path "%s" is not on a shared filesystem | **原因**: データベースのテーブルスペースコンテナのパスは、共有ファイルシステム上になければなりません。<br><br>**処置**: データベースのテーブルスペースコンテナが共有ファイルシステム上にあることを確認し、操作を再試行してください。 |
| 103034 | ERROR | A DB2 Hierarchy already exists for instance "%s" | **原因**: LifeKeeper によってすでに保護されている DB2 インスタンスを保護しようとしました。<br><br>**処置**: 別の DB2 インスタンスを LifeKeeper の保護用に選択する必要があります。 |
| 103035 | ERROR | The file system resource "%s" is not in-service | **原因**: DB2 リソースが依存するファイルシステムは in service である必要があります。<br><br>**処置**: ファイルシステムリソースが in service であることを確認して、「create」操作を再試行してください。 |
| 103036 | ERROR | Unable to create the hierarchy for raw device "%s" | **原因**: LifeKeeper は、リソース {raw device} を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから「create」操作を再試行する必要があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103037 | ERROR | A RAW hierarchy does not exist for the tag "%s" | 原因: LifeKeeper は、raw リソース {tag} を見つけられませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 103038 | ERROR | LifeKeeper was unable to create a dependency between the DB2 hierarchy "%s" and the Raw hierarchy "%s" | 原因: 要求された親 DB2 リソースと子 Raw リソースとの間の依存関係の作成に失敗しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、「create」操作を再試行してください。 |
| 103039 | ERROR | LifeKeeper could not disable the automatic startup feature of DB2 instance "%s" | 原因: DB2 の設定を更新しようとして、予期しないエラーが発生しました。<br><br>処置: システム起動時にインスタンスが自動で起動されないようにするために、DB2AUTOSTART を手動で更新する必要があります。 |
| 103040 | ERROR | DB2 version "%s" is not installed on server "%s" | 原因: LifeKeeper は DB2 のインストール場所を見つけられませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103041 | ERROR | The instance owner "%s" does not exist on target server "%s" | 原因:「canextend」操作または「extend」操作の間に、テンプレートサーバから DB2 インスタンスの所有者を取得できませんでした。<br><br>処置: 指定したサーバに DB2 インスタンスの所有者が存在していることを確認してください。所有者が存在しない場合、クラスタ内のすべてのサーバで、同じ uid および gid を使用して作成する必要 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | があります。 |
| 103042 | ERROR | The instance owner "%s" uids are different on target server "%s" and template server "%s" | 原因: DB2 インスタンスの所有者 {user} のターゲットサーバ {target server} のユーザ ID が、テンプレートサーバ {template server} のユーザ {user} の値と一致しません。<br><br>処置: DB2 インスタンスの所有者 {user} のユーザ ID はクラスタ内のすべてのサーバで一致している必要があります。ユーザ ID の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103043 | ERROR | The instance owner "%s" gids are different on target server "%s" and template server "%s" | 原因:DB2 インスタンスの所有者 {user} のターゲットサーバ {target server} のグループ ID が、テンプレートサーバ {template server} のユーザ {user} の値と一致しません。<br><br>処置: DB2 インスタンスの所有者 {user} のグループ ID はクラスタ内のすべてのサーバで一致している必要があります。グループ ID の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103044 | ERROR | The instance owner "%s" home directories are different on target server "%s" and template server "%s" | 原因: ターゲットサーバ {target server} のユーザ {user} のホームディレクトリが、テンプレートサーバ {template server} の DB2 インスタンスの所有者のホームディレクトリと一致しません。<br><br>処置: DB2 インスタンスの所有者 {user} のホームディレクトリは、クラスタ内のすべてのサーバで一致している必要があります。ディレクトリの不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103045 | ERROR | LifeKeeper was unable to get the DB2 "SVCENAME" parameter for the DB2 instance | 原因:「db2 get dbm cfg」コマンドの実行中に予期しないエラーが発生しました。<br><br>処置: DB2 の設定を確認してください。 |
| 103046 | ERROR | Unable to get the value of the DB2 "SVCENAME" parameter for the DB2 instance %s. | 原因: DB2 の「SVCENAME」パラメータが null に設定されています。<br><br>処置: DB2 の設定を確認してください。 |
| 103047 | ERROR | LifeKeeper was unable to get the contents of the "/etc/services" file on the server "%s" | 原因: テンプレートサーバの「/etc/services」に、DB2 インスタンスのサービス名が含まれていません。<br><br>処置:「/etc/services」内にある DB2 インスタンスのサービス名は、クラスタ内のすべてのサーバで一致している必要があります。サービス名の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103048 | ERROR | LifeKeeper was unable to get the contents of the "/etc/services" file on the server "%s" | 原因: ターゲットサーバの「/etc/services」に、DB2 インスタンスのサービス名が含まれていません。<br><br>処置:「/etc/services」内にある DB2 インスタンスのサービス名は、クラスタ内のすべてのサーバで一致している必要があります。サービス名の不一致をすべてのサーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103049 | ERROR | The "/etc/services" entries for the instance "%s" are different on target server "%s" and template server "%s" | 原因: インスタンスの「/etc/services」エントリに不一致があります。<br><br>処置:「/etc/services」内にある DB2 インスタンスのサービス名は、クラスタ内のすべてのサーバで一致している必要があります。サービス名の不一致をすべての |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | サーバで手動で修正し、「canextend」操作を再試行する必要があります。 |
| 103050 | ERROR | The home directory "%s" for instance "%s" is not mounted on server "%s" | **原因**: LifeKeeper はマルチパーティションインスタンス用の db2nodes.cfg を見つけられませんでした。<br><br>**処置**: ホームディレクトリがマウントされていることを確認し、操作を再試行してください。 |
| 103051 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: テンプレートサーバからのリソース情報の取得に失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 103052 | ERROR | LifeKeeper was unable to add instance "%s" and/or its variables to the DB2 registry | **原因**:「db2iset」コマンドの実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 103053 | ERROR | Usage: %s instance | |
| 103054 | ERROR | Unable to determine the DB2 instance type | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103055 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>**処置**: DB2 の設定を確認してください。 |
| 103056 | ERROR | Usage: %s instance | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103058 | ERROR | Usage: %s instance | |
| 103059 | ERROR | Usage: %s instance | |
| 103060 | ERROR | Unable to determine the DB2 instance home directory | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>処置: インスタンスの所有者の名前がインスタンス名と同じであることを確認し、操作を再試行してください。 |
| 103061 | ERROR | Unable to determine the DB2 instance type | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103062 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | 原因: DB2 Application Recovery Kit は、DB2 インスタンスの該当するノードを見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103063 | ERROR | Unable to determine the DB2 install path | 原因: DB2 Application Recovery Kit は、そのインスタンスの DB2 を見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103064 | ERROR | Usage: nodes -t tag -a add_nodenum \| nodes -t tag -d delete_nodenum \| nodes -t tag -p | |
| 103065 | ERROR | Invalid input provided for "%s" utility operation, characters are not allowed. | 原因:「nodes」コマンドで無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103066 | ERROR | Unable to get the information for resource "%s" | **原因**: LifeKeeper は、リソース {tag} を見つけられませんでした。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 103067 | ERROR | The DB2 instance "%s" is not a EEE or Multiple Partition instance | **原因**: リソースタグ {tag} はシングルパーティションインスタンスです。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 103069 | ERROR | Node "%s" is already protected by this hierarchy | **原因**:「nodes」コマンドで無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 103070 | ERROR | Node number "%s" is the last remaining node protected by resource "%s". Deleting all nodes is not allowed. | **原因**:「nodes」コマンドで無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 103071 | ERROR | LifeKeeper is unable to get the equivalent instance for resource "%s" | **原因**:「nodes」コマンドの実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 103072 | ERROR | Unable to set NodesInfo for resource "%s" on "%s" | **原因**:「nodes」コマンドの実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 103073 | ERROR | Unable to set NodesInfo for resource "%s" on "%s" | 原因:「nodes」コマンドの実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 103074 | ERROR | Usage: %s instance | |
| 103075 | ERROR | Usage: %s instance | |
| 103076 | ERROR | Unable to determine the DB2 instance type | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのタイプを特定できませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103077 | ERROR | Unable to determine the DB2 instance home directory | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>処置: インスタンスの所有者の名前がインスタンス名と同じであることを確認し、操作を再試行してください。 |
| 103078 | ERROR | The database server is not running for instance "%s" | 原因: DB2 インスタンスのプロセスチェックで、実行中のプロセスが見つかりませんでした。<br><br>処置: DB2 インスタンスを開始してください。 |
| 103079 | ERROR | LifeKeeper has detected an error while trying to determine the node number(s) of the DB partition server(s) for the instance | 原因: DB2 Application Recovery Kit は、DB2 インスタンスのノードを見つけることができませんでした。<br><br>処置: DB2 の設定を確認してください。 |
| 103080 | ERROR | One or more of the database partition servers for instance "%s" is down | 原因: すべてのデータベースパーティシ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ョンサーバが実行されている必要があります。<br><br>**処置**: すべてのデータベースパーティションサーバが実行されていることを確認し、操作を再試行してください。 |
| 103081 | ERROR | DB2 local recovery detected another recovery process in progress for "%s" and will exit. | |
| 103082 | ERROR | Failed to create flag "%s" | **原因**: DB2 ローカルリカバリ処理を制御するためのフラグを作成しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 103083 | ERROR | Failed to remove flag "%s" | **原因**: DB2 ローカルリカバリ処理を制御するためのフラグを削除しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 103084 | ERROR | Unable to determine the DB2 instance \"$Instance\" home directory | **原因**: DB2 Application Recovery Kit は、DB2 インスタンスのホームディレクトリを特定できませんでした。<br><br>**処置**: インスタンスの所有者の名前がインスタンス名と同じであることを確認し、操作を再試行してください。 |

# 8.1.3. DMMP Kit メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| 128005 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: オペレーティングシステムの信号 {signal} によって、{server} 上の {resource} の quickCheck が失敗しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128008 | ERROR | Usage: quickCheck -t <tag name> -i <id> | 原因: dmmp デバイスの quickCheck コマンドに不正な引数が指定されたため、コマンドが実行できませんでした。<br><br>処置: すべてのソフトウェアコンポーネントの正しいバージョンが適切にインストールされていることを確認してください。次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (quickCheck 対象の dmmp デバイスリソースを特定する ID) |
| 128010 | ERROR | quickCheck for "%s" failed checks of underlying paths, initiate recovery. retry count=%s. | 原因: dmmp キットが {count} 回の試行後、デバイスの quickCheck に失敗しました。保護されている dmmp リソースのリカバリが実行されます。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| 128021 | ERROR | unable to find device for uuid "%s". | **原因**: restore 操作時に一意の ID を使用してデバイスを見つけられませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。リストア対象の dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |
| 128025 | ERROR | Device "%s" failed to unlock. | **原因**: restore 操作時に正常に動作していない {device} が検出され、ロック解除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128026 | ERROR | Device "%s" failed to lock. | **原因**: リストア時に {device} をロックできませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128031 | ERROR | unable to find device for uuid "%s". | **原因**: remove 操作時に一意の ID を使用してデバイスを見つけられませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。削除対象の dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| 128034 | ERROR | Device "%s" failed to unlock. | **原因**: 削除時に {device} をロック解除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128036 | ERROR | unable to load existing information for device with uuid "%s". | **原因**: 一意の ID を使用してデバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |
| 128037 | ERROR | unable to load existing information for device "%s". | **原因**: 名前を使用してデバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。dmmp デバイスリソースを特定する正しいデバイス名でコマンドを再実行してください。 |
| 128038 | ERROR | unable to load existing information for device, no dev or uuid defined. | **原因**: 一意のデバイス ID もデバイス名も指定されていないため、デバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されていることを確認してください。dmmp デバイスリソースを特定する正しいデバイス ID またはデバイス名でコマンドを再実行してください。 |
| 128041 | ERROR | unable to load existing information for device with uuid "%s". | **原因**: 一意の ID を使用してデバイス情報を読み込めませんでした。<br><br>**処置**: リソースが正しく設定されている |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | ことを確認してください。dmmp デバイスリソースを特定する正しいデバイス ID でコマンドを再実行してください。 |
| 128057 | ERROR | All paths are failed on "%s". | 原因: LifeKeeper は、保護対象の dmmp デバイスへのすべてのパスが障害状態にあることを検出しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128058 | ERROR | could not determine registrations for "%s"! All paths failed. | 原因: LifeKeeper は、保護対象の dmmp {device} の登録を特定できませんでした。dmmp {device} へのすべてのパスが障害状態にあります。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128059 | WARN | path "%s" no longer configured for "%s", remove from path list. | 原因:LifeKeeper が検出した保護対象の {device} への {path} は、現在は有効ではなく、パスリストから削除されます。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128060 | WARN | registration failed on path "%s" for "%s". | 原因: LifeKeeper は、保護対象の dmmp {device} の {path} の登録を特定できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128062 | ERROR | all paths failed for "%s". | 原因: LifeKeeper は、保護対象の dmmp |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | {device} の有効なパスを確認できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128072 | ERROR | The daemon "%s" does not appear to be running and could not be restarted. Path failures may not be correctly handled without this daemon. | **原因**: LifeKeeper は dmmp デーモンが実行されていることを確認できなかったため、デーモンを再起動できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128078 | ERROR | "%s" resource type is not installed on "%s". | **原因**: dmmp デバイスをサポートするための Device Mapper Multipath Recovery Kit がシステムにインストールされていません。<br><br>**処置**: steeleye-lkDMMP Device Mapper Multipath Recovery Kit rpm をシステムにインストールしてください。 |
| 128083 | ERROR | This script must be executed on "%s". | **原因**: dmmp デバイスリソースを作成するための devicehier スクリプトの引数に不正なシステム名が指定されました。<br><br>**処置**: クラスタノードとコミュニケーションパスが適切に設定されていることを確認してください。devicehier スクリプトに正しいシステム名を指定してください。この名前は、コマンドを実行するシステムの名前に一致する必要があります。 |
| 128084 | ERROR | The device %s is not active. | **原因**: リソース作成時に LifeKeeper は、指定された {device} をシステム上の有 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | 効なデバイスとして見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (作成する dmmp デバイスリソースを特定する ID) |
| 128086 | ERROR | Failed to create "%s" hierarchy. | **原因**: LifeKeeper は、{device} のためのリソース階層を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128088 | ERROR | Error creating resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} で {tagname} を使用してリソースを作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128090 | ERROR | Failed to create dependency "%s"-"%s" on system "%s". | **原因**: LifeKeeper は、{system} で依存関係 {resource tag name} – {resource tag name} を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| 128091 | ERROR | Error creating resource "%s" on server "%s" | 原因: LifeKeeper は、{system} で {resource} を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128101 | ERROR | "%s" constructor requires a valid argument. | 原因: LifeKeeper は、dmmp リソースのオブジェクトを作成できませんでした。<br><br>処置: 次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (dmmp デバイスリソースを特定する ID) |
| 128102 | ERROR | Invalid tag "%s". | 原因: 指定されたタグ名のリソースインスタンスが見つかりませんでした。<br><br>処置: リソースが正しく設定されていることを確認してください。次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (dmmp デバイスリソースを特定する ID) |
| 128111 | ERROR | Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations. | 原因: LifeKeeper は {device} の登録の取得に失敗し、メッセージ「bad field in Persistent reservation in cdb」を出力しました。<br><br>処置: このストレージでパーシステントリザベーションがサポートされているか確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| 128112 | ERROR | Failed to get registrations for "%s": %s. Verify the storage supports persistent reservations. | 原因: LifeKeeper は {device} の登録の取得に失敗し、メッセージ「illegal request」を出力しました。<br><br>処置: このストレージでパーシステントリザベーションがサポートされているか確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128136 | ERROR | A previous quickCheck with PID "%s" running for device "%s" has been terminated. | 原因: LifeKeeper は、dmmp リソースの restore 操作時に前の quickCheck がまだ実行中であることを検出しました。LifeKeeper は quickCheck を終了しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128137 | ERROR | SCSI reservation conflict on %s during LifeKeeper resource initialization. Manual intervention required. | 原因: LifeKeeper は、dmmp リソースの restore 操作時に {device} で SCSI リザベーションコンフリクトを検出しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。手動による介入および {device} でのリザベーションコンフリクトの解消が必要です。 |
| 128138 | ERROR | unable to clear registrations on %s. | 原因: LifeKeeper は、{device} 上のすべての登録を消去できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要がありま |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | す。 |
| 128140 | WARN | registration failed on path %s for %s. | 原因: LifeKeeper は、{device} の {path} の登録を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128143 | ERROR | reserve failed (%d) on %s. | 原因: LifeKeeper は、{device} 上の {resource} の登録を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128145 | ERROR | The server ID "%s" returned by "%s" is not valid. | 原因: LifeKeeperは有効なホスト {ID} を生成できませんでした。<br><br>処置: デバイスを登録するための ID は、クラスタ内のサーバを一意に特定する 1 ～ 12 桁の 16 進数で構成されます。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128146 | ERROR | device failure on %s. SYSTEM HALTED. | 原因:LifeKeeper は {device} で障害を検出しました。サーバを再起動します。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| 128148 | ERROR | device failure on %s. SYSTEM HALTED DISABLED. | 原因: LifeKeeper は {device} で障害を検出しました。LifeKeeper の設定によって、再起動はスキップされました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。「SCSIHALT」設定を有効にして、いかなるデバイス障害が検出されたときも再起動が行われるようにしてください。 |
| 128149 | ERROR | device failure or SCSI Error on %s. SENDEVENT DISABLED. | 原因: LifeKeeper は {device} で障害を検出しました。LifeKeeper の設定によって、イベント生成はスキップされました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。「SCSIEVENT」設定を有効にして、いかなるデバイス障害が検出されたときも sendevent を使用できるようにしてください。 |
| 128150 | ERROR | %s does not have EXCLUSIVE access to %s, halt system. | 原因: LifeKeeper は {server} の {device} でリザベーションコンフリクトを検出しました。サーバを再起動します。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 128151 | ERROR | %s does not have EXCLUSIVE access to %s, halt system DISABLED. | 原因: LifeKeeper は {server} の {device} でリザベーションコンフリクトを検出しました。LifeKeeper の設定によって、再起動はスキップされました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。「RESERVATIONCONFLICT」設定を有効にして、いかなるリザベーションコンフリクトが検出されたときも再起動 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | が行われるようにしてください。 |
| 128154 | WARN | unable to flush buffers on %s. | **原因**: LifeKeeper は、dmmp リソースの削除時に {device} のバッファを消去できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128157 | WARN | %s utility not found, limited healthcheck for %s. | **原因**: LifeKeeper は、{device} の健全性チェックを行う「dd」ユーティリティを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128160 | ERROR | %s failed to read %s. | **原因**: LifeKeeper は、{utility} の使用中に {device} のディスク I/O テストに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128163 | ERROR | Registration ID "%s" for "%s" is not valid. | **原因**: LifeKeeper は {device} の有効な登録 {ID} の生成に失敗しました。<br><br>**処置**: デバイスを登録するための ID は、デバイスへのパスに由来する 4 桁の 16 進数で構成されます。隣接するログメッセージで詳細および関連メッセージを確 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | 認し、レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128170 | ERROR | Usage: canextend <Template system name> <Template tag name> | |
| 128500 | ERROR | Usage error | 原因: dmmp デバイスの restore コマンドに不正な引数が指定されたため、コマンドが実行できませんでした。<br><br>処置: 次の正しい引数を指定してコマンドを再実行してください。-t <Resource Tag> および -i <Resource ID> (リストアする dmmp デバイスリソースを特定する ID) |
| 128504 | ERROR | "%s" resource type is not installed on "%s". | 原因: dmmp デバイスをサポートするための Device Mapper Multipath Recovery Kit がシステムにインストールされていません。<br><br>処置: steeleye-lkDMMP Device Mapper Multipath Recovery Kit rpm をシステムにインストールしてください。 |
| 128506 | ERROR | Usage error | 原因: dmmp デバイスの devShared コマンドに不正な引数が指定されたため、コマンドが実行できませんでした。<br><br>処置: 次の正しい引数を指定してコマンドを再実行してください。<Template Resource System Name> および <Template Resource Tag> (作成する dmmp デバイスリソースを特定するタグ) |
| 128507 | FATAL | This script must be executed on "%s". | 原因: dmmp デバイスリソースを作成するための devicehier スクリプトの引数 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | に不正なシステム名が指定されました。<br><br>**処置**: devicehier スクリプトに正しいシステム名を指定してください。この名前は、コマンドを実行するシステムの名前に一致する必要があります。 |
| 128511 | ERROR | Failed to get the ID for the device "%s". Hierarchy create failed. | **原因**: dmmp デバイスリソースを作成するための devicehier スクリプトは、指定されたデバイスの SCSI ID を特定できませんでした。<br><br>**処置**: 指定されたデバイスパスが存在すること、およびそのパスがサポート対象の SCSI ストレージアレイのパスであることを確認してください。 |
| 128512 | ERROR | Failed to get the disk ID for the device "%s". Hierarchy create failed. | **原因**: dmmp ディスクリソースを作成するための devicehier スクリプトは、指定されたディスクの SCSI ID を特定できませんでした。<br><br>**処置**: 指定されたデバイスパスが存在すること、およびそのパスがサポート対象の SCSI ストレージアレイのパスであることを確認してください。 |
| 128513 | ERROR | Failed to create the underlying resource for device "%s". Hierarchy create failed. | **原因**: ベースとなる dmmp ディスクリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128515 | ERROR | Error creating resource "%s" on server "%s" | **原因**: dmmp デバイスリソースの作成に失敗しました。 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128517 | ERROR | Failed to create dependency "%s"-"%s" on system "%s". | **原因**: dmmp デバイスリソースと dmmp ディスクリソースとの間の親 / 子の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128519 | ERROR | Error creating resource "%s" on server "%s" | **原因**: 新しく作成した dmmp デバイスリソースを in service にする操作が失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 128521 | ERROR | Either TEMPLATESYS or TEMPLATETAG argument missing | **原因**: dmmp デバイスリソースの extend コマンドに不正な引数が指定されました。<br><br>**処置**:テンプレートシステムおよびタグの正しい名前を指定して dmmp デバイスリソースの extend を再実行してください。 |
| 128540 | ERROR | Usage error | **原因**: dmmp デバイスの SCSI ID を取得する getId コマンドに不正な引数が指定されました。 |

| Code | Severity | Message | Cause/Action |
|---|---|---|---|
| | | | **処置**: 次の正しい引数を指定してコマンドを再実行してください。-i <device path> または -b <device ID>。 |
| 128541 | ERROR | Usage error | **原因**: dmmp デバイスリソースを削除するコマンドに不正な引数が指定されました。<br><br>**処置**: 次の正しい引数を指定してコマンドを再実行してください。-t <dmmp device resource tag>。 |
| 128543 | ERROR | device node "$dev" does not exist. | **原因**: dmmp デバイスリソースをリストアするために必要なデバイスノードが存在しません。リストア時に udev デバイスの作成に割り当てられている待機時間を越えました。<br><br>**処置**: udev がデバイスを作成した後に dmmp デバイスリソースの restore を再実行してください。 |
| 128544 | ERROR | Usage error | **原因**: dmmp デバイスリソースを out of service にする remove コマンドに不正な引数が指定されました。<br><br>**処置**: 次の正しい引数を指定してコマンドを再実行してください。-t <dmmp device resource tag>。 |
| 128550 | ERROR | Failed to get path information.(tag=\"$opt_t\" | |
| 128551 | ERROR | Failed to get the status of path.(tag=\"$opt_t\" | |
| 128552 | ERROR | Failed to check the status of path.(tag=\"$opt_t\" | |
| 128553 | ERROR | Failed to get the resource information.(tag=\"$opt_t\" | |

# 8.1.4. EC2 Kit メッセージカタログ

以下の Recovery Kit for EC2 メッセージカタログには、Recovery Kit for EC2 の使用中に表示される可能性のあるすべてのメッセージのリストを記載しています。

総合メッセージカタログ は、SIOS Protection Suite for Linux の使用中に発生する可能性のあるすべてのメッセージの一覧を提供し、必要に応じて、エラーの原因およびエラー状態を解消するために必要な操作に関する追加の説明も記載しています。このリストで、受信したエラーコードを検索できます。

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 129100 | FATAL | Failed to load instance from LifeKeeper. | 原因: 無効なリソースタグまたは ID が指定されました。<br><br>処置: タグまたは ID が有効であることを確認し、コマンドを再実行してください。 |
| 129103 | FATAL | No resource matches tag "$self->{'tag'}". | 原因: 無効なリソースタグが指定されました。<br><br>処置: タグが有効であることを確認し、コマンドを再実行してください。 |
| 129104 | FATAL | An error occurred setting LifeKeeper resource information | 原因: LifeKeeper で内部エラーが発生しました。 |
| 129110 | ERROR | Could not get the Elastic Network Interface ID for $dev | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129111 | ERROR | Failed to get Allocation ID of Elastic IP "$elasticIp". | 原因: EC2 API の呼び出しに失敗しまし |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | た。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129113 | ERROR | Failed to get my instance ID. | **原因**: EC2 インスタンスのメタデータのアクセスに失敗しました。<br><br>**処置**: Amazon コンソールを確認し、操作を再試行してください。 |
| 129114 | ERROR | Failed to get ENI ID. | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129116 | ERROR | Failed to associate Elastic IP "$self->{'EIP'}" on "$self->{'DEV'}". | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129118 | WARN | $self->{'EIP'} is not associated with any instance. | **原因**: Elastic IP がインスタンスに関連付けられていません。<br><br>**処置**: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129119 | WARN | $self->{'EIP'} is associated with another instance. | **原因**: Elastic IP が別のインスタンスに関 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 連付けられています。<br><br>**処置**: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129120 | ERROR | Failed to recover Elastic IP. | **原因**: EC2 API が Elastic IP の関連付けに失敗しました。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129121 | ERROR | Recovery process ended but Elastic IP is not associated with this instance. Please check AWS console. | **原因**: EC2 API が Elastic IP の関連付けに失敗しました。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129122 | ERROR | Error creating resource \"$target_tag\" with return code of \"$err\". | **原因**: LifeKeeper がサーバ上にリソースインスタンスを作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129123 | ERROR | Failed to get ENI ID. | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129124 | WARN | $self->{'EIP'} is associated with another network interface. | **原因**: Elastic IP は正しいインスタンスに関連付けられていますが、ENI が正しく |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ありません。<br><br>**処置**: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129125 | ERROR | Link check failed for interface '$dev'. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: ネットワークインターフェースを確認し、リンクをアップ状態にしてください。 |
| 129126 | ERROR | Link check failed for interface '$dev'. Reason: down slave. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: ネットワークインターフェースを確認し、リンクをアップ状態にしてください。 |
| 129129 | WARN | The link for network interface '$self->{'DEV'}' is down. Attempting to bring the link up. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: LifeKeeper は、リンクをアップ状態にして Elastic IP とインターフェースを関連付けることにより、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129130 | ERROR | Failed to modify "$opt_t" to end pint URL "$endpoint". | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 129137 | ERROR | The link for network interface '$self->{'DEV'}' is still down. | **原因**: LifeKeeper がリンクをアップ状態にできませんでした。<br><br>**処置**: インターフェースが有効であり、アップ状態であることを確認してください。隣接するログメッセージで詳細を確認してください。 |
| 129139 | WARN | The link for network interface '$self->{'DEV'}' is down. | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、リンクが存在しないことを示します。<br><br>**処置**: ネットワークインターフェースを確認し、リンクをアップ状態にしてください。 |
| 129140 | ERROR | Could not get ENI ID for $self->{IP}. | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129142 | ERROR | Failed to update route table | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129143 | ERROR | You must have exactly one IP address resource as the parent of the RouteTable EC2 resource. Please reconfigure your resource hierarchy. | **原因**: ルートテーブルの EC2 リソースは親として IP リソースを 1 つのみ持つ必要があります。<br><br>**処置**: 必要に応じてリソース階層を修復してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 129144 | ERROR | $func called with invalid timeout: $timeout | 原因: /etc/default/LifeKeeper ファイルに無効なタイムアウト値が指定されました。<br><br>処置: /etc/default/LifeKeeper 内にあるすべての EC2_*_TIMEOUT 設定が正しいことを確認してください。 |
| 129145 | ERROR | $func action timed out after $timeout seconds | 原因: タイムアウト期間中に動作が完了しませんでした。<br><br>処置: その動作の EC2_*_TIMEOUT 値 (/etc/default/LifeKeeper 内) を増加することを検討してください。 |
| 129146 | ERROR | failed to run $func with timeout: $@ | 原因: これは内部エラーです。 |
| 129148 | ERROR | Amazon describe-route-tables call failed (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129150 | ERROR | Elastic IP \"$elasticIp\" is associated with another instance. | 原因: Elastic IP が正しいインスタンスに関連付けられていません。<br><br>処置: LifeKeeper は Elastic IP を関連付ける EC2 API を呼び出して、問題の修正を試行します。隣接するログメッセージで詳細を確認してください。 |
| 129151 | ERROR | Could not get the Association ID for Elastic IP "$elasticIp". | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソー |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ルを確認し、操作を再試行してください。 |
| 129152 | ERROR | Failed to disassociate Elastic IP "$self->{'EIP'}" on "$self->{'DEV'}". | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129153 | ERROR | Failed to disassociate Elastic IP "$elasticIp", (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129154 | ERROR | Amazon describe-addresses call failed (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129155 | ERROR | Amazon describe-address call failed (err=%s)(output=%s) | 原因: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>処置: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129157 | ERROR | curl call failed (err=%s)(output=%s) | 原因: EC2 インスタンスのメタデータのアクセスに失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: Amazon コンソールを確認し、操作を再試行してください。 |
| 129159 | ERROR | Amazon associate-address call failed (err=%s)(output=%s) | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129160 | ERROR | Amazon describe-addresses call failed (err=%s)(output=%s) | **原因**: EC2 API の呼び出しに失敗しました。ネットワークの問題と考えられます。<br><br>**処置**: ネットワークと Amazon コンソールを確認し、操作を再試行してください。 |
| 129161 | ERROR | Error deleting resource "$otherTag" on "$otherSys" with return code of "$err". | **原因**: LifeKeeper がサーバ上のリソースインスタンスを削除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129162 | ERROR | Could not getRouteTablesByIP | |
| 129163 | ERROR | Could not getRouteTablesByIP | |
| 129164 | ERROR | [$SUBJECT event] mail returned $err | |
| 129165 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 129167 | ERROR | snmptrap returned $err for Trap 180 | |
| 129168 | ERROR | COMMAND OUTPUT: cat /tmp/err$$ | |
| 129170 | ERROR | This resource is in the old format. Please update. | |
| 129403 | ERROR | END failed create of $TAG due to a $sig signal | **原因**: 作成プロセスがシグナルによって |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 中断されました。 |
| 129409 | ERROR | The IP resource $IP_RES is not \"ISP\". | **原因**: IP リソースが in service ではありません。<br><br>**処置**: リソースを in service にして、操作をもう一度行ってください。 |
| 129410 | ERROR | Could not find IP resource $IP_RES | **原因**: IP リソースが存在していることを確認し、操作を再試行してください。 |
| 129412 | ERROR | EC2 resource $ID is already protected | **原因**: 指定した ID を持つリソースはすでに存在しています。<br><br>**処置**: 残っている古いリソースをすべてクリーンアップしてから、新しいリソースを再作成してください。 |
| 129416 | ERROR | Error creating resource "$TAG" with return code of "$lcderror". | **原因**: LifeKeeper がリソースインスタンスを作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129418 | ERROR | Dependency creation between "$IP_RES" and "$TAG" failed with return code of "$lcderror". | **原因**: LifeKeeper がリソースの依存関係を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129420 | ERROR | In-service failed for tag "$TAG". | **原因**: LifeKeeper がリソースインスタンスを in service にできませんでした。<br><br>**処置**: 隣接するログメッセージで詳細お |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | よび関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129423 | ERROR | Could not get ENI ID for $dev. | |
| 129425 | ERROR | Failed to update route table | |
| 129426 | ERROR | In-service (dummy) failed for tag "$TAG". | |
| 129800 | ERROR | canextend checks failed for \"$self->{‘tag’}\" (err=$ret) | **原因**: ターゲットサーバで拡張前のチェックに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 129801 | ERROR | canextend checks failed for "$self->{'tag'}". EC2_HOME "$self->{EC2_HOME}" does not exist on $me. | |

# 8.1.5. ファイルシステムキットメッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124004 | FATAL | resource tag name not specified | **原因**:「quickCheck」操作で無効な引数が指定されました。<br><br>**処置**: 正しい引数が渡されたことを確認してください。 |
| 124005 | FATAL | resource id not specified | **原因**:「quickCheck」操作で無効な引数が指定されました。<br><br>**処置**: 正しい引数が渡されたことを確認してください。 |
| 124007 | FATAL | Failed to get resource information | **原因**: ファイルシステムリソースの情報フィールドに正しい情報が含まれていません。<br><br>**処置**: リソースの情報フィールドに正しい情報を入力するか、最近の「lkbackup」からシステムをリストアして元の情報フィールドをリストアします。 |
| 124008 | ERROR | getId failed | **原因**: ファイルシステムリソースは基本ディスクデバイスを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。リソース階層が有効であること、および必要なストレージキットがすべてインストールされていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124009 | ERROR | LifeKeeper protected filesystem is in service but quickCheck detects the following error: | **原因**: ファイルシステムキットはリソースに問題を発見しました。<br><br>**処置**: この直後に表示されるメッセージで詳細を確認してください。 |
| 124010 | ERROR | "$id" is not mounted | **原因**: ファイルシステムリソースがマウントされていません。<br><br>**処置**: 処置は必要ありません。ローカルリカバリでリソースの再マウントを許可してください。 |
| 124011 | ERROR | "$id" is mounted but with the incorrect mount options (current mount option list: $mntopts, expected mount option list: $infoopts) | **原因**: ファイルシステムリソースが不正にマウントされています。<br><br>**処置**: 処置は必要ありません。ローカルリカバリでリソースの再マウントを許可してください。 |
| 124012 | ERROR | "$id" is mounted but on the wrong device (current mount device: $tmpdev, expected mount device: $dev) | **原因**: ファイルシステムリソースに誤ったデバイスがマウントされています。<br><br>**処置**: 処置は必要ありません。ローカルリカバリでリソースの再マウントを許可してください。 |
| 124015 | ERROR | LifeKeeper protected filesystem "$tag" ($id) is $percent% full ($blocksfree free blocks). | **原因**: ディスク使用量が FILESYSFULLERROR の値（%）を超過しました。<br><br>**処置**: ファイルシステムからデータを削除または移行してください。 |
| 124016 | WARN | LifeKeeper protected filesystem "$tag" ($id) is $percent% full ($blocksfree free blocks). | **原因**: ディスク使用量が FILESYSFULLWARNの値（%）を超過しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: ファイルシステムからデータを削除または移行してください。 |
| 124020 | FATAL | cannot find device information for filesystem $id | **原因**: ファイルシステムリソースは基本ディスクデバイスを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。リソース階層が有効であること、および必要なストレージキットがすべてインストールされていることを確認してください。 |
| 124029 | ERROR | Failed to find child resource. | **原因**: ファイルシステムリソースは基本ディスクリソースを特定できませんでした。<br><br>**処置**: リソース階層が正しいことを確認してください。 |
| 124032 | FATAL | Script has hung. Exiting. | **原因**: アンマウントする必要があるものの、マウントされているファイルシステムに開かれているファイルが複数のプロセスにありました。それらのプロセスの終了に時間がかかりすぎています。<br><br>**処置**: このエラーが続く場合は、マウントポイントを使用している可能性のあるすべてのソフトウェアを一時的に停止し、アンマウントできるようにしてください。ファイルシステムをそれでもアンマウントできない場合は、サポートに問い合わせてください。 |
| 124042 | ERROR | file system $fs failed unmount; will try again | **原因**: アンマウントする必要があるものの、マウントされているファイルシステムに開かれているファイルが複数のプロセスにありました。場合によっては、それらのプロセスを消去するために、何度 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | か操作を実行する必要があります。<br><br>**処置**: 処置は必要ありません。プロセスをそのまま続行してください。 |
| 124046 | ERROR | file system $fsname failed unmount | **原因**: ファイルシステムをアンマウントできませんでした。<br><br>**処置**: このエラーが続く場合は、マウントポイントを使用している可能性のあるすべてのソフトウェアを一時的に停止し、アンマウントできるようにしてください。ファイルシステムをそれでもアンマウントできない場合は、サポートに問い合わせてください。 |
| 124049 | ERROR | Local recovery of resource has failed (err=$err) | **原因**: ファイルシステムリソースに、ローカルでは修正できない問題があります。<br><br>**処置**: 処置は必要ありません。リソースを別のシステムにフェイルオーバできるようにしてください。 |
| 124051 | WARN | getId failed, try count : $cnt/$try | |
| 124052 | ERROR | \"$id\" is mounted but filesystem is shutdown state. | |
| 124054 | ERROR | Failed to change mount option from \”$old_opts\” to \”$new_opts\” for migration | |
| 124103 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124104 | ERROR | $ERRMSG Required template machine name is null | **原因**: canextend 操作で無効な引数が指 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 定されました。<br><br>**処置**: 引数が正しいことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124105 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: canextend 操作で無効な引数が指定されました。<br><br>**処置**: 引数が正しいことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124106 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: リソースの基本ディスク情報を特定できませんでした。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124107 | ERROR | $ERRMSG Resource "$TemplateTagName" must have one and only one device resource dependency | **原因**: リソースの階層にある基本デバイスが多すぎます。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124108 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: テンプレートシステムでリソースを見つけられません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124109 | ERROR | $ERRMSG Can not access canextend for scsi/$DeviceResType resources on machine "$TargetSysName" | **原因**: ターゲットシステムに、必須のコンポーネントがいくつか見つかりませ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ん。<br><br>**処置**: ターゲットシステムに正しいキットがすべてインストールされ、ライセンスされていることを確認してください。 |
| 124110 | ERROR | $ERRMSG Either filesystem "$TemplateLKId" is not mounted on "$TemplateSysName" or filesystem is not shareable with "$TargetSysName" | **原因**: ファイルシステムがテンプレートシステムのサービス内にないか、ターゲットシステムに拡張するための要件を満たしていません。<br><br>**処置**: リソースがテンプレートシステムのサービス内にあることを確認し、ファイルシステムを拡張するための要件に関する製品ドキュメントを参照してください。 |
| 124111 | ERROR | $ERRMSG File system type "${FSType}" is not supported by the kernel currently running on "${TargetSysName}" | **原因**: カーネルのサポートがないため、ファイルシステムのタイプをターゲットシステムにマウントできません。<br><br>**処置**: ターゲットシステムにそのカーネルモジュールがすべてインストールされており、正常に設定されていることを確認してから、リソースを拡張してください。 |
| 124112 | ERROR | must specify machine name containing primary hierarchy | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124113 | ERROR | must specify primary ROOT tag | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてくだ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | さい。 |
| 124114 | ERROR | must specify primary mount point | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124115 | ERROR | must specify primary switchback type | **原因**: creFShier 操作で無効な引数が指定されました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124118 | ERROR | dep_remove failure on machine \""$PRIMACH"" for parent "$PRITAG" and child "$DEVTAG." | **原因**: 依存関係の作成後のクリーンアップに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124119 | ERROR | ins_remove failure on machine ""$PRIMACH"" for \"$PRITAG." | **原因**: インスタンスの作成後のクリーンアップに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124121 | ERROR | ins_remove failure on machine ""$PRIMACH" | **原因**: リソースの作成後のクリーンアップに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124122 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: このメッセージは、通常の状態では表示されません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124123 | ERROR | $ERRMSG Required template machine name is null | **原因**: depstoextend 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124124 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: depstoextend 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124125 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: リソースが基本ディスクリソースを特定できませんでした。<br><br>**処置**: 階層とすべての依存関係が正しいことを確認してから拡張してください。 |
| 124126 | ERROR | unextmgr failure on machine ""$PRIMACH" | **原因**: 失敗したリソース拡張操作の後のクリーンアップに失敗しました。<br><br>**処置**: 残りのリソースを手動でクリーンアップし、隣接するログメッセージで詳細を確認してください。 |
| 124128 | ERROR | unextmgr failure on machine ""$PRIMACH"" for "$PRITAG." | **原因**: 失敗したリソース拡張操作の後のクリーンアップに失敗しました。<br><br>**処置**: 残りのリソースを手動でクリーンアップし、隣接するログメッセージで詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124129 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 追加のログメッセージを探して、詳細を確認してください。 |
| 124130 | ERROR | $ERRMSG Required template machine name is null | **原因**: 拡張操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124131 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: 拡張操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124132 | ERROR | $ERRMSG Required target mount point is null | **原因**: 拡張操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124133 | ERROR | $ERRMSG Unable to access template resource "$TemplateTagName" | **原因**: 拡張中のタグがテンプレートシステムに存在しません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124134 | ERROR | $ERRMSG Detected conflict in expected tag name "$TargetTagName" on target | **原因**: 拡張するリソースと同じタグのリ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | machine. | ソースがすでにターゲットシステムにあります。<br><br>**処置**: 競合しているリソースのどちらか一方を別のタグで再作成してください。 |
| 124135 | ERROR | $ERRMSG Resource "$TemplateTagName" does not have required device resource dependency or unable to access this resource on template machine. | **原因**: リソースまたは基本ディスクリソースがテンプレートシステムで見つかりません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 124136 | ERROR | $ERRMSG Resource "$TemplateTagName" must have one and only one device resource dependency | **原因**: リソースは、テンプレートシステムの階層内に複数の基本デバイスを持っています。<br><br>**処置**: 拡張する前に階層が正しいことを確認し、ファイルシステムリソースが単一のディスクリソースにのみ依存していることを確認してください。 |
| 124137 | ERROR | $ERRMSG Can not access extend for scsi/$DeviceResType resources on machine "$TargetSysName" | **原因**: 所定のストレージタイプのサポートに必要なファイルがターゲットシステムにありません。<br><br>**処置**: 必要なキットがターゲットシステムにインストールされ、ライセンスされていることを確認してください。 |
| 124138 | ERROR | $ERRMSG Unable to access target device resource "$DeviceTagName" on machine "$TargetSysName&quot | **原因**: 必要な基本ディスクリソースがターゲットシステムに存在していません。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、ターゲットシステムが拡張対象リソースをホストするために正常に設定されていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124141 | ERROR | $ERRMSG Unable to find mount point "$TemplateLKId" mode on template machine | **原因**: テンプレートシステムのマウントポイント詳細を特定できません。<br><br>**処置**: リソースが in service で、テンプレートシステムでアクセス可能であることを確認してから拡張してください。 |
| 124142 | ERROR | $ERRMSG Unable to create or access mount point "$TargetLKId" on target machine | **原因**: ターゲットシステムでマウントポイントを作成できませんでした。<br><br>**処置**: マウントポイントの親ディレクトリが存在し、ターゲットシステムでアクセス可能であることを確認してください。 |
| 124143 | ERROR | $ERRMSG Two or more conflicting entries found in /etc/fstab on "$TargetSysName&quot | **原因**: デバイスまたはマウントポイントはターゲットシステムで複数回マウントされているようです。<br><br>**処置**: マウントポイントがターゲットシステムにマウントされていないことを確認してから拡張してください。 |
| 124144 | ERROR | $ERRMSG Failed to create resource instance on "$TargetSysName" | **原因**: ターゲットシステムでのリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。ターゲットサーバでログを必ず確認してください。 |
| 124145 | ERROR | $ERRMSG Failed to set resource instance state for "$TargetTagName" on \"$TargetSysName" | **原因**: ターゲットシステムでソースの状態を OSU に変更できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124146 | ERROR | must specify machine name containing primary hierarchy | **原因**: filesyshier 操作で無効な引数が指定されました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124147 | ERROR | must specify primary mount point | **原因**: filesyshier 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124149 | ERROR | create file system hierarchy failure | **原因**: リソースインスタンスを検索するプロセスが失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124150 | ERROR | create file system hierarchy failure | **原因**: システムが /etc/mtab ファイルの読み取りに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124152 | ERROR | create file system hierarchy failure | **原因**: 基本ディスクリソースを見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124153 | ERROR | create file system hierarchy failure | **原因**: ファイルシステムリソースの作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124154 | ERROR | create file system hierarchy failure | **原因**: リソースの情報フィールドを更新できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124155 | ERROR | create file system hierarchy failure | **原因**: スイッチバック戦略をリソースで設定できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124157 | ERROR | create file system hierarchy failure (conflicting entries in /etc/fstab\) | **原因**: マウントポイントを /etc/fstab ファイルから削除できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124160 | ERROR | Unknown error in script filesysins, err=$err | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124161 | ERROR | create filesys instance – existid – failure | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124163 | ERROR | create filesys instance – ins_list – failure | **原因**: 既存のリソースの確認に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124164 | ERROR | create filesys instance – newtag – failure | **原因**: システムはリソースの推奨タグの |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 生成に失敗しました。<br><br>**処置**: 通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124168 | ERROR | create filesys instance - ins_create - failure | **原因**: ファイルシステムリソースを作成できませんした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124169 | ERROR | filesys instance - ins_setstate - failure | **原因**:新しいファイルシステムリソースの状態を初期化できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124173 | ERROR | create filesys instance - dep_create - failure | **原因**: 基本ディスクに対するリソースの依存関係を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124174 | ERROR | machine not specified | **原因**: rmenu_mp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124175 | ERROR | mount point not specified | **原因**: rmenu_mp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポート |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | に問い合わせてください。 |
| 124177 | ERROR | unexpected multiple matches found | **原因**: 1 つまたは複数のシステムが、複数回使用されたファイルシステムまたはマウントポイントを示しています。<br><br>**処置**: ファイルシステムデバイスとマウントポイントを確認し、ファイルシステムが 1 回だけマウントされていることを確認してください。追加のログメッセージを探して、詳細を確認してください。 |
| 124178 | ERROR | machine name not specified | **原因**: rmenump 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124180 | ERROR | must specify filesystem type | **原因**: validfstype 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124181 | ERROR | mount point not specified | **原因**: validmp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポートに問い合わせてください。 |
| 124182 | ERROR | The mount point $MP is not an absolute path | **原因**: 絶対パスではない（「/」で始まらない）マウントポイントが指定されまし |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | た。<br><br>**処置**:「/」で始まる絶対パスでマウントポイントを指定してください。 |
| 124183 | ERROR | $MP is already mounted on $MACH | **原因**: 要求されたマウントポイントはシステムですでに使用中です。<br><br>**処置**: 使用中ではないマウントポイントを指定するか、操作を再試行する前にアンマウントしてください。 |
| 124184 | ERROR | The mount point $MP is already protected by LifeKeeper on $MACH | **原因**: システムは、指定したマウントポイントをすでに保護しています。<br><br>**処置**: まだ保護されていない別のマウントポイントを選択してください。 |
| 124185 | ERROR | The mount point $MP is not a directory on $MACH | **原因**: マウントポイントが通常のファイルなどの非ディレクトリを参照しています。<br><br>**処置**: ディレクトリを参照するマウントポイントを選択してください。 |
| 124186 | ERROR | The mount point directory $MP is not empty on $MACH | **原因**: 指定したマウントポイントは空でないディレクトリを参照します。<br><br>**処置**: 空のマウントポイントを選択するか、指定したディレクトリの内容を削除してから操作を再試行してください。 |
| 124187 | ERROR | server name not specified | **原因**: valuepmp 操作で無効な引数が指定されました。<br><br>**処置**: スクリプトが正常に読み出されたことを確認してください。通常の操作でこのエラーが発生した場合は、サポート |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | に問い合わせてください。 |
| 124188 | ERROR | There are no mount points on server $MACH | **原因**: サーバにファイルシステムリソースのマウントポイントがありません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124194 | WARN | Please correct conflicting "/etc/fstab" entries on server $UNAME for: $FSDEV, $FSNAME | **原因**: ファイルシステムリソースを削除したら、/etc/fstab の一部のエントリを手動でクリーンアップする必要があります。<br><br>**処置**: /etc/fstab ファイルを手動でクリーンアップします。 |
| 124195 | ERROR | getchildinfo found no $OKAPP child for $PTAG | **原因**: システムは子リソースを階層で見つけられませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、階層が正しいことを確認してから、操作を再試行してください。 |
| 124196 | ERROR | enablequotas - quotacheck may have failed for $FS_NAME | **原因**: 割り当て操作に失敗しました。<br><br>**処置**: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124198 | ERROR | enablequotas - quotaon failed to turn on quotas for $FS_NAME, reason | **原因**: 割り当て操作に失敗しました。<br><br>**処置**: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124200 | ERROR | The device node $dev was not found or did not appear in the udev create time limit of $delay seconds | **原因**: デバイスノード（/dev/...）が |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | undev によって作成されませんでした。これは、ストレージまたはストレージへのサーバ接続の問題を示している可能性があります。<br><br>**処置**: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124201 | WARN | Device $device not found. Will retry wait to see if it appears. | **原因**: これは、udev がストレージのデバイスノードエントリを作成している間、通常の条件下で発生する可能性もあります。このメッセージは繰り返し表示されないはずです。<br><br>**処置**: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124202 | ERROR | Command "$commandwithargs" failed. Retrying …. | **原因**: 指定されたコマンドは失敗しましたが、一時的なものである可能性もあります。この障害は、通常の操作で発生しますが、継続はしないはずです。<br><br>**処置**: このメッセージが何度も表示される場合は、隣接するログメッセージで詳細を確認してください。 |
| 124204 | WARN | cannot make file system $FSNAME mount point | **原因**: マウントポイントディレクトリを作成できませんでした。<br><br>**処置**: マウントポイントを作成できることを確認してください。これは、ファイルシステム権限やマウントオプションなどによる可能性があります。 |
| 124207 | ERROR | "fsck"ing file system $FSNAME failed, trying alternative superblock | **原因**: このメッセージは、通常のファイルシステムチェックに失敗したことを示します。このメッセージは、ext2 ファイ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ルシステムまたは代替スーパーブロックロケーションが使用されているファイルシステムの場合は問題ないと思われます。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124209 | ERROR | "fsck"ing file system $FSNAME with alternative superblock failed | **原因**: これは、代替スーパーブロックロケーションで ext2 ファイルシステム (または代替スーパーブロックロケーションが使用されている他のファイルシステム) の確認が失敗したことを示します。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、実行方法の手順を確かめてください。 |
| 124210 | WARN | POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME) | **原因**: ミラーソースと同期していないファイルシステムが in service になったか、フェイルオーバされました。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、ファイルシステムを安全に in service にする方法を製品ドキュメントで調べてください。 |
| 124211 | ERROR | Reason for fsck failure ($retval): $ret | **原因**: このログメッセージは一連のメッセージの一部で、fsck プロセスからの実際の終了コードを提供します。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、実行方法の手順を確かめてください。 |
| 124212 | ERROR | "fsck" of file system $FSNAME failed | **原因**: ファイルシステムの確認に失敗しました。これは通常、ファイルシステムが破損したことによるものです。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細を確認してください。製品ドキュメントを参照し、可能性のあるファイルシステムの破損を処理する方法を調べてください。 |
| 124213 | WARN | POSSIBLE FILESYSTEM CORRUPTION ON $FSNAME ($FPNAME) | **原因**: システムまたはユーザが、破損している可能性のあるファイルシステムを in service にしようとしました。これは、ミラーサイトと同期していないときにファイルシステムが切り替えられたか、フェイルオーバされた場合に起こることがあります。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、リソースを安全に in service にする方法を製品ドキュメントで調べてください。 |
| 124214 | ERROR | Reason for fsck failure ($retval) | **原因**: このメッセージは、ファイルシステムチェックの障害に関する前のログメッセージに続いて表示されるもので、fsck プロセスのプロセス終了コードを提供します。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124218 | ERROR | File system $FSNAME was found to be already | **原因**: このメッセージは、一連のメッセージの一部です。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124219 | ERROR | mounted after initial mount attempt failed. | **原因**: このメッセージは、一連のメッセージの一部です。通常の状態では表示されないはずですが、リソースが in service の場合は致命的なエラーの可能性もあります。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: lifekeeper ログおよび /var/log/messages の隣接するログメッセージで詳細を確認してください。 |
| 124220 | ERROR | File system $FSNAME failed to mount. | **原因**: ファイルシステムをマウントできませんでした。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124221 | WARN | Protected Filesystem $ID is full | **原因**: ファイルシステムの空き容量がありません。<br><br>**処置**: 使用していないデータをファイルシステムから削除するか、より大きなファイルシステムに移行してください。 |
| 124222 | WARN | Dependent Applications may be affected <> | **原因**: これは、あるリソースに対する操作が、リソースの階層に基づいて別のリソースにも実行される可能性が高いことを示します。<br><br>**処置**: 示されているリソースが影響を受けても許容できることを確認してから操作を続行してください。 |
| 124223 | ERROR | Put "$t" Out-Of-Service Failed By Signal | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124227 | ERROR | Put "$i" Out-Of-Service Failed | **原因**: 操作に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 124230 | ERROR | Put "$t" In-Service Failed By Signal | **原因**: このメッセージは、通常の状態では表示されません。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124231 | ERROR | Put "$t" In-Service Failed | **原因**: 操作に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 124234 | ERROR | Put "$t" In-Service Failed | **原因**: 操作に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |

# 8.1.6. Gen/App キットメッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 126105 | ERROR | script not specified - $PTH is a directory | **原因**: 指定したスクリプトパスはディレクトリです。<br><br>**処置**: スクリプトのパスを修正してください。 |
| 126110 | ERROR | script $PTH does not exist | **原因**: 指定したスクリプトパスは存在しません。<br><br>**処置**: スクリプトのパスを修正してください。 |
| 126115 | ERROR | script $PTH is a zero length file | **原因**: 指定したスクリプトは空のファイルです。<br><br>**処置**: スクリプトのファイルパスを修正し、スクリプトの内容を確認してください。 |
| 126117 | ERROR | script $PTH is not executable | **原因**: 指定したスクリプトは実行可能ではありません。<br><br>**処置**: スクリプトのファイルパスを修正し、スクリプトファイルの内容を確認したら、正常な実行権限があることを確認してください。 |
| 126125 | ERROR | required template machine name is null | **原因**: 入力テンプレートマシン名が null です。<br><br>**処置**: 入力テンプレートマシン名を修正してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 126130 | ERROR | required template resource tag name is null | **原因**: 入力テンプレートリソース {tag} が null です。<br><br>**処置**: 入力テンプレートリソースタグ名を修正してください。 |
| 126135 | ERROR | Unable to generate a new tag | **原因**:拡張の際に、「newtag」スクリプトを使用してターゲットノードでテンプレートタグ名と同じ新規タグを生成できませんでした。タグ名はすでに存在しています。<br><br>**処置**: 同じノードで重複するタグ名を避け、ログで詳細を確認してください。 |
| 126140 | ERROR | Unable to generate a new tag | **原因**: 拡張の際に、「newtag」スクリプトを使用してターゲットノードで入力ターゲットタグ名として新規タグを生成できませんでした。タグ名はすでに存在しています。<br><br>**処置**: 同じノードで重複するタグ名を避け、ログで詳細を確認してください。 |
| 126150 | ERROR | unable to remote copy template \"$_lscript\" script file | **原因**: テンプレートスクリプトファイルのリモートコピーに失敗しました。原因としては、テンプレートノードにテンプレートスクリプトファイルがない、またはスクリプトファイルを使用できない、あるいは「lcdrcp」プロセスで何らかのトランザクション障害が発生したことなどが考えられます。<br><br>**処置**: テンプレートスクリプトの有無/使用可能性を確認し、テンプレートノードへの接続を確かめてください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 126155 | ERROR | failed to create resource instance on \"$TargetSysName\ | 原因:「ins_create」によるリソースインスタンスの作成に失敗しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126160 | ERROR | failed to set resource instance state for \"$TargetTagName\" on \"$TargetSysName\ | 原因: GenApp リソース拡張の際に、「ins_setstate」によるリソースインスタンスの状態の設定に失敗しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126170 | ERROR | getlocks failure | 原因:リソース階層の作成時に管理ロックの取得に失敗しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126175 | ERROR | instance create failed | 原因:「appins」による GenApp インスタンスの作成に失敗しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126180 | ERROR | unable to set state to OSU | 原因: GenApp リソース作成の際に、「ins_setstate」によるリソースインスタンスの状態の設定に失敗しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126190 | ERROR | resource restore has failed | 原因: GenApp リソースのリストアに失 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126200 | ERROR | create application hierarchy rlslocks failure | **原因**: GenApp リソースの作成後にロックの解除に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126210 | ERROR | copy $ltype script $lscript failure | **原因**: リソース作成時に、ユーザ提供のスクリプトを適切な GenApp ディレクトリにコピーできませんでした。<br><br>**処置**: ユーザ提供のスクリプトの有無/使用可能性に加え、GenApp ディレクトリを確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126215 | ERROR | no $ltype script specified | **原因**: GenApp リソース作成時にユーザ定義スクリプトが見つかりません。<br><br>**処置**: 入力アクションスクリプトを確認し、リソース作成を再実行してください。 |
| 126220 | ERROR | no machine name specified | **原因**:GenApp リソース作成時に指定のマシン名が見つかりません。マシン名の入力内容が見つからないため、指定したユーザスクリプトのコピーに失敗しました。<br><br>**処置**: マシン名の入力内容を確認し、リソース作成を再実行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 126225 | ERROR | no resource tag specified | **原因**:リソース作成時に指定のタグ名が見つかりません。<br><br>**処置**: ソースタグ名の入力内容を確認し、リソース作成を再実行してください。 |
| 126230 | ERROR | $ERRMSG Script was terminated for unknown reason | **原因**: 不明な理由で GenApp リソースの拡張に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126235 | ERROR | $ERRMSG Required template machine name is null | **原因**: GenApp リソース拡張時にテンプレートマシン名の入力が見つかりません。<br><br>**処置**: テンプレートマシン名の入力内容を確認し、リソース拡張を再実行してください。 |
| 126240 | ERROR | $ERRMSG Required template resource tag name is null | **原因**: GenApp リソース拡張時にテンプレートリソースタグ名の入力が見つかりません。<br><br>**処置**: テンプレートリソースタグ名の入力内容を確認し、リソース拡張を再実行してください。 |
| 126245 | ERROR | $ERRMSG Can not access extend for $AppType/$ResType resources on machine \"$TargetSysName\ | **原因**: ターゲットノードでの GenApp リソース拡張の際に、「extend」スクリプトを見つけられませんでした。<br><br>**処置**:「extend」スクリプトの有無/使用可能性を確認し、GenApp リソース拡張を再実行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 126250 | ERROR | create application failure - ins_list failed | **原因**: GenApp リソース作成で「ins_list」を呼び出している際に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126255 | ERROR | create application failure - unable to generate a new tag | **原因**: GenApp リソース作成で新規タグの生成に失敗しました。<br><br>**処置**: 同じノードで重複するタグ名の使用を避けてください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126270 | ERROR | create application failure - ins_create failed | **原因**:「ins_create」による GenApp インスタンスの作成に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126275 | ERROR | create application failure - copy_actions failed | **原因**:「copy_actions」によるユーザ指定テンプレートスクリプトファイルのコピーに失敗しました。<br><br>**処置**: テンプレートスクリプトの有無/使用可能性を確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126290 | ERROR | Unable to obtain tag for resource with id $ID | **原因**:リカバリ時に入力 ID による GenApp リソースタグ名のフェッチに失敗しました。<br><br>**処置**: 入力 ID が正しいこと、および LCD の GenApp リソースの有無/使用可能性を確認してください。また、関連するエラーをログで確認し、レポートされ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ている問題を解決してください。 |
| 126300 | ERROR | generic application recover script for $TAG was not found or was not executable | **原因**: リカバリ時に GenApp リソースのユーザ定義スクリプトの特定に失敗しました。<br><br>**処置**: ユーザ定義スクリプトの有無/使用可能性を確認し、GenApp リカバリプロセスを再実行してください。 |
| 126310 | ERROR | -t flag not specified | **原因**: GenApp リソースリストア時にリソースタグ名の入力が見つかりません。<br><br>**処置**: リソースタグ名の入力内容を確認し、リソースリストアを再実行してください。 |
| 126315 | ERROR | -i flag not specified | **原因**: GenApp リソースリストア時にリソース内部 ID の入力が見つかりません。<br><br>**処置**: リソース内部 ID の入力内容を確認し、リソースリストアを再実行してください。 |
| 126327 | ERROR | END timeout restore of \"$TAG\" (forcibly terminating) | |
| 126335 | ERROR | restore script \"$LCDAS/$APP_RESTOREDIR/$TAG\" was not found or is not executable | **原因**: リストア時に GenApp リソースのユーザ定義スクリプトの特定に失敗しました。<br><br>**処置**: ユーザ定義スクリプトの有無/使用可能性を確認し、GenApp リストアプロセスを再実行してください。 |
| 126340 | ERROR | -t flag not specified | **原因**: GenApp リソース削除時にリソースタグ名の入力が見つかりません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: リソースタグ名の入力内容を確認し、リソース削除を再実行してください。 |
| 126345 | ERROR | -i flag not specified | **原因**: GenApp リソース削除時にリソース内部 ID の入力が見つかりません。<br><br>**処置**: リソース内部 ID の入力内容を確認し、リソース削除を再実行してください。 |
| 126357 | ERROR | END timeout remove of \"$TAG\" (forcibly terminating) | |
| 126365 | ERROR | remove script \"$LCDAS/$APP_REMOVEDIR/$TAG\" was not found or was not executable | **原因**: 削除時に GenApp リソースのユーザ定義スクリプトの特定に失敗しました。<br><br>**処置**: ユーザ定義スクリプトの有無/使用可能性を確認し、GenApp 削除プロセスを再実行してください。 |
| 126375 | ERROR | Script has hung checking \"$tag\". Forcibly terminating. | **原因**: 待機時間がユーザ定義のタイムアウトを超えたので、タグ名 {tag} の GenApp リソースの「quickCheck」スクリプトが強制的に終了しました。<br><br>**処置**: GenApp リソースのパフォーマンスを確認し、quickChecking を再起動してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126380 | ERROR | Usage error: no tag specified | **原因**: GenApp リソース quickCheck 時にリソースタグ名の入力が見つかりません。<br><br>**処置**: リソースタグ名の入力内容を確認し、リソース quickCheck を再試行して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください。 |
| 126385 | ERROR | Internal error: ins_list failed on $tag. | 原因: quickCheck プロセスで、「ins_list」による入力タグ名別の GenApp リソース情報のフェッチに失敗しました。<br><br>処置: 入力タグ名を修正し、quickCheck プロセスを再実行してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126390 | FATAL | Failed to fork process to execute $userscript: $! | 原因: GenApp リソースの「quickCheck」プロセスで、ユーザ定義の「quickCheck」スクリプトを実行するためにプロセスを分岐させられませんでした。<br><br>処置: ユーザ定義の「quickCheck」スクリプトの有無/使用可能性を確認し、「quickCheck」プロセスを再実行してください。 |
| 126391 | ERROR | quickCheck has failed for \"$tag\". Starting recovery. | 原因: タグ名 {tag} の GenApp リソースが失敗することが、ユーザ定義の健全性監視スクリプトにより判定されました。「quickCheck」およびリカバリプロセスが開始されます。<br><br>処置: ローカルリカバリが完了したら、GenApp リソースのパフォーマンスを確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 126392 | WARN | ${convtag}_TIMEOUT: This parameter is old. This parameter will not be supported soon. | |
| 126400 | ERROR | -t flag not specified | 原因:GenApp リソース削除プロセスで |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | リソースタグ名の入力が見つかりません。<br><br>**処置**: リソースタグ名の入力内容を確認し、リソース削除プロセスを再実行してください。 |
| 126405 | ERROR | -i flag not specified | <br>**原因**: GenApp リソース削除プロセスでリソース内部 ID の入力が見つかりません。<br><br>**処置**: リソース内部 ID の入力内容を確認し、リソース削除プロセスを再実行してください。 |
| 126478 | ERROR | Failed to create tag \'$new_leaf\'. | <br>**原因**:「creapphier」ユーティリティは、指定されたタグ {new_leaf} を作成できませんでした。<br><br>**処置**: /var/log/lifekeeper.log で「creapphier」からの追加メッセージを確認してください。 |
| 126479 | ERROR | Failed to extend tag \'$new_leaf\'. | <br>**処置**: /var/log/lifekeeper.log で拡張マネージャーからの追加エラーを確認してください。 |
| 126481 | ERROR | Failed to create dependency on \'$sys\' for \'$new_leaf\' to \'$hier{$leaf}{$sys}{'Tag'}\'. | <br>**処置**: /var/log/lifekeeper.log で「dep_create」関数のエラーを確認してください。 |
| 126484 | ERROR | Tag \'$root_tag\' is not in-service. | <br>**原因**: 指定されたタグ{root_tag} は、クラスター内のどのノードでも In Service になっていません。<br><br>**処置**: 指定されたタグ{root_tag} をクラスター内の任意のノードで In Service にし、「create_terminal_leaf」を再実行し |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | てください。 |
| 126485 | ERROR | Tag \'$root_tag_1\' was not found, select the root tag for a hierarchy to add a terminal leaf resource. | **原因**:「create_terminal_leaf」に渡された最初のタグ {root_tag_1} が、ユーティリティが実行されたシステムで見つかりませんでした。<br><br>**処置**: リソースがクラスタ内のノードで In-service であり、すべてのノードへリソースが完全に拡張されていることを確認してください。 |
| 126486 | ERROR | Unable to create leaf tag from \'$tag\'. | **原因**: 一意の端末リーフタグを作成できませんでした。100回試行した後、一意の端末リーフタグを作成できませんでした。<br><br>**処置**: 複数のリーフタグと、問題を暗示している可能性のある /var/log/lifekeeper.log のエラーを確認してください。 |
| 126487 | ERROR | Tag \'$root_tag_2\' was not found, select the root tag for a hierarchy to add a terminal leaf resource. | **原因**:「create_terminal_leaf」に渡された2番目のタグ {root_tag_2} が、ユーティリティが実行されたシステムで見つかりませんでした。<br><br>**処置**: リソースがクラスタ内のノードで In-service であり、すべてのノードへリソースが完全に拡張されていることを確認してください。 |
| 126488 | ERROR | Tag \'$root_tag_1' is not extended to 3 or more systems. | **原因**: 指定されたタグ {root_tag_1} は、3つ以上のノードに拡張されていません。<br><br>**処置**: 指定したタグ {root_tag_1} を最低でも3つのノードに拡張し、「create_terminal_leaf」を再度実行して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください。 |
| 126489 | ERROR | $cmd does not support SDRS resources. | **原因**: マルチサイト構成が検知されました。<br><br>**処置**: ありません |
| 126492 | ERROR | Remove resource $tag failed. | |
| 126494 | ERROR | Delete dependency failed on \'$sys\' for \'$tag\' to \'$parent\'. | |
| 126495 | ERROR | New tag \'$new_tag\' was modified during create, expected \'$new_leaf\'. | |
| 126496 | ERROR | Tag \'$root_tag_1\' is not in-service. | |
| 126497 | ERROR | Tag \'$root_tag_2\' is not in-service. | |
| 126498 | ERROR | Tag \'$root_tag_1\' and \'$root_tag_2' are not extended to same systems. | |

# 8.1.7. IP Kit メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 123006 | FATAL | Unknown version %s of IP address | 原因: IP アドレスが IPv4 でも IPv6 でも有効ではないようです。<br><br>処置: 有効な IP アドレスを指定してください。 |
| 123008 | ERROR | No pinglist found for %s. | 原因: この IP アドレスの pinglist を開こうとして問題が発生しました。<br><br>処置: この IP アドレスの pinglist を指定していることを確認してください。 |
| 123009 | ERROR | List ping test failed for virtual IP %s | 原因: ping リストのどのアドレスからも応答を受信しませんでした。<br><br>処置: このノードのネットワーク接続および ping リストの IP が存在するシステムを確認してください。 |
| 123013 | ERROR | Link check failed for virtual IP %s on interface %s. | 原因: 要求したインターフェースが「NO-CARRIER」となっています。これは、物理レイヤー接続にリンクが存在しないことを示します。<br><br>処置: インターフェースの物理接続を確認し、物理レイヤーリンクを有効にしてください。 |
| 123015 | ERROR | Link check failed for virtual IP %s on interface %s. | 原因: 要求したインターフェースは bonding インターフェースで、スレーブのいずれか 1 つが「NO-CARRIER」です。これは、物理レイヤー接続にリンク |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | が存在しないことを示します。<br><br>**処置**: スレーブインターフェースの物理接続を確認し、物理レイヤーリンクを有効にしてください。 |
| 123024 | ERROR | IP address seems to still exist somewhere else. | **原因**: IP アドレスはネットワークの別の場所で使用中のようです。<br><br>**処置**: 別の IP アドレスを選択して使用するか、この IP アドレスを特定して、現在の使用を無効にしてください。 |
| 123037 | ERROR | must specify machine name containing primary hierarchy | **原因**: creIPhier に指定した引数が不十分です。<br><br>**処置**: 必要な引数をすべて creIPhier に指定してください。 |
| 123038 | ERROR | must specify IP resource name | **原因**: creIPhier に渡された引数が不十分です。<br><br>**処置**: 必要な引数をすべて creIPhier に指定してください。 |
| 123039 | ERROR | must specify primary IP Resource tag | **原因**:「creIPhier」コマンドにプライマリ IP リソースタグの引数が指定されていません。<br><br>**処置**: 必要な引数をすべて指定してください。 |
| 123042 | ERROR | An unknown error has occurred in utility validmask on machine %s. | **原因**:「validmask」ユーティリティの実行中に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 123045 | ERROR | An unknown error has occurred in utility getlocks. | 原因:「getlocks」ユーティリティの実行中に予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細を確認してください。 |
| 123053 | ERROR | Cannot resolve hostname %s | 原因: IP アドレスに対してホスト名が指定されましたが、システムはホスト名を IP アドレスに解決できませんでした。<br><br>処置: ホスト名が正確かどうかを確認してから、名前の解決（DIS または /etc/hosts）が正常に機能しており、ホスト名の IP を返すことを確認してください。 |
| 123055 | ERROR | An unknown error has occurred in utility %s on machine %s. | 原因: IP リソースの作成中に障害が発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123056 | ERROR | create ip hierarchy failure: perform_action failed | 原因:作成中に IP アドレスをリストアしようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細を確認してください。 |
| 123059 | ERROR | Resource already exists on machine %s | 原因: すでに存在する IP アドレスを作成しようとしました。<br><br>処置: 既存のリソースを再利用するか、すでに存在している IP アドレスを手動で削除するか、別の IP アドレスを使用してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 123060 | ERROR | ins_create failed on machine %s | **原因**: IP リソースの作成中に予期しない障害が発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認してください。 |
| 123064 | ERROR | An unknown error has occurred in utility %s on machine %s. | **原因**: IP リソースの依存関係を作成中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123066 | ERROR | An error occurred during creation of LifeKeeper application=comm on %s. | **原因**:「app_create」の呼び出し中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123068 | ERROR | An error occurred during creation of LifeKeeper resource type=ip on %s. | **原因**:「typ_create」の呼び出し中に障害が発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 123089 | ERROR | Link check failed for virtual IP %s on interface %s. | |
| 123091 | ERROR | the link for interface %s is down | **原因**: 要求したインターフェースが「NO-CARRIER」となっています。これは、物理レイヤー接続にリンクが存在しないことを示します。<br><br>**処置**: インターフェースの物理接続を確認し、物理レイヤーリンクを有効にしてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 123093 | ERROR | the ping list check failed | **原因**: ping リストのどのアドレスからも応答を受信しませんでした。<br><br>**処置**: このノードのネットワーク接続および ping リストの IP が存在するシステムを確認してください。 |
| 123095 | ERROR | broadcast ping failed | **原因**: ブロードキャスト ping から応答を受信しませんでした。<br><br>**処置**: サブネット上の 1 つ以上のホストがブロードキャスト ping に応答することを確認してください。仮想 IP が正しいネットワークインターフェースにあることを確認してください。ブロードキャスト ping ではなく、pinglist の使用を検討してください。 |
| 123096 | ERROR | $msg | **原因**: 仮想 IP の実行可能性の判定に使用するブロードキャスト ping に失敗しました。<br><br>**処置**: このリソースの ping リストがプロパティパネルで正常に設定されていることを確認するか、/etc/default/LifeKeeper 設定ファイルに NOBCASTPING=1 を追加してブロードキャスト ping のチェックを確実に無効にしてください。 |
| 123097 | ERROR | exec_list_ping(): broadcast ping failed. | **原因**: 仮想 IP の実行可能性の判定に使用するブロードキャスト ping に失敗しました。<br><br>**処置**: このリソースの ping リストがプロパティパネルで正常に設定されていることを確認するか、/etc/default/LifeKeeper 設定ファイルに NOBCASTPING=1 を追加してブロードキャスト ping のチェックを確実に無効 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | にしてください。 |
| 123299 | ERROR | Unable to open %s. Reason %s | |
| 123410 | ERROR | Usage error OSUquickCheck | |
| 123411 | ERROR | OSUquickCheck: both tag and id name not specified | |
| 123412 | ERROR | resource $Tag not found on local server | |
| 123414 | ERROR | The link for network interface $IPObj->{'device'} is down | |
| 123415 | ERROR | No pinglist found for $IPObj->{'ipaddr'} | |
| 123416 | ERROR | List ping test failed for virtual IP $IPObj->{'ipaddr'} | |

# 8.1.8. Oracle キットメッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122500 | ERROR | Usage: %s %s | **原因**: 作成操作で無効なパラメーターが指定されました。<br><br>**処置**: パラメーターが正しいことを確認し、操作を再実行してください。 |
| 122501 | ERROR | DB instance "%s" is already protected on "%s". | **原因**: {server} で LifeKeeper によってすでに保護されている Oracle データベースインスタンス {sid} を保護しようとしました。<br><br>**処置**: 別のデータベースインスタンス {sid} を LifeKeeper の保護用に選択する必要があります。 |
| 122502 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122503 | ERROR | Unable to locate the oratab file "%s" on "%s". | **原因**: {server} のデフォルトまたは代替ロケーションで oratab ファイルが見つかりませんでした。<br><br>**処置**: oratab ファイルが存在しており、Oracle ユーザの適切な権限を持っていることを確認してください。「create」操作を完了するには有効な oratab ファイルが必要です。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122504 | ERROR | Unable to determine Oracle user for "%s" on "%s". | 原因: Oracle Application Recovery Kit は Oracle データベースインストールバイナリの所有権を特定できませんでした。<br><br>処置: Oracle バイナリの所有者は {server} の有効な非 root ユーザである必要があります。Oracle データベース所有権の権限と所有権を修正し、操作を再試行してください。 |
| 122505 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | 原因: データベースインスタンス {sid} が実行されていません。またはデータベースへの接続が、指定された認証情報では使用できませんでした。<br><br>処置:「create」操作を完了するには、データベースインスタンス {sid} を {server} で起動し、正しい認証情報を指定する必要があります。 |
| 122506 | ERROR | Unable to determine Oracle dbspaces and logfiles for "%s" on "%s". | 原因: 必要なテーブルスペース、ログファイル、関連データベースファイルの場所を特定するクエリーが失敗しました。これは、内部データベースエラーが原因の可能性もあります。<br><br>処置: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122507 | ERROR | Unknown chunk type found for "%s" on "%s". | 原因: 指定したテーブルスペース、ログファイル、または必要なデータベースファイルが LifeKeeper でサポートされているファイルまたはキャラクタデバイスタイプではありません。<br><br>処置: 指定したファイル {database_file} |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | は既存のキャラクタデバイスまたはファイルを参照する必要があります。Oracle インストールドキュメントを参照し、指定したファイル {database_file} を、サポートされているファイルまたはキャラクタデバイスタイプとして再作成してください。 |
| 122508 | ERROR | DB Chunk "%s" for "%s" on "%s" does not reside on a shared file system. | **原因**: 指定したテーブルスペース、ログファイル、またはその他の必要なデータベースファイル {database_file} は、クラスタ内の他のシステムと共有されているファイルシステムに存在します。<br><br>**処置**: コミュニケーションパスが正常に作成されたことを確認するには、LifeKeeper UI または「lcdstatus (1M)」を使用してください。ストレージ保護に必要なアプリケーションリカバリキットがインストールされたことを確認するには、「rpm」を使用してください。ファイルが実際には共有ストレージにないことを確認し、ない場合は、共有ストレージデバイスに移動してください。 |
| 122510 | ERROR | File system create failed for "%s" on "%s". Reason: | **原因**: LifeKeeper は、指定したサーバ {server} でリソース {filesystem} を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから「create」操作を再試行する必要があります。 |
| 122511 | ERROR | %s | **原因**: メッセージに「filesyshier」コマンドの結果が含まれています。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポート |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | されているエラーを修正してください。 |
| 122513 | ERROR | Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed.  Reason: | 原因: LifeKeeper は、データベースリソース {tag} と必要な子リソース {childtag} との間に依存関係を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。問題が修正されている場合は、{tag} と {childtag} との間の依存関係を手動で作成できる可能性があります。 |
| 122514 | ERROR | Unable to "%s" on "%s" during resource create. | 原因: Oracle Application Recovery Kit は、「rlslocks」コマンドで管理ロックを解除できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122516 | ERROR | Raw device resource created failed for "%s" on "%s".  Reason: | 原因: LifeKeeper は、指定したサーバ {server} でリソース {raw device} を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから「create」操作を再試行する必要があります。 |
| 122519 | ERROR | In-service attempted failed for tag "%s" on "%s". | 原因: {server} の {tag} に対する「perform_action」コマンドがデータベース {sid} の開始に失敗しました。in service 操作に失敗しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正し |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | てから「create」操作を再試行する必要があります。 |
| 122521 | ERROR | Create of app "%s" on "%s" failed with return code of "%d". | **原因**: 内部アプリケーションタイプを作成する「app_create」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122522 | ERROR | Create of typ "%s" for app "%s" on "%s" failed with return code of "%d". | **原因**: 内部リソースタイプを作成する「typ_create」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122524 | ERROR | Setting "resstate" for resource "%s" on "%s" failed with return code of "%d". | **原因**: リソースの状態を {state} に設定する「ins_setstate」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122525 | ERROR | The values specified for the target and the template servers are the same: "%s". | **原因**:「extend」操作でターゲットサーバとテンプレートサーバで指定した値が同じでした。<br><br>**処置**: {target server} と {template server} に対して正しいパラメータを指 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 定する必要があります。{target server} は、{tag} が拡張されるサーバです。 |
| 122526 | ERROR | Unable to locate the oratab file in "/etc" or in "%s" on "%s". | **原因**: {server} のデフォルトまたは代替ロケーションで oratab ファイルが見つかりませんでした。<br><br>**処置**: oratab ファイルが存在しており、Oracle ユーザの適切な権限を持っていることを確認してください。「extend」操作を完了するには有効な oratab ファイルが必要です。 |
| 122527 | ERROR | Unable to retrieve the Oracle user on "%s". | **原因**:「canextend」操作または「extend」操作の間に {template server} から Oracle ユーザを取得できませんでした。<br><br>**処置**: Oracle バイナリの所有者は、{target server} と {template server} の有効なユーザである必要があります。Oracle データベース所有権の権限と所有権を修正し、操作を再試行してください。 |
| 122528 | ERROR | The Oracle user and/or group information for user "%s" does not exist on the server "%s". | **原因**: LifeKeeper は、サーバ {server} の Oracle ユーザ {user} に対して Oracle ユーザおよびグループ情報を検索できません。<br><br>**処置**: 指定した {server} に Oracle ユーザ {user} が存在していることを確認してください。ユーザ {user} が存在しない場合、クラスタ内のすべてのサーバで、同じ uid および gid を使用して作成する必要があります。 |
| 122529 | ERROR | The id for user "%s" is not the same on template server "%s" and target server "%s". | **原因**: ターゲットサーバ {target server} にある Oracle ユーザ {user} のユーザ ID |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | が、テンプレートサーバ {template server} にあるユーザ {user} の値と一致しません。<br><br>**処置**:Oracle ユーザ {user} のユーザ ID はクラスタ内のすべてのサーバで一致している必要があります。ユーザ ID の不一致をすべてのサーバで手動で修正し、「extend」操作を再試行する必要があります。 |
| 122530 | ERROR | The group id for user "%s" is not the same on template server "%s" and target server "%s". | **原因**: ターゲットサーバ {target server} にある Oracle ユーザ {user} のグループ ID が、テンプレートサーバ {template server} にあるユーザ {user} の値と一致しません。<br><br>**処置**: Oracle ユーザ {user} のグループ ID はクラスタ内のすべてのサーバで一致している必要があります。グループ ID の不一致をすべてのサーバで手動で修正し、「extend」操作を再試行する必要があります。 |
| 122532 | ERROR | No file system or raw devices found to extend for "%s" on "%s". | **原因**: サーバ {template server} で Oracle リソース {tag} の依存ファイルシステムまたは Raw デバイスリソースが見つかりません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122533 | WARN | A RAMDISK (%s) was detected in the ORACLE Database configuration for "%s" on "%s". LifeKeeper cannot protect RAMDISK. This RAMDISK resource will not be protected by LifeKeeper! ORACLE hierarchy creation will continue. | **原因**: 指定したテーブルスペース、ログファイル、またはその他のデータベースファイル {database_file} が ramdisk として検出されました。現在の LifeKeeper 製品では、このタイプのリソースに保護を適用できません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: ramdisk は保護されません。すべての Oracle データベース操作で必要なデータベースファイル {database_file} が使用可能になるように手動で確認する必要があります。 |
| 122534 | ERROR | Failed to initialize object instance for Oracle sid "%s" on "%s". | **原因**: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122537 | ERROR | Update of instance info field for "%s" on "%s" failed (%s). | **原因**: 内部リソース情報フィールドを更新する「ins_setinfo」コマンドの実行中にエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122538 | ERROR | Initial connect with query buffer to database "%s" on "%s" failed, testing output. | **原因**: データベースの状態を判別するために Oracle データベース {sid} に接続しようとしたところ、失敗しました。<br><br>**処置**: 指定した認証情報で接続が失敗しました。隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122542 | ERROR | The "%s [ %s ]" attempt of the database "%s" appears to have failed on "%s". | **原因**: データベースインスタンス {sid} に対して メソッド {action_method} を使 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 用した Oracle アクション {action} がサーバ {server} で失敗しました。<br><br>**処置**: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122543 | ERROR | All attempts to "%s" database "%s" on "%s" failed | **原因**: サーバ {server} の Oracle データベース {sid} に対するアクション {action} を実行するためのあらゆる取り組みが失敗しました。<br><br>**処置**: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122544 | ERROR | Update of "%s" sid "%s" on "%s" failed.  Reason: "%s" "%s" failed: "%s". | **原因**: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。このエラーは oratab ファイルを開こうとしている間に発生しました。<br><br>**処置**: システム起動時にデータベースが自動で起動されないようにするために、{sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122545 | ERROR | Unable to locate the oratab file in "/etc" or in "%s" on "%s". | **原因**: {server} のデフォルトまたは代替ロケーションで oratab ファイルが見つかりませんでした。<br><br>**処置**: oratab ファイルが存在しており、Oracle ユーザの適切な権限を持っていることを確認してください。「extend」操作を完了するには有効な oratab ファイルが必要です。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122546 | ERROR | Unable to open file "%s" on "%s" (%s). | 原因: エラー {error} のため、サーバ {server} では、指定したファイル {file} を開くことも、ファイルにアクセスすることもできませんでした。<br><br>処置: 指定したファイル {file} の有無や権限を確認してください。隣接するログメッセージおよび関連エラーで詳細を確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122547 | ERROR | (cleanUpPids):Forcefully killing hung pid(s):pid(s)="%s" | 原因: プロセス {pid} が正常終了するための要求に応答できませんでした。プロセス {pid} は強制終了します。<br><br>処置: コマンドラインで、プロセス {pid} が終了していることを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122548 | ERROR | Unable to locate the DB utility (%s/%s) on this host. | 原因: {path/utility} にある Oracle バイナリおよび必要なデータベースユーティリティ {utility} がこのサーバ {server} では見つかりませんでした。<br><br>処置: Oracle バイナリおよび必要なソフトウェアユーティリティがインストールされ、サーバ {server} で適切に設定されていることを確認してください。Oracle バイナリを各ノードにローカルインストールするか、クラスタ内の全ノードで使用可能な共有ストレージに配置する必要があります。 |
| 122549 | ERROR | Oracle internal error or non-standard Oracle configuration detected. Oracle User and/or Group set to "root". | 原因: 検出された Oracle データベースインストールの所有権が、ルートユーザまたはルートグループに解決されています。ルートによる Oracle インストールの所有権は標準的ではない設定です。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: Oracle バイナリの所有者は {server} の有効な非 root ユーザである必要があります。Oracle データベース所有権の権限と所有権を修正し、操作を再試行してください。 |
| 122550 | ERROR | Initial inspection of "%s" failed, verifying failure or success of received output. | **原因**: 前の Oracle クエリー {query} またはコマンド {cmd} が成功を返すことができませんでした。<br><br>**処置**: 隣接するログメッセージおよび関連エラーで詳細を確認してください。Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122551 | ERROR | Logon failed with "%s" for "%s" on "%s". Please check username/password and privileges. | **原因**: 認証情報 {credentials} を使用してサーバ {server} のデータベースインスタンス {sid} へのログオンに失敗しました。無効なユーザ {user} またはパスワードが指定されました。<br><br>**処置**: Oracle データベースユーザ {user} およびパスワード {password} が有効であることを確認してください。また、Oracle データベースユーザ {user} は、試行されるアクションに対して十分な権限を持っている必要があります。 |
| 122552 | ERROR | %s | **原因**: メッセージに「sqlplus」コマンドの結果が含まれています。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122553 | ERROR | Unable to open file "%s" on "%s" (%s). | **原因**: エラー {error} のため、サーバ {server} では、指定したファイル {file} を開くことも、ファイルにアクセスする |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | こともできませんでした。<br><br>**処置**: 指定したファイル {file} の有無や権限を確認してください。隣接するログメッセージおよび関連エラーで詳細を確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122554 | ERROR | The tag "%s" on "%s" is not an Oracle instance or it does not exist. | **原因**: サーバ {server} の指定されたタグ {tag} は、既存の有効な Oracle リソースインスタンスを参照していません。<br><br>**処置**: UI または「lcdstatus (1M)」を使用して、リソースタグ {tag} の有無を確認してください。コマンド「ora-display」を使用するには、リソースタグ {tag} が Oracle リソースインスタンスである必要があります。 |
| 122555 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: Oracle リソースインスタンスの認定ユーザ、パスワード、データベースロールを更新している際に、保護されている Oracle インスタンスの内部表現を作成しようとしたところ、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122557 | ERROR | Update of user and password failed for "%s" on "%s". | **原因**: リソースタグ {tag} のユーザおよびパスワードを更新する要求に失敗しました。指定した認証情報では、サーバ {server} に対する初期検証/接続ができませんでした。<br><br>**処置**: 目的の操作に対して正しい認証情 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 報 {user/password} が指定されていることを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122559 | ERROR | Update of user and password failed for "%s" on "%s". | 原因: サーバ {server} のリソースタグ {tag} のユーザおよびパスワード情報の更新に失敗しました。<br><br>処置: 目的の操作に対して正しい認証情報 {user/password} が指定されていることを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122562 | ERROR | Unable to find the Oracle executable "%s" on "%s". | 原因: 必要な Oracle 実行ファイル {exe} がこのサーバ {server} では見つかりませんでした。<br><br>処置: Oracle バイナリおよび必要なソフトウェアユーティリティがインストールされ、サーバ {server} で適切に設定されていることを確認してください。Oracle バイナリを各ノードにローカルインストールするか、クラスタ内の全ノードで使用可能な共有ストレージに配置する必要があります。 |
| 122566 | ERROR | Unable to find Oracle home for "%s" on "%s". | 原因: Oracle ホームディレクトリ {Oracle home} には、Oracle インスタンス {sid} の正常な動作に必要なファイルが含まれていないようです。<br><br>処置: Oracle ホームディレクトリ {Oracle home} に Oracle バイナリ、有効な spfile{sid}.ora、または init{sid}.ora |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ファイルが含まれていることをコマンドラインで確認してください。 |
| 122567 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。指定した内部 ID {id} が、予想した SID {sid} と一致しません。<br><br>**処置**: パラメータが正しいことを確認してください。隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122568 | ERROR | DB Processes are not running on "%s". | **原因**: Oracle インスタンスのプロセスチェックで、サーバ {server} で実行されているプロセスは見つかりませんでした。<br><br>**処置**: ローカルリカバリが有効になっている場合、Oracle インスタンスはローカルで再起動されます。隣接するログメッセージで詳細および関連メッセージを確認してください。 |
| 122572 | ERROR | Failed to create flag "%s" on "%s". | **原因**: スタンバイノードへのフェイルオーバを促す Oracle ローカルリカバリ処理を制御するためのフラグを作成しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122574 | ERROR | all attempts to shutdown the database %s failed on "%s". | **原因**: ローカルリカバリプロセス中に、おそらくデータベース接続数が上限に達したため、Oracle データベースのシャッ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | トダウンに失敗しました。<br><br>**処置**: Oracle ログで、許容接続数の上限に達したことによって起こった接続の障害の有無を確認してください。その障害が見つかった場合は、接続数を増やしてください。また、ローカルリカバリの実行による quickCheck の障害発生が、接続の失敗を引き起こさないように、LK_ORA_NICE を 1 に設定してください。 |
| 122597 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護されている Oracle インスタンスの内部表現を作成中、拡張前のチェックの際に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、pre-extend を再試行してください。 |
| 122598 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 作成中の Oracle インスタンスの内部表現を作成しているときに、Oracle ホームディレクトリの有効性を確認しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、「create」を再試行してください。 |
| 122599 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、テンプレートシステムで Oracle ユーザをルックアップしようとして、予期しないエラーが発生しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、extend を再試行してください。 |
| 122600 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、リソースプロパティを表示しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、リソースプロパティの表示を再試行してください。 |
| 122601 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、有効なデータベース認証を確認しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、コマンドを再試行してください。 |
| 122603 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、Oracle リソースインスタンスに対してヘルスチェックを実行しようとして、予期しないエラーが発生しました。<br><br>**処置**: 正しい引数が quickCheck コマンドに渡されていることを確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、restore を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122604 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護中の Oracle インスタンスの内部表現を作成しているときに、Oracle リソースインスタンスに対してローカルリカバリを実行しようとして、予期しないエラーが発生しました。<br><br>**処置**: 正しい引数が「recover」コマンドに渡されていることを確認し、隣接するログメッセージと関連メッセージで詳細を確認してください。レポートされているエラーを修正してから、recover を再試行してください。 |
| 122606 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | **原因**: データベースインスタンス {sid} が実行されていません。またはデータベースへの接続が、指定された認証情報では使用できません。<br><br>**処置**: 選択した操作を完了するには、データベースインスタンス {sid} を {server} で起動し、正しい認証情報を指定する必要があります。 |
| 122607 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | **原因**: データベースインスタンス {sid} が実行されていません。またはデータベースへの接続が、指定された認証情報では使用できません。<br><br>**処置**: 選択した操作を完了するには、データベースインスタンス {sid} を {server} で起動し、正しい認証情報を指定する必要があります。 |
| 122608 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**:「remove」操作で、Oracle リソースを Out of Service にするために必要なリソースオブジェクトインスタンスを作成できませんでした。<br><br>**処置**: 正しい引数が「remove」コマンドに渡されていることを確認し、隣接するロ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | グメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、restore を再試行してください。 |
| 122609 | ERROR | Failed to create object instance for Oracle on "%s". | 原因:「restore」操作で、Oracle リソースを In Service にするために必要なリソースオブジェクトインスタンスを作成できませんでした。<br><br>処置: 正しい引数が「restore」コマンドに渡されていることを確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから、「restore」を再試行してください。 |
| 122610 | ERROR | Unable to "%s" on "%s" during resource create. | 原因: Oracle Application Recovery Kit は、リソース作成中に「getlocks」コマンドで管理ロックを作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create を再試行してください。 |
| 122611 | ERROR | %s | 原因: 要求された親 Oracle リソースと子 File System リソースとの間の依存関係の作成に失敗しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create 操作を再試行してください。 |
| 122612 | ERROR | %s | 原因: 要求された親 Oracle リソースと子 Raw リソースとの間の依存関係の作成に失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create 操作を再試行してください。 |
| 122613 | ERROR | %s | **原因**: 要求された親 Oracle リソースと子 Raw リソースとの間の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create 操作を再試行してください。 |
| 122614 | ERROR | %s | **原因**: 要求された親 Oracle リソースと子 Listener リソースとの間の依存関係の作成に失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、create 操作を再試行してください。 |
| 122616 | ERROR | %s | **原因**: Oracle データベースの要求された起動またはシャットダウンに失敗しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから、「restore」または「remove」操作を再試行してください。 |
| 122618 | ERROR | Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed.  Reason: | **原因**: LifeKeeper は、データベースリソース {tag} と必要な子リソース {childtag} との間に依存関係を作成できませんでした。<br><br>**処置**: 隣接するログメッセージで詳細お |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | よび関連メッセージを確認してください。問題が修正されている場合は、{tag} と {childtag} との間の依存関係を手動で作成できる可能性があります。 |
| 122619 | ERROR | Dependency creation between Oracle database "%s (%s)" and the dependent resource "%s" on "%s" failed.  Reason: | 原因: LifeKeeper は、データベースリソース {tag} と必要な子リソース {childtag} との間に依存関係を作成できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。問題が修正されている場合は、{tag} と {childtag} との間の依存関係を手動で作成できる可能性があります。 |
| 122625 | ERROR | Unable to find the Oracle executable "%s" on "%s". | 原因: quickCheck プロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>処置: Oracle 設定を確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |
| 122626 | ERROR | Unable to find the Oracle executable "%s" on "%s". | 原因: 削除プロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>処置: Oracle 設定を確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |
| 122627 | ERROR | Unable to find the Oracle executable "%s" on "%s". | 原因: リストアプロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>処置: Oracle 設定を確認し、隣接するロ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | グメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |
| 122628 | ERROR | Unable to find the Oracle executable "%s" on "%s". | **原因**: 復旧プロセスが、Oracle の実行可能ファイル「sqlplus」を見つけることができませんでした。<br><br>**処置**: Oracle 設定を確認し、隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされている問題を修正してください。 |
| 122632 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: 削除の際に、削除プロセスに渡されるリソースインスタンス {sid} が、{sid} の内部リソースインスタンス情報と一致しません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122633 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: リストアの際に、restore に渡されるリソースインスタンス {sid} が、{sid} の内部リソースインスタンス情報と一致しません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122634 | ERROR | Oracle SID mismatch. The instance SID "%s" does not match the SID "%s" specified for the command. | **原因**: リソースリカバリの際に、recovery に渡されるリソースインスタンス {sid} が、{sid} の内部リソースインスタンス情報と一致しません。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122636 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | 原因:{server} で Oracle リソース階層 {tag} の作成に失敗しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122638 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因:サーバ {server} の Oracle データベースリソース {tag} で作成アクションに失敗しました。シグナル {sig} が作成プロセスによって受信されました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122640 | ERROR | Error creating resource "%s" on server "%s" | 原因: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122641 | ERROR | Error creating resource "%s" on server "%s" | 原因: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122642 | ERROR | Error creating resource "%s" on server "%s" | 原因: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポート |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | されているエラーを修正してください。 |
| 122643 | ERROR | Cannot extend resource "%s" to server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を拡張できませんでした。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122644 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: リソースインスタンス情報 {tag} を {server} で取得しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正し、extend を再試行してください。 |
| 122645 | ERROR | Cannot access canextend script "%s" on server "%s" | 原因: {server} で依存関係の子リソース用の「canextend」スクリプトが見つからなかったので、LifeKeeper は拡張前のチェックを実行できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122646 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: リソースインスタンス情報 {tag} を {server} で取得しようとして、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正し、extend を再試行してください。 |
| 122647 | ERROR | Cannot extend resource "%s" to server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を拡張できませんでし |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | た。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122648 | ERROR | Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s" | **原因**: データベースリソースの拡張中に、同じ {tag} または {id} で異なるリソースアプリケーションおよびタイプを使用しているリソースインスタンスが見つかりました。<br><br>**処置**: リソース ID は一意のものにする必要があります。Oracle リソースインスタンスに合致する ID のリソースインスタンスを削除してください。 |
| 122649 | ERROR | Error creating resource "%s" on server "%s" | **原因**: Oracle リソースインスタンス {tag} を {server} に作成しようとして、予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122650 | ERROR | Cannot access extend script "%s" on server "%s" | **原因**: 依存関係にある子リソースのスクリプト {extend} がサーバ {server} で見つけられなかったので、サーバ {server} にデータベースリソース {resource} を拡張する要求が失敗しました。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122651 | ERROR | Cannot extend resource "%s" to server "%s" | **原因**: 依存関係にある子リソースの拡張でエラーが発生したため、サーバ {server} にデータベースリソース {resource} を拡張する要求が失敗しました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 隣接するログメッセージで詳細および関連メッセージを確認し、レポートされているエラーを修正してください。 |
| 122654 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**: quickCheck プロセスがシグナルを受信したため、データベース {sid} のヘルスチェックが終了しました。これはおそらく、quickCheck プロセスの処理に、割り当てられているよりも長い時間が必要となっているためです。<br><br>**処置**: Oracle リソースのヘルスチェック時間は ORACLE_QUICKCHECK_TIMEOUT の設定で制御します。ヘルスチェックプロセスの完了に今より長い時間を割り当てる場合は、45 秒よりも大きな値に設定してください。 |
| 122655 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**: 削除プロセスがシグナルを受信したため、データベース {sid} を「Out of Service」にする要求が終了しました。これはおそらく、削除プロセスの処理に、割り当てられているよりも長い時間が必要となっているためです。<br><br>**処置**: Oracle リソースの削除時間は ORACLE_REMOVE_TIMEOUT の設定で制御します。削除プロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122659 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**: リストアプロセスがシグナルを受信したため、データベース {sid} を「In Service」にする要求が終了しました。これはおそらく、リストアプロセスの処理に、割り当てられているよりも長い時間が必要となっているためです。<br><br>**処置**: Oracle リソースのリストア時間は |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ORACLE_RESTORE_TIMEOUT の設定で制御します。リストアプロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122663 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**: リカバリプロセスがシグナルを受信したため、失敗したデータベースのリカバリが終了しました。これはおそらく、リカバリプロセスの処理に、割り当てられているよりも長い時間が必要となっているためです。<br><br>**処置**: Oracle リソースのリカバリ時間は ORACLE_RESTORE_TIMEOUT と ORACLE_REMOVE_TIMEOUT の設定で制御します。リカバリの完了に今より長い時間を割り当てる場合は、一方または両方を 240 秒よりも大きな値に設定してください。 |
| 122670 | ERROR | Update of "%s" sid "%s" on "%s" failed.  Reason: "%s" "%s" failed: "%s". | **原因**: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。更新プロセスで使用する一時ファイルを開こうとして、エラーが発生しました。<br><br>**処置**: システム起動時にデータベースが自動で起動されないようにするために、{sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122671 | ERROR | Update of "%s" sid "%s" on "%s" failed.  Reason: "%s" "%s" failed: "%s". | **原因**: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。更新プロセスで使用する一時ファイルを閉じようとして、エラーが発生しました。<br><br>**処置**: システム起動時にデータベースが自動で起動されないようにするために、 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | {sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122672 | ERROR | Update of "%s" sid "%s" on "%s" failed.  Reason: "%s" "%s" failed: "%s". | 原因: データベース {sid} の oratab エントリを更新しようとしている際に予期しないエラーが発生しました。一時ファイルの名前を oratab に変更しようとして、エラーが発生しました。<br><br>処置: システム起動時にデータベースが自動で起動されないようにするために、{sid} の oratab ファイルエントリを手動で更新する必要があります。 |
| 122673 | ERROR | Unable to log messages queued while running as oracle user %s on %s. Reason: $! | 原因: メッセージをログファイルに追加しようとして、予期しないエラー {reason} が発生しました。これらのメッセージは、Oracle ユーザとして実行している間に生成されました。<br><br>処置: 障害に関する理由を参照し、修正処置を実行してください。 |
| 122674 | ERROR | Unable to open %s Reason: %s. | 原因: Oracle データベースへの接続を開き、データベース {cmd} を実行しようとしたときに、予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細および関連メッセージを確認してください。また、Oracle ログ (alert.log) および関連トレースログ (*.trc) で詳細を確認し、レポートされている問題を修正してください。 |
| 122680 | ERROR | Unable to find Oracle home for “%s” on “%s”. | 原因: Oracle ホームディレクトリ {Oracle home} には、Oracle インスタンス {sid} の正常な動作に必要なファイルが含まれていないようです。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: Oracle ホームディレクトリ {Oracle home} に Oracle バイナリ、有効な spfile{sid}.ora、または init{sid}.ora ファイルが含まれていることをコマンドラインで確認してください。 |
| 122681 | ERROR | Failed to create object instance for Oracle on "%s". | **原因**: 保護されている Oracle インスタンスの内部表現を作成中に予期しないエラーが発生しました。<br><br>**処置**: 隣接するログメッセージで詳細および関連メッセージを確認してください。レポートされているエラーを修正してから操作を再試行する必要があります。 |
| 122682 | ERROR | Unable to find the Oracle executable "%s" on "%s". | **原因**: 必要な Oracle 実行ファイル {exe} がこのサーバ {server} では見つかりませんでした。<br><br>**処置**: Oracle バイナリおよび必要なソフトウェアユーティリティがインストールされ、サーバ {server} で適切に設定されていることを確認してください。Oracle バイナリを各ノードにローカルインストールするか、クラスタ内の全ノードで使用可能な共有ストレージに配置する必要があります。 |
| 122683 | ERROR | Backup node %s is unreachable; abort username/password changes. | **原因**: バックアップノード {node} に接続できません。保留中のユーザー名/パスワードに対する変更は中断されました。<br><br>**処置**: 指定したバックアップノード {node} 上でSIOS LifeKeeper が起動中で、すべてのコミュニケーションパスが確立されていることが確認できたら、再度操作を行ってください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122684 | ERROR | The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper. | **原因**: サーバー {server} 上で Oracle リソース {resource} の restore 処理がタイムアウトしました。これは、In Service 処理が設定された時間を経過すると発生します。<br><br>**処置**: Oracle リソースのリストア時間は ORACLE_RESTORE_TIMEOUT の設定で制御します。リストアプロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122685 | ERROR | The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper. | **原因**: サーバー {server} 上で Oracle リソース {resource} の remove 処理がタイムアウトしました。 これは、Out of Service 処理が設定された時間を経過すると発生します。<br><br>**処置**: Oracle リソースの削除時間は ORACLE_REMOVE_TIMEOUT の設定で制御します。削除プロセスの完了に今より長い時間を割り当てる場合は、240 秒よりも大きな値に設定してください。 |
| 122686 | ERROR | The quickCheck of %s has timed out on server %s. The default TIMEOUT is 45 seconds. To increase the TIMEOUT, set ORACLE_QUICKCHECK_TIMEOUT in /etc/default/LifeKeeper. | **原因**: サーバー {server} 上で Oracle リソース {resource} に対する quickCheck がタイムアウトしました。 これは、health check 処理が設定された時間を経過すると発生します。<br><br>**処置**: Oracle リソースのヘルスチェック時間は ORACLE_QUICKCHECK_TIMEOUT の設定で制御します。ヘルスチェックプロセスの完了に今より長い時間を割り当てる場合は、45 秒よりも大きな値に設定してください。 |
| 122687 | ERROR | Usage: %s %s | **原因**: getalertlog スクリプトに対して無 |

| コード | 重大度 | メッセージ | 原因/処置 |
| --- | --- | --- | --- |
| | | | 効なパラメーターが指定されました。<br><br>処置: Oracle リソースのタグ名が、最初のコマンドラインのパラメーターとして提供される必要があります。パラメーターが正しいかどうかを確認し、再度操作を行ってください。 |

# 8.1.9. Oracle Listener キットメッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122005 | ERROR | Unable to "%s" on "%s" | 原因:「getlocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122007 | ERROR | Unable to "%s" on "%s" | 原因:「rlslocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122009 | ERROR | The path %s is not a valid file. | 原因: listener.ora ファイルがありません。<br><br>処置: ファイルが存在していることを確認し、操作を再試行してください。 |
| 122010 | ERROR | The listener user does not exist on the server %s. | 原因:「Stat」コマンドでユーザ ID を取得できませんでした。<br><br>処置: 操作を再試行してください。 |
| 122011 | ERROR | The listener user does not exist on the server %s. | 原因: UID が passwd ファイルにありません。<br><br>処置: UID が passwd ファイルに存在していることを確認し、操作を再試行して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください。 |
| 122012 | ERROR | The listener user does not exist on the server %s. | 原因:ユーザ名が passwd ファイルにありません。<br><br>処置: ユーザ名が passwd ファイルに存在していることを確認し、操作を再試行してください。 |
| 122023 | ERROR | The %s command failed (%d) | 原因: このメッセージに「lsnrctl」コマンドの戻りコードが含まれています。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122024 | ERROR | $line | 原因: メッセージに「lsnrctl」コマンドの結果が含まれています。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122039 | ERROR | Usage error | 原因: リストア操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122040 | ERROR | Script $cmd has hung on the restore of \"$opt_t\". Forcibly terminating. | 原因: リスナーの restore スクリプトがタイムアウト値に達しました。<br><br>処置: listener.ora が有効であること、および /etc/default/LifeKeeper の LSNR_START_TIME（デフォルトでは 35 秒）がリスナーの起動に必要な時間以上の値に設定されていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122041 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: LifeKeeper は、{server} でリソース {resource} をリストアできませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122045 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: リソース情報の取得に失敗しました。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122046 | ERROR | Usage error | 原因: リストア操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122049 | ERROR | The script $cmd has hung on remove of \"$opt_t\".  Forcibly terminating. | 原因: リスナーの remove スクリプトがタイムアウト値に達しました。<br><br>処置: listener.ora が有効であること、および /etc/default/LifeKeeper の LSNR_STOP_TIME (デフォルトでは 35 秒) がリスナーの停止に必要な時間以上の値に設定されていることを確認してください。 |
| 122051 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122055 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: LifeKeeper は、{server} でリソース {resource} に対して quickCheck を実 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 行できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122057 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122064 | WARN | The %s level is set to %s a %s will not occur. | **原因**: リスナーの最低限の保護レベルは起動と監視です。<br><br>**処置**: リスナーを手動で起動してください。 |
| 122066 | ERROR | Script has hung checking \"$tag\". Forcibly terminating. | **原因**: リスナーの quickCheck スクリプトがタイムアウト値に達しました。<br><br>**処置**: listener.ora が有効であること、および /etc/default/LifeKeeper の LSNR_STATUS_TIME (デフォルトでは 15 秒) がリスナーの確認に必要な時間以上の値に設定されていることを確認してください。 |
| 122067 | ERROR | Usage error | **原因**: quickCheck 操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122069 | ERROR | Usage error | **原因**: 削除操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 行してください。 |
| 122072 | ERROR | %s: resource "%s" not found on local server | **原因**: 復旧操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122074 | WARN | The local recovery attempt has failed but %s level is set to %s preventing a failover to another node in the cluster. With %s recovery set all local recovery failures will exit successfully to prevent resource failovers. | **原因**: オプションのリスナーリカバリレベルがローカルリカバリにのみ設定されています。<br><br>**処置**: リソースツリーを手動でスイッチオーバしてください。 |
| 122078 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | **原因**:LifeKeeper は、{server} でリソース {resource} を復旧できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122082 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122083 | ERROR | $cmd has hung checking \"$tag\". Forcibly terminating | **原因**: recover スクリプトがシグナルによって停止されました。<br><br>**処置**: listener.ora が有効であることを確認してください。 |
| 122084 | ERROR | Cannot extend resource "%s" to server "%s" | **原因**: LifeKeeper は、{server} でリソー |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ス {resource} を拡張できませんでした。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122085 | ERROR | Usage: %s %s | **原因**: canextend 操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122086 | ERROR | The values specified for the target and the template servers are the same. Please specify the correct values for the target and template servers. | **原因**: ターゲットサーバに指定した値とテンプレートサーバに指定した値が同じです。<br><br>**処置**: メッセージテキストに記載されている手順を実行してください。 |
| 122087 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122088 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: リソース情報からのリスナーユーザ名の取得に失敗しました。<br><br>**処置**: リソース情報フィールドが有効であることを確認し、操作を再試行してください。 |
| 122089 | ERROR | The listener user %s does not exist on the server %s. | **原因**: ユーザ名が passwd ファイルにありません。<br><br>**処置**: ユーザ名が passwd ファイルに存在していることを確認し、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122090 | ERROR | The id for user %s is not the same on template server %s and target server %s. | **原因**: ユーザ ID はどちらのサーバでも同じものにする必要があります。<br><br>**処置**: ユーザ ID を同じものにしてください。 |
| 122091 | ERROR | The group id for user %s is not the same on template server %s and target server %s. | **原因**: グループ ID はどちらのサーバでも同じものにする必要があります。<br><br>**処置**: グループ ID を同じものにしてください。 |
| 122092 | ERROR | Cannot access canextend script "%s" on server "%s" | **原因**: {server} で「canextend」スクリプトを見つけられなかったので、LifeKeeper は拡張前のチェックを実行できませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122097 | ERROR | Usage: %s %s | **原因**:「configActions」操作で無効な引数が指定されました。<br><br>**処置**: 引数を確認し、操作を再試行してください。 |
| 122098 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122099 | ERROR | Unable to update the resource %s to change the %s to %s on %s. | **原因**: LifeKeeper は情報フィールドに情報を入力できませんでした。<br><br>**処置**: LifeKeeper を再起動し、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122100 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122101 | ERROR | Unable to update the resource %s to change the %s to %s on %s. | 原因: LifeKeeper は {server} で情報フィールドに情報を入力できませんでした。<br><br>処置: {server} で LifeKeeper を再起動し、操作を再試行してください。 |
| 122103 | ERROR | Usage: %s %s | 原因: 作成操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122124 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122126 | ERROR | Unable to "%s" on "%s | 原因:「rlslocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122127 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: 関連するエラーをログで確認し、 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | レポートされている問題を解決してください。 |
| 122129 | ERROR | Unable to "%s" on "%s | 原因:「getlocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122131 | ERROR | Error creating resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>処置: LifeKeeper の設定を確認してください。 |
| 122133 | ERROR | Unable to create a file system resource hierarchy for the file system %s. | 原因:「filesyshier」の実行中に予期しないエラーが発生しました。<br><br>処置: 隣接するログメッセージで詳細を確認してください。 |
| 122135 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122140 | ERROR | Resource "%s" is not ISP on server "%s" Manually bring the resource in service and retry the operation | 原因: リスナーリソースが依存する IP リソース {tag} は ISP の必要があります。<br><br>処置: メッセージテキストに記載されている手順を実行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122141 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122144 | ERROR | Usage: %s %s | 原因:「create_ins」操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122145 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | 原因:「app_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122146 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | 原因:「typ_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122147 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | 原因:「newtag」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122148 | ERROR | Error creating resource "%s" on server "%s" | 原因: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: LifeKeeper の設定を確認してください。 |
| 122149 | ERROR | An error has occurred in utility %s on server %s. View the LifeKeeper logs for details and retry the operation. | **原因**:「ins_setstate」の実行中に予期しないエラーが発生しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122150 | ERROR | Error creating resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {resource} を作成できませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122151 | ERROR | Usage: %s %s | **原因**:「depstoextend」操作で無効な引数が指定されました。<br><br>**処置**: 引数を確認し、操作を再試行してください。 |
| 122152 | ERROR | Usage: %s %s | **原因**:「extend」操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122153 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122154 | ERROR | Cannot extend resource "%s" to server "%s" | **原因**: LifeKeeper は、{server} でリソー |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ス {resource} を拡張できませんでした。 |
| 122155 | ERROR | Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s" | **原因**: リスナーリソースの拡張中に、同じ {tag} または {id} で異なるリソースアプリケーションおよびタイプを使用しているリソースインスタンスが見つかりました。<br><br>**処置**: リソース ID は一意のものにする必要があります。Oracle Listener リソースインスタンスに合致する ID のリソースインスタンスを削除してください。 |
| 122156 | ERROR | Cannot access extend script "%s" on server "%s" | **原因**: LifeKeeper は、{server} でスクリプト EXTEND が見つからなかったので、リソース階層を拡張できませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122157 | ERROR | Usage: %s %s | **原因**:「getConfigIps」操作で無効な引数が指定されました。<br><br>**処置**: 引数を確認し、操作を再試行してください。 |
| 122158 | ERROR | The file %s is not a valid listener file. The file does not contain any listener definitions. | **原因**: リスナー定義が見つかりませんでした。<br><br>**処置**: リスナー定義が listener.ora にあることを確認し、操作を再試行してください。 |
| 122159 | ERROR | Usage: %s %s | **原因**:「getSidListeners」操作で無効な引数が指定されました。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 引数を確認し、操作を再試行してください。 |
| 122160 | ERROR | The file %s is not a valid listener file. The file does not contain any listener definitions. | **原因**: リスナー定義が見つかりませんでした。<br><br>**処置**: リスナー定義が listener.ora にあることを確認し、操作を再試行してください。 |
| 122161 | ERROR | Usage: %s %s | **原因**:「lsn-display」操作で無効な引数が指定されました。<br><br>**処置**: 引数を確認し、操作を再試行してください。 |
| 122162 | ERROR | Error getting resource information for resource "%s" on server "%s" | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 122163 | ERROR | Usage: %s %s | **原因**: updateHelper 操作で無効なパラメータが指定されました。<br><br>**処置**: パラメータを確認し、操作を再試行してください。 |
| 122164 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | **原因**: LifeKeeper は、{server} でリソース {resource} を更新できませんでした。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122166 | ERROR | Usage: %s %s | 原因:「updateHelper」操作で無効なパラメータが指定されました。<br><br>処置: パラメータを確認し、操作を再試行してください。 |
| 122170 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122171 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122172 | ERROR | Usage: %s %s | 原因:「updIPDeps」操作で無効な引数が指定されました。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122173 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | 原因: LifeKeeper は、{server} でリソース {resource} を更新できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122175 | ERROR | Unable to "%s" on "%s | 原因:「rlslocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | レポートされている問題を解決してください。 |
| 122177 | ERROR | Unable to "%s" on "%s | 原因:「getlocks」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122180 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122181 | ERROR | Unable to create a dependency between parent tag %s and child tag %s. | 原因:「dep_create」の実行中に予期しないエラーが発生しました。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 122183 | ERROR | The path %s is not a valid file. | 原因: listener.ora ファイルがありません。<br><br>処置: ファイルが存在していることを確認し、操作を再試行してください。 |
| 122185 | ERROR | The file %s is not a valid listener file. The file does not contain any listener definitions. | 原因: LifeKeeper は有効なリスナー定義の検索に失敗しました。<br><br>処置: 有効なリスナー定義が listener.ora にあることを確認し、操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122186 | ERROR | The value specified for %s cannot be empty. Please specify a value for this field. | 原因:設定ファイルまたは実行可能ファイルの {path} フィールドが空です。<br><br>処置: 空でない値を {path} に入力し、操作を再試行してください。 |
| 122187 | ERROR | The path %s is not a valid file or directory. | 原因: 定義済み {path} が無効です。<br><br>処置: {path} が存在していることを確認し、操作を再試行してください。 |
| 122188 | ERROR | The path %s is not a valid file or directory. | 原因: {path} がありません。<br><br>処置: {path} が存在していることを確認し、操作を再試行してください。 |
| 122189 | ERROR | The value specified for %s cannot be empty. Please specify a value for this field. | 原因: 設定および実行パスフィールドが空です。<br><br>処置: フィールドのパスを入力してください。 |
| 122190 | ERROR | Usage: %s %s | 原因:「valid_rpath」操作で無効な引数が指定されました。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122191 | ERROR | The values specified for the target and the template servers are the same. | 原因: valid_rpath の無効な引数。<br><br>処置: 引数を確認し、操作を再試行してください。 |
| 122192 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあるこ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | と、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122193 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122194 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122195 | ERROR | Unable to find the configuration file "oratab" in its default locations, /etc/oratab or %s on "%s" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122196 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | 原因:LifeKeeper は、{server} でリソース {resource} を削除できませんでした。<br><br>処置: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122197 | ERROR | Unable to find the configuration file \"oratab\" in its default locations, /etc/oratab or $listener::oraTab on \"$me\" | 原因: /etc/oratab または {path} に oratab ファイルがありません。<br><br>処置: oratab ファイルが {path} にあること、または /etc/default/Lifekeeper の ORACLE_ORATABLOC が有効なパスに設定されていることを確認してください。 |
| 122198 | ERROR | remove for $okListener failed. | |

# 8.1.10. Oracle PDB キットメッセージカタログ

Use **Control F** Control F を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122251 | ERROR | Update of pluggable database info field for "%s" on "%s" failed (%s). | |
| 122252 | ERROR | Initial connect with query buffer to database "%s" on "%s" failed, testing output. | |
| 122253 | ERROR | The Oracle database "%s" is not running or no open connections are available on "%s". | |
| 122261 | ERROR | The Oracle resource (%s) and dependency are not set on %s. | |
| 122262 | ERROR | Usage: %s %s | |
| 122263 | ERROR | The restore of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_RESTORE_TIMEOUT in /etc/default/LifeKeeper. | |
| 122264 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | |
| 122268 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122269 | ERROR | no dependency for Oracle on "%s". | |
| 122270 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122271 | ERROR | Usage: %s %s | |
| 122272 | ERROR | The remove of %s has timed out on server %s. The default TIMEOUT is 300 seconds. To increase the TIMEOUT, set ORACLE_REMOVE_TIMEOUT in /etc/default/LifeKeeper. | |
| 122273 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | |
| 122277 | ERROR | Usage: %s %s | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122278 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122279 | ERROR | no dependency for Oracle on "%s". | |
| 122280 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122281 | ERROR | The quickCheck of %s has timed out on server %s. The default TIMEOUT is 45 seconds. To increase the TIMEOUT, set ORACLE_QUICKCHECK_TIMEOUT in /etc/default/LifeKeeper. | |
| 122282 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | |
| 122284 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122285 | ERROR | no dependency for Oracle on "%s". | |
| 122287 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122288 | ERROR | Usage: %s %s | |
| 122291 | ERROR | Cannot extend resource "%s" to server "%s" | |
| 122292 | ERROR | The values specified for the target and the template servers are the same: "%s". | |
| 122294 | ERROR | Cannot access canextend script "%s" on server "%s" | |
| 122295 | ERROR | Usage: %s %s | |
| 122296 | ERROR | DB instance "%s" is not protected on "%s". | |
| 122297 | ERROR | Failed to create object instance for OraclePDB on "%s". | |
| 122298 | ERROR | Unable to locate the oratab file "%s" on "%s". | |
| 122299 | ERROR | END failed hierarchy "%s" of resource "%s" on server "%s" with return value of %d | |
| 122301 | ERROR | Unable to "%s" on "%s" during resource create. | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122302 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | |
| 122304 | ERROR | Unable to "%s" on "%s" during resource create. | |
| 122305 | ERROR | Unable to determine Oracle user for "%s" on "%s". | |
| 122306 | ERROR | Error creating resource "%s" on server "%s" | |
| 122308 | ERROR | Dependency creation between Oracle pluggable database "%s (%s)" and the dependent resource "%s" on "%s" failed. Reason | |
| 122309 | ERROR | %s | |
| 122311 | ERROR | In-service attempted failed for tag "%s" on "%s". | |
| 122312 | ERROR | Usage: %s %s | |
| 122313 | ERROR | Create of app "%s" on "%s" failed with return code of "%d". | |
| 122314 | ERROR | Create of typ "%s" for app "%s" on "%s" failed with return code of "%d". | |
| 122316 | ERROR | Create of resource tag via "newtag" on "%s" failed. | |
| 122318 | ERROR | Error creating resource "%s" on server "%s" | |
| 122320 | ERROR | Setting "resstate" for resource "%s" on "%s" failed with return code of "%d". | |
| 122321 | ERROR | Error creating resource "%s" on server "%s" | |
| 122322 | ERROR | Usage: %s %s | |
| 122323 | ERROR | Usage: %s %s | |
| 122324 | ERROR | Usage: %s %s | |
| 122325 | ERROR | Cannot extend resource "%s" to server "%s" | |
| 122326 | ERROR | Resource with either matching tag "%s" or id "%s" already exists on server "%s" for App "%s" and Type "%s" | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 122327 | ERROR | Error creating resource "%s" on server "%s" | |
| 122328 | ERROR | Cannot access extend script "%s" on server "%s" | |
| 122329 | ERROR | Cannot extend resource "%s" to server "%s" | |
| 122330 | ERROR | Usage: %s %s | |
| 122331 | ERROR | Failed to create object instance for OraclePDB on "%s". | |
| 122332 | ERROR | Usage: %s %s | |
| 122334 | ERROR | Backup node %s is unreachable; abort protection PDB changes. | |
| 122336 | ERROR | Update of protection PDB failed for "%s" on "%s". | |
| 122339 | ERROR | Usage: %s %s | |
| 122340 | ERROR | Usage: %s %s | |
| 122341 | ERROR | Usage: %s %s | |
| 122342 | ERROR | Failed to create object instance for OraclePDB on "%s". | |
| 122343 | ERROR | Usage: %s %s | |
| 122344 | ERROR | END failed %s of "%s" on server "%s" due to a "%s" signal | |
| 122348 | ERROR | Failed to create flag "%s" on "%s". | |
| 122350 | ERROR | Failed to create object instance for Oracle on "%s". | |
| 122351 | ERROR | no dependency for Oracle on "%s". | |
| 122353 | ERROR | Unable to find the Oracle executable "%s" on "%s". | |
| 122356 | ERROR | Usage: %s %s | |
| 122357 | ERROR | Failed to create object instance for OraclePDB on "%s". | |
| 122358 | ERROR | The selected oracle SID "%s" is not a CDB. | |
| 122359 | ERROR | No protectable PDB found for the | |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | selected SID "%s". | |
| 122360 | ERROR | No protected Oracle database found on "%s". | |

# 8.1.11. SCSI キットメッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 125102 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125103 | ERROR | `printf '%s is not shareable with any machine.' $DEV` | 原因: デバイスは他のシステムと共有されていないようです。<br><br>処置: デバイスがクラスタ内のすべてのサーバからアクセス可能であることを確認してください。すべての関連ストレージドライバおよびソフトウェアが正常にインストールおよび設定されていることを確認してください。 |
| 125104 | ERROR | `printf 'Failed to create disk hierarchy for "%s" on "%s"' $PRIMACH $DEV` | 原因: 物理ディスクを保護するリソースの作成に失敗しました。<br><br>処置: 隣接するログメッセージで詳細を確認し、レポートされている問題を解決してください。 |
| 125107 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125114 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125120 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | 原因: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125123 | ERROR | `printf 'Cannot access depstoextend script "%s" on server "%s"' $depstoextend` | 原因: {server} でスクリプト |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | $TargetSysName` | {DEPSTOEXTEND}を見つけられなかったので、LifeKeeper はリソース階層に対して拡張前のチェックを実行できませんでした。<br><br>**処置**: LifeKeeper の設定を確認してください。 |
| 125126 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $ChildTag $TemplateSysName` | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125129 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTagName $TemplateSysName` | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125155 | ERROR | SCSI $DEV failed to lock. | **原因**: SCSI デバイスをロックする際に問題が発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、レポートされている問題を解決してください。 |
| 125164 | ERROR | SCSI $INFO failed to unlock. | **原因**: SCSI デバイスをロック解除する際に問題が発生しました。<br><br>**処置**: 隣接するログメッセージで詳細を確認し、レポートされている問題を解決してください。 |
| 125181 | ERROR | `printf 'Template resource "%s" on server "%s" does not exist' $TemplateTag  $TemplateSysName` | **原因**: LifeKeeper は、{server} でリソース {tag} を見つけられませんでした。 |
| 125194 | ERROR | Failed to check disk.(tag=\"$opt_t\" | |

# 8.1.12. Quick Service Protection Recovery Kit メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 134003 | ERROR | catch a \"$sig\" signal | **原因**: create処理がシグナルによって中断されました。<br><br>**処置**: 前後のログメッセージを確認してください。 |
| 134004 | ERROR | Unable to getlocks on $server during resource create. Error ($rc) | **原因**: リソース階層の作成時に管理ロックの取得に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134005 | ERROR | The service \"$serviceName\" is not supported on $server. Error ($rc) | **原因**: サービスが存在しないか保護できないサービスです。<br><br>**処置**: 適切なサービス名を入力してください。 |
| 134006 | ERROR | The service \"$serviceName\" is already protected on $server. | **原因**: すでに保護されているサービスです。<br><br>**処置**: このサービスを保護するためのリソースを作成することはできません。 |
| 134007 | ERROR | Error creating resource $tag. Error ($rc) | **原因**: QSPリソースの作成に失敗しました。<br><br>**処置**: 前後のログメッセージを確認して、エラーの原因を取り除いてくださ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | い。 |
| 134011 | ERROR | In-service attempted failed for tag $tag. | **原因**: QSPリソースの起動に失敗しました。<br><br>**処置**: 保護したいサービスに関連したログを確認して、問題を解決してください。 |
| 134015 | ERROR | Unable to rlslocks on $server during resource create. Error ($rc) | **原因**: QSPリソースの作成後にロックの解除に失敗しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134103 | ERROR | Template resource \"$template_tag\" on server \"$template_sys\" does not exist | **原因**: テンプレートシステムでリソースを見つけられません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 134104 | ERROR | Template resource \”$template_tag\” on server \”$template_sys\” is not QSP resource (app=$ins[1], res=$ins[2]) | **原因**: テンプレートリソースはQSPリソースではありません。<br><br>**処置**: テンプレートリソースと同じタイプに拡張してください。 |
| 134105 | ERROR | The service \"$service\" is not supported on $me. Error ($check) | **原因**: 拡張先のサーバにサービスが存在しません。<br><br>**処置**: 拡張先サーバにサービスをインストールしてから拡張してください。 |
| 134106 | ERROR | The service \"$service\" is already protected on $me. | **原因**: 拡張先サーバに同じIDのリソース |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | が既にあります。<br><br>**処置**: 同じサービスのリソースを作成することはできません。 |
| 134203 | ERROR | catch a \"$sig\" signal | **原因**: リソース拡張処理がシグナルによって中断されました。<br><br>**処置**: 前後のログメッセージを確認してください。 |
| 134204 | ERROR | Template resource \"$template_tag\" on server \"$template_sys\" does not exist | **原因**: テンプレートシステムでリソースを見つけられません。<br><br>**処置**: テンプレートシステムで階層が正しいことを確認してから拡張してください。 |
| 134208 | ERROR | Error creating resource \"$tag\" on server \"$me\" | **原因**: LifeKeeperはターゲットサーバーにリソースインスタンスを作成できませんでした。<br><br>**処置**: 前後のログメッセージを確認して、エラーの原因を取り除いてください。 |
| 134401 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | **原因**: サービスの起動処理が規定時間を超えても終了しません。<br><br>**処置**: 保護しているサービスについて確認し、restore操作を再試行してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134405 | FATAL | Failed to fork process to execute service command: $! | **原因**: forkが失敗しました。これはシステムエラーです。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: forkが失敗する理由を特定してください。 |
| 134407 | ERROR | service command has failed for \"$tag\" | **原因**: serviceコマンドの実行に失敗しました。<br><br>**処置**: startオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原因を取り除いてください。 |
| 134501 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | **原因**: サービスの停止処理が規定時間を超えても終了しません。<br><br>**処置**: 保護しているサービスについて確認し、remove操作を再試行してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134505 | FATAL | Failed to fork process to execute service command: $! | **原因**: forkが失敗しました。これはシステムエラーです。<br><br>**処置**: forkが失敗する理由を特定してください。 |
| 134507 | ERROR | service command has failed for \"$tag\" | **原因**: serviceコマンドの実行に失敗しました。<br><br>**処置**: stopオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原因を取り除いてください。 |
| 134601 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | **原因**: サービスの監視処理が規定時間を超えても終了しません。<br><br>**処置**: 保護しているサービスについて確 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134605 | FATAL | Failed to fork process to execute service command: $! | 原因: forkが失敗しました。これはシステムエラーです。<br><br>処置: forkが失敗する理由を特定してください。 |
| 134607 | ERROR | service command has failed for \"$tag\" | 原因: serviceコマンドの実行に失敗しました。<br><br>処置: statusオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原因を取り除いてください。 |
| 134701 | ERROR | timeout $cmd for \"$tag\". Forcibly terminating. | 原因: サービスの回復処理が規定時間を超えても終了しません。<br><br>処置: 保護しているサービスについて確認してください。また、関連するエラーをログで確認し、レポートされている問題を解決してください。 |
| 134706 | FATAL | Failed to fork process to execute service command: $! | 原因: forkが失敗しました。これはシステムエラーです。<br><br>処置: forkが失敗する理由を特定してください。 |
| 134708 | ERROR | service command has failed for \"$tag\" | 原因: serviceコマンドの実行に失敗しました。<br><br>処置: startオプションでserviceコマンドを手動実行するとエラーになります。エラーメッセージを参考にしてエラーの原 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 因を取り除いてください。 |
| 134803 | ERROR | tag \"$tag\" does not exist on server \"$me\" | 原因: 指定されたタグが存在しません。これは内部エラーです。 |
| 134804 | ERROR | app type \”$ins[1]\” is not $app | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134805 | ERROR | res type \”$ins[2]\” is not $res | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134823 | ERROR | tag \"$tag\" does not exist on server \"$me\" | 原因: 指定されたタグが存在しません。これは内部エラーです。 |
| 134824 | ERROR | app type \”$ins[1]\” is not $app | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134825 | ERROR | res type \”$ins[2]\” is not $res | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134843 | ERROR | tag \"$tag\" does not exist | 原因: 指定されたタグが存在しません。これは内部エラーです。 |
| 134844 | ERROR | app type \”$ins[1]\” is not $app | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |
| 134845 | ERROR | res type \”$ins[2]\” is not $res | 原因: 指定されたタグはQSPリソースではありません。これは内部エラーです。 |

# 8.1.13. GUI メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 000200 | ERROR | pam_start() failed | |
| 000201 | ERROR | pam_authenticate failed (user %s, retval %d) | |
| 000202 | ERROR | pam_end() failed?!?! | |
| 000203 | ERROR | Did not find expected group ‘lkguest’ | |
| 000204 | ERROR | Did not find expected group ‘lkoper’ | |
| 000205 | ERROR | Did not find expected group ‘lkadmin’ | |
| 000208 | ERROR | pam_setcred establish credentials failed (user %s, retval %d) | **原因**: ユーザー {user} の有効なログイン資格情報を確立できません。 pam_setcred 呼び出しで返された値は次の通りです：{retval}<br><br>**処置**: 詳細については、/var/log/security および /var/log/messages を確認して下さい。 |
| 000209 | ERROR | pam_setcred delete credentials failed (user %s, retval %d) | **原因**: ユーザー {user} の有効なログイン資格情報を確立できません。 pam_setcred 呼び出しで返された値は次の通りです：{retval}<br><br>**処置**: 詳細については、/var/log/security および /var/log/messages を確認して下さい。 |
| 000902 | ERROR | Error removing system name from loopback address line in /etc/hosts file. You must do this manually before starting the GUI server. | **原因**: システム名が /etc/hosts ファイルから削除されていません。<br><br>**処置**: システム名を手動で削除し、GUI サーバを再起動して、次のコマンドを入力します: run <action name> |

| コード | 重大度 | メッセージ | 原因/処置 |
| --- | --- | --- | --- |
| 000918 | ERROR | LifeKeeper GUI Server error during Startup | **原因**: 異常な状態により、GUI サーバが終了しました。<br><br>**処置**: 関連するエラーをログで確認し、レポートされている問題を解決してください。 |

# 8.1.14. SAP メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112000 | FATAL | Usage: "%s %s". Specify the correct usage for the requested command. | 原因: SAP HANA createスクリプトに無効なパラメーターが指定された。<br><br>処置: 適切なパラメーターをSAP HANA createスクリプトに指定してください。 |
| 112004 | ERROR | Neither the SID/instance pair nor the tag parameter were specified for the internal "%s" routine on %s. If this was a command line operation, specify the correct parameters. Otherwise, consult the troubleshooting documentation. | 原因: 内部SAPオブジェクトの作成において無効なパラメーターが指定された。<br><br>処置: SIDおよびインスタンスまたは LifeKeeper のリソースタグ名のいずれかを指定してください。 |
| 112013 | ERROR | The file "%s" does not exist or was not readable on %s. Verify that the specified file exists and/or is readable. | 原因: sapservices ファイルが存在しないか、読み込み不可。<br><br>処置: sapservices ファイルが存在し、読み込み可能であることを確認してください。 |
| 112017 | ERROR | Error getting resource information for resource "%s" on server "%s" | 原因: 指定したサーバー上で指定したリソースに関連する情報を見つけることができない。<br><br>処置: 指定したサーバーに指定したリソースが存在していて、すべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112019 | ERROR | The attempt to update the resource information field for resource %s has failed on %s. View the resource properties manually using "ins_list -t <tag>" to verify that the resource is | 原因: 指定したサーバーで指定したリソースの LifeKeeper リソース情報フィールドを更新できない。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | functional. | **処置**: 指定したサーバーにリソースが存在しているか、もしくはLifeKeeper が起動しているかを確認してください。 |
| 112022 | ERROR | An error occurred while trying to find the IP address corresponding to "%s" on %s. Verify the IP address or host name exists in DNS or the hosts file. | **原因**: 指定したサーバーにIPアドレスもしくはホスト名が見つからない。<br><br>**処置**: IPアドレスもしくはDNS名がDNSもしくはローカルホストファイルに存在しているかを確認してください。 |
| 112024 | FATAL | There was an error verifying the NFS connections for SAP related mount points on $me. One or more NFS servers is not operational and needs to be restarted. | **原因**: SAP_NFS_CHECK_DIRS パラメーター に掲載されている少なくとも一つの NFS 共有ファイルシステムを現在使用できない。<br><br>**処置**: NFSサーバーの起動、すべての必要とされるNFS関連のサービスの起動、すべての必要とされるファイルシステムがエクスポートされていることを確認してください。 |
| 112028 | ERROR | Unable to determine the user name for the SAP administrative user for resource %s on %s. | **原因**: 指定したサーバー上に指定したリソース情報を確認することができない。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していて、すべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112029 | ERROR | The canextend script "%s" either does not exist or is not executable on %s. | **原因**: SAP canextend スクリプトが存在していないか、実行できない。<br><br>**処置**: SAP canextend スクリプトが存在していて、実行可能か確認してください。 |
| 112037 | ERROR | Unable to create an internal object for the SAP instance using SID %s, instance %s, | **原因**: 指定した SAP インスタンスを表す |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | and tag %s on server %s. Verify that all necessary SAP file systems are mounted and accessible before reattempting the operation. | ための内部の SAP オブジェクトを作成することができない。<br><br>処置: SAP インスタンスが正常にインストール、設定されていて、すべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112040 | ERROR | The SAP Directory "%s" ("%s") does not exist on %s. Verify that the directory exists and that the SAP software is properly installed. | 原因: 指定したSAP インストールディレクトリが指定したサーバーに存在していない。<br><br>処置: SAP ソフトウェアが正常にインストールされすべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112041 | ERROR | The required utility "%s" was not found or was not executable on %s. Verify the SAP installation and location of the required utility. | 原因: saphostexec または saposcol が見つからないか実行できない。<br><br>処置: SAP Host Agent package が正しくインストールされ、すべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112042 | ERROR | One or more SAP or LifeKeeper validation checks has failed on %s. Please update the SAP software on this host to include the SAPHOST and SAPCONTROL packages. | 原因: saphostexec または saposcol が見つからないか実行できない。<br><br>処置: SAP Host Agent package が正しくインストールされ、すべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112048 | ERROR | The SAP instance %s is already under LifeKeeper protection on server %s. Choose another SAP instance to protect or specify the correct instance. | 原因: 指定した SAP インスタンスはすでに指定したサーバーでLifeKeeper により保護されている。<br><br>処置: まだLifeKeeper により保護されていない SAP インスタンスを選択してく |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ださい。 |
| 112049 | ERROR | Unable to locate the SAP Mount directory on %s. Verify that all SAP file systems are mounted and accessible before reattempting the operation. | **原因**: SAP のマウント (sapmnt) ディレクトリの場所を特定できない。<br><br>**処置**: すべての必要となるファイルシステムがマウントされ、すべての必要となる SAP インスタンスプロファイルがアクセス可能になっていることを確認してください。 |
| 112050 | ERROR | Detected multiple virtual IP addresses/ host names for instance %s on %s. Verify that the instance is configured correctly. | **原因**: 指定したサーバー上に指定した SAP インスタンスに対する複数の仮想 IP もしくはホスト名が検出された。<br><br>**処置**: インスタンスに関連する仮想 IP もしくはホスト名が正しく構成されているかを確認してください。 |
| 112051 | ERROR | The "%s" or "%s" value in the default profile "%s" is still set to the physical host name on %s. The value(s) must be set to a virtual host name. | **原因**: 指定したサーバーで指定したホスト名のパラメーターが物理サーバーのホスト名に指定されている。<br><br>**処置**: ホスト名のパラメーターは仮想ホスト名に設定してください。 |
| 112053 | ERROR | Detected multiple instances under SID %s with the same instance number (%s) on %s. Each instance within a particular SID must have a unique instance number. | **原因**: 同じインスタンス番号のSAP インスタンスが、同じ SAP SID 下に検出された。<br><br>**処置**: SAP環境を再構築することで、指定した SAP SID 下の各インスタンスが一意のインスタンス番号をもつことができます。 |
| 112056 | ERROR | The NFS export for the path "%s" required by the instance %s for the "%s" directory does not have an NFS hierarchy protecting it on %s. You must create an | **原因**: 指定したファイルシステムのNFS エクスポートが LifeKeeper により保護されていません。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | NFS hierarchy to protect this NFS export before creating the SAP resource hierarchy. | **処置**: エクスポートしたファイルシステムの LifeKeeper NFS 階層を作成し、SAP リソースを作成してください。 |
| 112057 | ERROR | Unable to create a file system resource hierarchy for the file system “%s” on “%s”. | **原因**: 指定したサーバーで指定したファイルシステムを保護するための LifeKeeper ファイルシステムリソース階層を作成することができない。<br><br>**処置**: 詳細について LifeKeeper とシステムログを確認してください。 |
| 112058 | ERROR | Unable to create a dependency between parent tag “%s” and child tag “%s” on “%s”. | **原因**: 指定したサーバーで指定したリソース間での依存関係を作成できない。<br><br>**処置**: 詳細について LifeKeeper のログを確認してください。 |
| 112060 | ERROR | All attempts at local recovery for the SAP resource “%s” have failed on “%s”. A failover to the backup server will be attempted. | **原因**: 指定したサーバーで指定した SAP リソースを復旧できない。<br><br>**処置**: SAP リソース階層のフェイルオーバーは自動的に行われます。ユーザーの介入は必要ありません。 |
| 112061 | ERROR | The values specified for the target and the template servers are the same. Please specify the correct values for the target and template servers. | **原因**: SAP リソース階層の拡張中に提供されたテンプレート/ターゲットサーバーが同一。<br><br>**処置**: テンプレート・ターゲットサーバーの正しい名前を提供し、拡張処理を再度実行してください。 |
| 112062 | ERROR | Unable to find the home directory "%s" for the SAP administrative user "%s" on %s. Verify that the SAP software is installed correctly. | **原因**: 指定したサーバーで指定した SAP ユーザーのホームディレクトリが見つからない。<br><br>**処置**: SAP ソフトウェアが正常にインス |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | トールされ、指定した SID に対する適切な SAP 管理ユーザーがサーバーに存在していることを確認してください。 |
| 112063 | ERROR | The SAP administrative user "%s" does not exist on %s. Verify that the SAP software is installed correctly or create the required SAP user on %s. | 原因: 指定した SAP 管理ユーザーが指定したサーバー上に存在していない。<br><br>処置: SAP ソフトウェアが正常にインストールされているか確認し、必要に応じて SAP 管理ユーザーを作成してください。 |
| 112064 | ERROR | The group ID for user "%s" is not the same on template server "%s" and target server "%s". Please correct the group ID for the user so that it is the same on the template and target servers. | 原因: 指定したSAP 管理ユーザーのグループ IDがテンプレート/ターゲットサーバーで一致していない。<br><br>処置: 指定した SAP 管理ユーザーのグループ ID を修正することで、テンプレート/ターゲットサーバーで同一となります。 |
| 112065 | ERROR | The user ID for user "%s" is not the same on template server "%s" and target server "%s". Please correct the user ID so that it is the same on the template and target servers. | 原因: 指定した SAP 管理ユーザーの ID がテンプレート/ターゲットサーバーで一致していない。<br><br>処置: 指定した SAP 管理ユーザーの ID を修正することで、テンプレート/ターゲットサーバーで同一となります。 |
| 112066 | ERROR | Required SAP utilities could not be found in "%s" on %s. Verify that the SAP software is installed correctly. | 原因: 必要な SAP 実行ファイルもしくは SAP インスタンスプロファイルを配置することができない。<br><br>処置: SAP ソフトウェアが正常にインストールされ構成されていて、すべての必要なファイルシステムがマウントされているか確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112069 | ERROR | The command “%s” is not found in the “%s” perl module (“%s”) on “%s”. Please check the command specified and retry the operation. | **原因**: 指定したコマンドが指定したサーバーの sap perl modedule に見つからない。<br><br>**処置**: ユーザーから開始したコマンドアクションによりエラーとなった場合は、正しいルーチン名が remoteControl スクリプトに渡されているかを確認してください。通常のLifeKeeper 運用において発生したエラーの場合は、製品サポートへお問い合わせください。 |
| 112071 | ERROR | The file “%s” exists, but was not read and write enabled on “%s”. Enable read and write permissions on the specified file | **原因**: 指定したファイルが指定したサーバーで読み込み/書き込み権限をもっていない。<br><br>**処置**: 指定したサーバーで読み込み/書き込み権限を有効にしてください。 |
| 112073 | ERROR | Unable to create an internal object for the SAP instance using either SID "%s" and instance "%s" or tag "%s" on server "%s". Either the values specified for the object initialization (SID/instance pair or tag, system) were not valid, or an error occurred while attempting to gather information about the SAP instance. If all specified parameters are correct, verify that all necessary SAP file systems are mounted and accessible before reattempting the operation. | **原因**: 指定した SAP インスタンスを示すための内部 SAP オブジェクトを作成することができない。<br><br>**処置**: SAP インスタンスが正常にインストール、設定されていて、すべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112074 | WARN | Warning: The profile file %s for SID %s and Instance %s has Autostart is enabled on %s. Disable Autostart for the specified instance by setting Autostart=0 in the profile file | **原因**: Autostart パラメーターが指定したサーバーの指定したインスタンスプロファイルで有効になっている。<br><br>**処置**: 指定した SAP インスタンスがインスタンスプロファイルの中で ‘Autostart = 0’に設定されており、Autostartが有効になっています。Autostartを無効にしてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112076 | FATAL | Unable to start the sapstartsrv service for SAP SID $sid and SAP instance $Instance on $me. Verify that the sapservices file is correct and the process can be started manually. | 原因: 指定した SAP インスタンスに対して SAP Start Service (sapstartsrv) のプロセスを開始できない。<br><br>処置: sapservices ファイルが sapstartsrv プロセスを開始するための適切なコマンドを含んでいて、そのプロセスを手動で開始できるかを確認してください。 |
| 112077 | ERROR | Unable to stop the sapstartsrv service for SAP SID %s and SAP instance %s on %s. Verify that the sapservices file is correct and the process can be stopped manually. | 原因: 指定した SAP インスタンスに対して SAP Start Service (sapstartsrv) プロセスを停止することができない。<br><br>処置: sapservices ファイルが sapstartsrv プロセスを開始するための適切なコマンドを含んでいて、そのプロセスを手動で開始できるかを確認してください。 |
| 112078 | ERROR | ERSv1 is only supported in two-node clusters. Resource %s is unable to be extended to system %s. Upgrade to ERSv2 in order to extend the hierarchy to three or more nodes. | 原因: ERSv1インスタンスを表す SAP リソースを 3つ以上のノードに拡張することができない。<br><br>処置: ERS インスタンスを表す SAP リソースを3つ以上のノードに拡張する場合は、ERSv2 へアップグレードしてください。 アップグレード手順は、オンラインプロダクトドキュメント内に記載されています。 |
| 112082 | WARN | Instance %s is running a different version of the enqueue server than its corresponding enqueue replication server. This configuration is not supported by SAP and will lead to unexpected resource behavior. See SAP Note 2711036 – "Usage of the Standalone Enqueue Server 2 in an HA Environment" for more details. Please review the online product documentation for instructions on how to modify the instance profiles for the | 原因: エンキューサーバーとエンキューレプリケーションサーバーのバージョンが一致していない。<br><br>処置: オンラインプロダクトドキュメントに記載のインスタンスプロファイルの修正方法を確認の上、エンキューサーバーとエンキューレプリケーションサーバーのバージョンを一致させてください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | enqueue server and enqueue replication server so that they use the same version. | |
| 112086 | ERROR | The ERS resource corresponding to resource %s is in-service and maintaining backup locks on a remote system. Bringing resource %s in-service on %s would result in a loss of the backup lock table. Please bring resource %s in-service on the system where the corresponding ERS resource is currently in-service in order to maintain consistency of the lock table. In order to force resource %s in-service on %s, either (i) run the command \'/opt/LifeKeeper/bin/flg_create -f sap_cs_force_restore_%s\' as root on %s and reattempt the in-service operation or (ii) take the corresponding ERS resource out of service on the remote system. Both of these actions will result in a loss of the backup lock table. | **原因**: ASCS/SCS リソースに対応する ERS インスタンスがクラスタ内の異なるサーバーで起動し、バックアップのエンキューテーブルを保存していたので、ASCS/SCS リソースに対して in-service 処理が試された。<br><br>**処置**: バックアップのエンキューテーブルを取得するためにERSインスタンスが起動しているサーバーで、ASCS/SCS リソースを in-sevice にしてください。ASCS/SCS リソースを強制的に in-service にする場合は、 ‘/opt/LifeKeeper/bin/flg_create -f sap_cs_force_restore_<ASCS/SCS Tag>’ コマンドを実行し in-service 処理を試みるか、またはリモートサーバーで対応する ERS リソースを out of service にしてください。これら両方の操作を実行すると、バックアップのエンキューテーブルが失われます。 |
| 112089 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance types TYPE_CS, TYPE_ERS, or TYPE_NEW_ERS (1, 2, or 5). | **原因**: GetEnqVersion ルーチンがサポートされていない SAP インスタンスのタイプに対して呼び出された。インスタンスタイプ 1 (TYPE_CS), 2 (TYPE_ERS), 5 (TYPE_NEW_ERS) のみがサポート対象です。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112092 | ERROR | The profile "%s" either does not exist or cannot be read on %s. Unable to determine whether enqueue replication is enabled for resource %s. Please verify that the file exists and can be read. | **原因**: 指定したサーバー上に指定したインスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>**処置**: ファイルが存在していて、読み取り可能かどうかを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112095 | ERROR | Error creating resource “%s” on server “%s” | **原因**: 指定したサーバーに指定したリソースを作成することができない。<br><br>**処置**: 指定したサーバーにSAP ソフトウェアが正常にインストールされ、すべての必要なファイルシステムがマウントされアクセス可能になっていることを確認してください。 |
| 112096 | ERROR | Resource %s is not currently in-service on server %s. Manually bring the resource in-service and retry the operation. | **原因**: 依存関係の作成中に、指定したリソースがサーバーで In Service ではありませんでした。<br><br>**処置**: 指定したサーバーでリソースを In Service にし、再度操作を実施してください。 |
| 112101 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができない。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112102 | ERROR | Cannot extend resource “%s” to server “%s” | **原因**: 指定したサーバーへ拡張したリソースを作成することができない。<br><br>**処置**: SAP ソフトウェアがターゲットサーバーに正常にインストールされ、すべての必要なファイルシステムがターゲットシステムからアクセス可能になっていることを確認してください。 |
| 112103 | ERROR | Error creating resource “%s” on server “%s” | **原因**: 指定したサーバーに指定したリソースを作成することができない。<br><br>**処置**: 指定したサーバーにSAP ソフトウェアが正常にインストールされ、すべて |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | の必要なファイルシステムがマウントされアクセス可能になっていることを確認してください。 |
| 112104 | ERROR | The extend script "%s" either does not exist or is not executable on %s. | **原因**: 指定したサーバーに指定した extend スクリプトが存在しないか実行することができない。<br><br>**処置**: すべての必要なリカバリーキットがインストールされ、extend スクリプトが実行可能か確認してください。 |
| 112106 | ERROR | Unable to create an internal SAP object for resource "%s" on %s. If the tag is correct, verify that all necessary SAP file systems are mounted and accessible. | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112112 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112115 | FATAL | Error getting resource information for resource \”$tag\” on server \”$sap::me\”. | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112119 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112123 | ERROR | Error getting resource information for resource “%s” on server “%s” | **原因**: 指定したサーバー上の指定したリソース情報を確認することができませんでした。<br><br>**処置**: 指定したサーバーに指定したリソースが存在していてすべての必要なファイルシステムがマウントされているかを確認してください。 |
| 112124 | ERROR | Resource with either matching tag “%s” or id “%s” already exists on server “%s” for App “%s” and Type “%s” | **原因**: 指定したリソースタグ名もしくはID および同一のアプリケーション、タイプをもつリソースがすでに指定したサーバーに存在しています。<br><br>**処置**: SAP インスタンスが指定したサーバーでまだLifeKeeper に保護されていないことを確認してください。まだ保護されていない場合は、別のリソースタグ名を選択してください。 |
| 112125 | ERROR | Unable to create an SAP object for resource %s on system %s. | **原因**: 指定のサーバーで指定されたリソースを示すための SAP オブジェクトを作成することができない。<br><br>**処置**: 指定したサーバーにSAP ソフトウェアが正常にインストールされ、すべての必要なファイルシステムがマウントされアクセス可能になっていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112126 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | **原因**: SAP canextend スクリプトの使用に誤りがあります。<br><br>**処置**: canextend スクリプトに対する正しいパラメーターを指定してください。: canextend <template server> <template tag> |
| 112127 | FATAL | Usage: \”$cmd $usage\” Specify the correct usage for the requested command. | **原因**: SAP delete スクリプトの使用に誤りがある。<br><br>**処置**: SAP delete スクリプトに対する正しいパラメーターを指定してください。: delete [-U] -t <tag> -i <id> |
| 112128 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | **原因**: SAP depstoextend スクリプトの使用に誤りがある。<br><br>**処置**: depstoextend スクリプトに対する正しいパラメーターを指定してください。: depstoextend <template server> <template tag> |
| 112129 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | **原因**: SAP extend スクリプトの使用に誤りがある。<br><br>**処置**: extend スクリプトに対する正しいパラメーターを指定してください。: extend <template server> <template tag> <switchback> <target tag> |
| 112130 | FATAL | Usage: \”$cmd $usage\” Specify the correct usage for the requested command. | **原因**: SAP quickCheck スクリプトの使用に誤りがある。<br><br>**処置**: quickCheck スクリプトに対する正しいパラメーターを指定してください。: quickCheck -t <tag> -i <id> |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112131 | FATAL | Usage: “%s %s” Specify the correct usage for the requested command. | 原因: SAP recover スクリプトの使用に誤りがある。<br><br>処置: recover スクリプトに対する正しいパラメーターを指定してください。: recover -d <tag> |
| 112132 | FATAL | Usage: \”$cmd $usage\”. Specify the correct usage for the requested command. | 原因: SAP remoteControl スクリプトの使用に誤りがある。<br><br>処置: remoteControl スクリプトに対する正しいパラメーターを指定してください。: remoteControl <tag> <remote instance> <remote cmd> <remote cmd option> <primary system> <primary tag> |
| 112133 | FATAL | Usage: “%s %s”. Specify the correct usage for the requested command. | 原因: SAP remove スクリプトの使用に誤りがある。<br><br>処置: remove スクリプトに対する正しいパラメーターを指定してください。: remove -t <tag> -i <id> |
| 112134 | FATAL | Usage: \”$cmd $usage\”. Specify the correct usage for the requested command. | 原因: SAP restore スクリプトの使用に誤りがある。<br><br>処置: restore スクリプトに対する正しいパラメーターを指定してください。: restore -t <tag> -i <id> |
| 112137 | ERROR | The required parameter \"parent\" was either not provided or was invalid in the $func routine on $me. | 原因: CleanUpルーチンの使用に誤りがある。<br><br>処置: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112138 | ERROR | At least one required process for instance %s was not started successfully during "%s" on server %s. Please check the LifeKeeper and system logs for additional information. | **原因**: 指定したサーバーで、指定した SAP インスタンスに対する少なくとも1つのプロセスが正常に開始されませんでした。<br><br>**処置**: LifeKeeper/システムログ, SAP トレースファイルなどから原因を調査し、操作を再度行ってください。 |
| 112140 | FATAL | The tag parameter was not specified for the internal \"$func\" routine on $me. If this was a command line operation, specify the correct parameters. Otherwise, consult the troubleshooting documentation. | **原因**: 指定したサーバーでタグパラメーターが GetLK ルーチンに指定されていません。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112141 | ERROR | Either the SID ("%s") or instance ("%s") parameter was not specified for the "%s" routine on %s. | **原因**: SID もしくは SID のどちらかが指定したサーバーの StatusSapServer ルーチン に指定されていません。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112142 | ERROR | Either the SID ("%s"), instance ("%s"), or instance number ("%s") parameter was not specified for the "%s" routine on %s. | **原因**: SID、インスタンス、インスタンス番号のいずれかが指定のサーバーの StartSapServer ルーチンに提供されていない。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112143 | ERROR | The SID, instance, or instance number parameter was not specified for the "%s" routine on %s. | **原因**: SID、インスタンス、インスタンス番号のいずれかが指定のサーバーの StopSapServer ルーチンで指定されていない。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112173 | ERROR | The file “%s” does not exist or was not readable on “%s”. Verify that the specified file exists and/or is readable. | **原因**: 指定したファイルが指定したサーバーに存在しないか、読み込み不可になっている。<br><br>**処置**: ファイルが存在しているかを確認および/もしくは権限を修正して読み込み可能な状態にしてください。 |
| 112174 | ERROR | The file “%s” does not exist or was not readable on “%s”. Verify that the specified file exists and/or is readable. | **原因**: 指定したファイルが指定したサーバーに存在しないか、読み込み不可になっています。<br><br>**処置**: ファイルが存在しているかを確認および/もしくは権限を修正して読み込み可能な状態にしてください。 |
| 112175 | ERROR | The file “%s” does not exist or was not readable on “%s”. Verify that the specified file exists and/or is readable. | **原因**: 指定したファイルが指定したサーバーに存在しないか、読み込み不可になっている。<br><br>**処置**: ファイルが存在しているかを確認および/もしくは権限を修正して読み込み可能な状態にしてください。 |
| 112194 | ERROR | There was an error verifying the NFS connections for SAP related mount points on “%s”. One or more NFS servers is not operational and needs to be restarted. | **原因**: マウントポイントが /etc/default/LifeKeeper の SAP_NFS_CHECK_DIRS エントリに記述されている少なくとも一つの 重要なNFS 共有システムが現在無効となっている。<br><br>**処置**: すべての必要な NFS 共有ファイルシステムがアクセス可能であることを確認し、現在操作不可となっている NFS サーバーを再起動してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112195 | FATAL | There was an error verifying the NFS connections for SAP related mount points on $me. One or more NFS servers is not operational and needs to be restarted. | **原因**: マウントポイントが /etc/default/LifeKeeper の SAP_NFS_CHECK_DIRS エントリに記述されている少なくとも一つの 重要なNFS 共有システムが現在無効となっている。<br><br>**処置**: すべての必要な NFS 共有ファイルシステムがアクセス可能であることを確認し、現在操作不可となっている NFS サーバーを再起動してください。 |
| 112196 | WARN | There was an error verifying the NFS connections for SAP related mount points on “%s”. One or more NFS servers is not operational and needs to be restarted. | **原因**: マウントポイントが /etc/default/LifeKeeper の SAP_NFS_CHECK_DIRS エントリに記述されている少なくとも一つの 重要なNFS 共有システムが現在無効となっている。<br><br>**処置**: すべての必要な NFS 共有ファイルシステムがアクセス可能であることを確認し、現在操作不可となっている NFS サーバーを再起動してください。 |
| 112201 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. | **原因**: リソースタグの引数が GetLKEquivルーチンに提供されていません。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112203 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. Additional information available in the LifeKeeper and system logs. | **原因**: SAP インスタンス番号が IsInstanceRunning ルーチンに提供されていません。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112204 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. | **原因**: saphostexec もしくは saposcol コ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | マンドのどちらを使用するか判断できない。<br><br>**処置**: SAP kernel 7.3より前のSAP NetWeave に対して SAP_SRVHOST_CMD, SAP_HOSTCTL_CMD, SAP_OSCOL_CMD を使用している場合は、チューニングが正しく設定されていることを確認してください。 |
| 112205 | ERROR | The internal object value "%s" was empty. Unable to complete "%s" on %s. | **原因**: SAP インスタンスが 指定したサーバーのSAPRemExec ルーチンに提供されなかった。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112206 | ERROR | The required action parameter was not provided. Unable to complete "%s" on %s. | **原因**: 指定したサーバーで SAPRemExec ルーチンに対するアクションが提供されませんでした。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112208 | ERROR | Error getting the value of "%s" or "%s" from the default profile "%s" on %s. Verify that the specified value exists. | **原因**: 指定したサーバーで提供されたプロファイルから仮想IPもしくはホスト名を取得できない。<br><br>**処置**: プロファイルに適切なエントリーが存在していることを確認してください。 |
| 112209 | FATAL | Unable to gather required information from the SAP default profile for SID $sid ($DPFL) on $me. Verify that the default profile exists and is accessible. | **原因**: 指定したサーバーで提供されたプロファイルから SAP インスタンスに関する情報を取得できない。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: SAP ソフトウェアが正常にインストールされ提供されたプロファイルが存在および読み込み可能となっていることを確認してください。 |
| 112214 | ERROR | Unable to determine the status of the path "%s" ("%s") on %s. The path on %s may require the execution of the command: "mount <ip>:<export> %s". Verify that the SAP software is correctly installed and that all SAP file systems are mounted and accessible. | **原因**: 指定したパスのファイルシステムのステータスを判断できません。<br><br>**処置**: SAP ソフトウェアが正常にインストールされすべての必要となるファイルシステムがマウントされていることを確認してください。 |
| 112219 | ERROR | [HACONNECTOR:%s] Unable to write to file "%s" on %s. If the file already exists, manually enable write permissions on it. | **原因**: 指定したファイルが指定したサーバーで読み込み/書き込み権限をもっていない。<br><br>**処置**: 指定したサーバーの読み込み/書き込み権限を有効にしてください。 |
| 112220 | ERROR | Unable to start the sapstartsrv service for SID %s and SAP instance %s on %s. Verify that the sapservices file is correct and the process can be started manually. | **原因**: 指定した SAP インスタンスに対する SAP Start Service (sapstartsrv) プロセスを開始できない。<br><br>**処置**: sapservices ファイルが sapstartsrv プロセスを開始するための適切なコマンドを含んでいて、そのプロセスを手動で開始できるかを確認してください。 |
| 112221 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance types TYPE_CS, TYPE_ERS, or TYPE_NEW_ERS (1, 2, or 5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112222 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance types TYPE_CS, TYPE_ERS, or TYPE_NEW_ERS (1, 2, or 5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112223 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_CS (1). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112224 | ERROR | The internal \"%s\" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_CS (1). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112225 | ERROR | The internal \"%s\" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_NEW_ERS (5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOSカスタマーセンターへお問い合わせください。 |
| 112226 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance types TYPE_CS, TYPE_ERS, and TYPE_NEW_ERS (1, 2, and 5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出され |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | た。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112227 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This routine only supports SAP instance type TYPE_CS (1). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112228 | ERROR | The internal "%s" routine was called for a resource with unsupported instance type %s. This method supports only SAP instance type TYPE_NEW_ERS (5). | **原因**: サポートされていないインスタンスタイプの内部SAPオブジェクトに対して、指定されたルーチンが呼び出された。<br><br>**処置**: こちらは内部エラーです。SIOS カスタマーセンターへお問い合わせください。 |
| 112229 | ERROR | The profile \"%s\" either does not exist or cannot be read on %s. Unable to determine whether enqueue replication is enabled for resource %s. Please verify that the file exists and can be read. | **原因**: 指定したサーバー上に指定したインスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>**処置**: ファイルが存在していて、読み取り可能かどうかを確認してください。 |
| 112325 | ERROR | The $SAP_CONTROL utility cannot be located. Verify that all necessary file systems are mounted and that the $SAP_CONTROL utility can be located in the $sapadmin user's PATH. | **原因**: SAP インスタンスの管理に必要な sapcontrol ユーティリティが配置できない。<br><br>**処置**: 全ての必要なファイルシステムがマウントされ、sapcontrol ユーティリティがSAP 管理ユーザーの Path に配置されていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 112326 | ERROR | The \"which $SAP_CONTROL\" command for user $sapadmin returned $sapcmd as the location of the $SAP_CONTROL utility, but the utility could not be found or was not executable in this location. Verify that all necessary file systems are mounted and that the $SAP_CONTROL utility can be located in the $sapadmin user's PATH. | 原因: SAP インスタンスの管理に必要な sapcontrol ユーティリティが配置できない。<br><br>処置: 全ての必要なファイルシステムがマウントされ、sapcontrol ユーティリティがSAP 管理ユーザーのパスに配置されていることを確認してください。 |
| 112433 | ERROR | Unsupported SAPENQ_VERSION ($enqversion) for resource $tag on $me. Unable to obtain enqueue replication status. | 原因: 指定したサーバーで指定したリソースのSAPENQ_VERSIONパラメーターにサポートされていない値が検知された。<br><br>処置: 指定したリソースの info ファイルで、SAPENQ_VERSION に有効な値 (現在使用しているエンキューサーバーのバージョンを表す1 または 2) が設定されていることを確認してください。 |
| 112437 | ERROR | Profile \"$srvpf\" not found on $me. Unable to obtain enqueue replication status for instance $inst. | 原因: 指定したサーバー上に指定したインスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>処置: ファイルが存在していて、読み取り可能かどうかを確認してください。 |
| 112438 | ERROR | Unsupported SAPENQ_VERSION ($enqversion) for resource $tag on $me. Unable to obtain enqueue replication status for instance $inst. | 原因: 指定したサーバーで指定したリソースのSAPENQ_VERSION パラメーターにサポートされていない値が検知された。<br><br>処置: 指定したリソースの info ファイルで、SAPENQ_VERSION に有効な値 (現在使用しているエンキューサーバーのバージョンを表す1 または 2) が設定されていることを確認してください。 |
| 112470 | ERROR | The instance profile %s could not be found on %s. Verify that the SAP software is installed correctly and that all | 原因: 指定したサーバー上に指定したイ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | necessary file systems are mounted and accessible. | ンスタンスプロファイルが存在していない、もしくは読み取りができない。<br><br>**処置**: ファイルが存在していて、読み取り可能かどうかを確認してください。 |
| 112490 | ERROR | [HACONNECTOR] Unable to determine the corresponding tag for resource with ID \"$res\" on $me. | **原因**: 指定したサーバー上に指定したリソースID をもつ LifeKeeper SAP リソースを見つけることができない。<br><br>**処置**: fra コマンドの —res 引数として提供されるソース ID が有効な LifeKeeper SAP リソースに対応しているか確認してください。 |
| 112507 | ERROR | [HACONNECTOR] At least one required process for the instance was not killed successfully during the fra migrate action on $me. Aborting resource migration. | **原因**: fra マイグレーション実行中に SAP インスタンスが指定したサーバーで正常に停止しなかった。<br><br>**処置**: SAP インスタンスに対して起動中のプロセスを手動で停止し、マイグレーションを再度実行してください。 |
| 112539 | ERROR | [HACONNECTOR] Unable to find rpm information for one or more packages on $me. | **原因**: HA Connector gvi (“Get Version Information”)ルーチンは、LifeKeeper および/または the SAP Recovery Kit のバージョン番号を判別することができなかった。<br><br>**処置**: rpm -q コマンドでLifeKeeper Core および SAP Recovery Kit の rpm 情報を取得できるかを確認してください。 |
| 112976 | ERROR | There is no LifeKeeper protected resource with tag $tag on system $me. | **原因**: SAP canfailover スクリプトに渡されたリソースタグに対応する LifeKeeper リソースがない。<br><br>**処置**: リソースタグ名が正しいことを確 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | 認し再度コマンドを実行してください。 |
| 112977 | ERROR | Resource $tag is not a $app/$typ resource. Please use the $ins_app/$ins_typ resource-specific canfailover script instead. | 原因: SAP canfailover スクリプトに渡された LifeKeeper リソースが、SAP（appsuite/sap）リソースではない。<br><br>処置: 指定したリソースに対して適切なタイプ固有の canfailover スクリプトを使用してください。 |

# 8.1.15. SAP HANA Recovery Kit メッセージカタログ

**Control F** を使用して、各カタログの特定のエラーコードを検索します。エラーコードの検索には、画面右上の検索ボックスをご利用ください。

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136002 | ERROR | Usage: $usage | 原因: SAP HANA createスクリプトに無効な引数が指定された。<br><br>処置: 次の形式で適切な引数を指定してください：<br><リソースタグ> <SAP SID> <HDBインスタンス> [スイッチバックタイプ] [仮想IPリソースタグ] |
| 136003 | ERROR | END failed create of resource $tag on server $me with return value of $errcode. | 原因: SAP HANAリソースの作成中に障害が発生した。<br><br>処置: SAP HANAシステムレプリケーションがプライマリとセカンダリの両方のレプリケーションサイトで完全に構成および有効化されていることを確認し、リソース作成操作を再試行してください。 |
| 136005 | ERROR | An unknown error has occurred in utility rlslocks on server $me. View the LifeKeeper logs for details and retry the operation. | 原因: SAP HANAリソースの作成中にLifeKeeper rlslocksユーティリティで問題が発生する。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース作成操作を再試行してください。 |
| 136006 | ERROR | END failed create of resource $tag on server $me with signal $sig. | 原因: SAP HANAリソース作成中のシグナルによる割込み。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136008 | ERROR | An unknown error has occurred in utility getlocks on server $me. View the LifeKeeper logs for details and retry the operation. | 原因: SAP HANAリソースの作成中にLifeKeeper getlocksユーティリティで問題が発生する。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース作成操作を再試行してください。 |
| 136009 | ERROR | The SAP HANA product was not found in the directory $obj->{'hana_util_path'} on server $me. Verify that SAP HANA is correctly installed and that all necessary file systems are mounted. | 原因: リソースの作成中に、必須のSAP HANAバイナリが見つからなかった。<br><br>処置: クラスター内のすべてのサーバーでSAP HANAが正しくインストールされ、構成されていることを確認してください。 |
| 136010 | ERROR | Failed to create resource as id $id already exists on system $me. | 原因: 指定されたIDのLifeKeeperリソースがすでにシステムに存在する。<br><br>処置: SAP HANAデータベースがLifeKeeperによってすでに保護されているかどうかを確認してください。 |
| 136011 | ERROR | Failed to create new tag $tag for SAP HANA resource on $me. | 原因: 入力されたSAP HANAリソースタグは別のLifeKeeperリソースによってすでに使用されており、LifeKeeper newtagユーティリティは新しいタグを作成できなかった。<br><br>処置: SAP HANAリソースに対し、別のタグ名を選択してください。 |
| 136012 | ERROR | Failed creation of resource with $tag on system $me. | 原因: LifeKeeperで指定したSAP HANAリソースを作成できなかった。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース作成操作を再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136014 | ERROR | Failed to create resource dependency for parent $tag and child $virtual_ip_tag. | 原因: SAP HANAリソースとそれが依存する仮想IPリソース間の依存関係の作成に失敗した。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース作成操作を再試行してください。 |
| 136015 | ERROR | The info field for resource $tag could not be successfully generated using values [SID: $info_sid, Instance: $info_instance, Replication Mode: $info_repl_mode, Site Name: $info_site_name, Operation Mode: $info_oper_mode]. Please verify that SAP HANA System Replication is fully configured and enabled on both the primary and secondary systems before creating the SAP HANA resource. | 原因: SAP HANAの「リソース情報」フィールドの作成時に無効な値が見つかった。<br><br>処置: SAP HANAが正しくインストールおよび構成されていること、SAP HANAシステムレプリケーションがクラスター内のすべてのサーバーで完全に構成され有効になっていることを確認してください。 |
| 136016 | ERROR | The selected server $me is not the primary/source system for SAP HANA System Replication for the selected SID $sid and HDB instance $instance. Please select ‘Cancel’ and start this action on the primary/source HANA System Replication system. | 原因: リソースの作成が、HANAシステムレプリケーションのセカンダリシステムで開始されている。<br><br>処置: HANAシステムレプリケーションのプライマリシステムで作成操作を開始してください。 |
| 136031 | ERROR | END failed extend of resource $target_tag on server $me with return value of $err_code. | 原因: SAP HANAリソースの拡張中に問題が発生した。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース拡張操作を再試行してください。 |
| 136033 | ERROR | Usage: $usage | 原因: SAP HANA extendスクリプトに無効な引数が指定された。<br><br>処置: 次の形式で適切な引数を指定してください： |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | <テンプレートシステム> <テンプレートタグ> <スイッチバックタイプ> <ターゲットタグ> |
| 136035 | ERROR | Template resource $template_tag on server $template_sys does not exist. | 原因: 拡張するテンプレートSAP HANAリソースがテンプレートサーバー上に存在しない。<br><br>処置: テンプレートサーバー上に、拡張されているSAP HANAリソースが存在することを確認してください。 |
| 136036 | ERROR | Resource with matching id $target_id already exists on server $me for App $app_type and Type $res_type. | 原因: 同じLifeKeeper IDのSAP HANAリソースがターゲットシステムにすでに存在する。<br><br>処置: SAP HANAデータベースが、ターゲットサーバー上のLifeKeeperによってすでに保護されているかどうかを確認してください。 |
| 136037 | ERROR | Resource with matching tag $target_tag already exists on server $me for App $app_type and Type $res_type | 原因: 同じLifeKeeperリソースタグを持つSAPリソースがターゲットサーバーに既に存在する。<br><br>処置: SAP HANAデータベースが、ターゲットサーバー上のLifeKeeperによってすでに保護されているかどうかを確認してください。 |
| 136039 | ERROR | Error creating resource $target_tag on system $me. | 原因: ターゲットサーバーで同等のSAP HANAリソースを作成できなかった。<br><br>処置: LifeKeeperログファイルで見つかった問題を解決し、リソース拡張操作を再試行してください。 |
| 136040 | ERROR | The target tag ($target_tag) and template tag ($template_tag) must be the same. | 原因: プライマリシステムで使用される |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | リソースタグ名が、セカンダリシステムで使用されるリソースタグ名と異なる。両方の名前は同じでなければならない。<br><br>**処置**: リソースの作成では、プライマリシステムタグおよびセカンダリシステムタグと同じ名前を使用してください。 |
| 136045 | ERROR | Cannot extend resource $template_tag to server $me. | **原因**: SAP HANA canextendスクリプトは、リソースを特定のターゲットシステムに拡張できないことを示している。<br><br>**処置**: LifeKeeperログファイルで見つかった問題を解決し、リソース拡張操作を再試行してください。 |
| 136047 | ERROR | Usage: $usage | **原因**: SAP HANA canextendスクリプトに無効な引数が指定された。<br><br>**処置**: 次の形式で適切な引数を指定してください：<br><テンプレートシステム> <テンプレートタグ> |
| 136048 | ERROR | Resource $template_tag does not exist on server $template_sys. | **原因**: 拡張するテンプレートSAP HANAリソースがテンプレートサーバー上に存在しない。<br><br>**処置**: テンプレートサーバー上に、拡張しているSAP HANAリソースが存在することを確認してください。 |
| 136049 | ERROR | The system user $hana_user does not exist on server $me. | **原因**: SAP HANAデータベースのSAP管理ユーザーが、指定されたサーバーに存在しない。<br><br>**処置**: SAP HANAが正しくインストールおよび構成されていることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136050 | ERROR | The user id for user $hana_user ($template_uid) on template server $template_sys is not the same as user id ($uid) on target server $me. | 原因: SAP管理ユーザーのユーザーIDが、テンプレートサーバーとターゲットサーバーで異なっている。<br><br>処置: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |
| 136051 | ERROR | The group id for user $hana_user ($template_gid) on template server $template_sys is not the same as group id ($gid) on target server $me. | 原因: SAP管理ユーザーのグループIDが、テンプレートサーバーとターゲットサーバーで異なっている。<br><br>処置: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |
| 136052 | ERROR | The home directory for user $hana_user ($template_home) on template server $template_sys is not the same as home directory ($user_home) on target server $me. | 原因: SAP管理ユーザーのホームディレクトリが、テンプレートサーバーとターゲットサーバーで異なっている。<br><br>処置: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |
| 136053 | ERROR | The SAP HANA instance $instance does not exist for $sid on server $me. | 原因: 指定されたSAP HANAデータベースインスタンスのインストールディレクトリが見つからなかった。<br><br>処置: SAPのHANAがクラスター内のすべてのサーバーに適切にインストールおよび構成されていることを確認してください。 |
| 136055 | ERROR | The SAP HANA site name $target_obj->{‘site_name’} on server $me must be different from site name $template_obj->{‘site_name’} on $template_obj->{‘sys’}. | 原因: SAP HANAシステムレプリケーションのサイト名が、プライマリサーバーとセカンダリサーバーの両方で同じにな |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | っている。<br><br>**処置**: セカンダリサーバー上のSAP HANAデータベースを停止し、hdbnsutilユーティリティを使用して、別のサイト名でセカンダリレプリケーションサイトを再登録してください。 |
| 136056 | ERROR | Unable to obtain SAP HANA System Replication parameters for database $instance on server $me. Please verify that SAP HANA System Replication is enabled and properly configured and that the database instance is running on all servers in the cluster. | **原因**: 指定されたシステム上のデータベースのSAP HANAシステムレプリケーションパラメーターを決定できなかった。<br><br>**処置**: SAP HANAシステムレプリケーションが有効で適切に構成されていること、およびデータベースがクラスター内のすべてのサーバーで実行されていることを確認してください。 |
| 136057 | ERROR | Unable to create a HANA object for database $instance on server $me. Please verify that database instance $instance is properly installed. | **原因**: 指定したサーバー上の指定したインスタンスを表す内部HANAオブジェクトを作成できない。<br><br>**処置**: データベースインスタンスが正しくインストールされており、必要なすべてのファイルシステムがマウントされていることを確認してください。 |
| 136083 | ERROR | Usage: $usage | **原因**: SAP HANA deleteスクリプトの引数が無効。<br><br>**処置**: 有効なHANA LifeKeeperリソースタグ名とリソースIDの両方を入力してください。<br>入力方法：delete -t <tag> -i <id> [-U] |
| 136160 | ERROR | Unable to create SAP HANA object. The SID and instance for the SAP HANA database must be provided. | **原因**: SAP HANAオブジェクトの作成中に、SAP SIDまたはSAP HANAインスタンス名のいずれかが欠落していた。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136161 | ERROR | Unable to create SAP HANA object. Resource system name and tag name must be provided. | **原因**: SAP HANAオブジェクトの作成中に、システム名またはリソースタグ名のいずれかが欠落していた。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136162 | ERROR | Could not find any information regarding resource $tag on $sys. | **原因**: 指定されたタグを持つSAP HANAリソースに関する情報を取得できなかった。<br><br>**処置**: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136168 | ERROR | Unable to check status of SAP Host Agent on server $self->{‘sys’}. Command \"$curr_cmd\" returned exit code $ret. | **原因**: 指定されたサーバーでSAP Host Agentプロセスのステータスを判別できなかった。<br><br>**処置**: 詳細については、SAP Host Agentトレースファイル（dev_saphostexecなど）を確認してください。 |
| 136174 | ERROR | Unable to create SAP HANA object. The SID or instance value is missing. | **原因**:SAP HANAオブジェクトの作成時に、SAP SID、SAP HANAインスタンス名、またはSAP HANAシステムレプリケーション値のいずれかが欠落していた。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136182 | ERROR | Failed to $flg_action flag \"${HANA_FLAG_DATA_OUT_OF_SYNC}_$eqv_tag{$sys}\" on server $sys. | **原因**: 指定されたサーバーで !HANA_DATA_OUT_OF_SYNC_<tag> フラグを作成または削除できなかった。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136190 | ERROR | Failed to start SAP Host Agent processes on server $self->{'sys'}. Command \"$hostagent_cmd\" returned $ret. | **原因**: 指定されたサーバーでSAP Host Agentプロセスを開始できなかった。<br><br>**処置**: 詳細については、SAP Host Agentトレースファイル（dev_saphostexecなど）を確認してください。 |
| 136191 | ERROR | Failed to start SAP OS Collector process on server $self->{'sys'}. Command \"$oscol_cmd\" returned $ret. | **原因**: 指定されたサーバーでSAP OS Collectorプロセスを開始できなかった。<br><br>**処置**: 詳細については、SAP OSコレクタートレースファイル（dev_collなど）を確認してください。 |
| 136193 | ERROR | $takeover_text of SAP HANA System Replication for SAP HANA database $self->{'instance'} failed on server $node with exit code $ret. | **原因**: SAP HANAシステムレプリケーションで、指定されたデータベースのプライマリマスターとして指定されたサーバーを登録できなかった。<br><br>**処置**: 詳細については、SAP HANAトレースファイル（例：nameserver_<hostname>.xxxxx.xxx.trc）を確認してください。 |
| 136197 | ERROR | Update of resource info field for $eqv_tag{$sys} on $sys failed with exit code $setinfo_ret. Current info: [$info]. Attempted new info: [$new_info]. | **原因**: 指定されたサーバー上の指定されたリソースの情報フィールドを更新できなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136202 | EMERG | Failed to disable Autostart for SAP HANA instance $self->{'instance'} on server $sys with exit code $remexec_ret. Please manually set \"Autostart = 0\" in the instance profile $profile on $sys. | **原因**: Autostartパラメーターの値を、指定されたサーバー上の指定されたHDBインスタンスプロファイルで変更できなかった。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: HDBインスタンスプロファイルを手動で編集し、「Autostart = 0」に設定してください。 |
| 136205 | ERROR | Failed start of SAP Start Service for SAP HANA database $instance on server $sys. | **原因**: 指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗した。<br><br>**処置**: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136208 | ERROR | LifeKeeper was unable to determine the SAP HANA System Replication mode for database $rem_obj->{‘instance’} on server $rem_obj->{‘sys’} while attempting to identify the previous primary replication site. Please resolve the issue and bring the SAP HANA resource in-service on the system where the database should be registered as primary master. | **原因**: 指定されたサーバーでSAP HANAシステムレプリケーションモードを判別できず、以前のプライマリレプリケーションサイトを特定できなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136210 | ERROR | Failed start of SAP Start Service for SAP HANA database $rem_obj->{‘instance’} on server $rem_obj->{‘sys’}. Unable to stop the database on the server where it is currently registered as primary master. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗し、その結果現在のプライマリSAP HANAシステムレプリケーションサイトでデータベースを停止できなかった。<br><br>**処置**: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136212 | ERROR | Failed stop of SAP HANA database $rem_obj->{‘instance’} on server $rem_obj->{‘sys’} where it is currently registered as primary master. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースの停止に失敗した。<br><br>**処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136217 | ERROR | Failed start of SAP HANA database $instance on server $sys. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースの起動に失敗した。<br><br>**処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |
| 136220 | ERROR | Unable to register $sys as a secondary SAP HANA System Replication site for database $instance. The host name of the current primary replication site was not provided. | **原因**: セカンダリレプリケーションサイトの登録時に、現在のプライマリSAP HANAシステムレプリケーションサイトのホスト名を入力しなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136233 | EMERG | WARNING: A temporary communication failure has occurred between servers $self->{'sys'} and $sys. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please take one of the following resource hierarchies out of service: $self->{'tag'} on $self->{'sys'} or $eqv{$sys} on $sys. The server that the resource hierarchy is taken out of service on will become the standby server for SAP HANA database $self->{'instance'}. | **原因**: 一時的な通信障害により、指定された同等のHANAリソースがそれぞれのホストサーバーで同時にIn Serviceになった。<br><br>**処置**: セカンダリレプリケーションサイトになるサーバーで、HANAリソース階層全体をOut of Serviceにします。そのサーバーでデータベースが停止されると、LifeKeeperは次のquickCheckサイクル中にデータベースをセカンダリレプリケーションサイトとして自動的に登録します。 |
| 136234 | EMERG | WARNING: SAP HANA database $self->{'instance'} is running and registered as primary master on the following servers: $primary_node_list. Manual intervention is required in order to minimize the risk of data loss. To resolve this situation, please stop database $self->{'instance'} on the intended secondary replication site by running the command \'su – $self->{'sid_admin'} -c \"$SAP_CONTROL -nr $self->{'instance_number'} -function StopWait $HANA_STOP_WAIT 5\"\' on that server, then | **原因**: 指定されたSAP HANAデータベースが、2つのクラスターサーバーで同時にプライマリマスターとして実行および登録されている。<br><br>**処置**: メッセージに示されているコマンドを使用して、スタンバイサーバー上のデータベースを停止します。データベースが停止すると、LifeKeeperは自動的にスタンバイサーバーをセカンダリレプリ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | bring resource $self->{'tag'} in-service on the intended primary replication site. Once the resource is in-service, the stopped database will automatically be re-registered as a secondary replication site. | ケーションサイトとして登録します。 |
| 136238 | ERROR | Unable to create SAP HANA object. Resource system name and tag name must be provided. | 原因: SAP HANAオブジェクトの作成時に、サーバー名またはリソースタグ名のいずれかが欠落していた。<br><br>処置: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136239 | ERROR | Failed start of SAP Start Service for SAP HANA database $obj->{‘instance’} on server $obj->{‘sys’}. Unable to determine status of the database on $obj->{‘sys’}. | 原因: 指定されたサーバー上の指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗し、その結果データベースのステータスを判別できなかった。<br><br>処置: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136242 | ERROR | Unable to create SAP HANA object. At least one of the SAP HANA System Replication values is missing. | 原因: 指定したシステム上のデータベースに対し、SAP HANAシステムレプリケーションパラメーターを特定できない。<br><br>処置: SAP HANAシステムレプリケーションが有効になっていて適切に構成されていること、データベースがクラスター内のすべてのサーバーで実行されていることを確認してください。 |
| 136243 | ERROR | Unable to locate the pingnfs utility ($pingnfs) on server $me. Please verify that this utility exists and is executable. | 原因: エクスポートされたNFS共有で利用可能なテストに使用されるpingnfsユーティリティが見つからない。<br><br>処置: pingnfsユーティリティが所定の場所に存在し、実行可能であることを確認してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| 136245 | ERROR | Critical NFS shares being exported by server $sys ($export_list) are not currently available. Please verify that the NFS server is alive and that all NFS-related services are running. | 原因: 指定した重要なNFS共有ファイルシステムが現在利用できない。<br><br>処置: ファイルシステムをエクスポートするNFSサーバーが稼働していること、および必要なNFS関連サービスがすべて実行されていることを確認してください。 |
| 136248 | ERROR | Unable to open file $crit_mount_file on server $me. Please verify that this file exists and is read-enabled. | 原因: 指定したサーバー上の重要なNFS共有のマウント情報を含むファイルを開けない。<br><br>処置: ファイルが指定した場所に存在し、読み取りが有効になっていることを確認してください。 |
| 136253 | ERROR | The SAP HANA \"takeover with handshake\" feature is available only for SAP HANA versions 2.0 SPS04 and greater. Database $self->{'instance'} cannot be resumed. | 原因: 一時停止されたデータベースの再開は、2.0 SPS04より前のSAP HANAバージョンではサポートされていない。<br><br>処置:「Takeover with Handshake」に関連する機能を使用するには、SAP HANA 2.0 SPS04以降にアップグレードしてください。 |
| 136254 | ERROR | Attempt to resume suspended primary database $self->{'instance'} on server $self->{'sys'} failed with exit code $ret. | 原因: 指定したサーバーで中断されたデータベースインスタンスを再開しようとしたら失敗した。<br><br>処置:LifeKeeperとSAPのログファイルを調べて、失敗の原因を特定します。操作を再試行するか、対応するLifeKeeperリソースを別のサーバーでIn-Serviceにしてください。 |
| 136258 | ERROR | Attempt to register server $node as the secondary SAP HANA System Replication site for database $self->{'instance'} failed with exit code $ret. | 原因: 指定したサーバーを、指定したデータベースインスタンスのセカンダリSAP HANAシステムレプリケーションサイ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | トとして登録しようとしたが失敗した。<br><br>**処置**:LifeKeeperとSAPのログファイルを調べて障害の原因を特定し、見つかった問題を手動で修正してください。SAP HANAリソースがIn-Serviceになっている間、LifeKeeperは自動的に、セカンダリHSRロールへのバックアップサーバーの登録試行を継続します。 |
| 136263 | ERROR | Usage: $usage | **原因**: SAP HANAリソースのリストアスクリプトの引数が無効。<br><br>**処置**:次の形式で適切な引数を指定してください：<br>-t <リソースタグ> -i <リソースID> |
| 136265 | ERROR | Error getting resource information for $tag on server $me. | **原因**: 指定されたサーバーの指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>**処置**:指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136266 | ERROR | The resource $tag protecting SAP HANA database $instance is not in sync. To protect the data LifeKeeper will not restore the resource on $me. Please restore the resource on the previous source server to allow the resync to complete. | **原因**: SAP HANAシステムレプリケーションは、バックアップサーバー上でデータベースリソースをIn Serviceにする前に同期されていなかった。<br><br>**処置**: 前のプライマリサーバーでSAP HANAリソースをIn Serviceにし、再同期を完了できるようにしてください。 |
| 136275 | ERROR | Failed to determine SAP HANA System Replication mode for database $instance on server $me. | **原因**: 指定されたサーバー上の指定されたデータベースのSAP HANAシステムレプリケーションモードを判定できなかった。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 詳細については、SAP HANAトレースファイル（例：nameserver_<hostname>.xxxxx.xxx.trc）およびLifeKeeperログファイルを確認してください。 |
| 136351 | ERROR | Usage: $usage | **原因**: SAP HANA resource quickCheckスクリプトの引数が無効。<br><br>**処置**: 次の形式で適切な引数を指定してください：<br>-t <リソースタグ> -i <リソースID> |
| 136353 | ERROR | Error getting resource information for $tag. | **原因**: 指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>**処置**: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136354 | EMERG | The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKeeper and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved. | **原因**: 指定されたデータベースのSAP HANAシステムレプリケーションモードが、LifeKeeperの外部で変更されている。<br><br>**処置**: プライマリマスターとして登録する必要があるサーバーで、SAP HANAリソースをIn Serviceにしてください。 |
| 136363 | EMERG | LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved. | **原因**: 指定されたサーバーでSAP HANAシステムレプリケーションモードを判定できなかった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136375 | EMERG | An NFS server exporting a critical shared file system for resource $tag is currently unavailable. Resource monitoring for $tag will be suspended until the issue is resolv | **原因**: 重要なNFS共有ファイルシステムが現在利用不可である。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | ed. | **処置**: NFSサーバーが稼働していること、および必要なNFS関連サービスがすべて実行されていることを確認してください。 |
| 136376 | EMERG | WARNING: LifeKeeper resource $tag is designed for use in situations where SAP HANA System Replication (HSR) is disabled, but HSR was found to be enabled on server $me. Please ensure that the correct LifeKeeper resource type is being used for your current SAP HANA configuration. | **原因**: 指定したLifeKeeperリソースは、HANAシステムレプリケーション（HSR）が無効になっているSAP HANAデータベース環境を保護するように設計されているが、指定したサーバーではHSRが有効になっている。<br><br>**処置**: SAP HANAの構成に応じて、正しいLifeKeeperリソースタイプが使用されていることを確認してください。HSRが無効になっているSAP HANA構成から、HSRが有効になっているSAP HANA構成に移行した場合は、対応するSAP HANAリソースをLifeKeeperで再作成する必要があります。 |
| 136377 | EMERG | SAP HANA database $instance corresponding to resource $tag is currently suspended on server $me due to actions performed outside of LifeKeeper. In order to resume the suspended database, execute the command 'su – $obj->{'sid_admin'} -c \"$obj->{'hana_util'} -sr_resumeSuspendedPrimary\"' on server $me. Once the suspended database has been resumed, LifeKeeper will automatically register the other servers in the cluster as secondary replication sites. Otherwise, please bring resource $tag in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved. | **原因**: 指定したサーバー上のSAP HANAデータベースインスタンスが、LifeKeeperの外部で行われた操作により停止された。<br><br>**処置**: 対応するLifeKeeperリソースが現在In-Service（Active）になっているサーバー上で一時停止中のデータベースを再開する場合は、メッセージに記載されているコマンドを実行してください。それ以外の場合は、目的のプライマリーレプリケーションサイトでLifeKeeper SAP HANAリソースをIn-Serviceにしてください。 |
| 136450 | ERROR | Usage: $usage | **原因**: SAP HANA removeスクリプトに無効な引数が指定されている。<br><br>**処置**: 次の形式で適切な引数を指定して |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ください：<br><テンプレートタグ> <テンプレートID> |
| 136454 | ERROR | Error getting resource information for $tag. | 原因: 指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>処置: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136456 | ERROR | Failed start of SAP Start Service for SAP HANA database $instance on server $me. | 原因: 指定されたSAP HANAデータベースのSAP Start Serviceを開始できなかった。<br><br>処置: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136459 | EMERG | WARNING: LifeKeeper resource $tag is designed for use in situations where SAP HANA System Replication (HSR) is disabled, but HSR was found to be enabled on server $me. Please ensure that the correct LifeKeeper resource type is being used for your current SAP HANA configuration. | 原因: 指定したLifeKeeperリソースは、HANAシステムレプリケーション（HSR）が無効になっているSAP HANAデータベース環境を保護するように設計されているが、指定したサーバーではHSRが有効になっている。<br><br>処置: SAP HANAの構成に応じて、正しいLifeKeeperリソースタイプが使用されていることを確認してください。HSRが無効になっているSAP HANA構成から、HSRが有効になっているSAP HANA構成に移行した場合は、対応するSAP HANAリソースをLifeKeeperで再作成する必要があります。 |
| 136462 | ERROR | Failed to remove flag \"${hana::HANA_FLAG_LEAVE_DB_RUNNING}_$tag\" on server $me. This may cause subsequent remove actions for resource $tag on server $me to unintentionally fail to stop the database. | 原因: 指定したサーバーでLifeKeeperの!volatile!hana_leave_db_running_<tag>フラグを削除できない。<br><br>処置: 指定したサーバーでコマンド「/opt/ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | LifeKeeper/bin/flg_remove -f '!volatile!hana_leave_db_running_<tag>'」を実行して、フラグを手動で削除してください。フラグが存在する間は、指定したサーバー上のSAP HANAリソースのOut-of-service操作では、保護対象のデータベースインスタンスが実行されたままになります。 |
| 136550 | ERROR | Usage: $usage | 原因: HANAリソース回復スクリプトの引数が無効。<br><br>処置: 有効なHANA LifeKeeperリソースタグ名とリソースIDの両方を入力してください。<br>入力方法：recover -d <タグ> -n <id> |
| 136555 | ERROR | Error getting resource information for $tag on server $me. | 原因: 指定されたサーバー上の指定されたHANAリソースに関する情報を取得できなかった。<br><br>処置: サーバーがオンラインで、LifeKeeperが実行されており、HANAリソースが存在することを確認してください。 |
| 136556 | EMERG | The replication mode of the SAP HANA database $instance corresponding to resource $tag was modified outside of LifeKeeper and is no longer registered as primary master on server $me. Please bring the SAP HANA resource in-service on the server where the database should be registered as primary master. Resource monitoring for $tag will be suspended until the issue is resolved. | 原因: 指定されたデータベースのSAP HANAシステムレプリケーションモードが、LifeKeeperの外部で変更されている。<br><br>処置: プライマリマスターとして登録する必要があるサーバーで、SAP HANAリソースをIn Serviceにしてください。 |
| 136558 | EMERG | LifeKeeper was unable to determine the SAP HANA System Replication mode for database $instance corresponding to resource $tag on server $me. Resource monitoring for $tag will be suspended until the issue is resolved. | 原因: 指定されたサーバー上の指定されたデータベースに対し、SAP HANAシステムレプリケーションモードを決定でなかった。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |
| 136559 | ERROR | Resource $tag is no longer ISP on server $me. Exiting $cmd for $tag. | **原因**: 指定されたSAP HANAリソースが、指定されたサーバー上でISPではなくなった。<br><br>**処置**: 詳細については、LifeKeeperログファイルを確認してください。 |
| 136650 | ERROR | Usage: $usage | **原因**: SAP HANA hana_stop_all_dbs スクリプトに無効な引数が指定されている。<br><br>**処置**: 次の形式で適切な引数を指定してください：<br>hana_stop_all_dbs -t <タグ> |
| 136654 | ERROR | Error getting resource information for $tag. | **原因**: 指定されたSAP HANAリソースに関する情報を取得できなかった。<br><br>**処置**: 指定されたタグを持つSAP HANAリソースが指定されたサーバーに存在することを確認してください。 |
| 136658 | ERROR | Failed start of SAP Start Service for SAP HANA database $x->{'instance'} on server $x->{'sys'}. Could not determine status of SAP HANA DB on $x->{'sys'}. | **原因**: 指定されたSAP HANAデータベースのSAP Start Serviceの開始に失敗した。<br><br>**処置**: 詳細については、SAP Start Serviceトレースファイル（sapstartsrv.logなど）を確認してください。 |
| 136661 | ERROR | Failed stop of SAP HANA database $x->{'instance'} on server $x->{'sys'}. | **原因**: 指定されたサーバー上の指定されたSAP HANAデータベースの停止に失敗した。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: 詳細については、SAP HANAトレースファイルとLifeKeeperログファイルを確認してください。 |
| 136673 | ERROR | Usage: $usage | **原因**: SAP HANAのhana_takeover_with_handshakeスクリプトで無効な引数が指定された。<br><br>**処置**: 次の形式で適切な引数を指定してください： -t <tag> [-s <target server>] [-b] |
| 136674 | ERROR | Unable to remove the \"${hana::HANA_FLAG_LEAVE_DB_RUNNING}_$tag\" flag on server $isp_node. This may cause subsequent remove actions for resource $tag on server $isp_node to unexpectedly fail to stop the database. | **原因**: 指定したサーバーでLifeKeeperの!volatile!hana_leave_db_running_<tag>フラグを削除できない。<br><br>**処置**: 指定したサーバーでコマンド「/opt/LifeKeeper/bin/flg_remove -f '!volatile!hana_leave_db_running_<tag>'」を実行して、フラグを手動で削除してください。フラグが存在する間は、指定したサーバー上のSAP HANAリソースのOut-of-service操作では、保護対象のデータベースインスタンスが実行されたままになります。 |
| 136675 | ERROR | Script $cmd exited unexpectedly due to signal \"$sig\" on server $me. This may leave the $tag resource hierarchy as well as SAP HANA System Replication in an unexpected state. Please verify that the cluster resources are in the expected state. | **原因**: hana_takeover_with_handshakeスクリプトが、指定したサーバー上の指定したシグナルにより、予期せず終了した。<br><br>**処置**: SAP HANAクラスタリソースが期待通りの状態であることを確認します。そうでない場合は、見つかった問題を修正し、目的のプライマリーサーバーでSAP HANAリソース階層をIn-Serviceにします。 |
| 136677 | ERROR | Unable to find equivalent SAP HANA resource corresponding to $tag on server $target_node. | **原因**: 指定したサーバー上の指定したリ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ソースに対応する同等のSAP HANAリソースが見つからない。<br><br>**処置**: 指定したリソースタグが正しく、リソースが指定したターゲットサーバーに拡張されており、LifeKeeper がターゲットサーバー上で実行されて完全に初期化されていることを確認してください。 |
| 136678 | ERROR | Unable to obtain information about equivalent SAP HANA resource $tag on server $target_node. Verify that LifeKeeper is running and fully initialized. | **原因**: 指定したサーバー上の指定したリソースに対応する同等のSAP HANAリソースが見つからない。<br><br>**処置**: 指定したリソースタグが正しく、リソースが指定したターゲットサーバーに拡張されており、LifeKeeper がターゲットサーバー上で実行されて完全に初期化されていることを確認してください。 |
| 136679 | ERROR | Resource $tag is not a SAP HANA resource. | **原因**: 指定したリソースがSAP HANA（データベース/hana）リソースではない。<br><br>**処置**: リソースタグが正しいことを確認してください。 |
| 136680 | ERROR | Resource $tag is designed for use in an environment where SAP HANA System Replication is disabled. Takeover with handshake cannot be performed for this resource type. Please use the standard \"In Service…\" command instead. | **原因**:「Takeover with Handshake」に関連する機能は、SAP HANAシステムレプリケーションが有効な環境でのみ使用できる。<br><br>**処置**: 標準の「In Service…」オプションを使用して、SAP HANAリソースをIn-Serviceにしてください。 |
| 136681 | ERROR | SAP HANA resource $tag is not currently in-service on any server in the cluster. The resource must be in-service and SAP HANA System Replication must be in-sync | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAリソースがク |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | before performing a takeover with handshake. | ラスター内のどのサーバーでもIn-Serviceになっていない。<br><br>**処置**: 目的のプライマリーサーバーでSAP HANAリソースをIn-Serviceにしてください。 |
| 136682 | ERROR | SAP HANA resource $tag is currently in-service on multiple servers: $isp_node_list. The resource must be in-service on only one server and SAP HANA System Replication must be in-sync before performing a takeover with handshake. | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAリソースがクラスター内の複数のサーバーでIn-Serviceになっている。<br><br>**処置**: プライマリーマスターとして登録する予定のサーバーを除くすべてのサーバーで、リソースをOut-of-serviceにしてください。 |
| 136684 | ERROR | Unable to create internal SAP HANA object for resource $tag on server $target_node. Verify that all necessary file systems are mounted and that LifeKeeper is running and fully initialized on $target_node. | **原因**: 指定したサーバー上の指定したインスタンスを表す内部HANAオブジェクトを作成できない。<br><br>**処置**: データベースインスタンスが正しくインストールされ、必要なファイルシステムがすべてマウントされていることを確認してください。 |
| 136685 | ERROR | Takeover with handshake is only supported in SAP HANA versions 2.0 SPS04 and greater. The SAP HANA software must be upgraded in order to use this feature. Please use the standard \"In Service…\" command instead. | **原因**: 基礎となるSAP HANAデータベースのバージョンが2.0 SPS04未満の場合、「Takeover with Handshake」機能は使用できない。<br><br>**処置**:「Takeover with Handshake」機能を使用するには、SAP HANA 2.0 SPS04以降にアップグレードしてください。 |
| 136686 | ERROR | Takeover with handshake cannot be performed for database $target_obj->{'instance'} on server $target_node because the database is not currently running and registered as primary on any other server in the cluster. | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAデータベースインスタンスがクラスター内のどのサーバーでも実行されておらず、プライマリ |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | ーマスターとして登録されていない。<br><br>**処置**: 目的のプライマリーサーバーで対応するSAP HANAリソースをIn-Serviceにしてください。 |
| 136687 | ERROR | SAP HANA database $target_obj->{'instance'} is running and registered as primary on more than one server in the cluster. Please resolve this situation and reattempt the takeover. | **原因**:「Takeover with Handshake」の試行中、指定したSAP HANAデータベースインスタンスが複数のサーバーでプライマリーとして実行および登録されている。<br><br>**処置**: データベースインスタンスがすでに実行されており、対象のプライマリーサーバーにプライマリーとして登録されている場合は、対応するLifeKeeperリソースをそのサーバーでIn-Serviceにします。そうでない場合は、LifeKeeperで現在In-Serviceになっているサーバーを除くすべてのサーバーでデータベースインスタンスを停止し、LifeKeeperがシステムレプリケーションを再開できるようにしてから、テイクオーバーを再試行してください。 |
| 136688 | ERROR | The ${hana::HANA_FLAG_DATA_OUT_OF_SYNC}_$tag flag is set on server $target_node, indicating that SAP HANA System Replication is not currently in-sync. Please allow the replication to synchronize before reattempting the takeover. | **原因**: 指定したサーバーで「!HANA_DATA_OUT_OF_SYNC_<tag>」フラグが設定され、「Takeover with Handshake」の試行中にSAP HANAシステムレプリケーションが同期していないことを示している。<br><br>**処置**: システムレプリケーションが完全に同期するのを待ってから、テイクオーバーを再試行してください。 |
| 136689 | ERROR | Unable to create internal SAP HANA object for resource $tag on server $isp_node. Verify that all necessary file systems are mounted and that LifeKeeper is running and fully initialized on $isp_node. | **原因**: 指定したサーバー上の指定したインスタンスを表す内部HANAオブジェクトを作成できない。 |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | **処置**: データベースインスタンスが正しくインストールされ、必要なファイルシステムがすべてマウントされていることを確認してください。 |
| 136690 | ERROR | Unable to set the ${hana::HANA_FLAG_LEAVE_DB_RUNNING}_$tag flag on server $isp_node. Aborting takeover with handshake attempt for database $target_obj->{'instance'} on server $target_node. | **原因**:「Takeover with Handshake」の試行中に、指定したサーバーで、LifeKeeperフラグ「!volatile!hana_leave_db_running_<tag>」を設定できなかった。<br><br>**処置**: 詳細については、LifeKeeper のログファイルを確認してください。見つかった問題を修正し、テイクオーバーを再試行してください。 |
| 136692 | ERROR | Takeover with handshake for database $target_obj->{'instance'} failed on server $target_node. | **原因**: 指定したターゲットサーバー上の指定したSAP HANAデータベースインスタンスに対する「Takeover with Handshake」の試行が失敗した。<br><br>**処置**: /etc/default/LifeKeeper で HANA_HANDSHAKE_TAKEOVER_FAILBACK=true が指定されている場合、SAP HANAリソース階層は以前のデータベースホストサーバーで自動的にIn-Serviceに戻ります。そうでない場合は、SAP HANAリソース階層を対象のプライマリーサーバーで手動でIn-Serviceにする必要があります。 |
| 136695 | ERROR | Resource $tag does not exist on server $me. | **原因**: 指定したリソースが指定したサーバーに存在しない。<br><br>**処置**: リソースタグが正しいことを確認してください。 |
| 136696 | ERROR | Unable to verify the status of resource $tag on server $target_node. Assuming that it is not in-service. | **原因**:「Takeover with Handshake」の成否の確認中に、指定したサーバー上の指定したリソースのステータスを判別でき |

| コード | 重大度 | メッセージ | 原因/処置 |
|---|---|---|---|
| | | | なかった。<br><br>**処置**: 指定したサーバー上でLifeKeeperが実行され、完全に初期化されていること、およびローカルサーバーとターゲットサーバー間のコミュニケーションパスがアクティブであることを確認してください。/etc/default/LifeKeeper で HANA_HANDSHAKE_TAKEOVER_FAILBACK=true が指定されている場合、SAP HANAリソース階層は自動的に以前のデータベースホストサーバーでIn-Serviceになります。そうでない場合は、SAP HANAリソース階層を対象のプライマリーサーバー上で手動でIn-Serviceにする必要があります。 |
| 136697 | ERROR | Resource $res was not successfully brought in-service on server $target_node. | **原因**:「Takeover with Handshake」の試行中に、指定したリソースが指定したサーバーでIn-Serviceにならなかった。<br><br>**処置**: etc/default/LifeKeeper で HANA_HANDSHAKE_TAKEOVER_FAILBACK=true に設定されている場合、SAP HANAリソース階層は自動的に以前のデータベースホストサーバーでサービスをIn-Serviceにします。そうでない場合は、SAP HANAリソース階層を対象のプライマリーサーバーで手動でIn-Serviceにする必要があります。 |
| 136698 | ERROR | LifeKeeper is not running or is not fully initialized on server $me. | **原因**:「Takeover with Handshake」の試行中に、LifeKeeperが指定したサーバーで実行されていないか、完全に初期化されていない。<br><br>**処置**: LifeKeeperを /opt/LifeKeeper/bin/lkstart で起動するか、完全に初期化してから、テイクオーバーを再試行してください。 |

| コード | 重大度 | メッセージ | 原因/処置 |
| --- | --- | --- | --- |
| 136699 | ERROR | Unknown server $target_node. | **原因**: 指定したサーバーのホスト名が認識されない。<br><br>**処置**: サーバーのホスト名が正しく、ローカルサーバーとターゲットサーバーの間でコミュニケーションパスが作成されていることを確認してください。 |

# 9. SIOS Protection Suite for Linux サポートマトリックス

サポート対象のオペレーティングシステム
サポート対象のアプリケーション
サポート対象の仮想化環境

## サポート対象のオペレーティングシステム

| 製品 | サポート対象のオペレーティングシステム | SIOS Protection Suite for Linux / LifeKeeper for Linux | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | v9.2 | v9.2.1 | v9.2.2 | v9.3 | v9.3.1 | v9.3.2 | v9.4 | v9.4.1 | v9.5.0 | v9.5.1 | v9.5.2 |
| SIOS Protection Suite for Linux | Red Hat Enterprise Linux 5 | 5.0 ～ 5.11<br>64 ビット | 5.0 ～ 5.11<br>64 ビット | 5.0 ～ 5.11<br>64 ビット | | | | | | | | |
| | Red Hat Enterprise Linux 6 | 6.0 ～ 6.9<br>64 ビット<br>(*1) | 6.0 ～ 6.9<br>64 ビット<br>(*1) | 6.0 ～ 6.9<br>64 ビット<br>(*1) | 6.0 ～ 6.9<br>64 ビット<br>(*1) | 6.0 ～ 6.10<br>64 ビット<br>(*1) | 6.0 ～ 6.10<br>64 ビット<br>(*1) | 6.0 ～ 6.10<br>64 ビット<br>(*1) | 6.0 ～ 6.10<br>64 ビット<br>(*1) | 6.0 ～ 6.10<br>64 ビット<br>(*1) | 6.0 ～ 6.10<br>64 ビット<br>(*1) | |
| | Red Hat Enterprise Linux 7<br>(*18) | 7.0 ～ 7.4<br>64 ビット<br>(*2) | 7.0 ～ 7.4<br>64 ビット<br>(*2) | 7.0 ～ 7.4<br>64 ビット<br>(*2) | 7.0 ～ 7.5<br>64 ビット<br>(*2) | 7.0 ～ 7.6<br>64 ビット<br>(*2) | 7.0 ～ 7.6<br>64 ビット<br>(*2) | 7.0 ～ 7.7<br>64 ビット<br>(*2)<br>(*3) | 7.0 ～ 7.7<br>64 ビット<br>(*2) | 7.0 ～ 7.8<br>64 ビット<br>(*2)<br>(*13) | 7.0 ～ 7.9<br>64 ビット<br>(*2)<br>(*14) | 7.0 ～ 7.9<br>64 ビット<br>(*2) |
| | Red Hat Enterprise Linux 8<br>(*18) | | | | | | | 8.0<br>(*4) | 8.0<br>(*4) | 8.0, 8.1<br>(*4) | 8.0 ～ 8.3<br>(*4) | 8.0 ～ 8.4<br>(*4) |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | (*5) | (*5) | (*5) | (*5)<br>(*15)<br>(*16) | (*5) |
| | SUSE Linux Enterprise Server (SLES) 11<br>(*6) | 11.0 ～ SP4<br>64 ビット | 11.0 ～ SP4<br>64 ビット | 11.0 ～ SP4<br>64 ビット | 11.0 ～ SP4<br>64 ビット | 11.0 ～ SP4<br>64 ビット | 11.0 ～ SP4<br>64 ビット | 11.0 ～ SP4<br>64 ビット | 11.0 ～ SP4<br>64 ビット | | | |
| | SUSE Linux Enterprise Server (SLES) 12<br>(*7) (*18) | 12SP1 ～ SP2<br>64 ビット | 12SP1 ～ SP3<br>64 ビット | 12SP1 ～ SP3<br>64 ビット | 12SP1 ～ SP3<br>64 ビット | 12SP1 ～ SP3<br>64 ビット | 12SP1 ～ SP4<br>64 ビット | 12SP1 ～ SP4<br>64 ビット | 12SP1 ～ SP4<br>64 ビット | 12SP1 ～ SP5<br>64 ビット | 12SP1 ～ SP5<br>64 ビット<br>(*16) | 12SP1 ～ SP5<br>64 ビット |
| | SUSE Linux Enterprise Server (SLES) 15<br>(*18) | | | | | | 15.0<br>64 ビット<br>(*5)<br>(*8) | 15.0<br>64 ビット<br>(*5)<br>(*8) | 15.0 ～ SP1<br>64 ビット<br>(*5)<br>(*8) | 15.0 ～ SP1<br>64 ビット<br>(*5)<br>(*8) | 15.0 ～ SP2<br>64 ビット<br>(*5)<br>(*8)<br>(*16) | 15.0 ～ SP2<br>64 ビット<br>(*5)<br>(*8) |
| | Oracle Enterprise Linux 5 | 5.0 ～ 5.11<br>64 ビット | 5.0 ～ 5.11<br>64 ビット | 5.0 ～ 5.11<br>64 ビット | | | | | | | | |
| | Oracle Linux 6 | 6.3 ～ | 6.3 ～ | 6.3 ～ | 6.3 ～ | 6.3 ～ | 6.3 ～ | 6.3 ～ | 6.3 ～ | 6.3 ～ | 6.3 ～ | |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 6.9 (UEK R3、R4含む)<br>64ビット | 6.9 (UEK R3、R4含む)<br>64ビット | 6.9 (UEK R3、R4含む)<br>64ビット | 6.9 (UEK R3、R4含む)<br>64ビット | 6.10 (UEK R3、R4含む)<br>64ビット | 6.10 (UEK R3、R4含む)<br>64ビット | 6.10 (UEK R3、R4含む)<br>64ビット | 6.10 (UEK R3、R4含む)<br>64ビット | 6.10 (UEK R3、R4含む)<br>64ビット | 6.10 (UEK R3、R4含む)<br>64ビット | |
| | Oracle Linux 7<br>(*18) | 7.0 ~ 7.3 (UEK R3, R4含む)<br>64ビット | 7.0 ~ 7.4 (UEK R3, R4含む)<br>64ビット<br>(*2) | 7.0 ~ 7.4 (UEK R3, R4含む)<br>64ビット<br>(*2) | 7.0 ~ 7.5 (UEK R3, R4含む)<br>64ビット<br>(*2)<br>(*9) | 7.0 ~ 7.5 (UEK R3, R4含む)<br>64ビット<br>(*2)<br>(*9) | 7.0 ~ 7.6 (UEK R3, R4含む)<br>64ビット<br>(*2)<br>(*9) | 7.0 ~ 7.6 (UEK R3, R4, R5 含む)<br>64ビット<br>(*2) | 7.0 ~ 7.7 (UEK R3, R4, R5 含む)<br>64ビット<br>(*2) | 7.0 ~ 7.8 (UEK R3, R4, R5 含む)<br>64ビット<br>(*2)<br>(*13) | 7.0 ~ 7.9 (UEK R3, R4, R5 含む)<br>64ビット<br>(*2)<br>(*14)<br>(*16) | 7.0 ~ 7.9 (UEK R3, R4, R5, R6 含む)<br>64ビット<br>(*2)<br>(*17) |
| | Oracle Linux 8<br>(*18) | | | | | | | | | 8.0, 8.1<br>(*11) | 8.0 ~ 8.2 (UEK R6 は含まない)<br>(*5)<br>(*11)<br>(*16) | 8.0 ~ 8.3 (UEK R6 含む)<br>(*5)<br>(*11)<br>(*17) |
| | CentOS 5 | 5.0 ~ 5.11<br>64ビット | 5.0 ~ 5.11<br>64ビット | 5.0 ~ 5.11<br>64ビット | | | | | | | | |
| | CentOS 6 | 6.0 ~ 6.9 | 6.0 ~ 6.9 | 6.0 ~ 6.9 | 6.0 ~ 6.9 | 6.0 ~ 6.10 | 6.0 ~ 6.10 | 6.0 ~ 6.10 | 6.0 ~ 6.10 | 6.0 ~ 6.10 | 6.0 ~ 6.10 | |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | (*10) | 64ビット | 64ビット | 64ビット | 64ビット | 64ビット | 64ビット | 64ビット | 64ビット | 64ビット | 64ビット | |
| | CentOS 7 (*18) | 7.0 ～ 7.3 64ビット | 7.0 ～ 7.4 64ビット (*2) | 7.0 ～ 7.4 64ビット (*2) | 7.0 ～ 7.5 64ビット (*2) | 7.0 ～ 7.5 64ビット (*2) | 7.0 ～ 7.6 64ビット (*2) | 7.0 ～ 7.6 64ビット (*2) | 7.0 ～ 7.7 64ビット (*2) | 7.0 ～ 7.8 64ビット (*2) (*13) | 7.0 ～ 7.9 64ビット (*2) (*14) | 7.0 ～ 7.9 64ビット (*2) |
| | CentOS 8 (*18) | | | | | | | | | 8.0, 8.1 (*12) | 8.0 ～ 8.2 (*5) (*12) (*16) | 8.0 ～ 8.3 (*5) (*12) |

***1** 6.0 は推奨されません
***2** 一部のKernel versionでは非同期モードはサポートされません。
***3** 7.7でDataKeeperを使用する場合、LifeKeeperインストール時に 特定の手順 に従う必要があります。
***4** RHEL 7 から RHEL 8 のようにKernelをメジャーバージョンアップすることはサポートされません。
***5** DataKeeper においてセクターサイズが奇数のディスクは利用できません。
***6** *SLES11SP2はカーネルを3.0.42-0.7.3にアップグレードしてください。*
***7** *SLES12SP3はカーネルを4.4.82-6.9.1にアップグレードしてください。*
***8** SLES12からSLES15のようにKernelをメジャーバージョンアップすることはサポートされません。
***9** 7.5/7.6 では UEK R5 はサポートされません。
***10** *CentOS 6.0 – DataKeeper 設定はサポートされません。この制限は、各バージョンの過去のリリースノートに記載されていません。*
***11** OEL 7 から OEL 8 のようにKernelをメジャーバージョンアップすることはサポートされません。
***12** CentOS 7 から CentOS 8 のようにKernelをメジャーバージョンアップすることはサポートされません。
***13** 7.8でDataKeeperを使用する場合、LifeKeeperインストール時に 特定の手順 に従う必要があります。
***14** 7.9でDataKeeperを使用する場合、LifeKeeperインストール時に 特定の手順 に従う必要があります。
***15** 8.3でDataKeeperを使用する場合、LifeKeeperインストール時に 特定の手順 に従う必要があります。
***16** 重要 :いくつかの Linux リリースの md/raid1カーネルモジュールで発見された問題があります。この問題により、DataKeeperクラスターノードを使用しているSIOS Protection Suite for Linuxのサポートされているバージョン（v9.3.2～v9.5.1）の部分的な再同期では、すべてのブロックが再同期されない場合があります。影響を受けるディストリビューションおよびカーネルバージョンなどの詳細については こちら をご確認ください。
***17** UEK R6はカーネルを 5.4.17-2102.202.5 以降にアップグレードしてください。
***18** 非同期モードで作成し同期モードで拡張した DataKeeper リソースの構成はサポートされません。

# サポート対象のアプリケーション

| 製品 | サポート対象の アプリケーション | SIOS Protection Suite for Linux / LifeKeeper for Linux | | | | | |
|---|---|---|---|---|---|---|---|
| | | **v9.2** | **v9.2.1** | **v9.2.2** | **v9.3** | **v9.3.1** | **v9.3.2** |
| **Apache ARK** | Apache Web Server | 2.0, 2.2, 2.4 | 2.0, 2.2, 2.4 | 2.0, 2.2, 2.4 | 2.4 | 2.4 | 2.4 |
| **SAP ARK**<br><br>**(SAP** がサポートする **OS** につきましては **SIOS Protection Suite for SAP Solution Page** を参照してください。**)** | SAP NetWeaver | 7.0 including Enhancement Package 1-3,<br><br>7.3 including Enhancement Package 1, 7.4, 7.5 | 7.0 including Enhancement Package 1-3,<br><br>7.3 including Enhancement Package 1, 7.4, 7.5 | 7.0 including Enhancement Package 1-3,<br><br>7.3 including Enhancement Package 1, 7.4, 7.5 | 7.0 including Enhancement Package 1-3,<br><br>7.3 including Enhancement Package 1, 7.4, 7.5 | 7.0 including Enhancement Package 1-3,<br><br>7.3 including Enhancement Package 1, 7.4, 7.5 | 7.0 inc Enhan Packa<br><br>7.3 inc Enhan Packa 7.4, 7. |
| | SAP NetWeaver AS for ABAP | 7.51 innovation package | 7.51 innovation package | 7.51 innovation package | 7.51 innovation package | 7.51 innovation package | 7.51 innova packag innova packag |
| | SAP S/4HANA Platform | | | | | | |
| **SAP MaxDB ARK** | SAP MaxDB | 7.5, 7.6, 7.7, 7.8, 7.9 | 7.5, 7.6, 7.7, 7.8, 7.9 | 7.5, 7.6, 7.7, 7.8, 7.9 | 7.6, 7.7, 7.8, 7.9 | 7.6, 7.7, 7.8, 7.9 | 7.6, 7. 7.9 |
| **SAP HANA ARK** | SAP HANA | | | | | | |
| **Postfix ARK** | Postfix | (**1) | (**1) | (**1) | (**1) | (**1) | (**1) |
| **DB2 ARK** | IBM Db2 Enterprise Server Edition (ESE) および Workgroup Server Edition (WSE) | 9.0, 9.5, 9.7, 10.1, 10.5, 11.1 | 9.0, 9.5, 9.7, 10.1, 10.5, 11.1 | 9.0, 9.5, 9.7, 10.1, 10.5, 11.1 | 10.5, 11.1 | 10.5, 11.1 | 10.5, 1 |
| | IBM Db2 Express Edition | 9.0, 9.5, 9.7, 10.1, 10.5, 11.1 | 9.0, 9.5, 9.7, 10.1, 10.5, 11.1 | 9.0, 9.5, 9.7, 10.1, 10.5, 11.1 | 10.5, 11.1 | 10.5, 11.1 | 10.5, 1 |
| | IBM Db2 | | | | | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | Advanced, Standard and Community Editions | | | | | | |
| | IBM Db2 Advanced Workgroup Server Edition | | | | | | |
| | IBM Db2 Advanced Enterprise Server Edition | | | | | | |
| **PostgreSQL ARK** | PostgreSQL | 8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6 | 8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6 | 8.3, 8.4, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6, 10 | 9.3, 9.4, 9.5, 9.6,10 | 9.3, 9.4, 9.5, 9.6,10 | 9.3, 9.4 9.6,10, |
| | EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server | 8.3, 8.4, 9.1, 9.2, 9.3, 9.4, 9.5 | 8.3, 8.4, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6 | 8.3, 8.4, 9.1, 9.2, 9.3, 9.4, 9.5, 9.6,10 | 9.3, 9.4, 9.5, 9.6,10 | 9.3, 9.4, 9.5, 9.6,10 | 9.3, 9.4 9.6,10, |
| | EnterpriseDB Postgres Plus Solution Pack | 9.1, 9.2, 9.3 | 9.1, 9.2, 9.3 | 9.1, 9.2, 9.3 | | | |
| | EnterpriseDB Postgres Plus Standard Server | 8.4, 9.0 | 8.4, 9.0 | 8.4, 9.0 | | | |
| | Symfoware Server Enterprise Edition | 12.2, 12.3 | 12.2, 12.3 | 12.2, 12.3 | 12.2, 12.3 | 12.2, 12.3 | 12.2, 1 |
| | Symfoware Server Standard | 12.2, 12.3 | 12.2, 12.3 | 12.2, 12.3 | 12.2, 12.3 | 12.2, 12.3 | 12.2, 1 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | Edition | | | | | | |
| | Symfoware Server Lite Edition | 12.3 | 12.3 | 12.3 | 12.3 | 12.3 | 12.3 |
| | FUJITSU Software Enterprise Postgres Advanced Edition | 9.5 | 9.5 | 9.5 | 9.5 | 9.5 | 9.5, 10 |
| | FUJITSU Software Enterprise Postgres Standard Edition | 9.5, 9.6 | 9.5, 9.6 | 9.5, 9.6 | 9.5, 9.6 | 9.5, 9.6 | 9.5, 9.6 |
| | FUJITSU Software Enterprise Postgres Community Edition | | | | | | 10 |
| | PowerGres Plus | | | | | | |
| | PowerGres on Linux | | | | | | |
| **Oracle ARK** | Oracle Database Enterprise Edition | 10g R2, 11g, 11g R2, 12c, 12c R2 | 10g R2, 11g, 11g R2, 12c, 12c R2 | 10g R2, 11g, 11g R2, 12c, 12c R2 | 11g R2, 12c, 12c R2 | 11g R2, 12c, 12c R2, 18c | 11g R2 12c R2 19c |
| | Oracle Database Standard Edition | 10g R2, 11g, 11g R2, 12c | 10g R2, 11g, 11g R2, 12c | 10g R2, 11g, 11g R2, 12c | 11g R2 | 11g R2 | 11g R2 |
| | Oracle Database Standard Edition One | 10g R2, 11g, 11g R2, 12c | 10g R2, 11g, 11g R2, 12c | 10g R2, 11g, 11g R2, 12c | 11g R2 | 11g R2 | 11g R2 |
| | Oracle Database Standard Edition 2 | 12c, 12c R2 | 12c, 12c R2 | 12c, 12c R2 | 12c, 12c R2 | 12c, 12c R2c 18c | 12c, 12 18c 19 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Sybase ASE ARK** | SAP Adaptive Server Enterprise | 15.5, 15.7, 16.0 | 15.5, 15.7, 16.0 | 15.5, 15.7, 16.0 | 15.7, 16.0 | 15.7, 16.0 | 15.7, 1 |
| **Samba ARK** | Standard Samba file shares | (**1) | (**1) | (**1) | (**1) | (**1) | (**1) |
| **NFS Server ARK** | Linux kernel version | Kernel 2.6 以降<br>(**2) | Kernel 2.6 以降<br>(**2) | Kernel 2.6 以降<br>(**2) | Kernel 2.6 以降<br>(**2) | Kernel 2.6 以降<br>(**2) | Kernel 降<br>(**2) |
| **NAS ARK** | NFS version of Mounted NFS file systems from an NFS server or Network Attached Storage (NAS) device | NFSv2<br>(**2)<br>NFSv3<br>NFSv4 | NFSv2<br>(**2)<br>NFSv3<br>NFSv4 | NFSv2<br>(**2)<br>NFSv3<br>NFSv4 | NFSv2<br>(**2)<br>NFSv3<br>NFSv4 | NFSv2<br>(**2)<br>NFSv3<br>NFSv | NFSv2<br>(**2)<br>NFSv3<br>NFSv4 |
| **MySQL ARK** | MySQL Community Edition, MySQL Enterprise Edition | 5.1, 5.5, 5.6, 5.7 | 5.1, 5.5, 5.6, 5.7 | 5.1, 5.5, 5.6, 5.7 | 5.5, 5.6, 5.7 | 5.5, 5.6, 5.7, 8.0 | 5.5, 5.6 8.0 |
| | MariaDB | 5.5, 10.0 | 5.5, 10.0 | 5.5, 10.0 | 5.5, 10.0 | 5.5, 10.0 | 5.5, 10 |
| **WebSphere MQ ARK** | WebSphere MQ/IBM MQ | 7.1, 7.5, 8.0, 9.0<br>(**5) | 7.1, 7.5, 8.0, 9.0<br>(**5) | 7.1, 7.5, 8.0, 9.0<br>(**5) | 7.5, 8.0, 9.0 | 7.5, 8.0, 9.0 | 7.5, 8.0 9.1 |
| **Software RAID (md) ARK** | RHEL5, OL5, CentOS5 | 5.0 ~ 5.11 | 5.0 ~ 5.11 | 5.0 ~ 5.11 | | | |
| | RHEL6, OL6, CentOS6 | 6.0 ~ 6.7 | 6.0 ~ 6.7 | 6.0 ~ 6.7 | 6.0 ~ 6.7 | 6.0 ~ 6.7 | 6.0 ~ |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | SLES11 | 11.0 ~ SP1 | 11.0 ~ SP1 | 11.0 ~ SP1 | 11.0 ~ SP1 | 11.0 ~ SP1 | 11.0 ~ |
| **VMDK as Shared Storage ARK** | VMware vSphere | | | | | | |

**1 サポート対象の Linux ディストリビューションに付属
**2 NFSv2はRHEL7/ CentOS7/ OL7以降ではサポートされません。
**3 NFS over UDPはRHEL8以降ではサポートされません。
**4 10.1、10.2を除く
**5 9.0.5 はサポートされません。
**6 RHEL 6.x/CentOS 6.x/OEL 6.x および SLES11.x をサポートしていません。
**7 SSPではサポートしていません。
**8 VMDK ARKでリソースの作成する前に別途手順が必要です。詳細については こちら を確認してください。
**9 サービスの停止を伴わないLifeKeeperの停止を実施した場合、システム全体が停止する可能性があります。詳細については こちら をご確認ください。

## サポート対象の仮想化環境

| 製品 | サポート対象の仮想化環境 | SIOS Protection Suite for Linux / LifeKeeper for Linux | | | | | |
|---|---|---|---|---|---|---|---|
| | | v9.2 | v9.2.1 | v9.2.2 | v9.3 | v9.3.1 | v9.3.2 |
| SIOS Protection Suite for Linux | VMware vSphere | 4.0, 4.1, 5.0, 5.1, 5.5, 6.0, 6.5 | 4.0, 4.1, 5.0, 5.1, 5.5, 6.0, 6.5 | 4.0, 4.1, 5.0, 5.1, 5.5, 6.0, 6.5 | 5.0, 5.1, 5.5, 6.0, 6.5, 6.7 | 5.0, 5.1, 5.5, 6.0, 6.5, 6.7 | 5.0, 5.1 6.0, 6.5 |
| | KVM | (***1) | (***1) | (***1) | (***1) | (***1) | (***1) |
| | Oracle VM Server for x86 | 3 (***2) | 3 (***2) | 3 (***2) | 3 (***2) | 3 (***2) | 3 (***2) |
| | Citrix XenServer | 5.0 以降 (***2) | 5.0 以降 (***2) | 5.0 以降 (***2) | | | |
| | Nutanix Acropolis Hypervisor（AHV） | 20160925.57 | 20160925.57 | 20160925.57 | 20160925.57 | 20160925.57 | 201609 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | 20160925.90 | 20160925.90 | 20160925.90 | 20160925.90 | 20160925.90 | 20160 |

*****1 サポート対象の Linux ディストリビューションに付属**
*****2 共有ディスク構成をサポートしていません。**
**SPSとの組み合わせで利用可能なクラウド環境については こちら を参照してください。**

SPS が現在サポートしているディスクアレイストレージモデルやアダプタ、およびそれらの認定の種類の一覧については サポートストレージ一覧 を参照してください。

SAP 関連のバージョン要件は「 SIOS Protection Suite for SAP ソリューションページ 」をご確認ください。

# 10. サポートストレージ

## LifeKeeper for Linux v9 サポートストレージ一覧

最終更新日: 2021/5/25

下記の表は、LifeKeeper for Linuix v9 環境で利用可能な、認定済の共有ストレージの一覧表です。ハードウェアの構成検討の際に確認してください。

### 認定済み共有ストレージについて

LifeKeeper の共有ストレージとして利用するストレージは、種類によって認定が必要なものと不要なものがあります。

以下のストレージは、認定されたものを使用する必要があります。下記の表に使用予定のストレージが記載されていることを確認してください。

- SCSI / FC / iSCSI / SAS など、SCSI-2/3 Reservation による I/O フェンシングを LifeKeeper で行うことが前提となる、複数ノードで同一のデータを参照する共有ストレージ

### 認定が不要な構成

以下の構成にストレージ認定はありません。

- NAS ストレージ (NAS RecoveryKit が必要です)
- DataKeeper による DataReplication を構成するディスク装置全般(内蔵/外付けを問いません)
- vSphere上でVMDK ARKによって保護される共有ストレージ上の仮想ディスク
- 以下の条件をすべて満たす環境で利用されるストレージ
  - OSやハードウェア、プラットフォームでサポート済みのストレージである
  - LifeKeeper のSCSI Reservation 機能をオフにする
  - LifeKeeper のQuorum/Witnessによるフェンシング機能を採用する

    ※上記3つの条件を満たす構成で利用される共有ストレージは、弊社の認定が無いものであっても利用することができます。

### ご利用いただけないハードウェア

USB や IEEE1394 などで接続する、コンシューマー向けストレージの利用はサポートされません。

## DELL

| ベンダー名 (*1) | ストレージモデル名 | バスタイプ | 接続構成 (*2) | サポート可否 | 必要な ARK(*3) (マルチパス ARK等) | サポートする LKのバージョ | 注意事項 |
|---|---|---|---|---|---|---|---|

| | | | | | | ン(*4) | |
|---|---|---|---|---|---|---|---|
| **DELL** | Dell EqualLogic: PS4100 PS4110 PS4210 PS6100 PS6110 PS6210 PS6610 | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | LUN 数が多い場合 (20 以上) は、 */etc/default/LifeKeeper* の REMOTETIMEOUT 設定を `REMOTETIMEOUT=600` に変更してください。 |
| | | | Single Path (DMMP) | YES | DMMP ARK | | |
| | MD3800f MD3820f MD3860f | FC | vSphere (RDM) | No | – | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | MD3400 MD3420 MD3460 | SAS | vSphere (RDM) | No | – | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | SCv3000 Series: SCv3000 SCv3020 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | | SAS | vSphere | No | – | v9.0 – v9.5.2 | |
| | | | Multi Path | YES | DMMP ARK | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | (DMMP) | | | | |
| | | | Single Path | No | – | | |
| | SCv2000 Series: SCv2000 SCv2020 SCv2080 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | | SAS | vSphere | No | – | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | Dell Storage: SC7020 SC5020 SC4020 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | Dell Compellent: SC9000 SC8000 | FC | vSphere (RDM) | No | – | – | |
| | | | Multi Path | No | – | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | (DMMP) | | | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | |
| | | | Single Path | No | – | | |

## Fujitsu

| ベンダー名 (*1) | ストレージモデル名 | バスタイプ | 接続構成 (*2) | サポート可否 | 必要な ARK(*3) (マルチパス ARK等) | サポートする LKのバージョン(*4) | 注意事項 |
|---|---|---|---|---|---|---|---|
| **Fujitsu** | ETERNUS: DX60 S2 DX80 S2 DX90 S2 DX410 S2 DX440 S2 DX8100 S2 DX8700 S2 DX60 S3 DX100 S3 DX200 S3 DX500 S3 DX600 S3 DX8700 S3 DX8900 S3 DX200F DX60 S4 DX100 S4 DX200 S4 DX500 S4 DX600 S4 DX60 S5 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | ETERNUSをBootデバイスとして使用した場合、プライマリノードからのBootデバイスアクセスが不可となったときに、ファイルシステムエラーが発生し、セカンダリノードへ切り替わらない可能性があります。<br>対応として、以下のようにファイルシステムエラー時にパニックするように設定してください。<br>例：xfsの場合は下記を設定後サーバをリブートしてください。<br>`echo 'fs.xfs.panic_mask=127' > /etc/sysctl.d/01-xfs.conf` |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Multi Path (EMPD) | YES | No Multipath ARK | | |
| | | | Single Path | YES | No Multipath ARK | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (EMPD) | YES | No Multipath ARK | | |
| | | | Single Path | YES | No Multipath ARK | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | DX100 S5<br>DX200 S5<br>DX500 S5<br>DX600 S5<br>DX900 S5<br>AF250<br>AF650<br>AF250 S2<br>AF650 S2<br>AF250 S3<br>AF650 S3<br>DX8900<br>S4 | | | | | | |

## Hitachi / Hitachi Vantara

| ベンダー名 (*1) | ストレージモデル名 | バスタイプ | 接続構成 (*2) | サポート可否 | 必要な ARK(*3) (マルチパスARK等) | サポートする LKのバージョン(*4) |
|---|---|---|---|---|---|---|
| **Hitachi / Hitachi Vantara** | Hitachi Virtual Storage Platform G1000<br>Hitachi Virtual Storage Platform G1500<br>Hitachi Virtual Storage Platform F1500 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (HDLM) | YES | HDLM ARK | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | YES | – | |
| | | ISCSI | Multi Path (HDLM) | No | – | – |
| | | | Single Path | No | – | |
| | Hitachi Virtual Storage Platform | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (HDLM) | YES | HDLM ARK | |
| | | | Single Path | YES | – | |
| | | ISCSI | Multi Path | No | – | – |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | (HDLM) | | | |
| | | | Single Path | No | – | |
| | Hitachi Virtual Storage Platform Mid Range Family:<br>Hitachi Virtual Storage Platform F900 (VSP F900)<br>Hitachi Virtual Storage Platform F700 (VSP F700)<br>Hitachi Virtual Storage Platform F370 (VSP F370)<br>Hitachi Virtual Storage Platform F350 (VSP F350)<br>Hitachi Virtual Storage Platform G900 (VSP G900)<br>Hitachi Virtual Storage Platform G700 (VSP G700)<br>Hitachi Virtual Storage Platform G370 (VSP G370)<br>Hitachi Virtual Storage Platform G350 (VSP G350)<br>Hitachi Virtual Storage Platform G150 (VSP G150)<br>Hitachi Virtual Storage Platform G130 (VSP G130) | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (HDLM) | YES | HDLM ARK | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | YES | No Multipath ARK | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 |
| | | | Single Path | YES | No Multipath ARK | |
| | Hitachi Virtual Storage Platform Mid Range Family:<br>Hitachi Virtual Storage Platform F800 (VSP F800)<br>Hitachi Virtual Storage Platform F600 (VSP F600)<br>Hitachi Virtual Storage Platform F400 (VSP F400)<br>Hitachi Virtual Storage Platform G800 (VSP G800)<br>Hitachi Virtual Storage Platform G600 (VSP G600)<br>Hitachi Virtual Storage Platform G400 (VSP G400)<br>Hitachi Virtual Storage Platform G200 (VSP G200)<br>Hitachi Virtual Storage Platform G100 (VSP G100) | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (HDLM) | YES | HDLM ARK | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | YES | No Multipath ARK | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 |
| | | | Single Path | YES | No Multipath ARK | |
| | Hitachi Unified Storage VM (HUS VM) | FC | vSphere (RDM) | YES | No Multipath | v9.0 – v9.5.2 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | ARK | |
| | | | Multi Path (HDLM) | YES | HDLM ARK | |
| | | | Single Path | YES | No Multipath ARK | |
| | | ISCSI | Multi Path (HDLM) | No | – | – |
| | | | Single Path | No | – | |
| | Hitachi Unified Storage 100 Series:<br>Hitachi Unified Storage 150 (HUS 150)<br>Hitachi Unified Storage 130 (HUS 130)<br>Hitachi Unified Storage 110 (HUS 110) | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (HDLM) | YES | HDLM ARK | |
| | | | Single Path | YES | No Multipath ARK | |
| | | ISCSI | Multi Path (HDLM) | No | – | – |
| | | | Single Path | No | – | |

## Hitachi

| ベンダー名 **(*1)** | ストレージモデル名 | バスタイプ | **接続構成 (*2)** | サポート可否 | 必要な **ARK(*3)** (マルチパス **ARK**等) | サポートする **LK**のバージョン**(*4)** | **注意事項** |
|---|---|---|---|---|---|---|---|
| **Hitachi** | BR1200 | ISCSI | Multi Path (HDLM) | No | – | – | |
| | | | Multi Path (DMMP) | No | – | | |
| | | | Single Path | No | – | | |
| | | SAS | Multi Path | YES | No | v9.0 – | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | (RDAC) | | Multipath ARK | v9.5.2 | シングルパスとマルチパスのいずれの場合も RDAC ドライバを使用する必要があります。<br><br>HDLM および HDLM ARK を使用する構成はサポートされません。 |
| | | | Single Path (RDAC) | YES | No Multipath ARK | | |
| | BR1250 | ISCSI | Multi Path (HDLM) | No | – | – | |
| | | | Multi Path (DMMP) | No | – | | |
| | | | Single Path | No | – | | |
| | | SAS | Multi Path (NECSPS) | YES | NEC SPS ARK | v9.0 – v9.5.2 | シングルパスとマルチパスのいずれの場合も SPS NEC iStorage StoragePathSavior (NECSPS) Recovery Kit を使用する必要があります。HDLM および HDLM ARK を使用する構成はサポートされません。<br><br>SPS NEC iStorage StoragePathSavior (NECSPS) Recovery Kit を RHEL7 系環境で使用する場合は、LifeKeeper for Linux v9.1 以上をご使用ください。 |
| | | | Single Path (NECSPS) | YES | NEC SPS ARK | | |
| | BR1650E BR 1650S | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (HDLM) | YES | HDLM ARK | | |
| | | | Single Path | YES | – | | |
| | | ISCSI | Multi Path (HDLM) | No | – | – | |
| | | | Single Path | No | – | | |

## HPE

| ベンダー名 | ストレージモデル名 | バスタイプ | 接続構成 **(*2)** | サポート | 必要な **ARK(*3)** (マルチ | サポートする | 注意事項 |
|---|---|---|---|---|---|---|---|

| (*1) | | | | 可否 | パス ARK等) | LKのバージョン(*4) | |
|---|---|---|---|---|---|---|---|
| HPE | Primera A630, A650, A670, C630, C650, C670 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | Nimble Storage: AF20 AF20Q AF40 AF60 AF80 AF1000 AF3000 AF5000 AF7000 AF9000 CS1000H CS1000 CS3000 CS5000 CS7000 HF20C HF20H HF20 HF40 HF60 | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.2.2 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | 3PAR: 3PAR StoreServ 8200 3PAR StoreServ 8400 3PAR StoreServ 8440 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 3PAR StoreServ 8450 | | | | | | |
| | 3PAR: 3PAR StoreServ 7400 3PAR StoreServ 7400c 3PAR StoreServ 7440 3PAR StoreServ 7440c 3PAR StoreServ 7450 3PAR StoreServ 7450c | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | **注記:** デフォルトのパスチェッカとのリザベーション競合を返します。この競合を回避するには、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。<br>`DMMP_REGISTRATION_TYPE=hba`<br><br>およびユーザーフレンドリーデバイスマッピングはサポートされません。<br>“multipath.conf” に次のパラメータを設定してください。<br>`“user_friendly_names no”`<br>＊vSphere(RDM)で使用する場合は、上記の設定は必要ありません。 |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | **注記:** リザベーション競合を返す場合があります。この競合を回避するため、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。<br>`DMMP_REGISTER_IGNORE=TRUE` |
| | | | Single Path | No | – | | |
| | 3PAR: 3PAR StoreServ 7200 3PAR StoreServ 7200c | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | **注記:** デフォルトのパスチェッカとのリザベーション競合を返します。この競合を回避するには、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。<br>`DMMP_REGISTRATION_TYPE=hba` |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | 3PAR: 3PAR StoreServ 9450 3PAR StoreServ 20800 R2 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.2.1 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 3PAR StoreServ 20840 R2<br>3PAR StoreServ 20850 R2 | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | 3PAR: 3PAR StoreServ 20450<br>3PAR StoreServ 20800<br>3PAR StoreServ 20840<br>3PAR StoreServ 20850 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.1 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | 3PAR StoreServ 10400 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | 3PAR StoreServ 10800 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | **注記:** デフォルトのパスチェッカとのリザベーション競合を返します。この競合を回避するには、/etc/default/LifeKeeper 内に次のパラメータを設定 (追加) してください。<br>`DMMP_REGISTRATION_TYPE=hba` |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | Single Path | No | – | | およびユーザーフレンドリーデバイスマッピングはサポートされません。<br>“multipath.conf” に次のパラメータを設定してください。<br>“user_friendly_names no”<br>＊vSphere(RDM)で使用する場合は、上記の設定は必要ありません。 |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | MSA: MSA2060 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | | SAS | vSphere (RDM) | No | – | – | |
| | | | Multi Path (DMMP) | No | – | | |
| | | | Single Path | No | – | | |
| | MSA: MSA2050 MSA2052 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.2 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.2 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | | SAS | vSphere (RDM) | No | – | – | |
| | | | Multi Path (DMMP) | No | – | | |
| | | | Single Path | No | – | | |
| | MSA: MSA2042 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.1 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.1 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | | SAS | vSphere (RDM) | YES | No Multipath ARK | v9.1 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | MSA: MSA1040 MSA2040 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.0 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | | SAS | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | StoreVirtual (LeftHand) Series LeftHand OS version 12.6 | FC | vSphere (RDM) | No | – | – | LifeKeeper for Linux は HPE 社 StoreVirtual(LeftHand) ストレージの OS（SAN/iQ）バージョン 12.6 をサポートします。<br>本バージョンの OS が使われている全ての StoreVirtual シリーズ（HC250/HC380を含む）がサポート対象であり、仮想ストレージ・アプライアンスである StoreVirtual VSA も含まれます。<br>利用可能な仮想ストレージ・アプライアンスについての詳細は各 OS（SAN/iQ）バージョンのリリースノートをご確認ください。 |
| | | | Multi Path (DMMP) | No | – | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK | v9.1 – v9.5.2 | |
| | | | Single Path | No | – | | |
| | HP P9500 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | YES | No Multipath ARK | | |
| | XP7 Storage System | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | Multi Path (HDLM) | YES | HDLM ARK | | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | YES | No Multipath ARK | | |

## IBM

| ベンダー名 **(*1)** | ストレージモデル名 | バスタイプ | 接続構成 **(*2)** | サポート可否 | 必要な **ARK(*3)** **(**マルチパス **ARK**等**)** | サポートする **LK**のバージョン**(*4)** | 注意事項 |
|---|---|---|---|---|---|---|---|
| IBM | IBM SAN Volume Controller* *IBM TotalStorage Proven | FC | vSphere (RDM) | No | – | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | YES | No Multipath ARK | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | IBM Storwize V7000 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | ISCSI | Multi Path (DMMP) | YES | DMMP ARK Quorum/ Witness Server Kit | v9.0 – v9.5.2 | パートナーテストによりiSCSI (iscsi-initiator-utils-6.2.0.872-34.el6.x86_64) と DMMP (device-mapper-1.02.66-6.el6、device-mapper-multipath-0.4.9-46.el6) を使用する IBM Storwize V7000 (Firmware Version 6.3.0.1)を 認定しています。<br>テストは、LifeKeeper for Linux v7.5 と RHEL 6.2 を使用して行われました。<br><br>**制限事項:** IBM Storwize V7000 を iSCSI 接続で利用する場合は、LifeKeeper 側の Reservation を無効とした上で、Quorum/ Witness Server Kit および STONITH を適用する必要があります。<br>具体的な構成方法についてはご利用中の LifeKeeperのバージョン用のオンラインマニュアルにてご確認ください。<br>http://jpdocs.us.sios.com/ |
| | | | Single Path | No | – | | |
| | IBM Storwize V3700 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (DMMP) | YES | DMMP ARK | | |
| | | | Single Path | No | – | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |
| | | SAS | vSphere (RDM) | No | – | – | |
| | | | Multi Path (DMMP) | No | – | | |
| | | | Single Path | No | – | | |
| | IBM XIV Storage System | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | LifeKeeper を使用する IBM XIV Storage System で 32 個以上の LUN を作成する必要がある場合、 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | Multi Path (DMMP) | YES | DMMP ARK | | 詳細については IBM の営業担当者にお問い合わせください。 |
| | | | Single Path (DMMP) | YES | DMMP ARK | | |
| | | ISCSI | Multi Path (DMMP) | No | – | – | |
| | | | Single Path | No | – | | |

## Lenovo

| ベンダー名 (*1) | ストレージモデル名 | バスタイプ | 接続構成 (*2) | サポート可否 | 必要なARK(*3) (マルチパスARK等) | サポートするLKのバージョン(*4) |
|---|---|---|---|---|---|---|
| Lenovo | Lenovo Storage V3700 V2<br>Lenovo Storage V3700 V2 XP | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | No | – | |
| | | ISCSI | Multi Path (DMMP) | No | – | – |
| | | | Single Path | No | – | |
| | | SAS | vSphere (RDM) | No | – | – |
| | | | Multi Path (DMMP) | No | – | |
| | | | Single Path | No | – | |
| | IBM Storwize V3700 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | No | – | |
| | | ISCSI | Multi Path (DMMP) | No | – | – |
| | | | Single Path | No | – | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | SAS | vSphere (RDM) | No | – | – |
| | | | Multi Path (DMMP) | No | – | |
| | | | Single Path | No | – | |

## NEC

| ベンダー名 (*1) | ストレージモデル名 | バスタイプ | 接続構成 (*2) | サポート可否 | 必要な ARK(*3) (マルチパス ARK等) | サポートする LKのバージョン(*4) | 注意事項 |
|---|---|---|---|---|---|---|---|
| NEC | NEC iStorage: M10e M11e M100 M110 M300 M12e M120 M320 | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | SPS NEC iStorage StoragePathSavior (NECSPS) Recovery Kit を RHEL7 系環境で使用する場合は LifeKeeper for Linux v9.1 以上をご使用ください。 |
| | | | Multi Path (NECSPS) | YES | NECSPS ARK | | |
| | | | Single Path (NECSPS) | YES | NECSPS ARK | | |
| | | ISCSI | Multi Path (NECSPS) | YES | NECSPS ARK | v9.0 – v9.5.2 | |
| | | | Single Path (NECSPS) | No | – | | |
| | | SAS | vSphere (RDM) | No | – | v9.0 – v9.5.2 | |
| | | | Multi Path (NECSPS) | YES | NECSPS ARK | | |
| | | | Single Path (NECSPS) | YES | NECSPS ARK | | |
| | NEC iStorage: M310 M310F M320F | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 | |
| | | | Multi Path (NECSPS) | YES | NECSPS ARK | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | M500<br>M510<br>M700<br>M710<br>M710F<br>M520<br>M720<br>M720F | | Single Path (NECSPS) | YES | NECSPS ARK | | |
| | | ISCSI | Multi Path (NECSPS) | YES | NECSPS ARK | v9.0 – v9.5.2 | |
| | | | Single Path (NECSPS) | No | – | | |

## Pure Storage

| ベンダー名 (*1) | ストレージモデル名 | バスタイプ | 接続構成 (*2) | サポート可否 | 必要なARK(*3) (マルチパスARK等) | サポートするLKのバージョン (*4) |
|---|---|---|---|---|---|---|
| Pure Storage | FA-400 Series | FC | vSphere (RDM) | YES | No Multipath ARK | v9.0 – v9.5.2 |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | No | – | |
| | | ISCSI | Multi Path (DMMP) | No | – | – |
| | | | Single Path | No | – | |
| | FlashArray//m Series<br>//m10<br>//m20<br>//m50<br>//m70 | FC | vSphere (RDM) | No | – | v9.0 – v9.5.2 |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | No | – | |
| | FlashArray//x Series<br>//x10<br>//x20<br>//x50<br>//x70 | FC | vSphere (RDM) | No | – | v9.0 – v9.5.2 |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | No | – | |
| | FlashArray//x Series<br>//X10R2<br>//X20R2<br>//X50R2<br>//X70R2<br>//X90R2 | FC | vSphere (RDM) | No | – | v9.3.2 – v9.5.2 |
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | No | – | |
| | | iSCSI | vSphere (RDM) | No | – | v9.0 – v9.5.2 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | Multi Path (DMMP) | YES | DMMP ARK | |
| | | | Single Path | No | – | |

## (*1): EMC ストレージのご利用に関して

LifeKeeper は E-Lab の認定を取得しております。EMC 社のストレージのサポート状況については下記のマトリクスをご参照ください。

http://www.emc.com/interoperability

## (*2): 接続構成の説明

| | |
|---|---|
| vSphere (RDM) | VMWare ESX サーバーで共有ディスクを Raw Device Mapping (RDM) で接続した構成 (*5) |
| Multi Path (DMMP) | Device Mapper Multipath を使用した共有ディスクのマルチパス構成 |
| Multi Path (EMPD) | ETERNUS Multipath Driver を使用した共有ディスクのマルチパス構成 |
| Multi Path (HDLM) | Hitachi Dynamic Link Manager を使用した共有ディスクのマルチパス構成 |
| Multi Path (RDAC) | IBM Redundant Disk Array Controller を使用した共有ディスクのマルチパス構成 |
| Multi Path (NECSPS) | iStorage StoragePathSavior を使用した共有ディスクのマルチパス構成 |
| Single Path (DMMP) | Device Mapper Multipath を使用した共有ディスクのシングルパス構成 |
| Single Path (RDAC) | IBM Redundant Disk Array Controller を使用した共有ディスクのシングルパス構成 |
| Single Path | 特別なマルチパスドライバを用いないシングルパス接続 |

## (*3): マルチパスソフトウェアに関して

マルチパスソフトウェアの接続情報については必ず各ベンダーのサポート情報をご確認ください
マルチパスソフトウェアと LifeKeeper の対応の詳細は、リリースノートでご確認ください

## (*4): サポートする LifeKeeper のバージョンに関して

このドキュメントは、LifeKeeper for Linux v9 を対象としています。
以前のバージョンのサポートストレージにつきましては、各バージョンのリリースノートをご確認ください。

## (*5): VMware vSphere での共有ディスクの利用について

以下の URL にある、VMware vSphere 上で共有ディスクを利用する際の構成ガイドをご確認ください。
https://lkdkuserportal.sios.jp/hc/ja/articles/360037730991

表の iSCSI 接続に関する記載は物理環境と vSphere 環境で共通です。その為、バスタイプ “iSCSI” の接続構成の項目に “vSphere” の掲載は行っていません。

# 11. Evaluation Guides

DataKeeper for Linux Evaluation Guide

SIOS Protection Suite for Linux Evaluation Guide for Cloud Environments

# 11.1. DataKeeper for Linux Evaluation Guide

## Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product with DataKeeper to enable real time, host based, block-level data replication

There are five phases in this process:

- Phase 1 – Prepare to Install
- Phase 2 – Configure Storage
- Phase 3 – Install SIOS Protection Suite for Linux
- Phase 4 – Configure your LifeKeeper Cluster
- Phase 5 – Test Your Environment

# 11.1.1. DataKeeper for Linux Terms to Know

The following terms are used throughout this document and, while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

## Network Communication Terms

**Crossover cable** – A cable used to directly connect computing devices together, instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

## Types of LifeKeeper Servers

**Server** – A computer system dedicated to running software application programs.

**Active Server** – This is the server where the resource hierarchy is currently running (IN SERVICE).

**Standby Server** – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

**Primary Server** – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

**Secondary Server** – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

**Source Server** – In a LifeKeeper cluster, using data replication, this is the Active Server. It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

**Target Server** – In a LifeKeeper cluster, using data replication, this is the Standby Server. The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

## SIOS DataKeeper Terms

**Replication** – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

**Asynchronous** – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there and can be written on them.

**Rate of Change** – A measure of the amount of data which is changing over a set period of time.

**Compression** – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

**Throttling** – An optionally implemented mechanism to limit the bandwidth used for replication.

## LifeKeeper Product Terms

**Communications Path** – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

**Heartbeat** – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

**Split Brain** – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

**Failover** – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

**Switchover** – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

**Switchback** – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

**Resource** – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

**Extend a Resource** – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously

**Resource Hierarchy** – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

**Shared Storage** – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally called I/O fencing.

**Data Replication (Disk Mirroring)** – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

**Source** – The partition on the source server used for replication. The “gold” copy of the data.

**Target** – The partition on the target server used for replication.

**Switchable IP Address** – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

# 11.1.2. The Evaluation Process

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux within a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

**Important:** Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

# 11.1.3. Prepare to Install DK for Linux

## Hardware Requirements

### Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional disk partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system ( / ) and boot (/boot) partitions are not eligible for replication.

## Software Requirements

### Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
    - RedHat Enterprise Linux 6.x, 7.x, or 8.x
    - CentOS Linux 6.x, 7.x, or 8.x
    - Oracle Enterprise Linux 6.x, 7.x, or 8.x
    - SuSE Linux Enterprise Server 11, 12, or 15
    - See Linux Release Notes for a full list of supported Operating Systems
- Current patches / security updates are recommended.
- Satisfied dependencies; especially if the Linux installation package selection was base/minimal you will need to refer to the dependencies documentation at Linux Dependencies
- It is recommended the firewall be disabled
    - # service iptables stop (systemctl stop firewalld)
    - # chkconfig iptables off (systemctl disable firewalld)
    - See here for information regarding the ports SIOS Protection Suite for Linux uses.
- Disable SELinux :

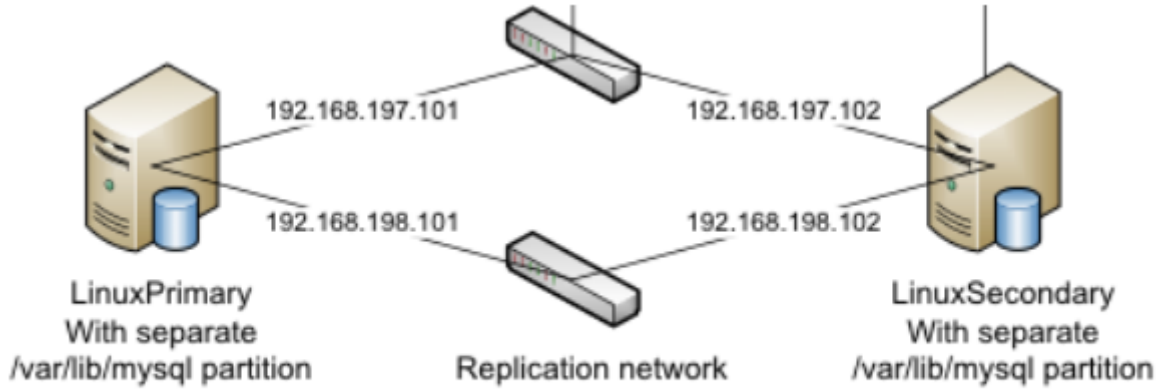
    - Edit /etc/selinux/config
    - Set SELINUX=disabled (note: permissive mode is also acceptable)
- Check the configuration of your /etc/hosts file
    - localhost.localdomain and localhost are the only entries that can be on 127.0.0.1
    - Create a separate entry for your hostname with a static address
- GUI Authentication with PAM
    - SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).
    - Users are identified and authenticated against the system's PAM configuration. Privilege levels are determined from group membership as provided through PAM.
    - In order to access the GUI, a user must be a member in one of the three LifeKeeper groups: lkadmin, lkoper or lkguest.
    - See Configuring GUI Users for more information.

# Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi-homed, between two LAN segments (the second NIC in each server could even be connected via a cross-over cable if a second physical network is not available). The second NIC is optional in this configuration, but highly recommended in production environments to avoid a single point of failure.

In this example evaluation scenario we will be replicating direct attached storage




**Network Configuration Example**

## IMPORTANT: Rate of Change Analysis

When replicating data in real time, its critical to ensure that you have sufficient bandwidth to keep the replication in a mirroring state at all times. To perform a Rate of Change analysis on your server, which will collect and analyze Write activity over time vs. bandwidth, refer to Measuring Rate of Change on a Linux System.

### Primary Server and Secondary Servers

• Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

Example:

192.168.197.101 LinuxPrimary

192.168.197.102 LinuxSecondary

• See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.

• Public Network connection(s) configured with:

º Static IP address

º Correct subnet mask

- º Correct gateway address

• Private Network connection(s) configured with:

- º Static IP address (on a different subnet from the public network)
- º Correct network mask
- º No gateway IP address
- º No DNS server addresses

# 11.1.4. Configure Storage for DK for Linux

## Before You Begin

Ensure the following:

- You have an extra disk/partition on both servers that can be used for data replication. A target volume's size must equal to or larger than the size of its source disk/partition.

## Partition local storage for use with SIOS DataKeeper for Linux

### Primary Server

On your Primary server, perform the following actions:

1. Identify an existing free, unused disk partition to use as the source of the mirror. Alternatively, create a new partition. Use the "gdisk" utility to partition your disk appropriately. In this example /dev/sdb is an unused disk where we will create a single partition

   a. gdisk /dev/sdb

   b. Press "n" to create a new partition

   c.This example uses a new disk, so we will use all default values (Partition 1, entire disk and Linux filesystem partition type) Hit Enter four times to confirm these parameters.

   d. Press "w" to write the partition table

   e. Press "Y" to confirm to overwrite existing partitions

**Example**

[root@LinuxPrimary ~]# **gdisk /dev/sdb**

GPT fdisk (gdisk) version 0.8.10

Partition table scan:
MBR: protective
BSD: not present
APM: not present
GPT: present

Found valid GPT with protective MBR; using GPT.

Command (? for help): **n**

Partition number (1-128, default 1): **<enter>**
First sector (34-2047, default = 34) or {+-}size{KMGTP}: **<enter>**
Last sector (34-2047, default = 2047) or {+-}size{KMGTP}: **<enter>**
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): **<enter>**
Changed type of partition to 'Linux filesystem'

Command (? for help): **w**

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING PARTITIONS!!

Do you want to proceed? (Y/N): **Y**
OK; writing new GUID partition table (GPT) to /dev/sdb.
Warning: The kernel is still using the old partition table.
The new table will be used at the next reboot.
The operation has completed successfully.

[root@LinuxPrimary ~]#

2. Format the newly created disk partition

   # mkfs.ext3 /dev/sdb1

3. Mount the partition at the desired location, for example /var/lib/mysql

   # mount /dev/sdb1 /var/lib/mysql

4. Note: there is no need to add an entry to /etc/fstab. Lifekeeper will take care of mounting this automatically.

**Result**
[root@LinuxPrimary ~]# df /var/lib/mysql
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sdb1 253855 11083 229666 5% /var/lib/mysql

## Secondary Server

5. On your Secondary server, only perform Step #1 above, where you partition the disk. The size of the Target disk/partition needs to be the same size, or greater, than our Source disk/partition. There is no need to format or mount the filesystem.

# 11.1.5. Install SIOS Protection Suite for Linux

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

| SPS for Linux Component | Install Location |
|---|---|
| LifeKeeper Software | /opt/LifeKeeper |
| LifeKeeper Config File | /etc/default/LifeKeeper |

Perform the following actions on **both** Primary and Secondary server.

## Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.

2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After successful download you will have downloaded contents similar to the follow directory listing:

      [root@LinuxPrimary ~]# ls -l <directory> total 63680

      -rw-r—r— 1 root root 23163 May 30 14:03 EULA.pdf

      -rw-r—r— 1 root root 536 May 30 14:03 readme.txt

      -rw-r—r— 1 root root 65179648 May 30 14:03 sps.img

3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

## Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

   # mount -o loop sps.img /mnt

```
# cd /mnt

# ./setup
```

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

```
# cd /root

# umount /mnt
```

# Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

```
# /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic
```

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

```
# /opt/LifeKeeper/bin/lklicmgr

License File: 20101230.lic
```

| Product | Type | Expiry |
|---|---|---|
| LifeKeeper for Linux | Eval | 27 Mar 2013 (87 days) |

| | | |
|---|---|---|
| SIOS Data Replication ARK | Eval | 27 Mar 2013 (87 days) |

# Start the SIOS Protection Suite for Linux

1. Start:

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

# 11.1.6. Configure the Cluster – DK for Linux

## Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

> **Important** Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously

## Access the LifeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application, or as an applet within your Java-Enabled Web Browser.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 errors.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

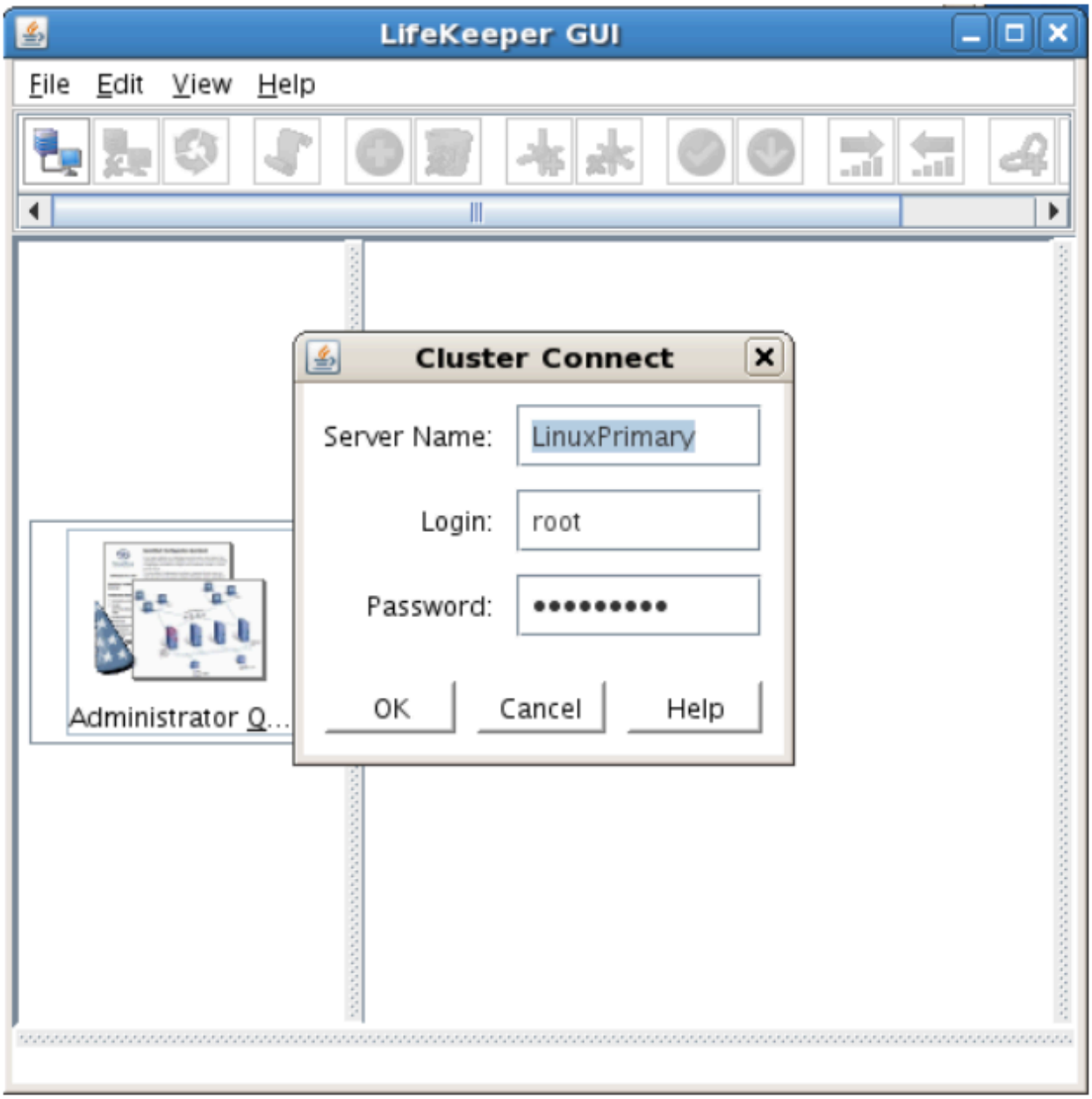
   a. /opt/LifeKeeper/bin/lkGUIapp &

3. To Connect to the LifeKeeper GUI Applet from a Web Browser, go to:

   a. http://<hostname>:81

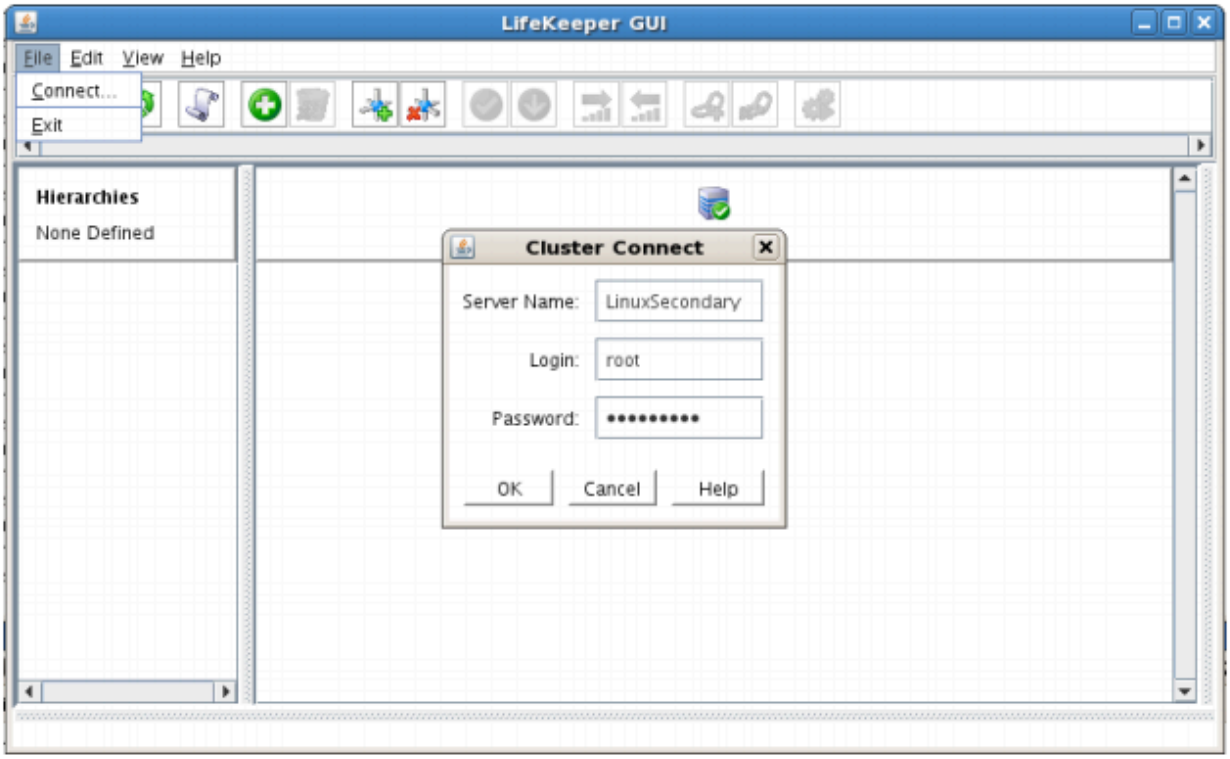
4. Enter the name of the server you wish to connect to (this field will be populated with the name of the server you are on, if you are running the GUI from a server with LifeKeeper installed) along

with your root credentials and click OK.



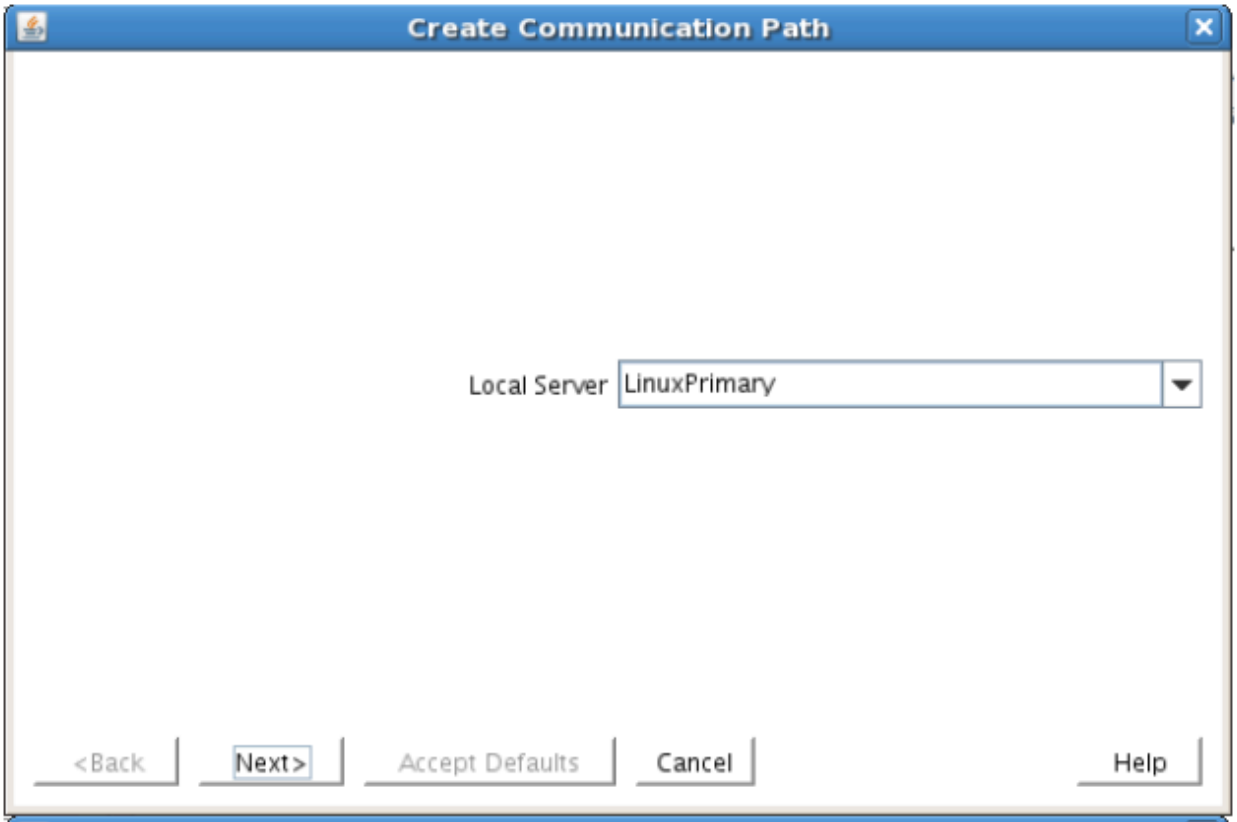

## Create Communication (Comm) Paths

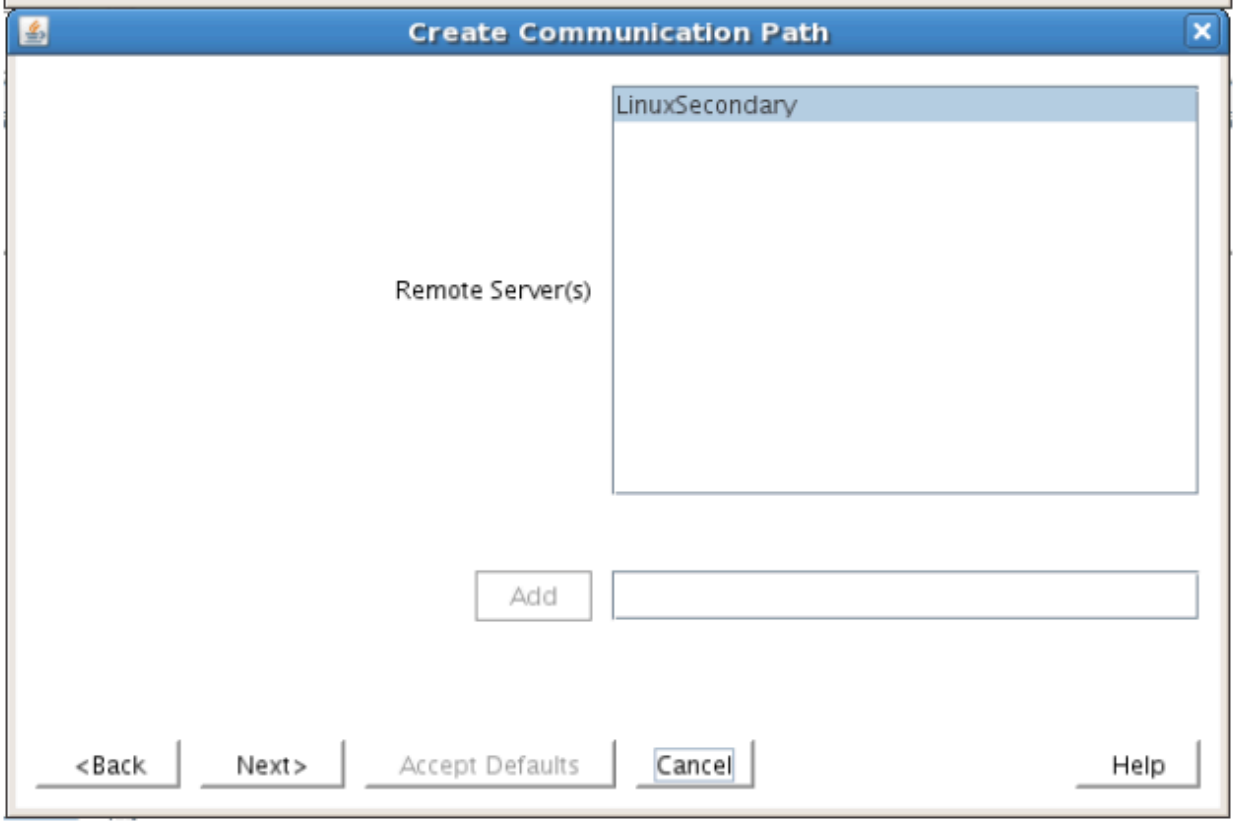
5. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



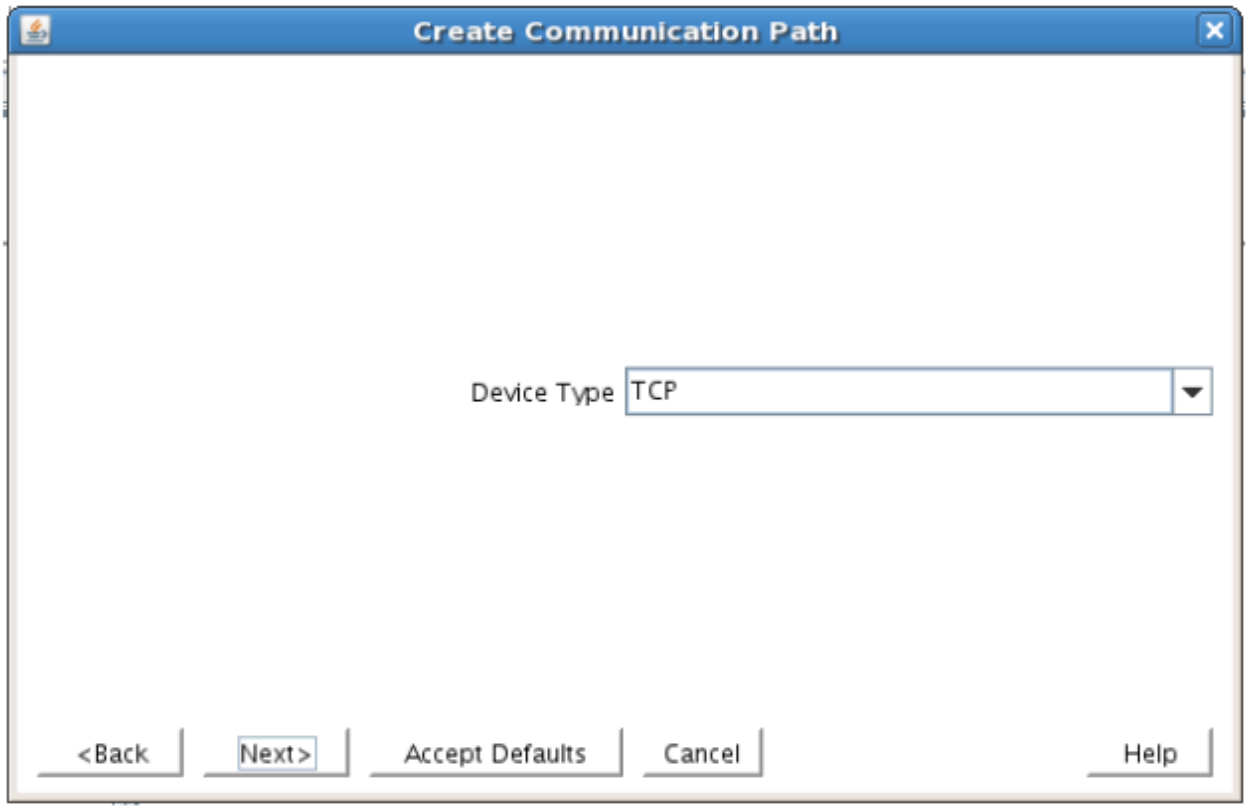

6. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

7. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

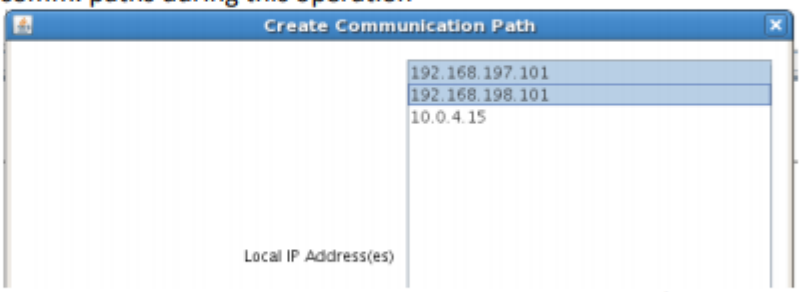


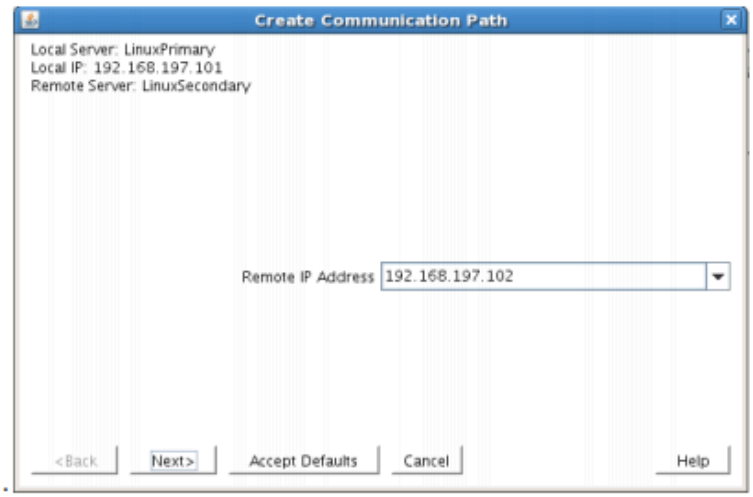




8. Select TCP for Device Type and Click Next.



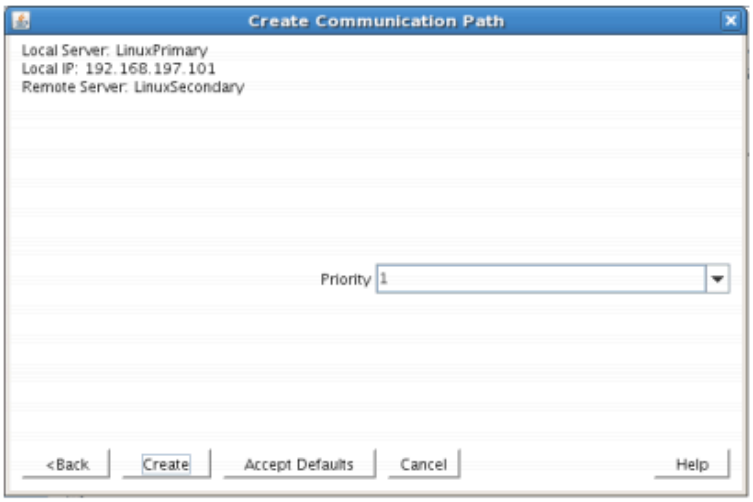

9. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

| Field | Tips |
|---|---|
| ***For TCP/IP Comm Path...*** | |
| Local IP Address | Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation   |
| Remote IP Address | Choose the IP address to be used by the remote server for this comm path |

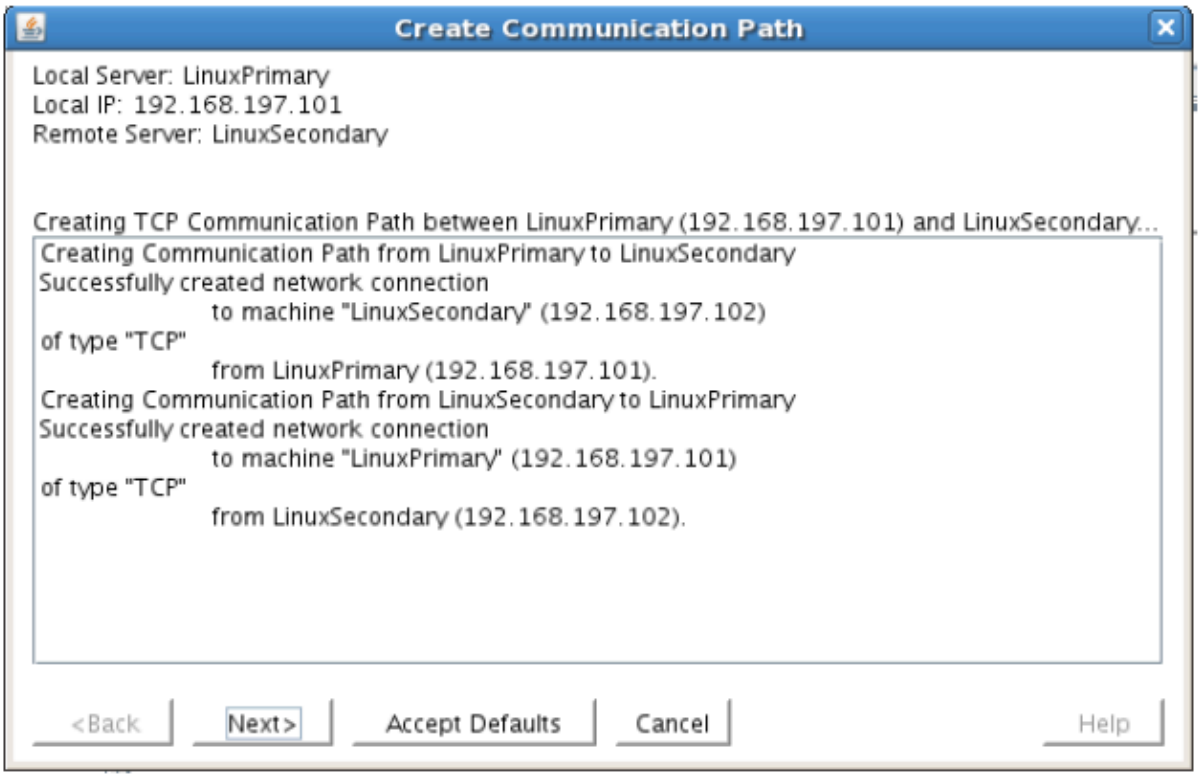



Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority




10. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



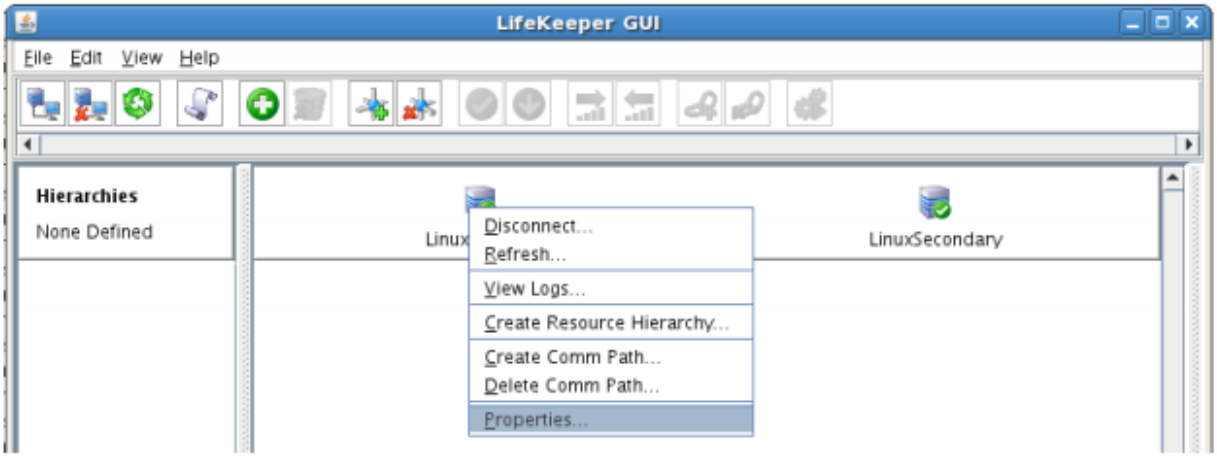

If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

11. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

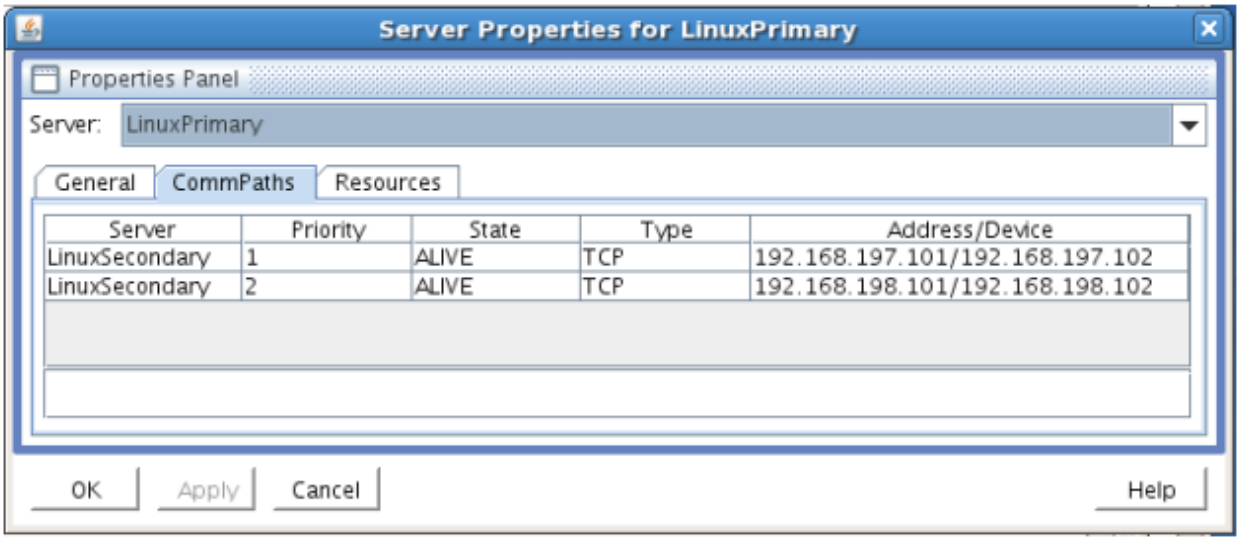
## Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.




2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.




# Create the LifeKeeper Hierarchy

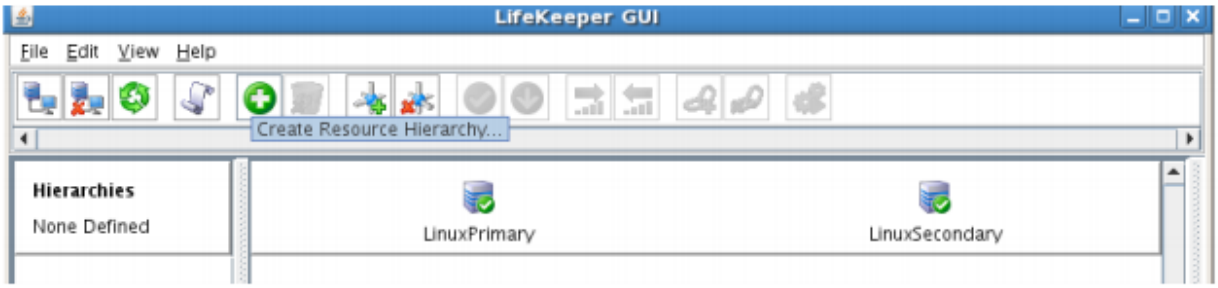
## Create a Mirror and Begin Data Replication

In this section we will setup and configure the Data Replication resource, which be used to synchronize our MySQL's data between cluster nodes. The data we will replicate resides in the /var/lib/mysql partition on our Primary cluster node

Please note:

- The source volume to be replicated must be mounted on the Primary server
- The target volume, which will received replicated data, must NOT be mounted on the Secondary server.
- The target volume's size must equal to or larger than the size of its source volume.

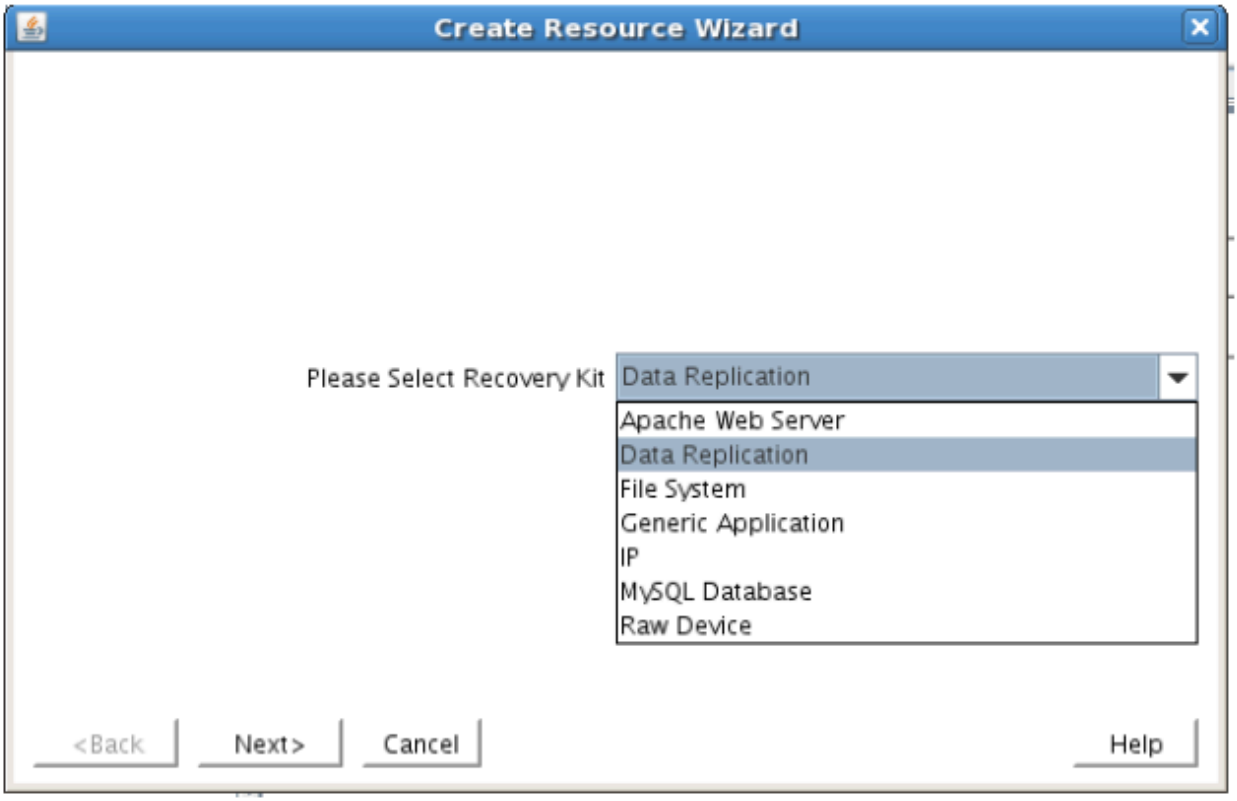
1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.




The Create Resource Wizard dialog box will appear with a drop down list box displaying all

recognized Recovery Kits installed within the cluster.

2. Select Data Replication and click Next.

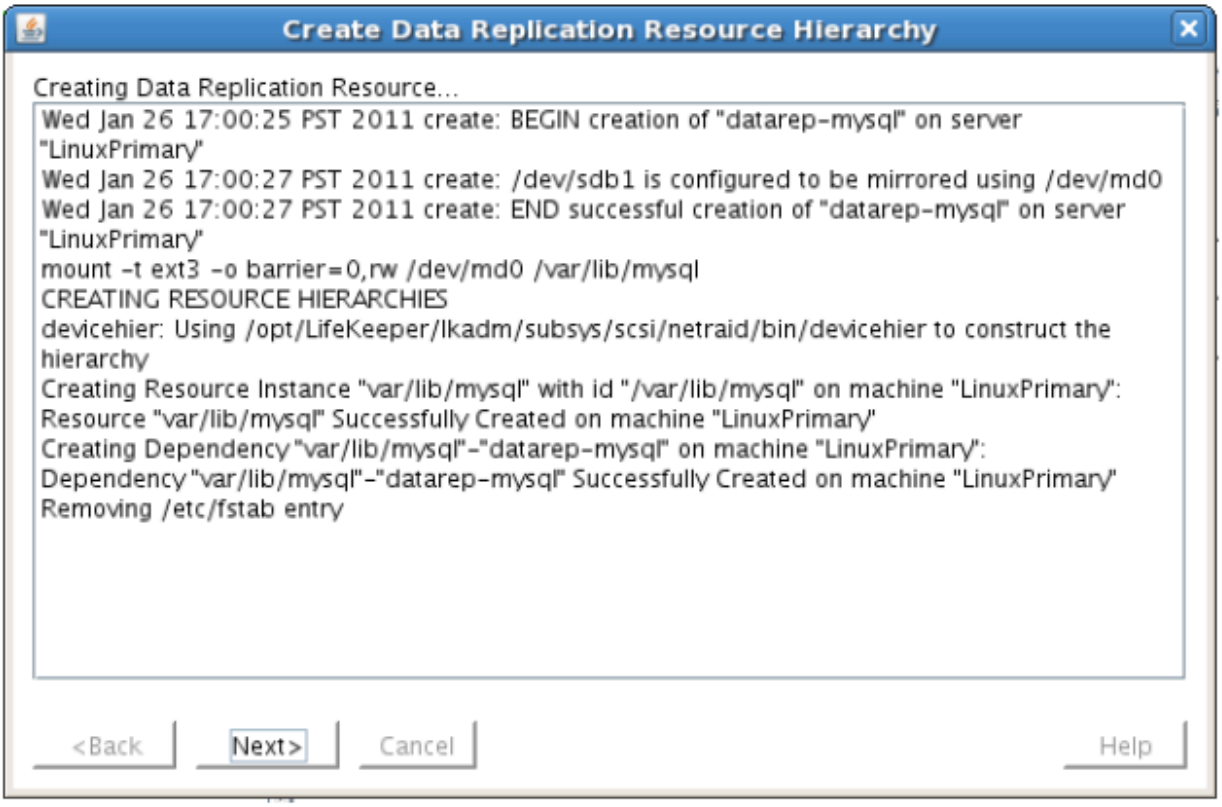



3. Follow the Data Replication wizard, and enter the following values:

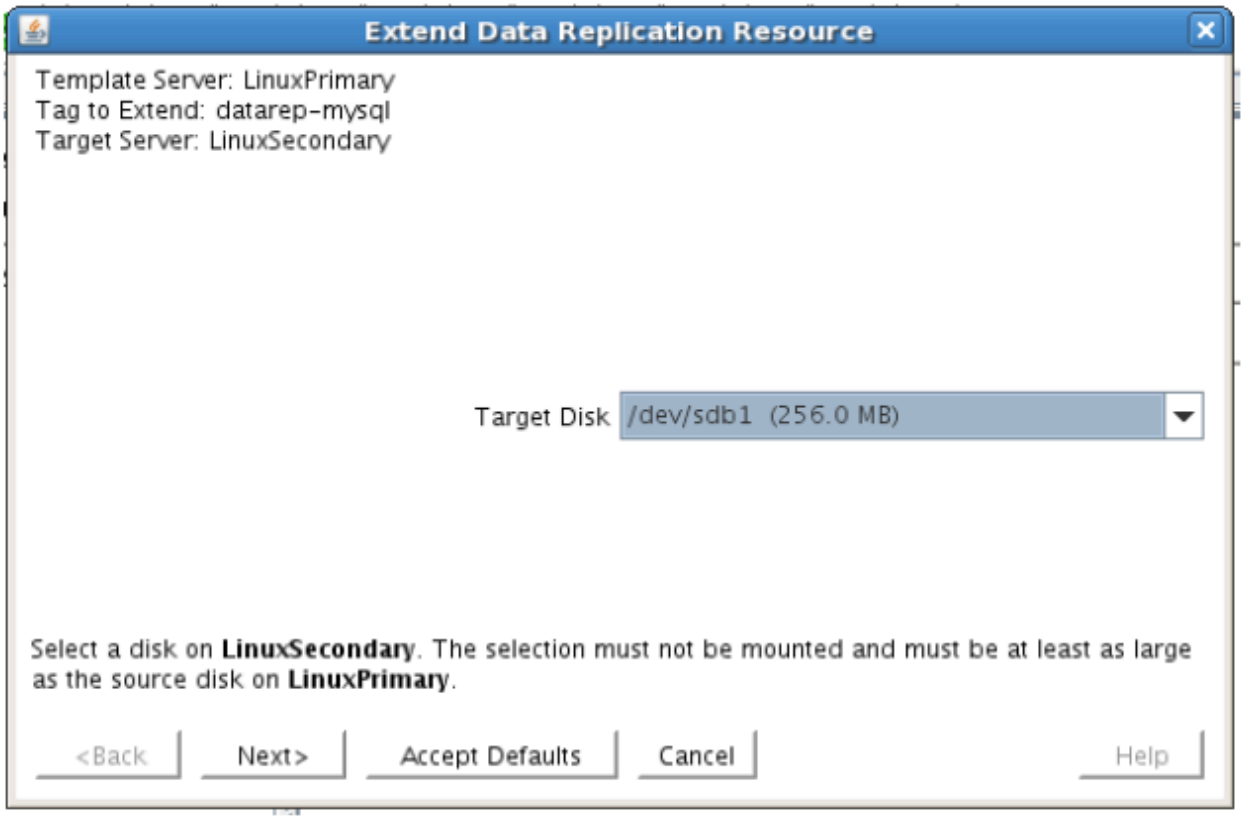
| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type<br><br>Server | Intelligent<br><br>LinuxPrimary (Primary Cluster Node, i.e. Mirror Source) |
| Hierarchy Type<br><br>Existing Mount Point | Select: “**Replicate Existing Filesystem**”<br><br>At this step you will select the mounted partition to replicate. In our example, select “/var/lib/mysql” |
| Data Replication Resource Tag<br><br>File System Resource Tag | Leave as default<br><br>Leave as default<br><br>Leave as default (Note: if using high speed SSD storage you will want to create a small partition and use it for bitmap placement, i.e. /bitmaps) |

| | |
|---|---|
| Bitmap File | |
| Enable Asynchronous Replication | Leave as default (Yes) |

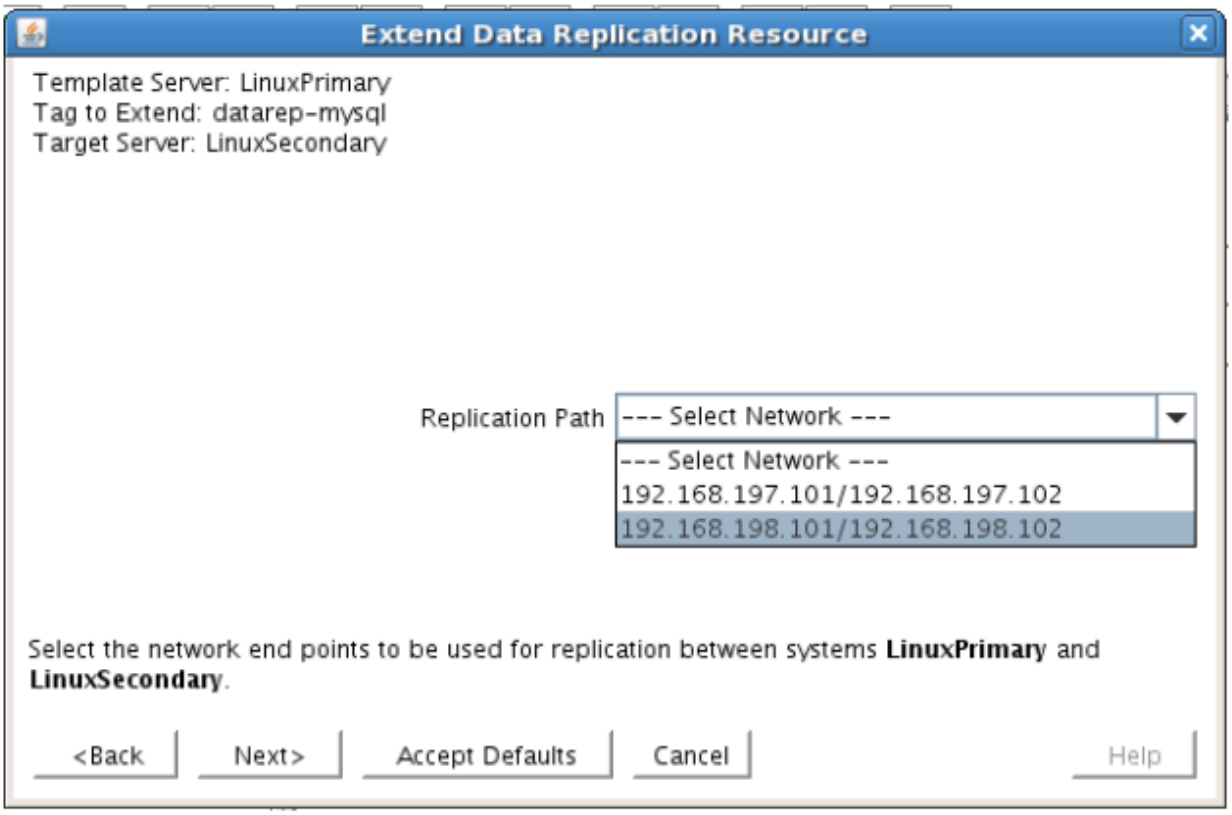
4. Click Next to begin creation of the Data Replication resource hierarchy. Status will be displayed in the GUI as follows:



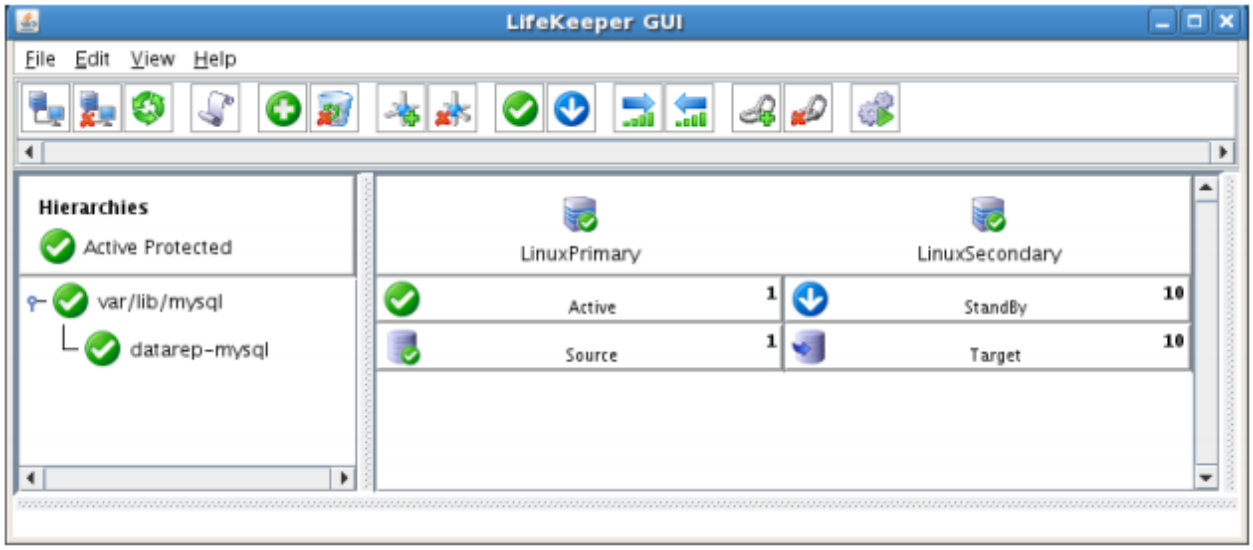

5. Click Next to begin the process to Extend the Data Replication Resource. Select all default settings. When it asks for the target disk, select a free partition on your Target server which is the same size (or greater) than the Source Volume we are replicating. This partition should NOT be mounted on the Target system.




6. Continue through the wizard, and you will be prompted to select the network you would like replication to take place over. In general, it's a best practice to separate your user/application and your replication traffic. In our example setup we will replicate over our backend network, 192.168.198.X



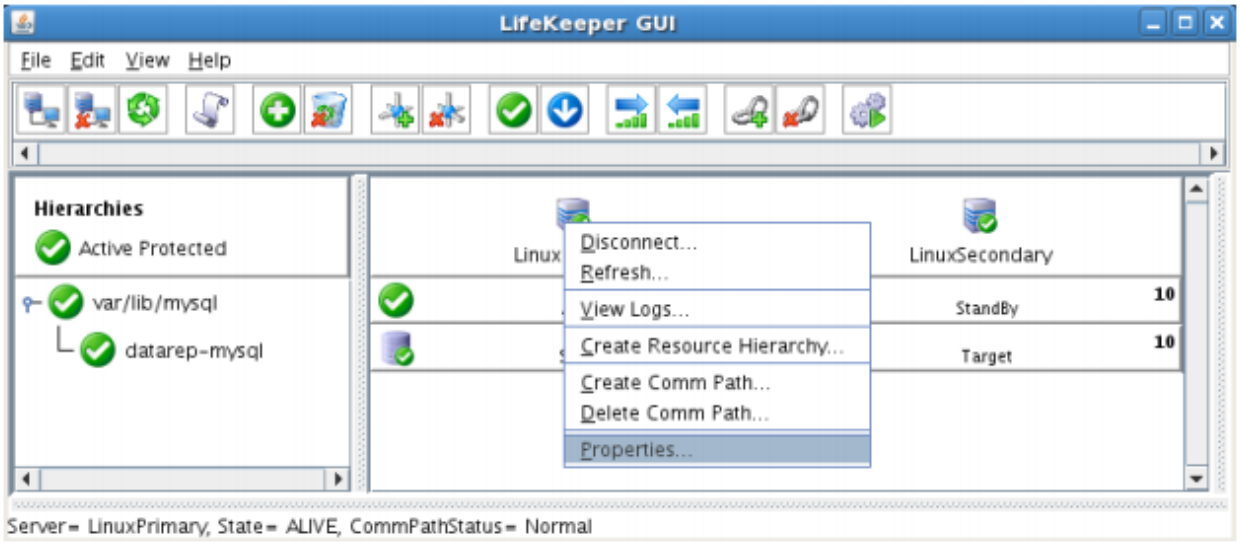
7. Click Next and continue through the wizard. Once completed, your resource hierarchy will look as follows




## Disable Automatic Failover

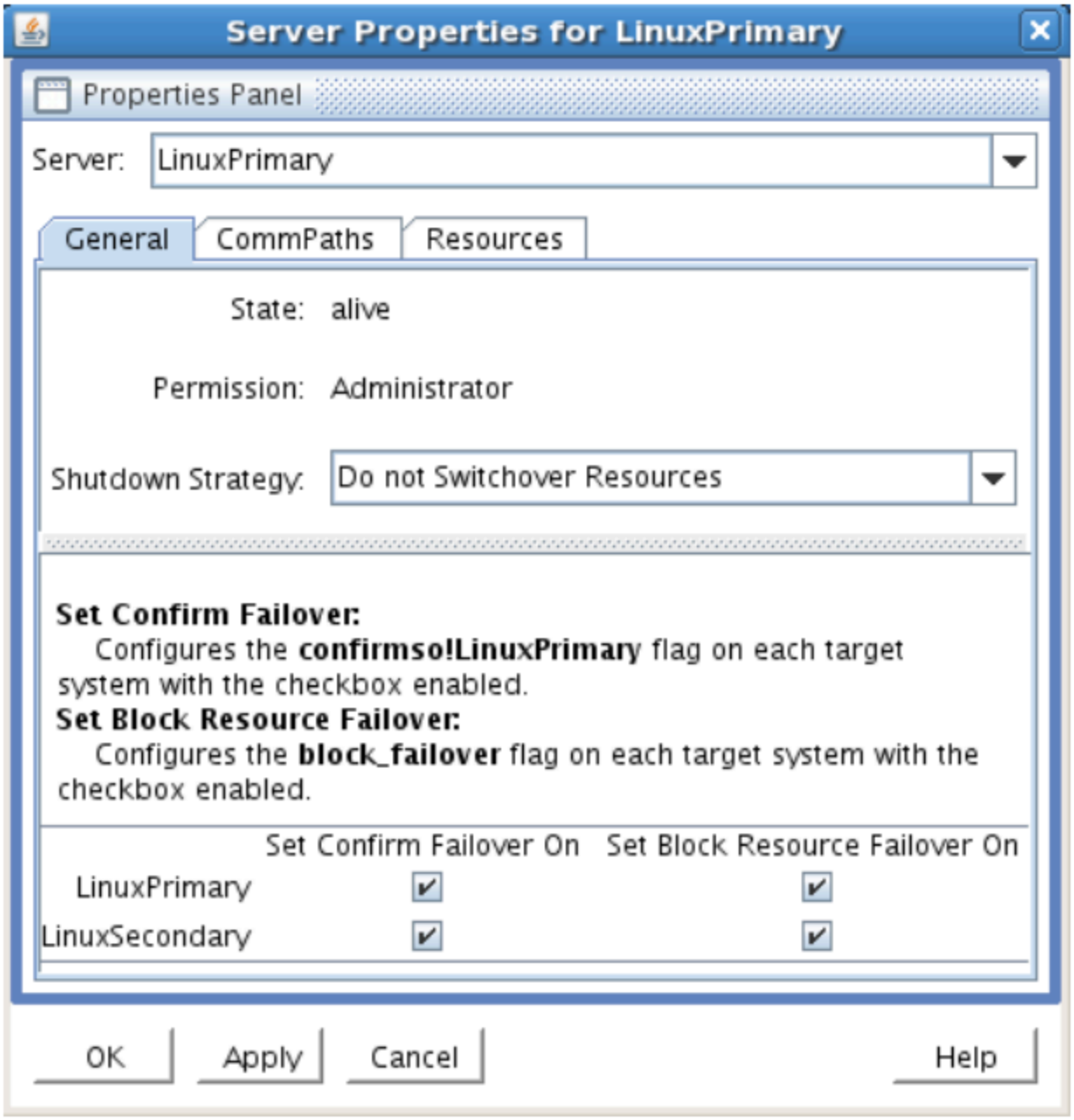
In this section we will review the procedure for disabling automatic failover to the standby server.

1. From the LifeKeeper GUI, right click on one of the cluster nodes and select Properties.

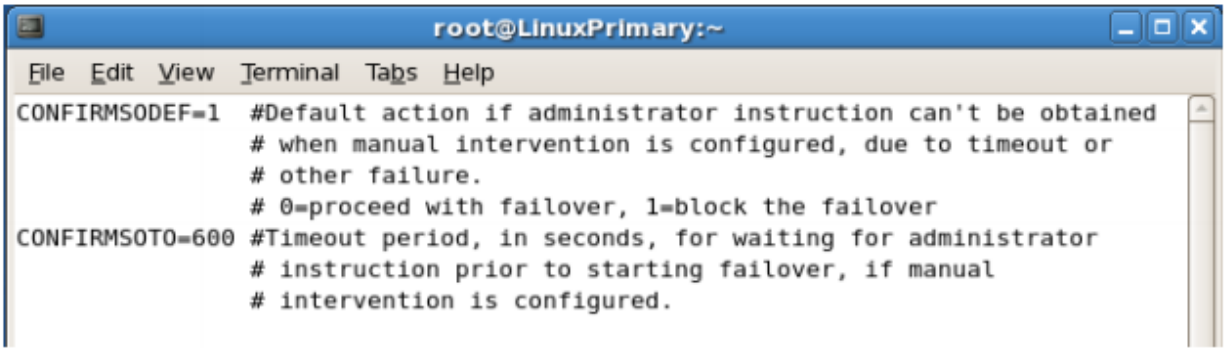



2. Select the Source server from the “Server:” drop down at the top of the window

3. Once the Server Properties window loads, check all boxes at the bottom of the page. This will prevent any automatic failovers from happening.




4. Click Apply
5. Repeat steps 2-4, this time selecting the Target server from the “Server:” drop down
6. Next, edit /etc/default/LifeKeeper on both nodes
    a. Set CONFIRMSODEF=1 (change from 0 to 1)


root@LinuxPrimary:~
File Edit View Terminal Tabs Help
CONFIRMSODEF=1 #Default action if administrator instruction can't be obtained
# when manual intervention is configured, due to timeout or
# other failure.
# 0=proceed with failover, 1=block the failover
CONFIRMSOTO=600 #Timeout period, in seconds, for waiting for administrator
# instruction prior to starting failover, if manual
# intervention is configured.

# 11.1.7. Test Your DK for Linux Environment

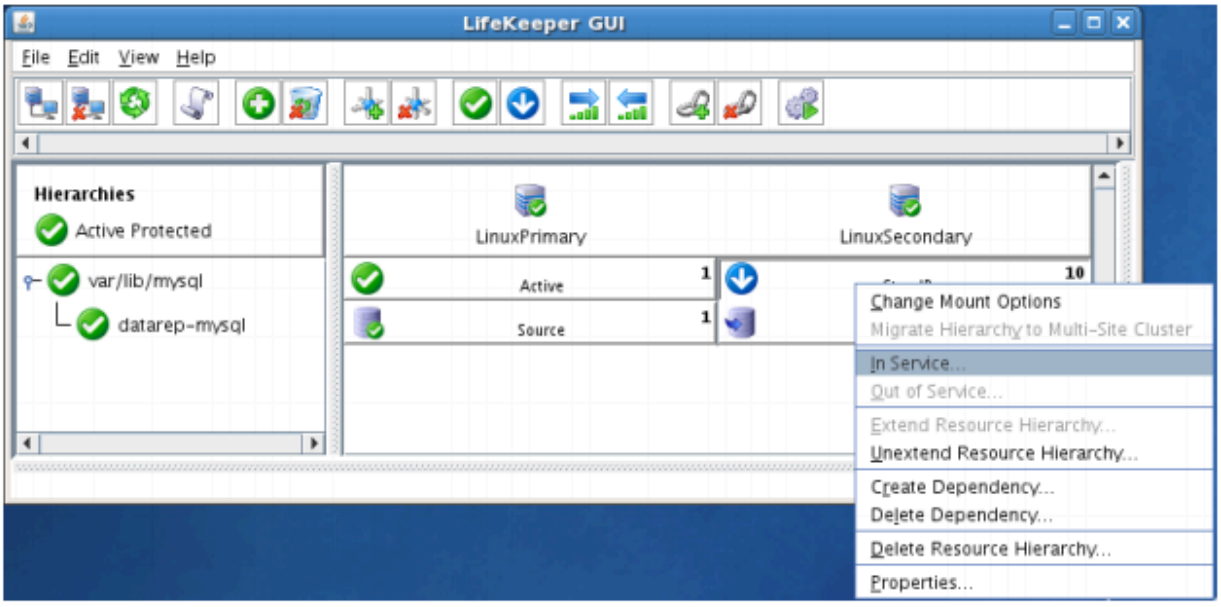
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

> **Note:** For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

1. Manual Switchover of the Mirror to Secondary Server

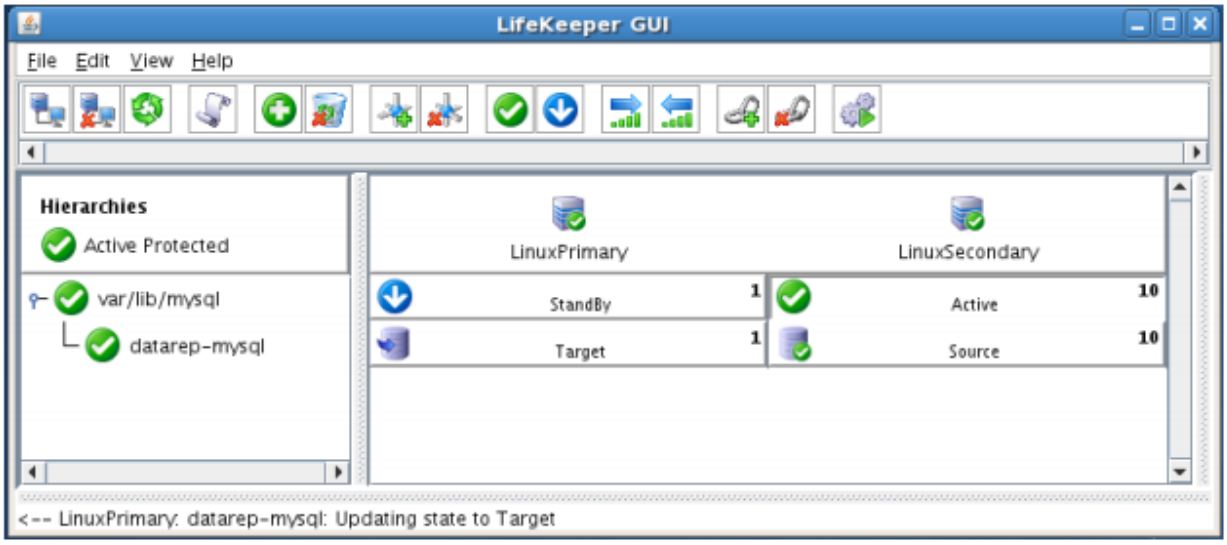
   **Procedure:**

- From the LifeKeeper GUI, right click on the top of the resource hierarchy on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



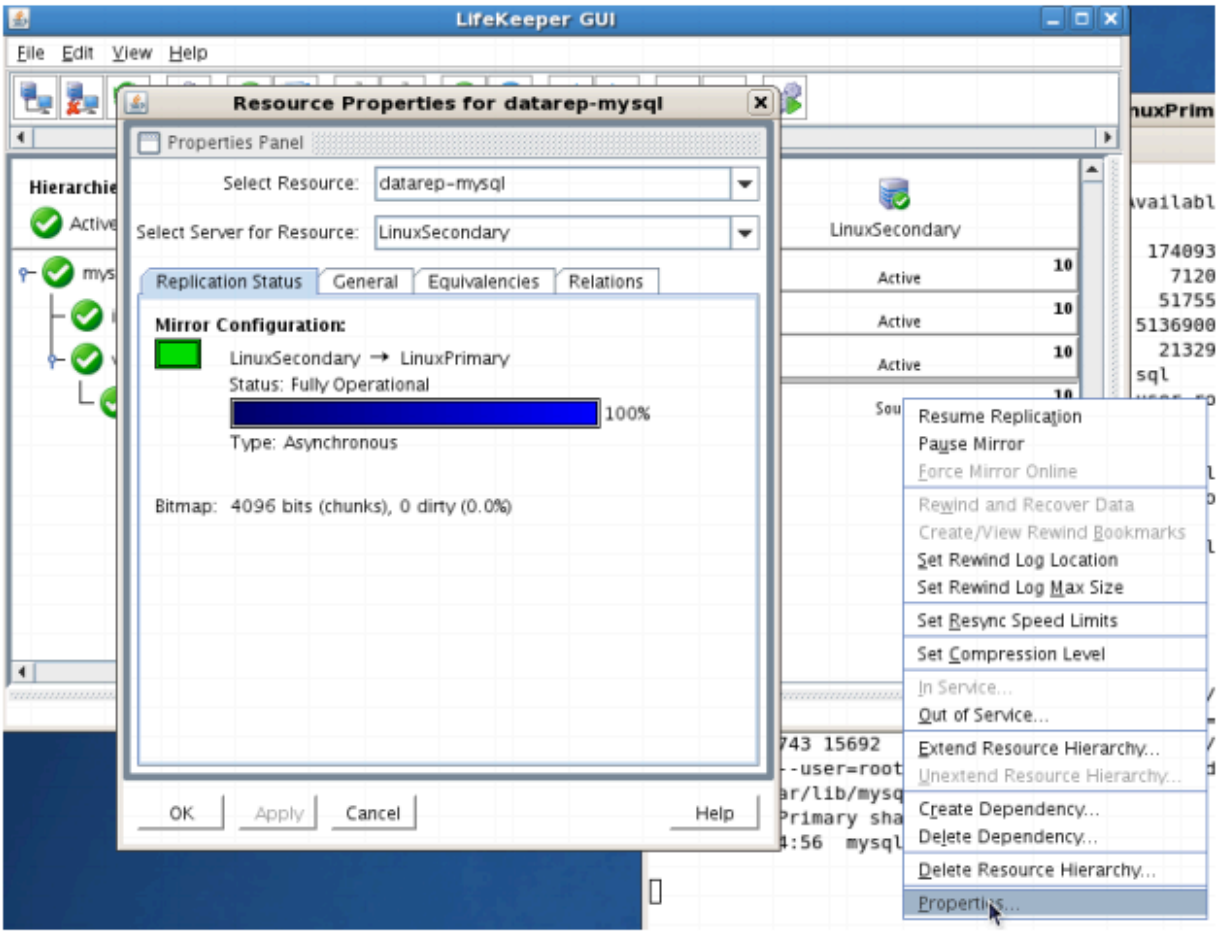

**Expected Result:**

- All resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources, all resources will be brought in service on LINUXSECONDARY.
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXSECONDARY -> LINUXPRIMARY
- At this point, all resources are now active on LINUXSECONDARY.




**Tests/Verification:**

- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties




- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXSECONDARY

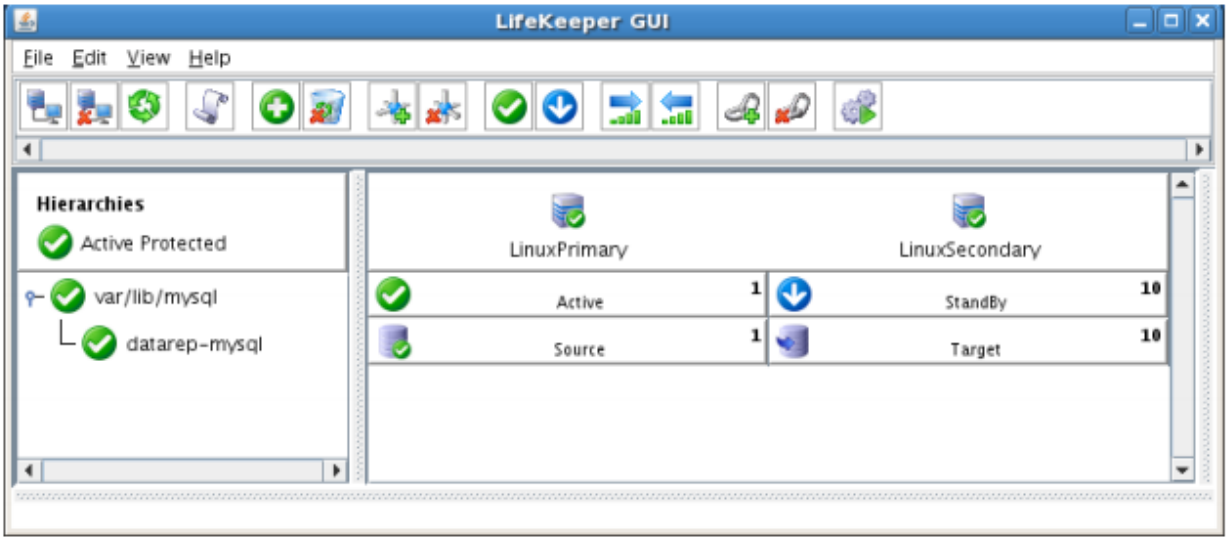
2. Manual Switchover of the Mirror back to Primary Server

**Procedure:**

- From the LifeKeeper GUI, right click on the top level of the resource hierarchy on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

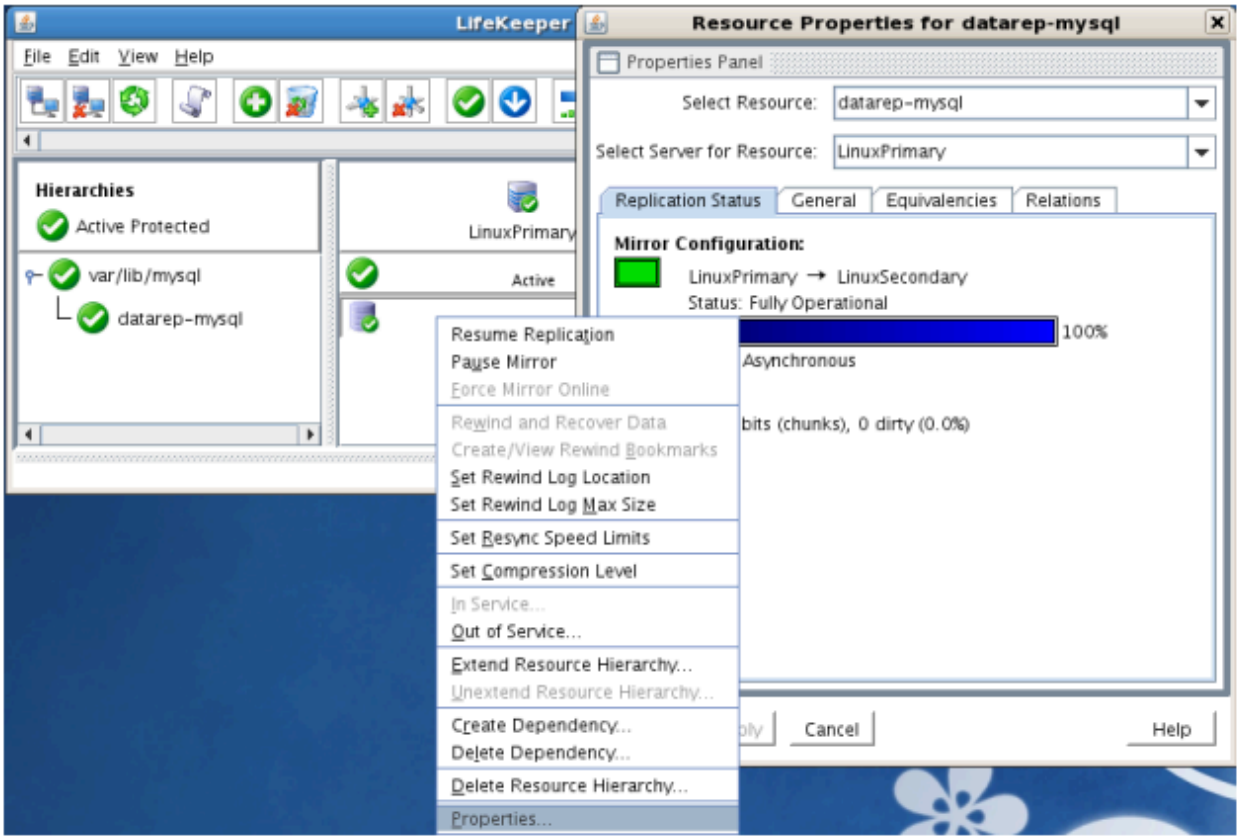
**Expected Result:**

- All resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources, all resources will be brought in service on LINUXPRIMARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXPRIMARY -> LINUXSECONDARY




**Tests/Verification:**

- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties




- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY

# 11.2. SIOS Protection Suite for Linux Evaluation Guide for Cloud Environments

This Evaluation Guide explains the basic concepts of the SIOS Protection Suite and then walks through core user interface elements to show how various resources may be protected. The guide discusses the cloud environment from several perspectives and may also be used to evaluate the SIOS Protection Suite in an on-premise environment.

# 11.2.1. Before Starting an Evaluation of SIOS Protection Suite for Linux

This section discusses general topics to consider when evaluating software bringing High Availability to mission-critical workloads.

- High Availability, RTO, and RPO
- SIOS Protection Suite for Linux – Integrated Components
- Benefits of SIOS Protection Suite for Linux
- How Workloads Should be Distributed when Migrating to a Cloud Environment
- Public Cloud Platforms and their Network Structure Differences
- How a Client Connects to the Active Node
- How does Data Replication between Nodes Work?
- What is “Split Brain” and How to Avoid It

# 11.2.1.1. High Availability, RTO, and RPO

High availability (HA) is an information technology term that refers to a computer software or component that is operational and available for more than 99.99% of the time. End users of an application, or system, experience less than 52.5 minutes per year of service interruption. This level of availability is typically achieved through the use of high availability clustering, a configuration that reduces application downtime by eliminating single points-of-failure through the use of redundant servers, networks, storage, and software.

## What are recovery time objectives (RTO) and recovery point objectives (RPO)?

In addition to 99.99% availability time, high availability environments also meet stringent recovery time and recovery point objectives. Recovery time objective (RTO) is a measure of the time elapsed from application failure to restoration of application operation and availability. It is a measure of how long a company can afford to have that application down. Recovery point objectives (RPO) are a measure of how up-to-date the data is when application availability has been restored after a downtime issue. It is often described as the maximum amount of data loss that can be tolerated when a failure happens. SIOS high availability clusters deliver an RPO of zero and an RTO of minutes.

## What is a high availability cluster?

In a high availability cluster, important applications are run on a primary server node, which is connected to one or more secondary nodes for redundancy. Clustering software, such as SIOS LifeKeeper, monitors clustered applications and dependent resources to ensure they are operational on the active node. System level monitoring is accomplished via intervallic heartbeats between cluster nodes. If the primary server fails, the secondary server initiates recovery after the heartbeat timeout interval is exceeded. For application level failures, the clustering software detects that an application is not available on the active node. It then moves the application and dependent resources to the secondary node(s) in a process called a failover, where operation continues and meets stringent RTOs.

In a traditional failover cluster, all nodes in the cluster are connected to the same shared storage, typically a storage area network (SAN). After a failover, the secondary node is granted access to the shared storage, enabling it to meet stringent RPOs.

# 11.2.1.2. SIOS Protection Suite for Linux – Integrated Components

SIOS Protection Suite includes the following software components to protect an organization's mission-critical systems.

## SIOS LifeKeeper

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

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

## SIOS DataKeeper

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

## Application Recovery Kits (ARKs)

Application Recovery Kits (ARKs) include tools and utilities that allow LifeKeeper to manage and control a specific application or service. When an ARK is installed for a specific application, LifeKeeper is able to monitor the health of the application and automatically recover the application if it fails. These Recovery Kits are non-intrusive and require no changes within the application in order for it to be protected by LifeKeeper.

There is a comprehensive library of 'off-the-shelf' Application Recovery Kits available as part of the SIOS Protection Suite portfolio. The types and quantity of ARKs supplied vary based on the edition of SIOS Protection Suite purchased.

# 11.2.1.3. Benefits of SIOS Protection Suite for Linux

- SIOS software supports multiple operating system versions and flavors (both Linux and Windows).
  - Consistent user experience to protect mission-critical resources regardless of the operating system.

- SIOS delivers high availability solutions for multiple environments including on-premises, virtualized (VMware) and cloud environments including public platforms from AWS, Azure, GCP as well as privately hosted cloud environments. The same tools can be used across different environments.
  - Hybrid environments are also supported. It is possible to configure an on-premise node as the primary location with the secondary node being hosted in a private cloud and a third node located on a public cloud platform providing an additional DR (disaster recovery) option.

- SIOS provides application-aware protection mechanisms including Application Recovery Kits (ARKs), available for the world's leading providers of enterprise applications and databases.
  - Our comprehensive library of ARKs protect the broadest range of applications 'off-the-shelf'.
  - Other 'non-standard' or legacy applications can be protected with the built-in 'GenApp' ARK or by developing a custom ARK either in-house or in collaboration with SIOS engineers.

- Unlike open-source tools (which force the user to manually establish environmental parameters in advance and then type a complex set of command-line parameters), a series of wizard based installation and configuration screens enables intuitive selection of resources requiring protection and helps in selecting the type and extent of protection to be provisioned.
  - Wizards scan the system and environment to identify the resources to protect. In most cases the operator only needs to confirm default selections and enter unique environment-specific host IDs etc. to complete the installation.
  - This reduces the likelihood of misconfiguration of the HA solution and the inevitable unexpected downtime as the result of a system or application failure.

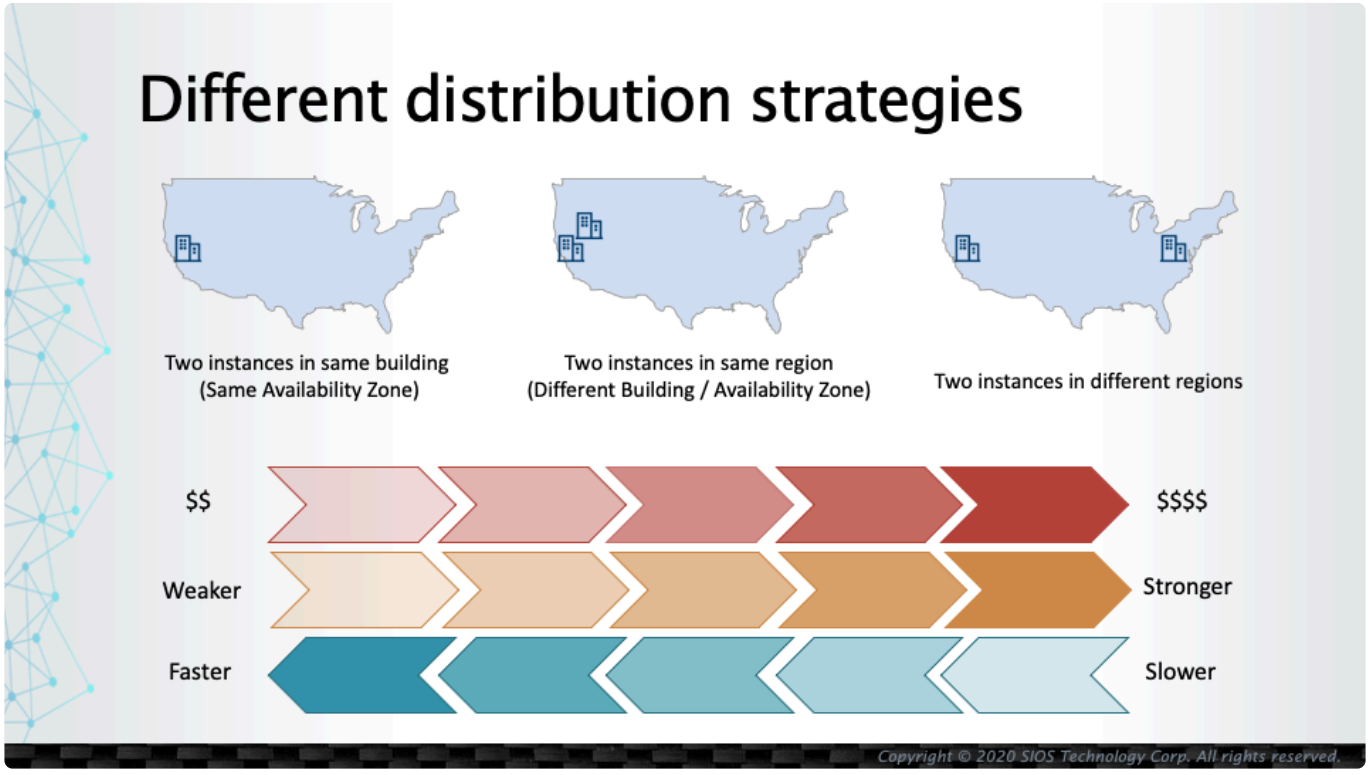
- SIOS provides a comprehensive range of technical support options including 24×7 critical support, offering support options tailored to the available budget, systems complexity or criticality of the application requiring high availability.

# 11.2.1.4. How Workloads Should be Distributed when Migrating to a Cloud Environment

Determining how Workloads (nodes) should be distributed is a common topic of discussion when migrating to the public cloud with High Availability in mind. If workloads are located within an on-premise environment, more often than not the locations of these workloads are defined by the location(s) of established datacenters. In many cases choosing another location in which to host a workload is not an available option. With a public cloud offering there are a wide range of geographical regions as well as availability zones to choose from.

An Availability Zone is generally analogous to one or more datacenters (physical locations) being located in the same physical region (e.g., in California). These datacenters may be located in different areas but are connected using high-speed networks to minimize connection latency between them. (Note that hosting services across several datacenters within an availability region should be transparent to the user).

As a general rule, the greater the physical distance between workloads, the more resilient the environment becomes. It's a reasonable assumption that natural disasters such as earthquakes won't affect different regions at the same time (e.g., both U.S. west coast and east coast at the same time). However, there is still a chance of experiencing service outages across different regions simultaneously due to system-wide failures (some cloud providers have previously reported simultaneous cross-region outages such as in the US & Australia). It may be appropriate to consider creating a DR (disaster recovery) plan defined across different cloud providers.




Another perspective worthy of consideration is the cost to protect the resources. Generally the greater

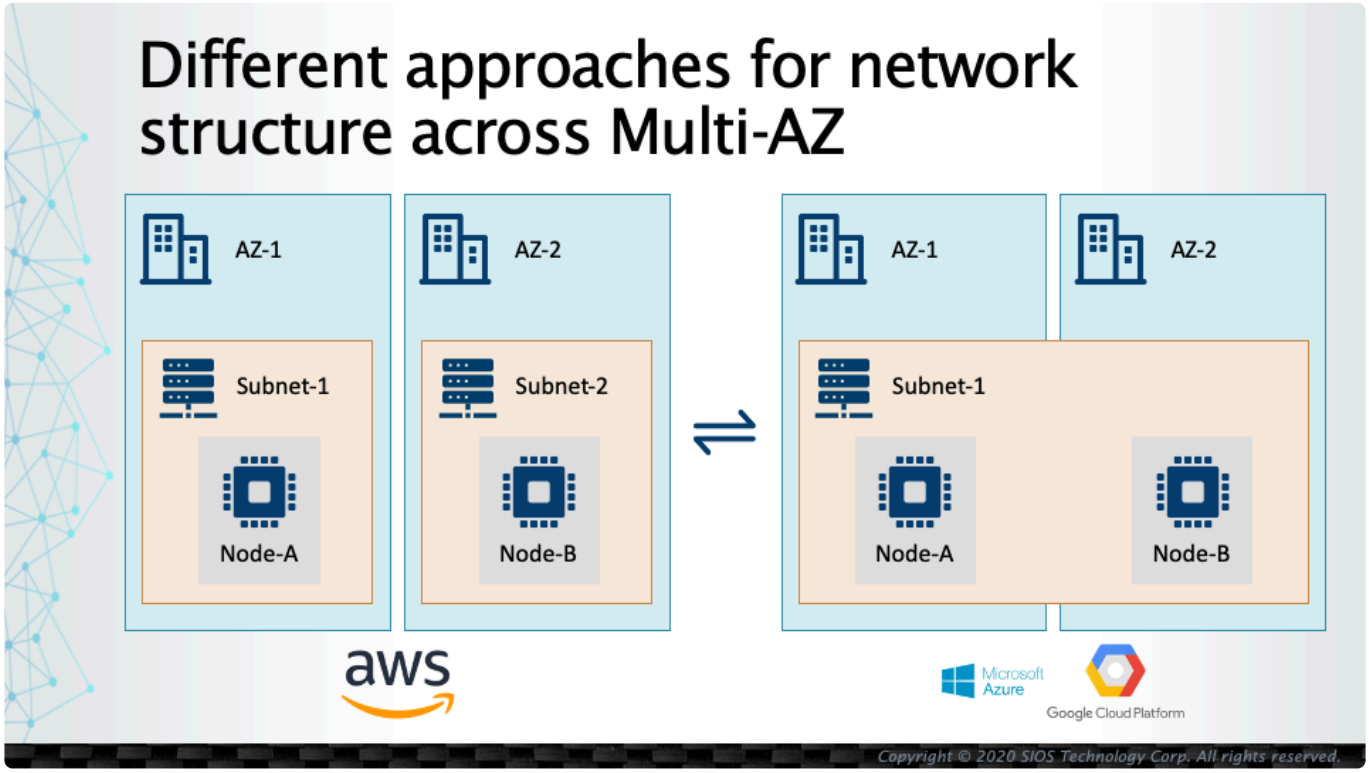
the distance between workloads, the more costs are incurred for data transfer. In many cases, data transfer between nodes within the same datacenter (Availability Zone) is free while it might costs $0.01/GB or more to transfer data across Availability Zones. This additional cost might be doubled (or more) when data is transferred across regions (i.e. $0.02 / GB). In addition, due to the increased physical distance between workloads, greater data latency between nodes should be anticipated between locations. Through consideration of these factors, generally speaking, it is recommended to distribute workloads across Availability Zones within the same Region.

# 11.2.1.5. Public Cloud Platforms and their Network Structure Differences

There are several public cloud platforms including Amazon Web Services (AWS), Microsoft Azure and Google Cloud Platform (GCP). While there are many similarities in their infrastructures, there are some differences. In many cases a VPC (Virtual Private Cloud) or a VNET (Virtual Network) that is tied to a region is created. One or more VPCs may be defined for a logical group of applications. By so doing, different systems are divided into separate unconnected networks unless different VPCs are specifically connected.

Under a VPC many different subnets can be defined. Based on the purpose, some subnets are configured as “public” subnets which are accessible to the internet and some are configured as “private” subnets which are not accessible to the internet.

Some cloud providers (such as Azure and GCP) allow subnets to be defined across Availability Zones (different datacenters), while some (such as AWS) do not allow subnets to be defined across Availability Zones. In the latter case, a subnet will need to be defined for each Availability Zone.




In this guide, we’ll use different Availability Zones for each node. Once the basic functionality of the SIOS product is understood, it might be appropriate to explore different scenarios (similar to those in use in your own network infrastructure) that involve distributing workloads across different subnets, modifying the IP ranges for these subnets, changing the manner in which the network is connected to the Internet, etc.

# 11.2.1.6. How a Client Connects to the Active Node

As discussed earlier, once a High Availability cluster has been configured, two or more nodes run simultaneously and users connect to the “active” node. When an issue occurs on the active node, a “failover” condition occurs and the “standby” node becomes the new “active” node. When a failover occurs there must be a mechanism that either allows a client to detect the failover condition and to reconnect, or a seamless transfer of the user’s active client session to the active node.

## A Virtual IP Address

Usually a “virtual” IP address is created when a cluster is configured and the client communicates with the active node using a virtual IP address. When a failover occurs, the virtual IP address is reassigned to the new active node and the client reconnects to the same virtual IP address.

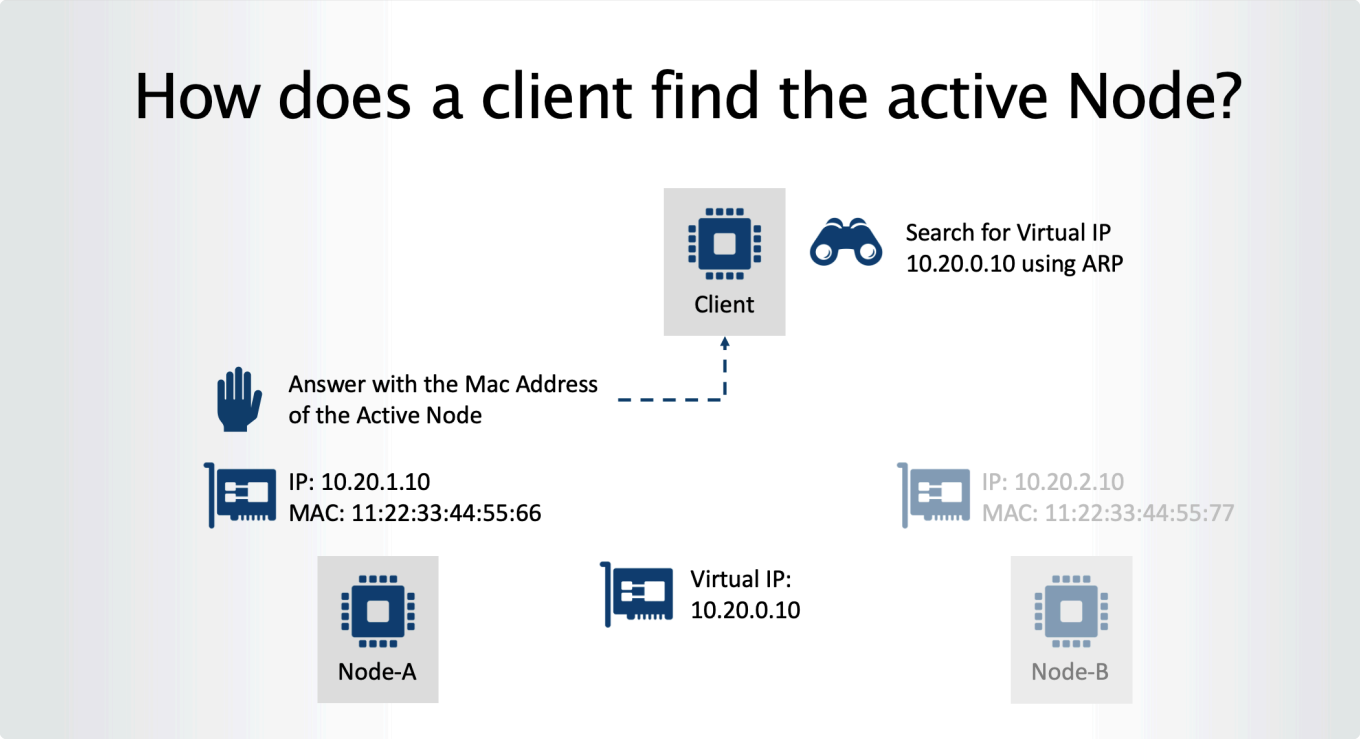
As an example, let us assume that there are two nodes, A and B, with IP addresses of `10.20.1.10` and `10.20.2.10`. In this example, we will define a virtual IP address of 10.20.0.10 which should be considered to be assigned to the current active node.

This is similar to assigning a second IP address to one network interface card on one node. If the command `ip a` is entered on the active node, both IP addresses will appear (as on lines 10 and 12 in this Linux example):

```
1  [root@node-a default]# ip a
2  1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
3      link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
4      inet 127.0.0.1/8 scope host lo
5         valid_lft forever preferred_lft forever
6      inet6 ::1/128 scope host
7         valid_lft forever preferred_lft forever
8  2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
9      link/ether 00:0d:3a:6e:1f:74 brd ff:ff:ff:ff:ff:ff
10     inet 10.20.1.10/22 brd 10.20.3.255 scope global noprefixroute eth0
11        valid_lft forever preferred_lft forever
12     inet 10.20.0.10/22 scope global secondary eth0
13        valid_lft forever preferred_lft forever
14     inet6 fe80::20d:3aff:fe6e:1f74/64 scope link
15        valid_lft forever preferred_lft forever
```

## The ARP Protocol

When a client attempts to find a server using an IP address, the client typically uses ARP (Address Resolution Protocol) to find the MAC (Media Access Control) address of the target machine.




Once a client broadcasts a message to find the target IP address, the active node answers with its MAC address and the client resolves the request and connects to it.

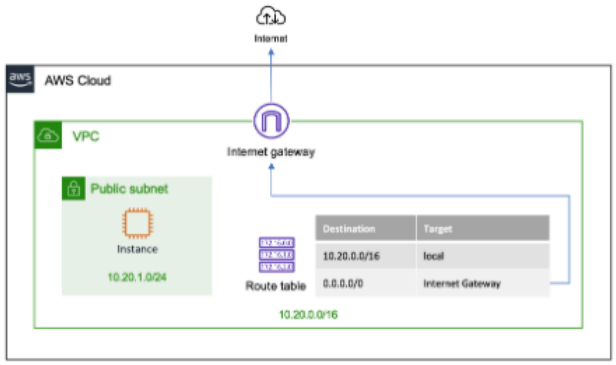
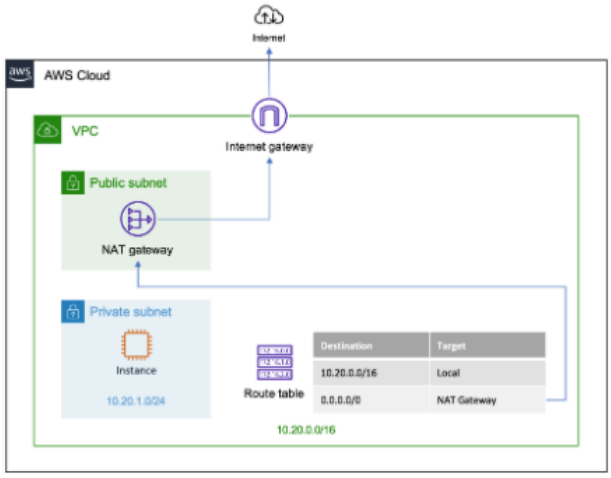
## ARP Alternatives for the Cloud Environment

In the cloud environment, however, it is not possible to identify the active node using ARP as many layers are abstracted in the virtual environment. An alternative method based on the network infrastructure in use in the specific cloud environment may be required. There are normally several options, and a selection should be made from the following list.

- AWS Route Table Scenario
- AWS Elastic IP Scenario
- AWS Route53 Scenario
- Azure Internal Load Balancer Scenario
- GCP Internal Load Balancer Scenario

# 11.2.1.6.1. AWS Route Table Scenario

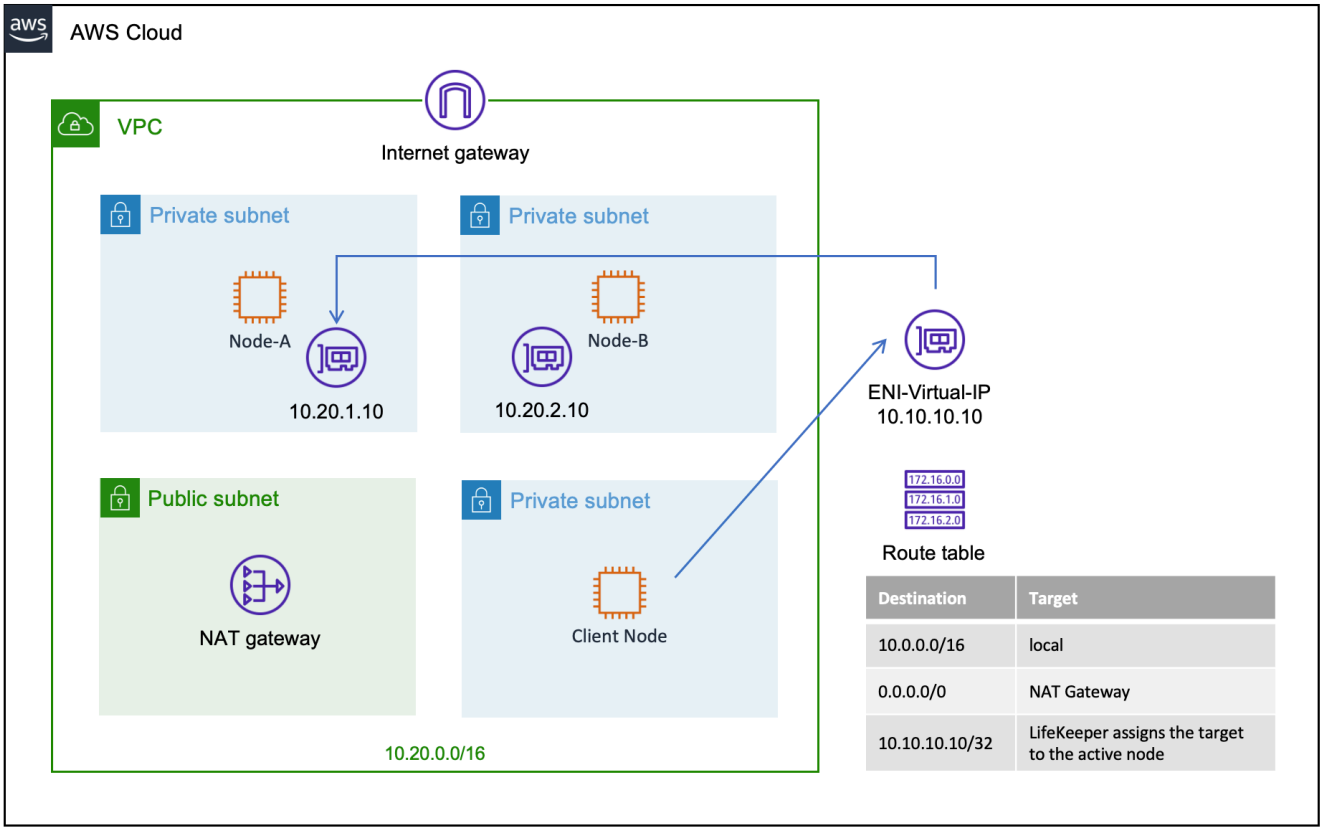
When an instance is created on AWS, one of the first steps is to create a Route Table. The Route Table defines the route for each Destination. In the following diagram the Instance has two routes, one for `10.20.0.0/16`, which is the VPC boundary (local network). Any traffic within `10.20.0.0/16` is considered a local connection. The other route is for `0.0.0.0/0` which will allow IP traffic to route outside the local VPC boundary (to other VPCs or the Internet). To connect to the internet either an Internet Gateway is needed (in the case of a public subnet) or a NAT Gateway should be specified (in the case of a private subnet).

| When Instances are located in Public Subnet | When Instances are located in Private Subnet |
|---|---|
|  |  |



## Finding the Active Node within the VPC

When it comes to finding an active node in a clustering environment, we must define a virtual IP address ***outside of the VPC CIDR*** so that a ***route*** can be defined. In fact, defining a Virtual IP means ***creating a new Destination*** in the Route table.

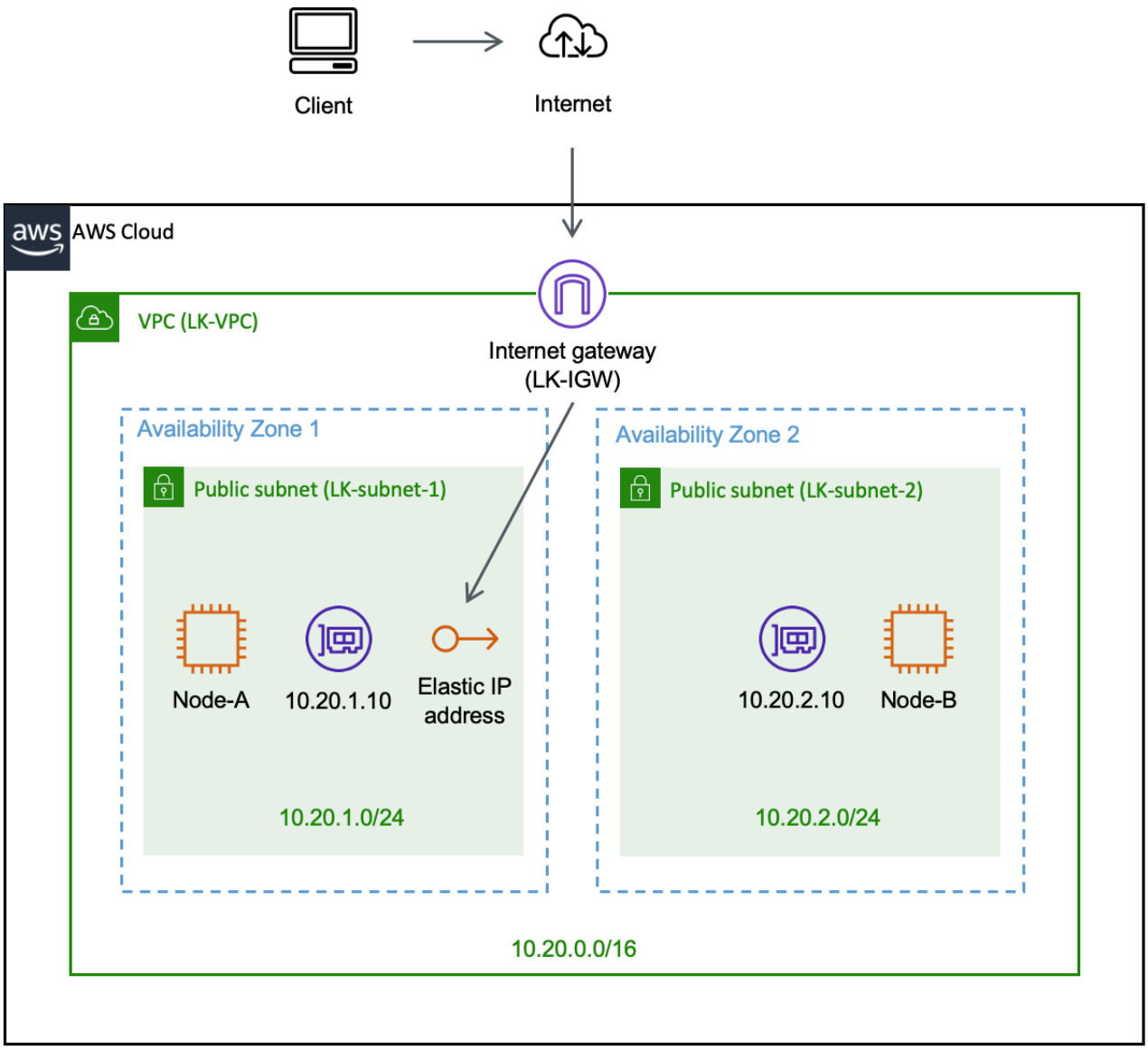



Once a new destination outside of VPC CIDR has been defined, SIOS Protection Suite can change the target of this route to the active node (either `10.20.1.10` or `10.20.2.10` in the example configuration shown above). A client node should simply look for `10.10.10.10` as the destination and the routing table guides the traffic to the active node. The routing table is updated dynamically by SIOS's Recovery Kit for EC2.

The technical details of how this scenario works are discussed later in Creating an AWS EC2 Resource (RouteTable Scenario).

# 11.2.1.6.2. AWS Elastic IP Scenario

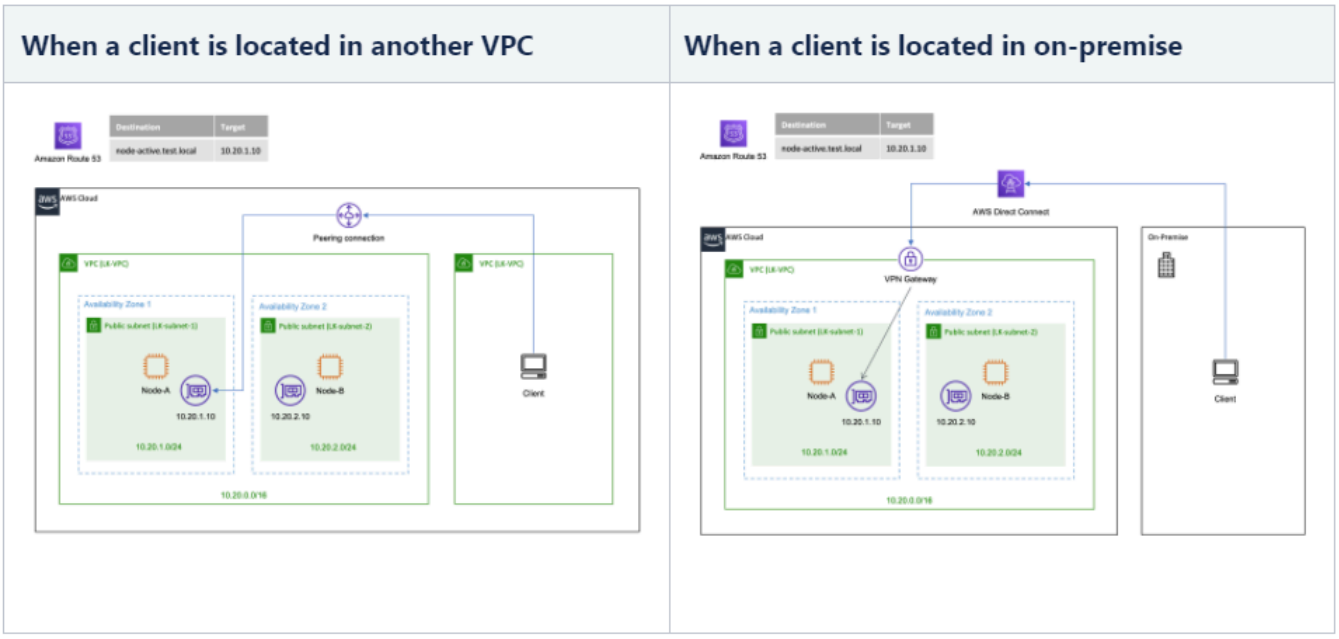
If clients are located on the internet and require public access to the active cluster node, the Elastic IP scenario should be selected. In this example the SIOS Recovery Kit for EC2 switches the Elastic IP when switching between nodes and attaches the Elastic IP to the ENI (Elastic Network Interface) of the new active node.




The technical details of how this scenario works are discussed later in Creating AWS EC2 Resource (Elastic IP Scenario).

# 11.2.1.6.3. AWS Route53 Scenario

The previous sections discussed the Route Table Scenario when the client is located within the same VPC as the target node. However, if the client node is located within another VPC or located in an on-premise environment (connected to AWS via Direct Connect or VPN), the route table scenario cannot be used to specify the active target node. In this case Route53 may be used to route client traffic to the active target node.




The Route53 Recovery Kit (provided as part of SIOS Protection Suite) can update the DNS entry allowing the client to connect to the active node as long as it can connect to the node via either VPC peering or Direct Connect.

> The technical details of how this scenario works are discussed later in Creating AWS Route53 Resource.

# 11.2.1.6.4. Azure Internal Load Balancer Scenario

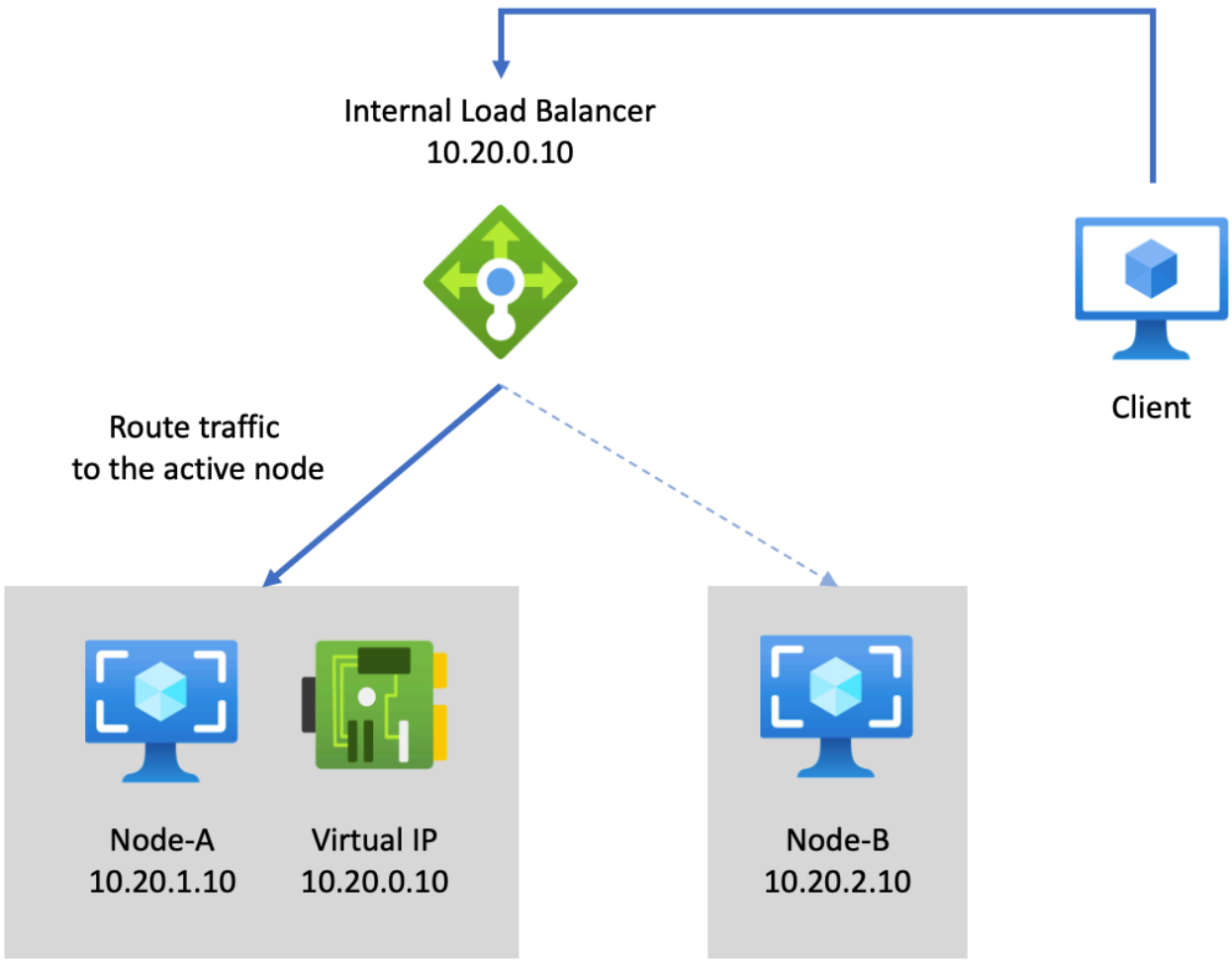
On Azure, using a Virtual IP may not enable traffic to be resolved to the active node as many infrastructure layers are abstracted in the cloud environment. To resolve this issue, an Internal Load Balancer should be used to forward traffic to the active node.

As an example, let us assume there are two nodes (`10.20.1.10` and `10.20.2.10`), working as a cluster and a virtual IP `10.20.0.10` is created that points to the active node.

In this case, an Azure Load Balancer is created with a Frontend IP address of `10.20.0.10` and a Virtual IP address with the same IP address `10.20.0.10` is created.

This ensures the client always connects to the load balancer first via the Virtual IP address `10.20.0.10`.

The load balancer uses a health probe to determine which node is currently hosting the application, and all client connections are forwarded to that active node.



The technical details of how this scenario works are discussed later in Azure – Using Internal Load Balancer.

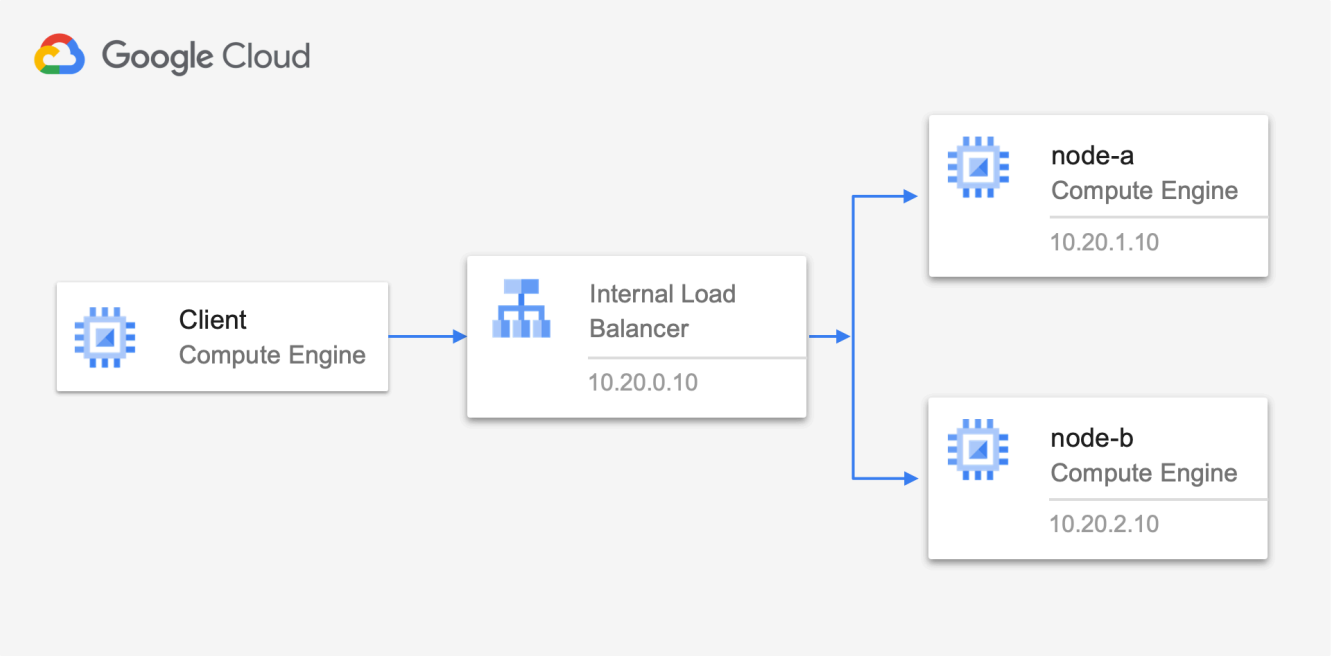
# 11.2.1.6.5. GCP Internal Load Balancer Scenario

On GCP, using a Virtual IP may not enable traffic to be resolved to the active node as many infrastructure layers are abstracted in the cloud environment. To resolve this issue, an Internal Load Balancer should be used to forward traffic to the active node.

As an example, let us assume that there are two nodes (`10.20.1.10` and `10.20.2.10`), working as a cluster and a virtual IP `10.20.0.10` is created that points to the active node.

In this case, an Internal Load Balancer is created with a Frontend IP address of `10.20.0.10`.

The load balancer uses a health probe to determine which node is currently hosting the application, and all client connections are forwarded to that active node.




The technical details of how this scenario works are discussed later in GCP – Using an Internal Load Balancer.

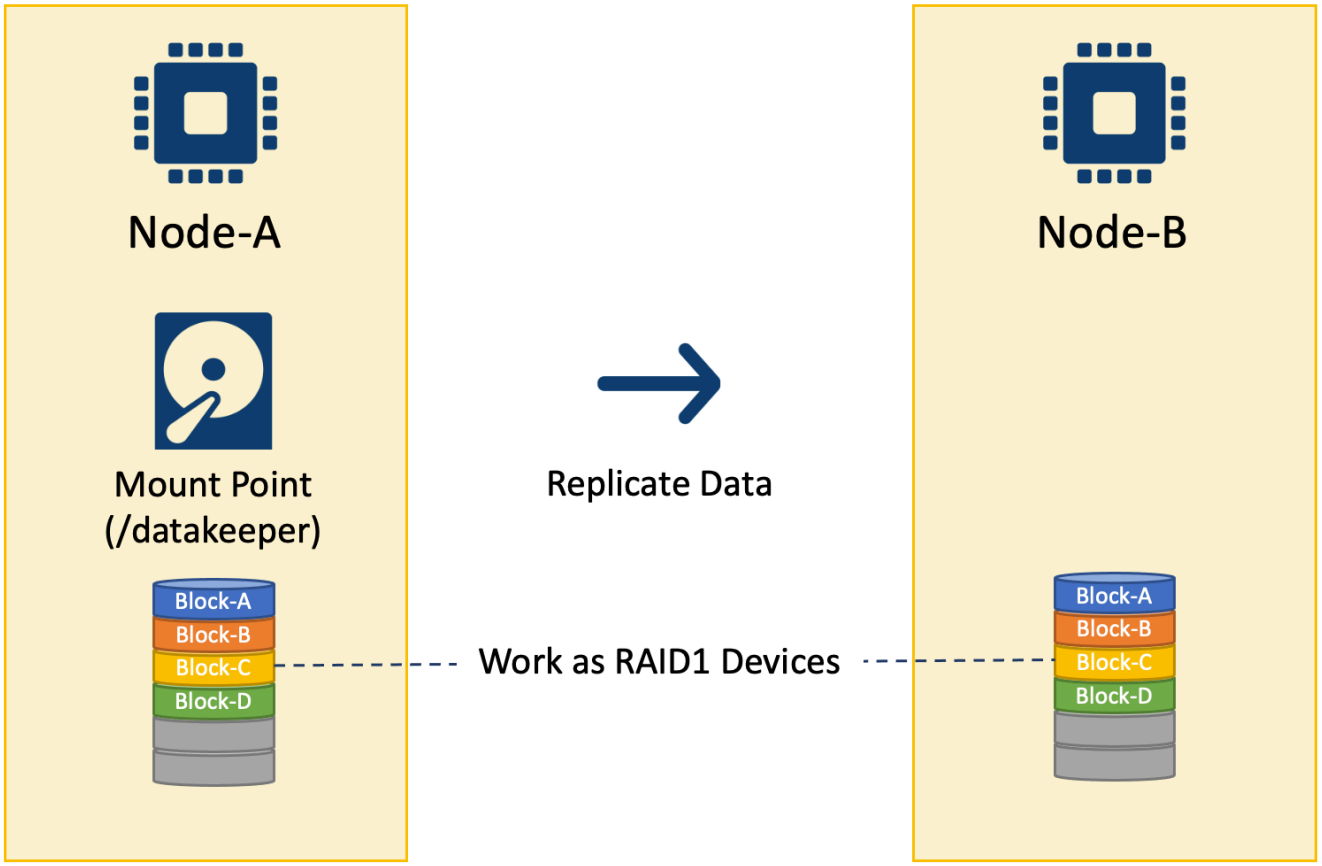
# 11.2.1.7. How does Data Replication between Nodes Work?

In the traditional datacenter scenario, data is commonly stored on a storage area network (SAN). The cloud environment doesn't typically support shared storage.

SIOS DataKeeper presents 'shared' storage using replication technology to create a copy of the currently active data. It creates a NetRAID device that works as a RAID1 device (data mirrored across devices).

Data changes are replicated from the Mirror Source (disk device on the active node – Node A in the diagram below) to the Mirror Target (disk device on the standby node – Node B in the diagram below).

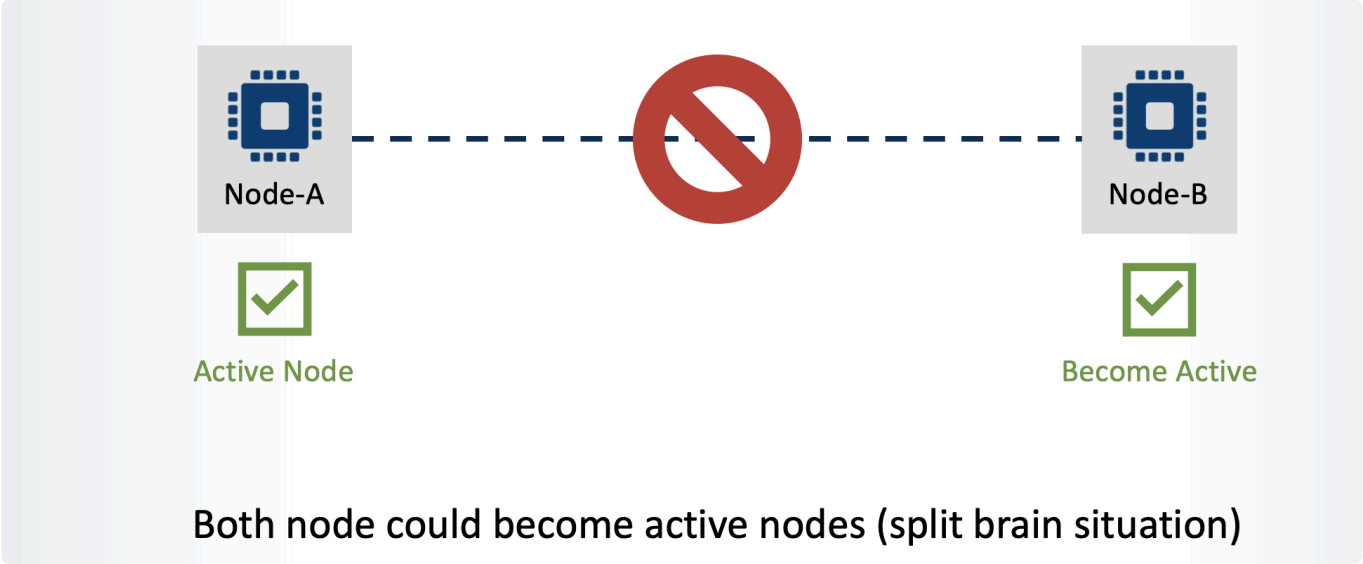
In order to guarantee consistency of data across both devices, only the active node has write access to the replicated device (/datakeeper mount point in the example below). Access to the replicated device (the /datakeeper mount point) is not allowed while it is a Mirror Target (i.e., on the standby node).



# 11.2.1.8. What is “Split Brain” and How to Avoid It

As we have discussed, in a High Availability cluster environment there is one active node and one or more standby node(s) that will take over service when the active node either fails or stops responding.

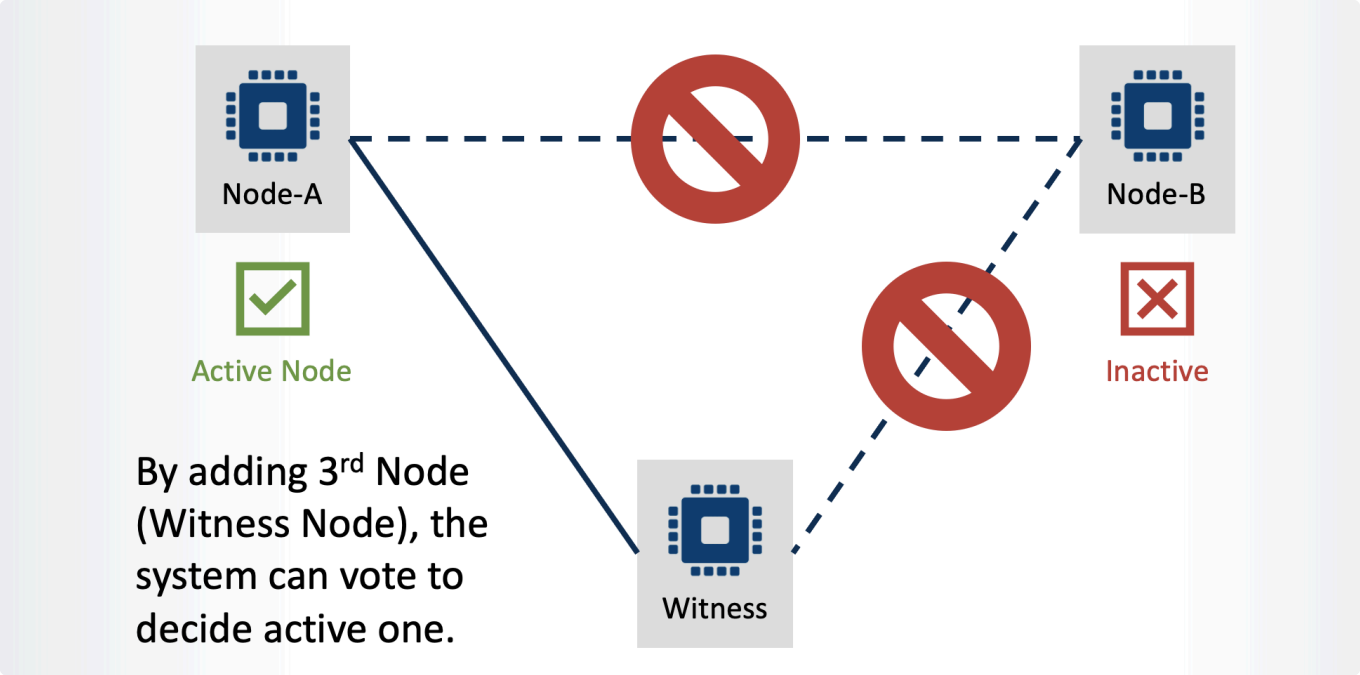
This sounds like a reasonable assumption until the network layer between the nodes is considered. What if the network path between the nodes goes down?




Neither node can now communicate with the other and in this situation the standby server may promote itself to become the active server on the basis that it believes the active node has failed. This results in both nodes becoming ‘active’ as each would see the other as being dead. As a result, data integrity and consistency is compromised as data on both nodes would be changing. This is referred to as **“Split Brain”**.

To avoid a split brain scenario, a Quorum node (also referred to as a ‘Witness’) should be installed within the cluster. Adding the quorum node (to a cluster consisting of an even number of nodes) creates an odd number of nodes (3, 5, 7, etc.), with nodes voting to decide which should act as the active node within the cluster.

In the example below, the server rack containing Node B has lost LAN connectivity. In this scenario, through the addition of a 3rd node to the cluster environment, the system can still determine which node should be the active node.




Quorum/Witness functionality is included in the SIOS Protection Suite. At installation, Quorum / Witness is selected on all nodes (not only the quorum node) and a communication path is defined between all nodes (including the quorum node).

The quorum node doesn't host any active services. Its only role is to participate in node communication in order to determine which are active and to provide a 'tie-break vote' in case of a communication outage.

SIOS also supports IO Fencing and Storage as quorum devices, and in these configurations an additional quorum node is not required.

# 11.2.2. Documentation Style Used in this Guide

Most of the steps documented in this tutorial describe an environment consisting of two nodes (node-a and node-b). Examples of user command execution are shown within the guide. The sections below show the styles used to indicate the different nodes or users used to execute commands.

## Nodes to Execute Actions

When a particular set of command needs to be executed on a specific node, the prompt contains the user-name & host-name as follows:

```
[root@node-a]# <command name and its parameters are listed here>
```

An in-line comment similar to the below will also be shown:

> You need to execute the following steps on the primary node (node-a).

If the name of the node is not specified in the prompt, those commands need to be executed on both nodes (see the following example):

```
# <command name and its parameters are listed here>
```

An in-line comment similar to the below will also be shown:

> You need to execute the following steps for each node.

## Users to execute commands

Within the guide, the user who executes the command is identified by the prompt used.

If the command is executed as a regular user (e.g., `ec2-user` for most of the EC2 Instances), the prompt is shown as:

```
$ date
Thu Nov 26 22:58:22 UTC 2020
```

Once you become the `root` user, the prompt is described as `#`

```
$ sudo su -
# shutdown -h now
```

If a specific user needs to execute a command, the name of the user is included in the prompt as follows:

```
[oracle]$ passwd
```

## Option to Select while Working with the SIOS GUI (Wizards)

One of the benefits of using SIOS Protection Suite is that wizards scan the system and then provide suggested values within the installation screens. In most cases, it is only necessary to confirm and select the default values. Where default values are chosen, the guide indicates them with a checkmark . If no checkmark is shown, a specific value (such as an environment-specific IP address) may need to be entered.

# 11.2.3. Configuring Network Components and Creating Instances

This section outlines the computing resource required for each node, the network structure and the process required to configure these components.

- Network Structure Used in This Tutorial
- Computing Resources Used in This Tutorial

In the following sections we will configure these network components and create instances on the cloud platforms.

- Creating an Instance on AWS from Scratch
- Creating an Instance on Azure from Scratch
- Creating an Instance on GCP from Scratch

# 11.2.3.1. Network Structure Used in this Tutorial

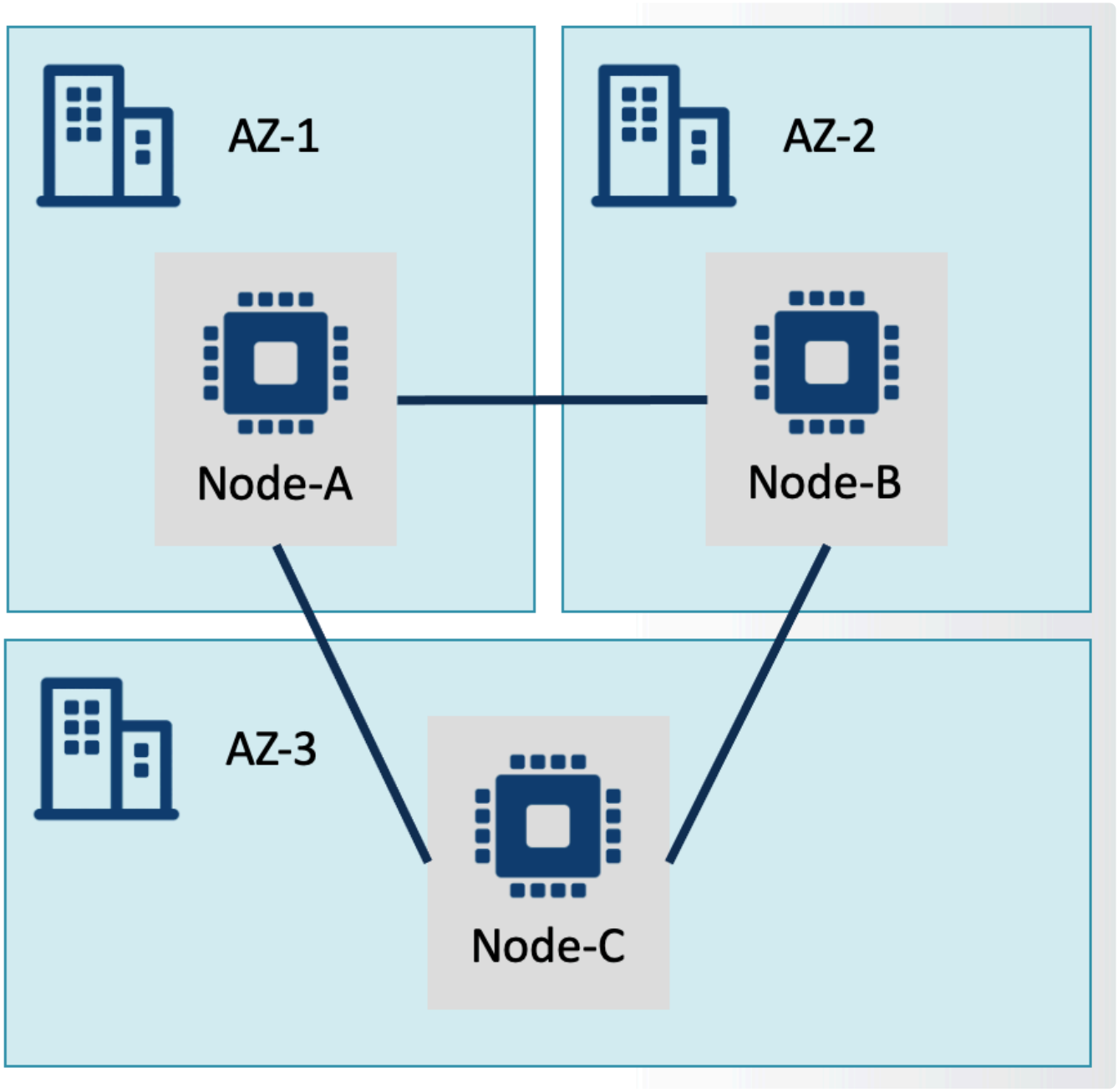
In this tutorial three nodes will be created. Node-A and Node-B will be used as the Active node and Standby node, respectively. In addition to these two nodes, Node-C (the Quorum/Witness node), will also be created. (See What is for an explanation of the need for a Quorum/Witness Node).

As discussed in How Workloads Should be Distributed when Migrating to a Cloud Environment, SIOS strongly recommends the use of split workloads within different Availability Zones (‘datacenters’).

> ✻ Although this Evaluation Guide discusses the Quorum / Witness Node (node-c), this third node may be skipped to simplify the learning experience. However, for a production environment (or a formal PoC project), SIOS strongly recommends the use of the Quorum/Witness node.

In the following topics, nodes in different Availability Zones will be created.

- Creating an Instance in AWS from Scratch
- Creating an Instance in Azure from Scratch
- Creating an Instance in GCP from Scratch


AZ-1
AZ-2
Node-A
Node-B
AZ-3
Node-C

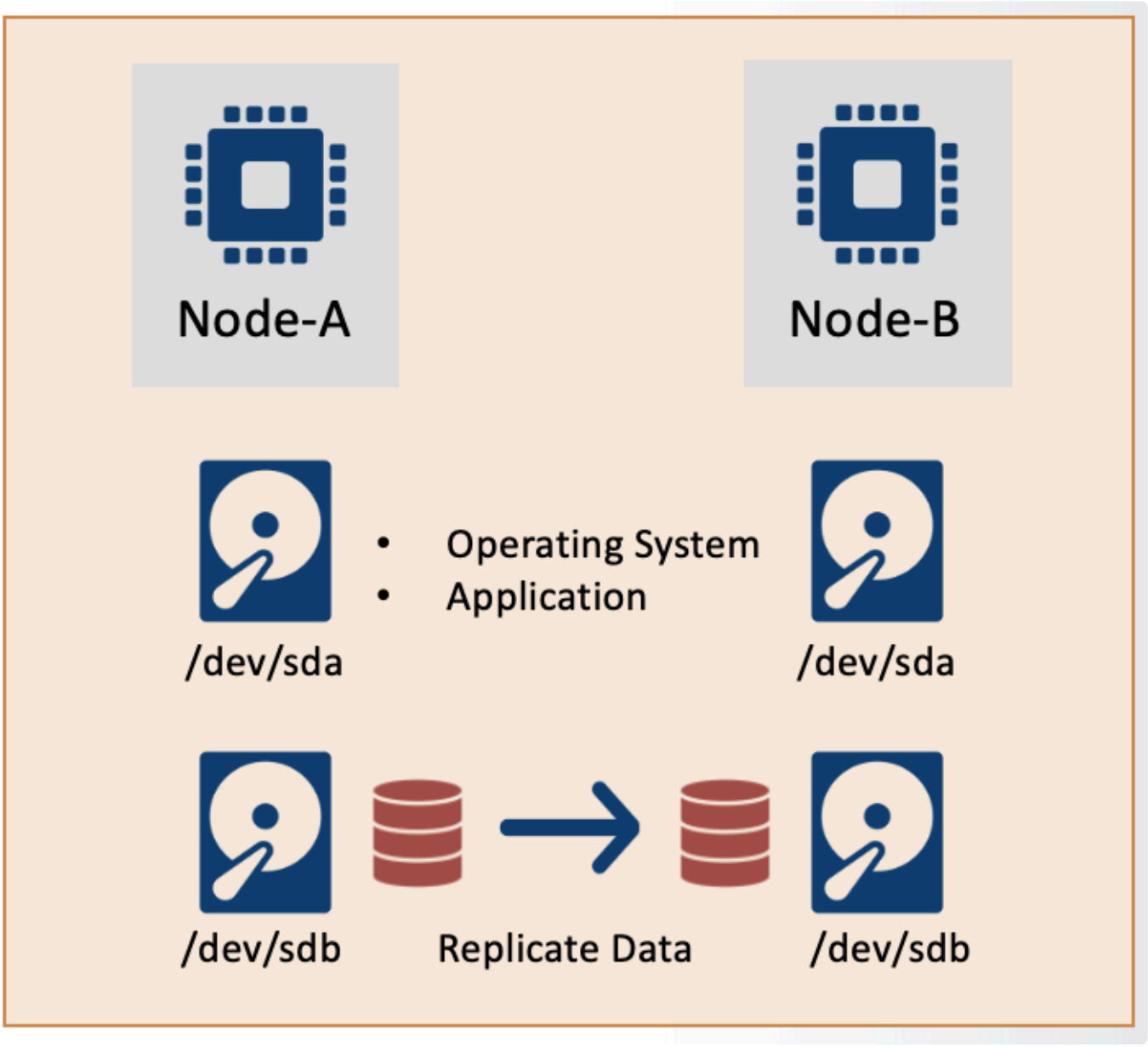
# 11.2.3.2. Computing Resources Used in this Tutorial

As discussed earlier, two nodes will be configured in a network to complete tasks in this tutorial.

Please refer to the SIOS Protection Suite for Linux Release Notes as well as the resource plan for the resource(s) to be protected (e.g., Oracle). Check to ensure that the correct operating system is selected and that the appropriate resources (e.g., disk space, memory, CPU, etc.) are available.

> **Note:** Although this guide focuses on clusters that use local replicated storage, LifeKeeper clusters can also be built using certified shared storage solutions. If shared storage is to be leveraged in the cluster please refer to Storage and Adapter Configuration.

It is important to understand that two disks will be needed for each node. The first disk is used to boot the system (this has an operating system and the applications including SIOS Protection Suite installed), while the other disk will be configured to replicate data from the “active” node to the “standby” node (note that the direction of data replication will change once a “switchover” occurs).




**Note:** The device name such as */dev/sda* may vary based on the environment.

This guide assumes that the following resources are attached to each of the nodes:

| Resource | Minimum Requirement | Evaluating with Oracle |
|---|---|---|
| Storage device for operating system & installing applications | 10 Gb | 30 Gb |
| Storage device for data replication between nodes | 8 Gb | 20 Gb |
| Memory | 1 Gb | 2 Gb |

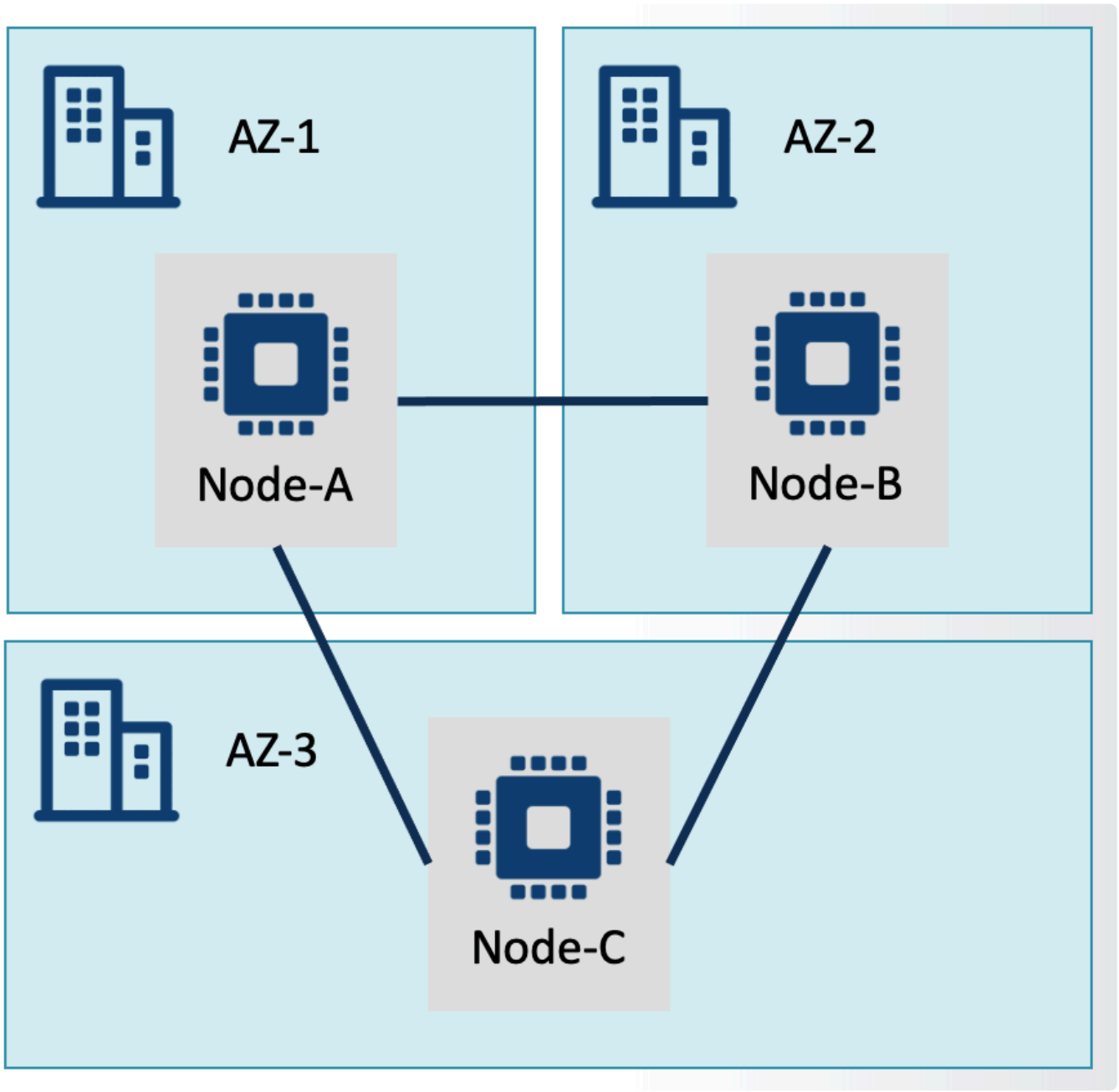
As SIOS is not able to make resource recommendations, please refer to the application

vendor (e.g., SAP) in order to ensure that the correct resources are provisioned to run a particular application.

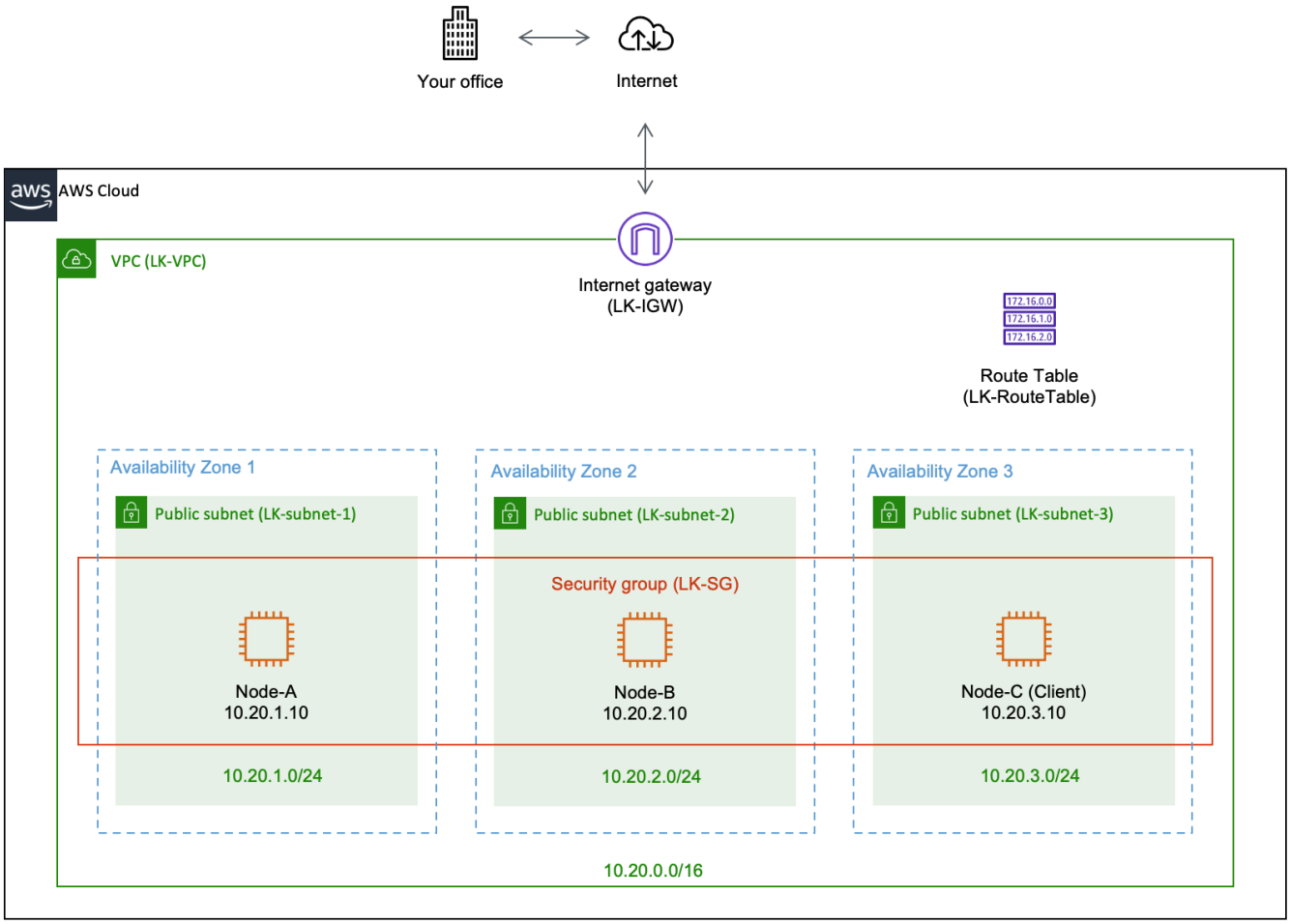
# 11.2.3.3. Creating an Instance in AWS from Scratch

**Disclaimer:** The user interface may vary between regions or change over time. Please refer to the documentation provided by AWS if screenshots shown below are different from your experience.

This Evaluation Guide uses the following network structure and instances.




In AWS, these components can be defined as per the diagram below. One important aspect in AWS is that you need to create a subnet for each Availability Zone. Because the instances are distributed on different Availability Zones, these instances are assigned to different subnets.




Components listed in this diagram are listed in the following table:

| Component | Name | Parameter | Value |
|---|---|---|---|
| VPC | LK-VPC | IPv4 CIDRs | 10.20.0.0/16 |
| Subnet | **Common values across subnets** | **VPC assignment** | **LK-VPC** |
| | | **Auto-assign public IP** | **Yes** |
| | LK-subnet-1 | IPv4 CIDR | 10.20.1.0/24 |
| | LK-subnet-2 | IPv4 CIDR | 10.20.2.0/24 |
| | LK-subnet-3 | IPv4 CIDR | 10.20.3.0/24 |
| Internet Gateway | LK-IGW | VPC association | LK-VPC |
| Route Table | LK-RouteTable | Subnet association | LK-public-subnet |
| | | Destination=10.20.0.0/16 | local |
| | | Destination=0.0.0.0/0 | LK-IGW |
| Security Group | LK-SG | Type=All traffic | Source=LK-SG |
| | | Type=SSH | Source=Your |

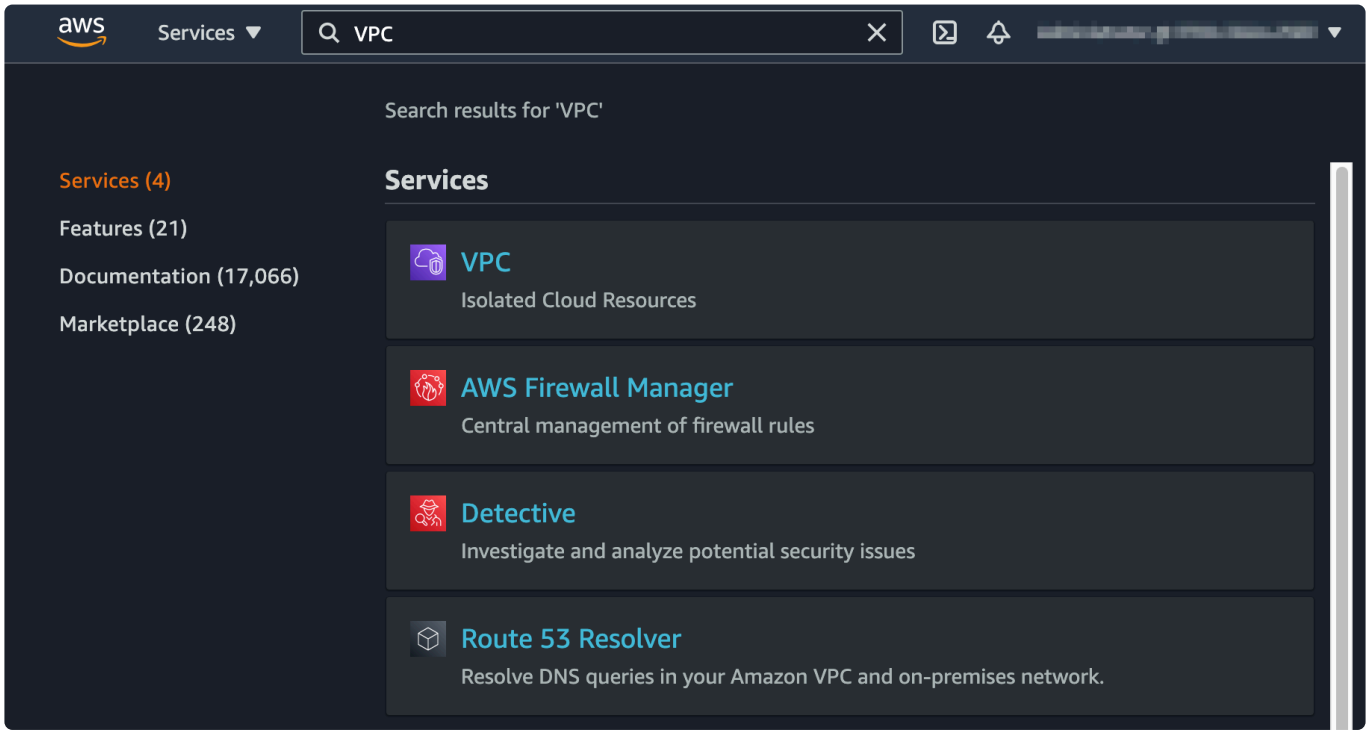
| | | | |
|---|---|---|---|
| | | | Office's WAN IP |
| Instances | **Common values across instances** | **VPC** | **LK-VPC** |
| | | **Security Group** | **LK-SG** |
| | | **Source/Dest Checking on a network interface (ENI)** | **Disabled** |
| | node-a | Subnet | LK-subnet-1 |
| | | Private IP Address | `10.20.1.10` |
| | node-b | Subnet | LK-subnet-2 |
| | | Private IP Address | `10.20.2.10` |
| | node-c | Subnet | LK-subnet-3 |
| | | Private IP Address | `10.20.3.10` |

The following sections step through the processes necessary for creating these network components, beginning with the first instance.

- Switching between AWS Services
- Deciding on an AWS Region
- Creating the VPC
- Creating a Subnet
- Creating an Internet Gateway and Assigning it to the VPC
- Creating the Route Table
- Creating a Security Group
- Creating the First EC2 Instance
- Creating the Second and Third Instances

# 11.2.3.3.1. Switching between AWS Services

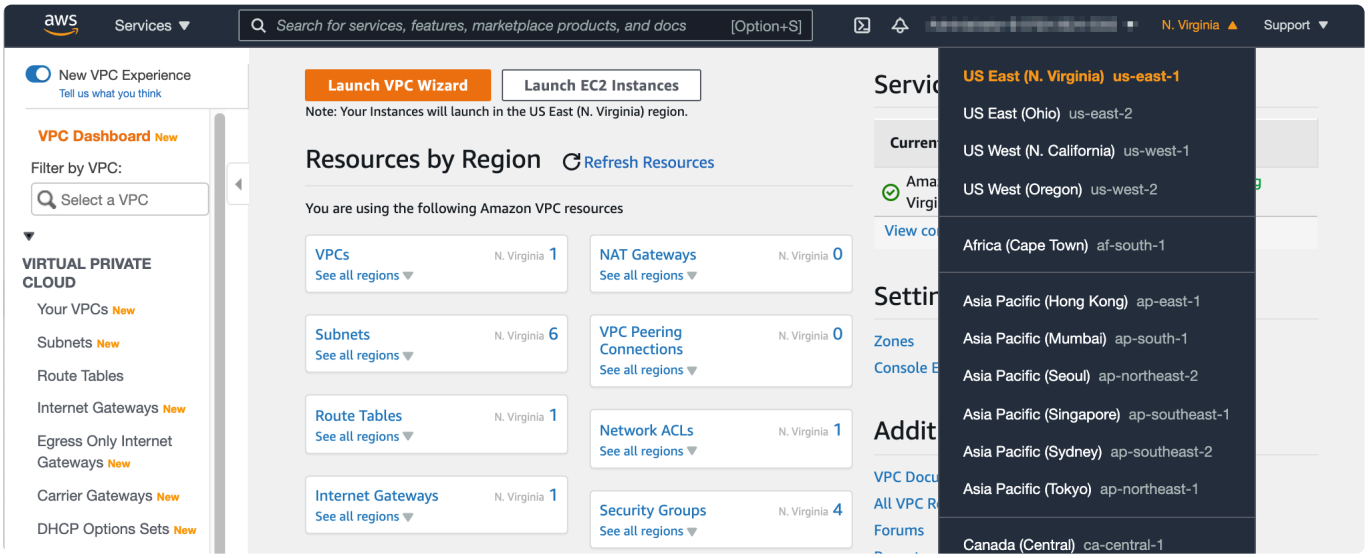
AWS has a significant and growing number of services available. To move to a different service you can type in the name of the service in the search box at the top of the AWS console. Select the service from the list. The screenshot below shows the VPC (Virtual Private Cloud) service as a selection choice.



# 11.2.3.3.2. Deciding on an AWS Region

AWS has regions in many geographic locations. It may be beneficial to select a region in close geographic proximity to the workplace location. Note that not all regions have 3+ Availability Zones, so exercise caution when making a selection in order to guarantee that the chosen region supports the intended configurations. When VPC is selected from the service menu, the current region can be seen at the top right corner of the AWS console.

Refer to the AWS documentation for more information.



# 11.2.3.3.3. Creating the VPC

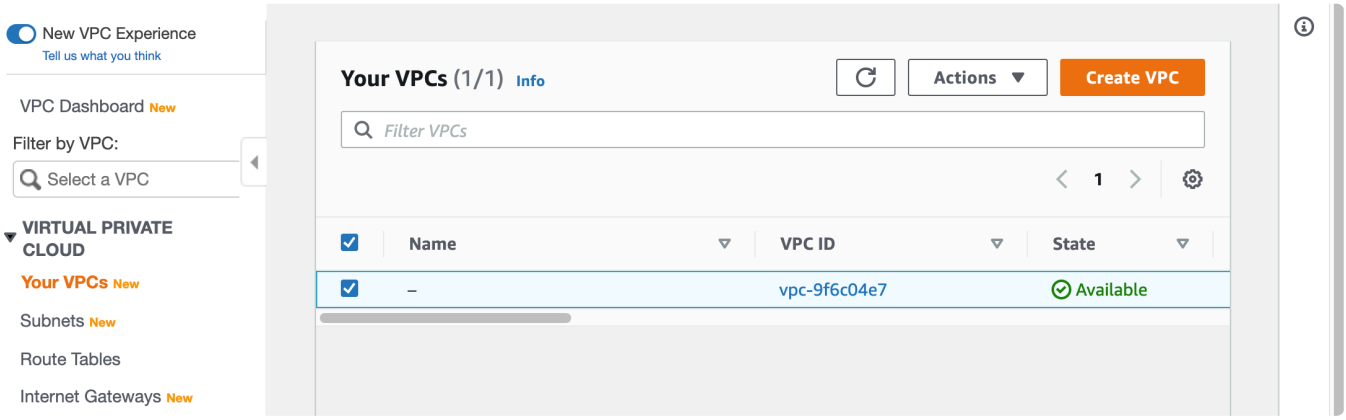
A VPC (Virtual Private Cloud) is an AWS resource that represents a local network. Different VPCs can be defined within the AWS cloud to logically separate different systems.

Refer to the AWS documentation for more information.

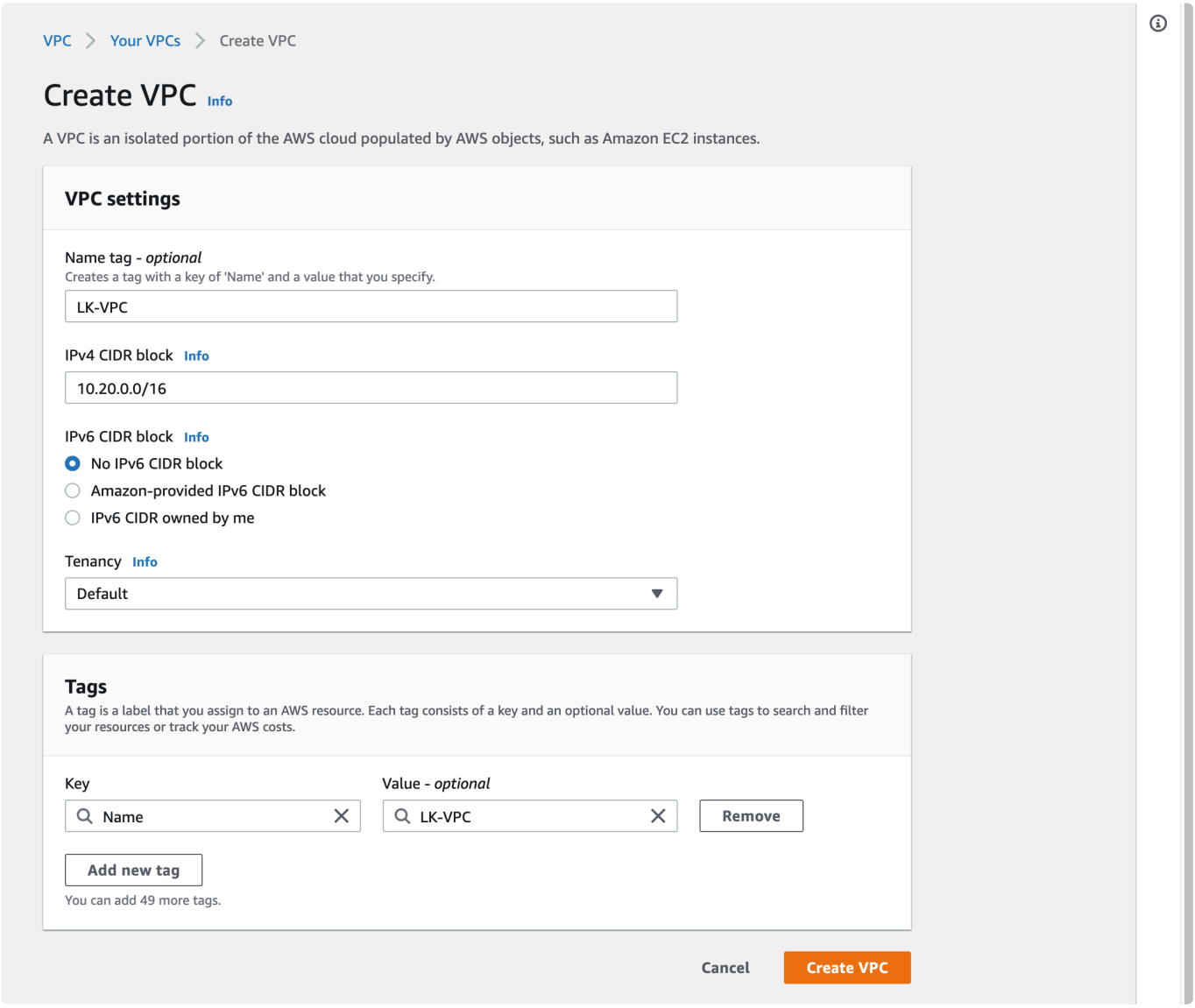
In this section we will create a VPC for testing LifeKeeper (named LK-VPC) as follows:

1. Select “Your VPCs” from the navigation pane located at the left side of the console.

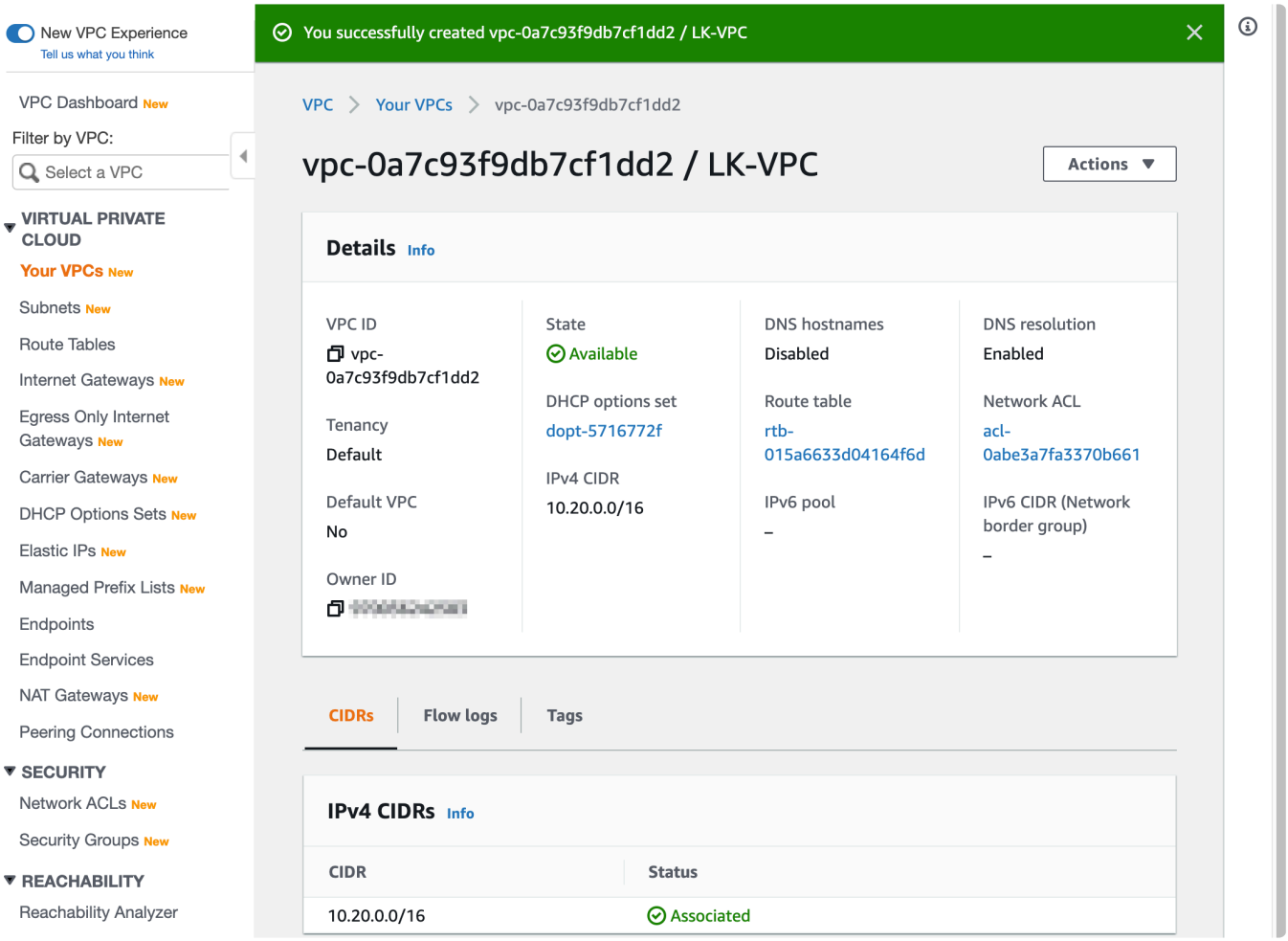
2. There is a VPC created by AWS (default VPC). However, we are going to create a custom VPC based on the specification defined in this evaluation guide. Select “Create VPC” in the top right corner.




3. In the “Create VPC” wizard, enter the name of the VPC (LK-VPC) and the IPv4 CIDR block `10.20.0.0/16`.




4. Now the VPC `LK-VPC` is created with the parameters specified.


New VPC Experience
Tell us what you think
VPC Dashboard New
Filter by VPC:
Select a VPC
VIRTUAL PRIVATE CLOUD
Your VPCs New
Subnets New
Route Tables
Internet Gateways New
Egress Only Internet Gateways New
Carrier Gateways New
DHCP Options Sets New
Elastic IPs New
Managed Prefix Lists New
Endpoints
Endpoint Services
NAT Gateways New
Peering Connections
SECURITY
Network ACLs New
Security Groups New
REACHABILITY
Reachability Analyzer
You successfully created vpc-0a7c93f9db7cf1dd2 / LK-VPC
VPC > Your VPCs > vpc-0a7c93f9db7cf1dd2
vpc-0a7c93f9db7cf1dd2 / LK-VPC
Actions
Details Info
VPC ID
vpc-0a7c93f9db7cf1dd2
Tenancy
Default
Default VPC
No
Owner ID
State
Available
DHCP options set
dopt-5716772f
IPv4 CIDR
10.20.0.0/16
DNS hostnames
Disabled
Route table
rtb-015a6633d04164f6d
IPv6 pool
–
DNS resolution
Enabled
Network ACL
acl-0abe3a7fa3370b661
IPv6 CIDR (Network border group)
–
CIDRs
Flow logs
Tags
IPv4 CIDRs Info
CIDR
Status
10.20.0.0/16
Associated

# 11.2.3.3.4. Creating a Subnet

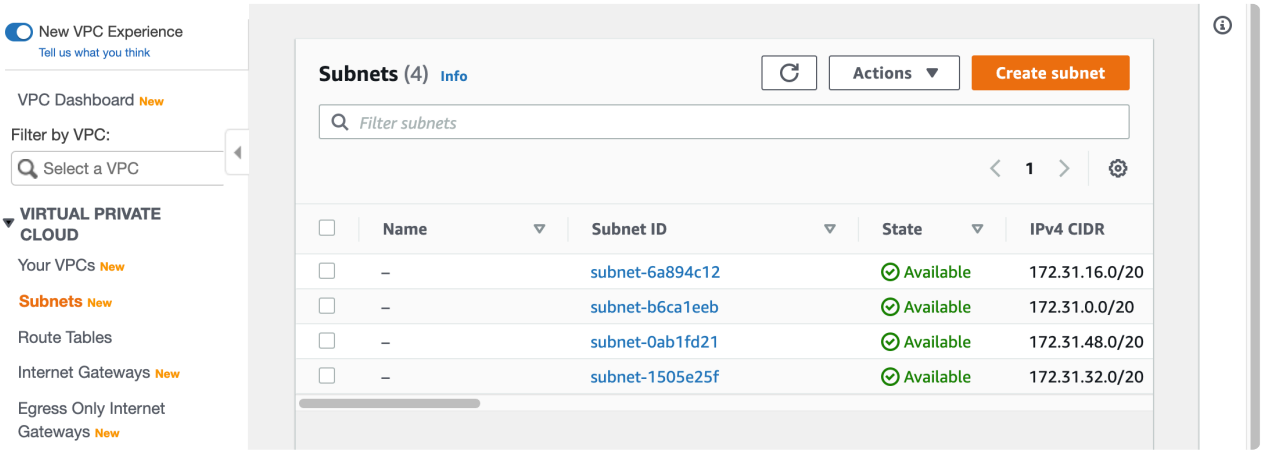
The basic concept of a subnet in AWS is the same as in on-premise environments. Namely it is a network within a network. Separate subnets can be defined to create a logical or physical boundary between components. Unique AWS-specific concepts are as follows:

- If a system is deployed across different Availability Zones (different datacenters), a separate subnet must be defined for each Availability Zone.
- If instances within a subnet need direct access from/to the Internet, it should be configured as a “public” subnet. Note that if instances need to be separated from/to the Internet, it should be configured as a “private” subnet. Consequently, there may need to be more than 2 subnets defined per Availability Zone.

  Refer to the AWS documentation for more information.

In this tutorial, we will use three (3) Availability Zones and use only the “public” pattern to simplify the process. A “public” subnet is created as follows:

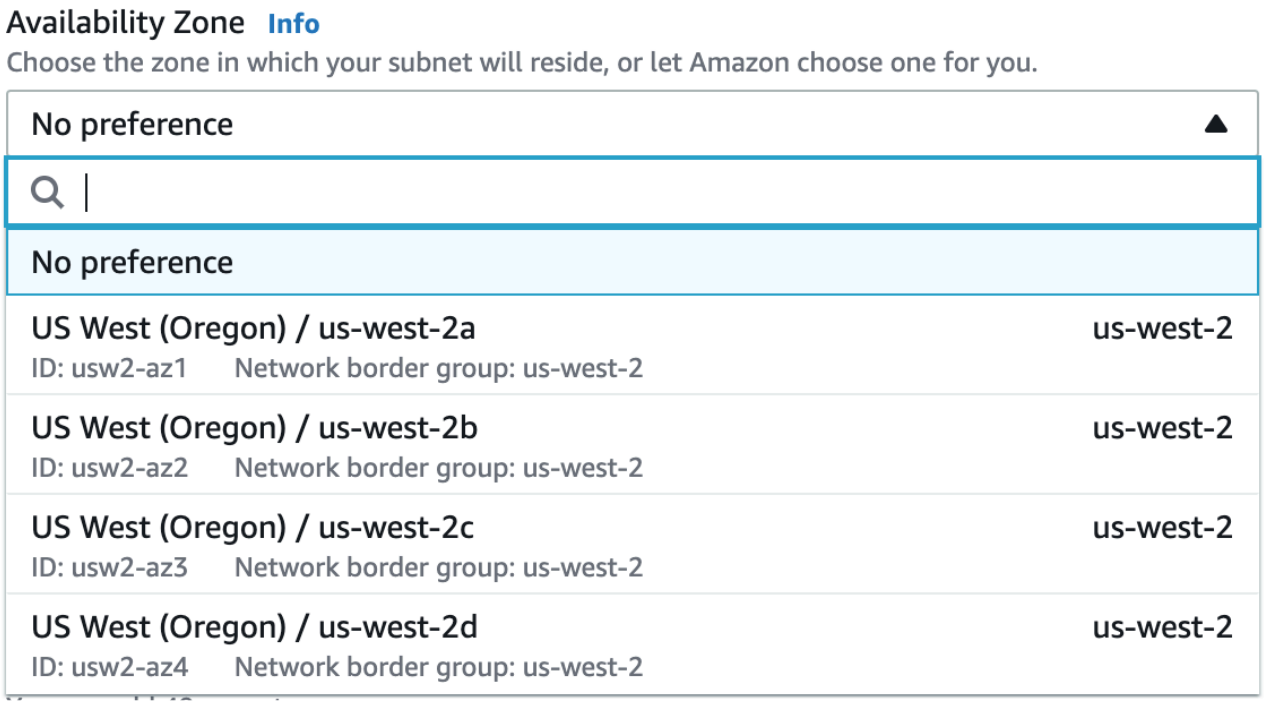
1. Go to the Subnet page by selecting “Subnets” from the navigation pane at the left side.
2. As per the VPC, there are pre-defined subnets for the default VPC. Select “Create Subnet” from the top right corner.



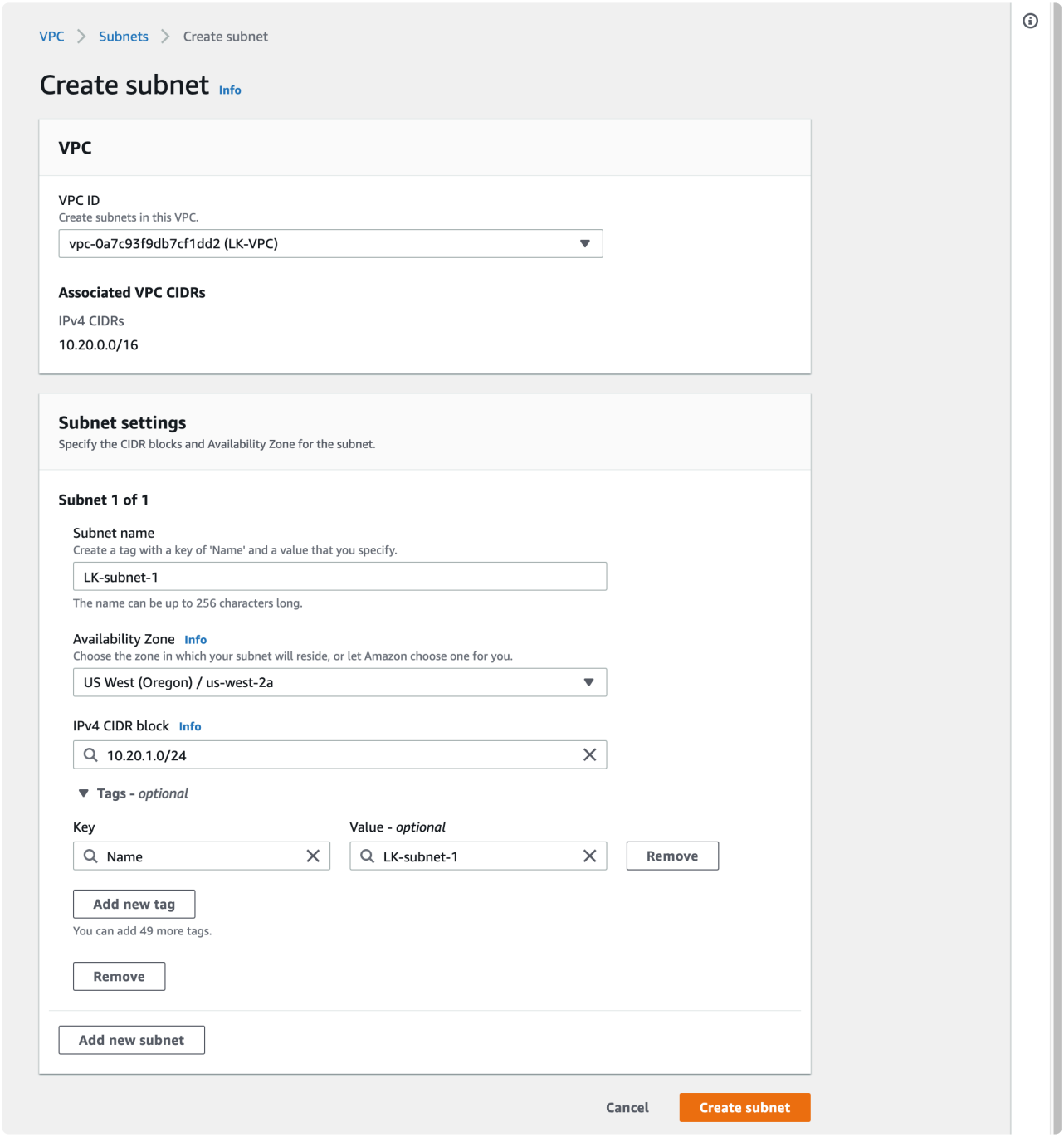

3. Specify following parameters as shown in the screenshot:

- VPC ID: `LK-VPC`
- Subnet Name: `LK-subnet-1`
- Availability Zone: Select the first option.
- IPv4 CIDR block: `10.20.1.0/24`
- Tag: Name = `LK-subnet-1`

While working in this wizard, a list of Availability Zones can be seen by selecting the dropdown box for the Availability Zones:




Select the first Availability Zone (in this case `us-west-2a`)

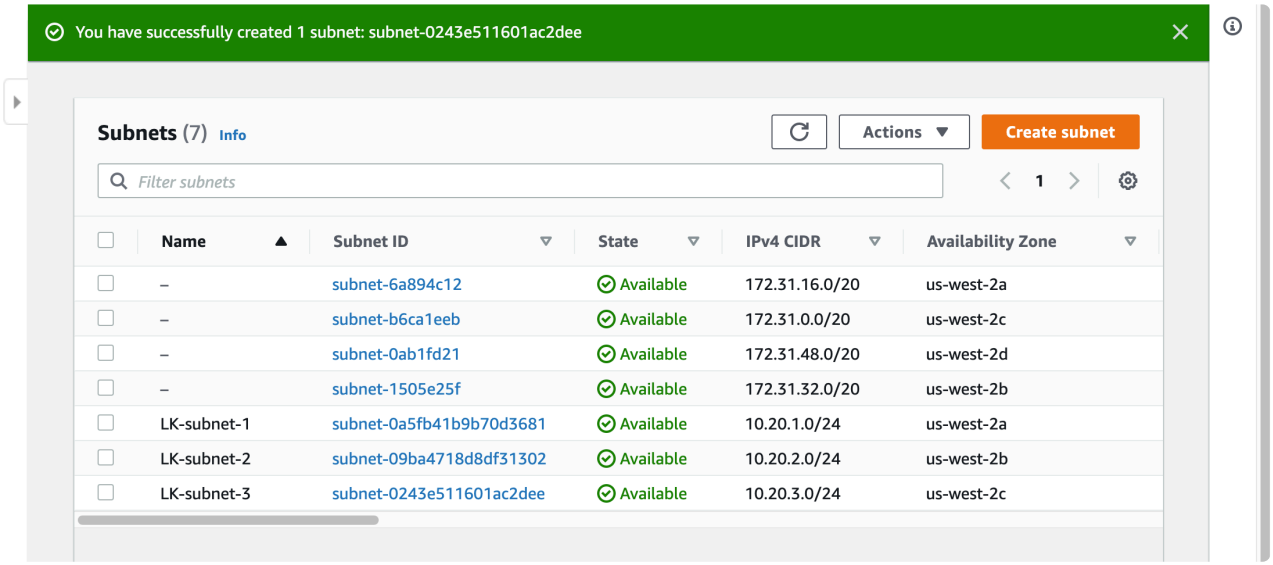



4. Now the first subnet on Availability Zone `us-west-2a` is created. Return to the “Subnet Page”.

5. Create the second subnet `LK-subnet-2` with CIDR block `10.20.2.0/24` in another Availability Zone `us-west-2b`.

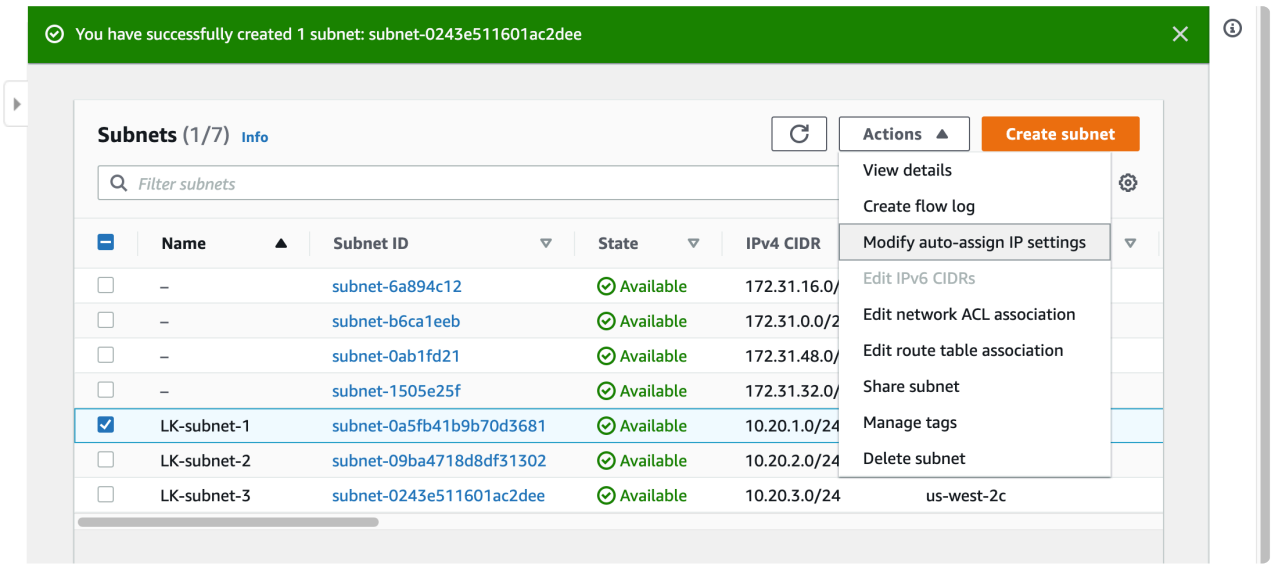
6. Create the third subnet LK-subnet-3 with a CIDR block 10.20.3.0/24 on the Availability Zone `us-west-2c`.

7. Once the three subnets have been created, the list of subnets should be as shown in the screenshot below. The first subnets in the picture are the “default“ subnets created for “default” VPC followed by the three subnets we have just created.

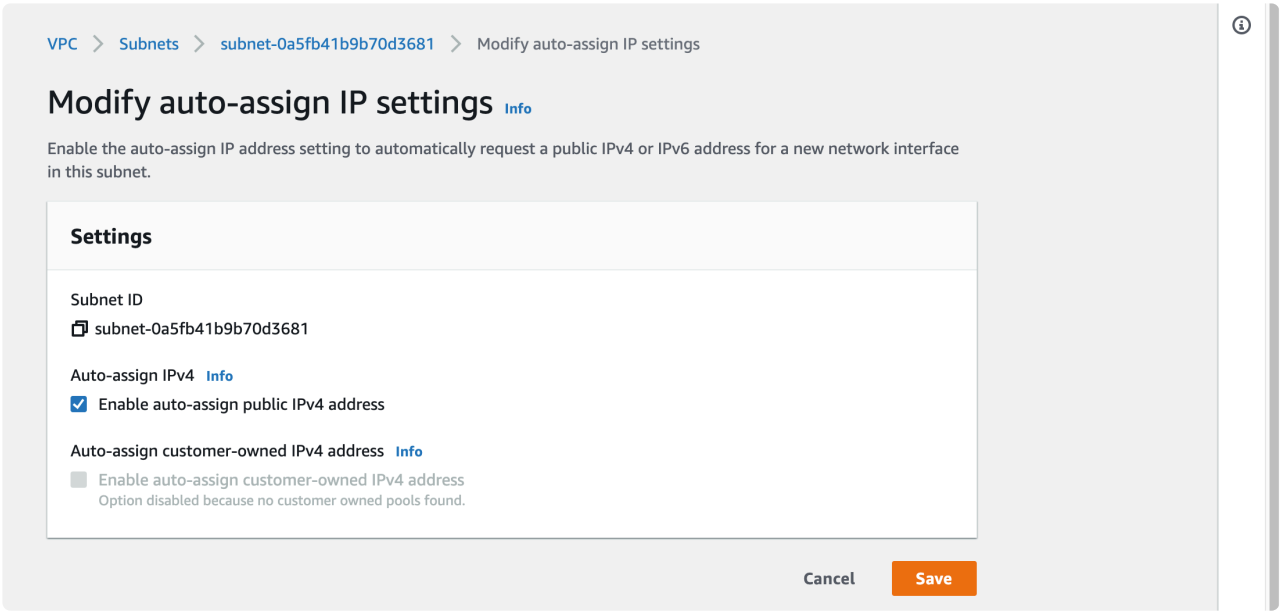



8. In this tutorial we will create a public subnet. This signifies that we want to assign a public IP address for instances we create under this subnet. To do so, select the subnet we have just created and then select “Modify auto-assign IP settings” from the Actions menu.

**Note:** This change must be made for all 3 subnets, one by one.




9. Select “Enable auto-assign public IPv4 address”, and save the change.




10. Now the subnets are ready to use.

# 11.2.3.3.5. Creating an Internet Gateway and Assigning it to the VPC

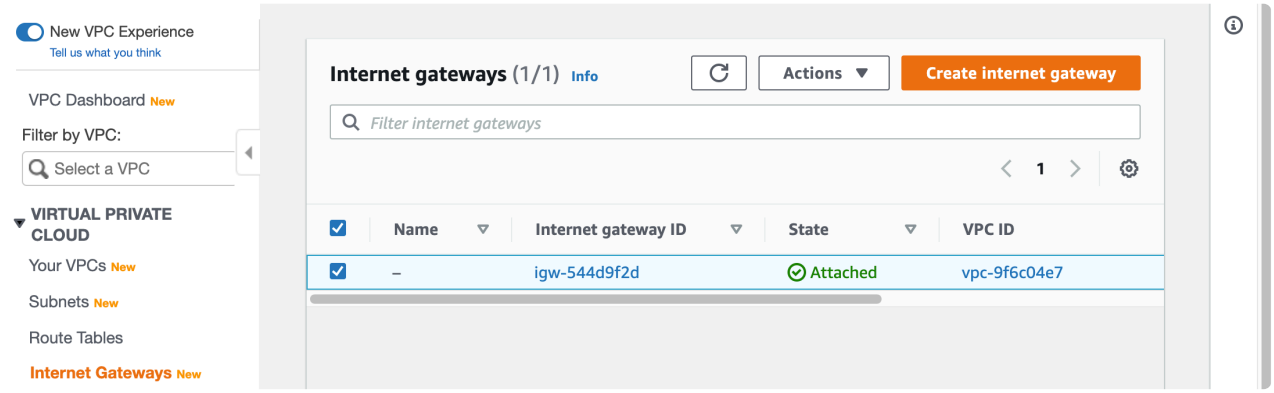
The next step is to create an internet gateway. An internet gateway is a VPC component that allows communication between the VPC and the internet.

Refer to the AWS documentation for more information.

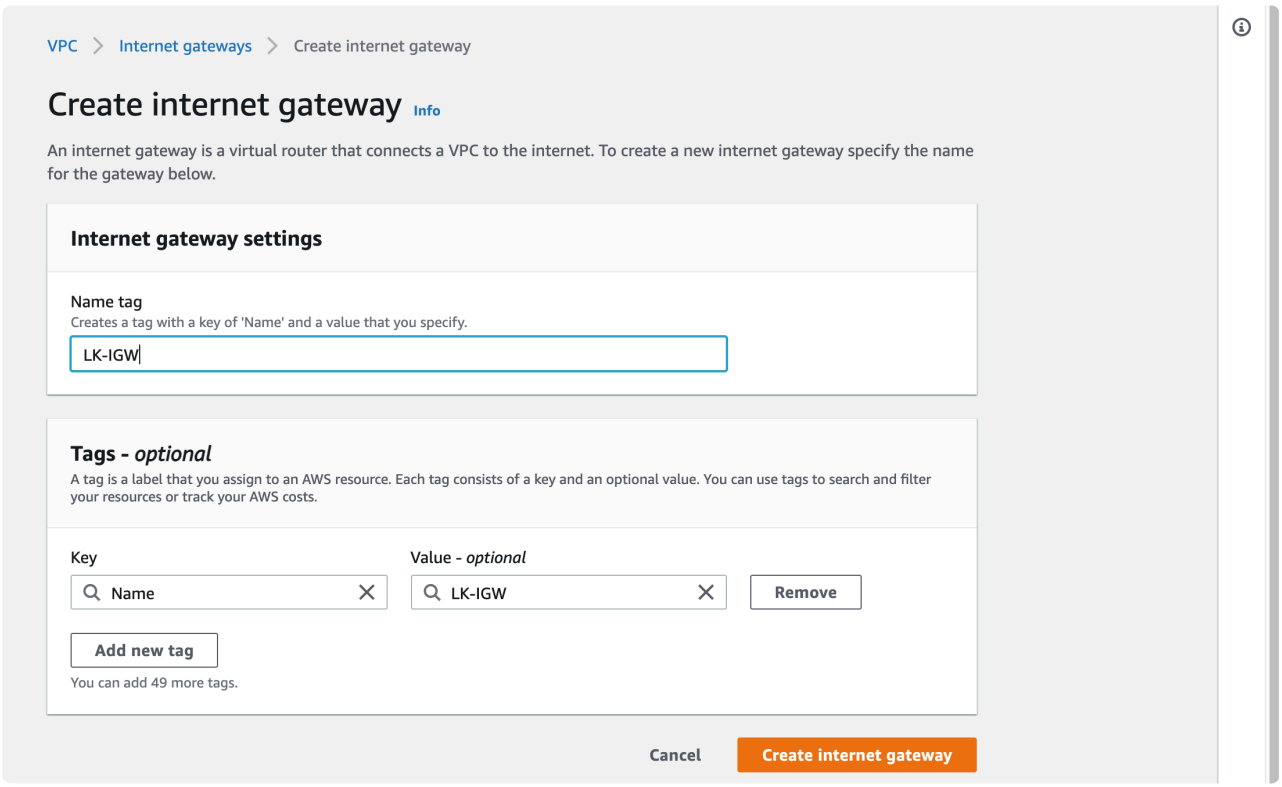
Let’s create it and associate it with the VPC:

1. Select “Internet Gateways” from the navigation pane located at the left side.

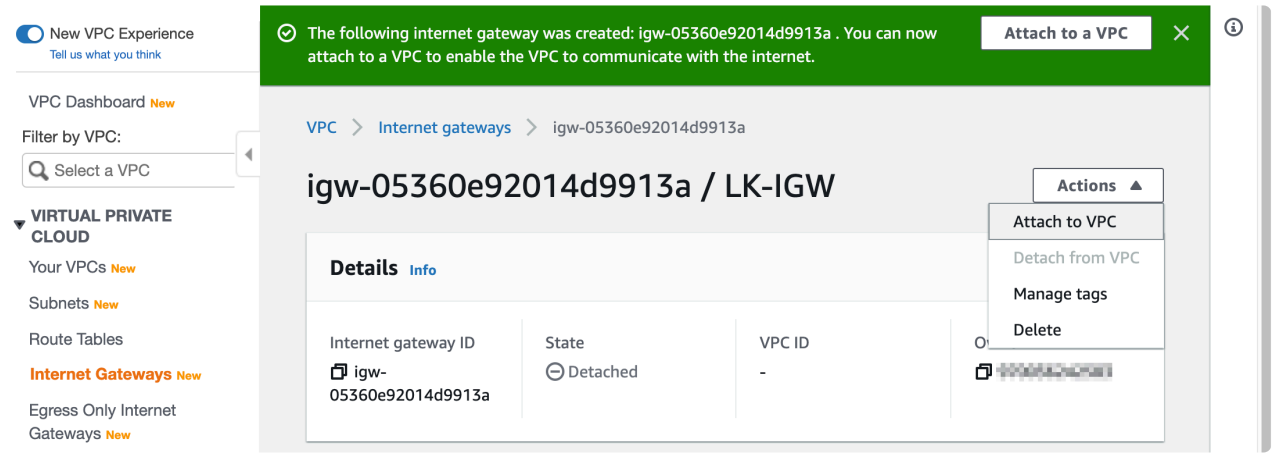
2. You may see an existing Internet Gateway as a part of default VPC. Select “Create Internet Gateway” from the top right corner.



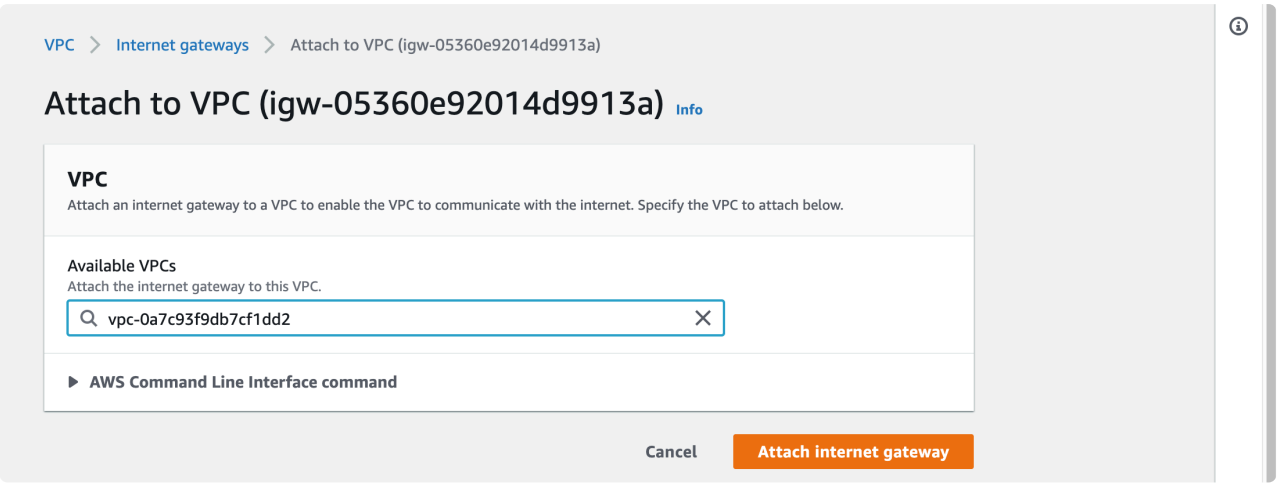

3. Specify the name of the Internet Gateway as follows.



4. Once the Internet Gateway is created, select “Attach to VPC”.




5. Select `LK-VPC` as the VPC to associate with.




6. The Internet Gateway is now configured.

# 11.2.3.3.6. Creating the Route Table

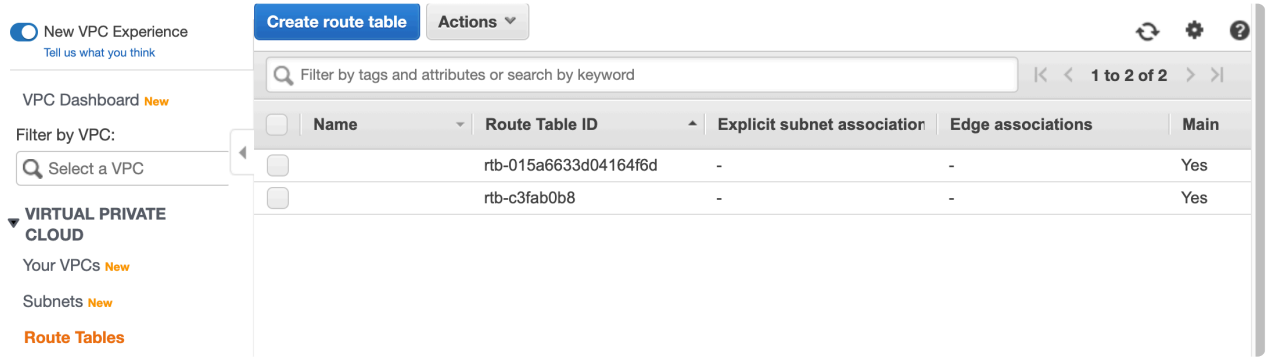
The route table defines how the traffic from instances within a subnet should be transferred.

Refer to the AWS documentation for more information.

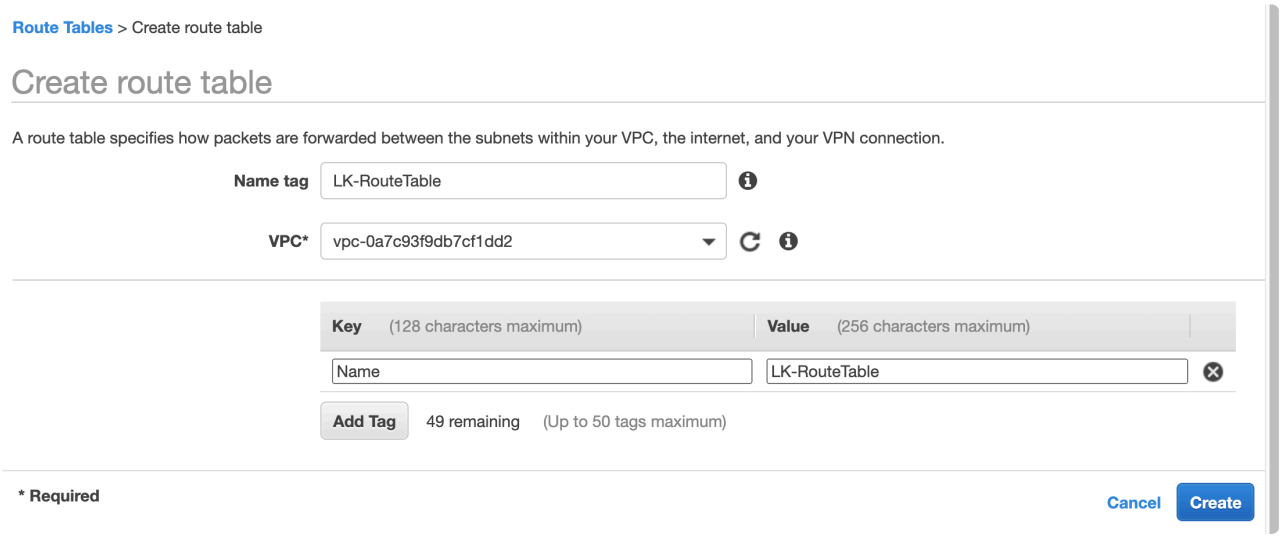
In this section we will create a route table and add a route to the internet via an Internet Gateway.

1. Select “Route Tables” from the left navigation pane.

2. Select “Create route table” at the top left corner.

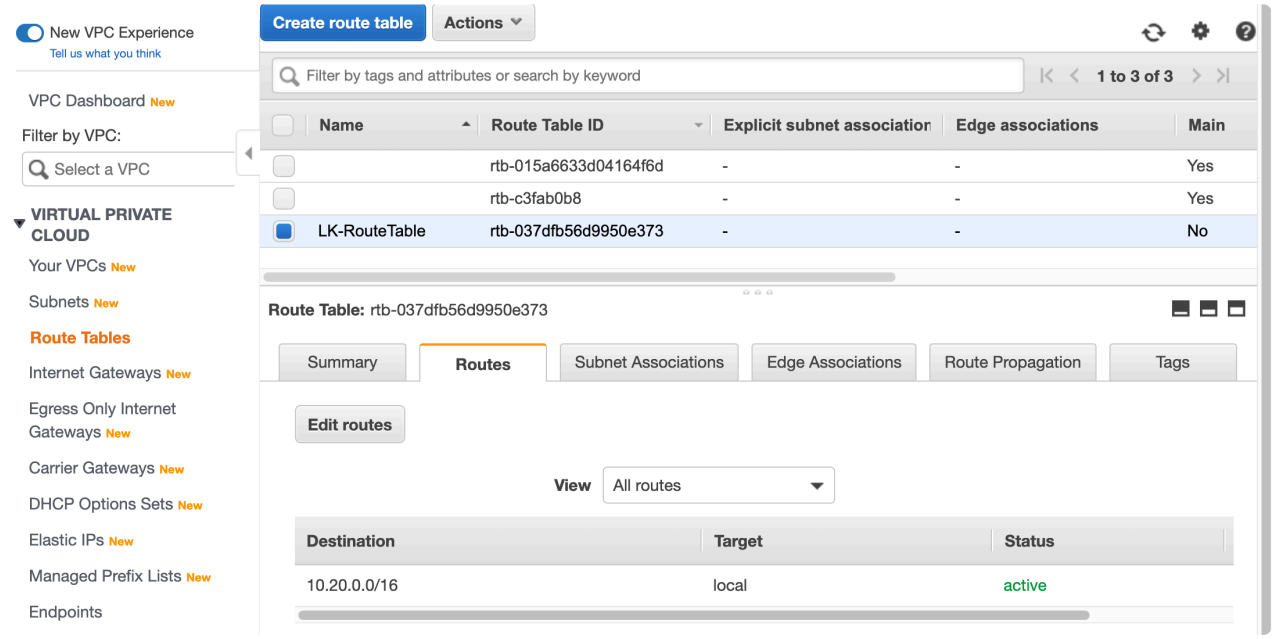



3. Set the name of this route table as `LK-RouteTable`. Also, define the “Name” Tag. Please ensure that `LK-VPC` is selected as the associated VPC.

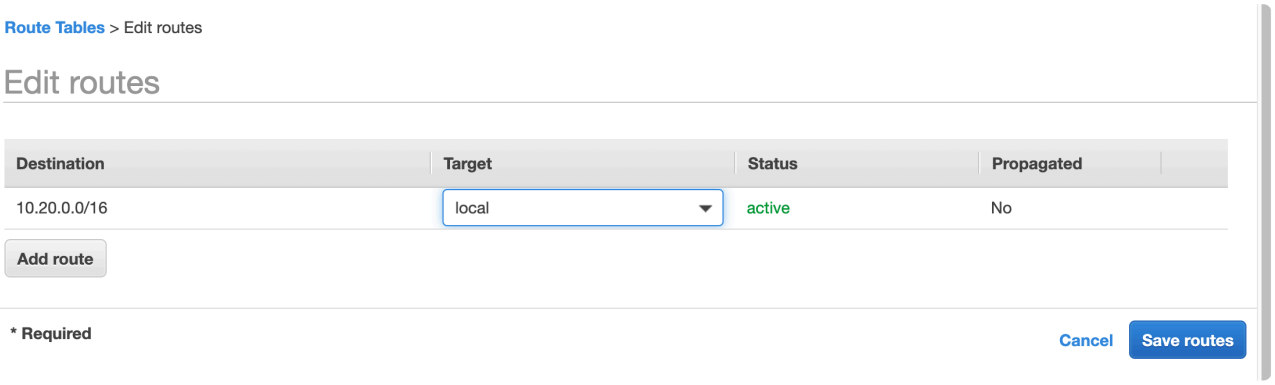



   Once these fields are defined, select “Create”.

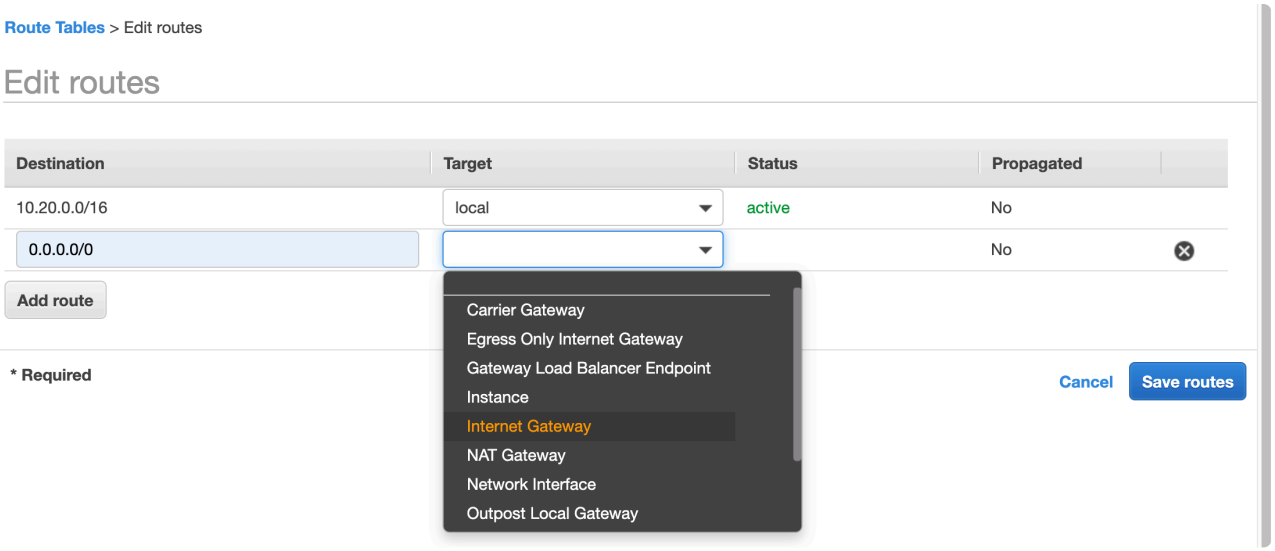
4. Go to the “Routes” page and click “Edit Routes”.



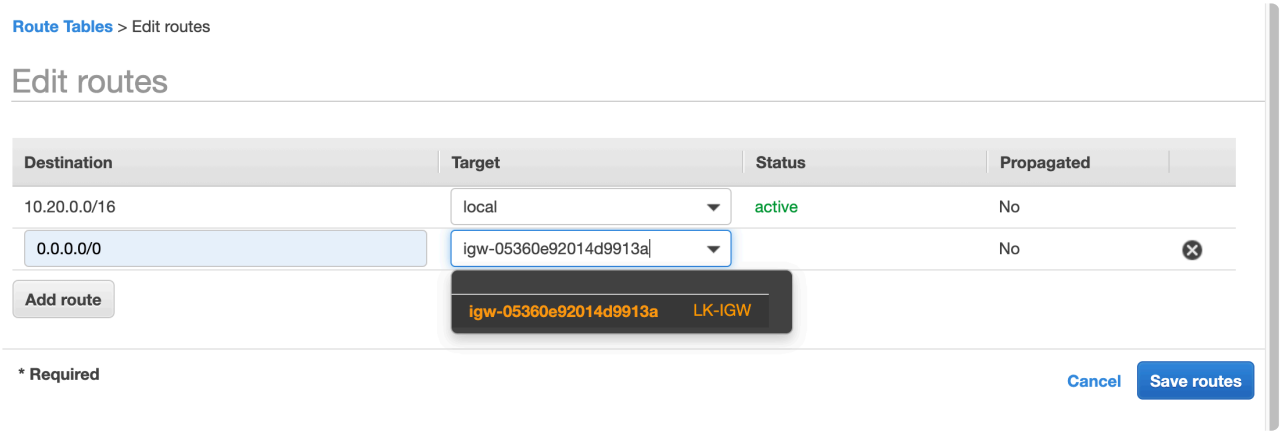

5. Local traffic is already defined. Now click “Add route”.




6. Enter `0.0.0.0/0` (meaning any IPv4 address, i.e., the internet) as the destination, and select “Internet Gateway” as Target.

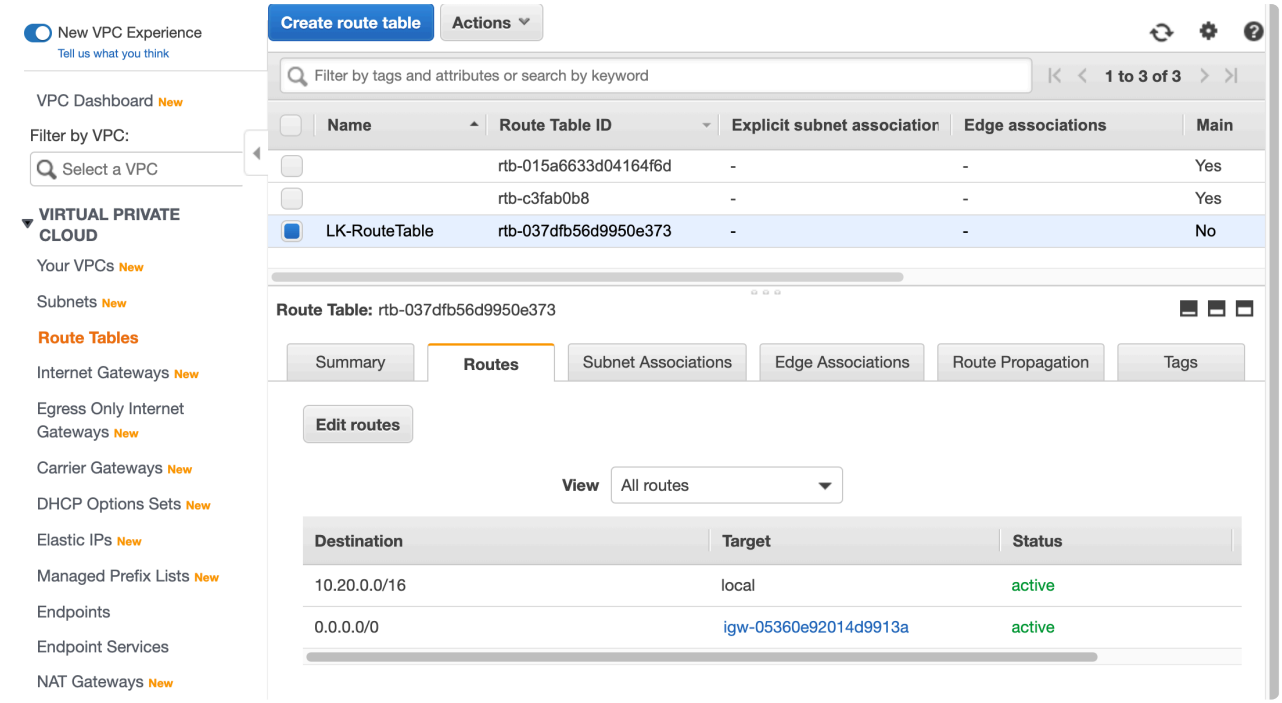



7. Select “LK-IGW” as the Internet Gateway.

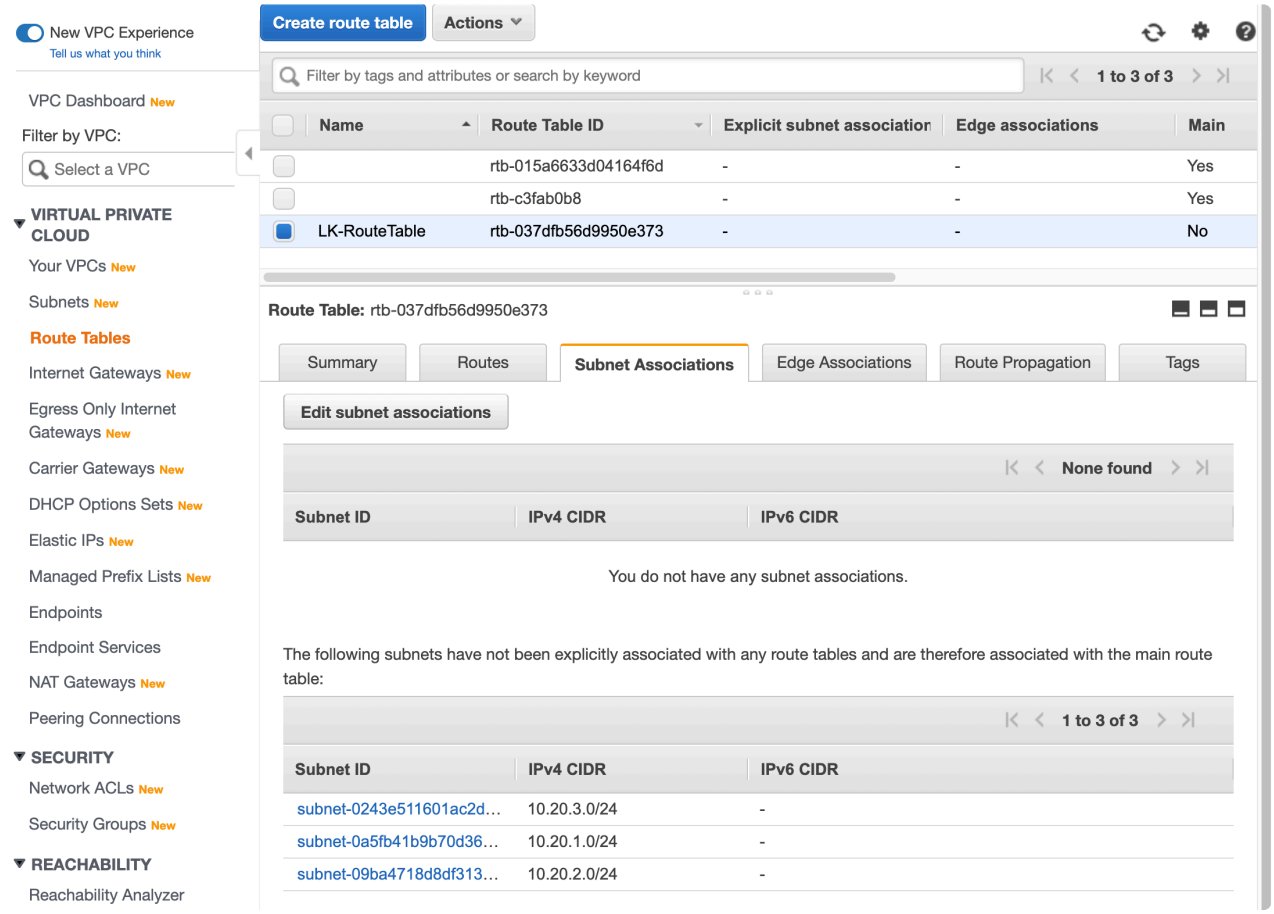



Once a route is defined for 0.0.0.0/0, click “Save Routes” to close the wizard.

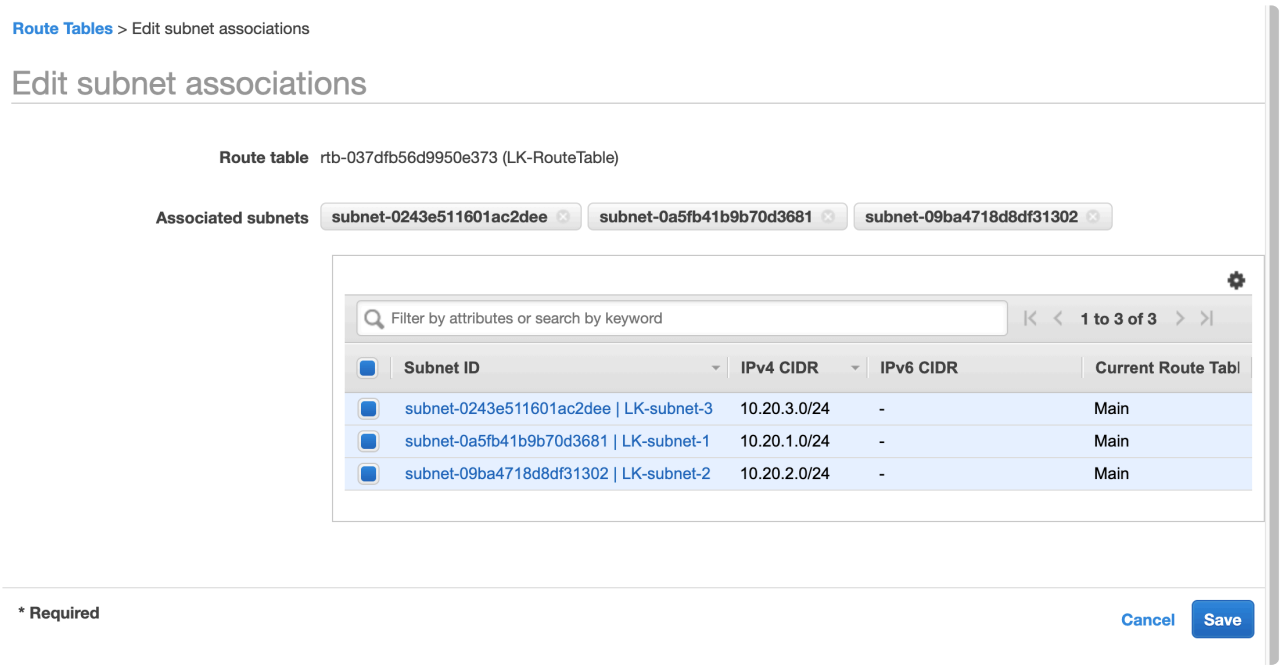
8. Now the routes are defined.




9. The next step is to associate the route table with newly created subnets (LK-subnet-1, LK-subnet-2 and LK-subnet-3). Select the “Subnet Associations” tab.




10. Click “Edit subnet associations” and select “LK-subnet-1”, “LK-subnet-2”, and “LK-subnet-3”.




Now the new Route Table (“LK-RouteTable”) is defined and associated with the LK-subnet-1/2/3 subnets.

# 11.2.3.3.7. Creating a Security Group

A Security Group works as a firewall, and “allow” rules can be defined from each source. A source can be an internet address or security group, and a security group can be assigned to an Elastic Network Interface. As a network interface is attached to an Instance, this works as a firewall rule for each instance.

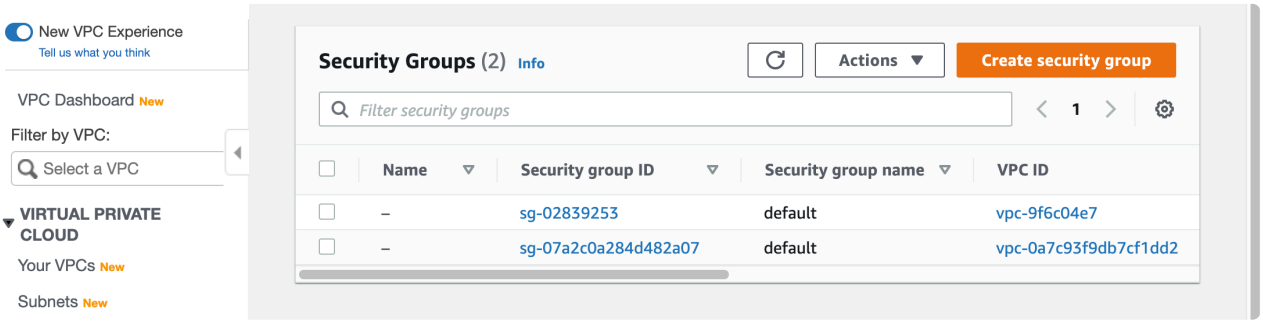
Refer to the AWS documentation for more information.

In this section, we will define two “allow” rules as follows:

- Allow access from the remote work location.
- Allow access between instances (node-a and node-b). This is done by allowing the same security group as the source and then assigning the same security group to both instances.

> **Note:** Once a production environment is defined, the instances may belong to different security groups. In this case, a corresponding “allow” rule between security groups should be defined.

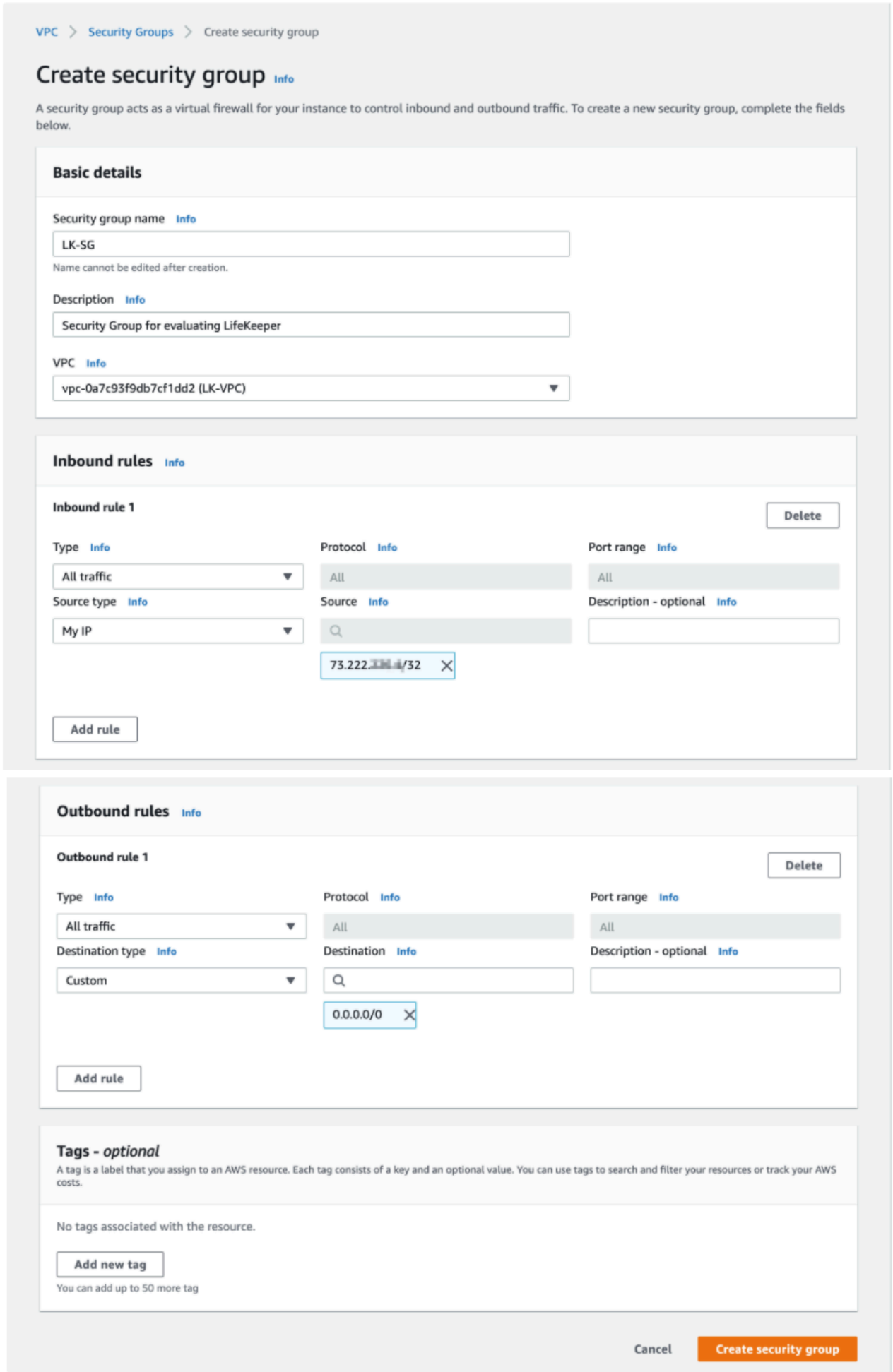
1. Select “Security Groups” from the left navigation pane. As previously mentioned, default security groups will already exist. Click “Create security group”.



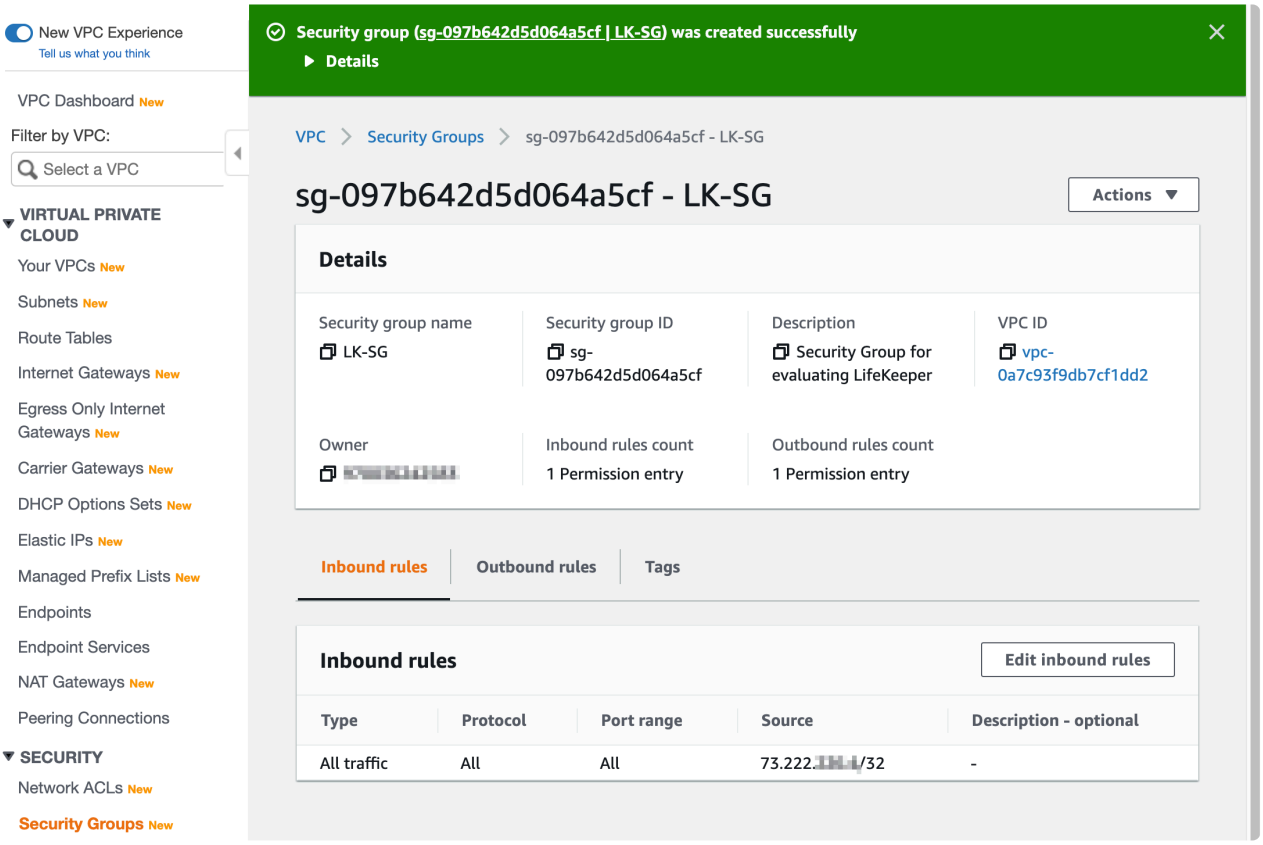

2. Enter the values for the new security group as shown in the screenshot using the following parameters:

- Name: `LK-SG`
- VPC: `LK-VPC`
- Create the first Inbound Rule. Select “Add Rule” and select “All Traffic” as the type (for now). Then select “My IP” as Source type. This automatically selects the user’s WAN address as source. If required, update the source address range based on the in use WAN address (it may be appropriate to change this from a specific IP address to a range of IP addresses).

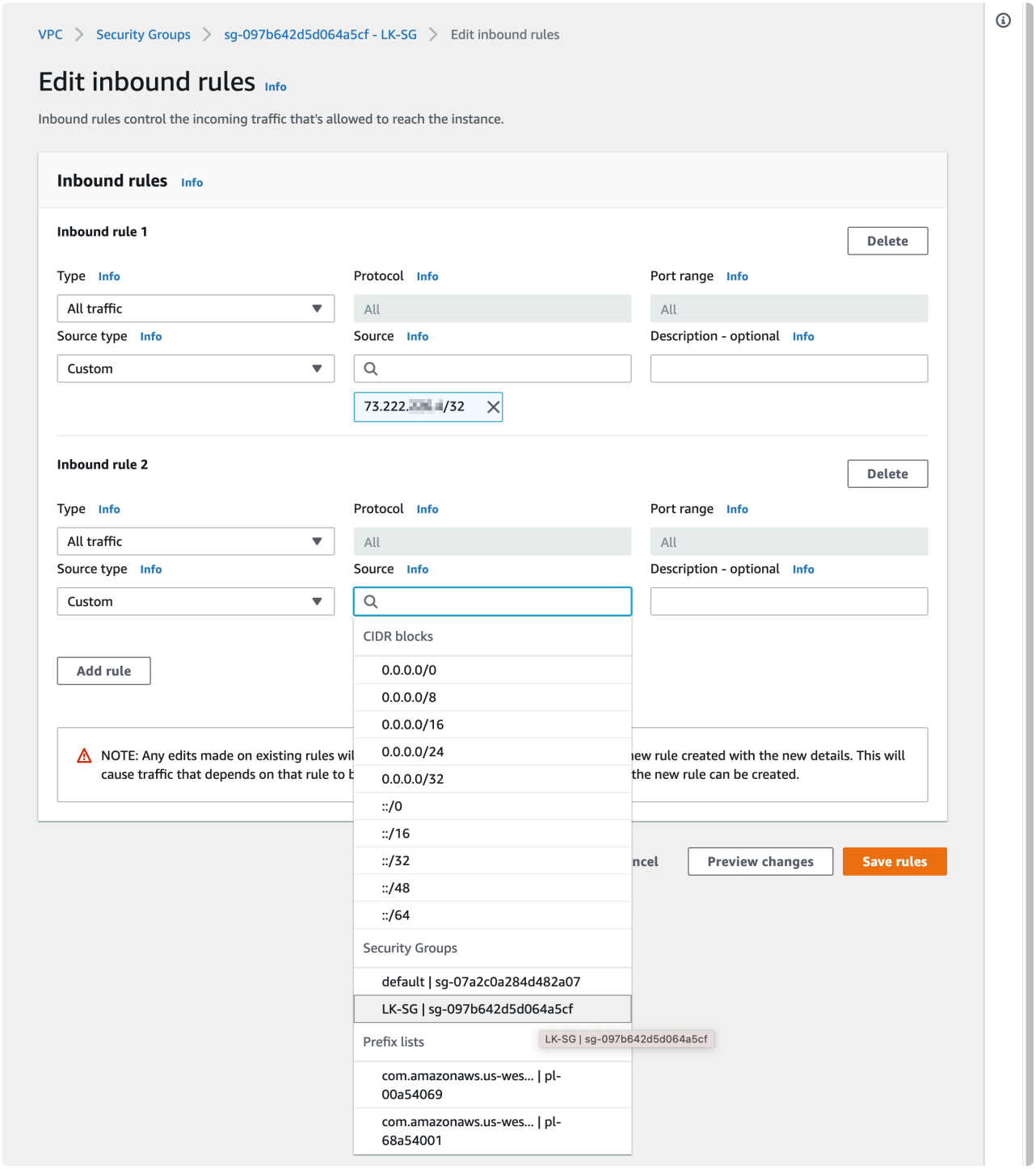
  Once the values are confirmed, click “Create security group” at the bottom right.


VPC > Security Groups > Create security group
Create security group Info
A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.
Basic details
Security group name Info
LK-SG
Name cannot be edited after creation.
Description Info
Security Group for evaluating LifeKeeper
VPC Info
vpc-0a7c93f9db7cf1dd2 (LK-VPC)
Inbound rules Info
Inbound rule 1
Delete
Type Info
All traffic
Protocol Info
All
Port range Info
All
Source type Info
My IP
Source Info
73.222.[redacted]/32
Description - optional Info
Add rule
Outbound rules Info
Outbound rule 1
Delete
Type Info
All traffic
Protocol Info
All
Port range Info
All
Destination type Info
Custom
Destination Info
0.0.0.0/0
Description - optional Info
Add rule
Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
No tags associated with the resource.
Add new tag
You can add up to 50 more tag
Cancel
Create security group

3. Once new security group has been created, click “Edit inbound rules” to allow traffic within the same Security Group.

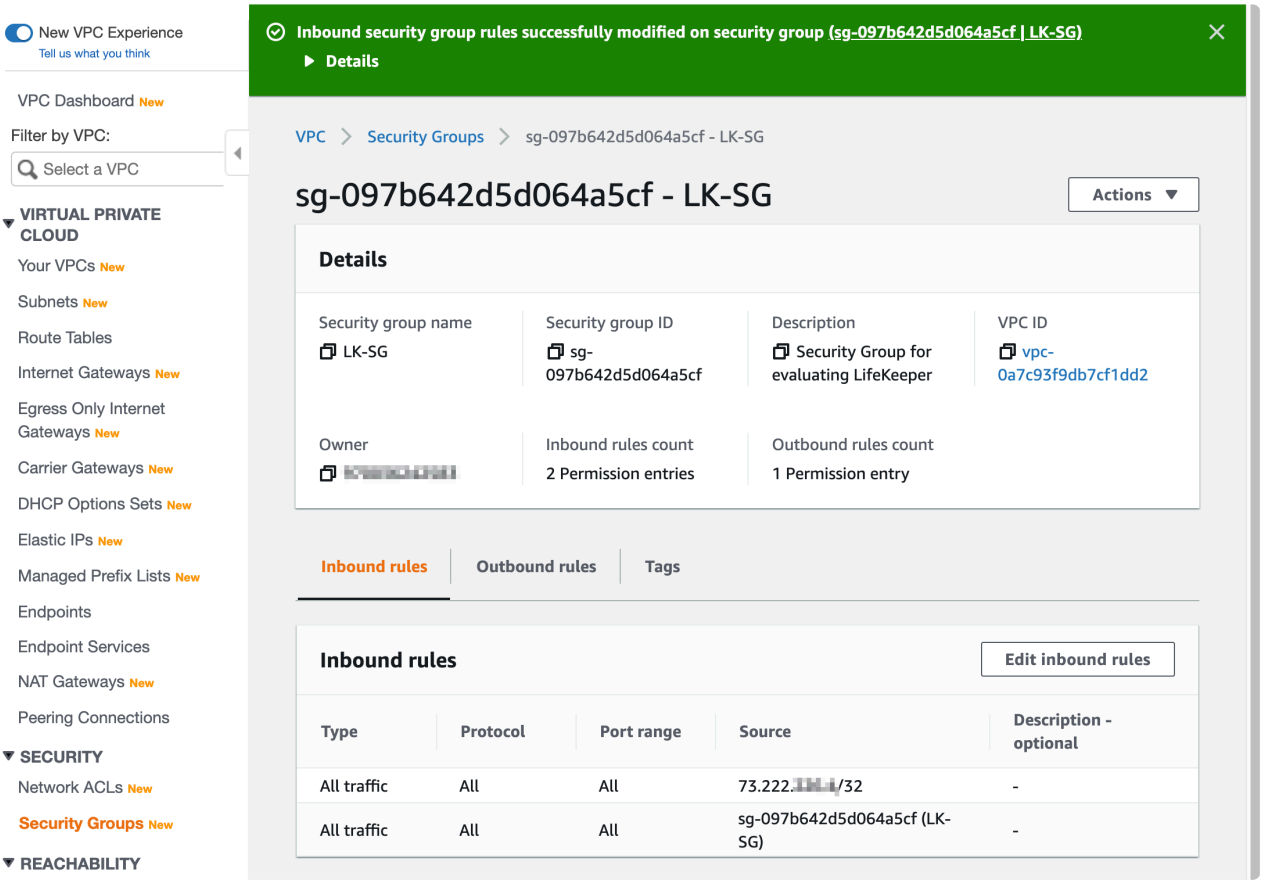



4. Once the “Edit inbound rules” page appears, select “Add rule”, then select “LK-SG” (which we just created). With this change, any instances that are associated with “LK-SG” can communicate with each other.




Once the change is confirmed, click “Save rules”. The Security Group is now created.

5. Confirm the parameters for the newly created Security Group.

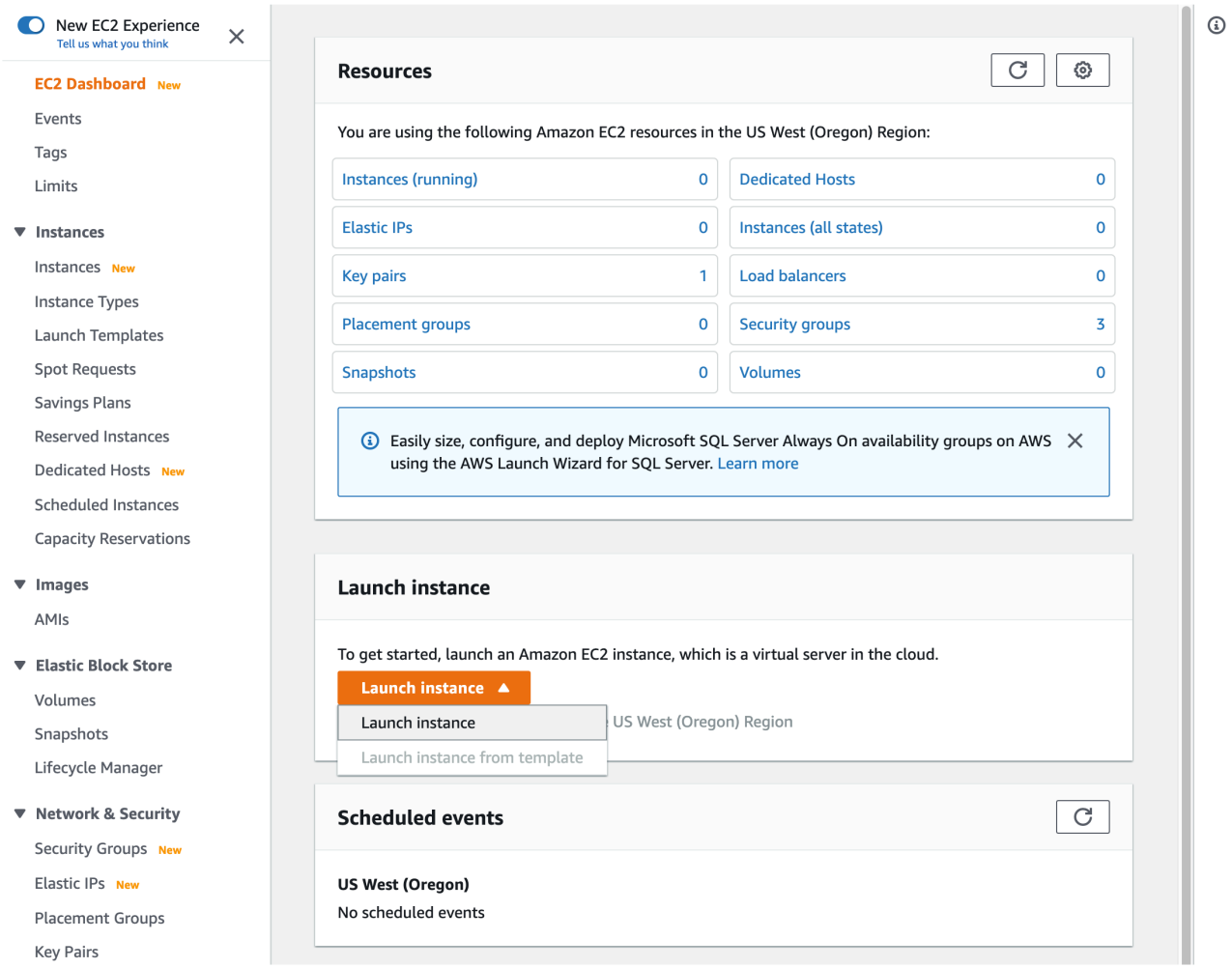

New VPC Experience
Tell us what you think
VPC Dashboard New
Filter by VPC:
Select a VPC
VIRTUAL PRIVATE CLOUD
Your VPCs New
Subnets New
Route Tables
Internet Gateways New
Egress Only Internet Gateways New
Carrier Gateways New
DHCP Options Sets New
Elastic IPs New
Managed Prefix Lists New
Endpoints
Endpoint Services
NAT Gateways New
Peering Connections
SECURITY
Network ACLs New
Security Groups New
REACHABILITY
Inbound security group rules successfully modified on security group (sg-097b642d5d064a5cf | LK-SG)
Details
VPC > Security Groups > sg-097b642d5d064a5cf - LK-SG
sg-097b642d5d064a5cf - LK-SG
Actions
Details
Security group name
LK-SG
Security group ID
sg-097b642d5d064a5cf
Description
Security Group for evaluating LifeKeeper
VPC ID
vpc-0a7c93f9db7cf1dd2
Owner
Inbound rules count
2 Permission entries
Outbound rules count
1 Permission entry
Inbound rules
Outbound rules
Tags
Inbound rules
Edit inbound rules
Type
Protocol
Port range
Source
Description - optional
All traffic
All
All
73.222.
/32
-
All traffic
All
All
sg-097b642d5d064a5cf (LK-SG)
-

# 11.2.3.3.8. Creating the First EC2 Instance

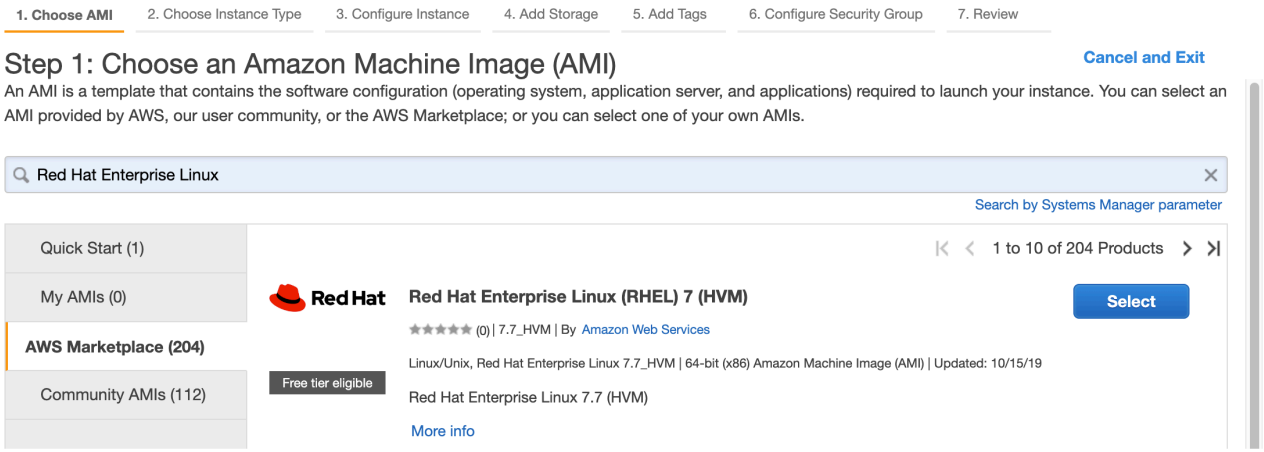
In previous sections we covered configuration of the network. Now we are going to create the first instance. As discussed in Computing Resources Used in this Tutorial, we need two disks. This section also describes how to create the second disk.

1. Go to “EC2” service. It may be necessary to enter “EC2” in the top search box of the AWS Console to select EC2.

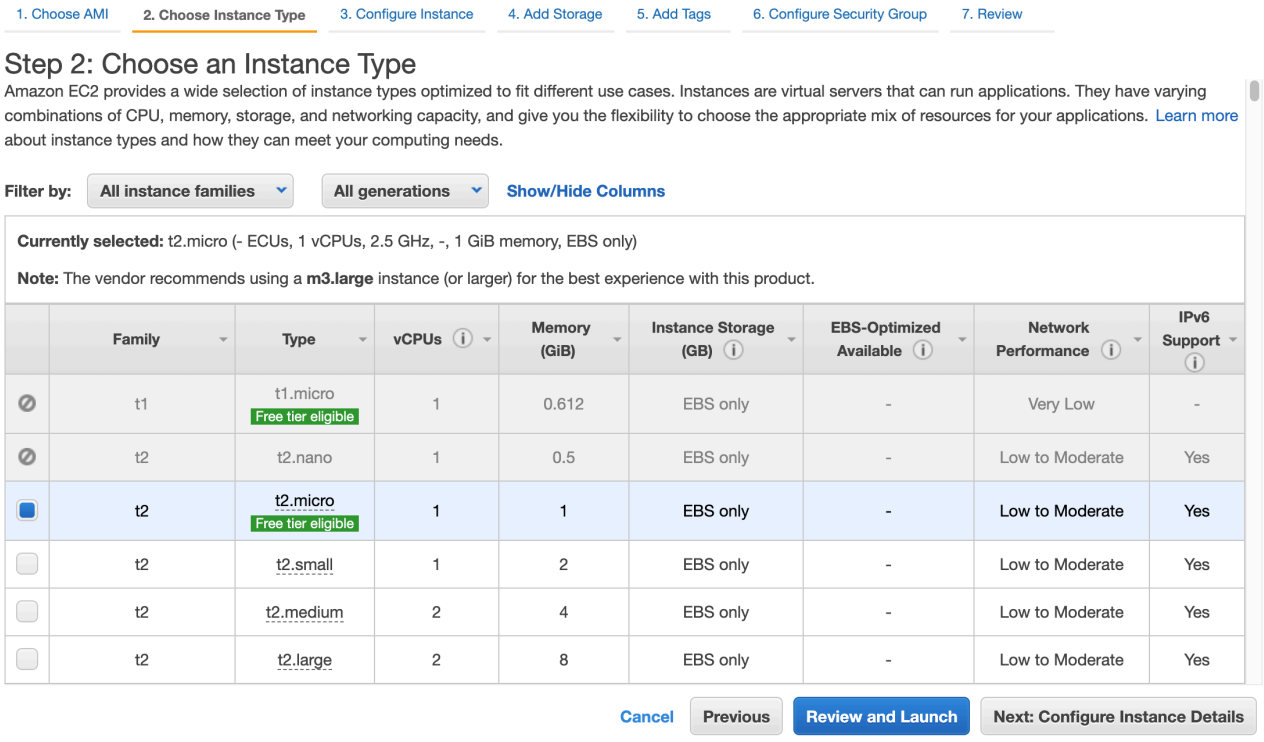
2. Click “Launch Instance”.




3. Click “Operating System”. Enter the name of the operating system and select the specific Machine Image. Before selecting the operating system and its version, please review the system requirements for both SIOS Protection Suite and the application you are going to protect.




4. Click “Instance Size”. This tutorial uses `t2.micro` since it is defined as the minimum system requirement for evaluation of SIOS Protection Suite and it may qualify for Free Tier usage. Again, it may be necessary to select a larger instance size depending on the application to be protected.

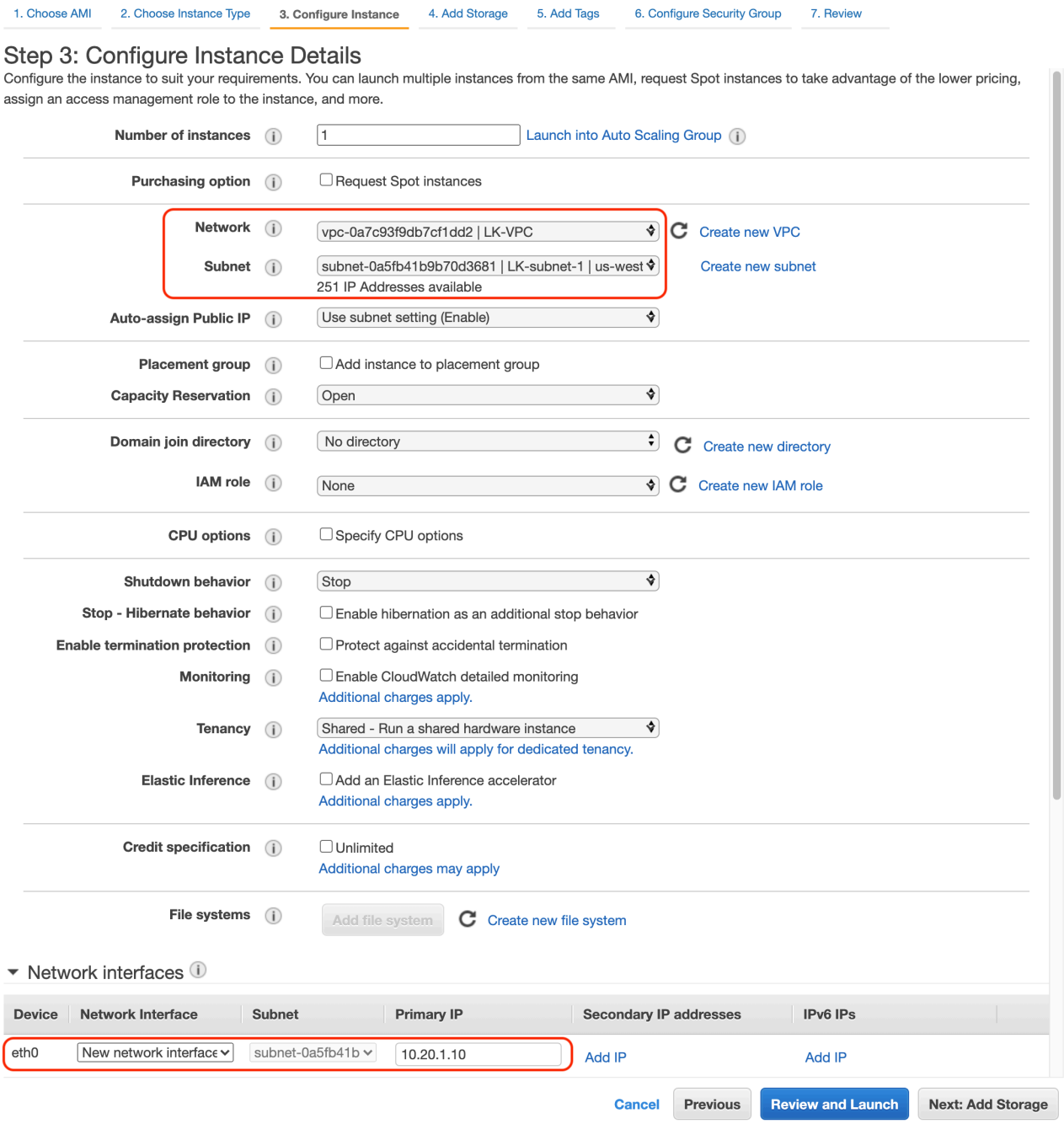



Once the instance size has been chosen, click “Next: Configure Instance Details”.

5. In the “Configure Instance Details” wizard, ensure that the following parameters are used:

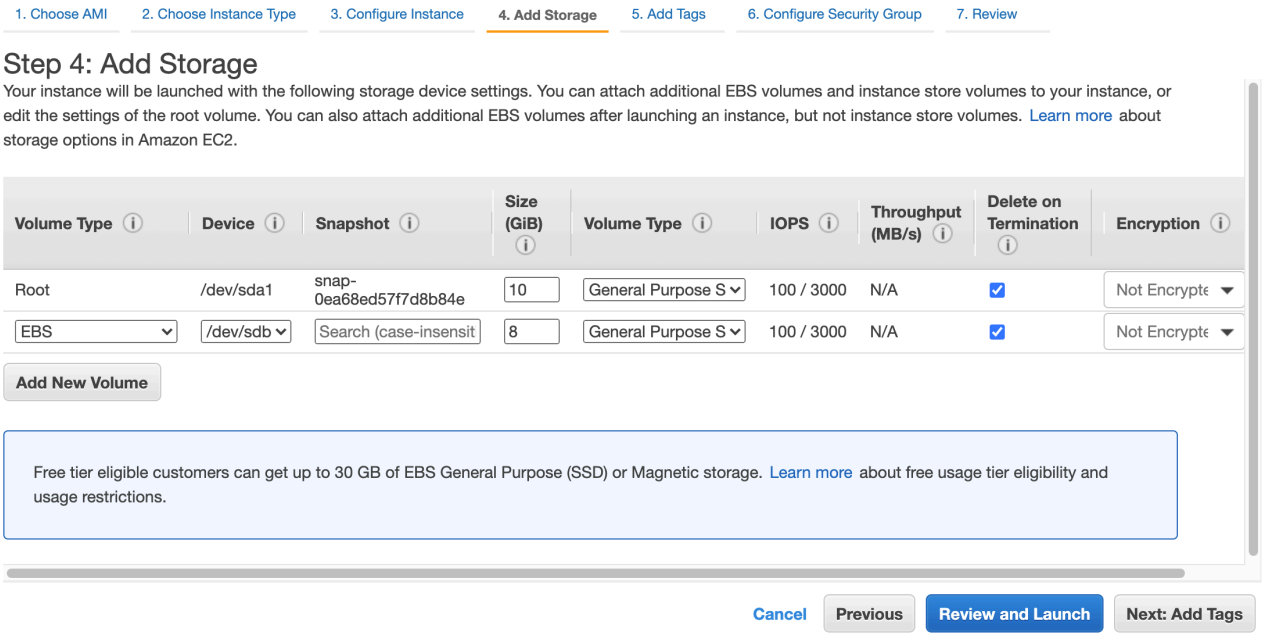
- VPC: `LK-VPC`
- Subnet: `LK-subnet-1`
- Network Interface: Please enter `10.20.1.10` for the first instance (node-a).

**Note:** When creating node-b and node-c, please refer to parameters related to node-b & node-c.



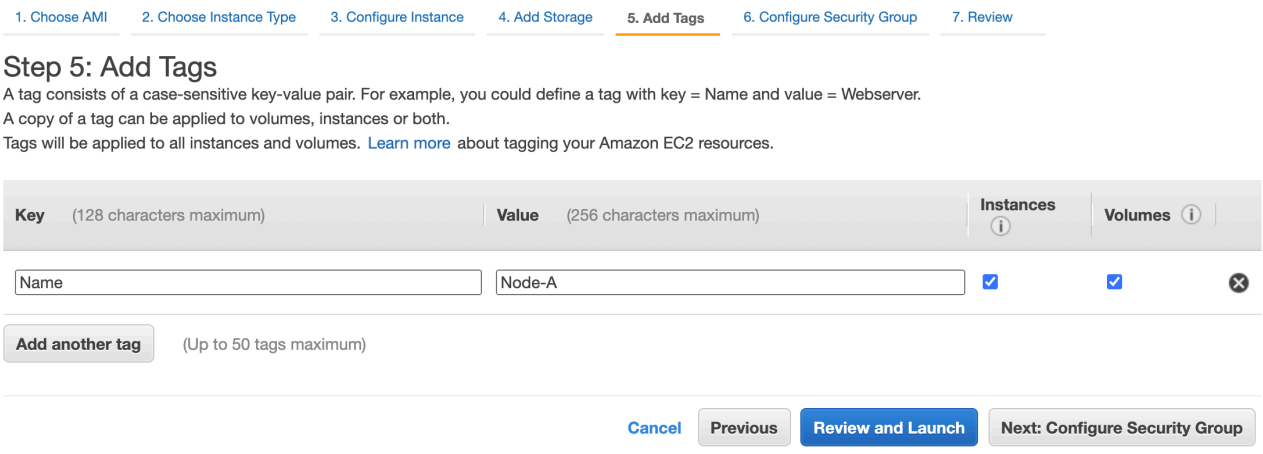

Once these values are confirmed, click “Next: Add Storage”.

6. The wizard has already created the first storage device (volume). Click “Add New Volume” and create a new disk. Per the minimum requirements previously discussed in this guide, an 8GiB disk should be sufficient. However, this may differ depending on the volume of the data to be protected (replicated).



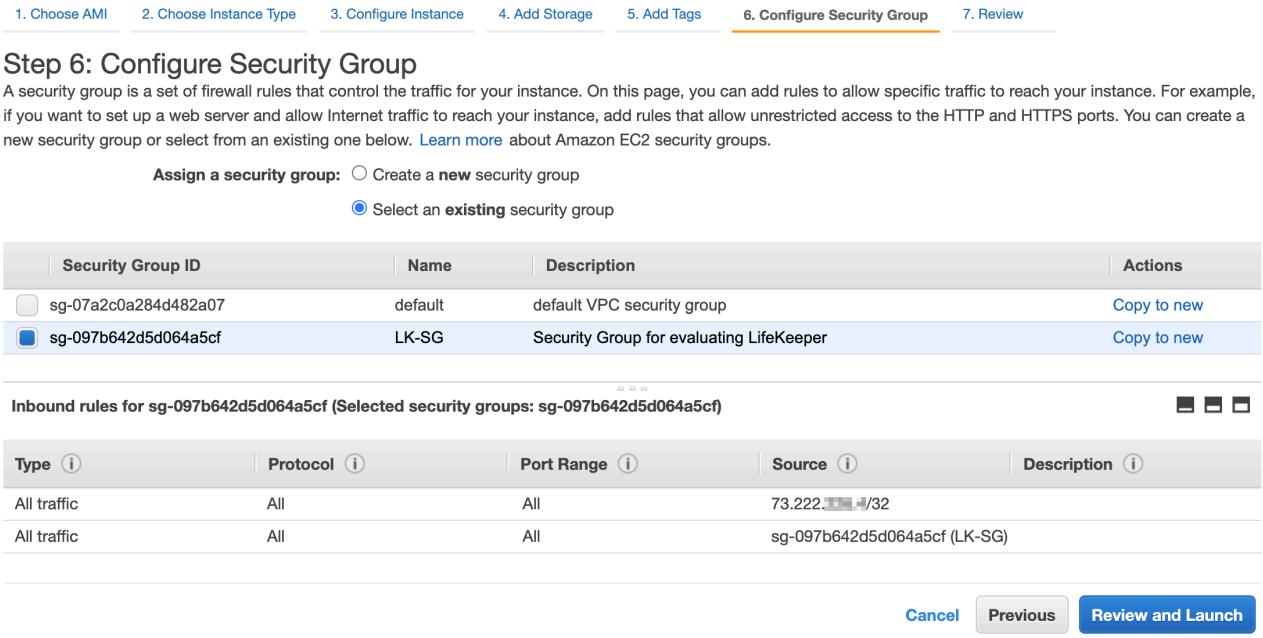

Once the selection is confirmed, click “Next: Add Tags”.

7. Add a tag with key “Name” and value “Node-A” to make it easier to identify the instance from the AWS EC2 Console.

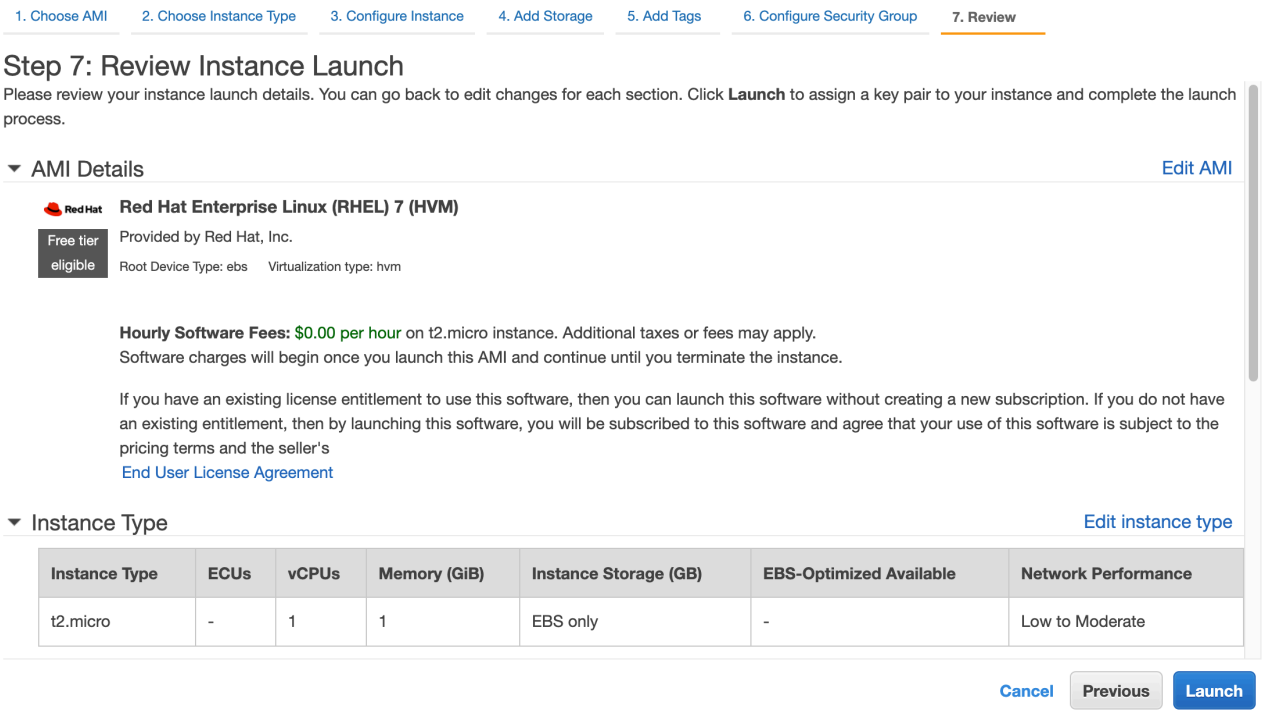



Once the “Name” tag is defined, click “Next: Configure Security Group”.

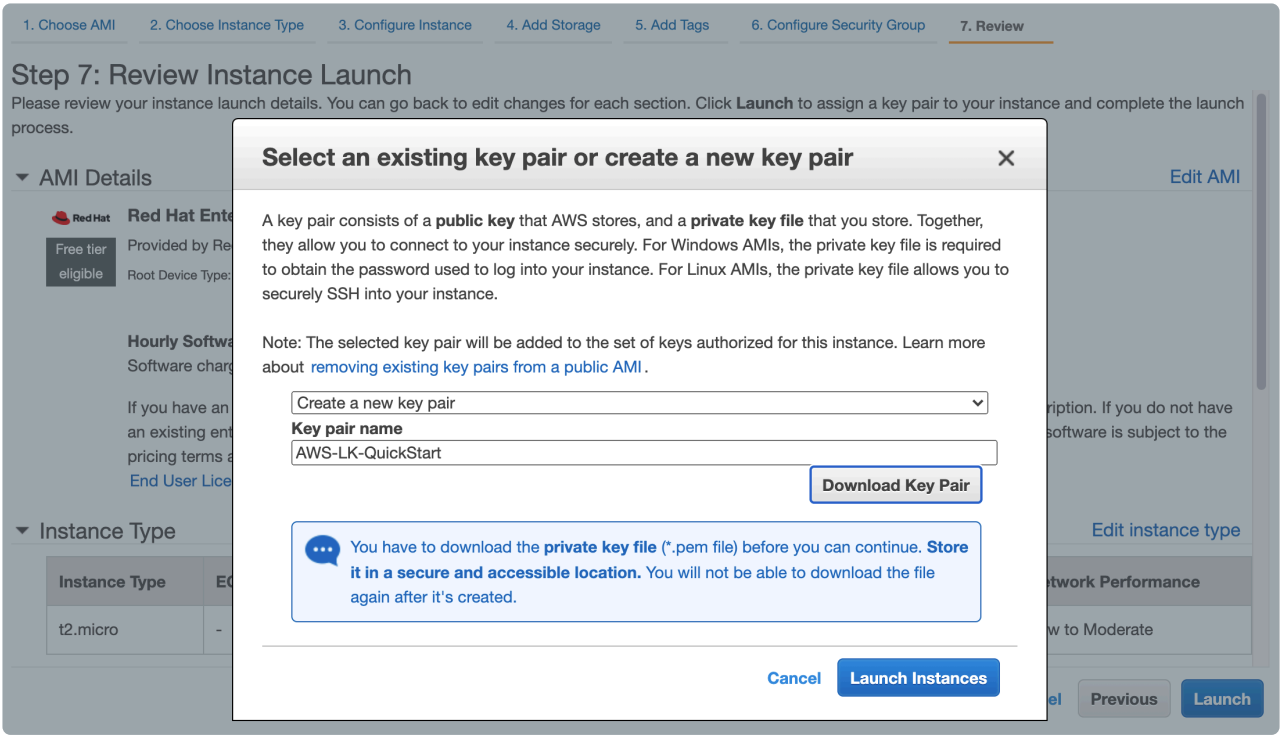
8. Select the Security Group previously created in an earlier section (LK-SG):



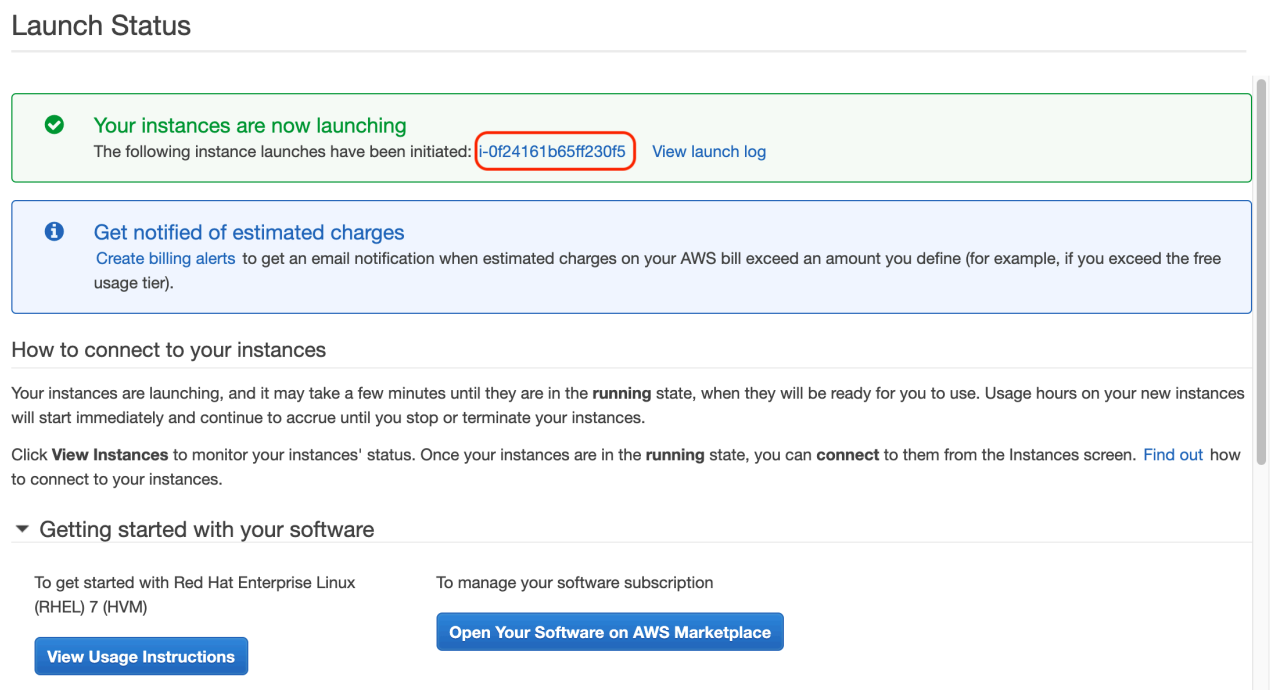

9. Review and confirm all selections, then click “Launch”.



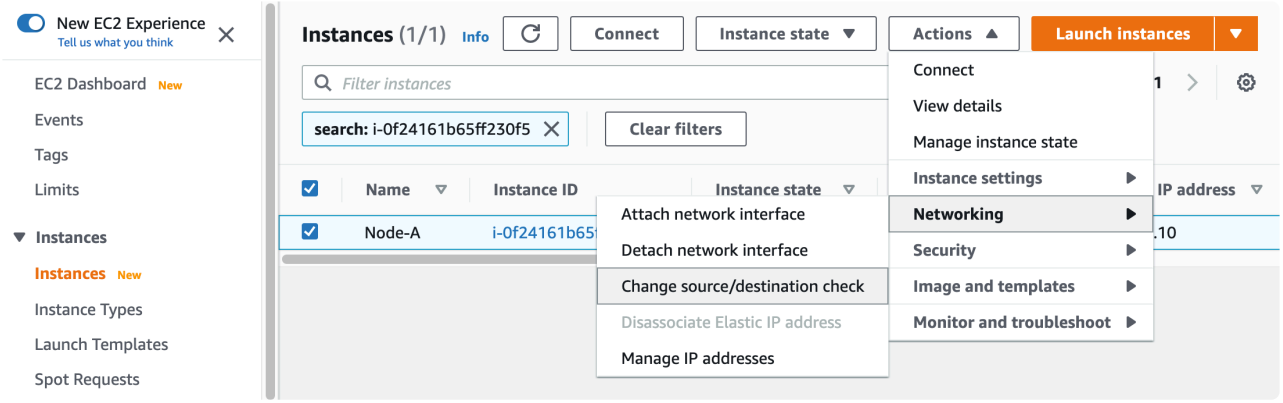

10. In order to access the newly created Linux EC2 instance, we will connect via ssh. AWS uses a key pair to authenticate user ssh sessions. Create a name for the key pair and download it to your local system. The private key will be needed to access the EC2 instance. Click “Launch Instances”.



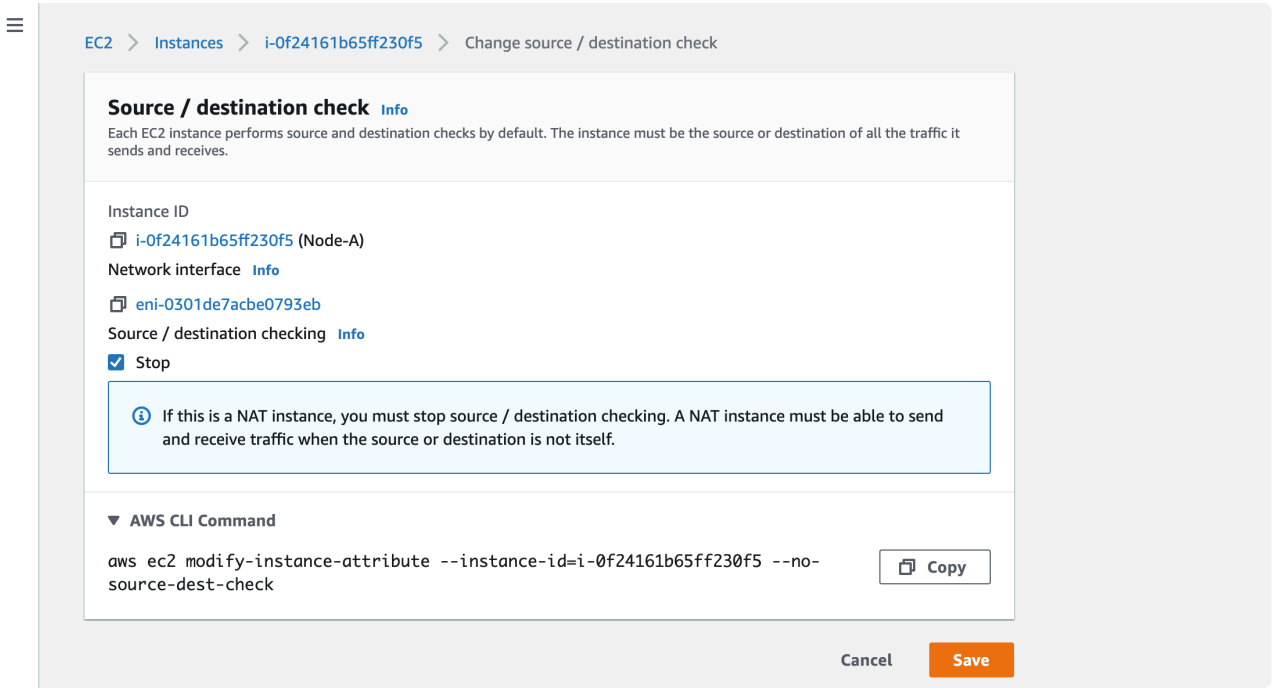

11. The “Launch Status” page appears. Select the instance ID to view its details.




12. Once the Instance is created we need to change the network configuration. As the active node changes from time to time and we are using a Virtual IP address, we should disable the source/destination check on the network interface. To do this, select the Instance, then select Actions > Networking > Change source/destination check.

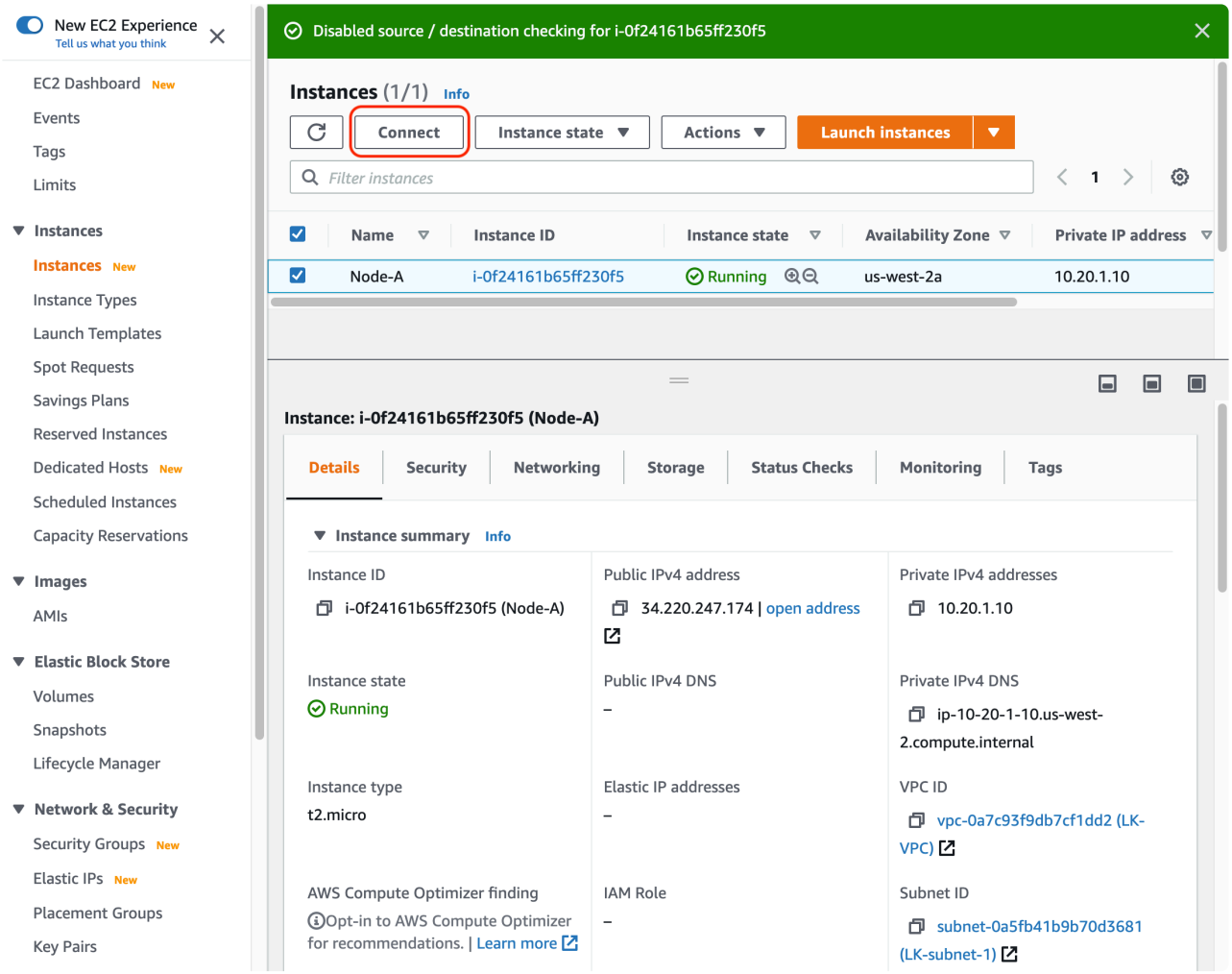



13. On the “Source / destination check” page, check “Stop” and save the change.

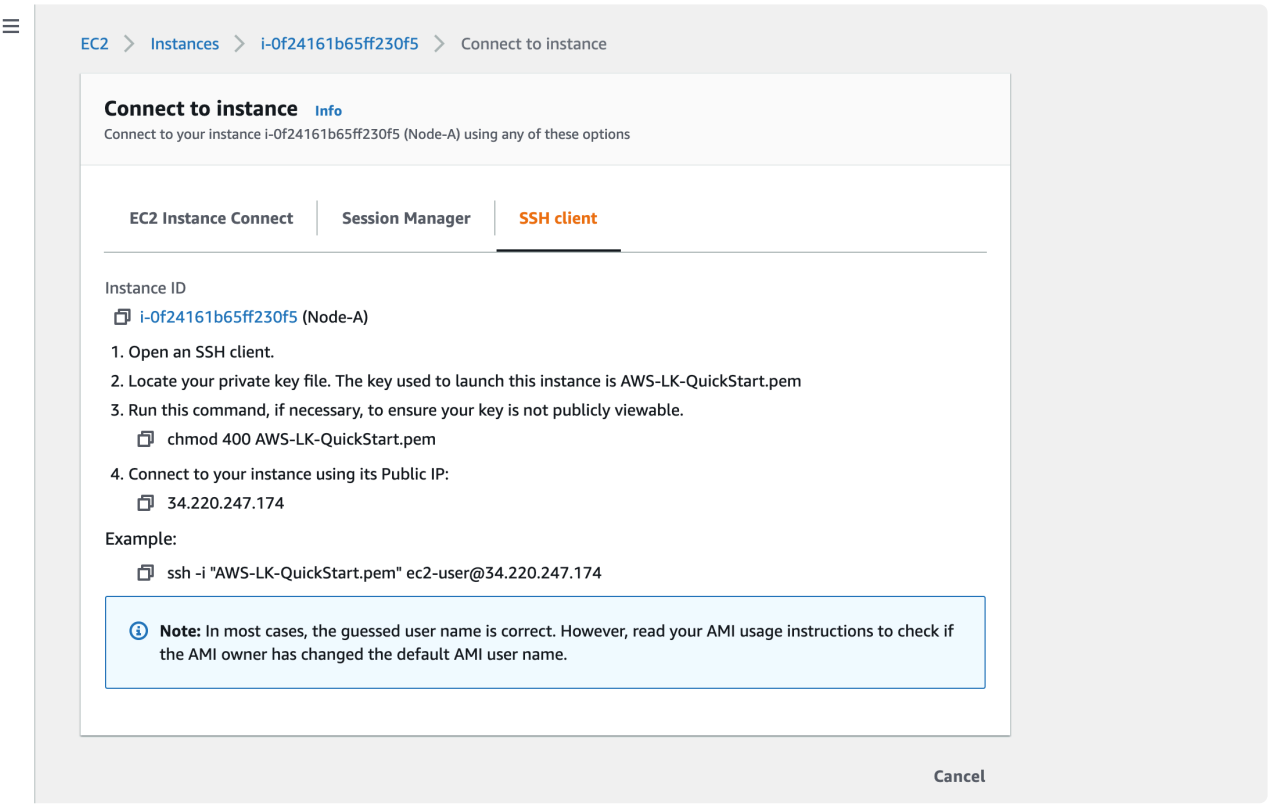



Now the instance is created and ready for us to connect.

14. The details of the instance may be reviewed on this page. Select “Connect” at the top of the page to see instructions on how to connect.




15. Here we can view the instructions that explain how to connect to the instance. On a Windows client, please refer to [Setup X Window client software on Microsoft Windows] for details.

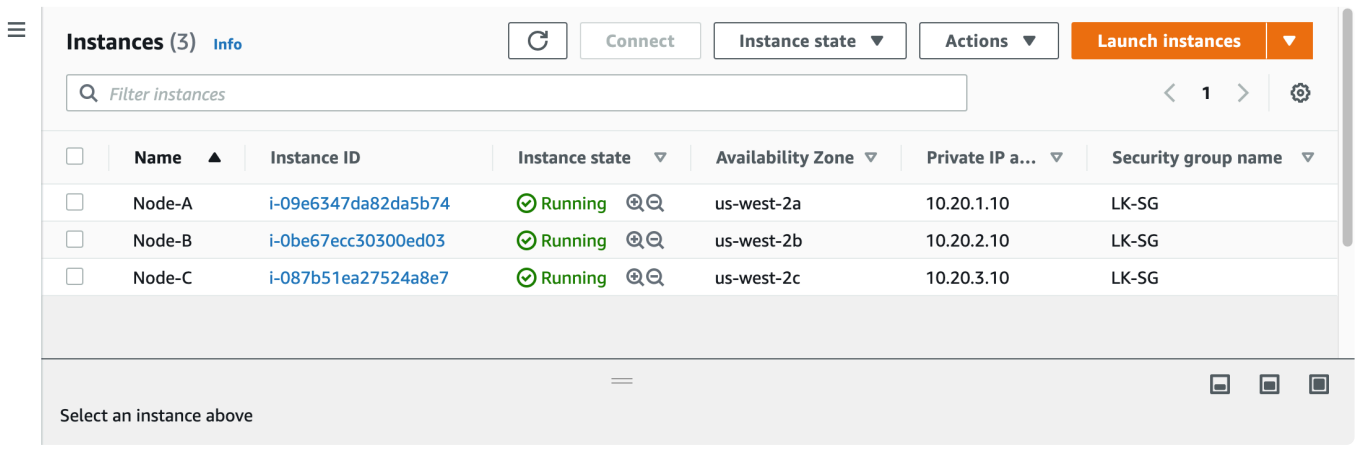


# 11.2.3.3.9. Creating the Second and Third Instances

The steps required to create the second node (node-b) and third node (node-c) are almost the same as previously described for the first node (node-a). The following table illustrates the differences between the three nodes and the details required to create these instances.

| Name | Parameter | Value |
|---|---|---|
| **Common values across instances** | **VPC** | **LK-VPC** |
| | **Security Group** | **LK-SG** |
| | **Source/Dest Checking on a network interface (ENI)** | **Disabled** |
| node-a | Subnet | LK-subnet-1 |
| | Private IP Address | `10.20.1.10` |
| | Second Disk (Storage Device) | You need to have a second disk |
| node-b | Subnet | LK-subnet-2 |
| | Private IP Address | `10.20.2.10` |
| | Second Disk (Storage Device) | You need to have a second disk |
| node-c | Subnet | LK-subnet-3 |
| | Private IP Address | `10.20.3.10` |
| | Second Disk (Storage Device) | You do not need to attach a second disk |

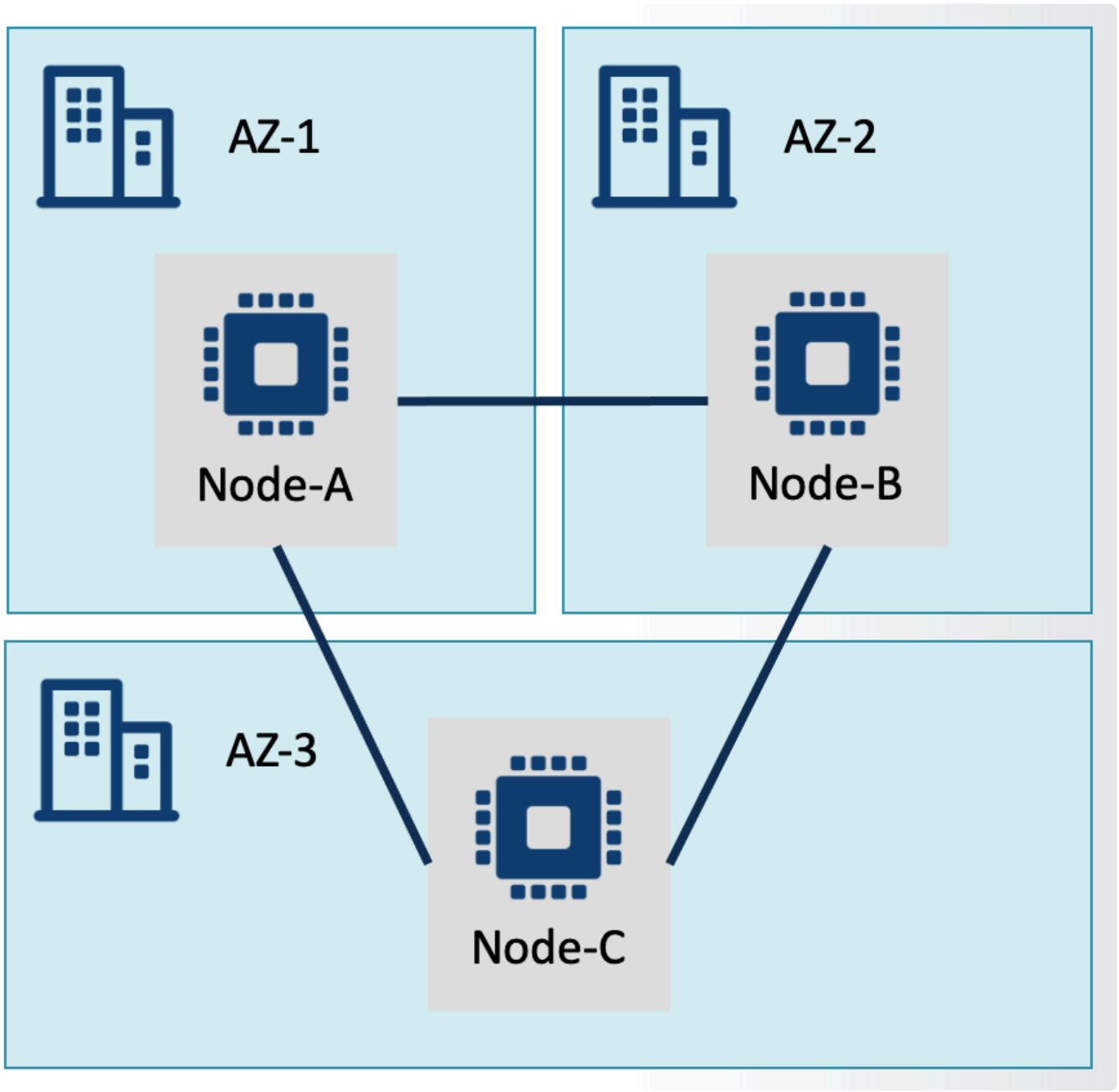
Once created, the list of instances should look like this:



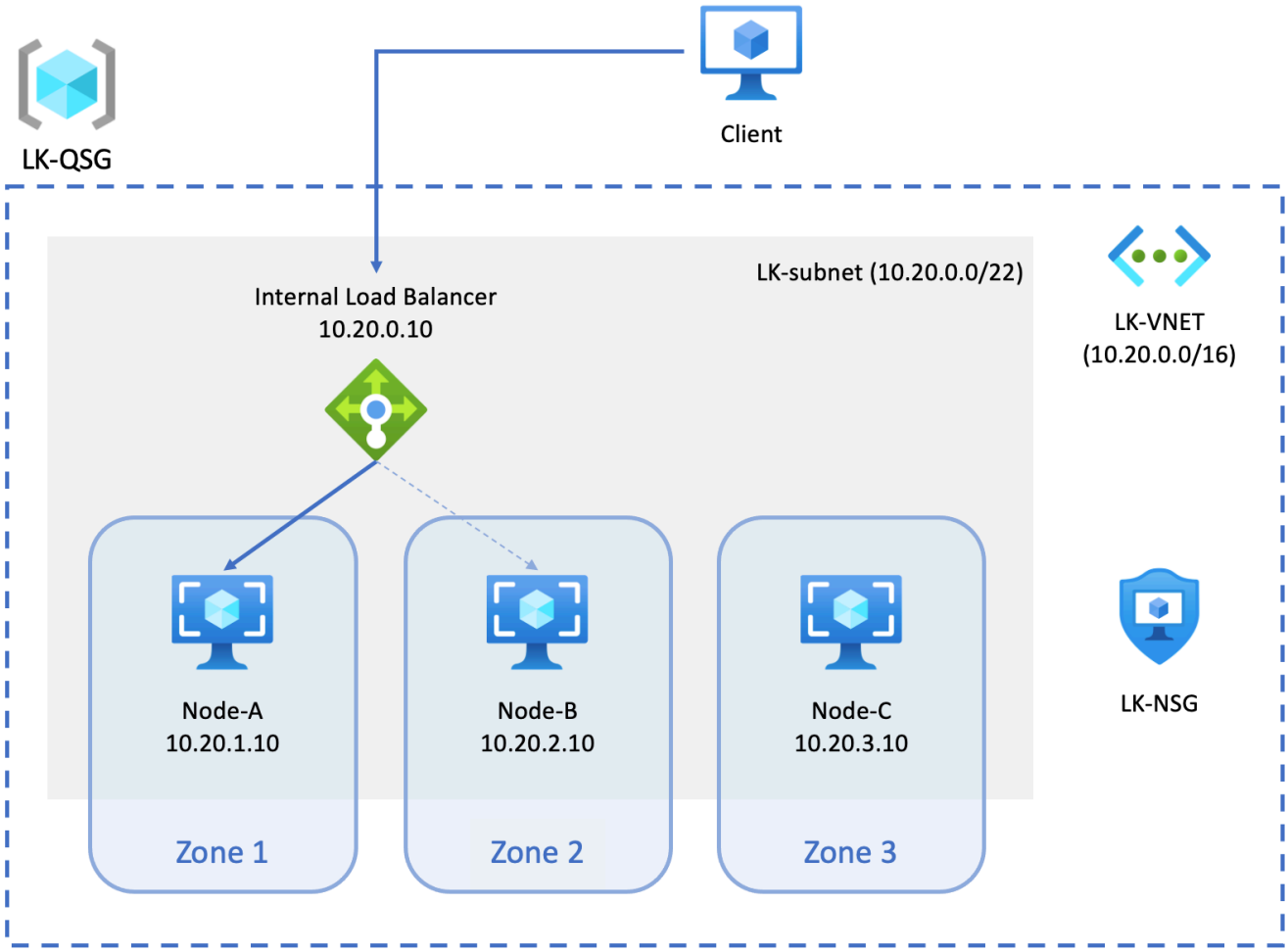
# 11.2.3.4. Creating an Instance in Azure from Scratch

**Disclaimer:** The user interface may vary between regions or change over time. Please refer to the documentation provided by Azure if the screenshots shown below are different from your experience.

The Evaluation Guide uses the following network structure and instances.




On Azure, these components can be defined as seen in the diagram below.




In this section we will create these components with the exception of the Internal Load Balancer. We will create the Internal Load Balancer later.

Components listed in this diagram are described in the following table.

To create this network structure, the following components need to be implemented:

| Component | Name | Parameter | Value |
|---|---|---|---|
| Resource Group | LK-QSG | | |
| Virtual network | LK-VNET | IPv4 CIDRs | 10.20.0.0/16 |
| | | Subnet (LK-subnet) | 10.20.0.0/22 |
| Network Security | LK-NSG | Additional Inbound Rule<br><br>Type=SSH | Source=Your Office's WAN IP<br><br>Destination=Any |

| | | | |
|---|---|---|---|
| | | Additional Outbound Rule<br><br>Type=Any | Source=VirtualNetwork<br><br>Destination=AzureLoadBalancer |
| Virtual Machine | Common values across VM | Resource Group | LK-QSG |
| | | Virtual network | LK-VNET |
| | | Subnet | LK-subnet |
| | | Network Security Group | LK-NSG |
| | node-a | Availability Zone (*1) | 1 |
| | | Private IP Address (*2) | `10.20.1.10` |
| | node-b | Availability Zone (*1) | 2 |
| | | Private IP Address (*2) | `10.20.2.10` |
| | node-c | Availability Zone (*1) | 3 |
| | | Private IP Address (*2) | `10.20.3.10` |

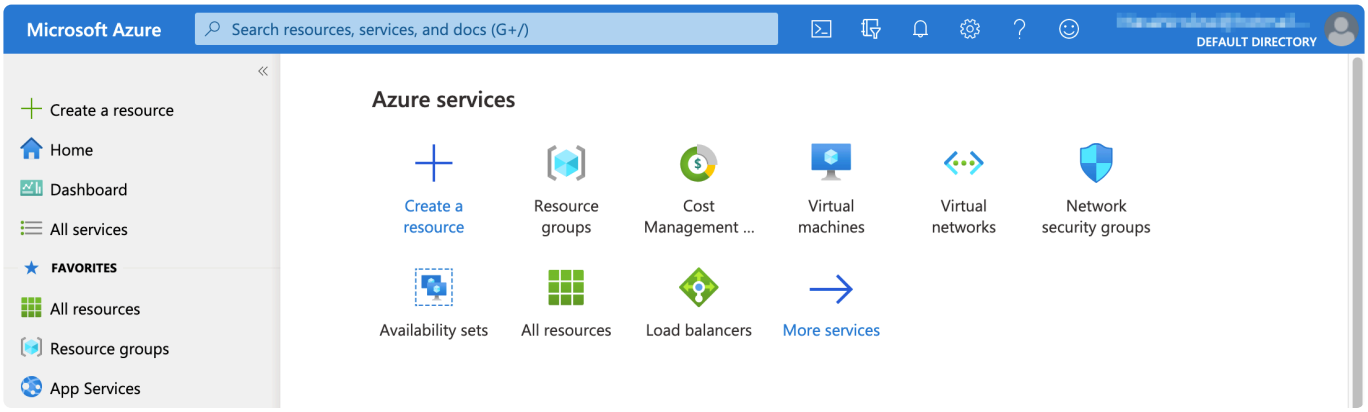
- **(*1)** This MUST be chosen at the time of creating a virtual machine.
- **(*2)** Wizard will automatically assign a DHCP based IP address. This will need to be modified after creating a virtual machine.

The following sections describe the steps necessary for creating these network components, beginning with the first instance.

- Switching between Azure Services
- Deciding on an Azure Region
- Creating the Resource Group
- Creating a Virtual Network
- Creating a Network Security Group
- Creating the First Azure Virtual Machine
- Creating the Second and Third Virtual Machines

# 11.2.3.4.1. Switching between Azure Services

Azure has a significant and growing number of services available. Typical services that you would use are accessible from the navigation bar located on the left hand side or listed in the home screen (select “Home” at the navigation bar).




If you are unable to find a particular service, you may type in the name of the service in the search box at the top.

# 11.2.3.4.2. Deciding on an Azure Region

Azure has regions in many geographic locations. It may be beneficial to select a region in close geographic proximity to your workplace location. Note that not all regions have 3 Availability Zones, so exercise caution when making a selection in order to guarantee that the chosen region supports the intended configurations. When you create a Resource Group or other resources in later steps, please make sure to select the correct region.

Refer to the Azure documentation for more information.

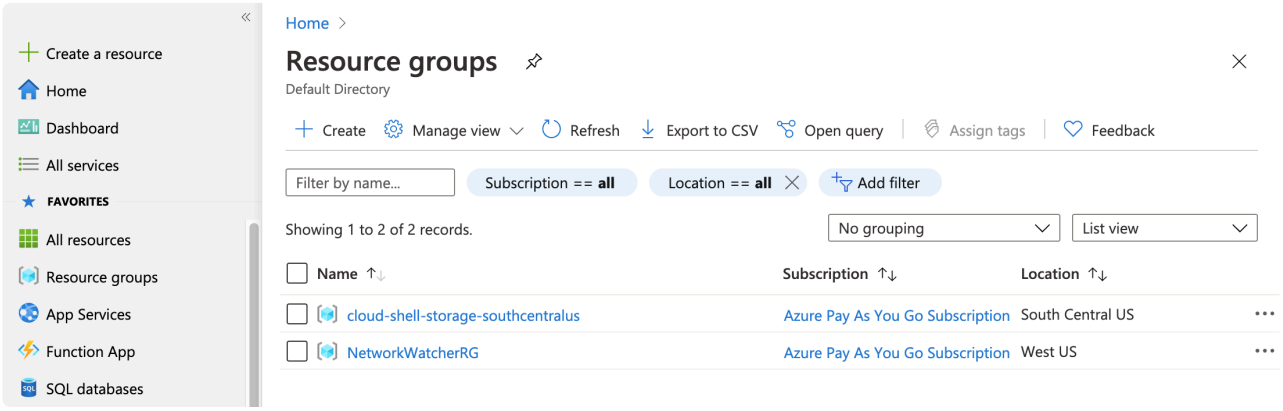
# 11.2.3.4.3. Creating the Resource Group

A Resource Group is the logical container in which you can create resources such as network components, computer resources, storage, etc.

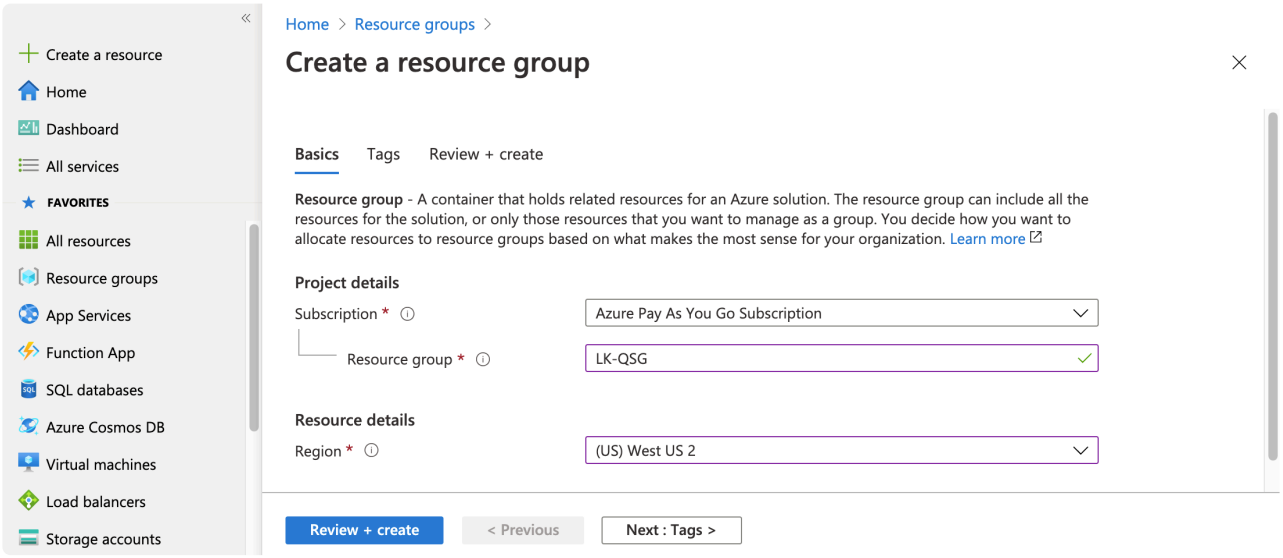
Refer to the Azure documentation for more information.

In this section, we will create a Resource Group for testing LifeKeeper (we will name it `LK-QSG`) as follows:

1. Select “Resource groups” from the home screen to browse the list of existing resource groups, then click “Create” located at the top.

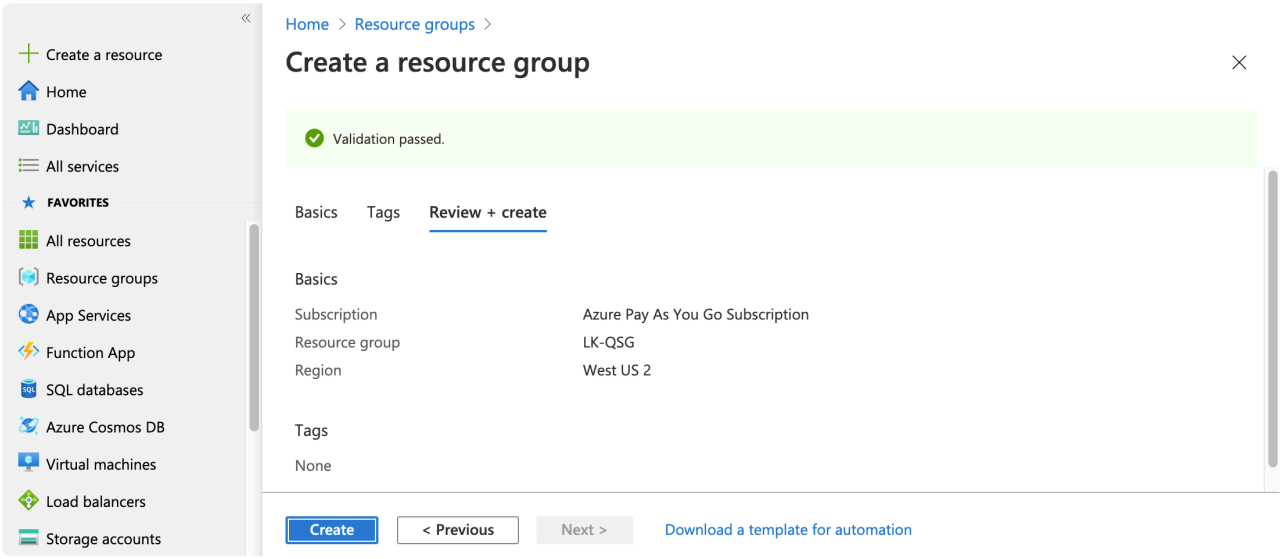



2. Specify the name of Resource group as `LK-QSG`, then select the Region to create the resource group. Note that the region used to create the resource group must have 3 availability zones.



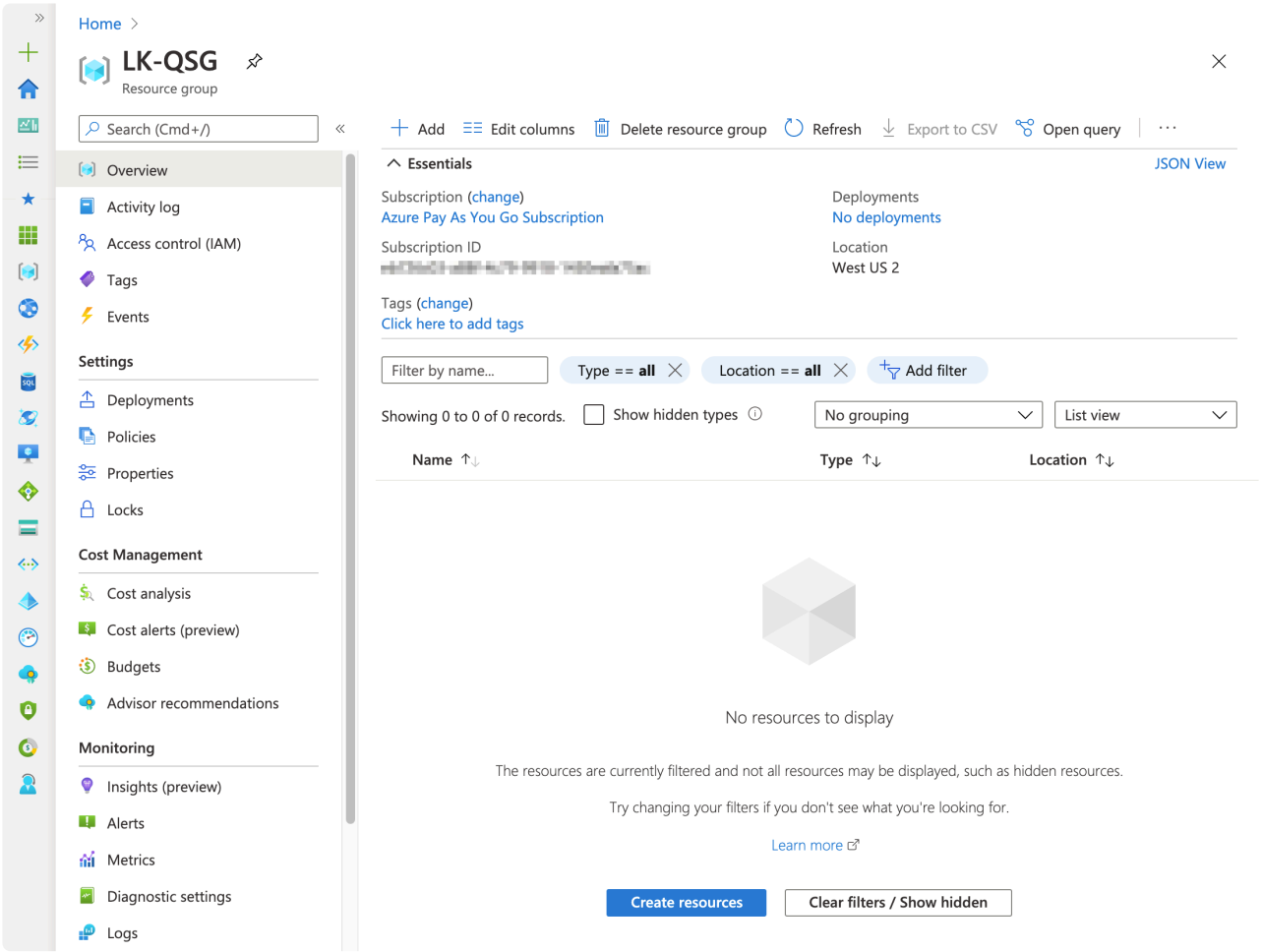

Once you make the selections, click “Review + Create”.

3. The wizard evaluates these values and you can now create the resource if the validation passes.




Click “Create” to create the resource group.

4. Now the resource group is created.



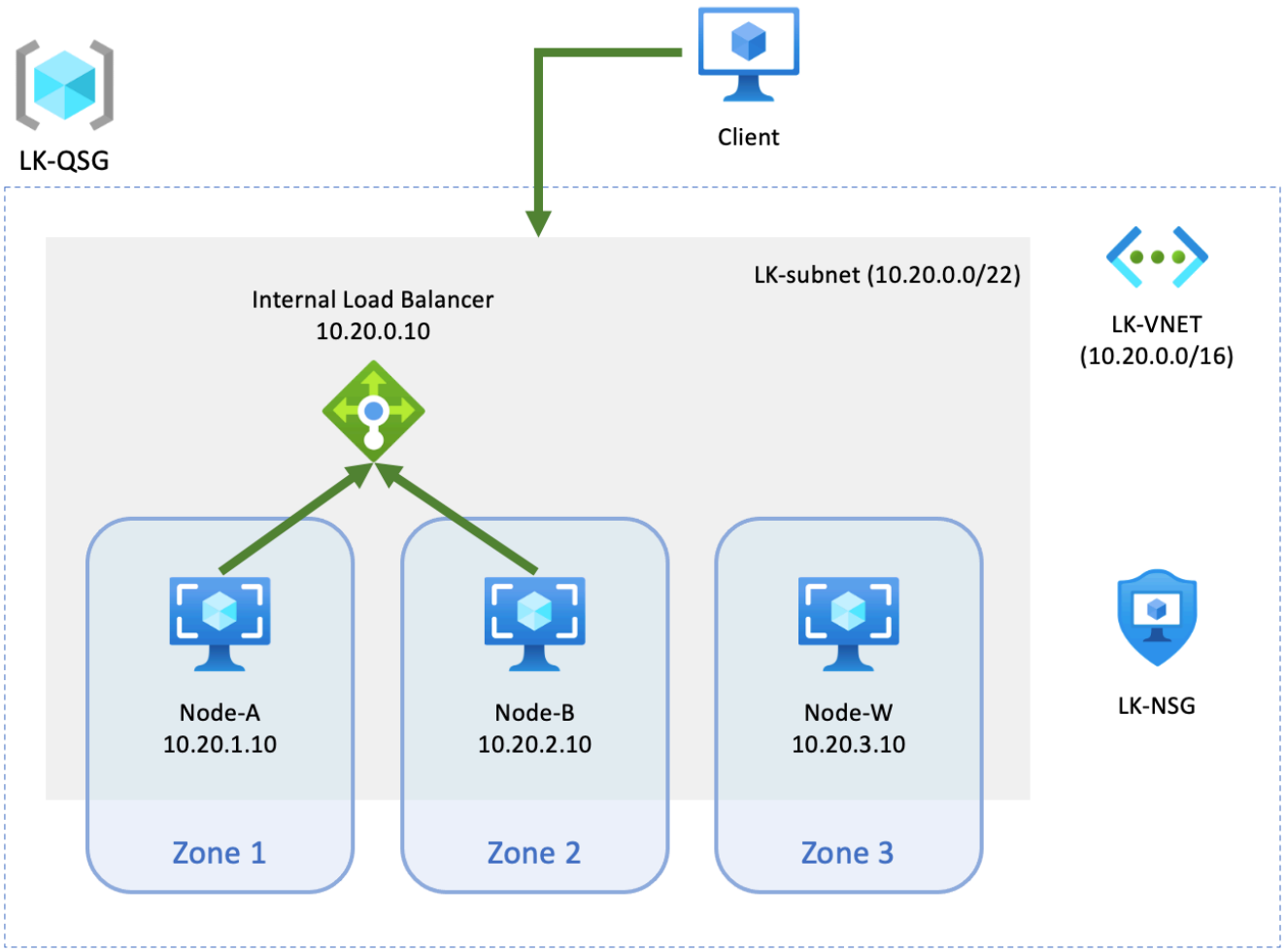
# 11.2.3.4.4. Creating a Network Security Group

A Security Group works as a firewall, allowing you to define both “allow” and “deny” rules. A source can be an internet address or a security group, and a security group can be assigned to a Virtual Machine or a subnet.

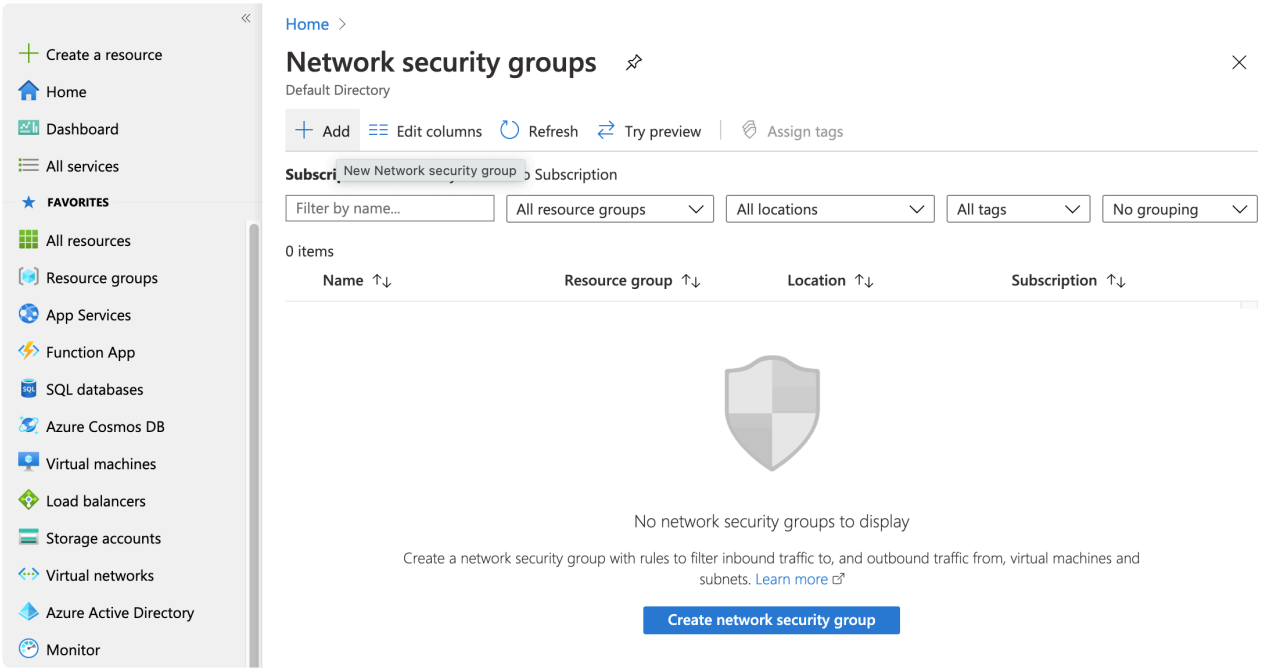
Refer to the Azure documentation for more information.

A new security group already has several “allow” and “deny” rules (those are defined with priority 65xxx). We will add following additional rules to “allow” access (see the green arrows).

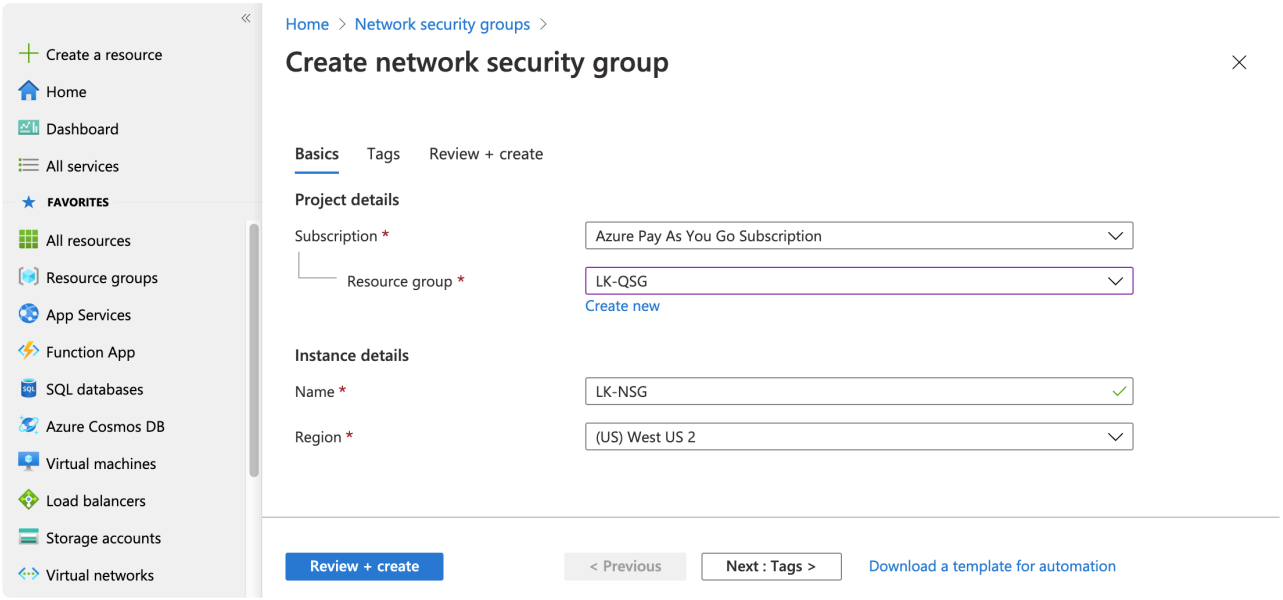
- Allow traffic from the work location to LK-subnet
- Allow traffic from Virtual Machines to Internal Load Balancer




1. Select “Network Security Group” from the home screen and click “Add” to create a new one.

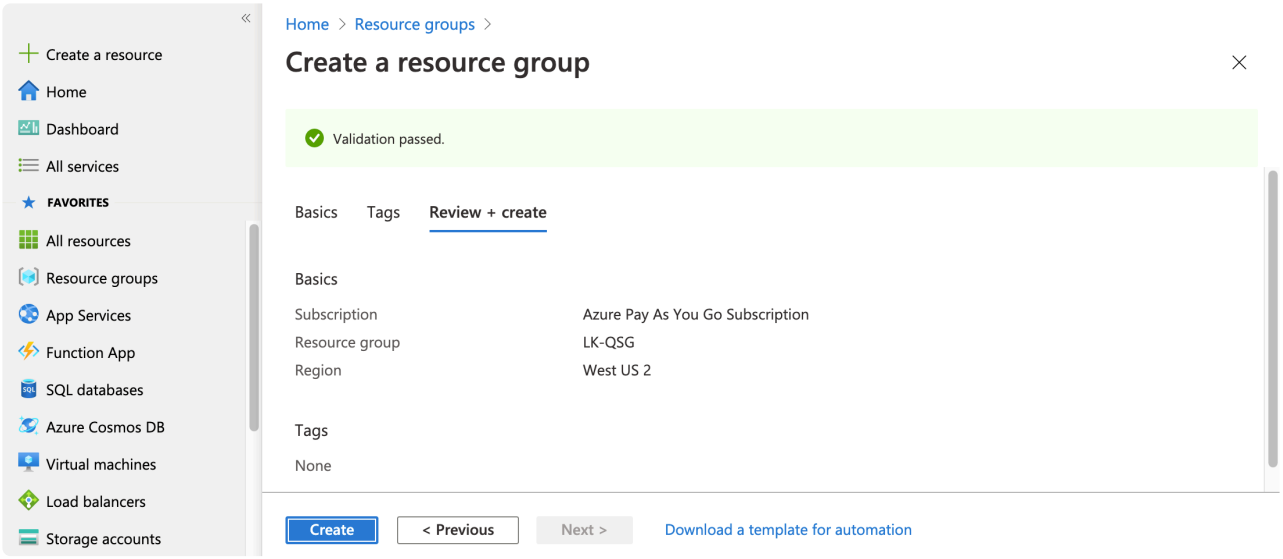



2. Select the Resource Group `LK-QSG`, specify the name of new security group as `LK-NSG`, then select a region.

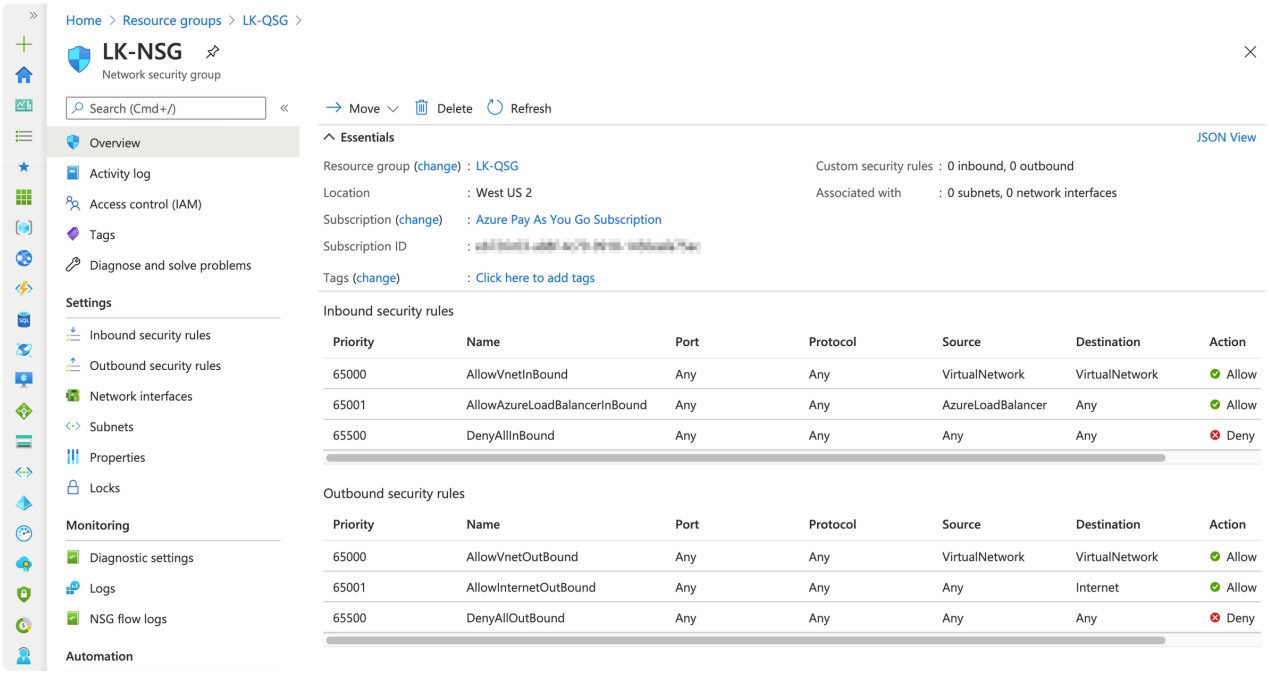



Once you specify the values, click “Review + create”.

3. Once the wizard validates your input, click “Create”.

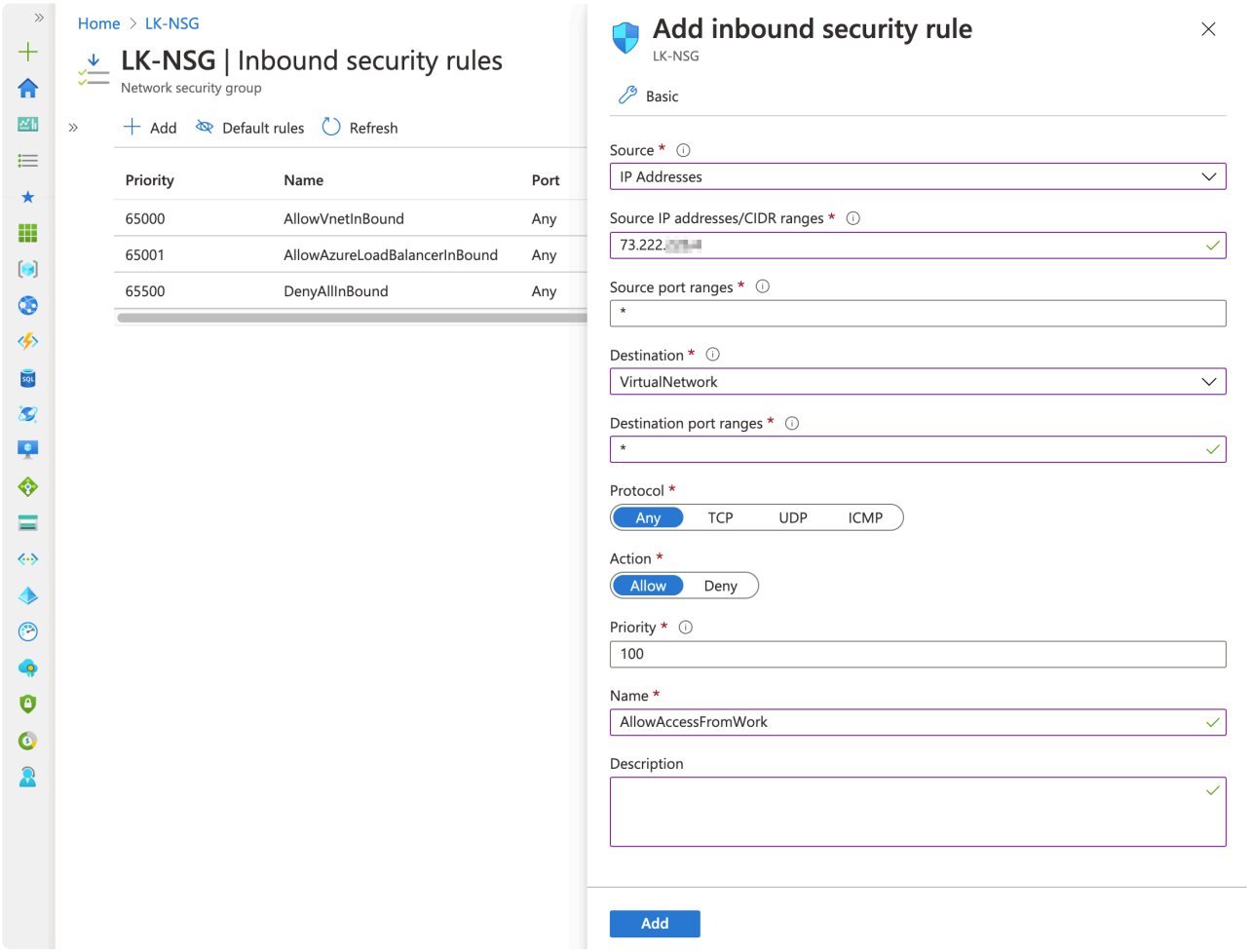



4. The new security group `LK-NSG` is now created. You can see that there are several inbound and outbound rules already defined.




5. Select the Inbound security rules from the left side and click “Add” to enter the new rule. Enter the following values:

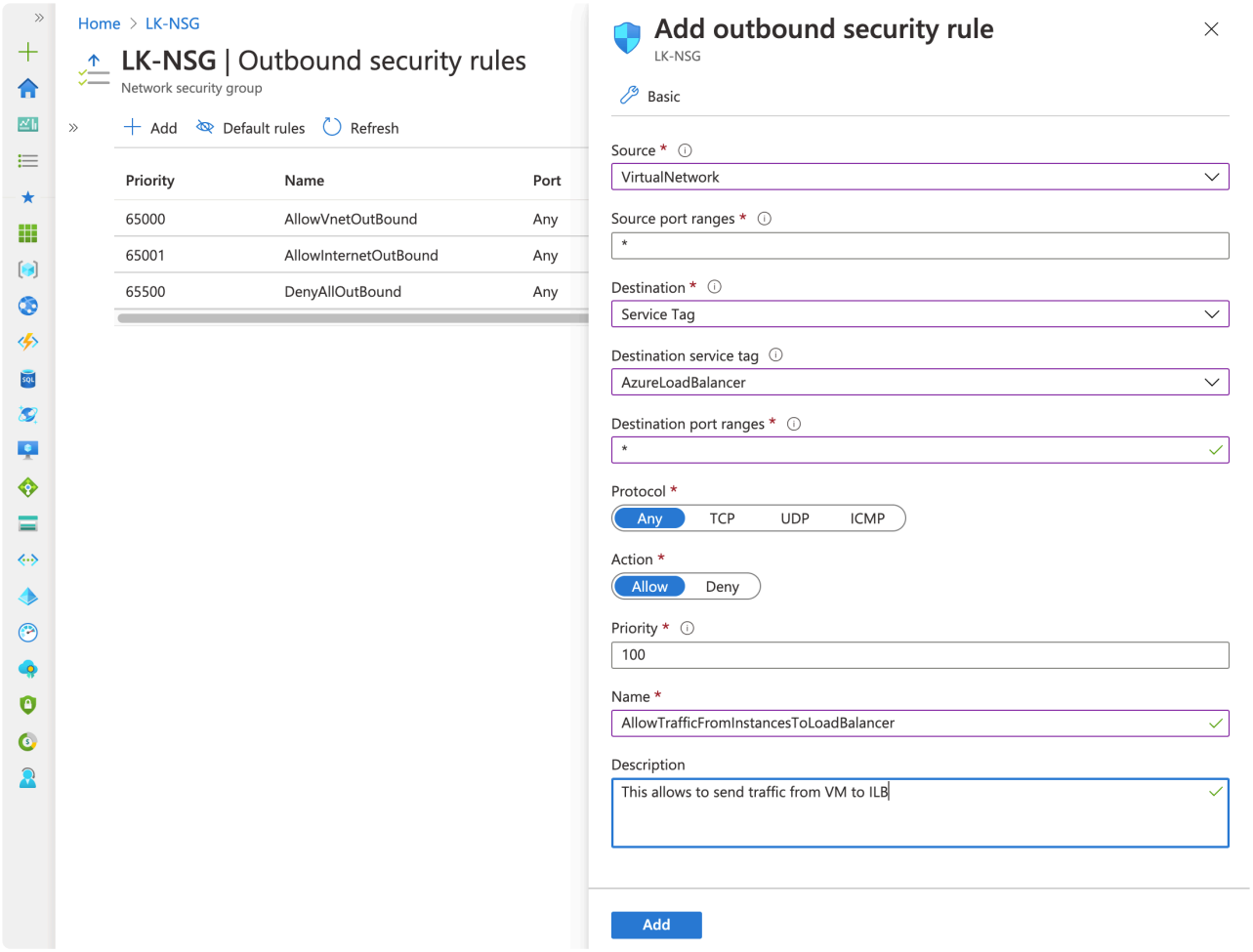
- Source IP: Your WAN IP address(es)
- Source Port Range: *
- Destination: Virtual Network
- Destination Port Ranges: *
- Name: AllowAccessFromWork (any name that represents this configuration is fine)




After you enter the values, click “Add”.

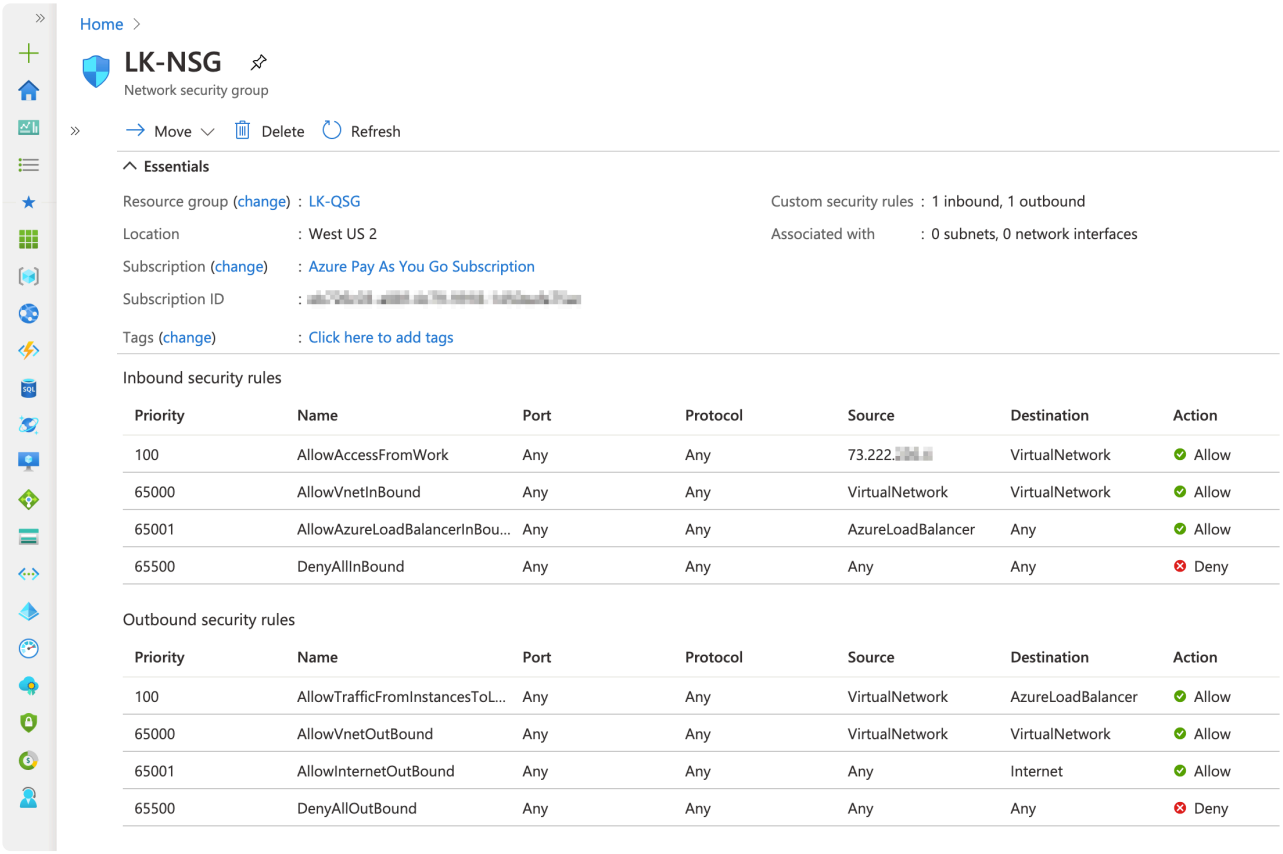
6. Select Outbound security rules from the left side and click “Add” to enter the new rule. Enter the following values:

- Source: VirtualNetwork
- Source port range: *
- Destination: Service Tag
- Destination Service Tag: AzureLoadBalancer
- Destination port ranges: *
- Name: AllowTrafficFromInstancesToLoadBalancer (any name that represents this configuration is fine)




After you enter the values, click “Add”.

7. Now the new Security group is defined.




Inbound security rules

| Priority | Name | Port | Protocol | Source | Destination | Action |
|---|---|---|---|---|---|---|
| 100 | AllowAccessFromWork | Any | Any | 73.222. | VirtualNetwork | Allow |
| 65000 | AllowVnetInBound | Any | Any | VirtualNetwork | VirtualNetwork | Allow |
| 65001 | AllowAzureLoadBalancerInBou... | Any | Any | AzureLoadBalancer | Any | Allow |
| 65500 | DenyAllInBound | Any | Any | Any | Any | Deny |

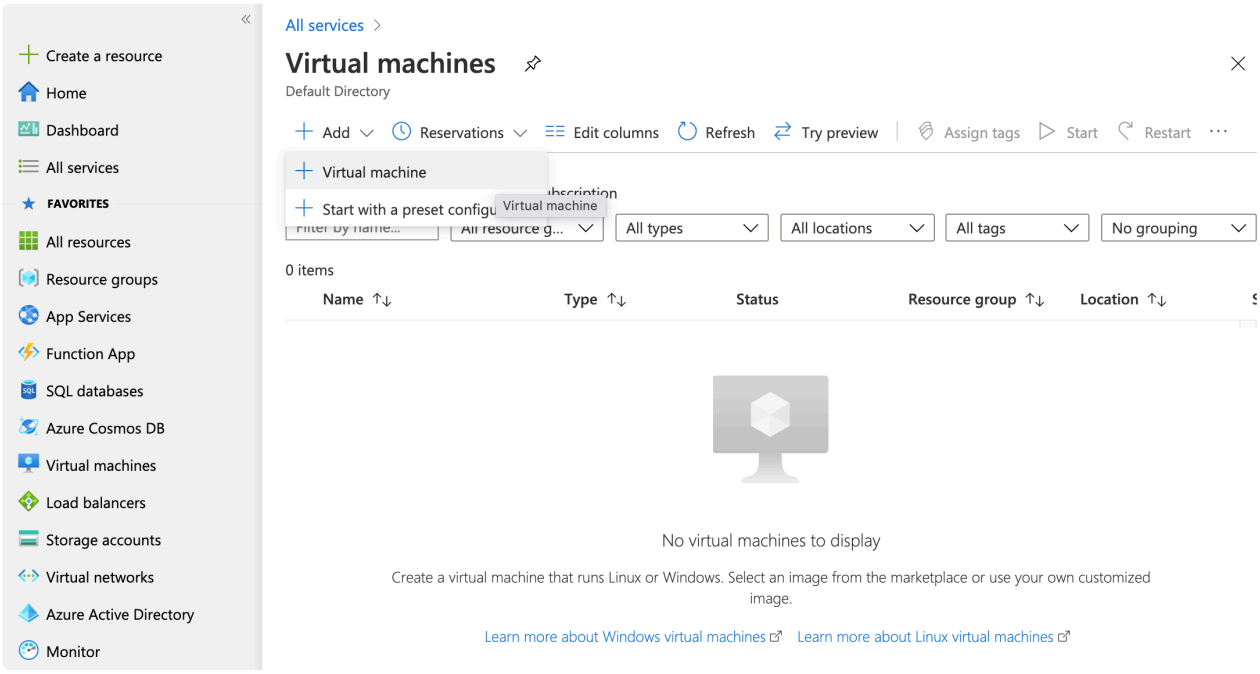
Outbound security rules

| Priority | Name | Port | Protocol | Source | Destination | Action |
|---|---|---|---|---|---|---|
| 100 | AllowTrafficFromInstancesToL... | Any | Any | VirtualNetwork | AzureLoadBalancer | Allow |
| 65000 | AllowVnetOutBound | Any | Any | VirtualNetwork | VirtualNetwork | Allow |
| 65001 | AllowInternetOutBound | Any | Any | Any | Internet | Allow |
| 65500 | DenyAllOutBound | Any | Any | Any | Any | Deny |

# 11.2.3.4.5. Creating the First Azure Virtual Machine

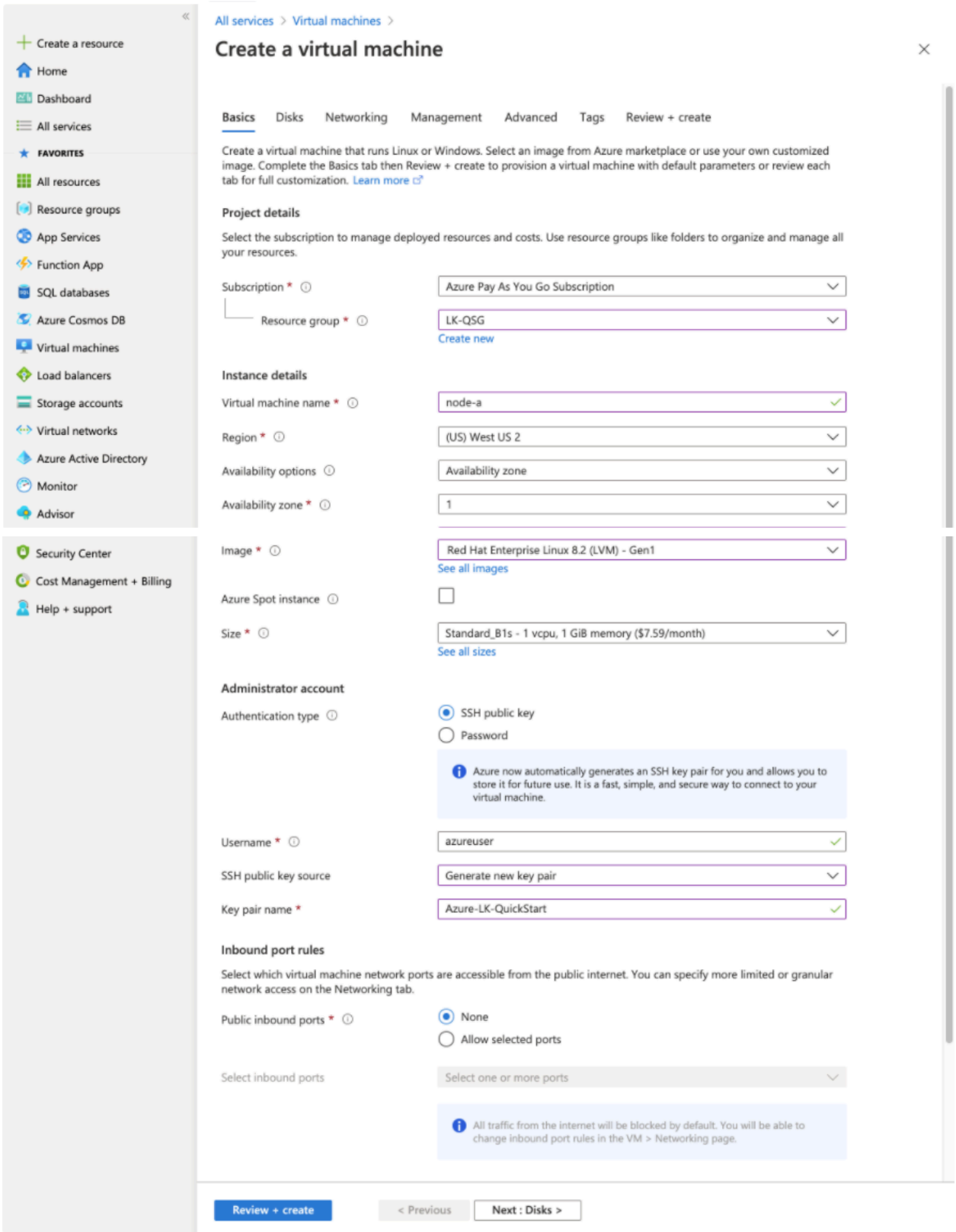
In previous sections we covered configuration of the network. Now we are going to create the first instance. As discussed in Computing Resources Used in this Tutorial, we need two disks. This section also discusses how to create the second disk.

1. Go to “Virtual Machine” and click “Add” to create a new VM.



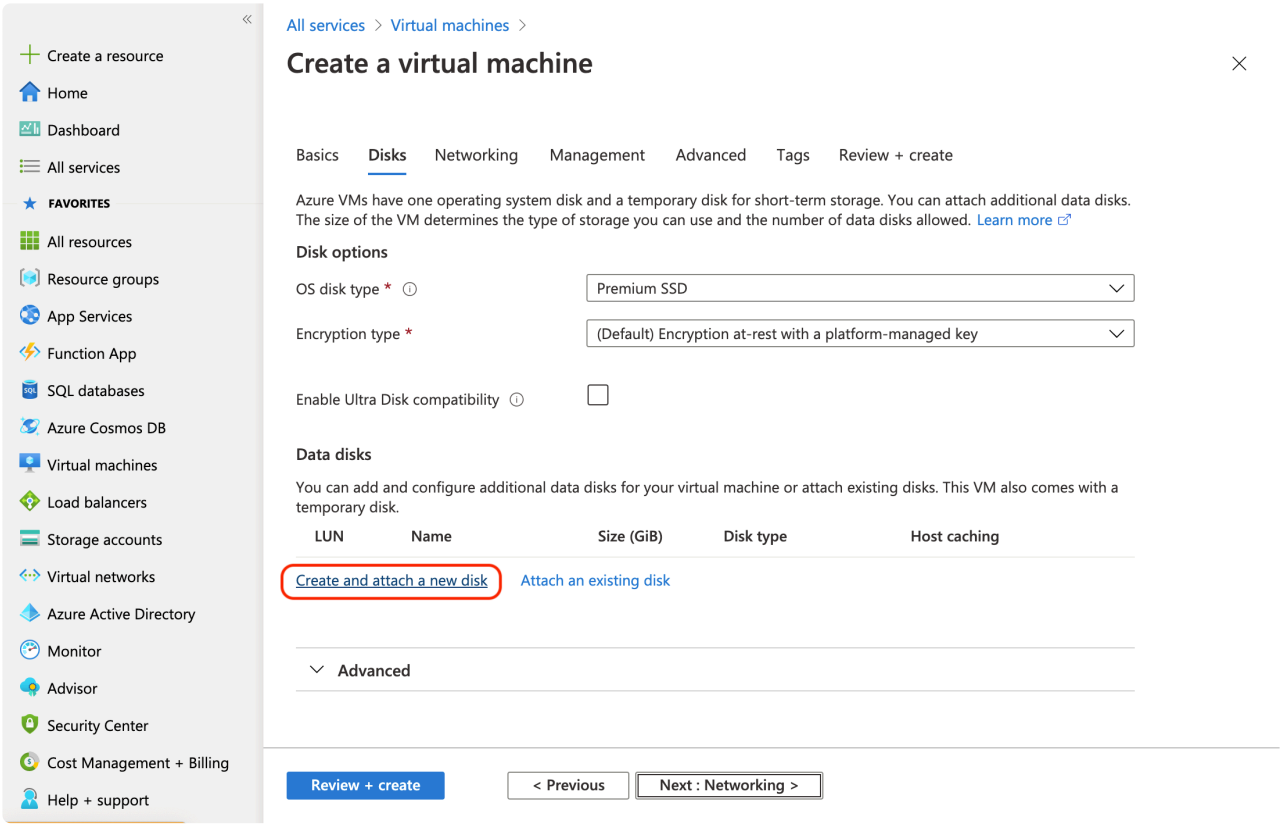

2. Enter the following values:

- Resource group: `LK-QSG`
- Virtual machine name: `node-a` (when you create 2nd or 3rd nodes, this would be `node-b` and `node-c`)
- Region: Specify region based on your decision earlier
- Availability options: `Availability Zone`
- Availability Zone: `1` (when you create 2nd or 3rd nodes, this would be `2` and `3`)
- Image: Select supported operating system
- Size: You can select image size based on the application you are going to protect. Minimum Requirement here is `Standard_B1s`
- Authentication type: You can select each option. In general, cloud providers recommend using `SSH public key`.
- username: `azureuser` (you can leave the default value)
- SSH public key source: `Generate new key pair` if you have not created a key before
- Key pair name: specify name (this screenshot uses `Azure-LK-QuickStart`)
- Public inbound ports: `None`



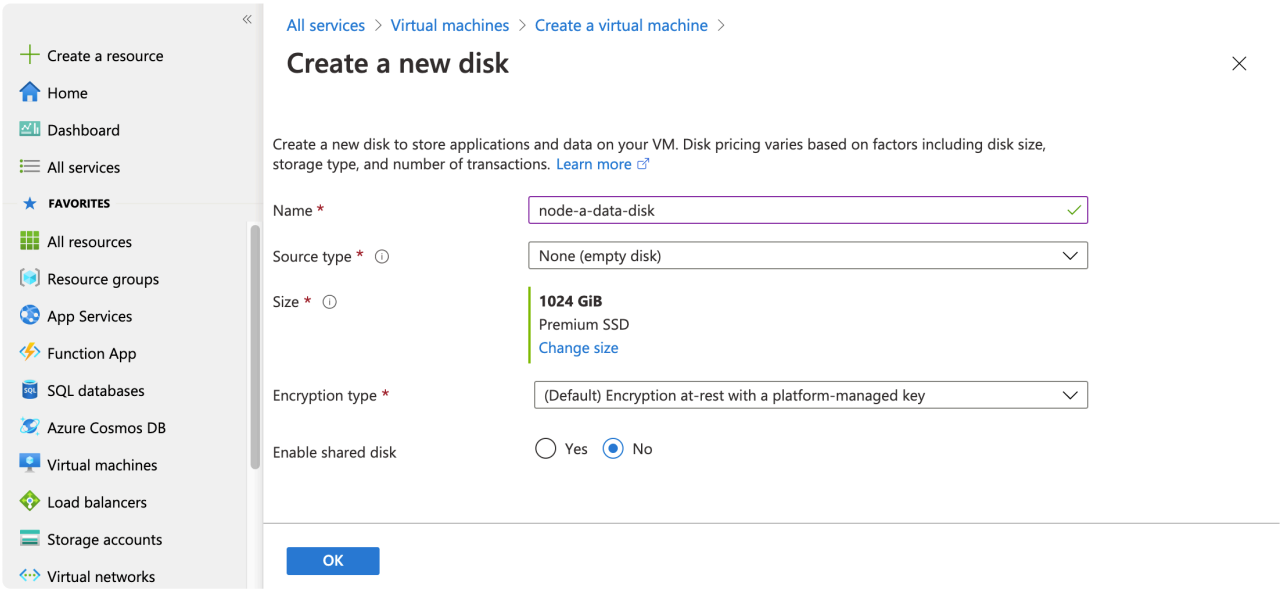

Once you enter or select these values, click “Next: Disks”.

3. By default, the wizard creates one disk to boot the operating system. You need an additional disk for data replication.



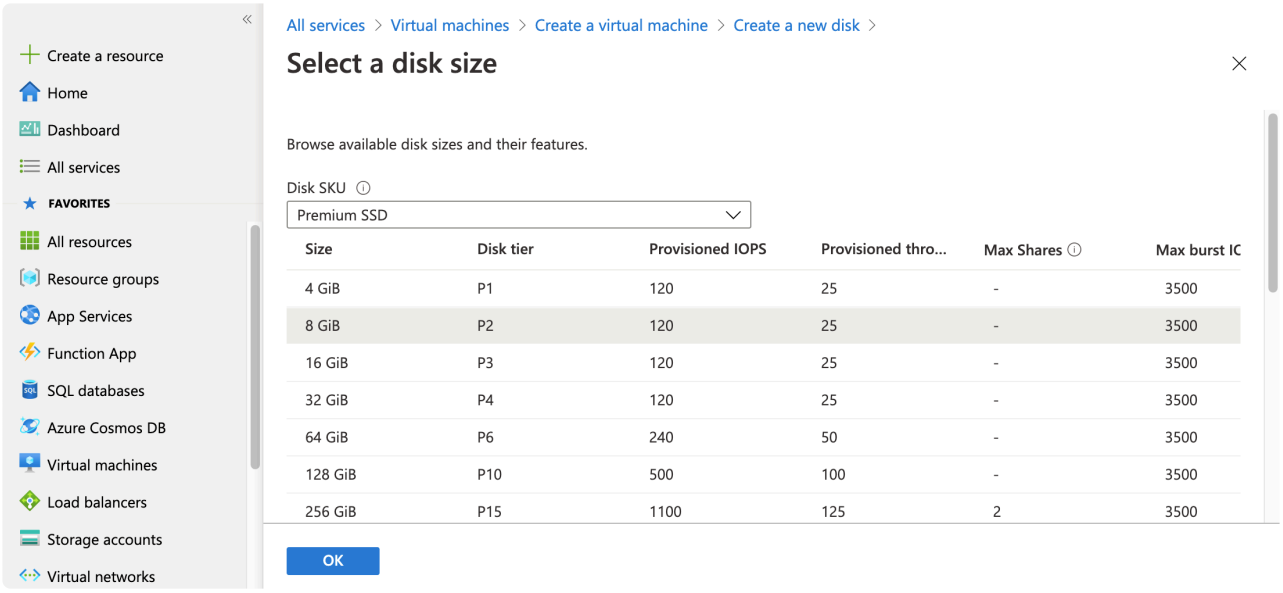

Click “Create and attach a new disk”.

4. Select the disk size. To evaluate SIOS Protection Suite, please create at least an 8 GiB disk.

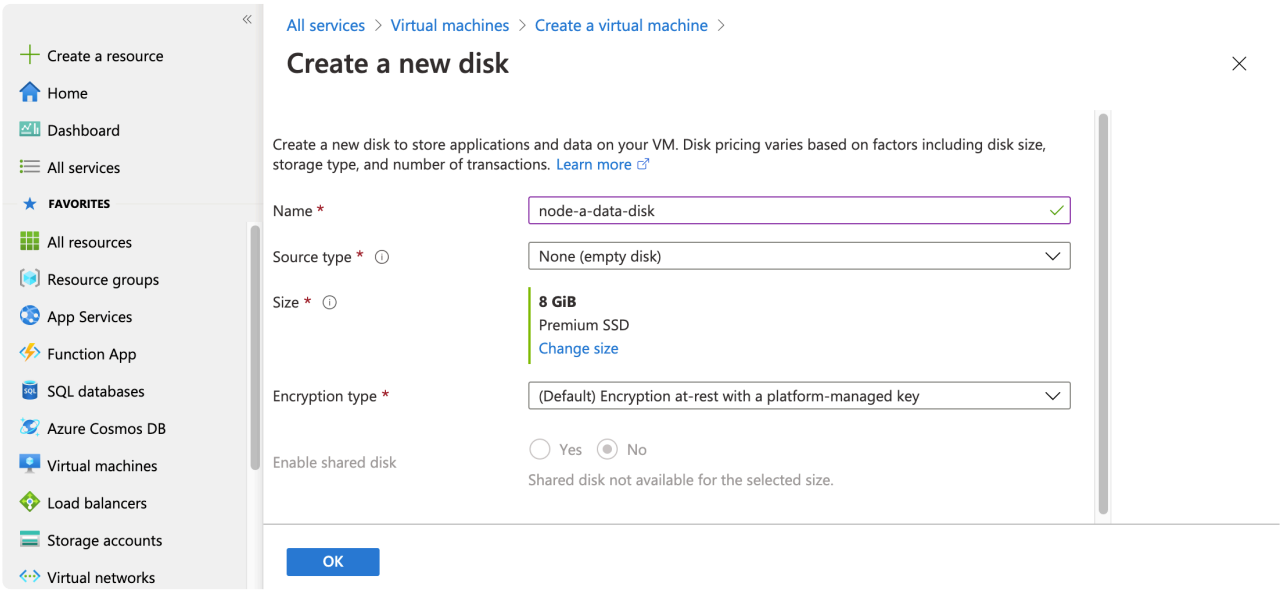



Click “Change size” to specify the disk size.

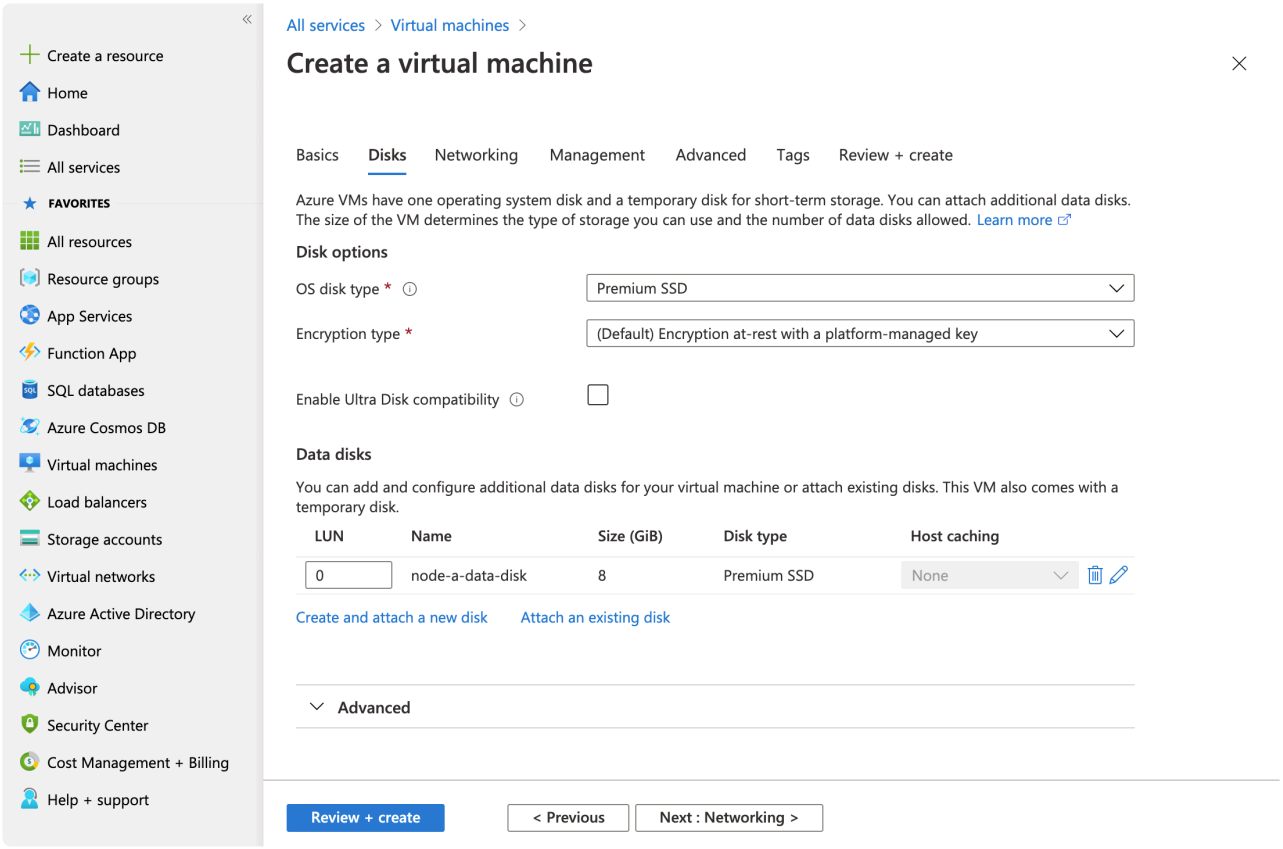
5. Select the disk size (select at least 8 GiB if you are evaluating SIOS Protection Suite), then select “OK”.




6. Now the disk size is specified. Click “OK” to confirm.

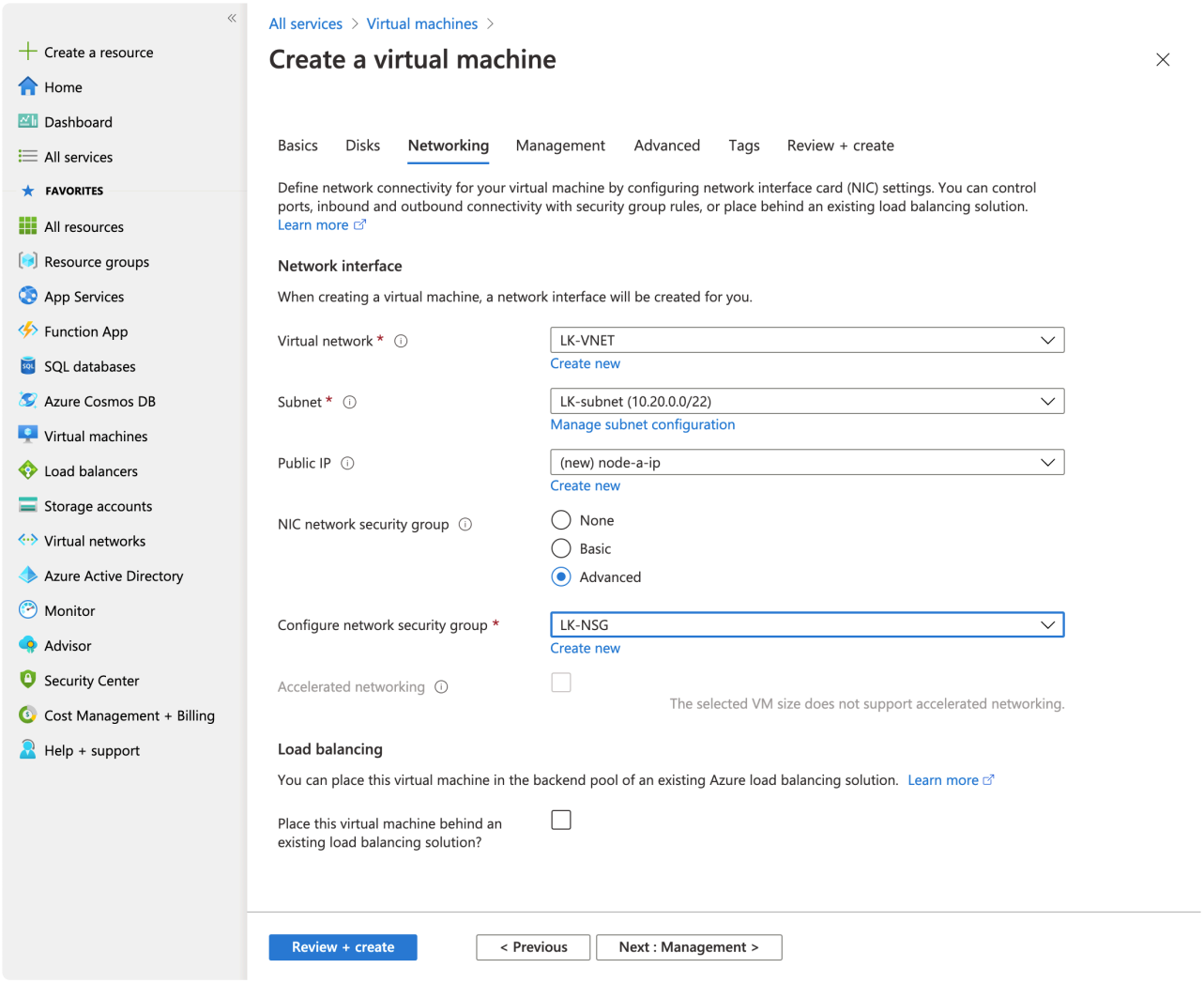



7. Once the disk configuration is finished, click “Next: Networking”.



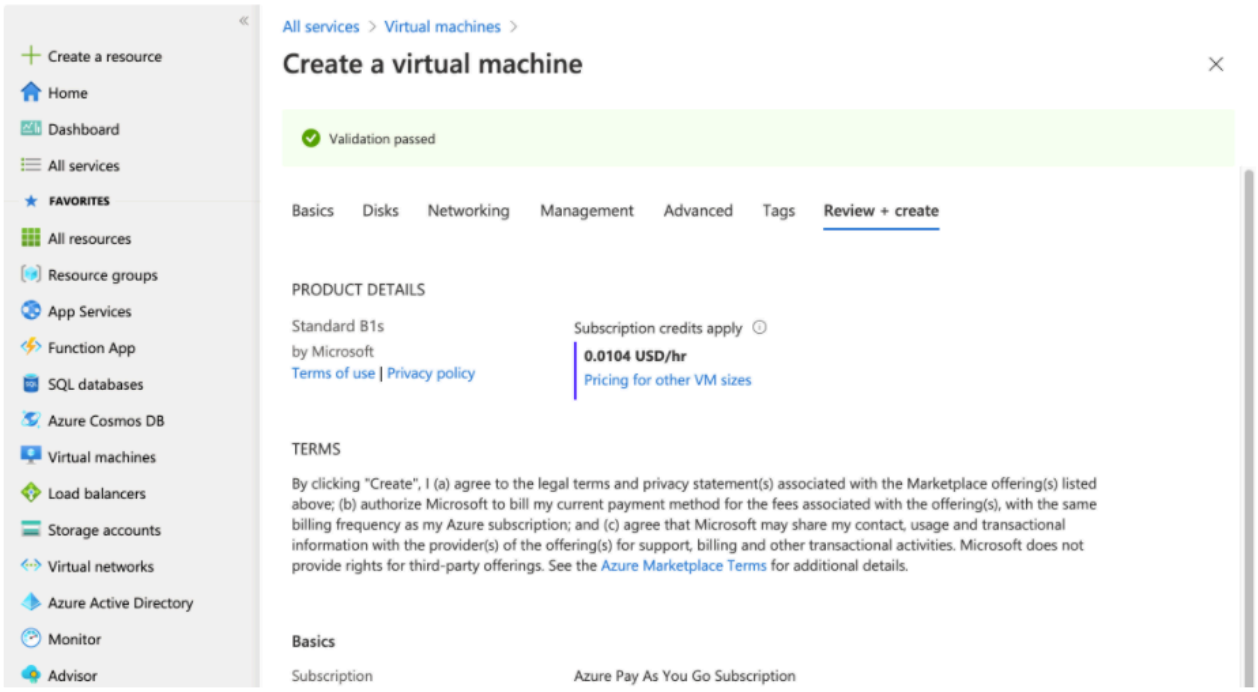

8. Select the following parameters for the network:

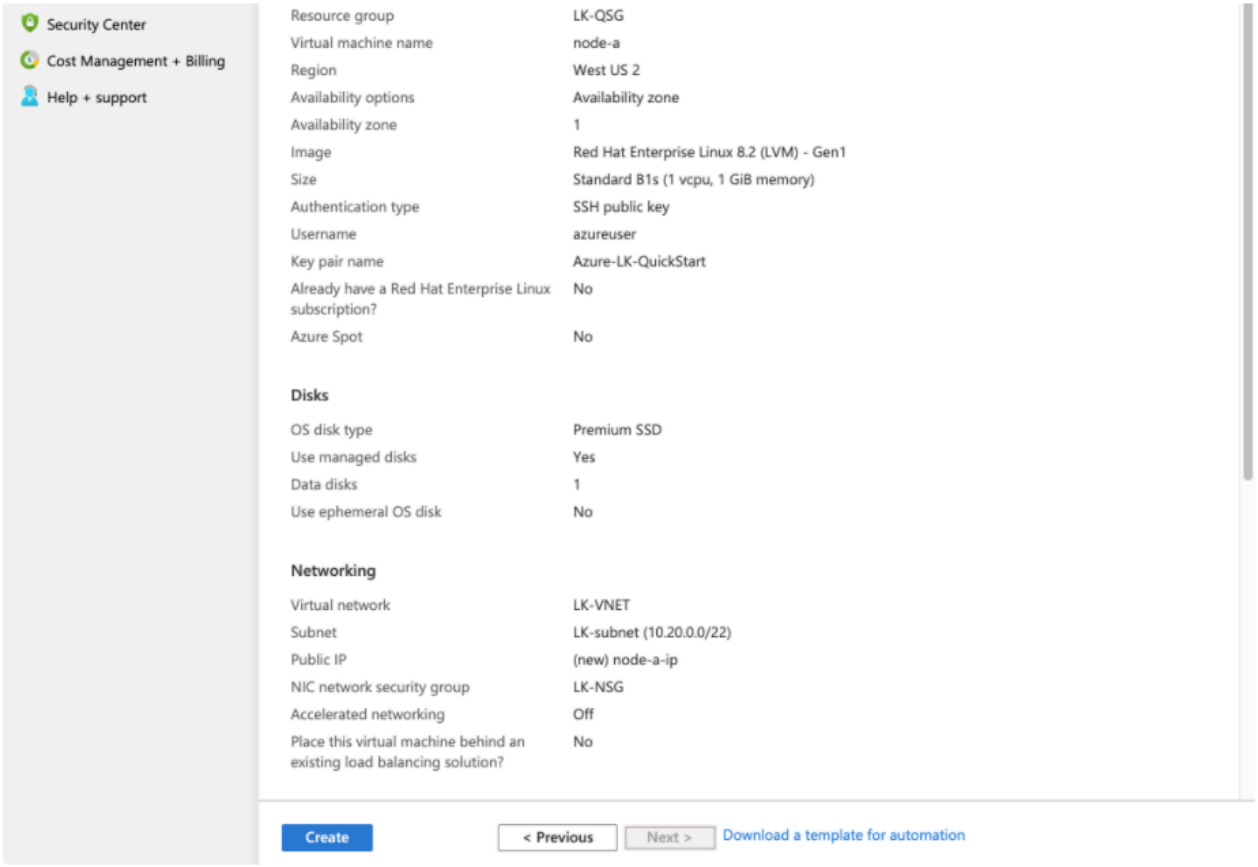
- Virtual network: `LK-VNET`
- Subnet: `LK-subnet`
- Public IP: `(new) node-a-ip`
- NIC network security group: `Advanced`
- Configure network security group: `LK-NSG`



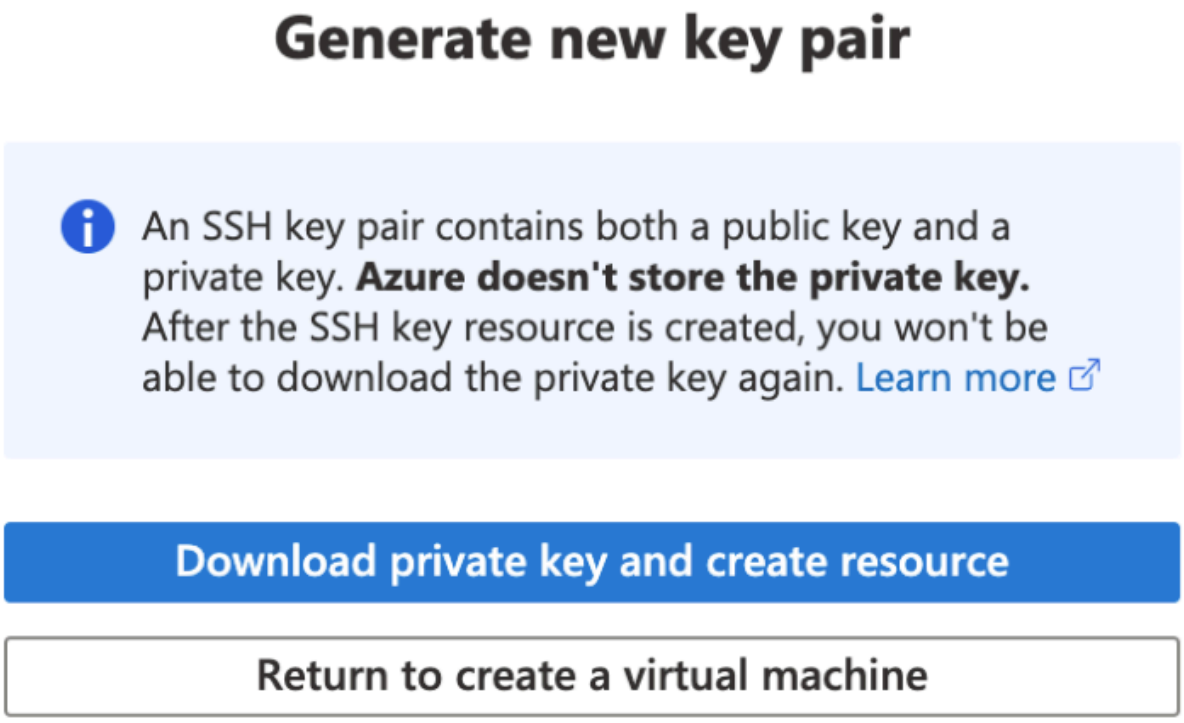

Once you specify the values, click “Review + create” to complete.

9. Once the wizard validates your input, click “Create”.



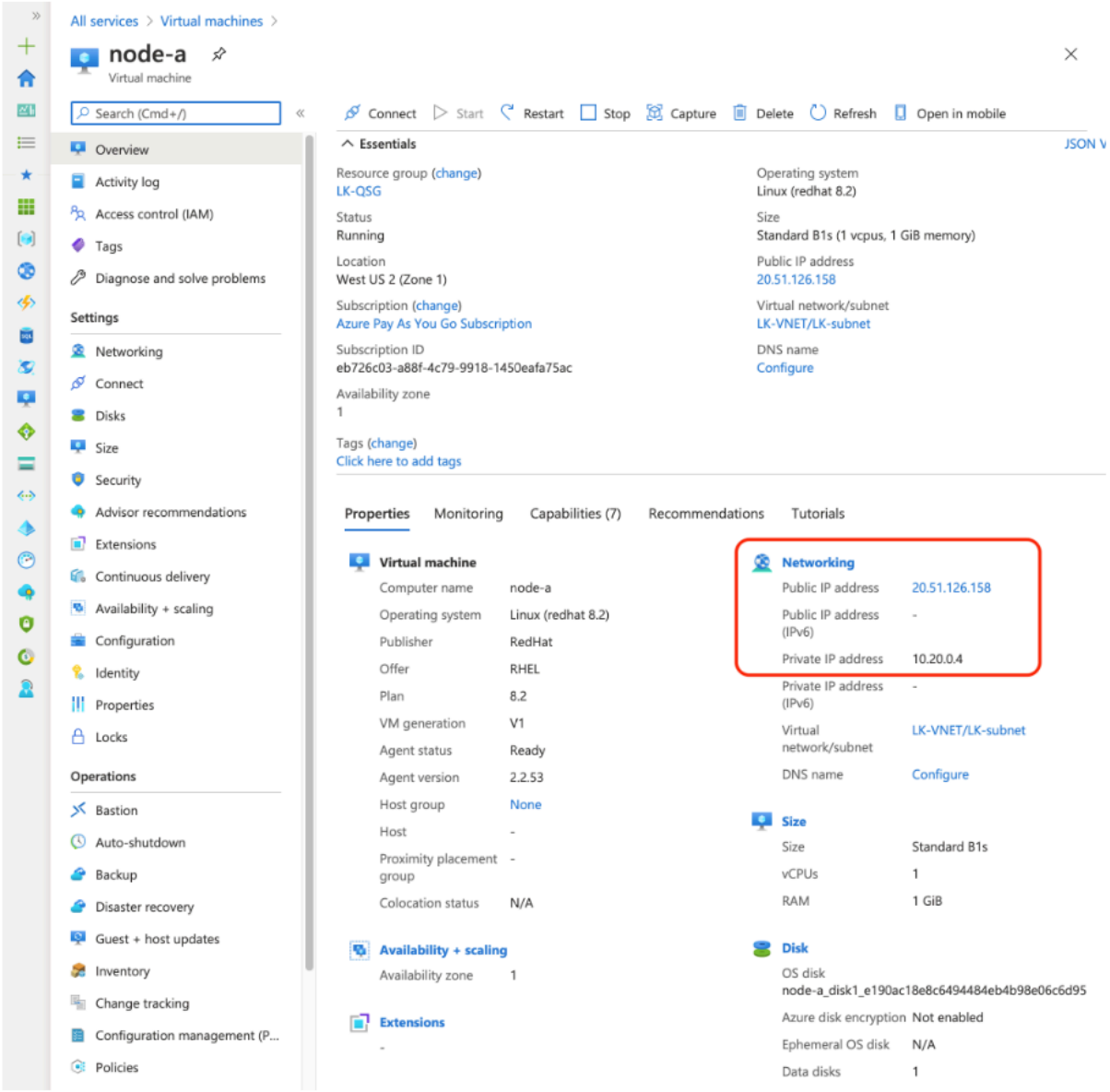



10. If this is the first time you are using the key pair, download the private key file. You will need the key file to connect to the instance later.

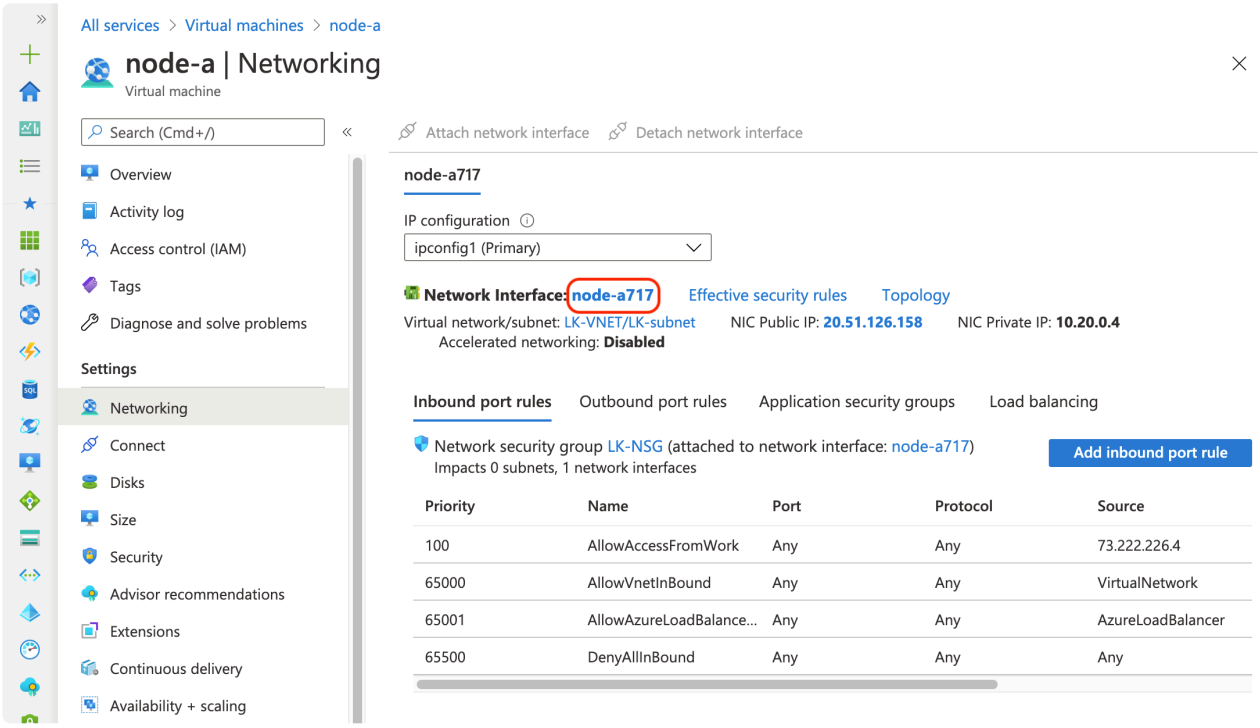



11. Once the virtual machine is created, you can see the configuration of the VM. You may notice that

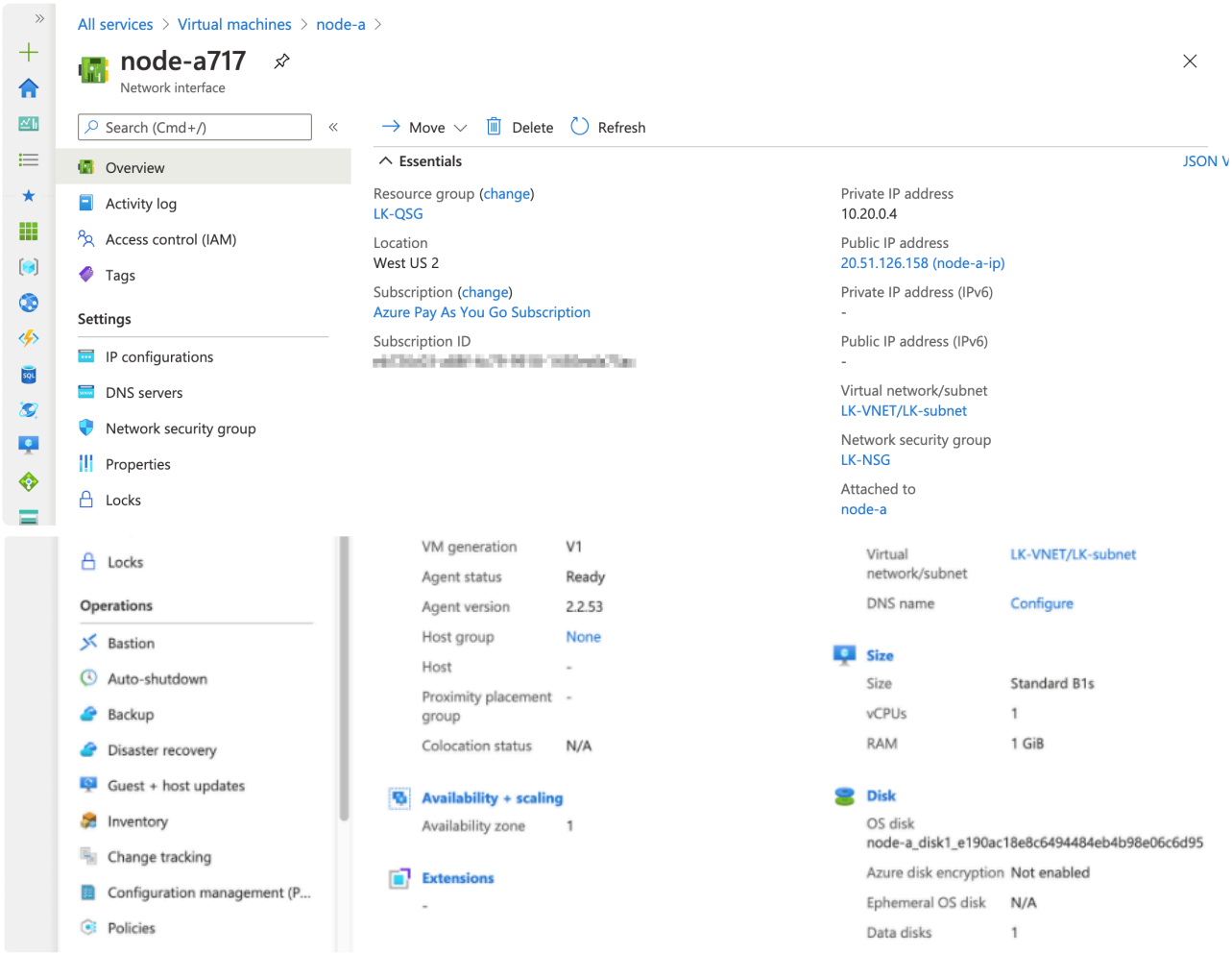
we haven't specified the IP address yet and the DHCP assigned address 10.20.0.4 was chosen automatically. Click "Networking" (the blue link) to change the network configuration.



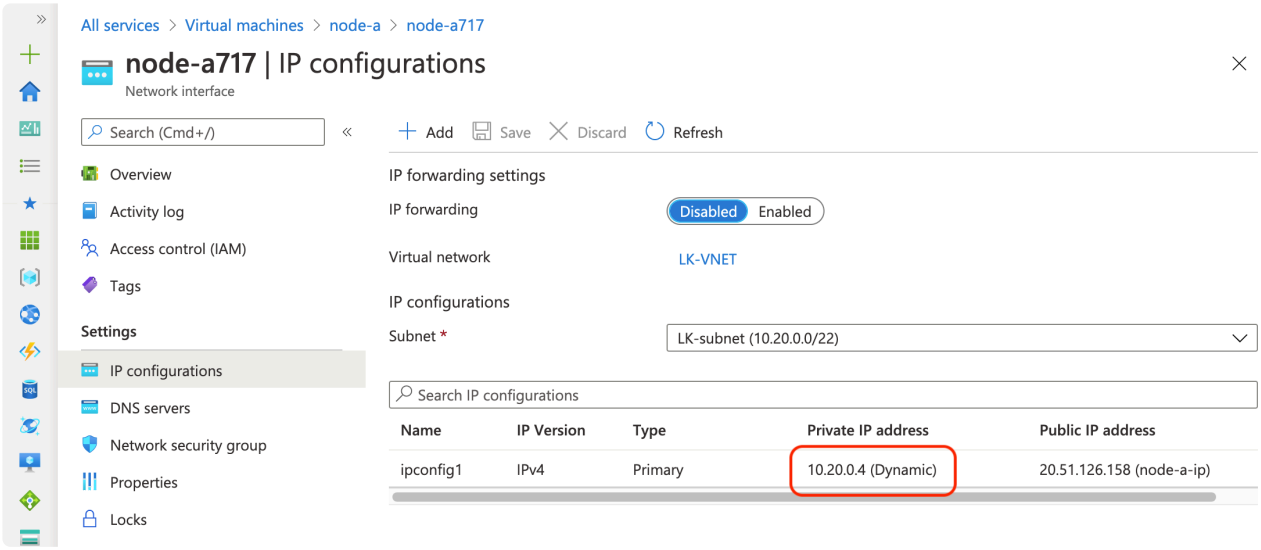

12. You can see in the screenshot below that the virtual machine has a network interface (node-a717). Click `node-a717` to change configuration of this network card.



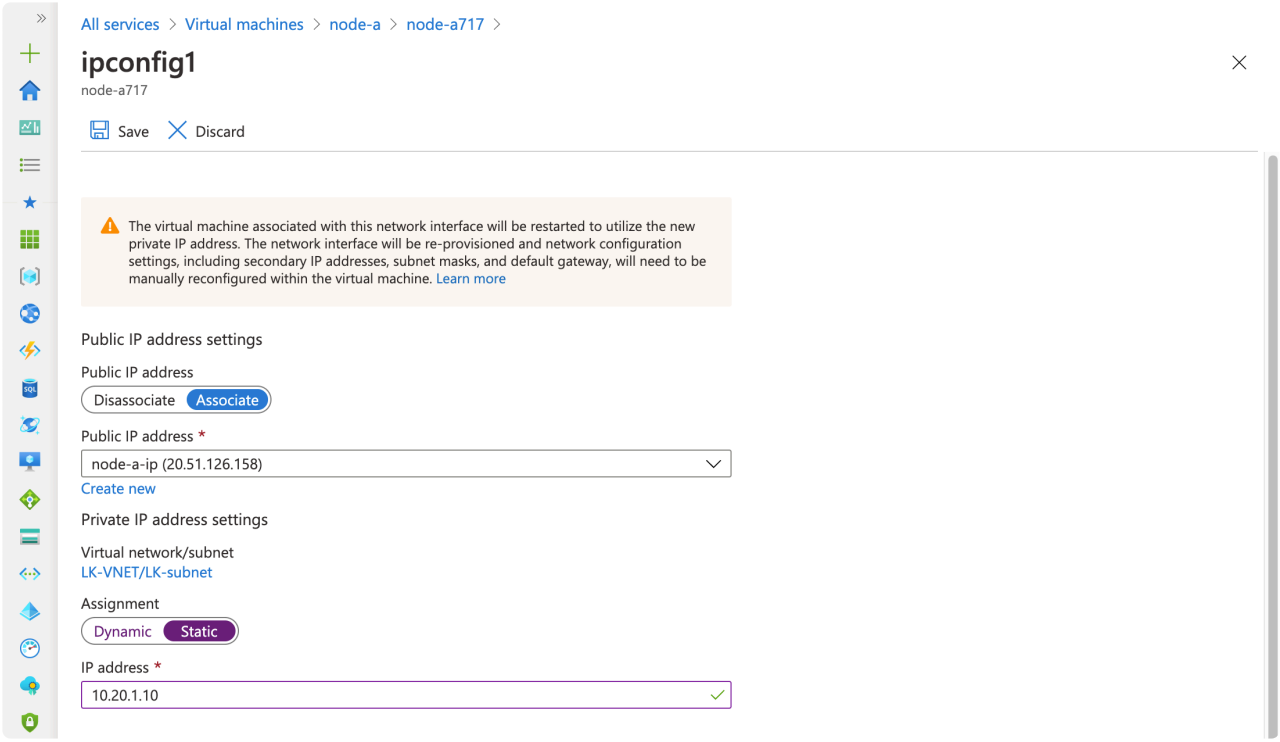

13. Now you can see the details of the network card. Click “IP configurations”.




14. Select the IP address to go to configuration page.




15. Change Assignment from `Dynamic` to `Static` and set the IP address to `10.20.1.10`.




Once you specify the IP address, click “Save”.

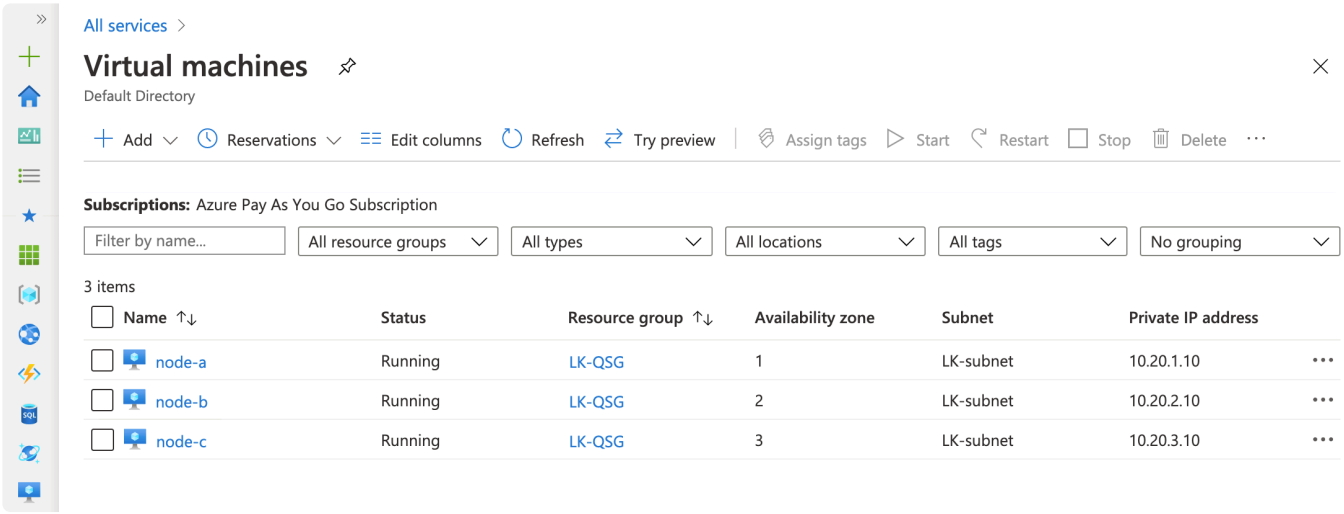
16. The configuration of the virtual machine is done.

# 11.2.3.4.6. Creating the Second and Third Virtual Machines

The steps required to create the second node (node-b) and third node (node-c) is almost the same as previously described for the first node (node-a). The following table illustrates the differences between the three nodes and the details required to create these instances.

| Name | Parameter | Value |
|---|---|---|
| **Common values across VM** | **Resource Group** | **LK-QSG** |
| | **Virtual network** | **LK-VNET** |
| | **Subnet** | **LK-subnet** |
| | **Network security Group** | **LK-NSG** |
| node-a | Availability Zone (*1) | 1 |
| | Private IP Address (*2) | `10.20.1.10` |
| | Second Disk (Storage Device) | You need to have a second disk |
| node-b | Availability Zone (*1) | 2 |
| | Private IP Address (*2) | `10.20.2.10` |
| | Second Disk (Storage Device) | You need to have a second disk |
| node-c | Availability Zone (*1) | 3 |
| | Private IP Address (*2) | `10.20.3.10` (node-c) |
| | Second Disk (Storage Device) | You do not need to create a second disk |

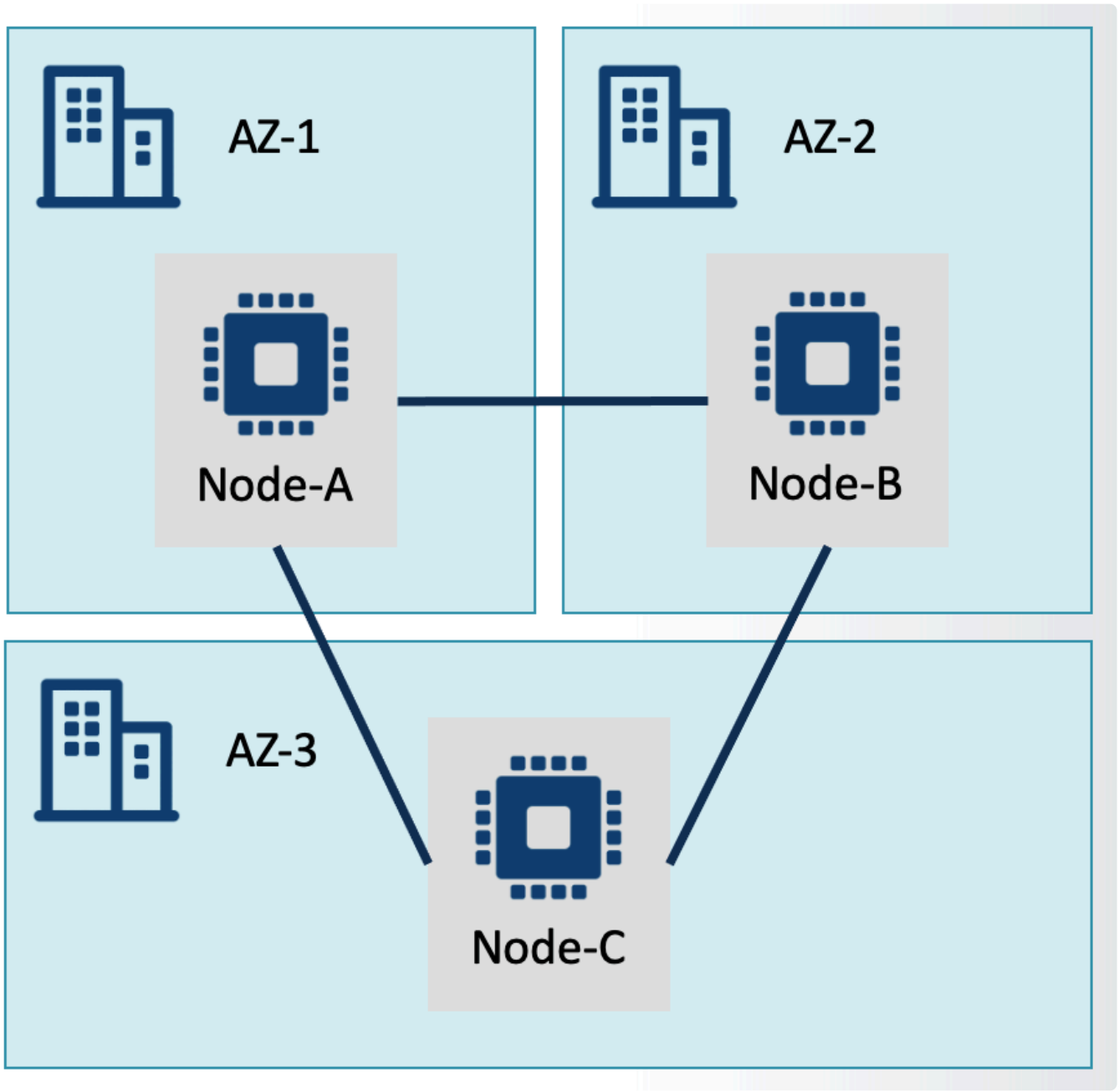
Once created, the list of instances should look like the following.


All services
Virtual machines
Default Directory
Add
Reservations
Edit columns
Refresh
Try preview
Assign tags
Start
Restart
Stop
Delete
Subscriptions: Azure Pay As You Go Subscription
Filter by name...
All resource groups
All types
All locations
All tags
No grouping
3 items
Name
Status
Resource group
Availability zone
Subnet
Private IP address
node-a Running LK-QSG 1 LK-subnet 10.20.1.10
node-b Running LK-QSG 2 LK-subnet 10.20.2.10
node-c Running LK-QSG 3 LK-subnet 10.20.3.10

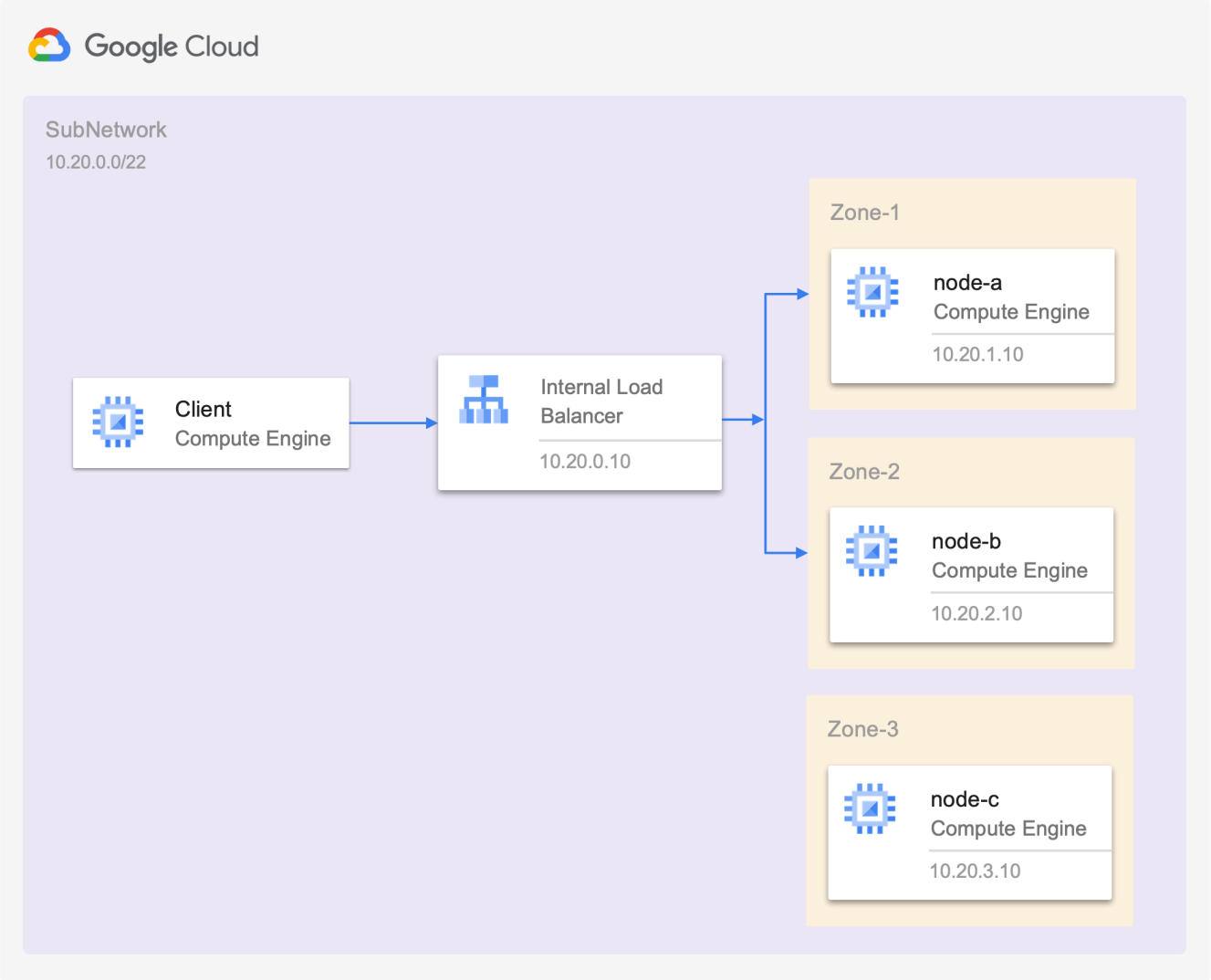
# 11.2.3.5. Creating an Instance in GCP from Scratch

**Disclaimer:** The user interface may vary between regions or change over time. Please refer to the documentation provided by GCP if the screenshots shown below are different from your experience.

This Evaluation Guide uses the following network structure and instances.




On GCP, these components can be defined as per the diagram below.




In this section we will create these components, with the exception of the Internal Load Balancer. We will create the Internal Load Balancer later.

The components shown in this diagram are described in the following table.

To create this network structure, the following components must be implemented:

| Component | Name | Parameter | Value |
|---|---|---|---|
| VPC network | lk-vpc | Region | us-west1 (*1) |
| | | Subnet (lk-subnet) | 10.20.0.0/22 |
| Virtual Machine | Common values across VM | **VPC network** | **lk-vpc** |
| | | **Subnet** | **lk-subnet** |
| | | **Network Tags** | **lk-node** |
| | | **External IP** | **Ephemeral** |
| | node-a | Zone | us-west1-a (*1) |

| | | | |
|---|---|---|---|
| | | Hostname | node-a.internal |
| | | Internal IP | 10.20.1.10 |
| | node-b | Zone | us-west1-b (*1) |
| | | Hostname | node-b.internal |
| | | Internal IP | 10.20.2.10 |
| | node-c | Zone | us-west1-c (*1) |
| | | Hostname | node-b.internal |
| | | Internal IP | 10.20.3.10 |
| Firewall Rules | Common values across Rules | **Network** | **lk-vpc** |
| | | **Type** | **Ingress** |
| | | **Target** | **lk-node** |
| | | **Action** | **Allow** |
| | fw-allow-ssh | Source | WAN IP Address of work location |
| | | Protocols / ports | all |
| | fw-allow-lk-node-connection | Source | Tags: lk-node |
| | | Protocols / ports | all |
| | fw-allow-health-check (*2) | Source | 130.211.0.0/22, 35.191.0.0/16 |
| | | Protocols / ports | all |

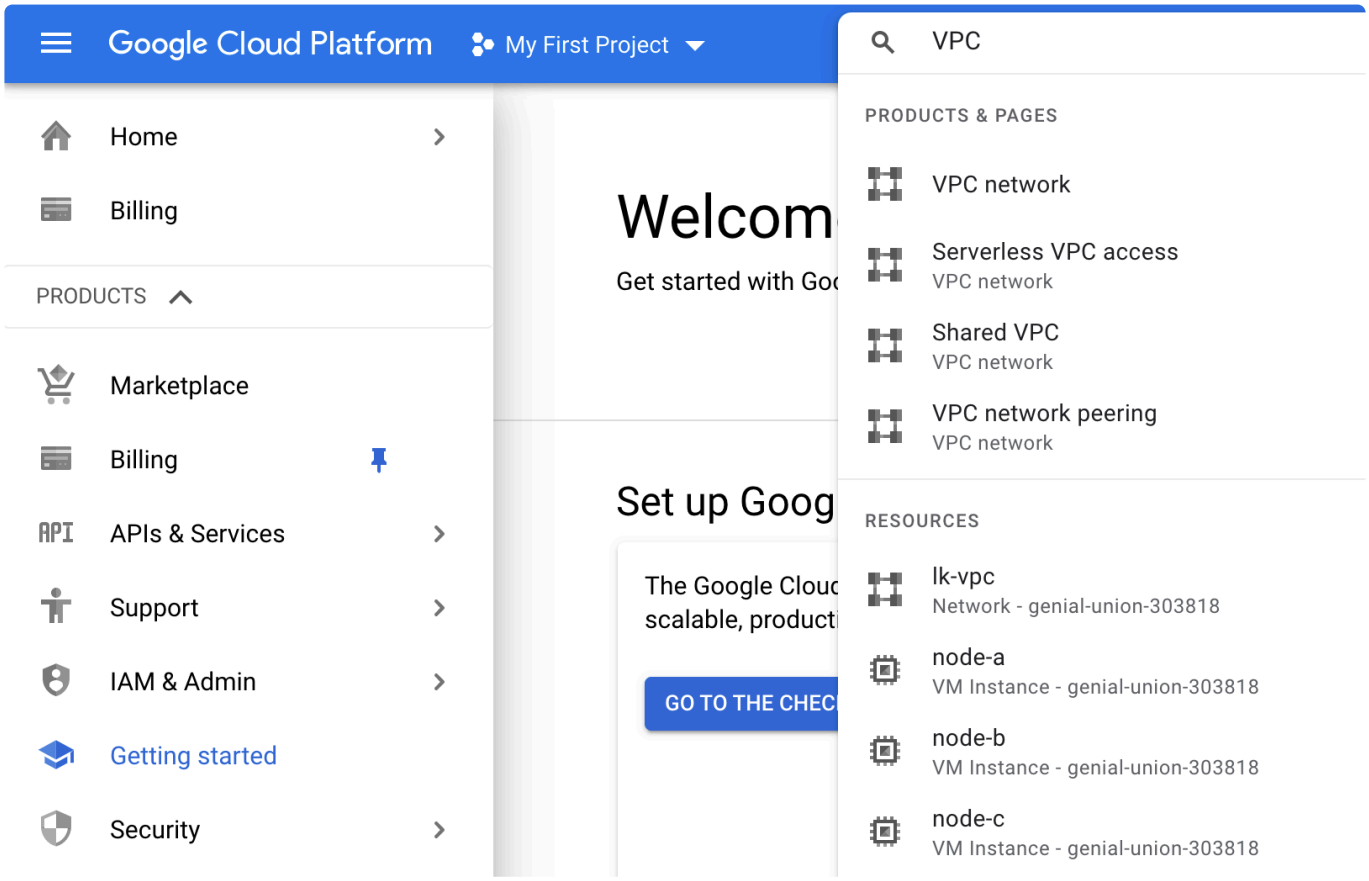
- (*1) This example uses us-west1 as Region.
- (*2) This is required to allow the Internal Load Balancer to check the status of each nodes.

The following sections step through the processes necessary for creating these network components, beginning with the first instance.

- Switching between GCP Services
- Deciding on a GCP Region
- Creating the Project
- Creating a VPC Network
- Creating a New SSH Key
- Creating the First GCP VM
- Configuring the Firewall Rules
- Creating the Second and Third VM

# 11.2.3.5.1. Switching between GCP Services

GCP has a significant and growing number of services available. To move to a different service you may want to type the name of the service in the search box at the top of the GCP console. Select the service from the list. The screenshot below shows how to select the VPC Network service.

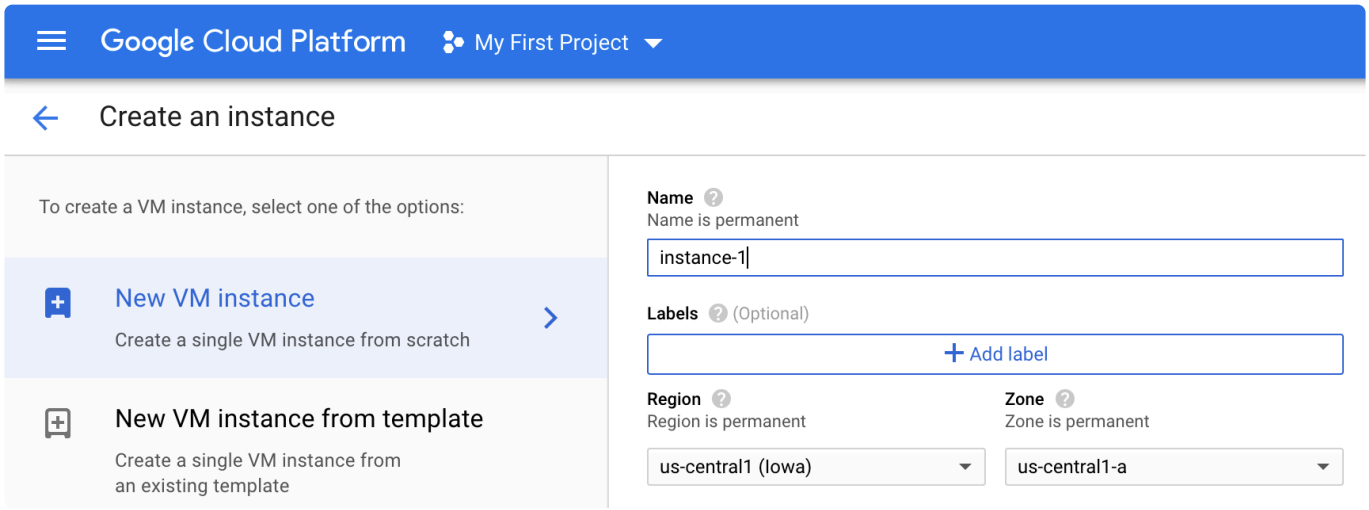


# 11.2.3.5.2. Deciding on a GCP Region

GCP has regions in many geographic locations and it may be beneficial to select a region in close geographic proximity to the workplace location. Each region has three or more Zones, but available machine types are slightly different between regions.

Refer to the GCP documentation for more information.

When you create a VM, select the Region and Zone. These cannot be changed once you create a VM.



# 11.2.3.5.3. Creating the Project

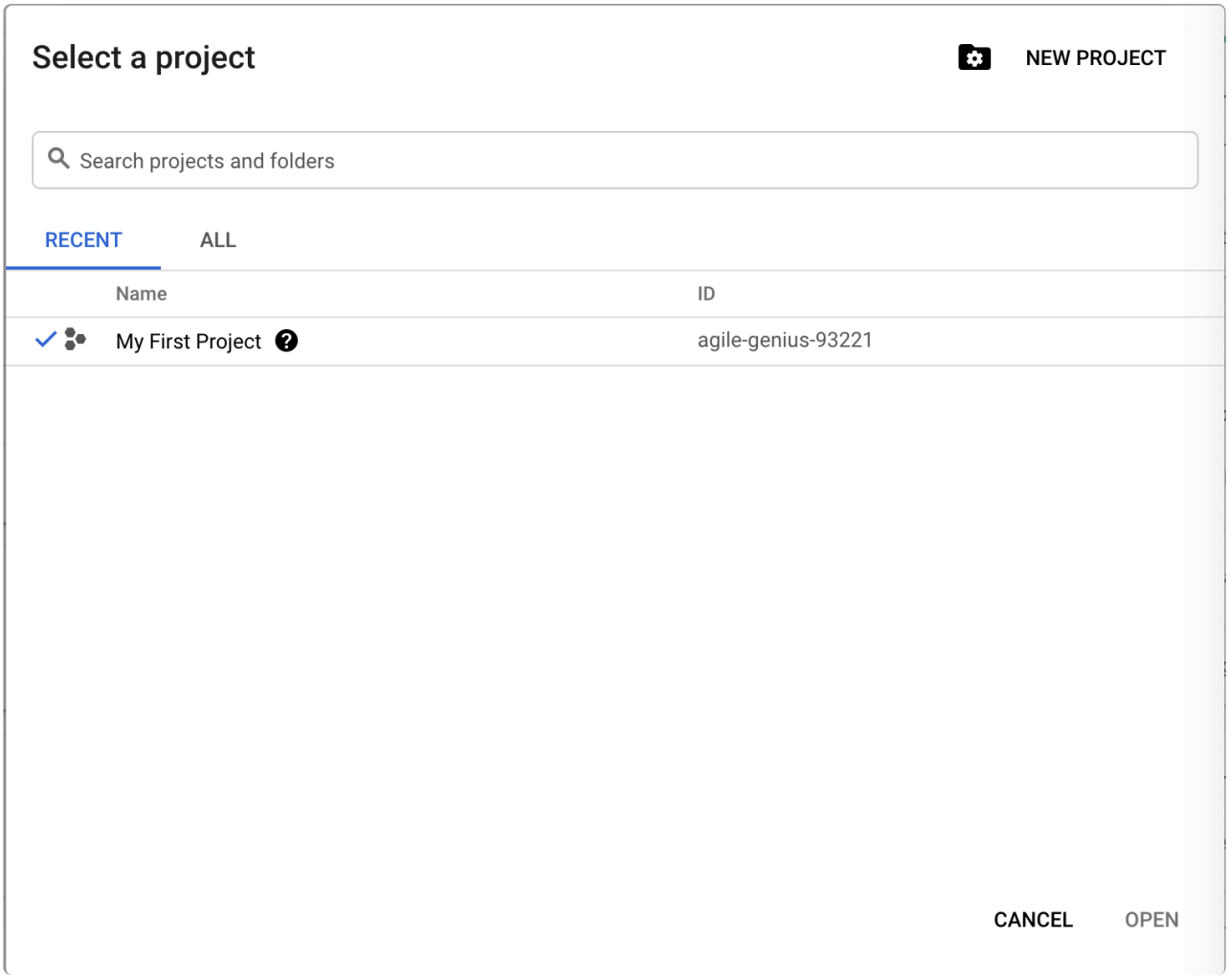
A Project is the logical container in which you can create resources such as network components, computer resources, storage, etc.

Refer to the GCP documentation for more information.

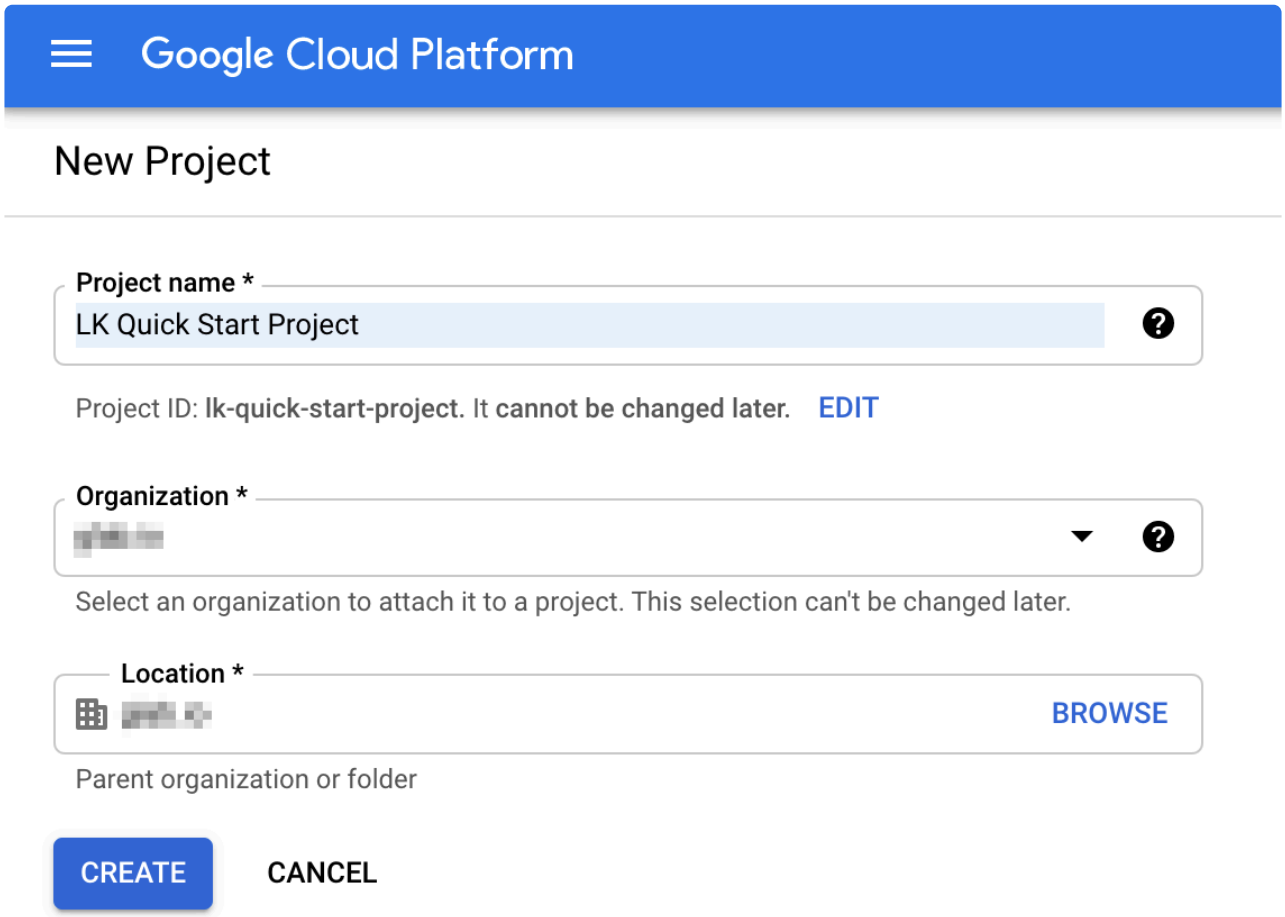
If you have just signed up to GCP, you have a default project called “My First Project”. You can use this project. However, you can also create a new project (we will name it LK Quick Start Project) as follows:

1. Select “My First Project” (or any other active project you are working on) located at the top.

2. On the “Select a project” screen, click “NEW PROJECT” located at the top right side.

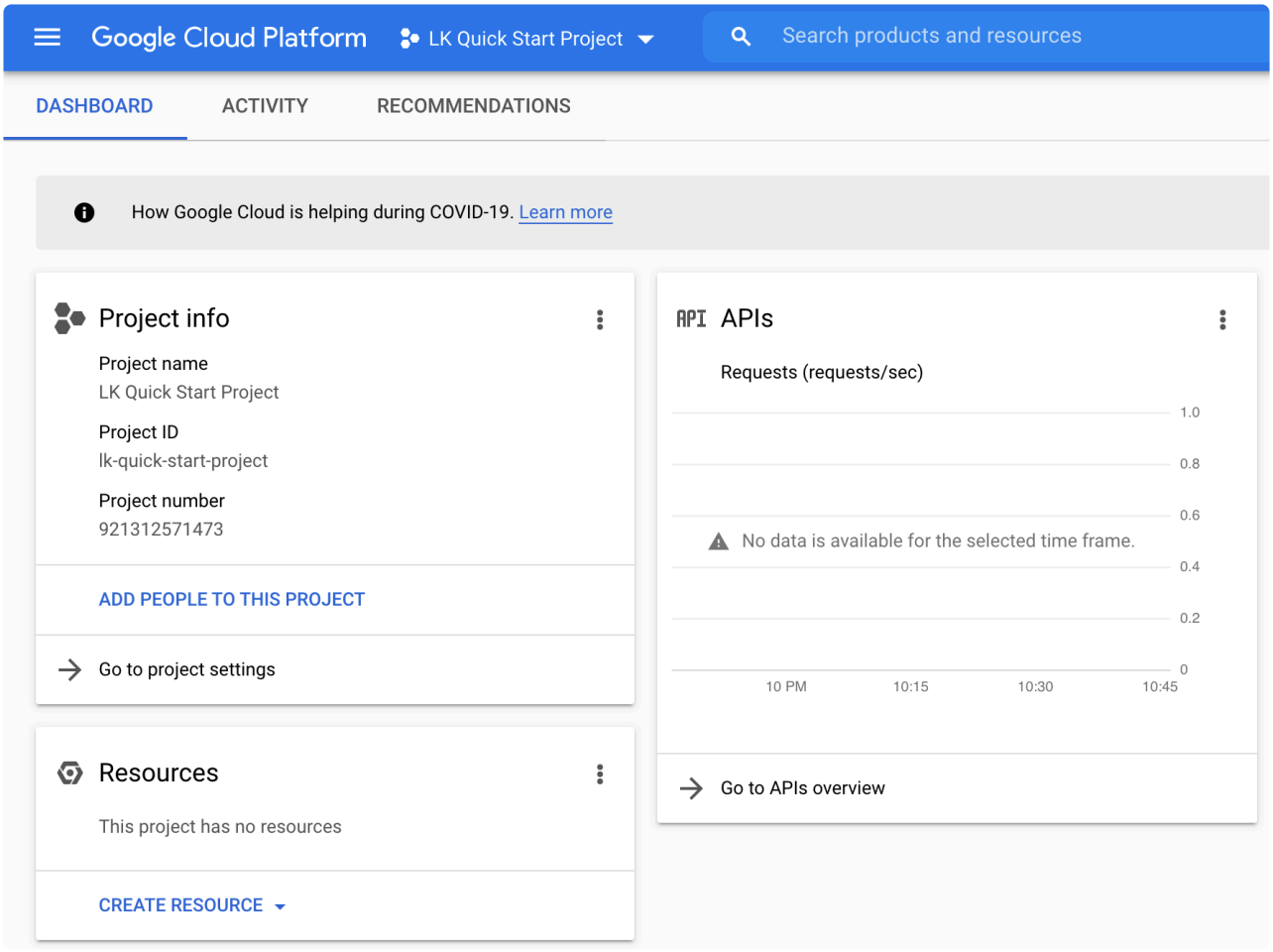


3. Specify the name of the Project as `LK Quick Start Project`.




Once you fill in the fields, click “Create”.

4. Now the new Project is created.


Google Cloud Platform
LK Quick Start Project
Search products and resources
DASHBOARD
ACTIVITY
RECOMMENDATIONS
How Google Cloud is helping during COVID-19. Learn more
Project info
Project name
LK Quick Start Project
Project ID
lk-quick-start-project
Project number
921312571473
ADD PEOPLE TO THIS PROJECT
Go to project settings
Resources
This project has no resources
CREATE RESOURCE
APIs
Requests (requests/sec)
1.0
0.8
0.6
No data is available for the selected time frame.
0.4
0.2
0
10 PM
10:15
10:30
10:45
Go to APIs overview

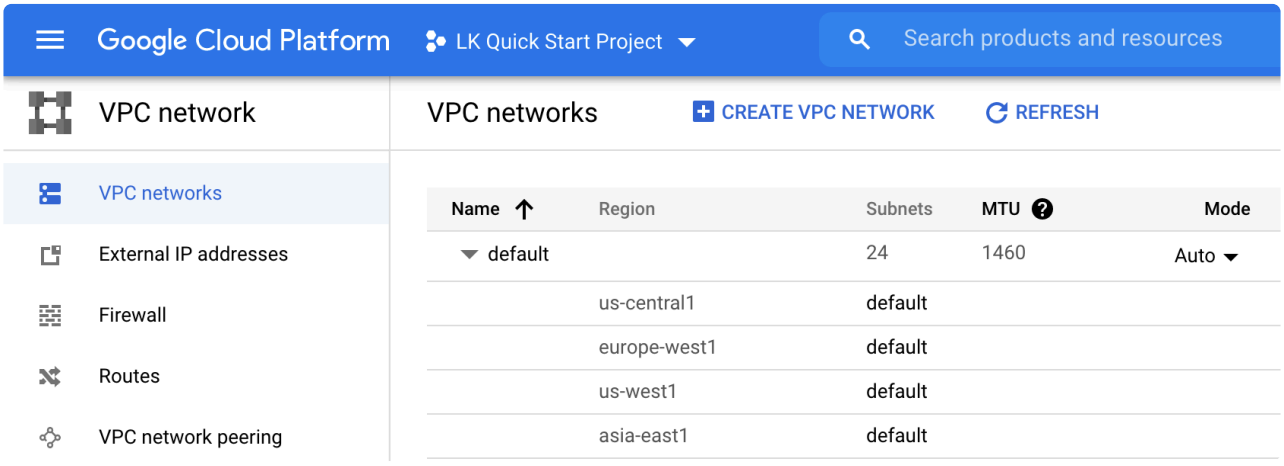
# 11.2.3.5.4. Creating a VPC Network

A VPC network is a GCP resource that represents a local network. Different VPC networks can be defined within the GCP to logically separate different systems.

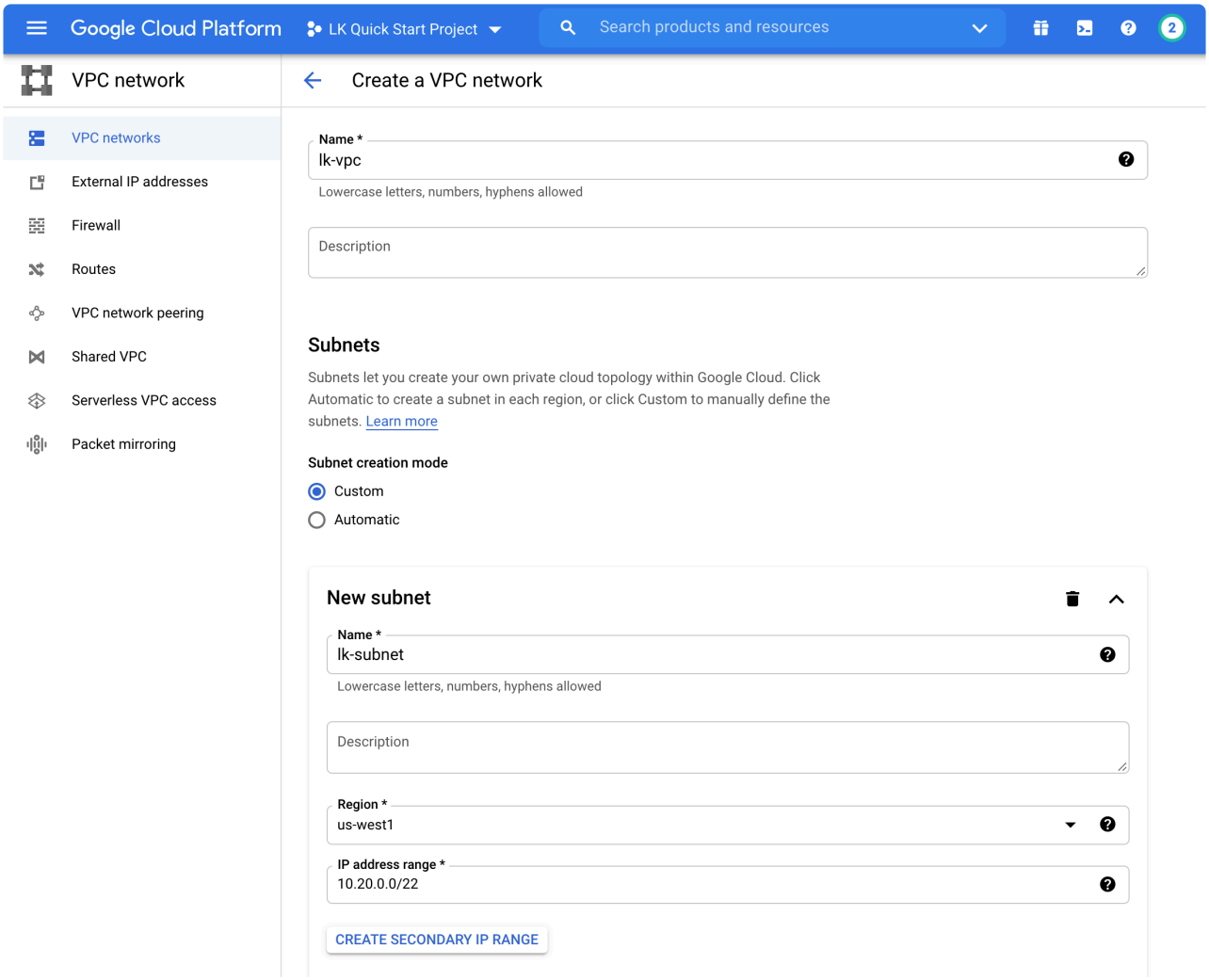
Refer to the GCP documentation for more information.

In this section we will create a VPC network for testing LifeKeeper (we will name it `lk-vpc`) as follows:

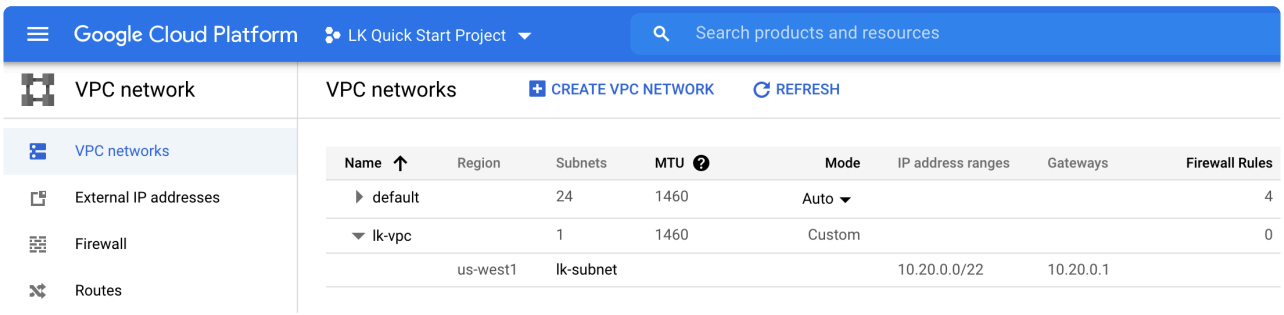
1. Select “VPC network” to see the current list of VPC networks. There is a “default” VPC network and its subnets for each region defined. Click “CREATE VPC NETWORK” located at the top.




2. In the “Create a VPC network” wizard, set the name of the VPC network as `lk-vpc`, the region to create the VPC network in, the name of the subnet, and the CIDR block `10.20.0.0/22`.




3. Once you review the values you entered, click “CREATE” at the bottom. Now the new VPC network is created.

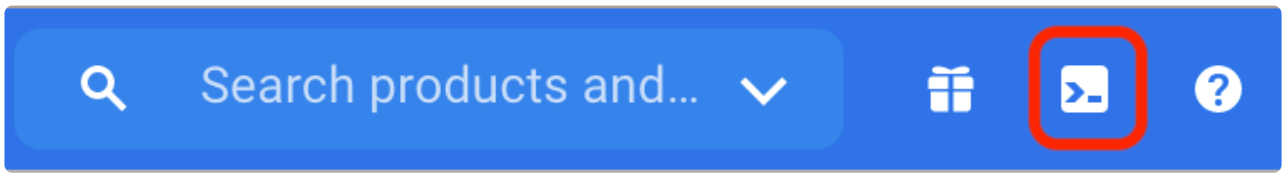


# 11.2.3.5.5. Creating a New SSH Key

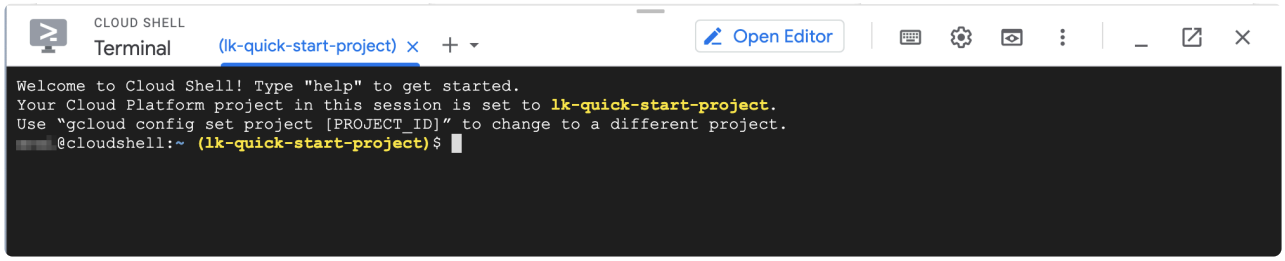
You will need to use an ssh key to connect to VMs. This section outlines how you can create a new ssh key on the GCP console and then download the private key to your local system.

Refer to the GCP documentation for more information.

1. Activate the Cloud Shell by clicking [>_] at the top of the console.



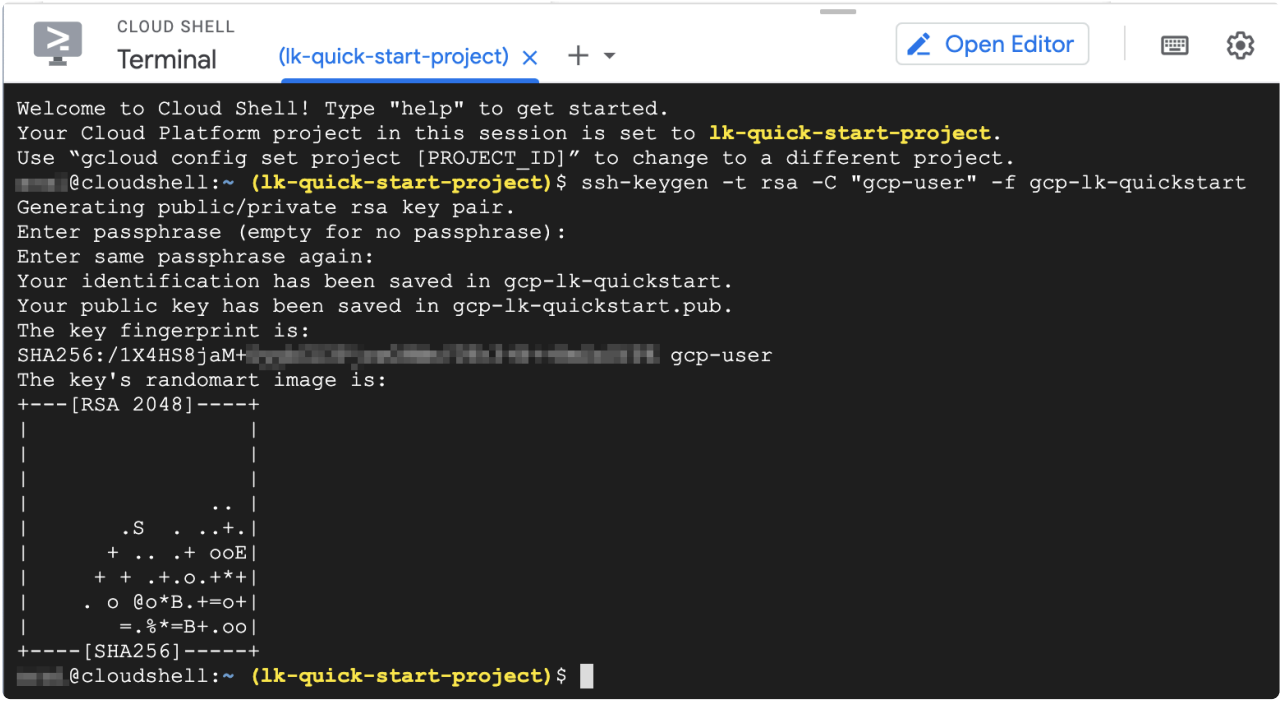

2. Once you activate the Cloud Shell you will see the shell screen at the bottom of the browser.




3. Create a new ssh key by using the `ssh-keygen` tool.

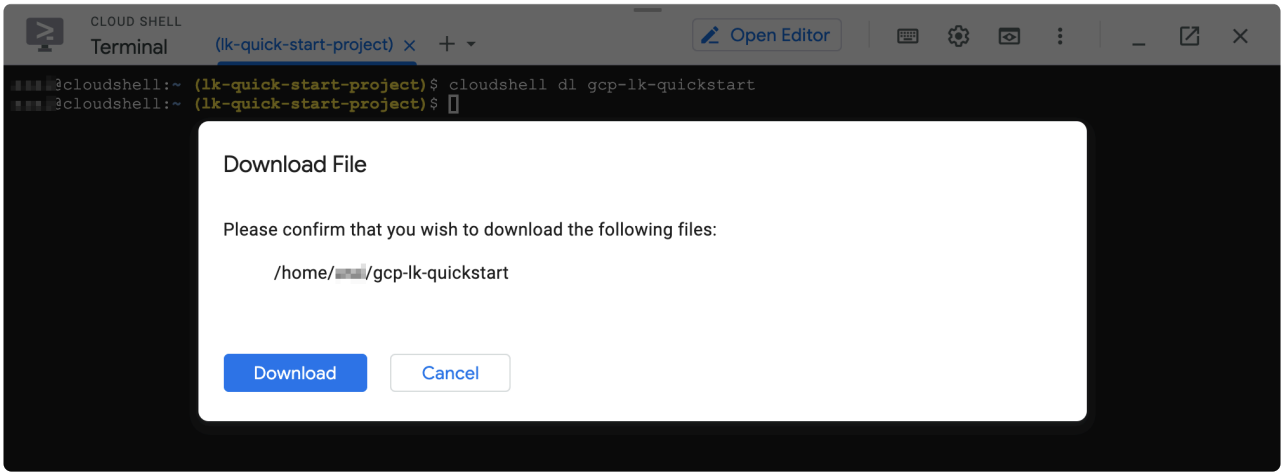
```
1 ssh-keygen -t rsa -C "gcp-user" -f gcp-lk-quickstart
```

4. Now a new key `gcp-lk-quickstart` is generated with a user name of `gcp-user`.




5. In a later step you will need to register a public key `gcp-lk-quickstart.pub` to a new VM. Type `cat gcp-lk-quickstart.pub` to copy the public key.

6. Download the private key with `cloudshell dl gcp-lk-quickstart`.

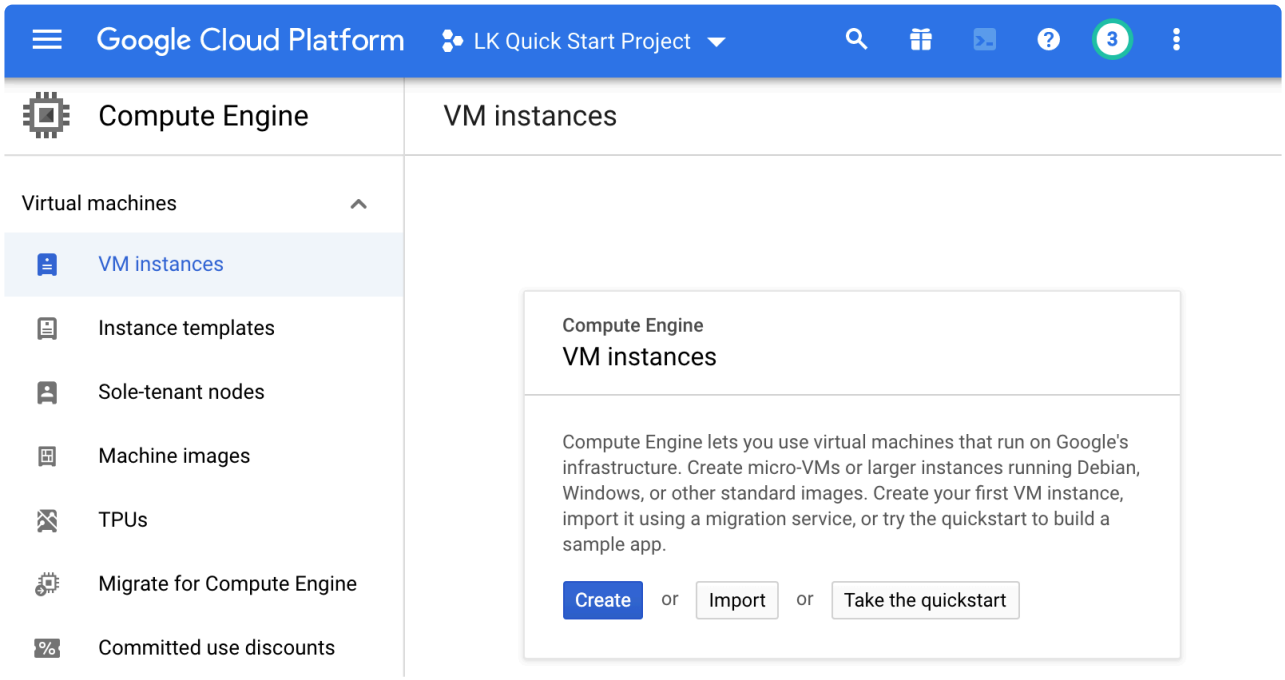



You will need this private key to connect to the VM in a later step.

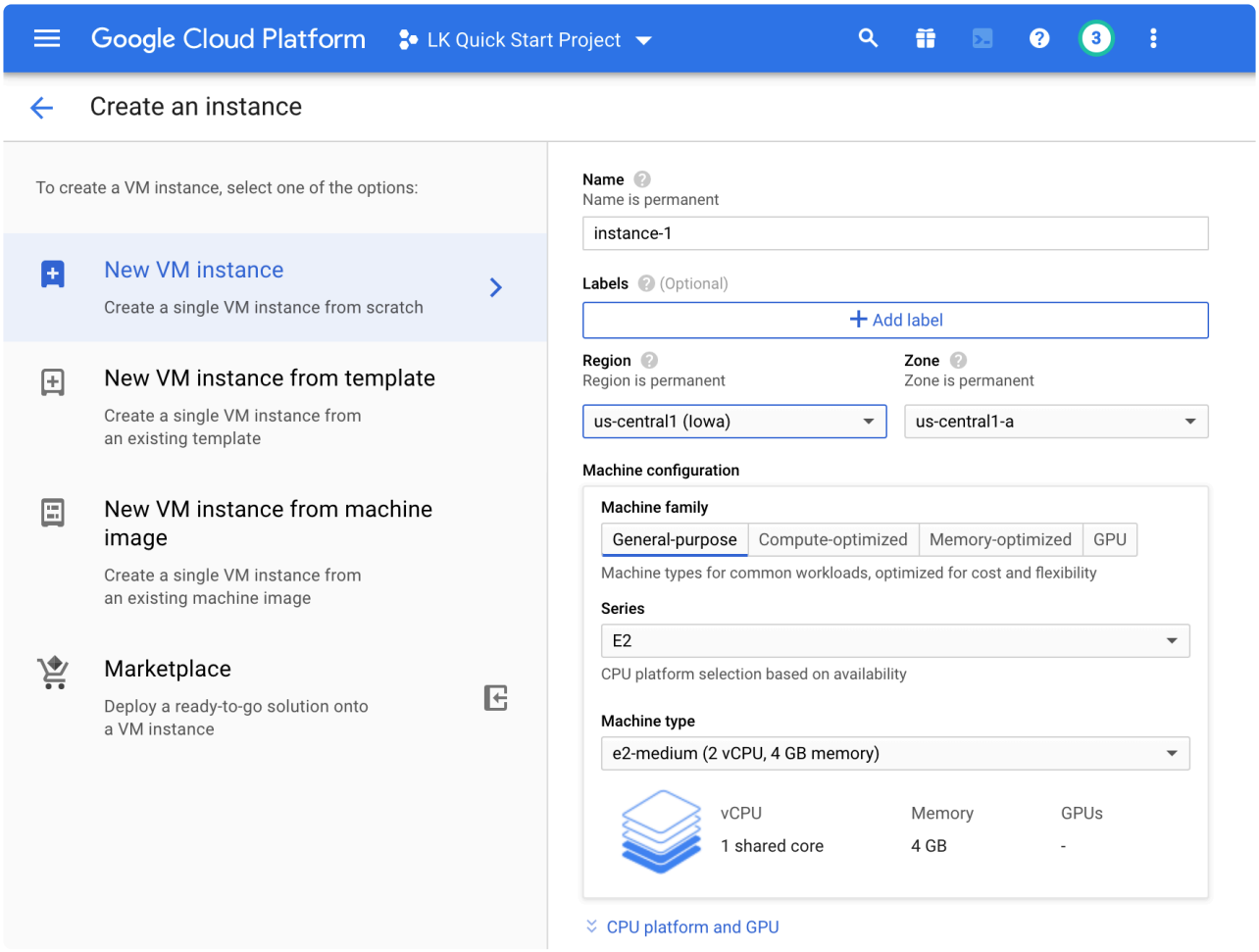
# 11.2.3.5.6. Creating the First GCP VM

In previous sections we covered the configuration of the network. Now we are going to create the first VM. As discussed in Computing Resources Used in this Tutorial, we need two disks. This section also discusses how to create the second disk.

1. Go to “Compute Engine” > “VM Instance” and click “Create” to create a new VM.

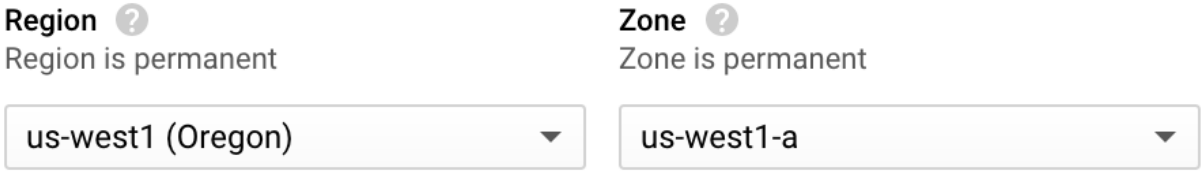



2. The “Create an instance” wizard appears.

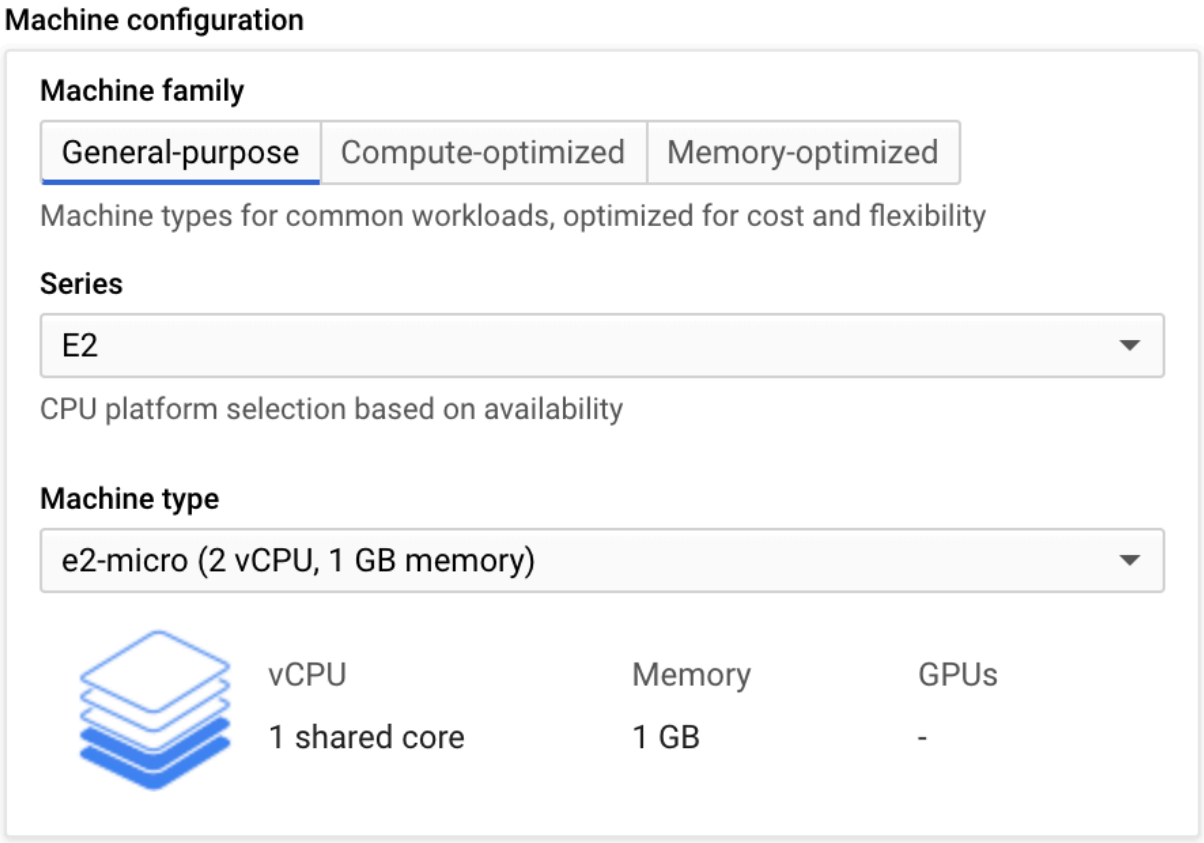



3. Enter the name of the new VM: node-a (when you create 2nd or 3rd nodes, this would be node-b and node-c).

4. Select the Region and Zone of the VM. Enter a nearby region, then select the Zone. Select the zone ending with `-a` for node-a, `-b` for node-b and `-c` for node-c. Note that the zones in some regions may be named differently.

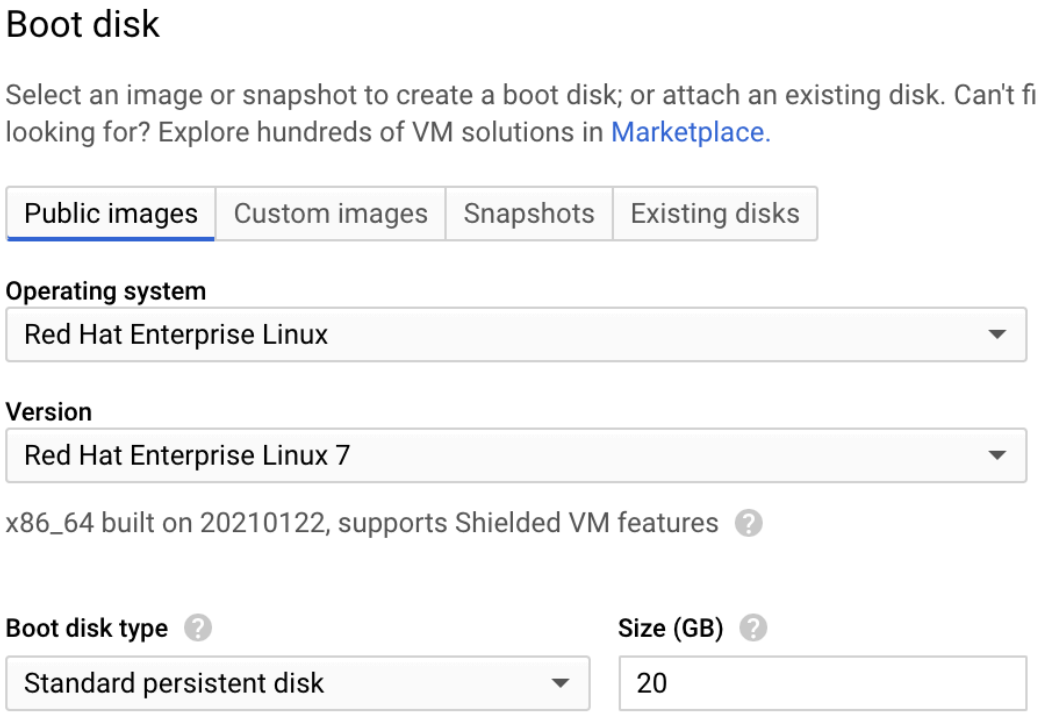



5. Select the machine configuration. Select the required CPU and Memory requirements based on the application you are protecting.

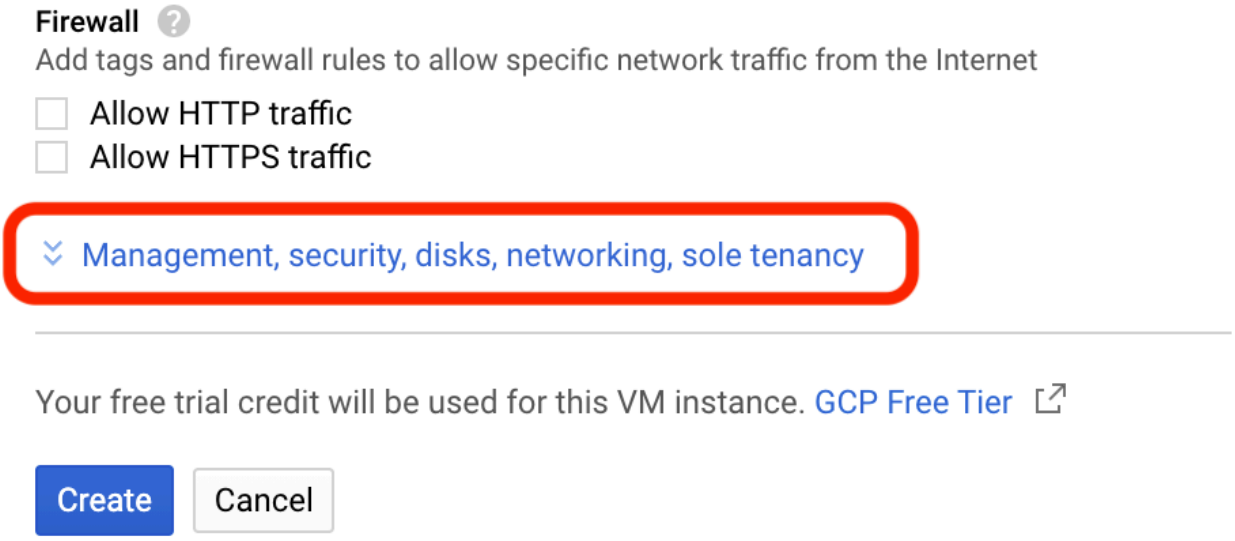



6. Select an operating system. By default, Debian GNU/Linux 10 is selected. Select an operating system that is supported by both SIOS Protection Suite as well as the database and/or the application you are planning to use. Click “Change” and select Red Hat Enterprise Linux 7.

> ! If planning to create an SAP or SAP HANA cluster, select the “Red Hat Enterprise Linux 8.2 for SAP Applications” boot image in this step.

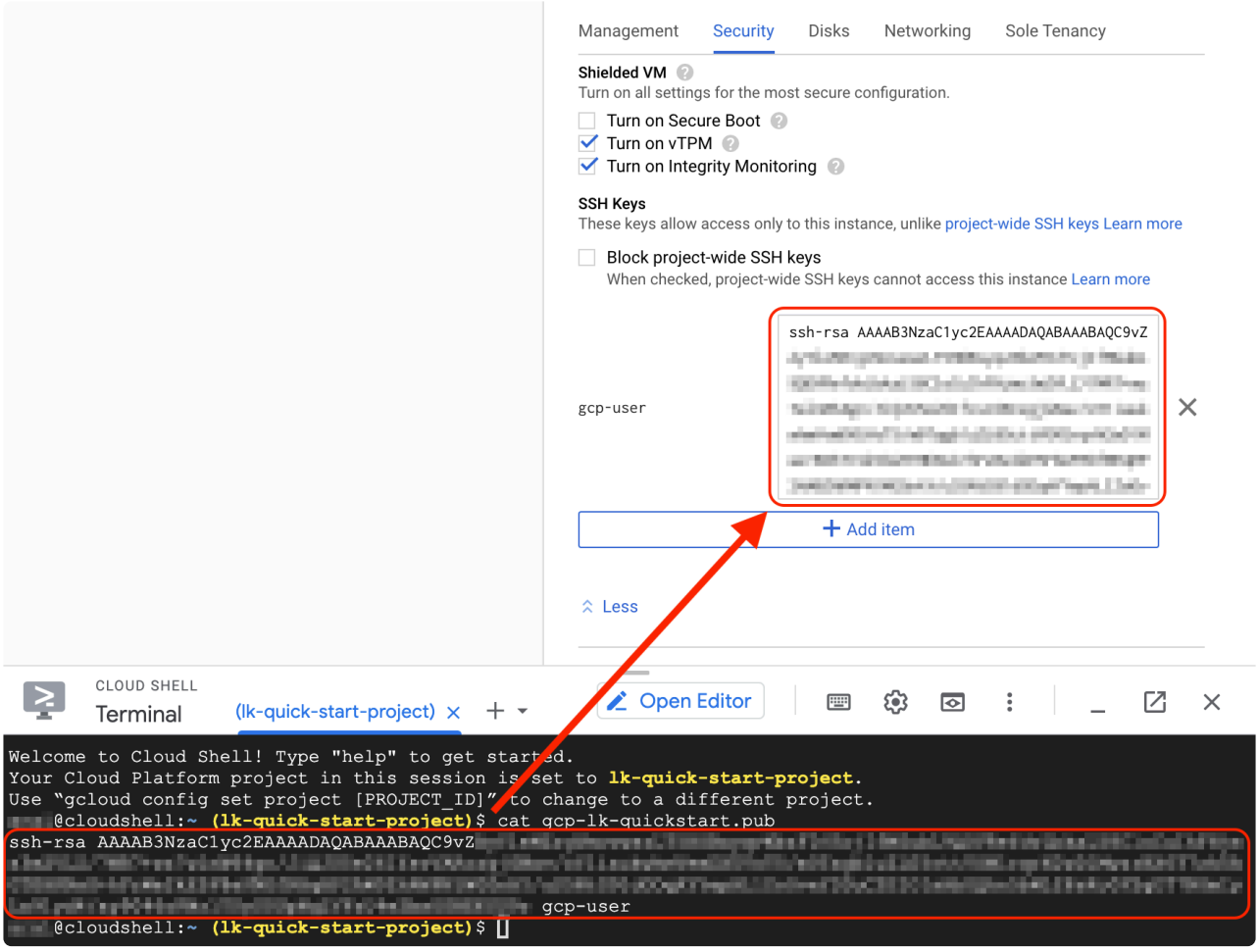



7. Before selecting “Create”, expand the drop down list for “Management, security, disks, networking, sole tenancy”.

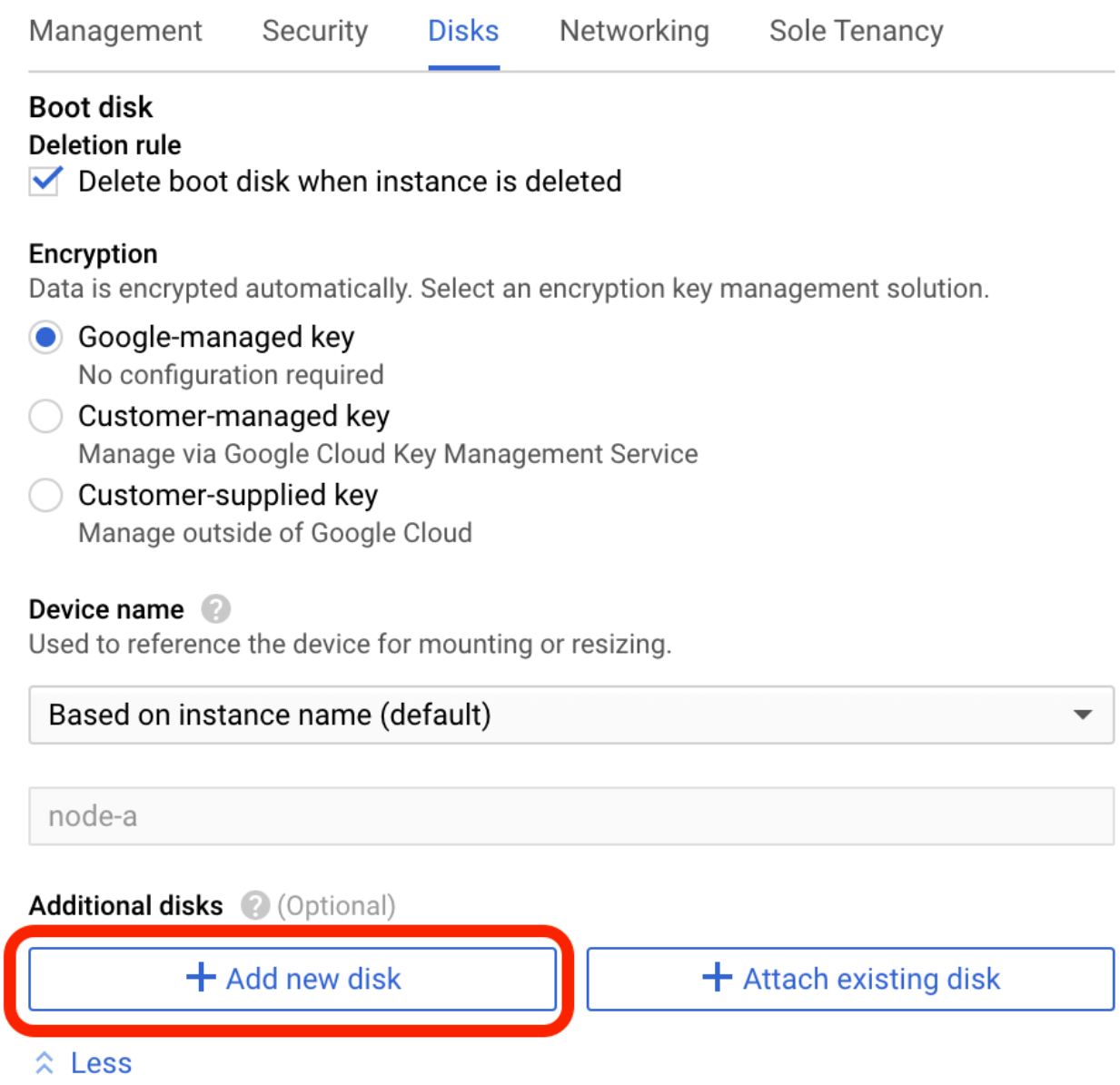



8. Go to the Security Tab and enter the public key that was downloaded in the Creating a New SSH Key section.

Make sure there is no whitespace if you copy and paste the public key.

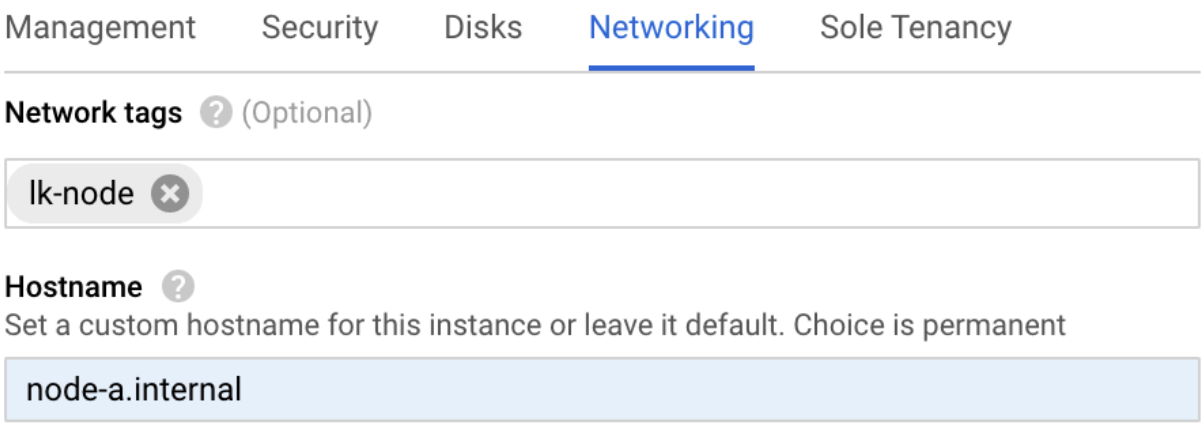



9. Go to the “Disks” tab. We need an additional disk for data replication on node-a and node-b. To add the nodes, click “+ Add new disk“.

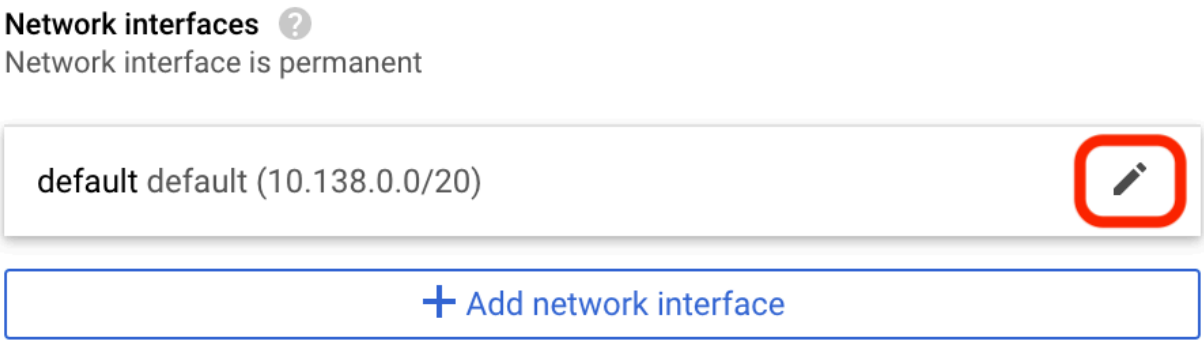



10. Specify the “Name” and the “Size” of the disk.

11. Go to the “Networking” tab. Specify “Network tags” and “Hostname”. The “Network tags” are used to identify the VM in the Firewall configuration (discussed in the next step).

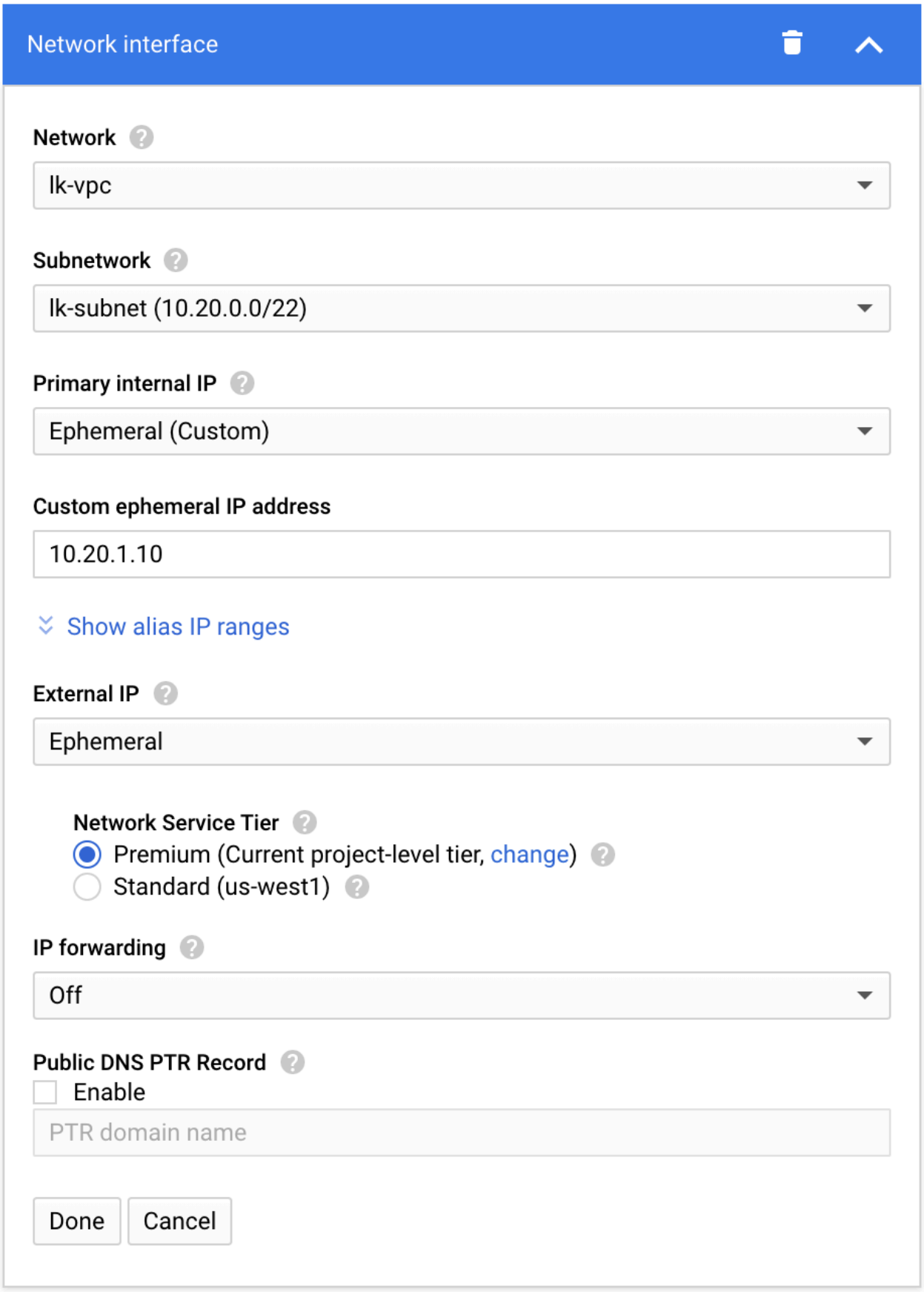



12. Edit the network interface. Select the “Edit” icon.




13. Enter the parameters of the network interface as follows:

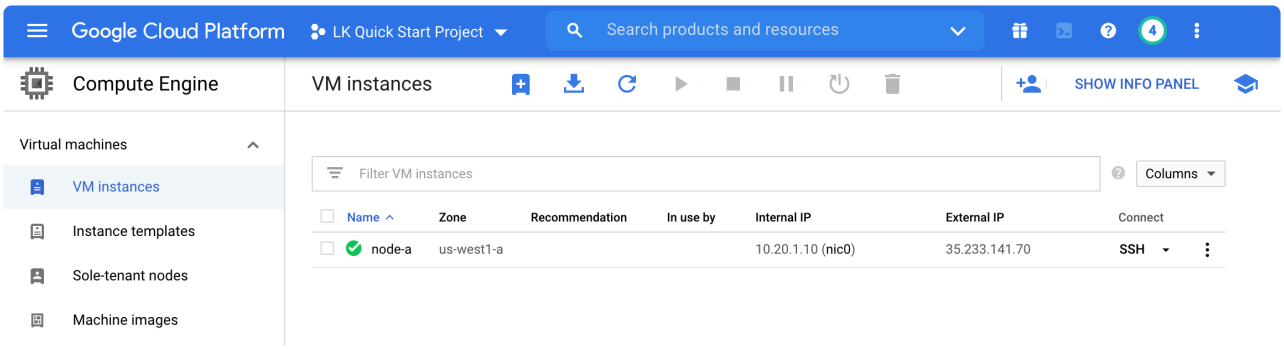
- Network: `lk-vpc`
- Subnetwork: `lk-subnet (10.20.0.0/22)`
- Primary internal IP: Ephemeral (Custom)
- Custom ephemeral IP address: `10.20.1.10` for node-a, `10.20.2.10` for node-b, `10.20.3.10` for node-c.
- External IP: Ephemeral
- IP forwarding: Off




Click “Done” once you enter the values.

Confirm the values you entered, then click “Create” at the bottom.

14. Congratulations! You have completed the configuration of the virtual machine.



# 11.2.3.5.7. Configuring the Firewall Rules

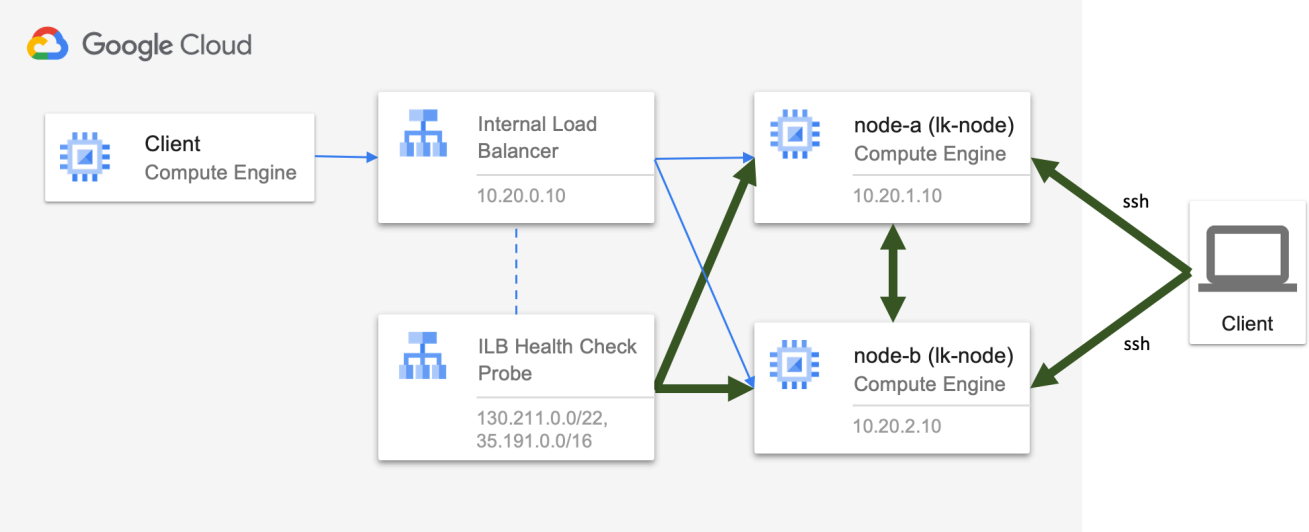
With a set of firewall rules, you can define both “allow” and “deny” rules for each type of traffic. A source can be an internet address, an Internal Load Balancer, or a group of VMs. A group of VMs is identified by a tag assigned to a VM.

Refer to the GCP documentation for more information.

Once you go to “VPC network” > “Firewall”, a set of firewall rules are already defined for the “default” VPC. Now we are going to define the following rules for `lk-vpc`.

- Allow ssh traffic from your remote work location to VMs with `lk-node` tag. (`fw-allow-ssh`)
- Allow all traffic between VMs with `lk-node` tag. (`fw-allow-lk-node-connection`)
- Allow all health check probes from the Internal Load Balancer to VMs with `lk-node` tag. (`fw-allow-health-check`)

The traffic allowed by these firewall rules is highlighted by the thick arrows in the diagram below.



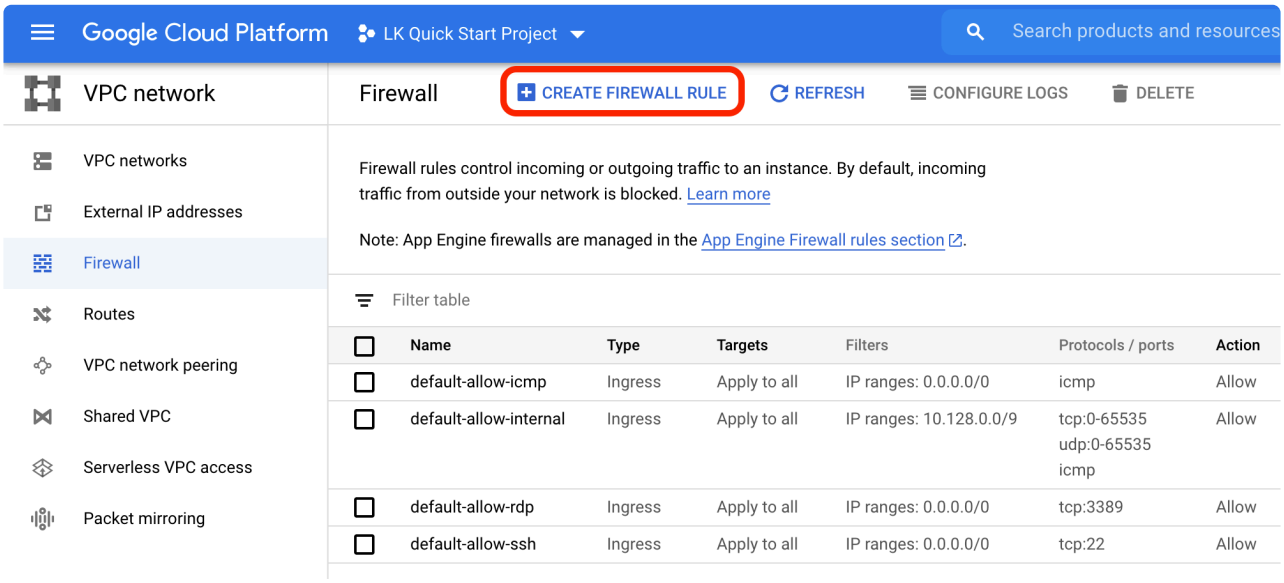

The following table outlines how we should configure these rules.

| Name | Parameter | Value |
|---|---|---|
| Common values across VM | **Network** | **lk-vpc** |
| | **Type** | **Ingress** |
| | **Target** | **lk-node** |
| | **Action** | **Allow** |
| fw-allow-ssh | Priority | 1000 |

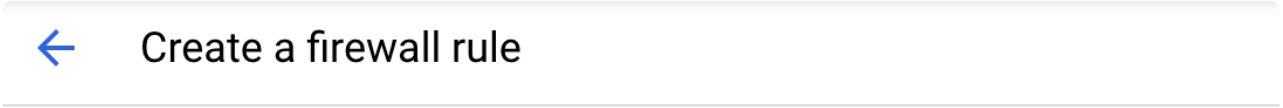
| | | |
|---|---|---|
| | Source | WAN IP Address of work location |
| | Protocols / ports | tcp:22 |
| fw-allow-lk-node-connection | Priority | 1100 |
| | Source | Tags: lk-node |
| | Protocols / ports | all |
| fw-allow-health-check (*1) | Priority | 1200 |
| | Source | 130.211.0.0/22, 35.191.0.0/16 |
| | Protocols / ports | all |

(*1) GCP Documentation on Load Balancer Health Check Probe

1. Select “VPC network” > “Firewall” from the home screen. Now you see a list of firewall rules for the default vpc. Click “CREATE FIREWALL RULE” located at the top and create the first rule “fw-allow-ssh“.



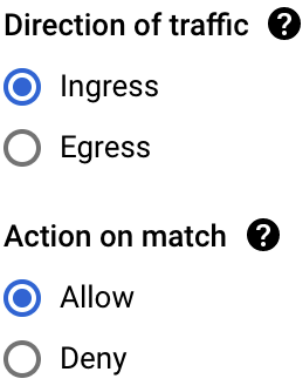

2. The following steps describe how to allow an ssh connection from your work location so that you will be able to configure the node from your work location. In the “Create a firewall rule” wizard, specify the name as `fw-allow-ssh`.




3. Select Network as lk-vpc.

4. Priority can be any value between 0 and 65535. Leave the default value of `1000`. For other rules, you can use 1100, 1200, etc. This will allow for other priority values to be injected if additional rules are needed in the future.

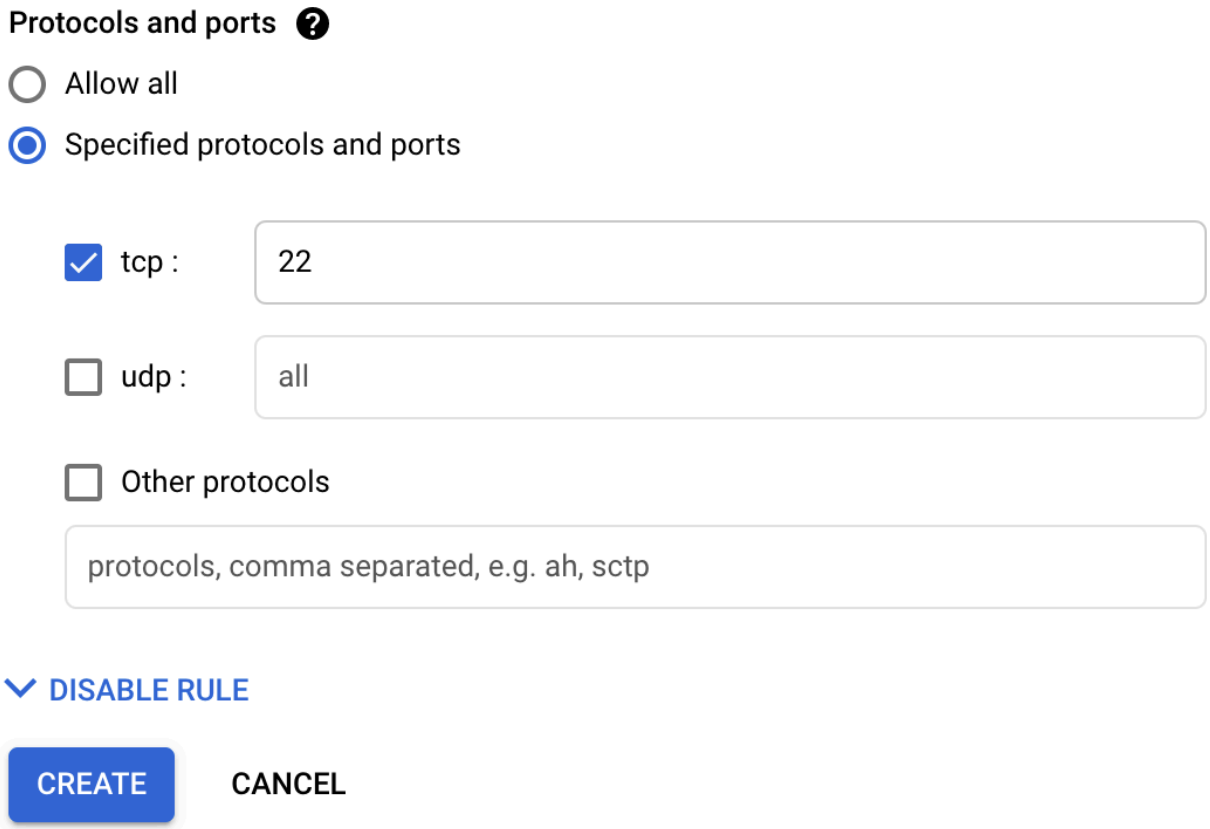
5. Select “Ingress” for “Direction of traffic” and “Allow” for “Action on match”. With these selections incoming traffic matched with patterns specified in the following steps are “Allowed”.



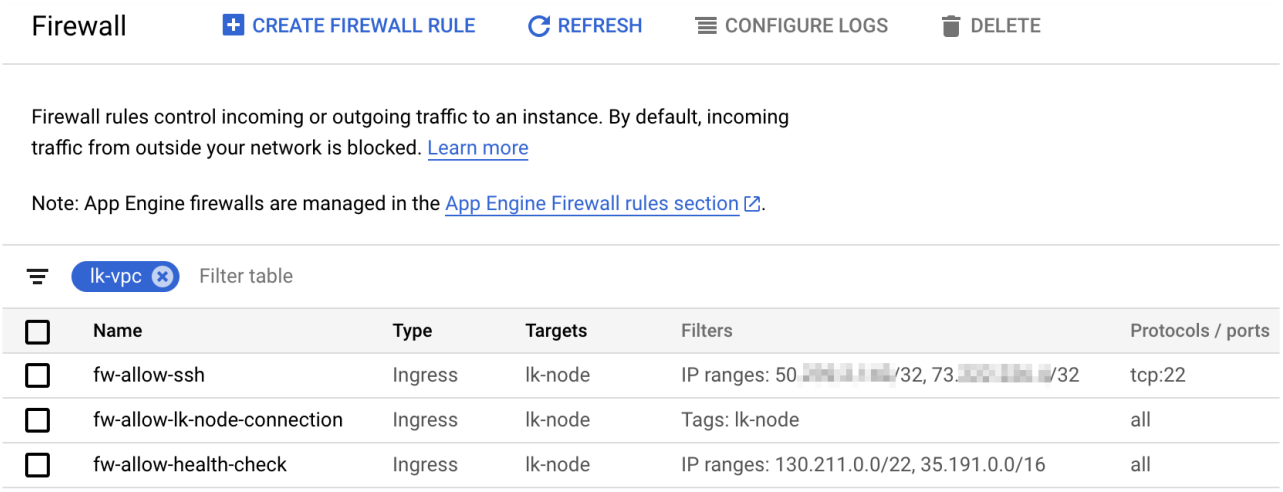

6. Select Targets (destination of the traffic). Make sure to select the `lk-node` tag that is assigned to the VM you created.

7. Select source (WAN IP address(es) of your work location).

8. Finally, select the protocol to allow. Confirm all values you have entered and create a rule.




9. Now a rule is defined. Use the same steps to create other rules.

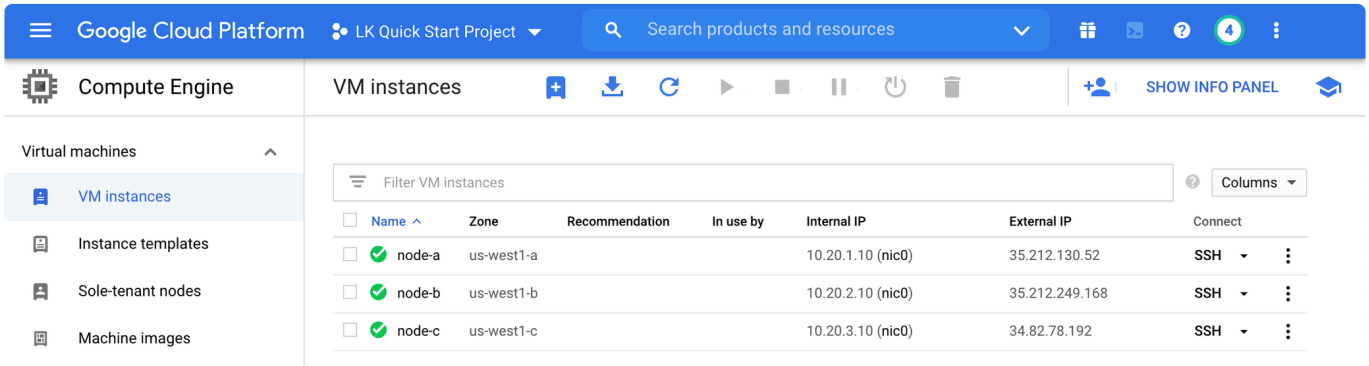

Firewall
CREATE FIREWALL RULE
REFRESH
CONFIGURE LOGS
DELETE
Firewall rules control incoming or outgoing traffic to an instance. By default, incoming traffic from outside your network is blocked. Learn more
Note: App Engine firewalls are managed in the App Engine Firewall rules section.
lk-vpc
Filter table
Name
Type
Targets
Filters
Protocols / ports
fw-allow-ssh
Ingress
lk-node
IP ranges: 50 /32, 73 /32
tcp:22
fw-allow-lk-node-connection
Ingress
lk-node
Tags: lk-node
all
fw-allow-health-check
Ingress
lk-node
IP ranges: 130.211.0.0/22, 35.191.0.0/16
all

# 11.2.3.5.8. Creating the Second and Third VM

The steps required to create the second node (node-b) and third node (node-c) are almost the same as previously described for the first node (node-a). The following table illustrates the differences between the three nodes and the details required to create these instances.

| Name | Parameter | Value |
|---|---|---|
| Common values across VM | **VPC network** | **lk-vpc** |
| | **Subnet** | **lk-subnet** |
| | **Network Tags** | **lk-node** |
| | **External IP** | **Ephemeral** |
| node-a | Zone | us-west1-a (*1) |
| | Hostname | node-a.internal |
| | Internal IP | `10.20.1.10` |
| | Second Disk (Storage Device) | You need to have a second disk |
| node-b | Zone | us-west1-b (*1) |
| | Hostname | node-b.internal |
| | Internal IP | `10.20.2.10` |
| | Second Disk (Storage Device) | You need to have a second disk |
| node-c | Zone | us-west1-c (*1) |
| | Hostname | node-c.internal |
| | Internal IP | `10.20.3.10` |
| | Second Disk (Storage Device) | You do not need to create a second disk |

Once created, the list of VMs should look like this:



# 11.2.4. Configure Linux Nodes to Run SIOS Protection Suite for Linux

This section outlines steps to configure the Linux nodes before installing SIOS Protection Suite for Linux.

In order to connect a Linux node, it may be necessary to install the ssh client software on Windows.

- Connecting to a Linux Node from Windows Client Using ssh

## Configuring the Nodes

The following steps should be executed for each node:

- Set a Hostname for Each Instance
- Disable SELinux
- Disable the Firewall
- Set a Password for the Root User
- Install x11

# 11.2.4.1. Connecting to a Linux Node from Windows Client Using ssh

Linux nodes can accept a login using ssh and an ssh client is therefore required. This page explains how to use PuTTY to connect to the Linux node using a public/private key pair (instead of a password). It also covers how to create a basic configuration on the Linux node and install SIOS Protection Suite for Linux. After installing SIOS Protection Suite, we will add more tools to enable x11 to use the GUI based applications.

## Download and Install Tools

Download and install the following software:

- PuTTY (ssh client on Microsoft Windows)

Download the latest PuTTY software and install it.

## Converting the Private Key (.pem)

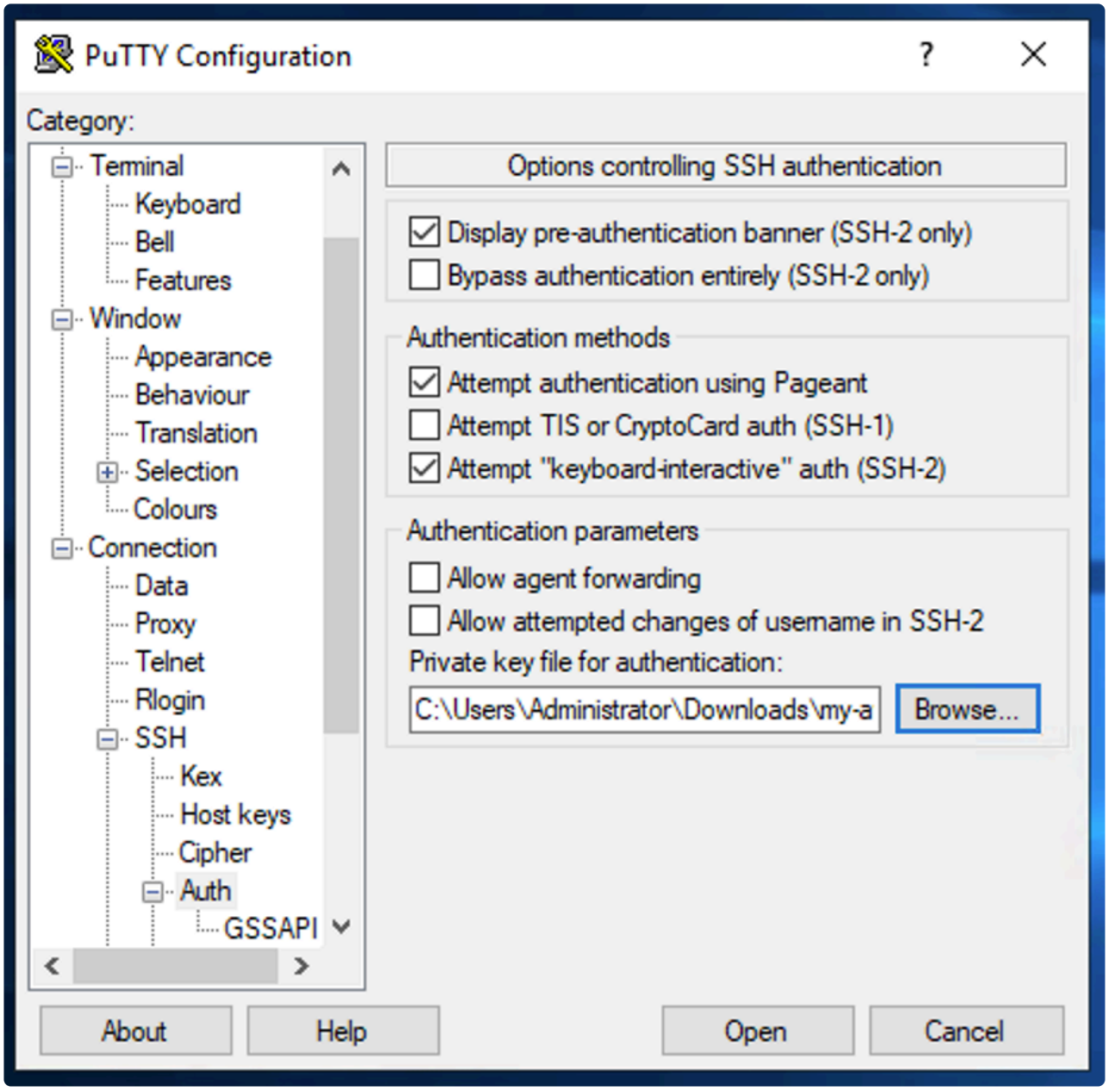
If you have a Private Key file in a `<filename> .pem` format, convert it to a PuTTY Private Key File `(*.ppk)` file.

To convert the file:

1. Start the ‘PuTTY Key Generator’.
2. Select Load, then specify your `<filename>.pem` file.
3. Select ‘Save private key’ as `<filename>.ppk`.

## Configure PuTTY

1. Open PuTTY and select Connection > SSH > Auth.
2. Select the private key file at ‘Private key file for authentication’.

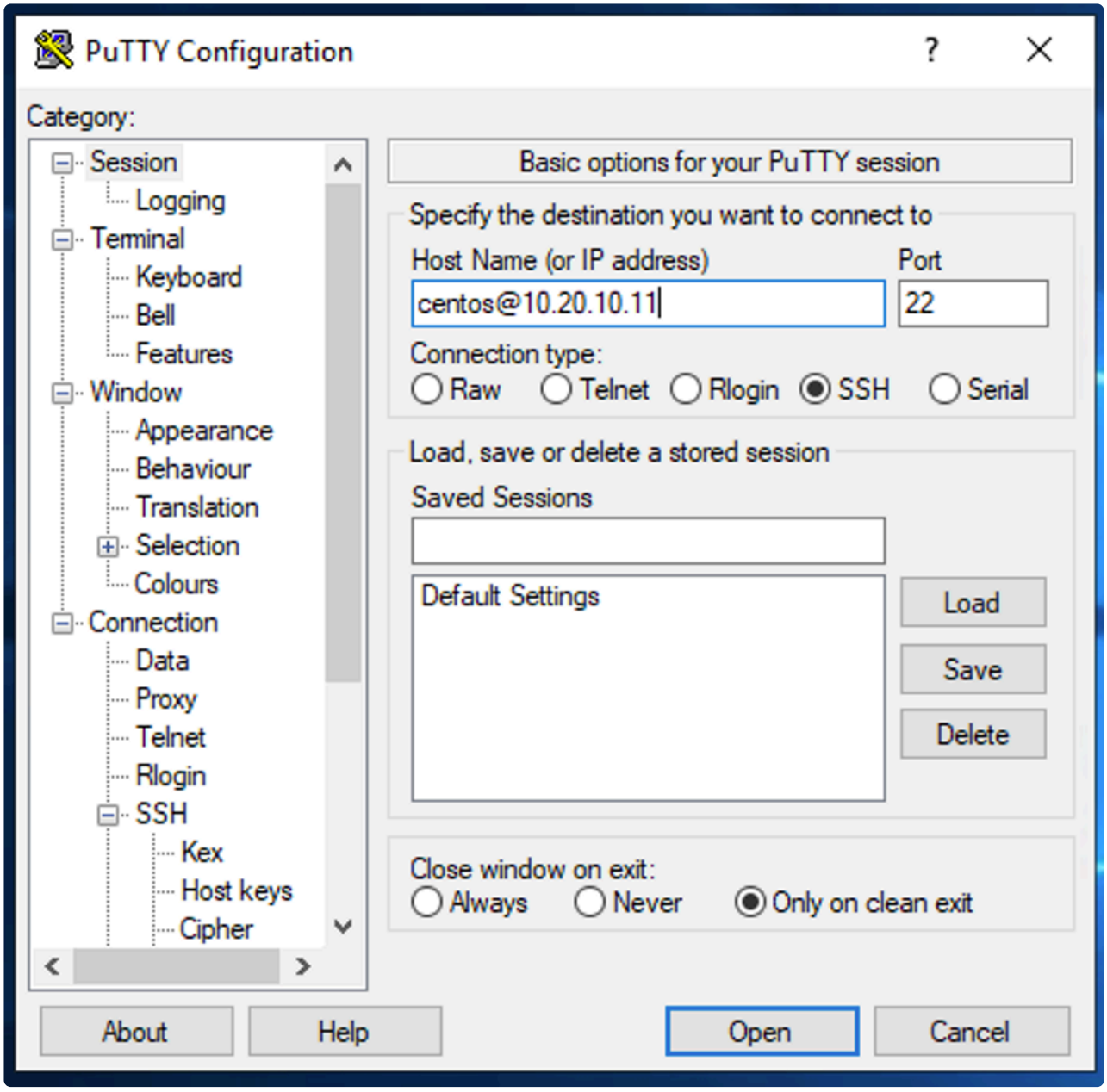



.

3. You may want to save your configuration as a ‘Session’ page to avoid reconfiguring each time.

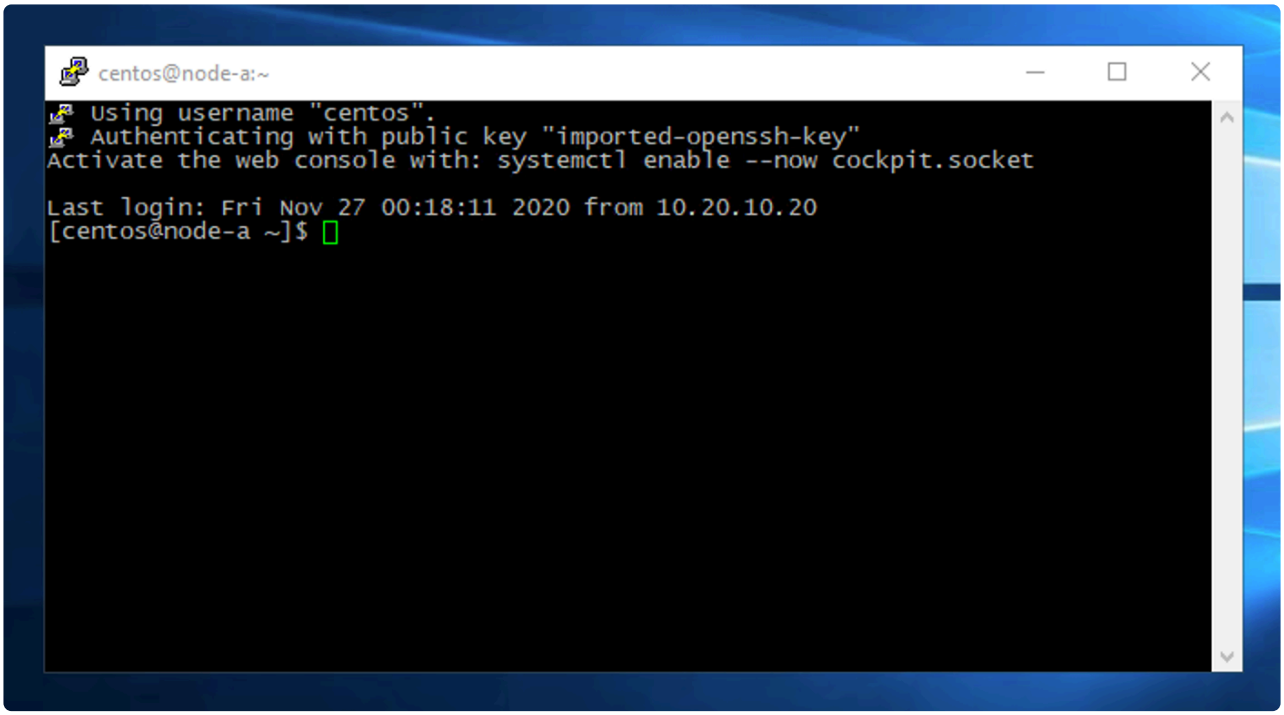
## Connect to a Linux Instance

1. Start the session from PuTTY.




2. Confirm the security alert that indicates the server has not been connected to before. Select “Yes” to proceed.

3. You are now connected!


centos@node-a:~
Using username "centos".
Authenticating with public key "imported-openssh-key"
Activate the web console with: systemctl enable --now cockpit.socket
Last login: Fri Nov 27 00:18:11 2020 from 10.20.10.20
[centos@node-a ~]$

# 11.2.4.2. Set a Hostname for Each Instance

Execute the following commands for each Node.

Once the instances are created, set a hostname for each node and then add an entry to `/etc/hosts` file so that the nodes can locate each other. You can also use DNS if it's available in your environment.

1. use `hostnamectl` command to set a hostname:

```
$ sudo su -
# hostname
ip-10-20-1-10.us-west-2.compute.internal
# hostnamectl set-hostname node-a
```

2. If a DNS server is available, register your hostname(s) using the server. If there is not a DNS server available, using `/etc/hosts` is recommended to resolve the hostnames.

```
# cat /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
```

3. Add the following lines to the `/etc/hosts` file so that `node-a` and `node-b` can be resolved to an IP address(es).

```
10.20.1.10 node-a
10.20.2.10 node-b
10.20.3.10 node-c
```

4. Create a backup of `/etc/hosts`, edit the file and confirm the changes.

```
# cp -p /etc/hosts /etc/hosts.org
# vi /etc/hosts
# cat /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
10.20.1.10 node-a
10.20.2.10 node-b
10.20.3.10 node-c
```

5. Make sure you can ping these nodes by name.

```
# ping node-b
PING node-b (10.20.2.10) 56(84) bytes of data.
64 bytes from node-b (10.20.2.10): icmp_seq=1 ttl=64 time=0.868 ms
64 bytes from node-b (10.20.2.10): icmp_seq=2 ttl=64 time=0.892 ms
^C
```

# 11.2.4.3. Disable SELinux

Security-Enhanced Linux (SELinux) is a Linux kernel security module that provides a mechanism for supporting access control security policies, including mandatory access controls (MAC).

https://selinuxproject.org/page/Main_Page

You will need to disable it in order to use SIOS Protection Suite for Linux.

## Steps to Disable SELinux

1. Check `sestatus`.

```
$ sudo su
# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      32
```

2. Set enforce to 0.

```
# set enforce 0
```

3. Confirm current value specified at `/etc/selinux/config`.

```
# grep -e SELINUX= /etc/selinux/config
# SELINUX= can take one of these three values:
SELINUX=enforcing
```

4. In step 3, SELinux is set to `enforcing`. Replace the value with `disabled`.

```
# cp -p /etc/selinux/config /etc/selinux/config.org
# sed -i -e 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/selinux/config
```

5. Confirm the change.

```
# grep -e SELINUX= /etc/selinux/config
# SELINUX= can take one of these three values:
SELINUX=disabled
```

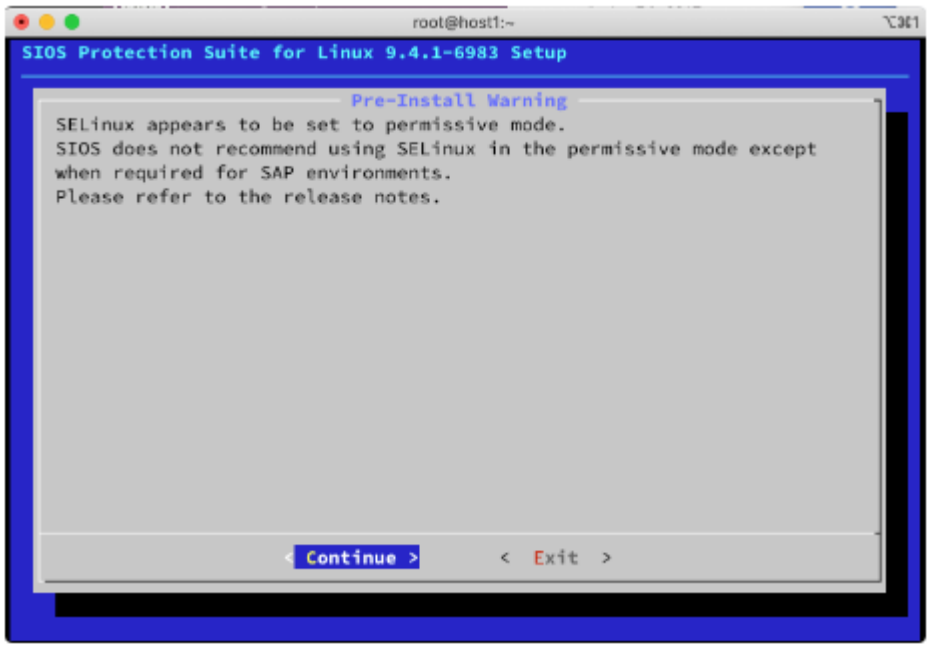
6. Restart the machine.

```
# shutdown -r now
```

7. After the restart, check the status again.

```
$ sestatus
SELinux status:                 disabled
```

> ! In the latest SPS-Linux versions “permissive“ mode is now supported. The installer will still warn if set to permissive but we can safely continue.



# 11.2.4.4. Disable the Firewall

✻ The following commands must be executed for each node.

In this section we will disable the `firewalld` service to install SIOS Protection Suite for Linux. The firewall can be left active but the ports required by SIOS Protection Suite and the application you are protecting will have to be configured (not shown here).

## Steps to Disable `firewalld`

Disable `firewalld` using the following steps.

1. Check the current status.

```
# firewall-cmd --state
running
```

! If the following is displayed (indicating that `firewalld` is not installed), steps 2 and 3 can be skipped.

```
# firewall-cmd --state
bash: firewall-cmd: command not found
```

2. Stop and disable `firewalld`.

```
# systemctl stop firewalld
# systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.

# systemctl mask --now firewalld
Created symlink from /etc/systemd/system/firewalld.service to /dev/null.
```

3. Confirm that it is disabled.

```
# firewall-cmd --state
not running
```

# 11.2.4.5. Set a Password for the Root User

✻ The following commands must be executed for each node.

This tutorial uses the `root` user to login into the SIOS Protection Suite for Linux GUI and it requires a password to perform operations. Please define a password for root user.

1. Set a `root` password.

```
# passwd
Changing password for user root.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

# 11.2.4.6. Install x11

The SIOS Protection Suite for the Linux GUI uses the X Window System. We need to install it on each node.

- Install x11 to the machine.
- Install x11 related packages and enable it with the following steps:

1. Install x11 related packages.

```
# yum install  xorg-x11-server-Xorg xorg-x11-xauth -y
Last metadata expiration check: 0:00:51 ago on Wed 18 Nov 2020 12:36:11 AM UT
C.
Dependencies resolved.
================================================================================
==================================
 Package                          Architecture    Version                    Re
pository                            Size
================================================================================
==================================
Installing:
 xorg-x11-server-Xorg             x86_64          1.20.8-6.el8               rh
el-8-appstream-rhui-rpms          1.5 M
 xorg-x11-xauth                   x86_64          1:1.0.9-12.el8             rh
el-8-appstream-rhui-rpms            39 k
(snip)
  xorg-x11-drv-libinput-0.29.0-1.el8.x86_64                xorg-x11-drv-ves
a-2.4.0-3.el8.x86_64
  xorg-x11-server-Xorg-1.20.8-6.el8.x86_64                 xorg-x11-server-co
mmon-1.20.8-6.el8.x86_64
  xorg-x11-xauth-1:1.0.9-12.el8.x86_64                     xorg-x11-xkb-util
s-7.7-28.el8.x86_64
Complete!
```

2. Check the `X11Forwarding` parameter in `/etc/ssh/sshd_config` to confirm that it is set to `yes`. If it is not set to yes, edit the file.

```
# grep -e X11Forwarding /etc/ssh/sshd_config
X11Forwarding yes
```

3. Assuming the system is booted to X11, start GUI sessions at start-up. If not, this step can be ignored.

```
# systemctl set-default graphical.target
Removed /etc/systemd/system/default.target.
Created symlink /etc/systemd/system/default.target → /usr/lib/systemd/system/g
raphical.target.
```

4. Restart the machine.

```
1 # shutdown -r now
```

# 11.2.5. Install SIOS Protection Suite for Linux

The following commands should be executed for each node.

## Mount ISO Image for Install

1. Download the ISO image and license key files to the local environment.
2. Copy the ISO image to each of the nodes.
3. Copy the license file to each of the nodes.
4. Log into each node and mount the installer image.

```
$ ls
evalkeys.txt  sps951.img
$ sudo su
# mount -o loop -t iso9660 sps951.img /media
mount: /media: WARNING: device write-protected, mounted read-only.
```

5. Check the content of the image.

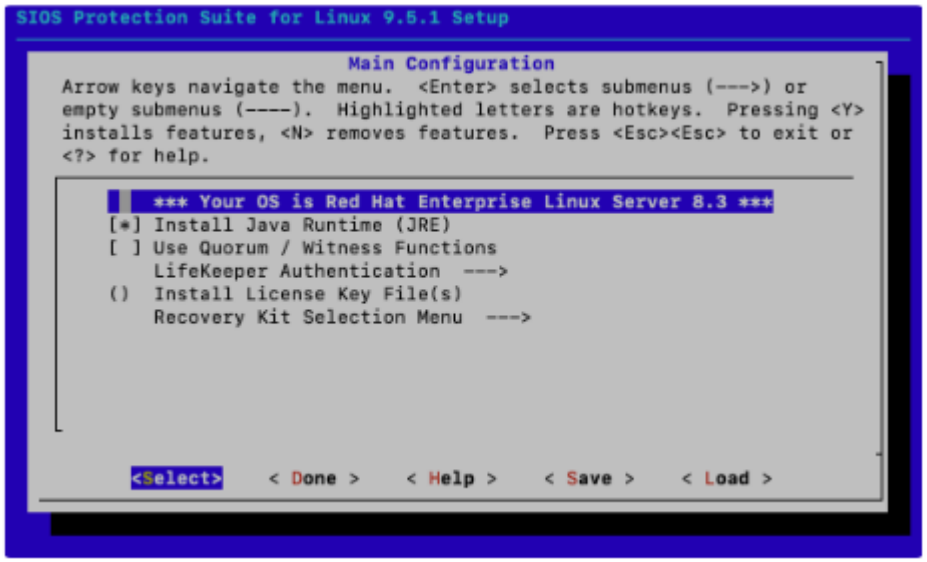
```
# cd /media
# ls
CentOS     core                              java    README        setuplibs
Chef       create_response_file              kits    RHAS          shfuncs
common     HADR-9.5.1-7154.src.rpm           OEL     sapconnector  SuSE
COPYRIGHT  HADR-generic-9.5.1-7154.x86_64.rpm quorum  setup         TRANS.TBL
```

## Install SIOS Protection Suite for Linux

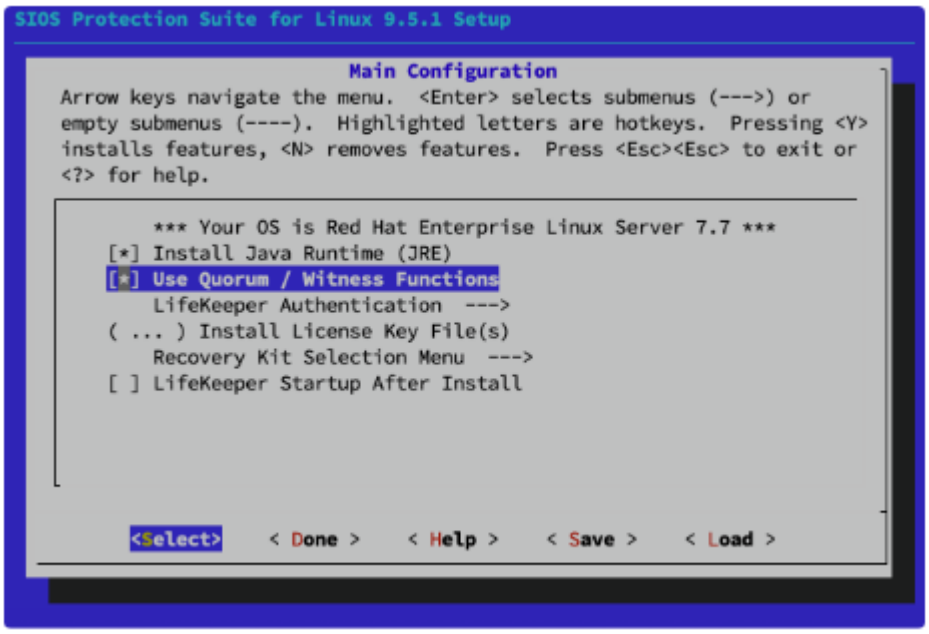
The installer should be located at `/media/setup`. Execute the setup script to install SIOS Protection Suite for Linux.

```
# ./setup
SIOS Protection Suite for Linux Setup
Validating files.............................OK
Collecting system information.................................done.
Preparing configuration information............done.
```

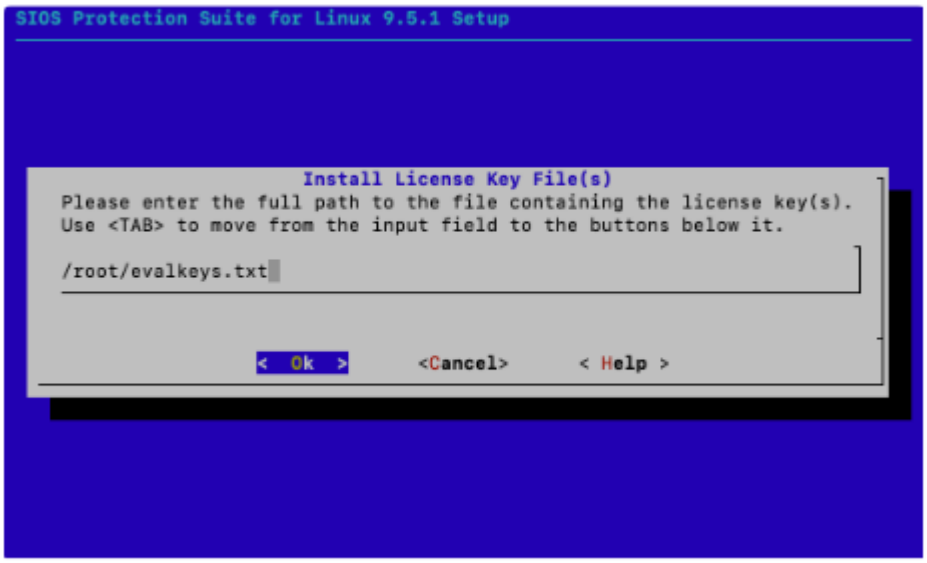
1. The GUI Installer starts as shown below.




2. Check “Use Quorum / Witness Functions”.

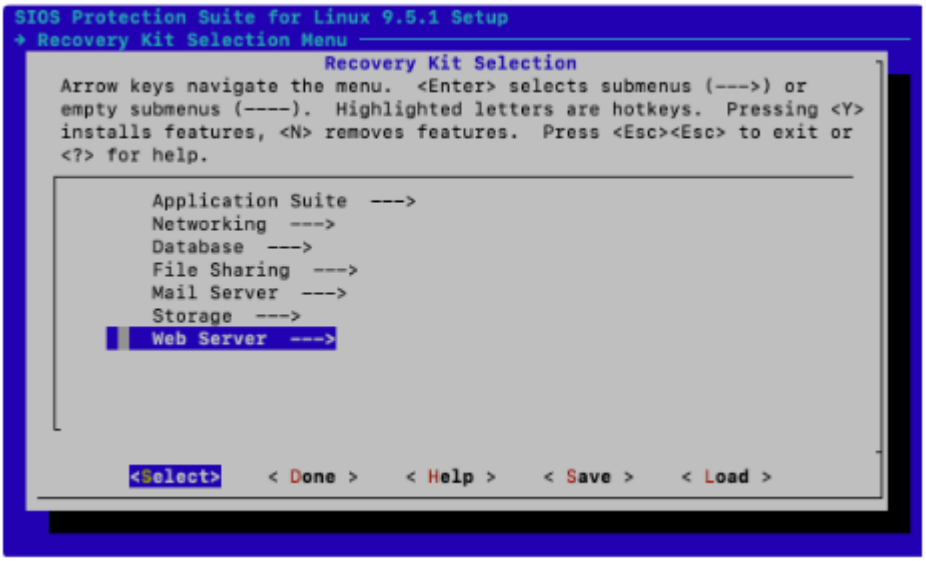



3. Specify the location of the license file.

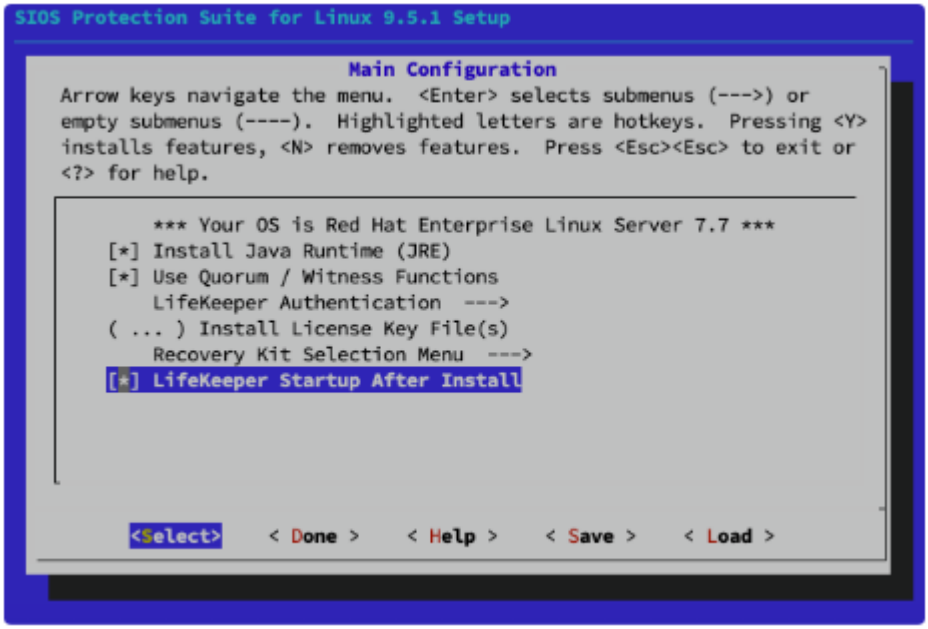



4. Select the Recovery Kits to be installed.

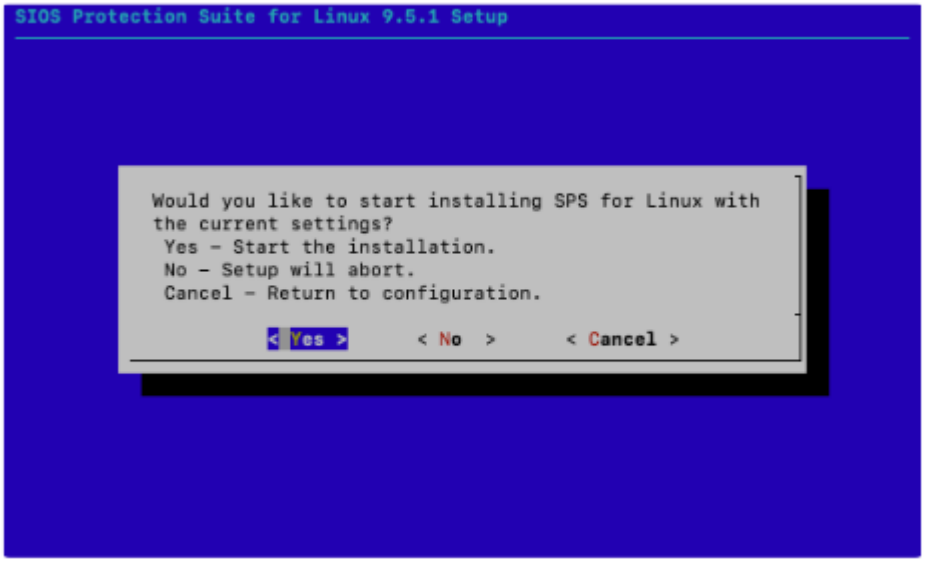
> ! On node-c (the Witness Node), **NO** recovery kit should be selected.



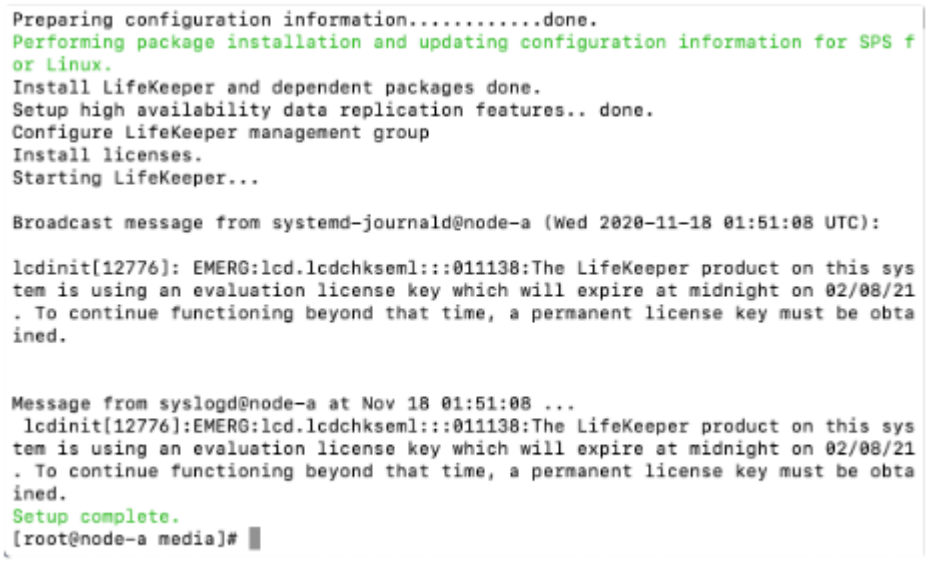

5. Check “LifeKeeper Startup After Install”, then select “Done”.




6. Start the Install process.




7. Install is now completed. Repeat the steps for node-b and node-c.




## Additional Environment Specific Tasks to Complete

If AWS is being used, complete the following steps:

- Disable PING Broadcasting
- Install AWS CLI
- Assign Permission to Use EC2 Recovery Kit

If Azure is being used, complete the following step:

- Disable PING Broadcasting

# 11.2.5.1. Install AWS CLI

The following commands must be executed on each node.

First, the AWS CLI must be installed. Detailed steps are documented at Installing, updating, and uninstalling the AWS CLI version 2 on Linux.

Once the CLI is installed, add the location of the CLI to the `/etc/default/LifeKeeper` file.

1. Check the location of `aws` CLI.

```
# curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
# unzip awscliv2.zip
# sudo ./aws/install
You can now run: /usr/local/bin/aws --version
```

2. Update `/etc/default/LifeKeeper` to include this location in the PATH.

```
# vi /etc/default/LifeKeeper
# grep -e PATH= /etc/default/LifeKeeper
PATH=/opt/LifeKeeper/bin:/bin:/usr/bin:/usr/sbin:/sbin:/usr/local/bin
```

# 11.2.5.2. Assign Permission to Use EC2 Recovery Kit

> ! In order to complete the following steps, the AWS CLI must be installed on each node. See Additional Environment Specific Tasks to Complete for more information.

To use the EC2 Recovery Kit, the instance must have Roles that are allowed to update the RouteTable entries or reassign ENI (Elastic Network Interface).

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress
- ec2:DescribeRouteTables
- ec2:ReplaceRoute

To achieve this, create a policy as seen below (note that it might be desirable to limit the resources that may be accessed), then assign it to a Role.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "ec2:DisassociateAddress",
                "ec2:DescribeAddresses",
                "ec2:AssociateAddress",
                "ec2:DescribeRouteTables",
                "ec2:ReplaceRoute"
            ],
            "Resource": "*"
        }
    ]
}
```

Once a Role is defined, assign it to the EC2 instances.

# 11.2.5.3. Disable PING Broadcasting

When LifeKeeper is used in a cloud environment, it may not be possible to use broadcast ping to find other machines on the subnet. Therefore, this should be disabled by changing the configuration of LifeKeeper as follows:

1. Check the current configuration.

```
# grep -e NOBCASTPING /etc/default/LifeKeeper
NOBCASTPING=0     # Can be used to disable the broadcast ping mechanism
```

2. `NOBCASTPING` must be set to 1.

```
# cp -p /etc/default/LifeKeeper /etc/default/LifeKeeper.org
# sed -i -e 's/NOBCASTPING=0/NOBCASTPING=1/g' /etc/default/LifeKeeper
```

3. Confirm the change.

```
# grep -e NOBCASTPING /etc/default/LifeKeeper
NOBCASTPING=1     # Can be used to disable the broadcast ping mechanism
```

# 11.2.6. Login and Basic Configuration Tasks

By now you should have 3 nodes, each with SIOS Protection Suite for Linux installed and ready for use.

## Connecting to Linux Node with the GUI

SIOS Protection Suite has a GUI application to configure the cluster. Please refer to Connecting to a Linux Node with x11 Forwarding for more details about accessing the LifeKeeper GUI.

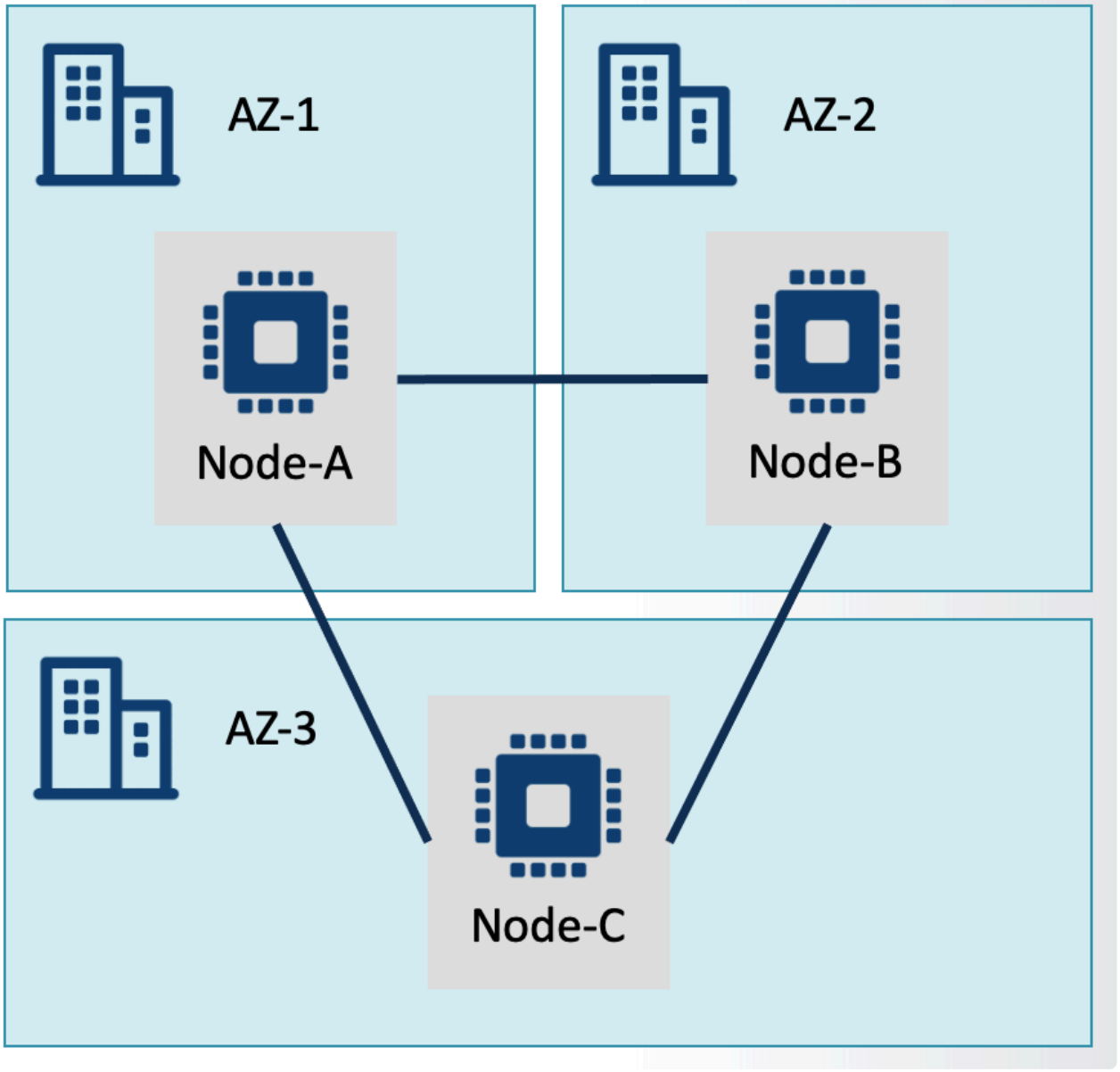
## Basic Configuration

In this tutorial there are three nodes with the following IP address range:

| Node | IP Address |
|---|---|
| node-a (Active) | 10.20.1.10 |
| node-b (Standby) | 10.20.2.10 |
| node-c (Quorum / Witness) | 10.20.3.10 |

The basic configuration is completed using the following steps:

- Connecting to the First node
- Connecting to Other Nodes
- Define the Communication Path **Note:** There are three nodes in the cluster including node-c (Quorum/Witness). Therefore, 3 communication paths must be defined in order to create bidirectional communication paths between all pairs of nodes.
  - `node-a` **<—>** `node-b`
  - `node-a` **<—>** `node-c`
  - `node-b` **<—>** `node-c`


AZ-1
AZ-2
Node-A
Node-B
AZ-3
Node-C

# 11.2.6.1. Connecting to a Linux Node with “X11 Forwarding”

X Window System is used to enable bitmap based Graphical User Interface on Unix based operating systems. This is commonly referred to as “X11”. One of the interesting usages of X11 is the ability to transfer “a window” to the terminal (a machine which an operator uses). The original X11 process runs on the server but the X11 User Interface appears on the client machine terminal, This is called “X11 forwarding”.

In order to use X11 forwarding an X11 graphic server must be run (this is called an X11 server even though it actually runs on the client machine) on the local machine.

The LifeKeeper GUI runs on X Window Platform and therefore a client that runs X11 software is required (unless a GUI login is used on these nodes directly).

## Install Client Software to Microsoft Windows

To use Microsoft Windows as an “x11 server” it is necessary to install software on the Microsoft Windows client. The following page explains the basic steps required to install the software on a Windows client.

- Setup X Window Client Software on Microsoft Windows

## Setup Linux Nodes to Accept X11 Forwarding

1. Find the current value of the DISPLAY environment variable as the regular user that was set up in the previous steps.

   ```
   $ echo $DISPLAY
   localhost:10.0
   ```

2. View the current list of xauth keys for the user and identify the one associated with the current display that X11 is being forwarded on (10 in this example).

   ```
   $ xauth list
   node-a/unix:11 MIT-MAGIC-COOKIE-1 7289ba26871a37b94ff359df829e2686
   node-a/unix:10 MIT-MAGIC-COOKIE-1 bae592c842916f23e3ba066ba594c5d0
   ```

3. Become the root user.

   ```
   $ sudo -i
   ```

4. View the DISPLAY environment variable for the root user.

   ```
   # echo $DISPLAY
   ```

```
localhost:10.0
```

> Note: If the DISPLAY variable is not set for the root user, set it to the same value found in step 1:
> `# export DISPLAY=localhost:10.0`

5. View the current list of xauth keys for the root user.

   ```
   # xauth list
   node-a/unix:1 MIT-MAGIC-COOKIE-1 5b249d176f8da2f7803343f6830002eb
   node-a/unix:2 MIT-MAGIC-COOKIE-1 e62f7abaeda1d200576aac40aec2ba83
   ```

   If there is no entry for the display found in step 1, add the relevant entry from the xauth list output in step 2 to the list of keys for the root user.

   ```
   # xauth add node-a/unix:10 MIT-MAGIC-COOKIE-1
   bae592c842916f23e3ba066ba594c5d0
   ```

6. Once the DISPLAY variable and xauth keys are set up correctly for the root user, it is now possible to run graphical applications as root from an SSH session.

   ```
   # /opt/LifeKeeper/bin/lkGUIapp &
   [1] 23957
   openjdk version "12.0.2" 2019-07-16
   OpenJDK Runtime Environment (build 12.0.2+10)
   OpenJDK 64-Bit Server VM (build 12.0.2+10, mixed mode, sharing)
   Setting up secure random number generator
   Random number setup completed
   Connecting to rmi://node-a:82/LKRemoteInterface at:
         Sat Mar 13 04:23:19 UTC 2021
   Connection to node-a succeeded at:
         Sat Mar 13 04:23:20 UTC 2021
   ```

# 11.2.6.2. Setup X Window Client Software on Microsoft Windows

After following the steps in Connecting to a Linux Node from Windows Client Using ssh, PuTTY has been installed and configured. This following explains how to enable X11 forwarding so that it is possible to connect with nodes and use GUI based applications.

## Download and Install Tools

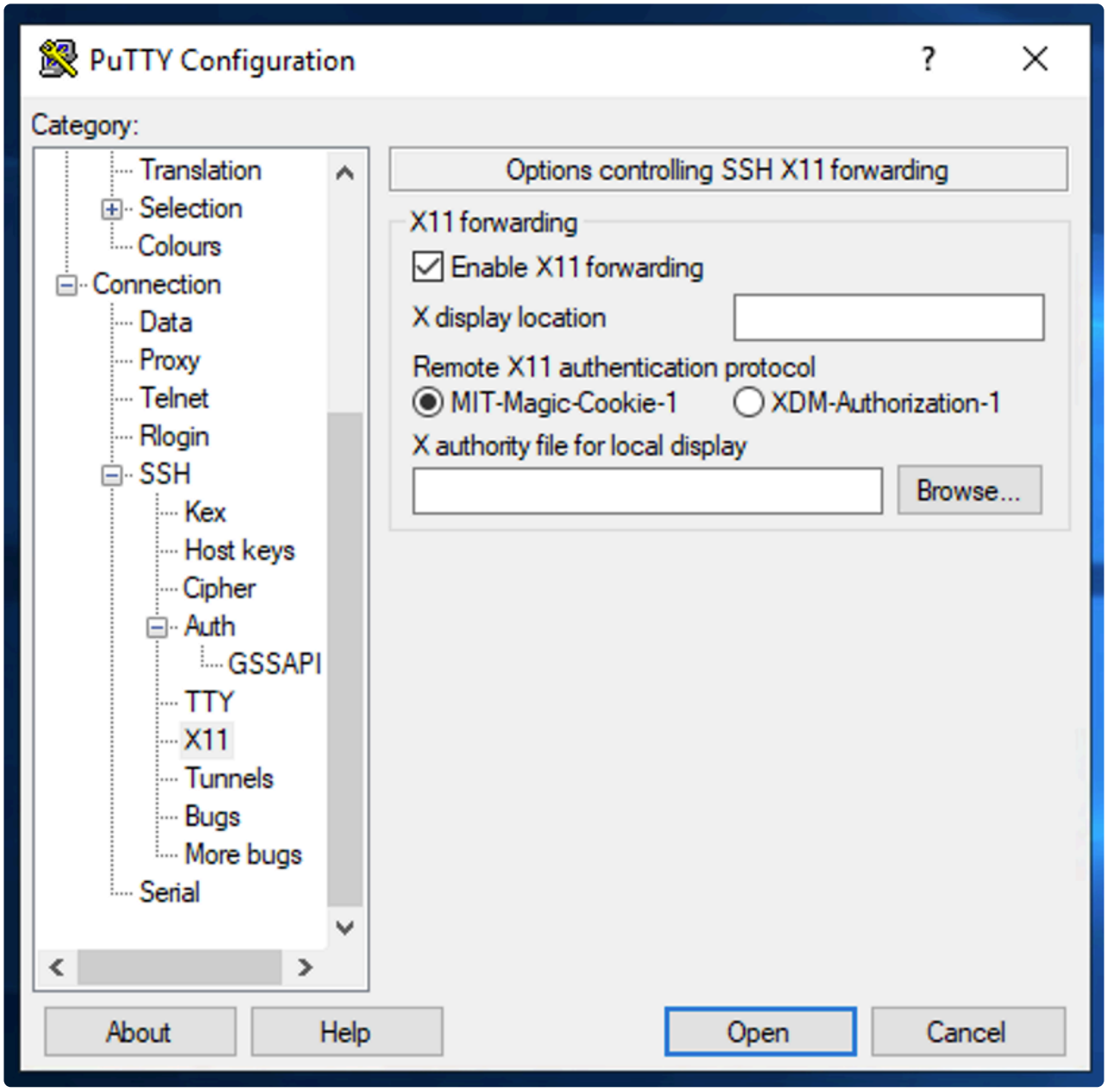
Download and install the following additional software:

- XMing (X11 server on Microsoft Windows, other X11 windows clients are available)

## Configure PuTTY

1. Open PuTTY and display Connection > SSH > Auth.

2. Display Connection > SSH > X11 and check ‘Enable X11 forwarding’.

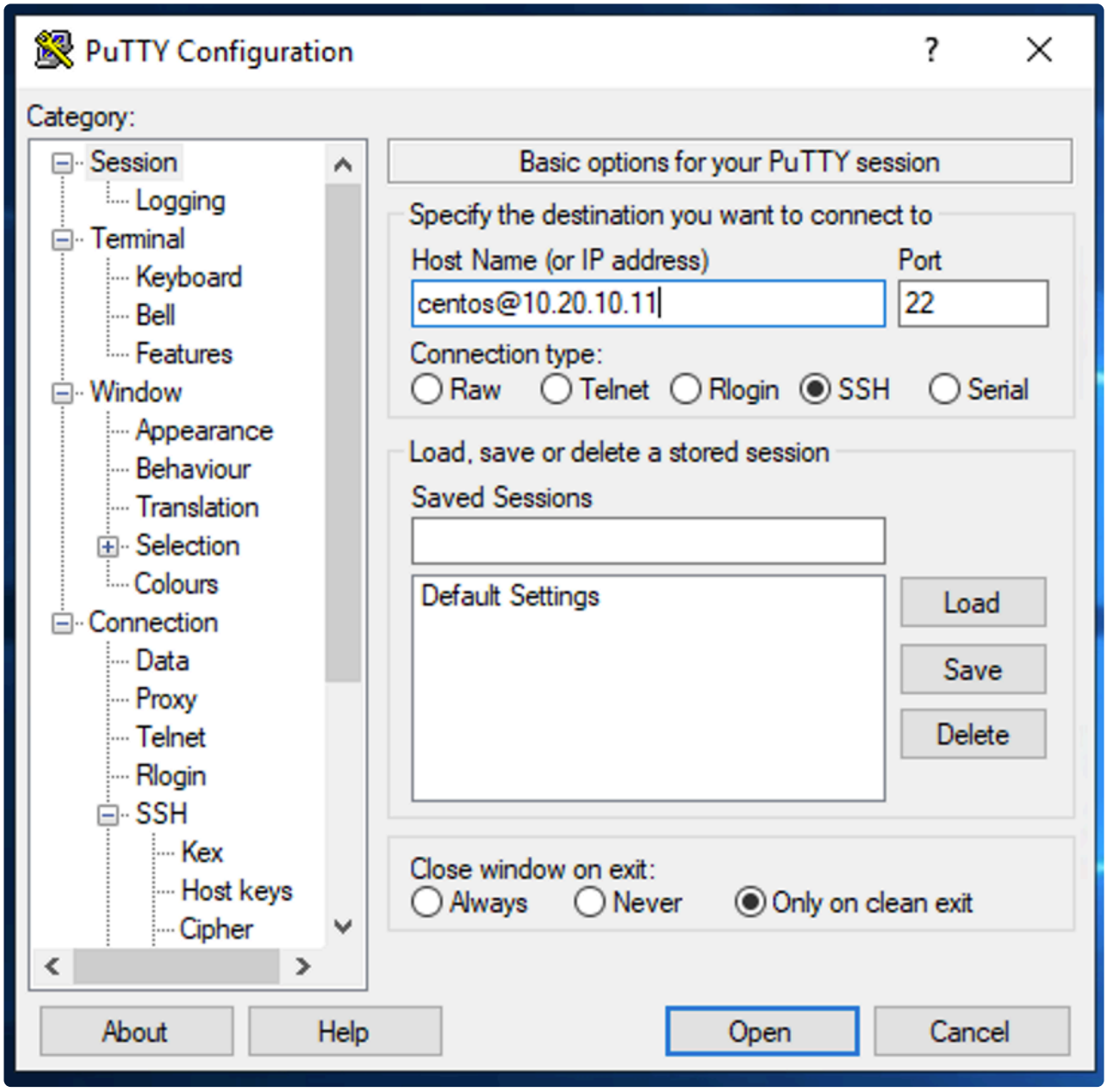



3. It is recommended that the configuration is saved on the ‘Session’ page (otherwise these steps will need to be repeated each time).

## Connect to a Linux Instance

1. Start Xming (ensure that the Ming icon is shown in the system tray).

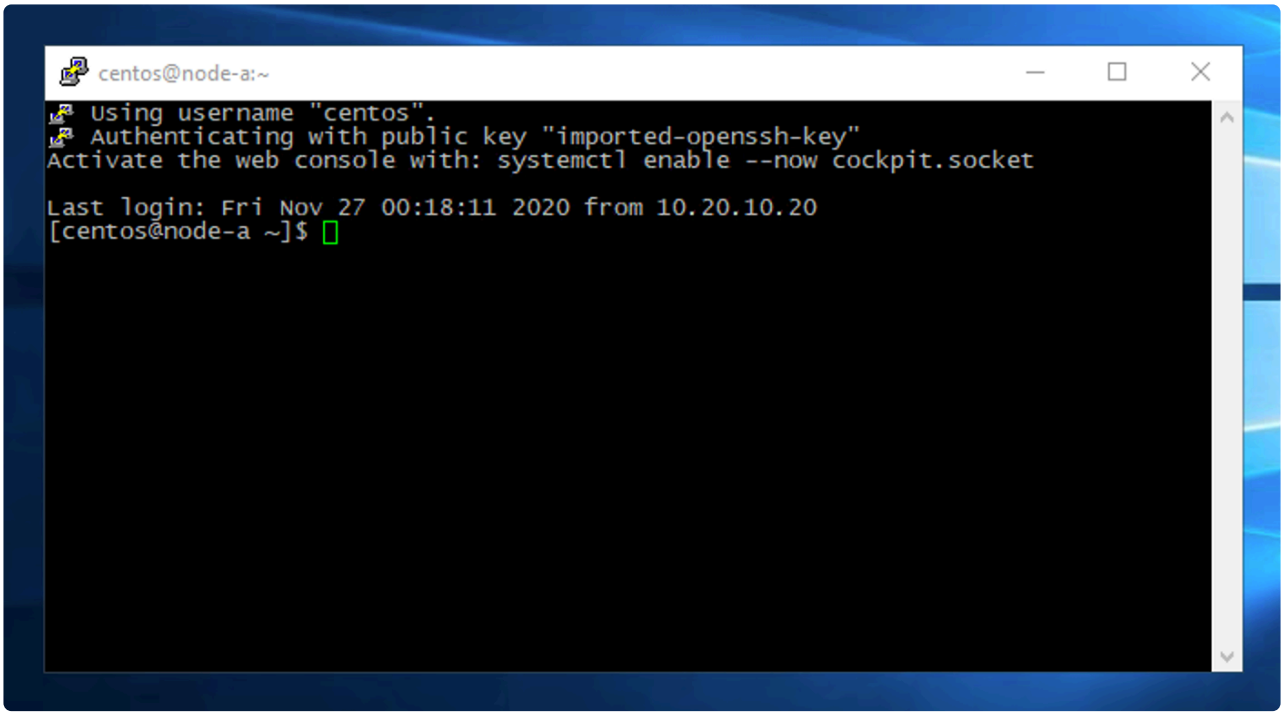
2. Start the session from PuTTY.




3. Confirm the security alert indicating the server has not connected before. Select “Yes” to proceed.

4. If the warning ~/.Xauthority does not exist is seen, use touch to create the file, then reconnect to the node.

   You should now be connected.

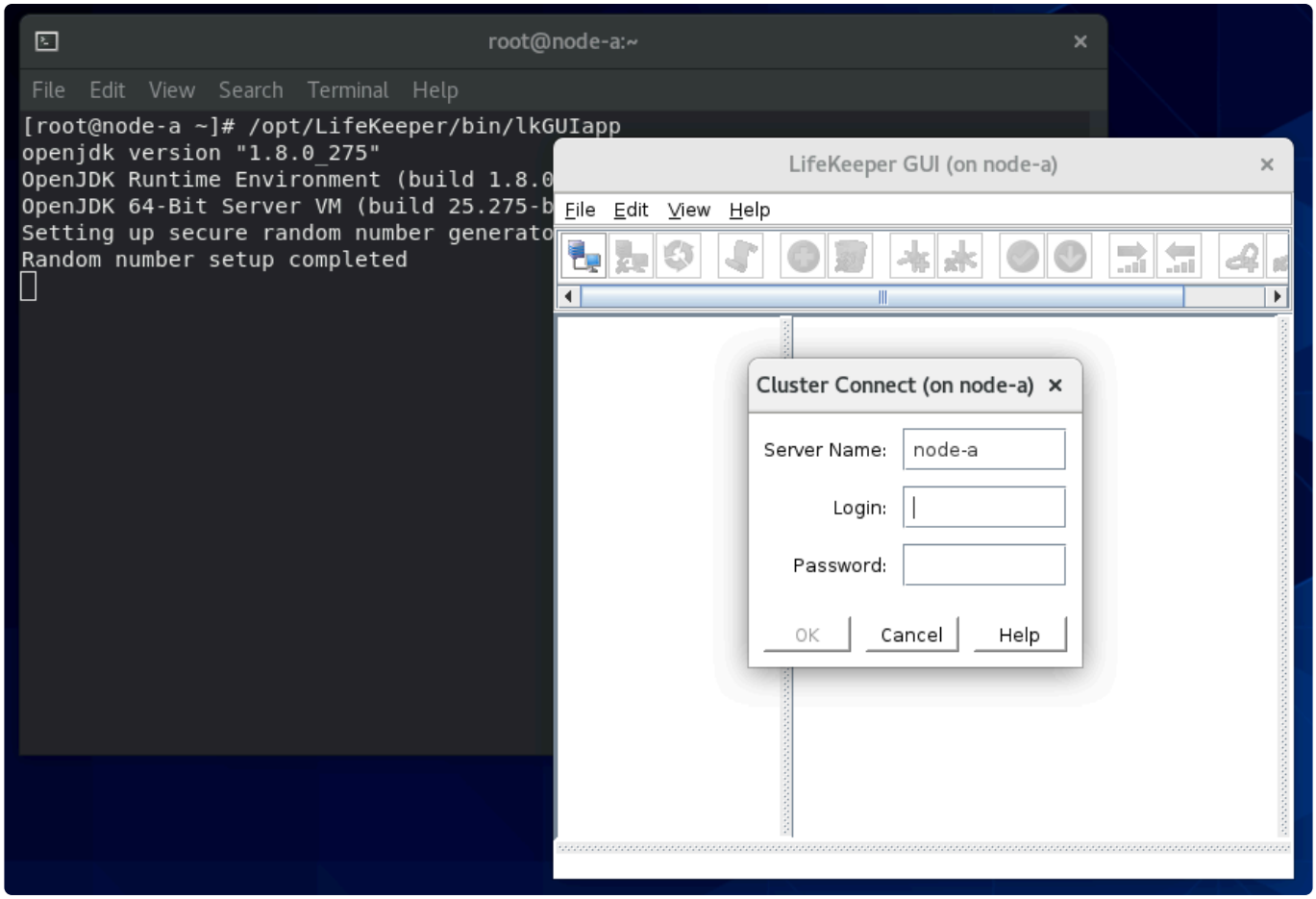

centos@node-a:~
Using username "centos".
Authenticating with public key "imported-openssh-key"
Activate the web console with: systemctl enable --now cockpit.socket
Last login: Fri Nov 27 00:18:11 2020 from 10.20.10.20
[centos@node-a ~]$

# 11.2.6.3. Connecting to the First Node (node-a)

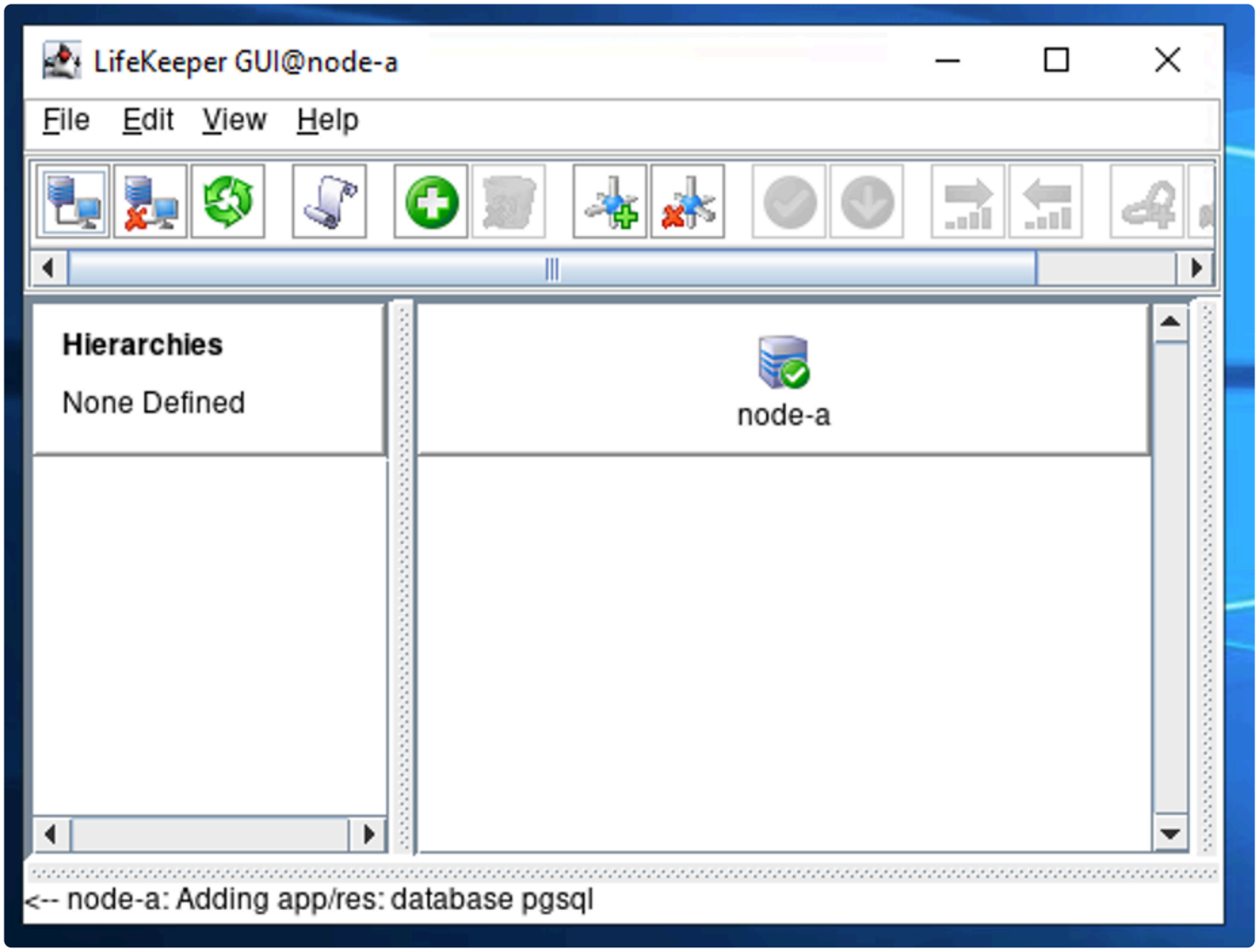
Once logged in to the node with “X11 forwarding” enabled, type the following command to start the SIOS Protection Suite for Linux GUI:

```
# /opt/LifeKeeper/bin/lkGUIapp &
```

In the login dialog pop-up, enter the root username and password to login:

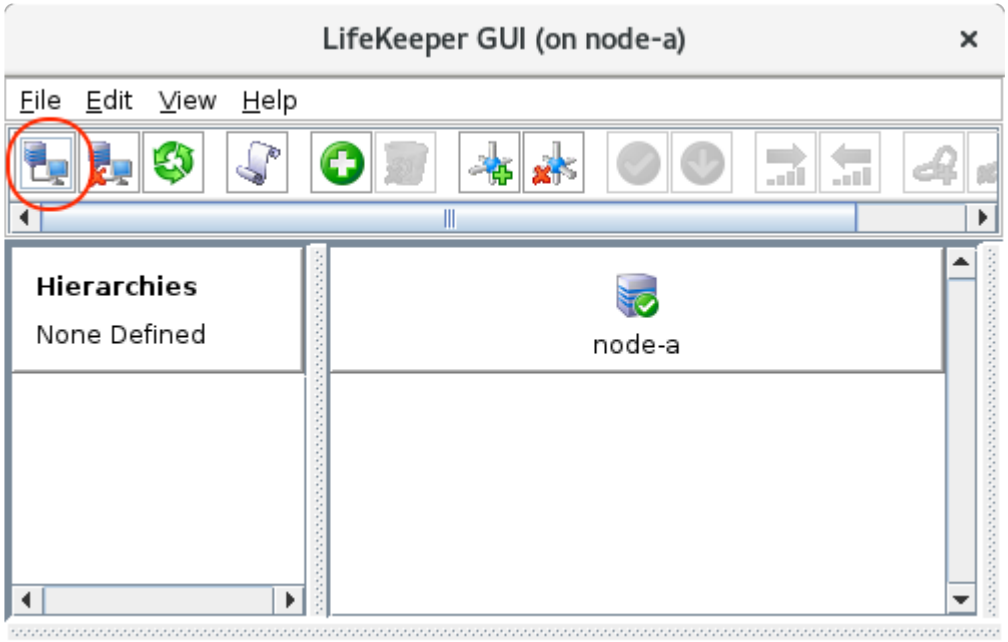



After successfully logging in, you should see node-a in the GUI.

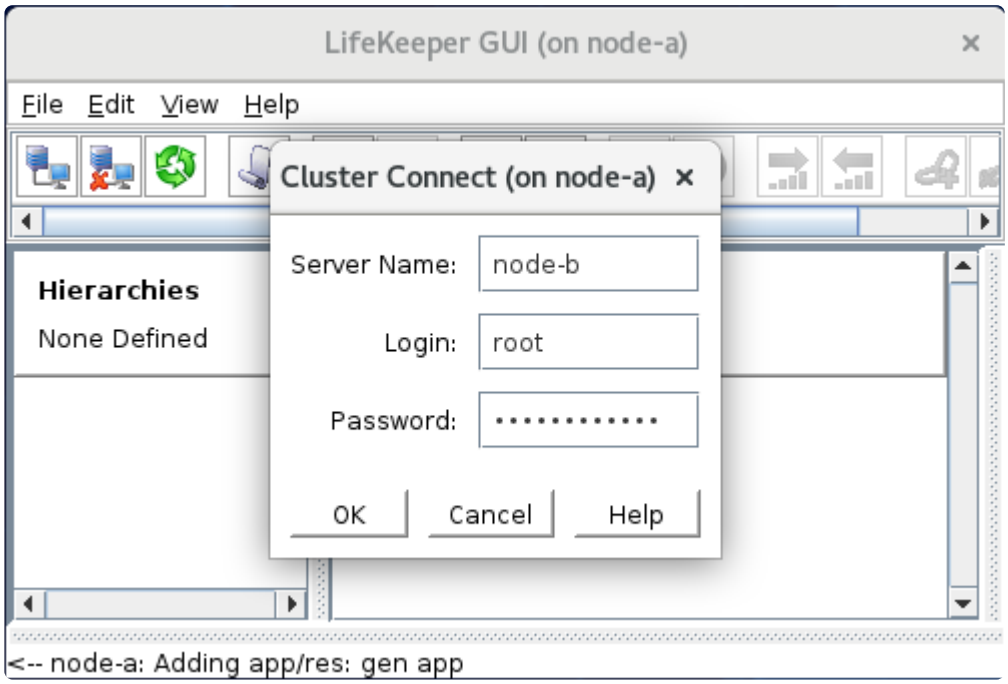

LifeKeeper GUI@node-a
File Edit View Help
Hierarchies
None Defined
node-a
<-- node-a: Adding app/res: database pgsql

# 11.2.6.4. Connecting to Other Nodes (node-b and node-c)

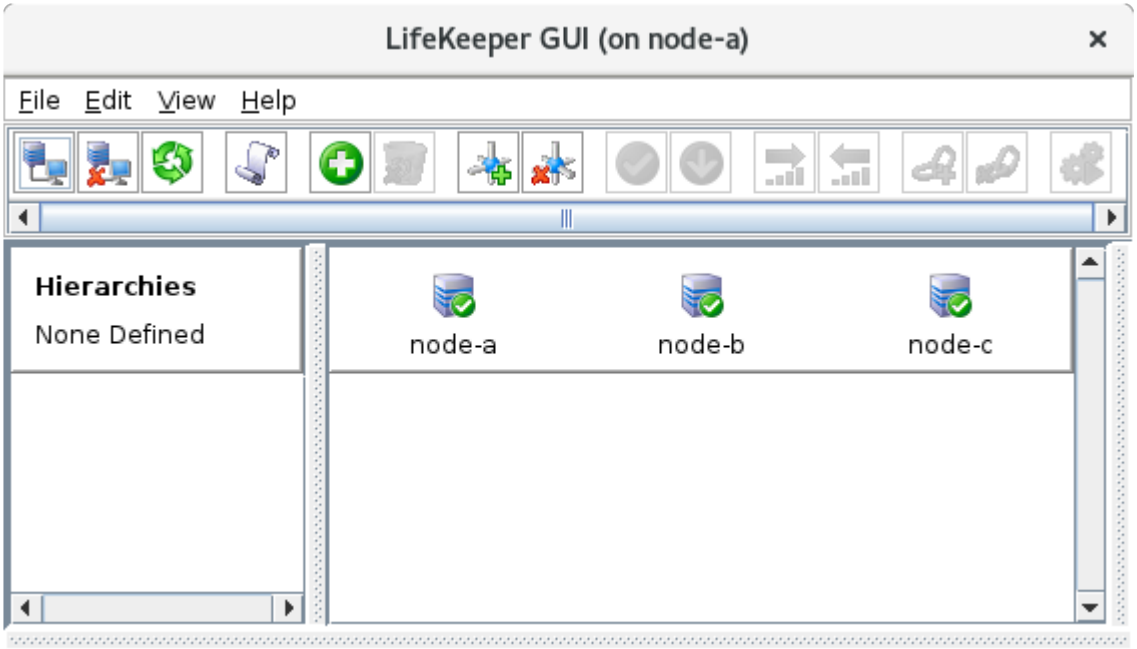
1. Select “Connect” on LifeKeeper (the first icon on the toolbar).




2. The login dialog is displayed again. Enter node-b and the root username and password to connect to the second node.




3. Connect to `node-c` the same way. Now you have three nodes displayed in the LifeKeeper GUI.




<-- node-c: Adding app/res: scsi device

# 11.2.6.5. Define Communication Paths

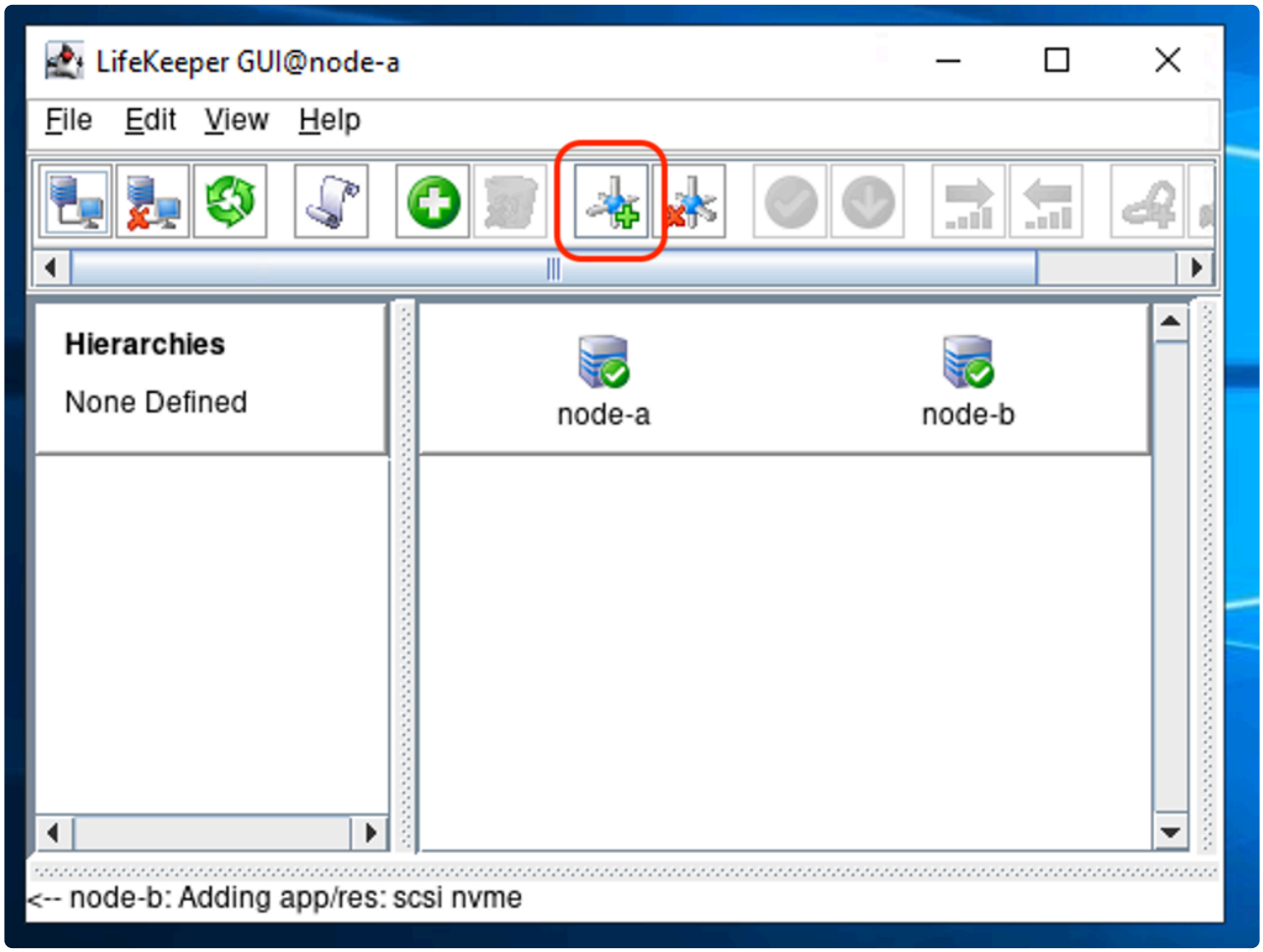
A Communication Path is a mechanism that allows the three nodes to communicate with each other and to transfer the data (for data replication purposes) between nodes. In a typical setup and especially in an on-premise environment, multiple communication paths are required to avoid a single point of failure. In this example, the network card in each node, or network routers switching between these nodes can be a point of failure.

In a cloud environment, however, having one communication path between nodes may be good enough as the underlying network layer is abstracted (we don't have visibility of these layers).

The next step is to define communication paths between all pairs of nodes.

> **Important Note:** In a cluster consisting of three or more nodes, communication paths must be defined between **all pairs** of nodes. In a three-node cluster, this means that three communication paths must be defined (node-a ↔ node-b, node-a ↔ node-c, and node-b ↔ node-c).

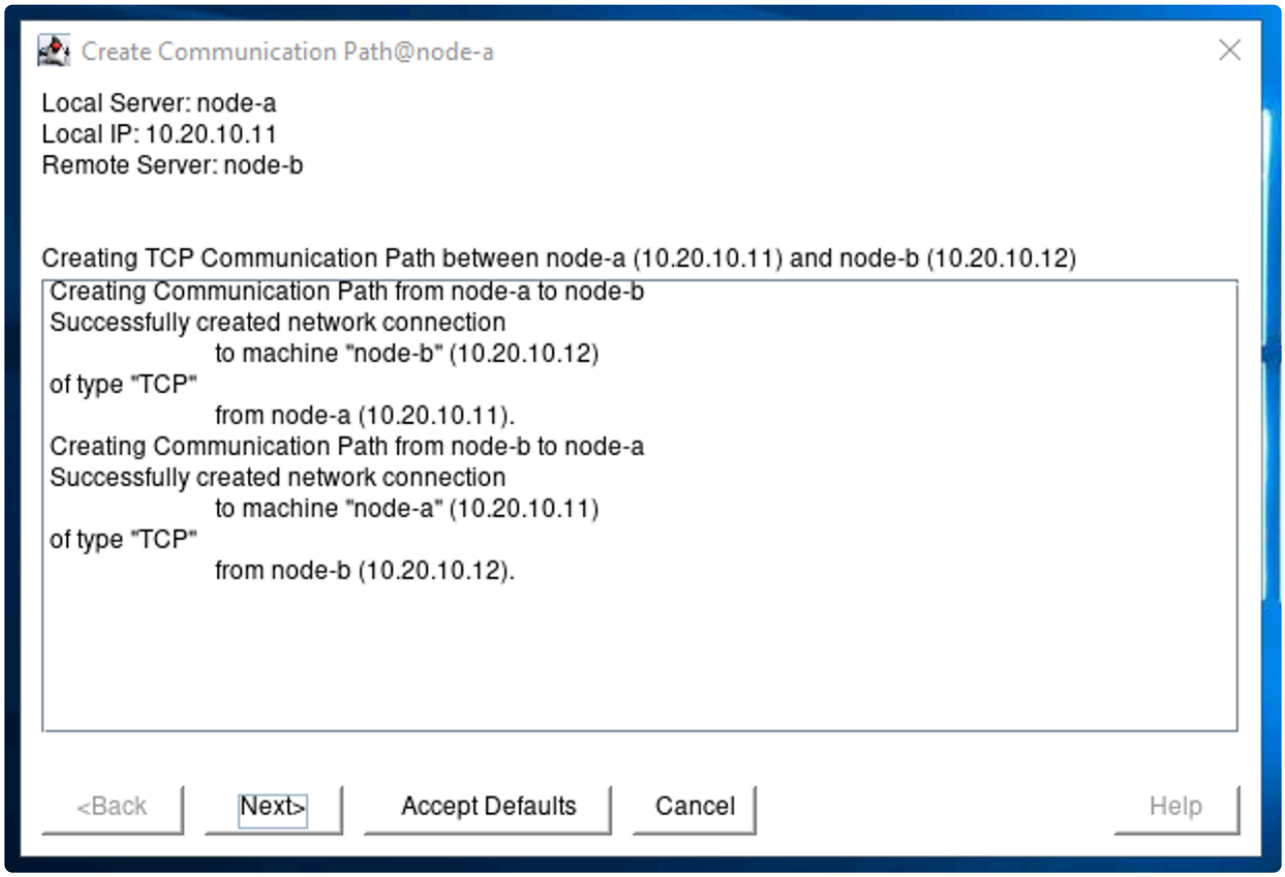
1. Click "Add communication path".



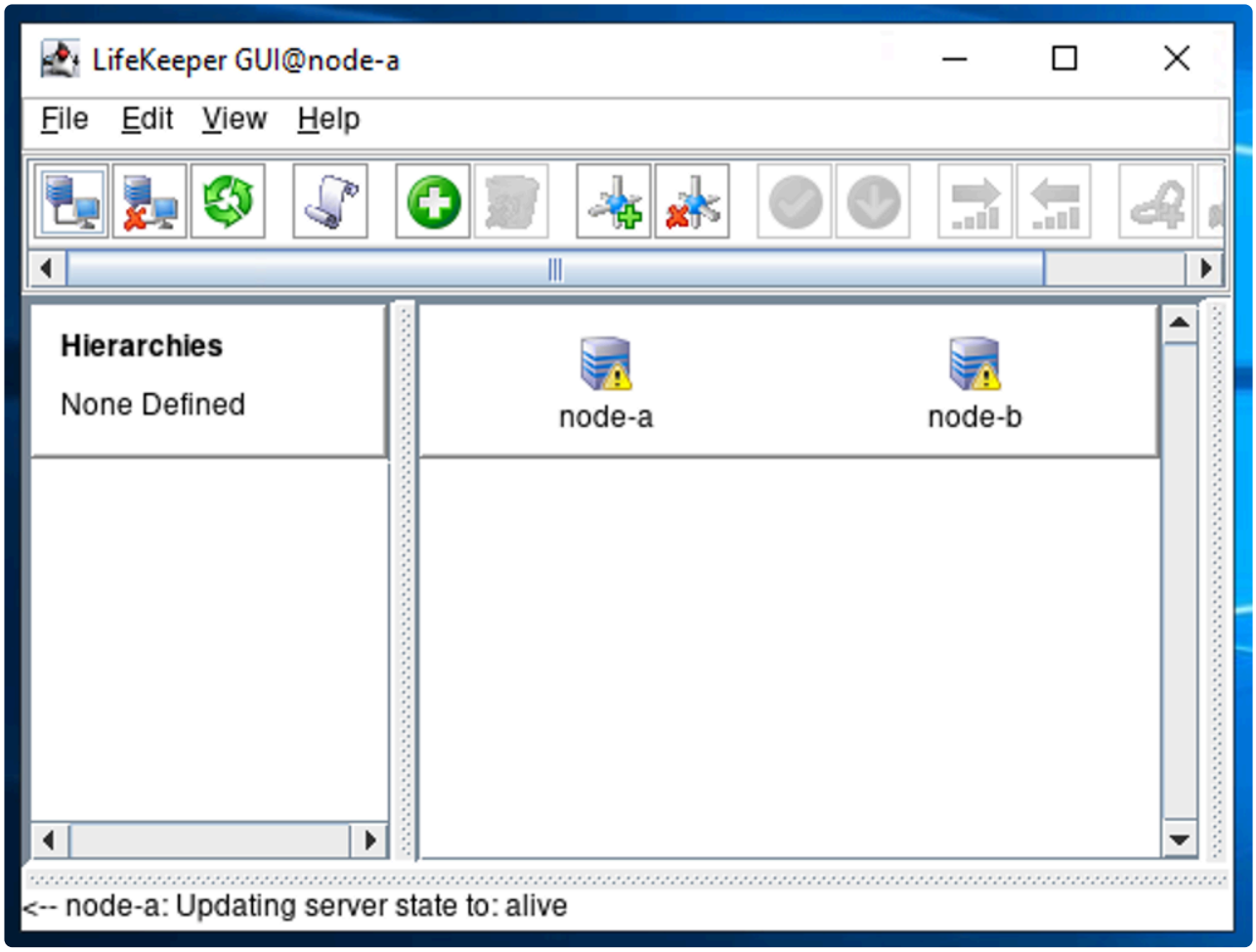

2. Select the following values. In most cases, these values should already be filled and selected.

| Field | Value |
|---|---|
| Local Server | node-a |
| Remote Server(s) | node-b |
| Device Type | TCP |
| Local IP Address(es) | 10.20.1.10 |
| Remote IP Address | 10.20.2.10 |
| Priority | 1 |

3. The summary of what has been configured will be displayed. Select “Next”.



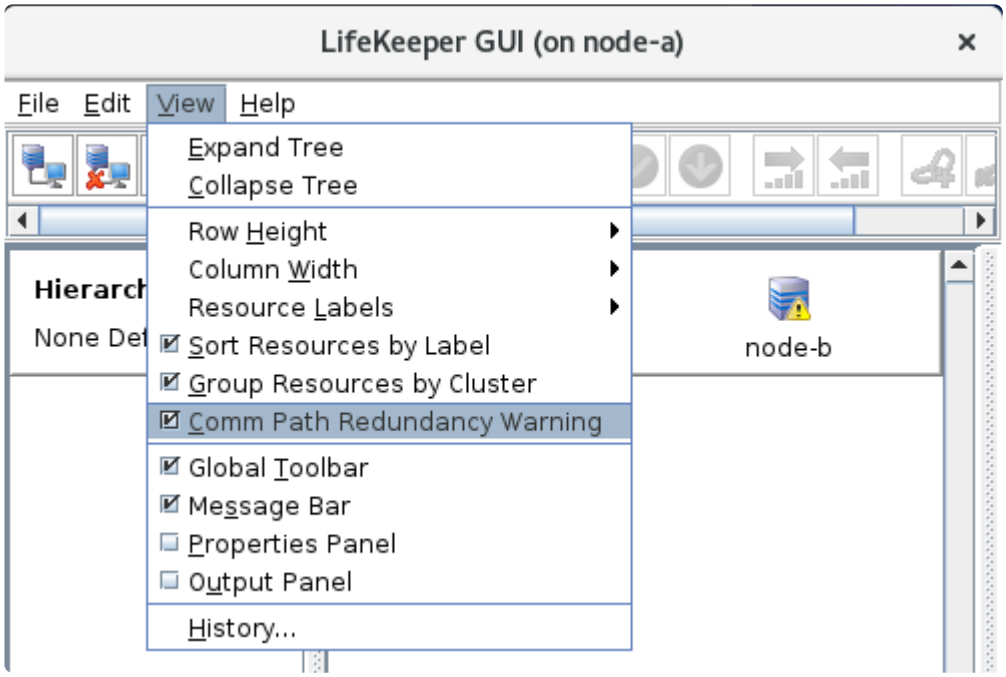

4. A communication path has now been defined.



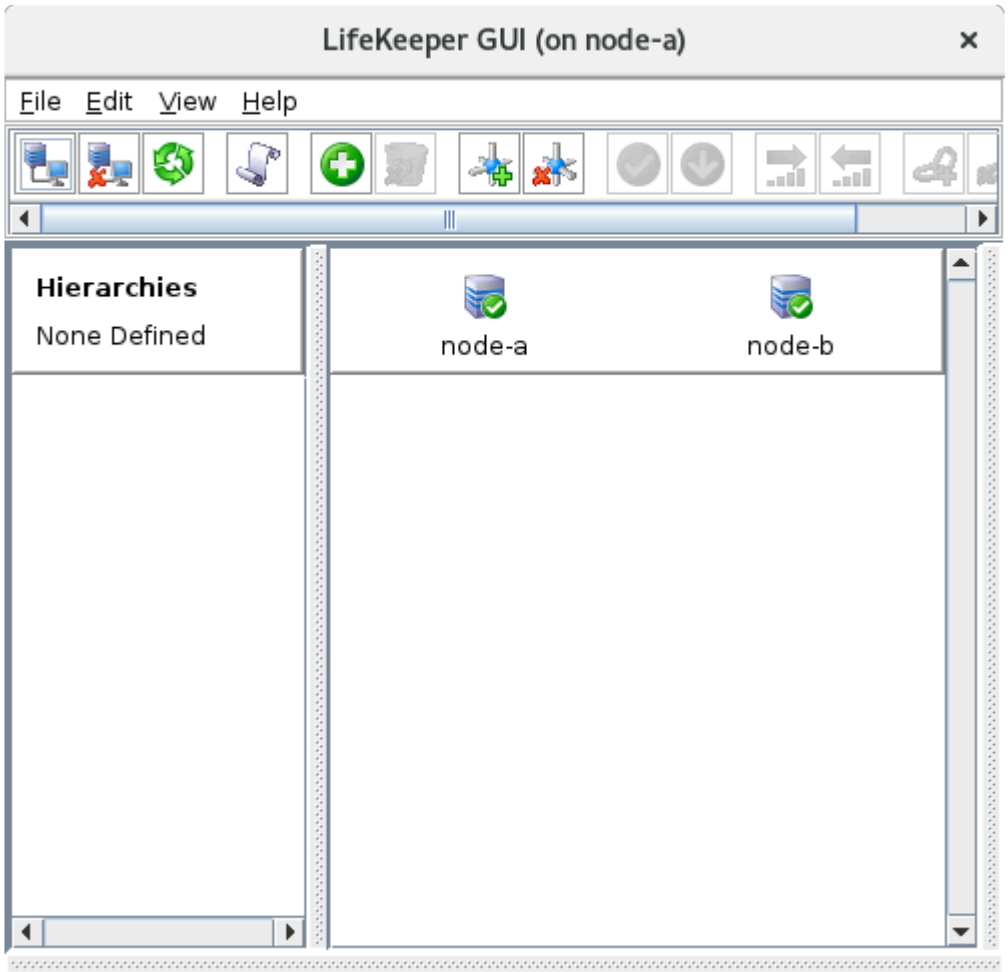

Once the communication path has been defined, you may notice a yellow triangle ( ) on each node. This is because we have only defined a single communication path. If a second communication path is defined, this changes to a green checkmark ( ).

## Turn off the Comm Path Redundancy Warning

This warning may be ignored if you are deploying in the cloud. The warning sign can be turned off by unchecking the “Comm Path Redundancy Warning” check box in the View menu of the LifeKeeper GUI.




Once unchecked, the icon will change from to .




<-- node-b: Updating server state to: alive

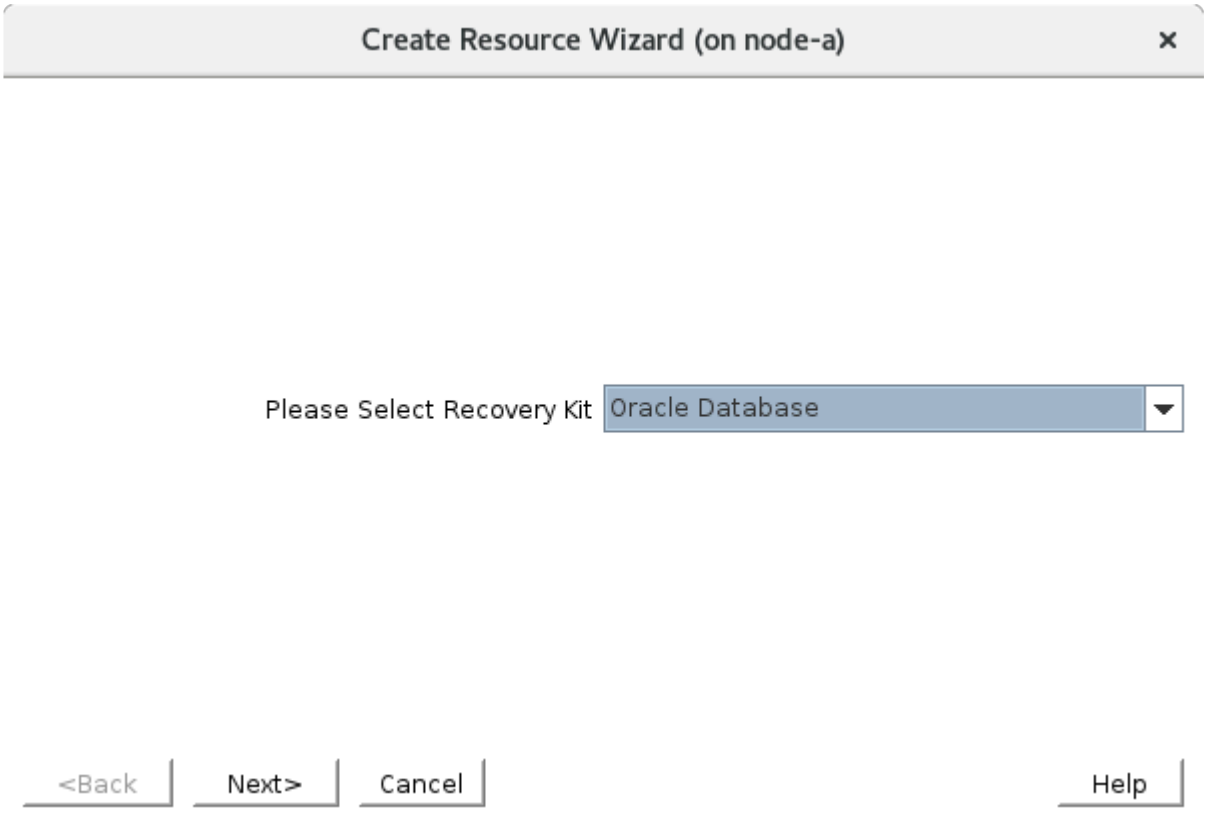
# 11.2.7. Protecting Our Resources

## A Typical Resource Protection Scenario with LifeKeeper

With LifeKeeper, most of the configuration can be completed using wizards. The wizard collects information about the resource(s) that are available for protection, and these resources can be protected by answering a series of simple questions. In most cases it is only necessary to select the default parameters. A standard example of LifeKeeper resource creation is given below.

1. Create Resource Wizard

    a. Select Resource Type (IP Resource, Data Replication, Oracle, etc.)




    b. Select Switchback Type

    c. Select the “active” server (where the application is currently running)

    d. Select the IP address to protect, configuration file location, and disk to protect

    e. Select the Resource Tag (name)

    f. Review the selections

2. Pre-Extend Wizard

    a. Select the target server (which will take over as the application host in the event of

switchover or failover)

b. Select Switchback Type

c. Select the Priority

d. Review the selections

3. Extend the Resource Hierarchy

   a. Select the IP address of the target, configuration file location, and disk to replicate data to

   b. Select the Resource Tag (name)

   c. Review the selections

4. Hierarchy Integrity Verification

   Once the configuration is completed, the wizard also automatically defines the dependencies between resources (for example, to protect a database resource it is necessary to mount the underlying disk(s) before starting the database).

# Switching Between Nodes

We are now ready to protect our resources.

As discussed earlier, cloud environments work differently than on-premise environments. If you are using a cloud environment, please read through Switching between Nodes in a Cloud Environment for relevant information about routing traffic to the active node in a cloud environment. We will configure the necessary resources after creating the IP resource, and the parameters selected for the IP resource in the next section will vary based on the cloud environment used.

If working in an on-premise environment, simply going through the following page is sufficient.

- Creating an IP Resource

If working in a cloud environment, select the relevant scenario from the following page:

- Switching between Nodes in a Cloud Environment

Once the IP resource is protected, it is recommended that you initiate a switchover (where the “standby” node becomes the “active” node) to test the functionality.

- Switch to Standby Node to Confirm Switchover is Working

# Setup Disk Replication

In most scenarios, data replication should be defined between nodes (especially when evaluating in the cloud). Follow the steps below to define data replication between nodes.

- How does Data Replication between Nodes Work?
- How to Create Data Replication of a File System

# Protect our Resources

Once the basic preparation is completed, you can now protect resources such as databases and applications.

- Protecting an Oracle Resource
- Protecting MSSQL Using Quick Service Protection
- Protecting a PostgreSQL Resource
- Protecting an NFS Resource
- Protecting SAP Resources
- Protecting SAP HANA Resources

# 11.2.7.1. Creating an IP Resource

The first resource to be created on LifeKeeper should be an IP resource. As discussed earlier, a client can find the Active Node using a Virtual IP Address. Below is the list of IP addresses that are referenced in this section.

## IP Addresses for Nodes

| Field | Value |
|---|---|
| node-a | 10.20.1.10 |
| node-b | 10.20.2.10 |

## IP Address for the IP Resource Based on the Scenario

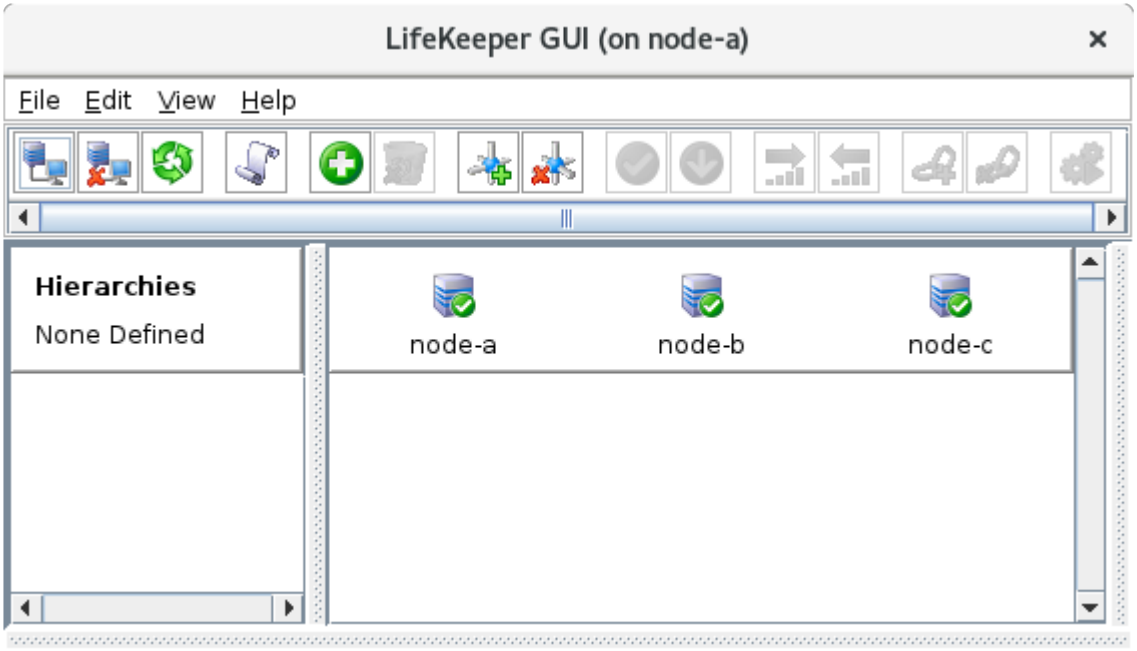
In a cloud environment, please review Switching between Nodes in a Cloud Environment to identify the most applicable scenario before selecting an IP address.

<table>
<tr><th>Platform</th><th>Scenario</th><th>How to Select an IP Address</th><th>Sample IP Addresses Used in this Guide</th><th>Default Name of the IP Address</th></tr>
<tr><td rowspan="3">AWS</td><td>Route Table Scenario</td><td>IP address that is outside of the VPC CIDR</td><td>10.10.10.10</td><td>ip-10.10.10.10</td></tr>
<tr><td>Route 53 Scenario</td><td>IP address that is assigned to the “active” node</td><td>10.20.1.10</td><td>realip</td></tr>
<tr><td>Elastic IP Scenario</td><td colspan="3" rowspan="2">Not required to create an IP Resource. Skip this page.</td></tr>
<tr><td colspan="2">GCP</td></tr>
<tr><td colspan="2">Azure</td><td rowspan="2">An available IP address on the subnet</td><td rowspan="2">10.20.0.10</td><td rowspan="2">ip-10.20.0.10</td></tr>
<tr><td colspan="2">On-premise environment</td></tr>
</table>

## Configure an IP Resource

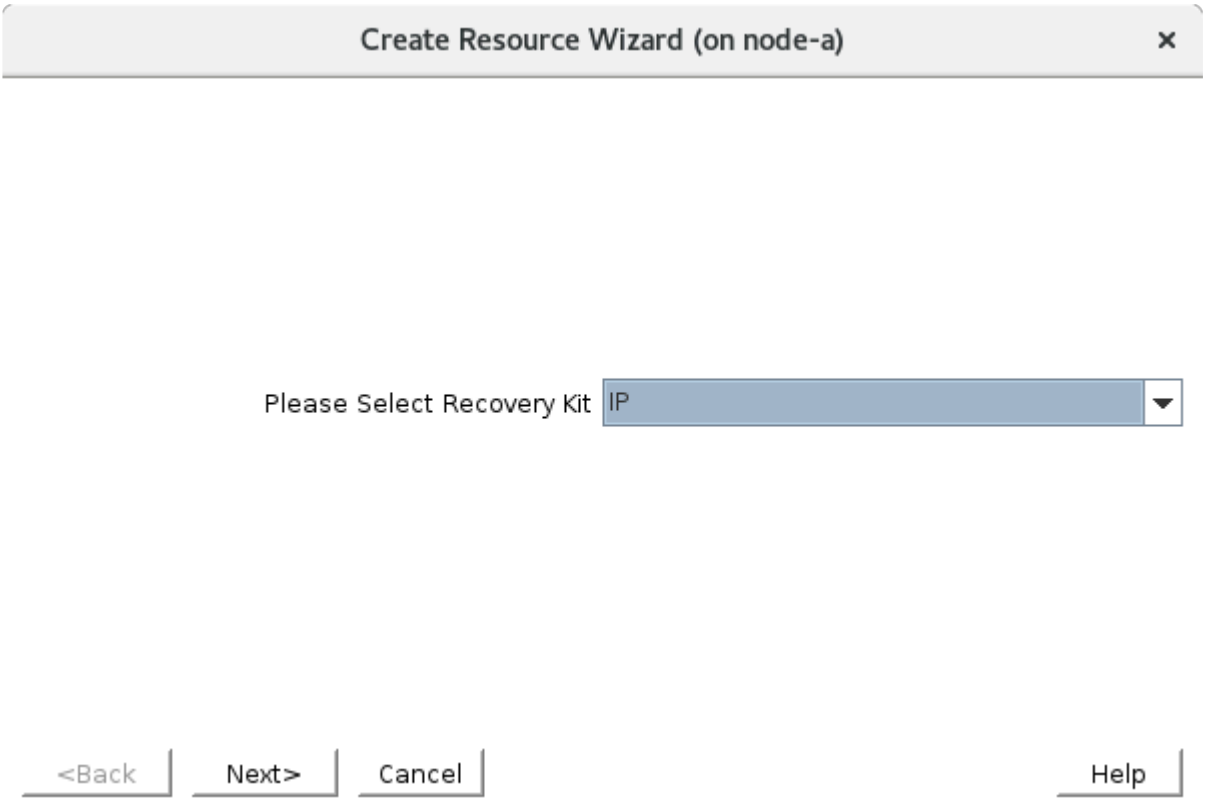
This example uses 10.10.10.10 as the IP address to be used for “Route Table Scenario“ in AWS.

1. Click the button in the LifeKeeper User Interface (or select Edit > Server > Create Resource Hierarchy).




<-- node-c: Adding app/res: scsi device

2. The Create Resource Wizard (on node-a) will be displayed and the type of resource to be protected can be selected. Select “IP” and then click “Next>”.




3. Select the default values. The only field you need to enter is the virtual IP address `10.10.10.10` in the IP Resource field.

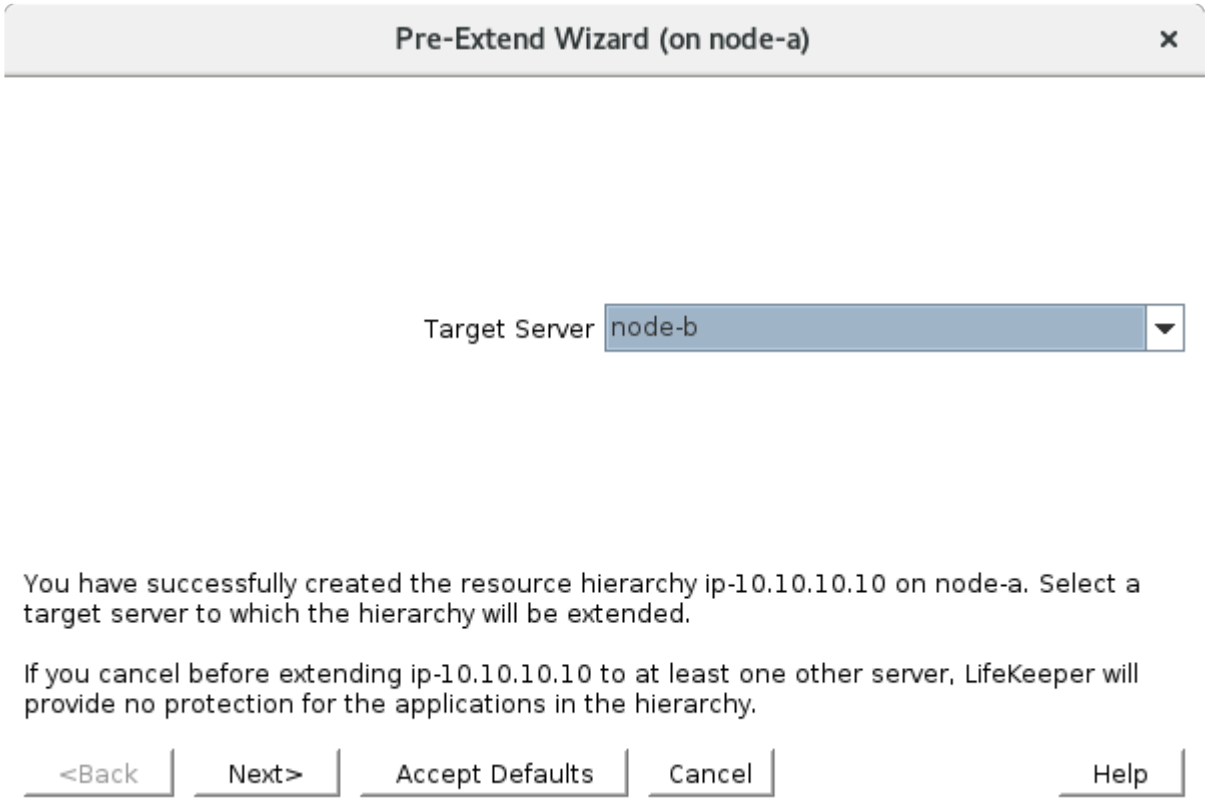
**Note:** For items with a checkmark ( ), review and use the default values.

| Field | Value |
|---|---|
| Switchback Type | Intelligent |
| Server | node-a |
| IP Resource | 10.10.10.10 |
| Net Mask | 255.255.255.0<br>Note: This value may vary based on the network. |
| Network Interface | eth0 |
| IP Resource Tag | ip-10.10.10.10 |

Values such as Net Mask are displayed based on how the local system is configured. Therefore the default values seen may vary.

4. Once you confirm the resource is successfully created, click “Next>”.

5. The next step is the “Pre-Extend Wizard (on node-a)”.



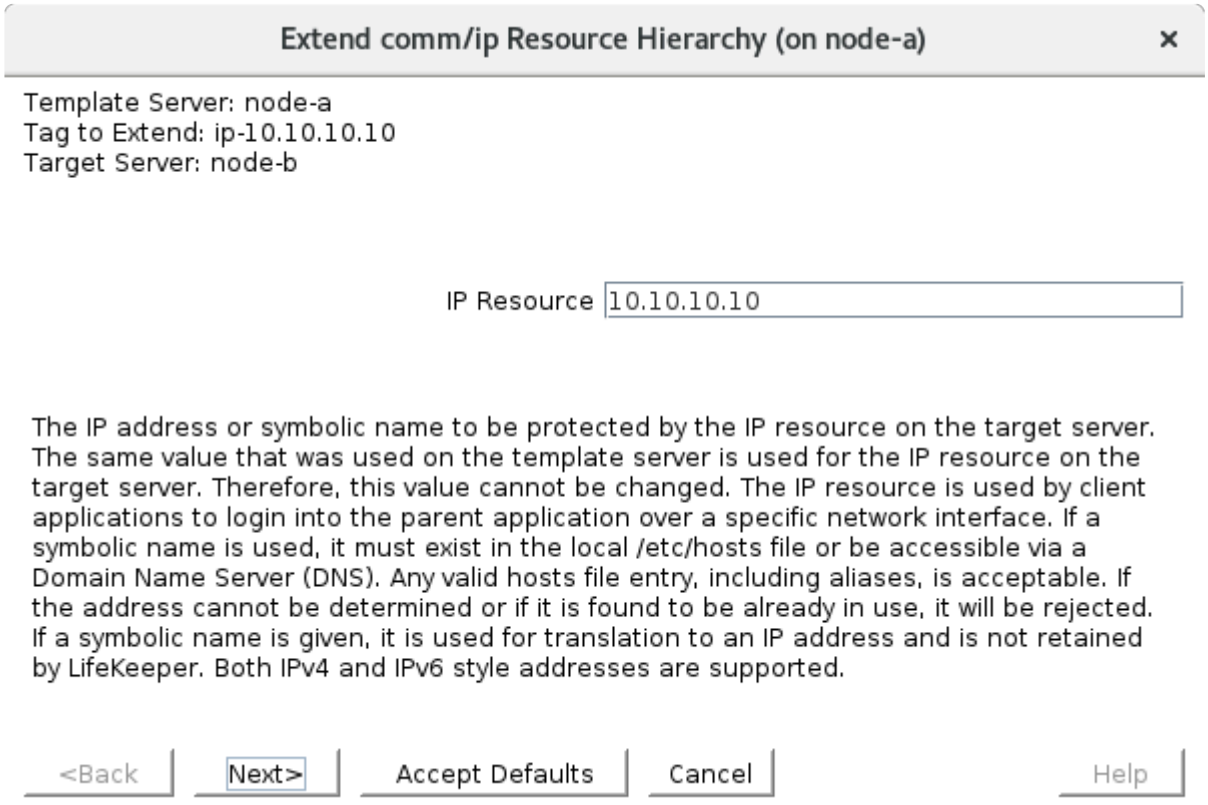

Select all of the defaults as shown below.

| Field | Value |
|---|---|

| | |
|---|---|
| Target Server | node-b |
| Switchback Type | Intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

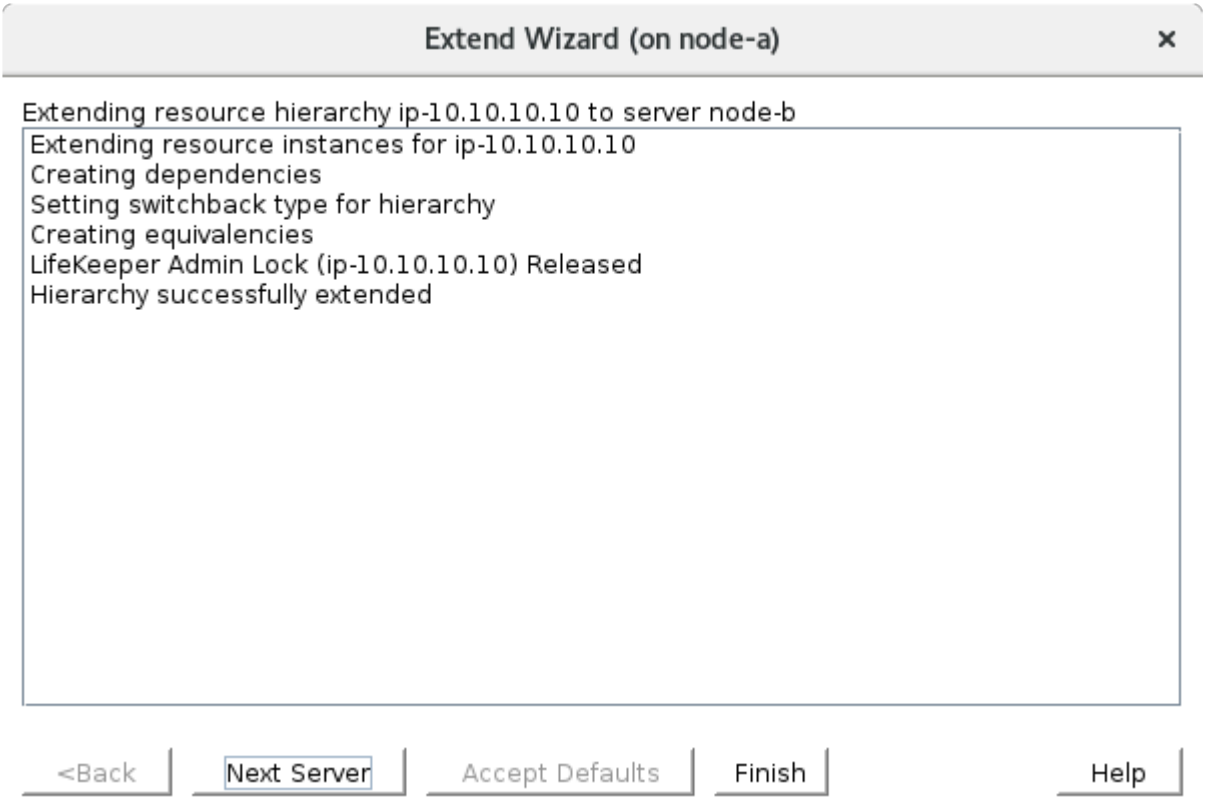
The Pre-Extend check should be successfully completed. Click “Next”.

6. The next step is “Extend comm/IP Resource Hierarchy (on node-a)”. Click “Next” and fill in the fields.



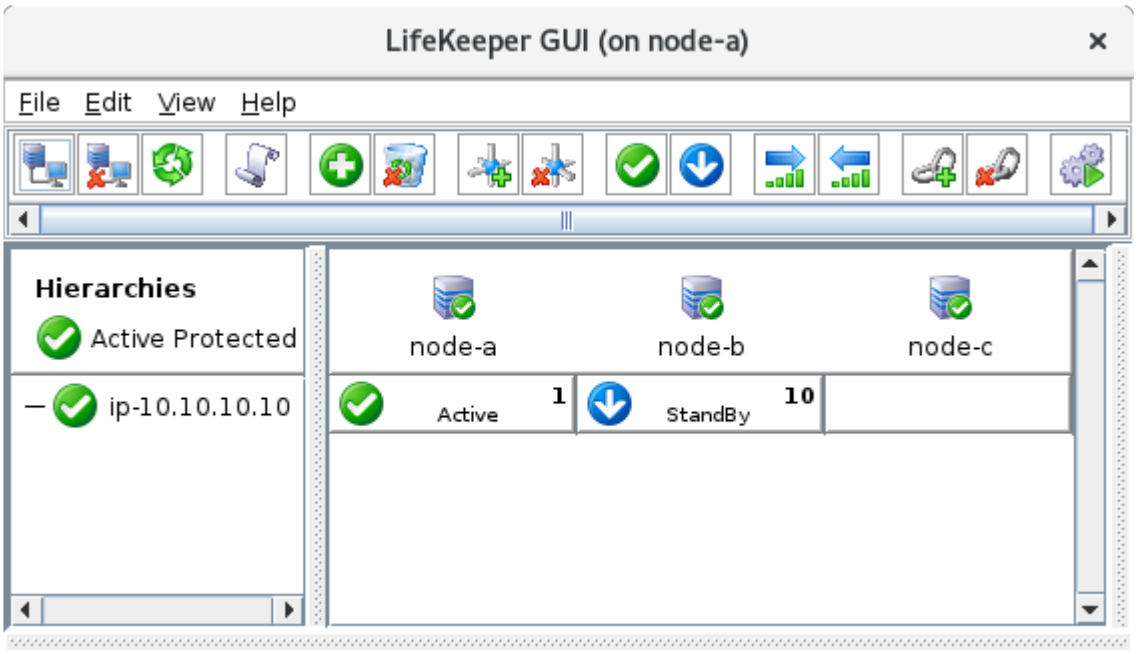

| Field | Value |
|---|---|
| Net Mask | 255.255.255.0<br>Note: This value may vary based on the network. |
| Network Interface | eth0 |
| IP Resource Tag | ip-10.10.10.10 |

7. Now the resource is extended. Select “Finish”.




8. The verification is now complete and the wizard can be closed.

9. The IP Resource (ip-10.10.10.10) will be displayed on the LifeKeeper User Interface.




<-- node-b: ip-10.10.10.10: Updating equivalency list

As we have seen, resources may be easily protected by answering the questions in the resource creation wizard. Most of the values are prepopulated based on an understanding of the environment and therefore minimal user interaction is required.

# 11.2.7.2. Switching between Nodes in a Cloud Environment

If you are operating in a cloud environment, there is an extra step to switch a virtual IP address between nodes. The following topics explain the background information and general approaches for each cloud environment:

- AWS Route Table Scenario
- AWS Elastic IP Scenario
- AWS Route53 Scenario
- Azure Internal Load Balancer Scenario
- GCP Internal Load Balancer Scenario

The following sections explain the configuration steps required in each scenario:

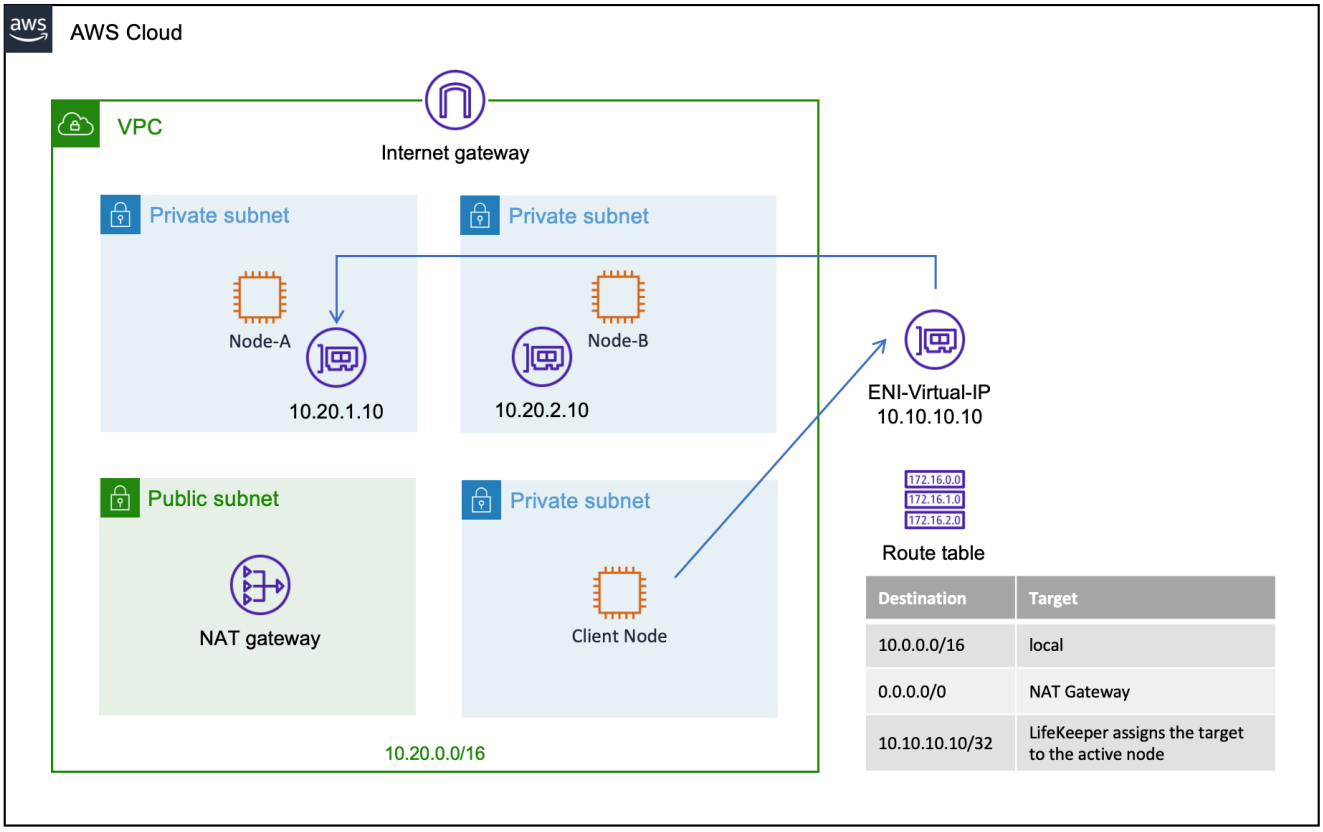
- Creating an AWS EC2 Resource (RouteTable Scenario)
- Creating an AWS EC2 Resource (Elastic IP Scenario)
- Creating an AWS Route53 Resource
- Azure – Using an Internal Load Balancer
- GCP – Using an Internal Load Balancer

SIOS Protection Suite also provides a resource type which will respond to TCP health check probes from an internal load balancer in order to route traffic to the current application host:

- Responding to Load Balancer Health Checks

# 11.2.7.2.1. Creating an AWS EC2 Resource (RouteTable Scenario)

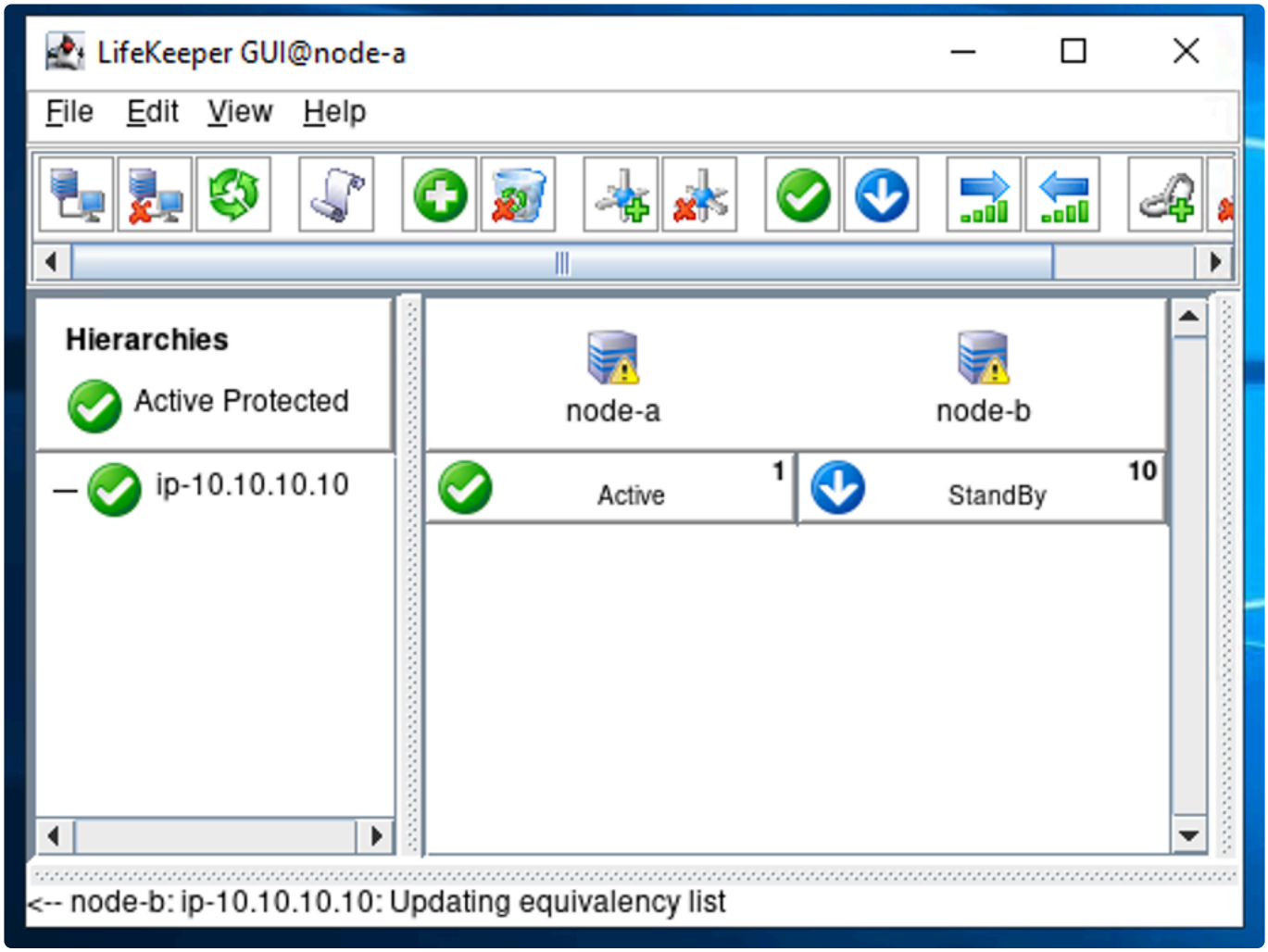
Here we will discuss how the Route Table can be used to switch between nodes by creating an entry in the Route Table that points to the active node.




## Create an IP Resource

Now you can create an IP resource. Note that the IP address of the resource must be located outside of the VPC. In this example, the IP resource is created using `10.10.10.10` (the VPC is defined as 10.20.0.0/16). Refer to Creating an IP Resource for more information.

Once the IP resource has been created, the LifeKeeper User Interface should look like this.




## Assign Permission to Update Route Table Entries to Instances

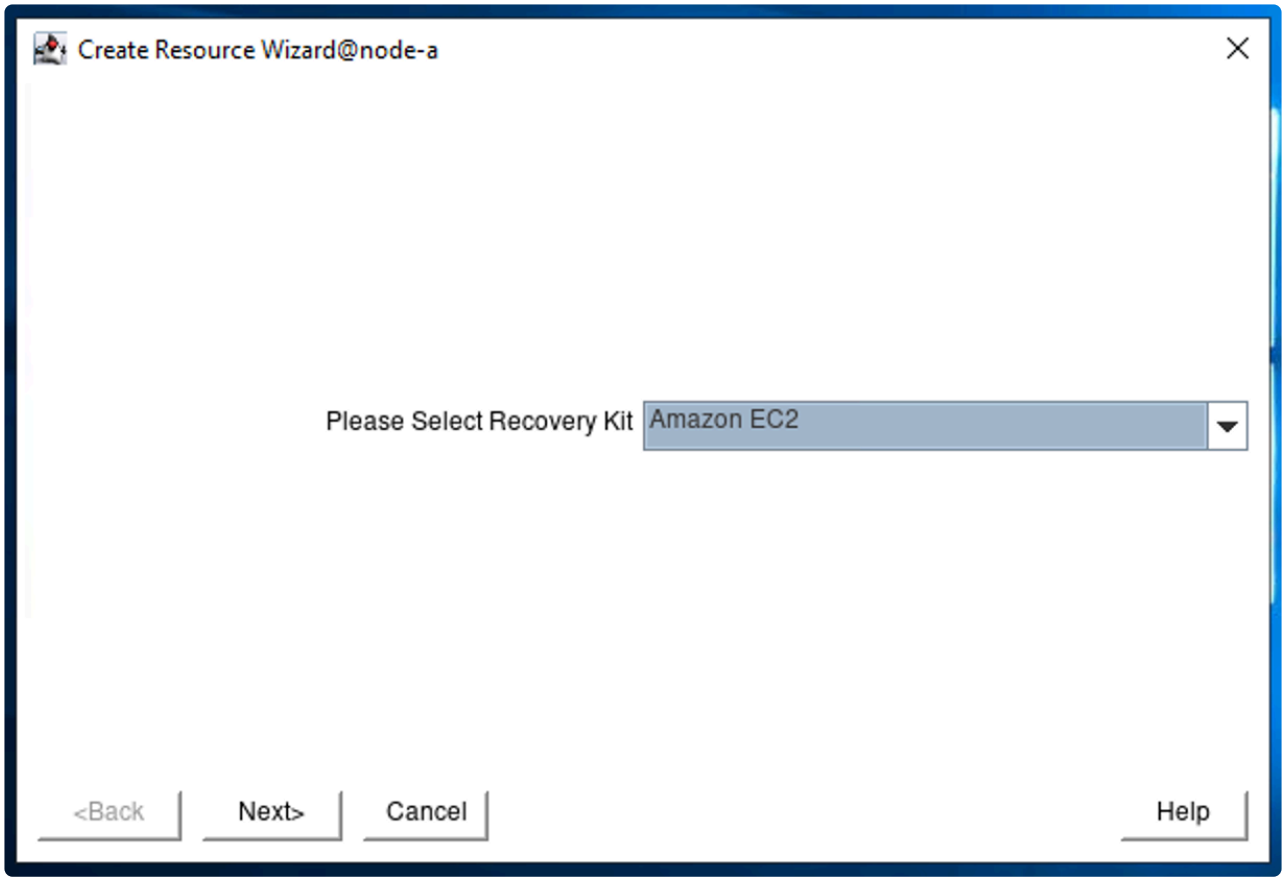
> ! To complete the steps below, the AWS CLI must be installed on each node.

- Install AWS CLI
- Assign Permission to Use EC2 Recovery Kit

## Creating an EC2 Resource (Route Table Scenario)

Creating an EC2 Resource should be straightforward once an IP Resource has been created.

1. Select the icon to start the “Create Resource Wizard”.
2. Select “Amazon EC2” as the Recovery Kit.




3. On the “Create Resource Wizard @ node-a”, specify the following values.

**Note:** For items with a checkmark ( ) review the default value and use the value suggested.

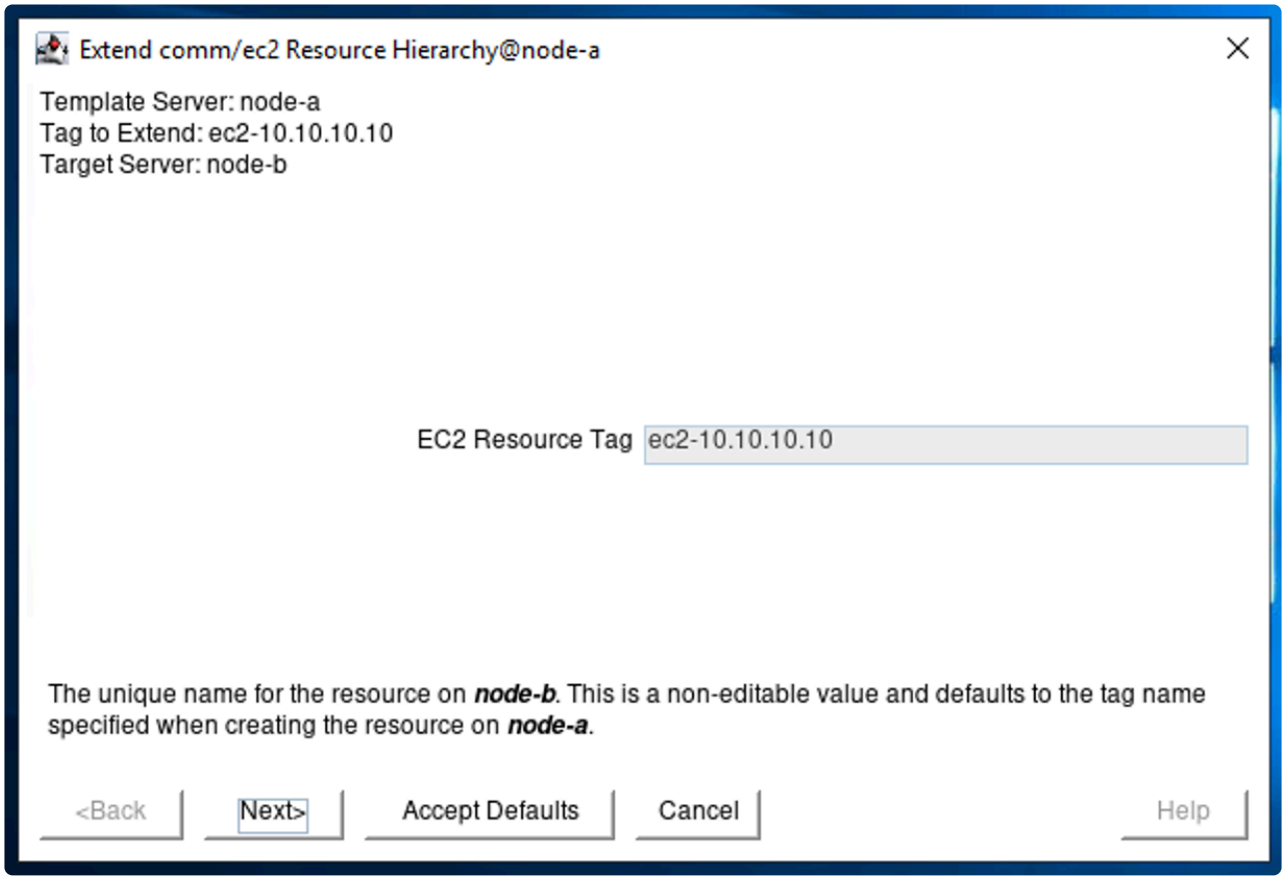
| Field | Value |
|---|---|
| Switchback Type | Intelligent |
| Server | node-a |
| EC2 Resource Type | Route Table (Backend Cluster) |
| IP Resource | ip-10.10.10.10 |
| EC2 Resource Tag | ec2-10.10.10.10 |

4. On the “Pre-Extend Wizard” specify the following values:

| Field | Value |
|---|---|
| Target Server | node-b |

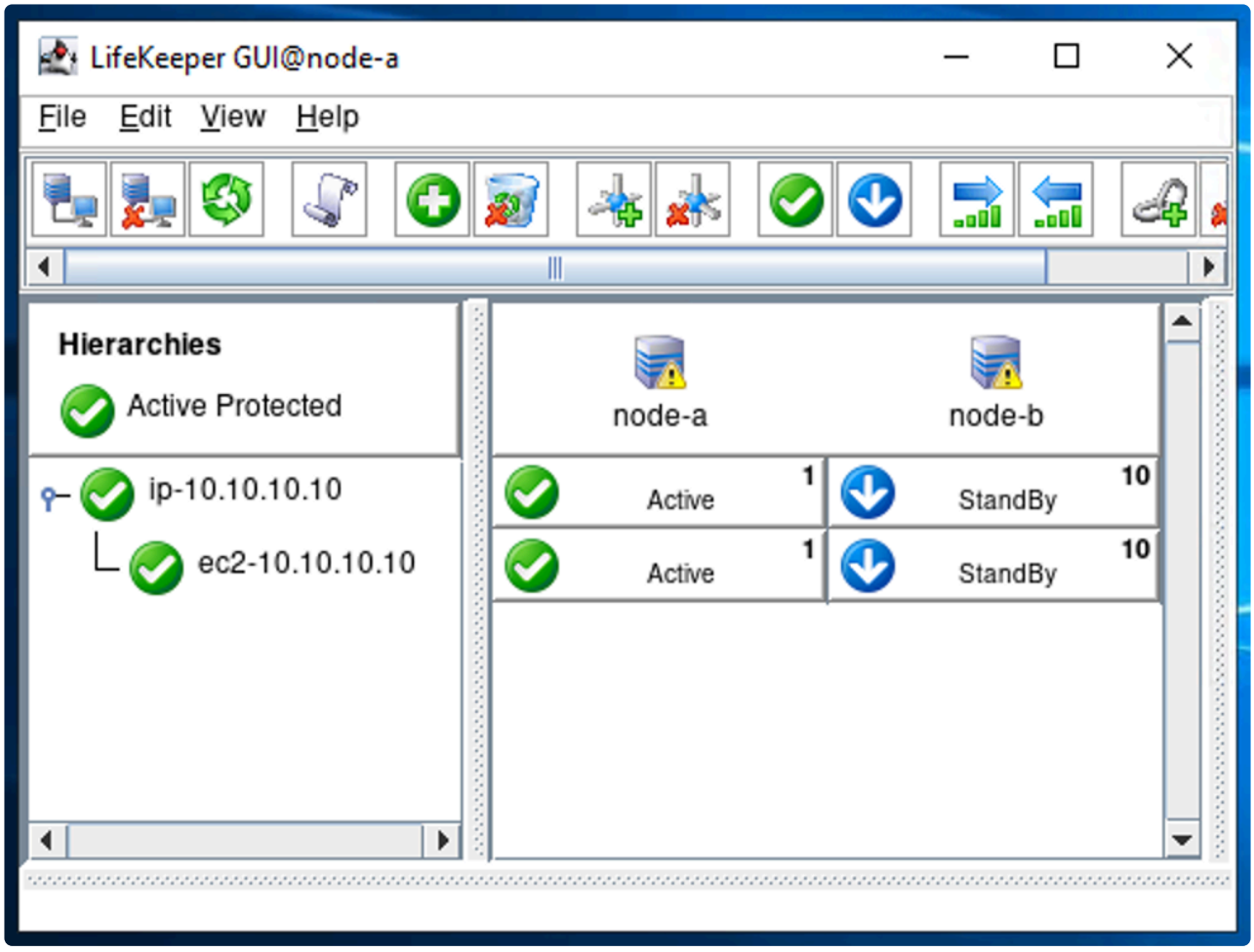
| | |
|---|---|
| Switchback Type | Intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

5. Once Pre-Extend is completed, move on to “Extend comm/ec2 Resource”. Select “ec2-10.10.10.10”.



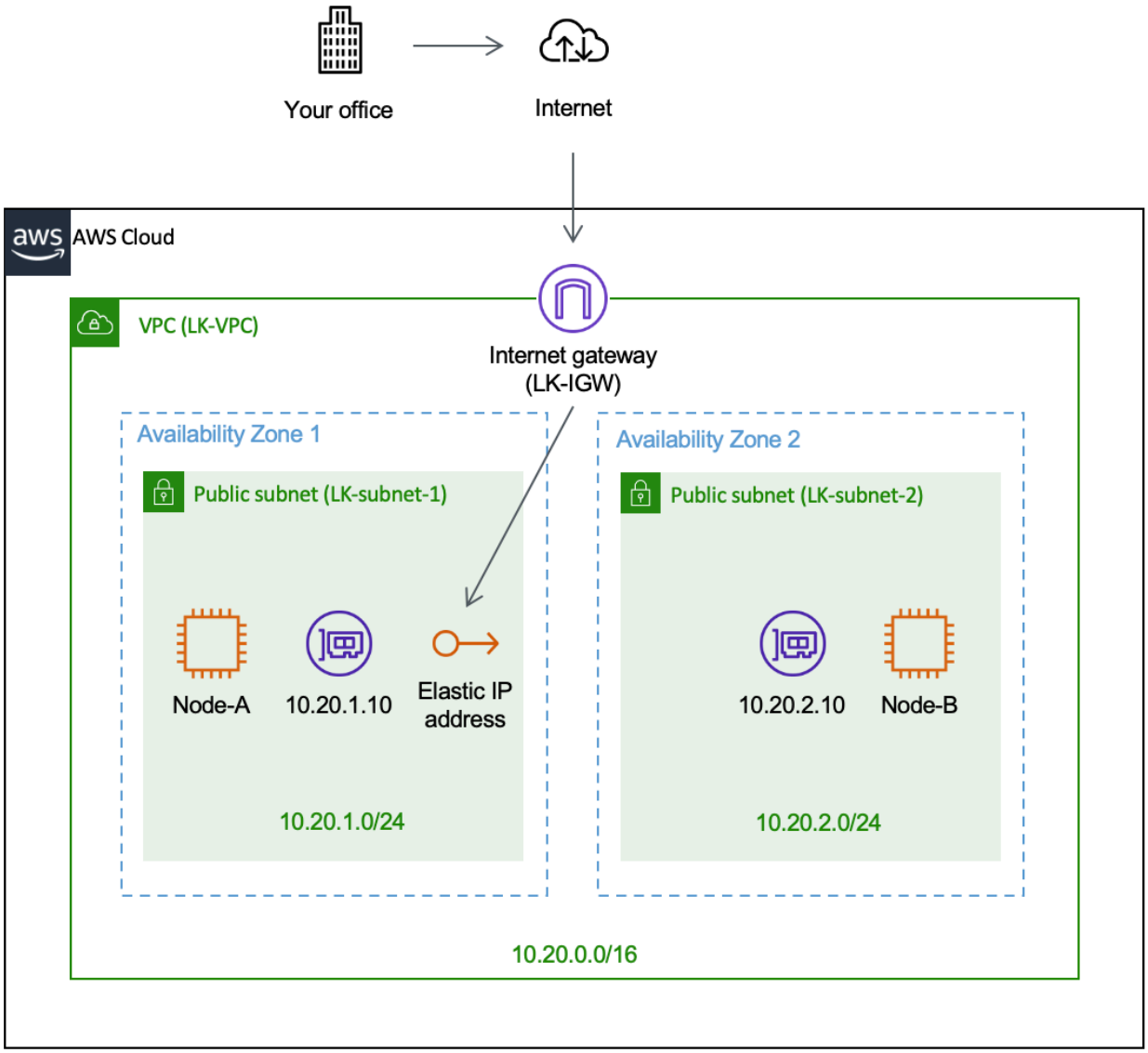

6. Once “Hierarchy successfully extended” is displayed, the creation of the resource is now complete.

7. The EC2 resource is created as shown below.


LifeKeeper GUI@node-a
File Edit View Help
Hierarchies
Active Protected
ip-10.10.10.10
ec2-10.10.10.10
node-a
node-b
Active 1
StandBy 10
Active 1
StandBy 10

# 11.2.7.2.2. Creating an AWS EC2 Resource (Elastic IP Scenario)

Here we will discuss how to switch between nodes using an Elastic IP address by creating an entry that points to the active node.




## Assign Permission to Update Route Table Entries to Instances

! To complete the steps below, the AWS CLI must be installed on each node.

- Install AWS CLI

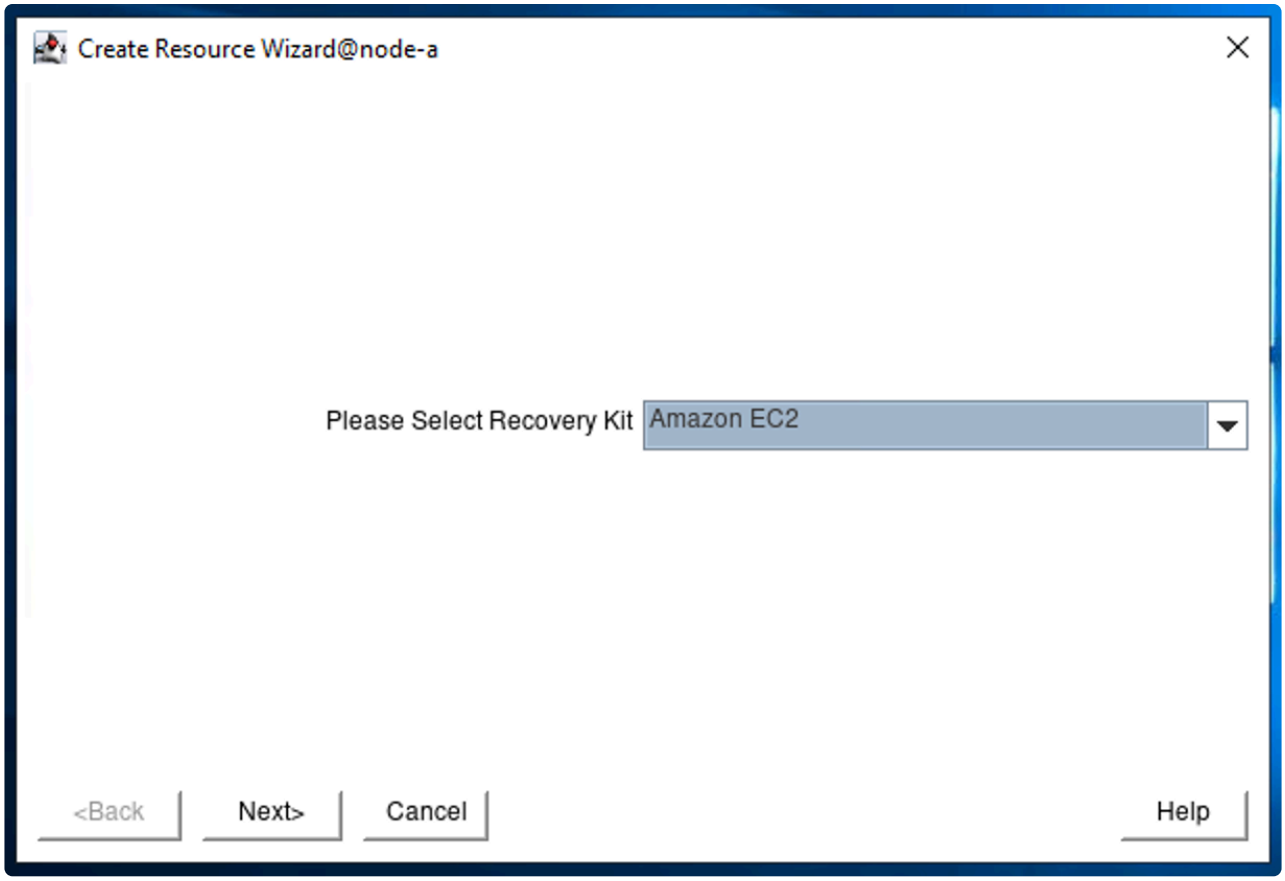
- Assign Permission to Use EC2 Recovery Kit

You will also need to acquire an Elastic IP. In the example below the Elastic IP address is 50.18.115.213. Note that this IP address is an example value.

# Creating an EC2 Resource (Elastic IP Scenario)

Complete the following steps to create an EC2 resource using the Elastic IP (Frontend Cluster) scenario.

1. Select “Amazon EC2” as the Recovery Kit.




2. On the “Create Resource Wizard @ node-a”, specify the following values.

**Note:** For items with a checkmark ( ✓ ) review the default value and use the value suggested.

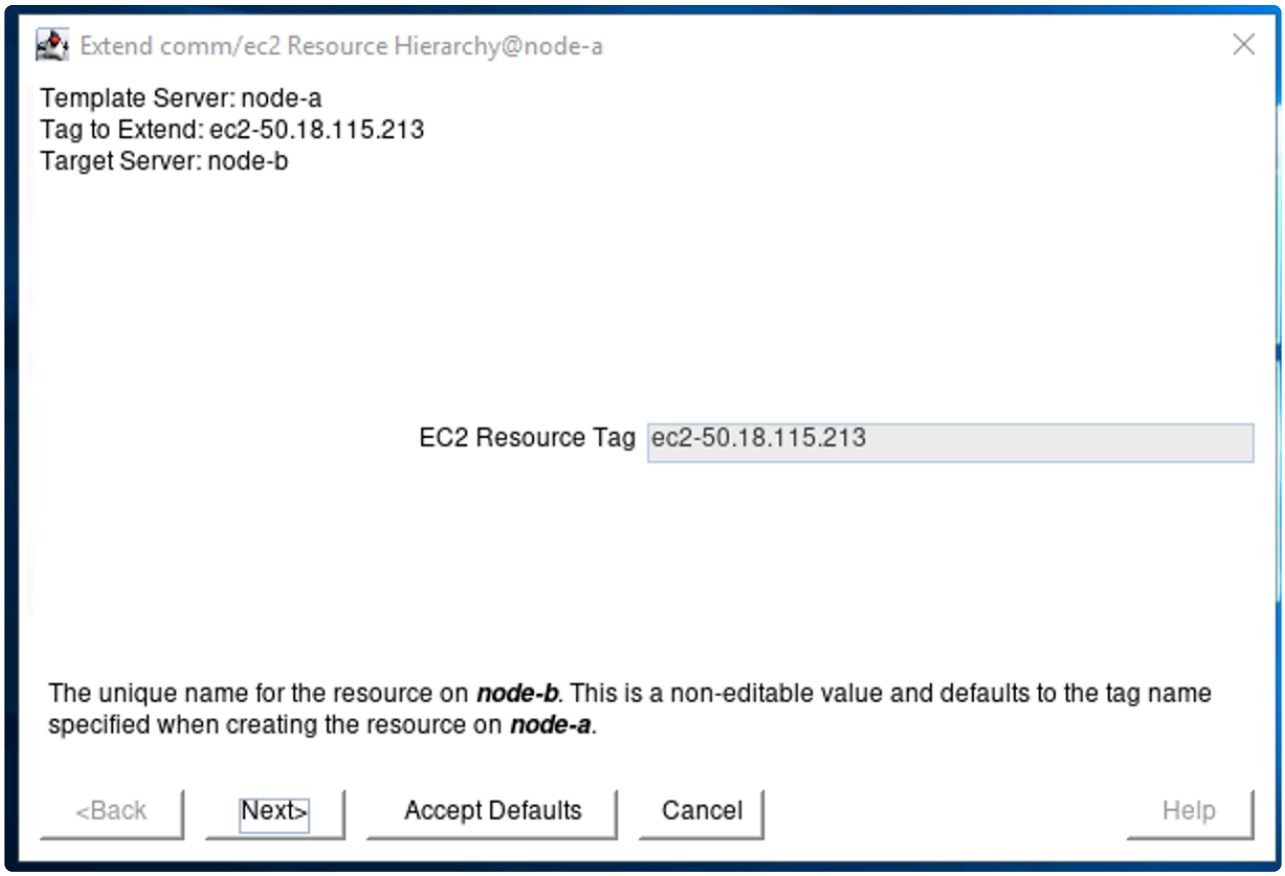
| Field | Value |
|---|---|
| Switchback Type | Intelligent ✓ |
| Server | node-a ✓ |
| EC2 Resource Type | Elastic IP (Frontend Cluster) |

| | |
|---|---|
| Network Interface | eith0 |
| Elastic IP | Select an available Elastic IP from the list.<br>For example, 50.18.115.213 |
| EC2 Resource Tag | ec2-50.18.115.213 |

3. On “Pre-Extend Wizard” specify the following values:

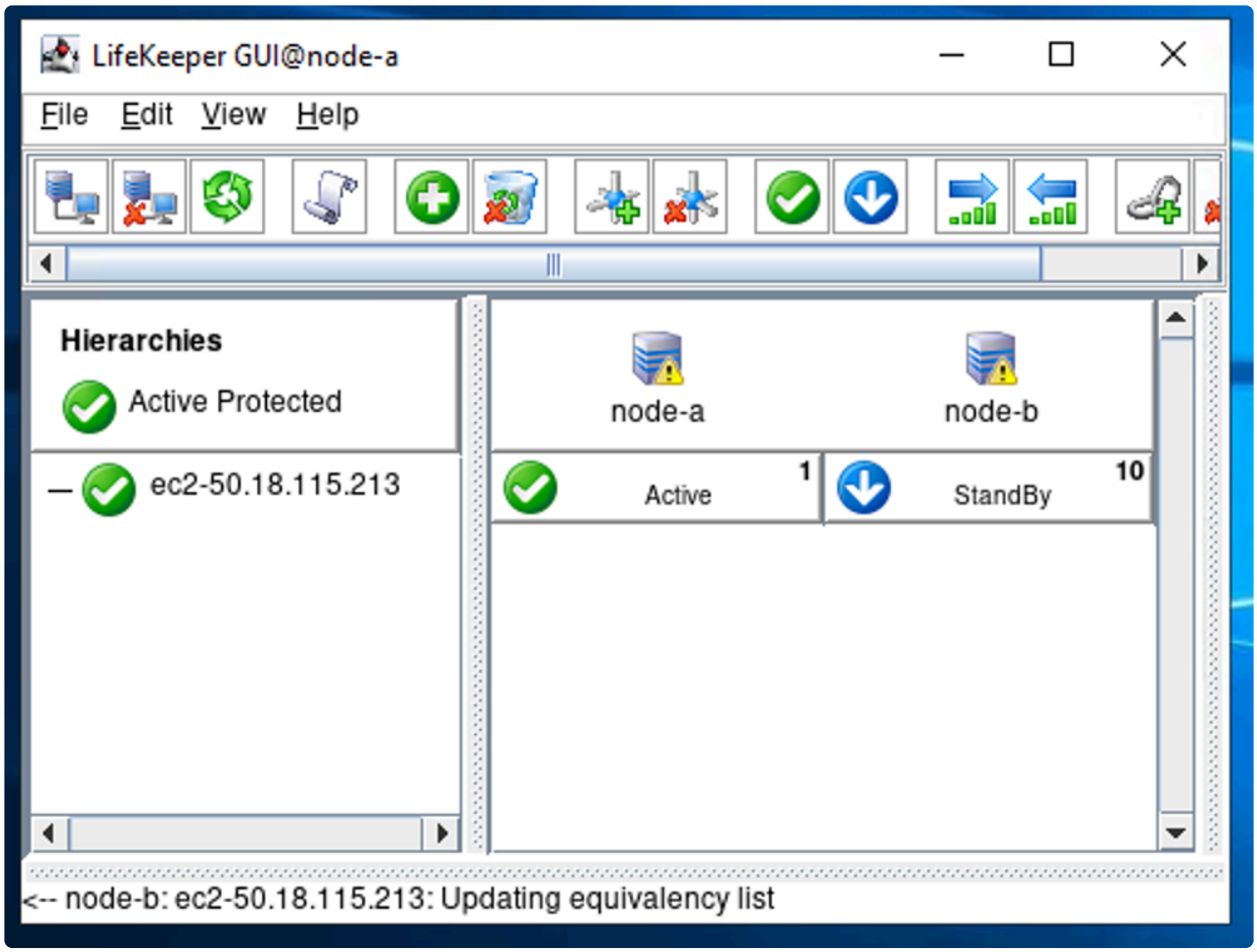
| Field | Value |
|---|---|
| Target Server | node-b |
| Switchback Type | Intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

4. Once Pre-Extend is completed, move on to “Extend comm/ec2 Resource”. Select “ec2-50.18.115.213”.



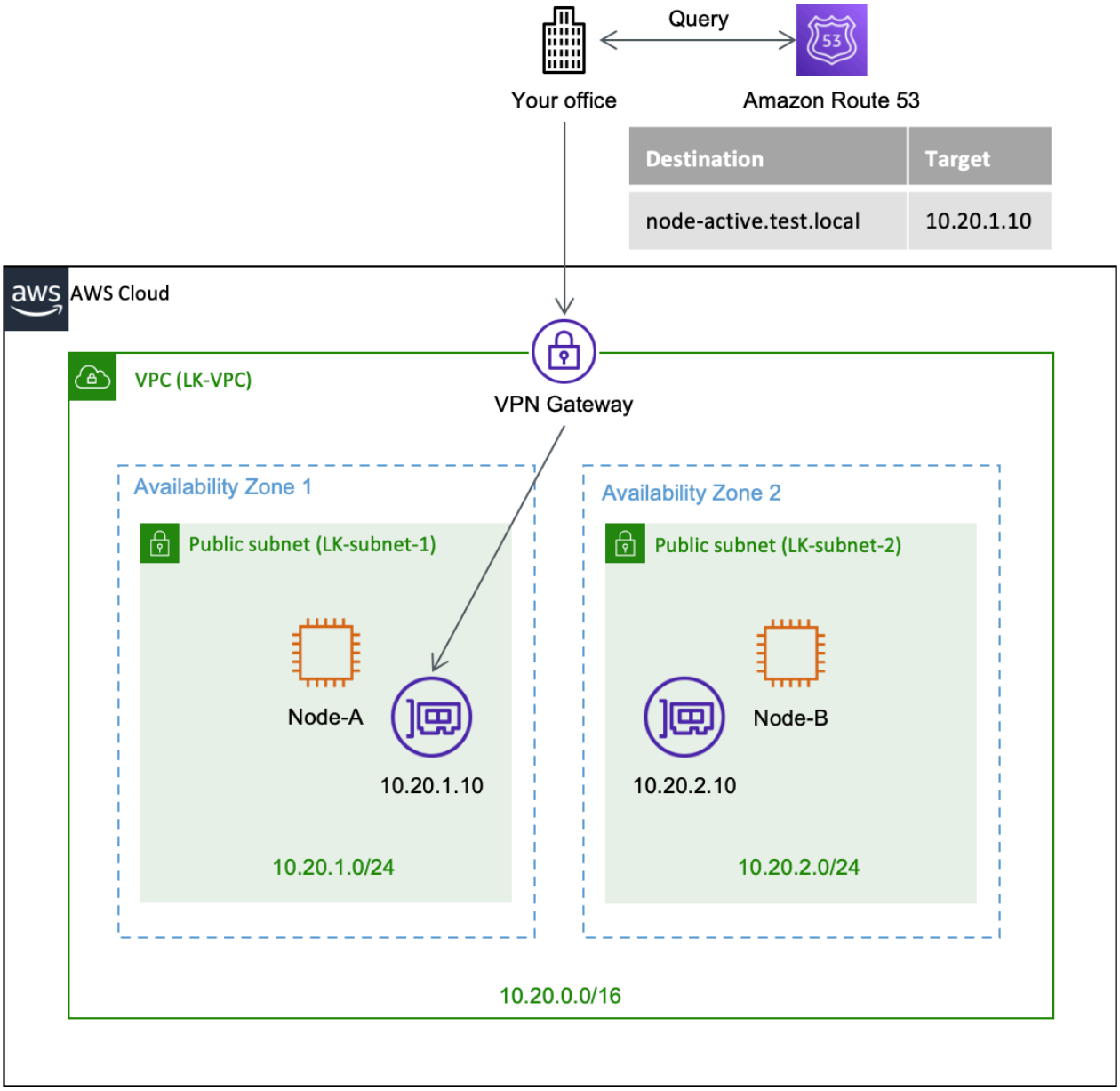

5. Once “Hierarchy successfully extended” is displayed, the creation of the resource is complete.

6. The EC2 resource is now created as shown below.



# 11.2.7.2.3. Creating an AWS Route53 Resource

Here we will discuss how to switch between nodes using Route53 by creating a DNS entry that points to the active node.




## Record Set on Route 53 to be Created

| Name | Type | Value |
|---|---|---|
| node-a | A-IPv4 Address | 10.20.1.10 |
| node-b | A-IPv4 Address | 10.20.2.10 |

| | | |
|---|---|---|
| node-active | A-IPv4 Address | 10.20.1.10 (same as node-a) |

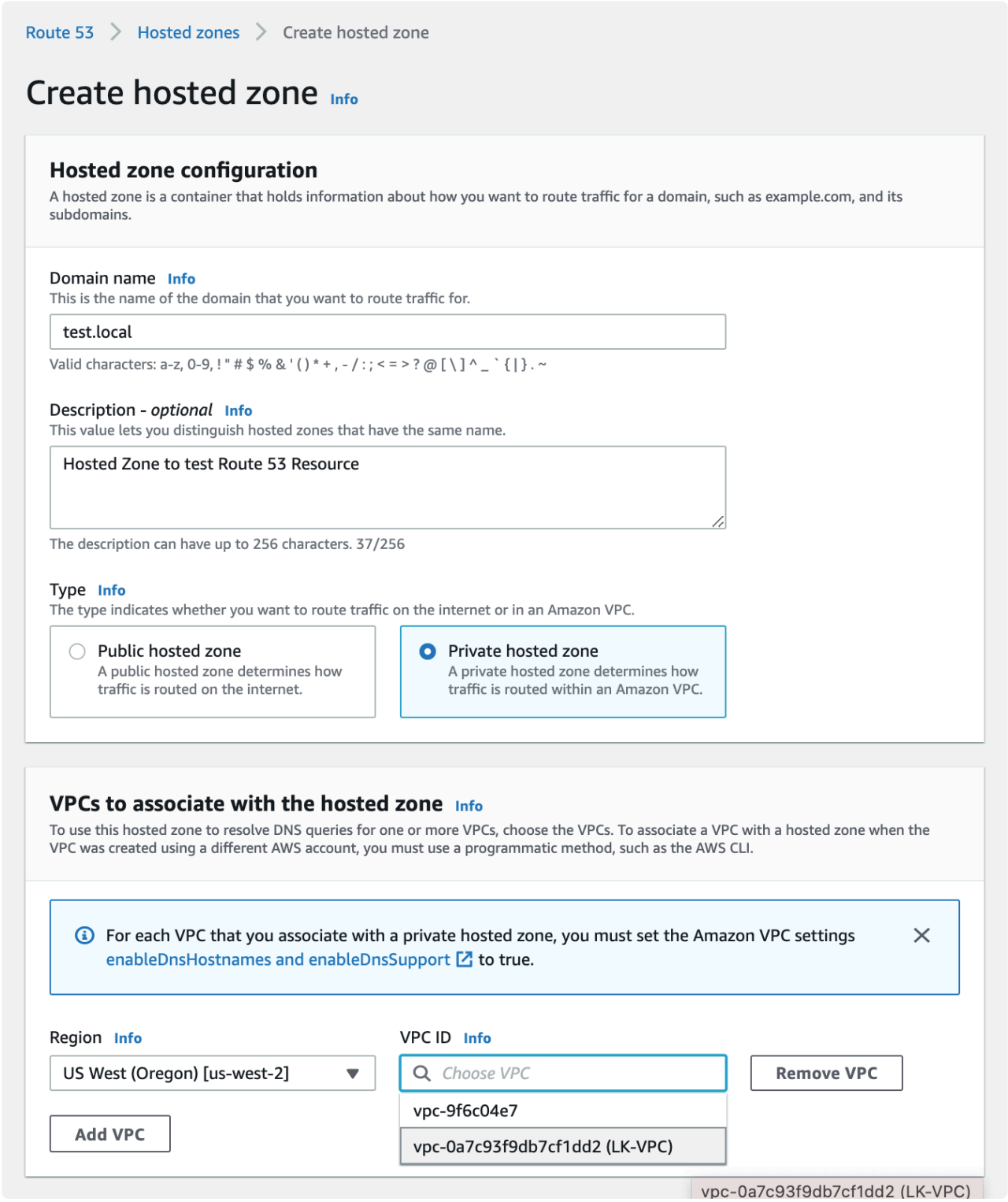
## Create a Hosted Zone

First, you need a Hosted Zone (this should look like a DNS domain such as example.com).

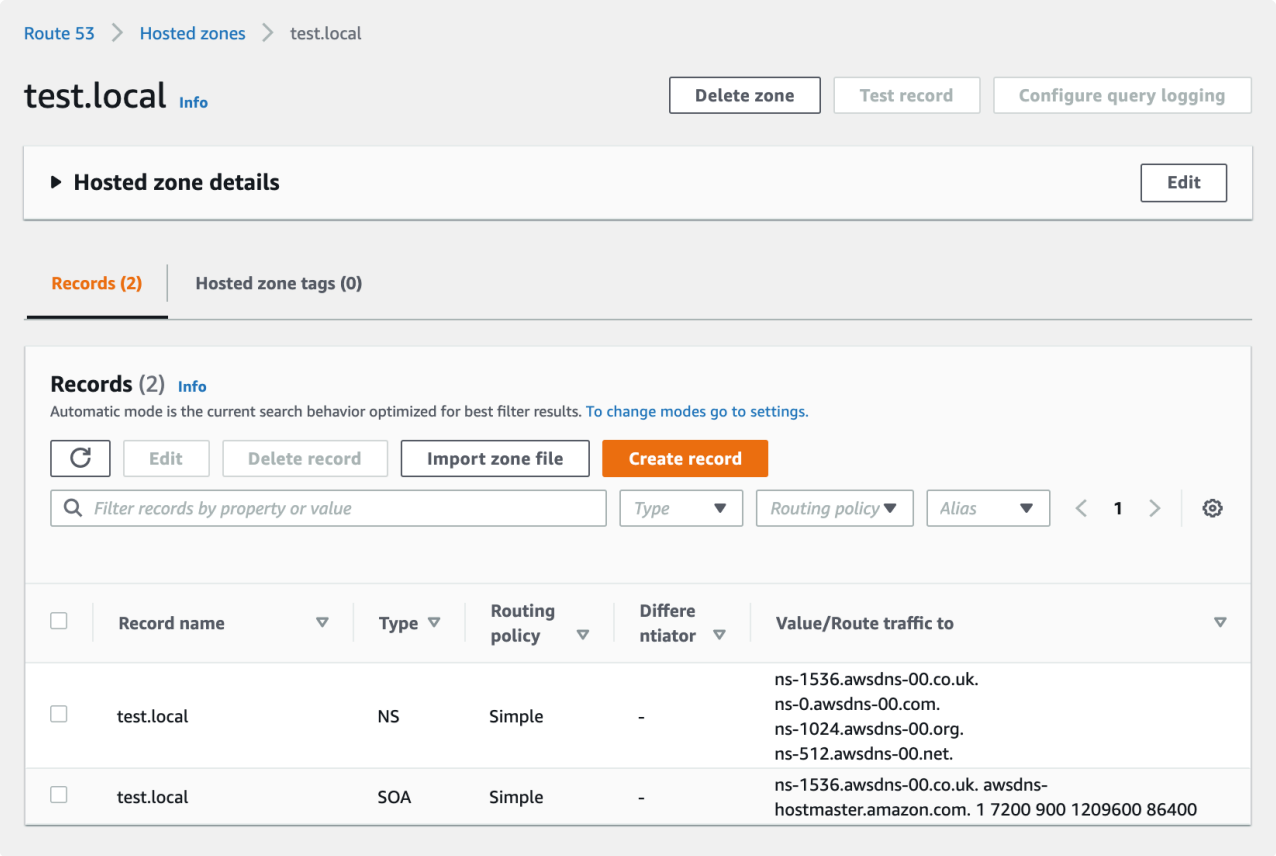
> If you already have a Hosted Zone, skip this section and use your Hosted Zone.

1. On the AWS management console, go to Route53.
2. Select Hosted Zone.
3. Select “Create Hosted Zone”.
4. Enter the following values:

| Field | Value |
|---|---|
| Domain | test.local |
| Type | Private hosted zone |
| VPC Association | Select region and name of VPC (e.g., LK-VPC) |



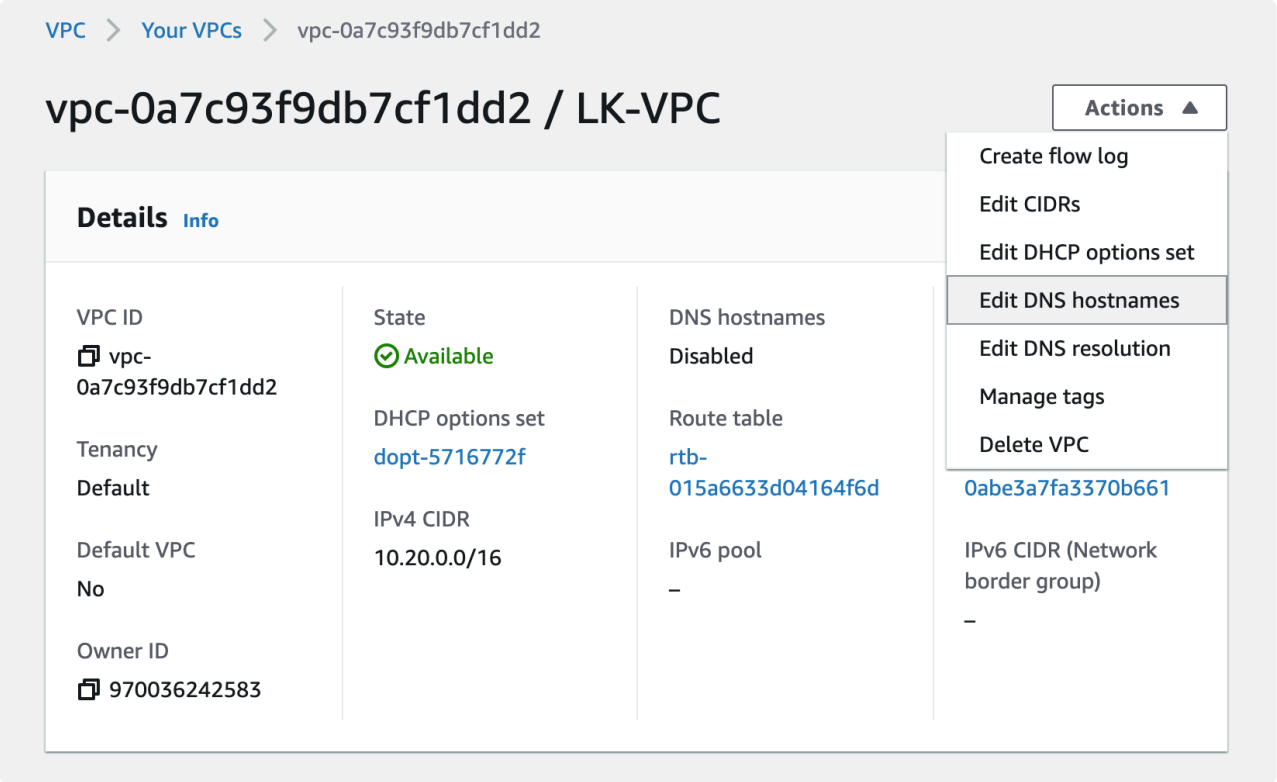

5. The Hosted Zone is now created. At this point, there are only two entries (NS & SOA) created.




6. To connect from another VPC, **the other VPC** must be associated.

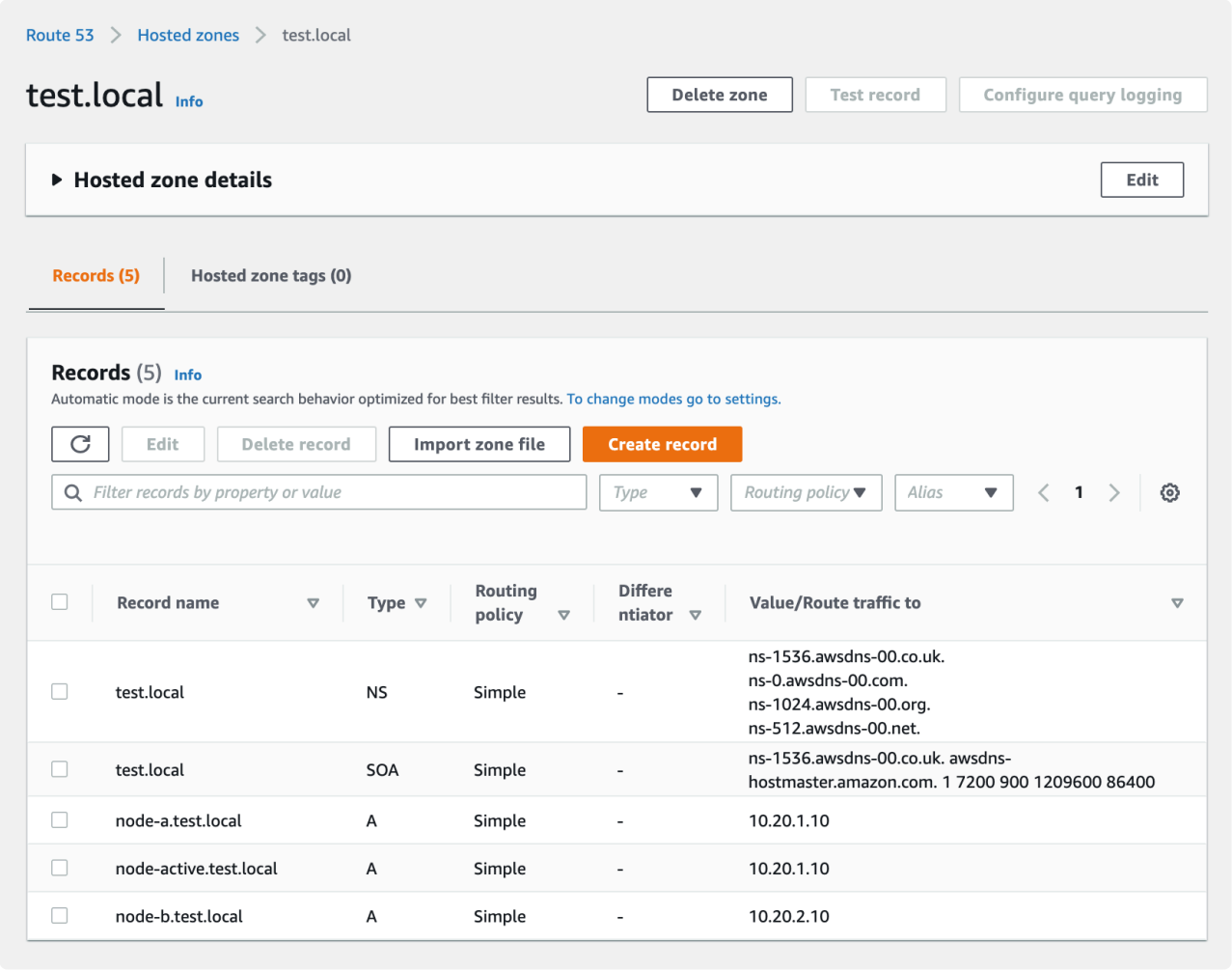
> ! In order to use DNS on VPC, enable `DNS Hostnames` and `DNS Resolution` in the VPC settings.

Go to the VPC settings and ensure that these two values are set to `enabled`.




# Create Entries on the Hosted Zone

1. Create records as shown below:



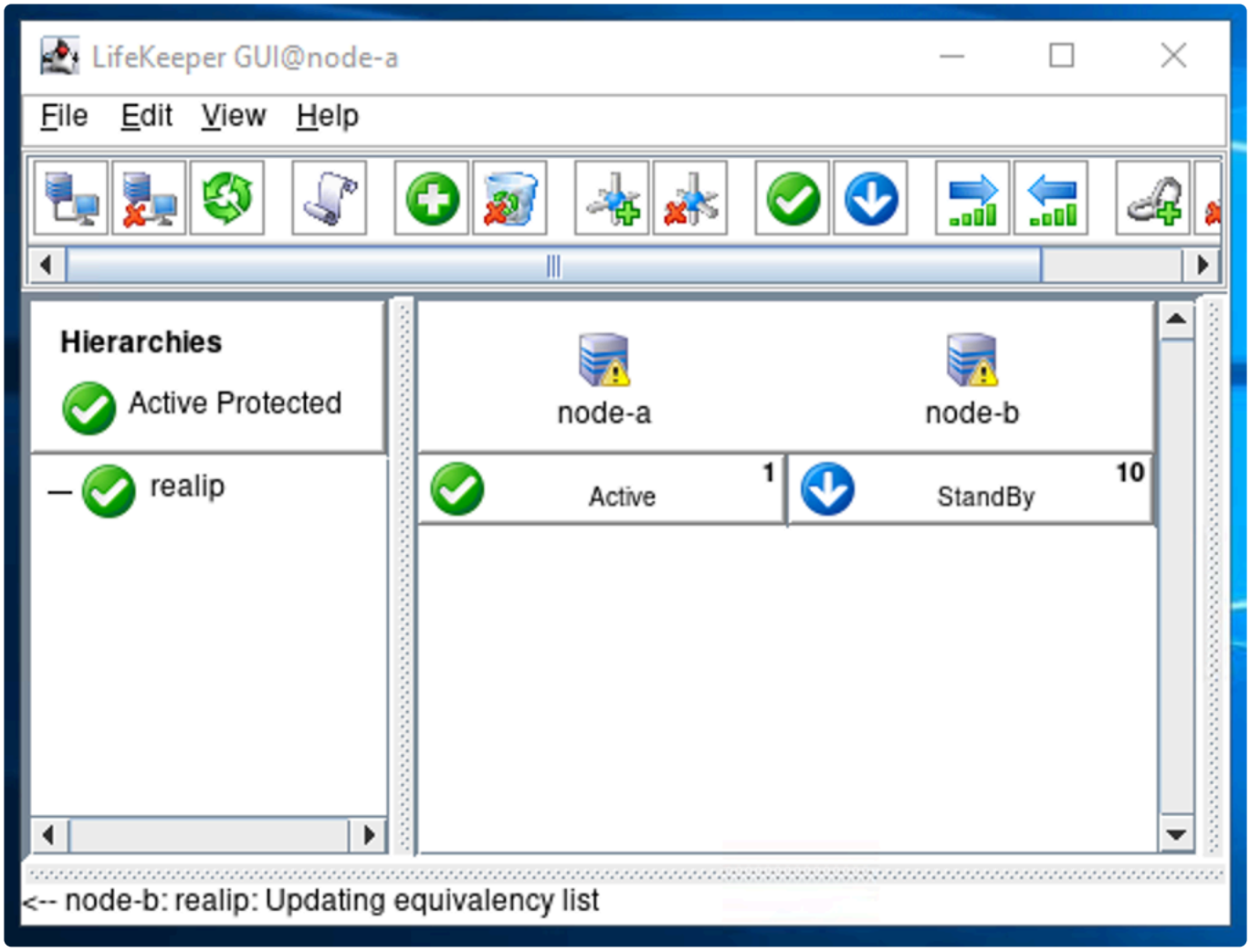

2. Confirm the hosts can be resolved by name.

```
[root@node-a ~]# ping node-a.test.cl
PING node-a.test.cl (10.20.1.10) 56(84) bytes of data.
64 bytes from node-a (10.20.1.10): icmp_seq=1 ttl=64 time=0.019 ms
64 bytes from node-a (10.20.1.10): icmp_seq=2 ttl=64 time=0.022 ms
```

# Create an IP Resource

Now you can create an IP resource. Note that the IP address of the resource has to be an active node (the IP address of node-active.test.cl: `10.20.1.10`). Refer to Creating an IP Resource for more information. Once the IP address of the active node (`10.20.1.10`) has been selected, the “IP Resource Tag” field is set to “realip” and the IP address of node-b (secondary) is set to `10.20.2.10` (the actual IP address of the second node). Those are automatically selected by selecting the “realip” id for the first node.

Once the IP resource is created, the LifeKeeper User Interface should look like this.




# Assign Permission to Update Route 53 Entries to Instances

> ! To complete the steps below, the AWS CLI must be installed on each node.

- Install AWS CLI

To allow LifeKeeper (running on the EC2 Instance) to update a Route53 entry, the instance must have roles that allow DNS updates.

- route53:ListHostedZones
- route53:GetChange
- route53:ChangeResourceRecordSets
- route53:ListResourceRecordSets

To achieve this, create a policy like the one seen below (note that it may be necessary to limit the resources that may be accessed), then assign it to a Role.

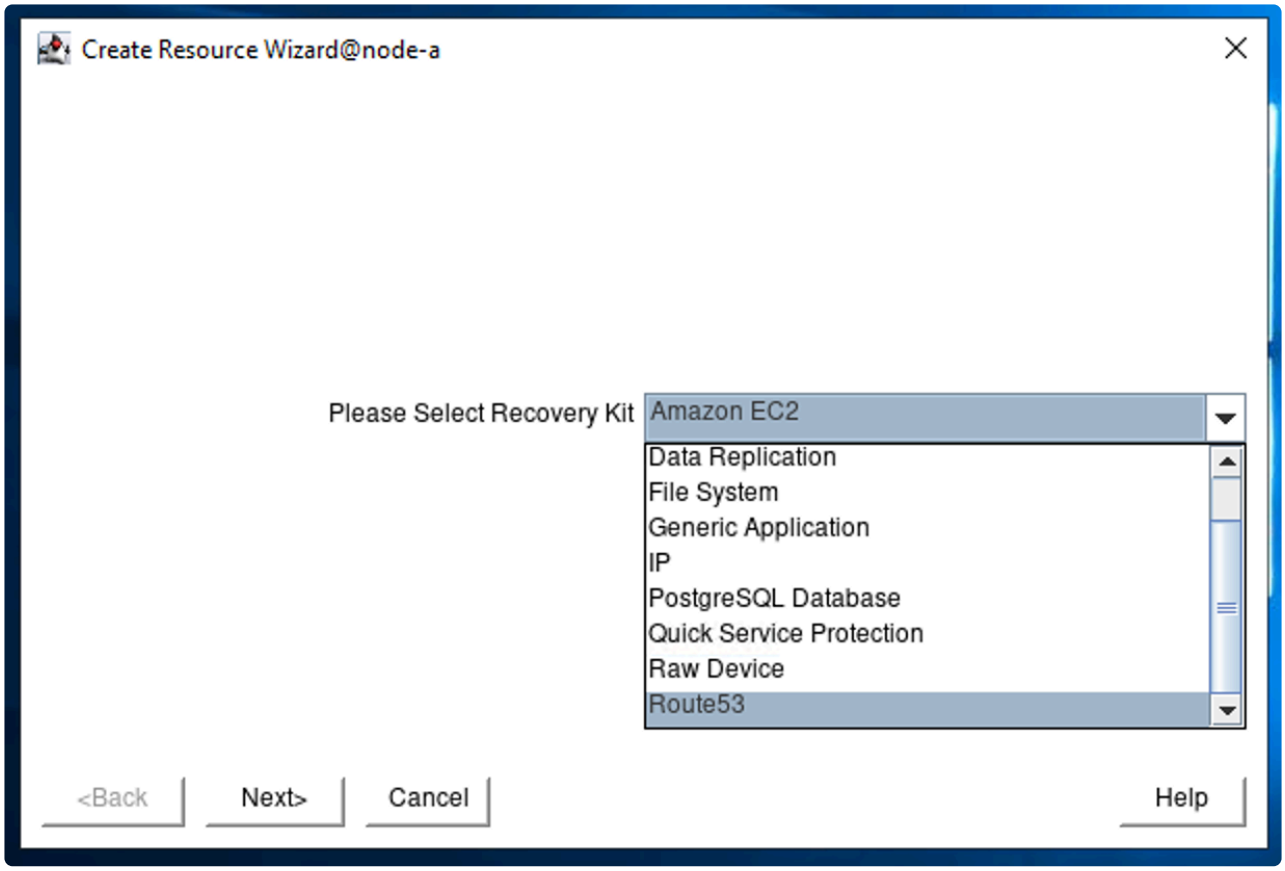
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "route53:GetChange",
                "route53:ChangeResourceRecordSets",
                "route53:ListResourceRecordSets"
            ],
            "Resource": [
                "arn:aws:route53:::hostedzone/*",
                "arn:aws:route53:::change/*"
            ]
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "route53:ListHostedZones",
            "Resource": "*"
        }
    ]
}
```

Once a Role is defined, assign it to these EC2 Instances.

## Create the Route53 Resource

Creating the Route53 Resource should be straightforward once we have created the IP Resource.

1. Select Route53 as Recovery Kit.




2. On the “Create Resource Wizard @ node-a”, specify the following values.

**Note:** For items with a checkmark ( ) review the default value and use the value suggested.

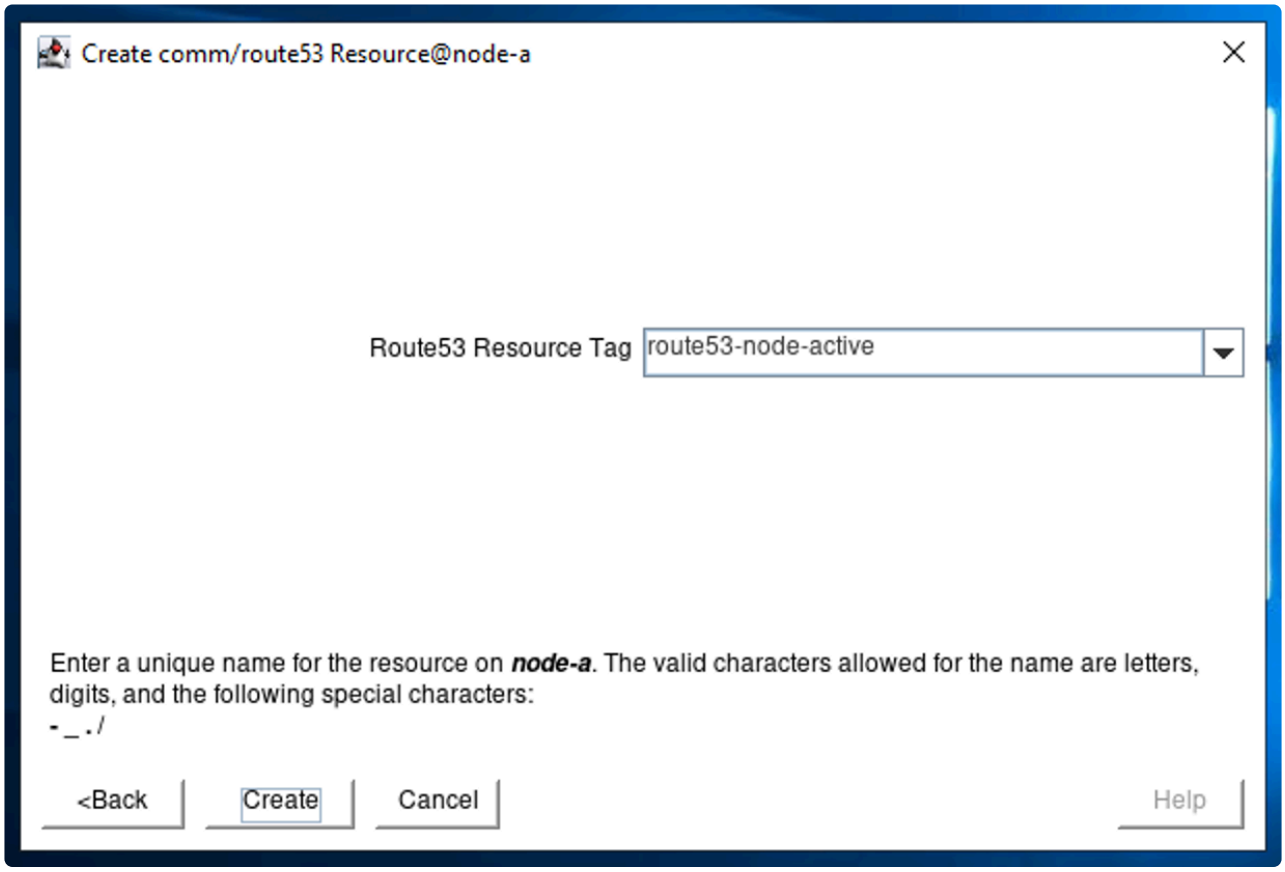
| Field | Value |
|---|---|
| Switchback Type | Intelligent |
| Server | node-a |
| Domain Name (Route53 hosted zone) | test.local |
| Host Name | node-active |
| IP Resource | realip |
| Route53 Resource Tag | reoute53-node-active |

3. On the “Pre-Extend Wizard” specify the following values:

| Field | Value |
|---|---|
| Target Server | node-b |

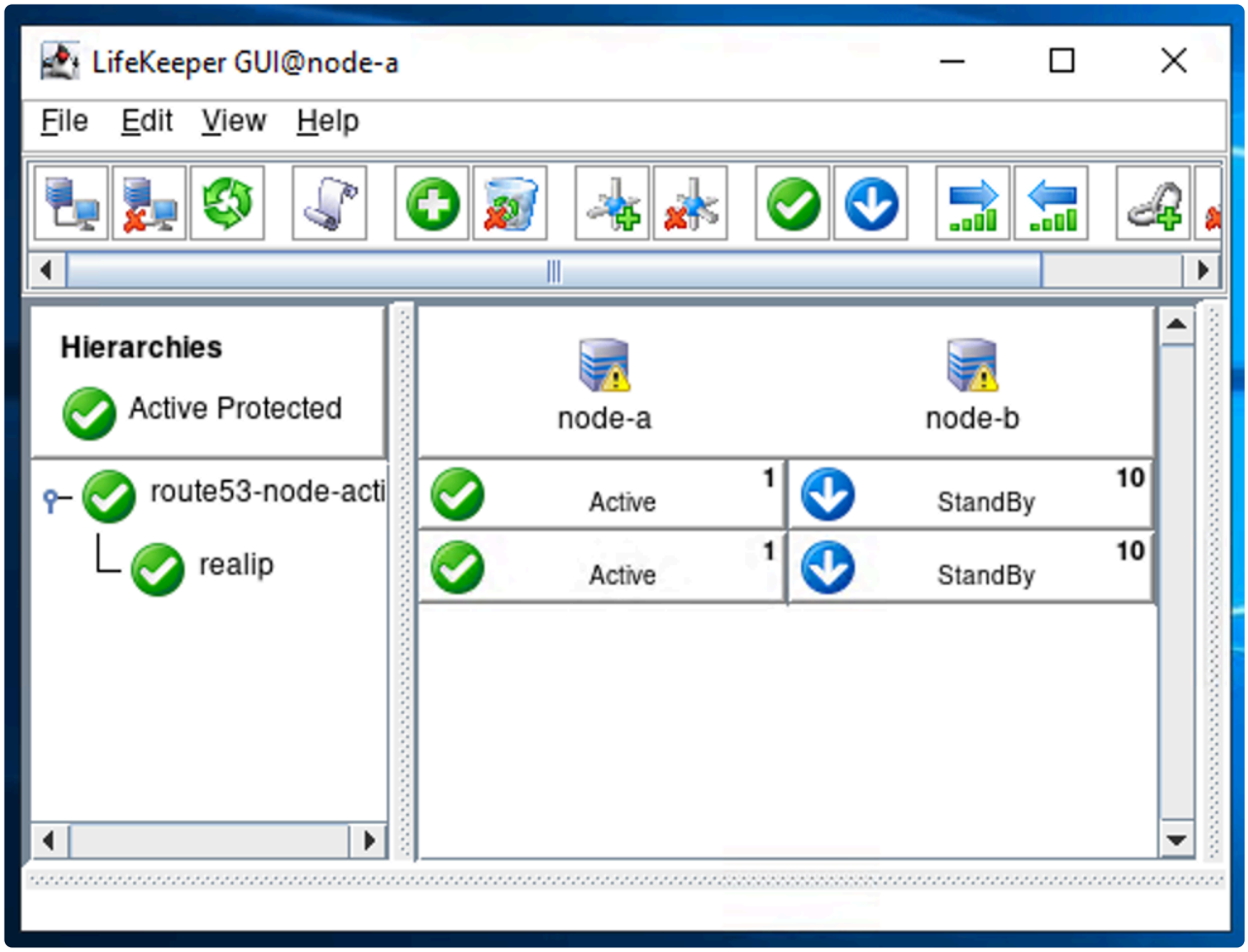
| | |
|---|---|
| Switchback Type | Intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

4. Once Pre-Extend is completed, move on to “Extend comm/route53 Resource”. Select “route53-node-active”.




5. Once “Hierarchy successfully extended” is displayed, the creation of the resource is complete.

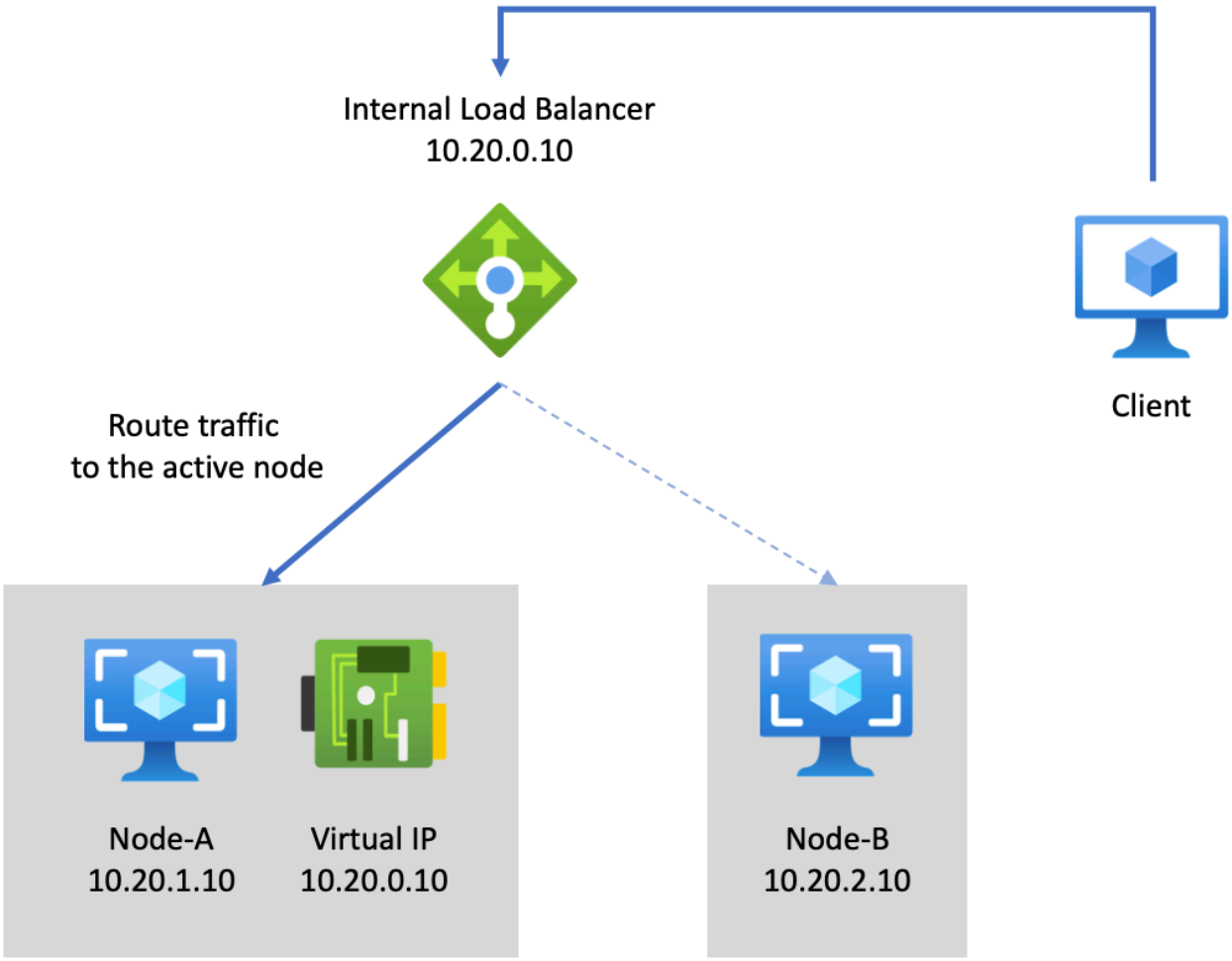
6. The Route53 resource is now created as shown below.




Notice that realip is located under route53-node-active. This indicates that the “route53-node-active” resource depends on the “realip” resource.

# 11.2.7.2.4. Azure – Using an Internal Load Balancer

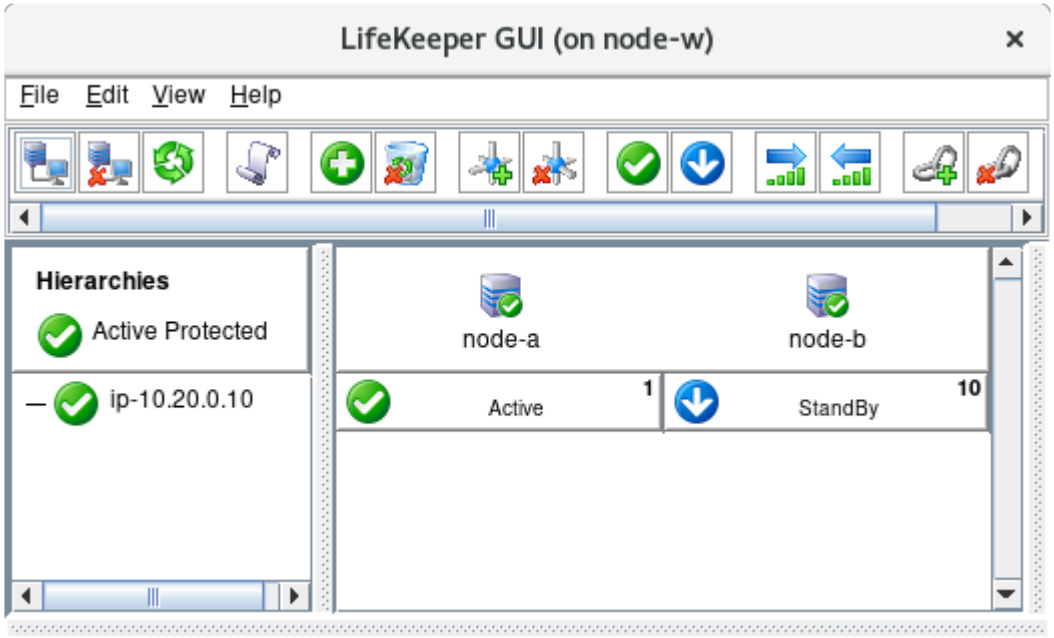
Here we will discuss how to switch between nodes using an Azure Internal Load Balancer by creating an Internal Load Balancer that points to the active node. Clients connect to the Frontend IP address provided by the Internal Load Balancer. The VMs are checked regularly by the “Health Check Probe” function and the Internal Load Balancer routes requests to the active node.




## Create an IP Resource

On Azure, a client connects to a load balancer so that it can distribute the traffic to the active node. In theory, an IP Resource (that represents a Virtual IP address) is not needed. However, as this section describes later, it is still necessary to create a Virtual IP address.

In Azure’s case, the Virtual IP address should be the same as the IP address for a load balancer in front of these nodes. In the example below we will use the IP address `10.20.0.10`.




> ! Before configuring a load balancer, it is important to check which port the application listens to (for example, port 80 for http). To ensure that the application is running, the Internal Load Balancer periodically checks the status of the application (this is referred to as a “health check probe”). It then sends incoming requests to the active node. The following sections use port 80 as an example, but this should be modified as appropriate for the application being protected.

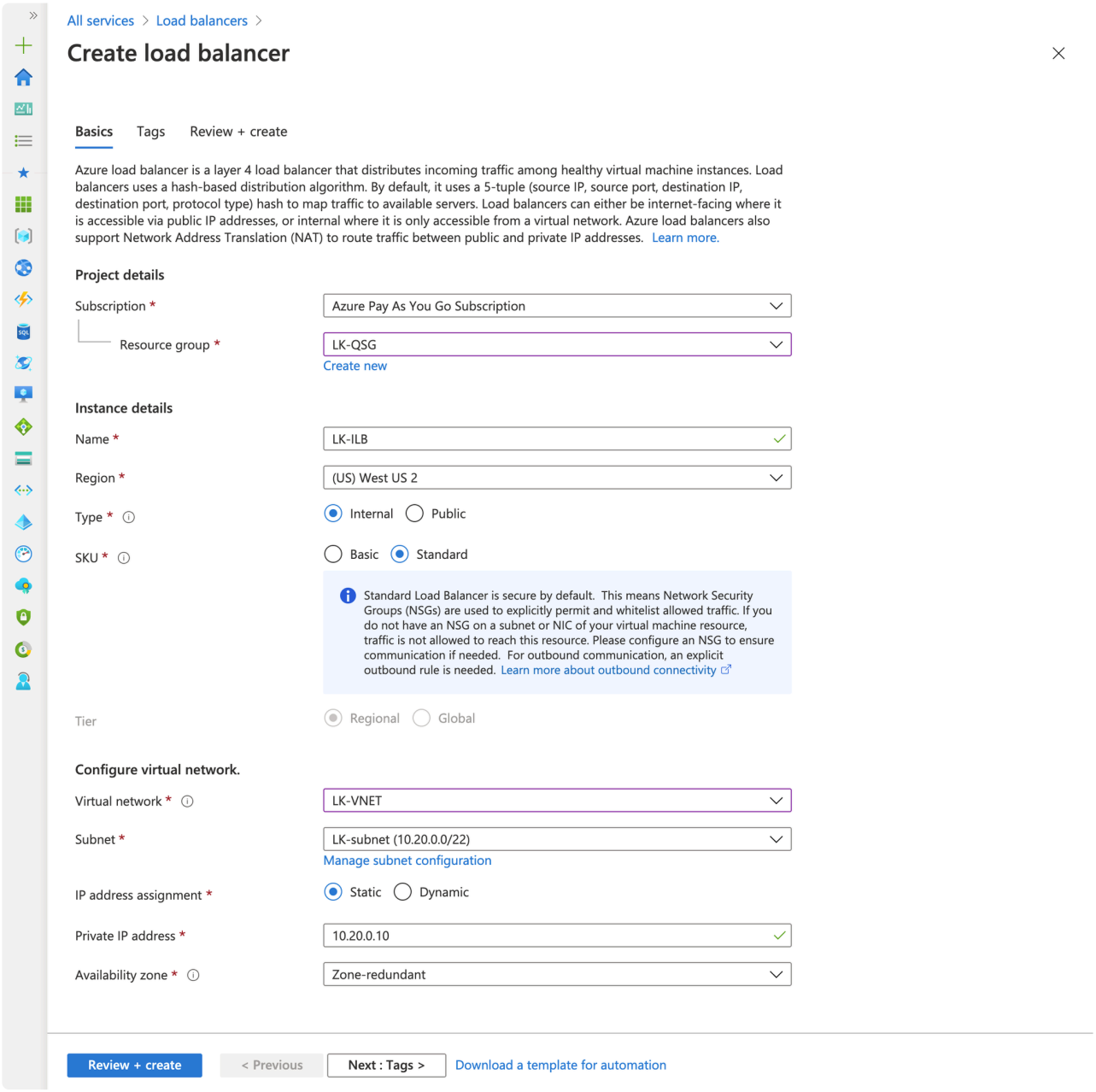
## Create an Internal Load Balancer

An internal load balancer distributes traffic to the active node and can be created using the following steps. In order to configure the load balancer, start the application on node-a to ensure the load balancer is working before configuring it through LifeKeeper.

1. On the Azure Portal, go to Load Balancer, click “Add”, select the following parameters and then click “Review + Create”. Once the parameters are confirmed, the Load Balancer can be created.

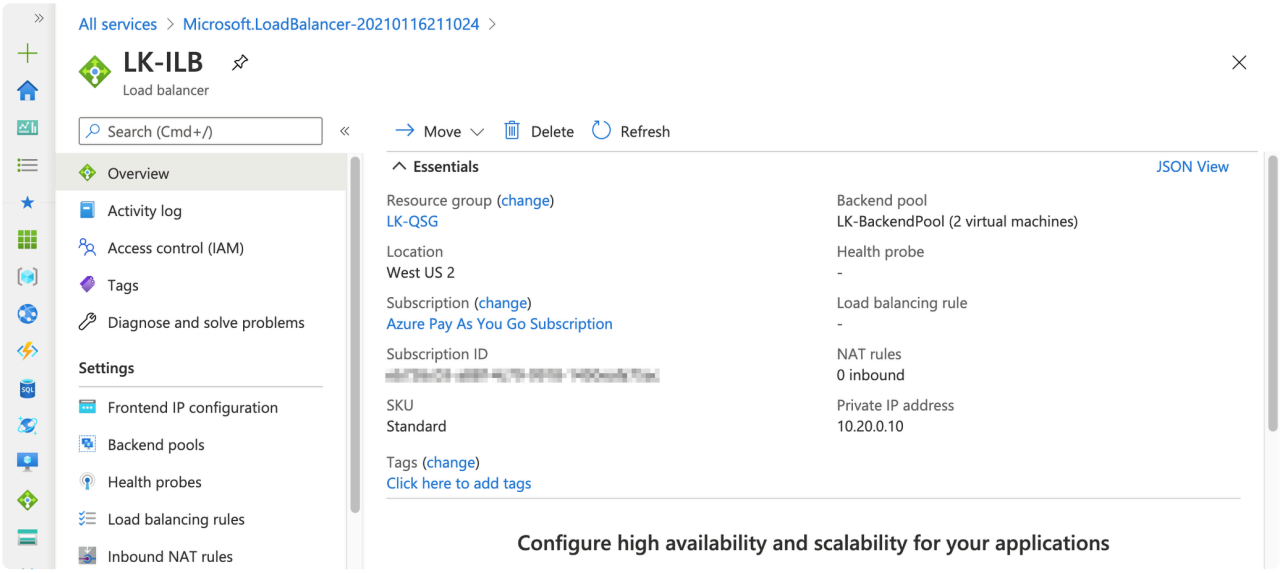
| Item | Value |
|---|---|
| Resource Group | LK-QSG |
| Name | LK-ILB |
| Region | (same as the Virtual Machines) |
| Type | Internal |
| SKU | Standard (select Standard as the workload is distributed across Availability Zones) |
| Virtual Network | LK-VNET |
| Subnet | LK-subnet |

| | |
|---|---|
| IP address assignment | Static |
| Private IP Address | 10.20.0.10 |
| Availability zone | Zone-redundant |

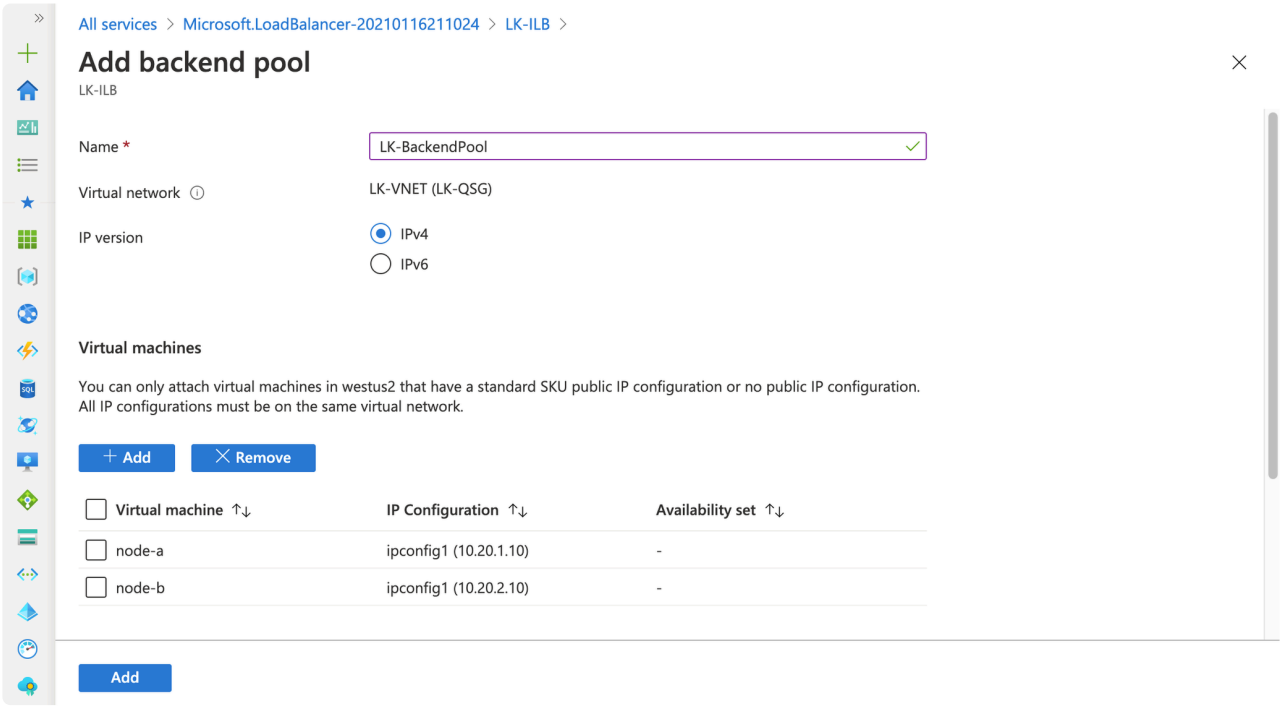



2. Once the load balancer is created, go to the resource page (LK-ILB). Configure the following sections of the load balancer (located in the Setting section on the left side of the Load Balancer configuration page).

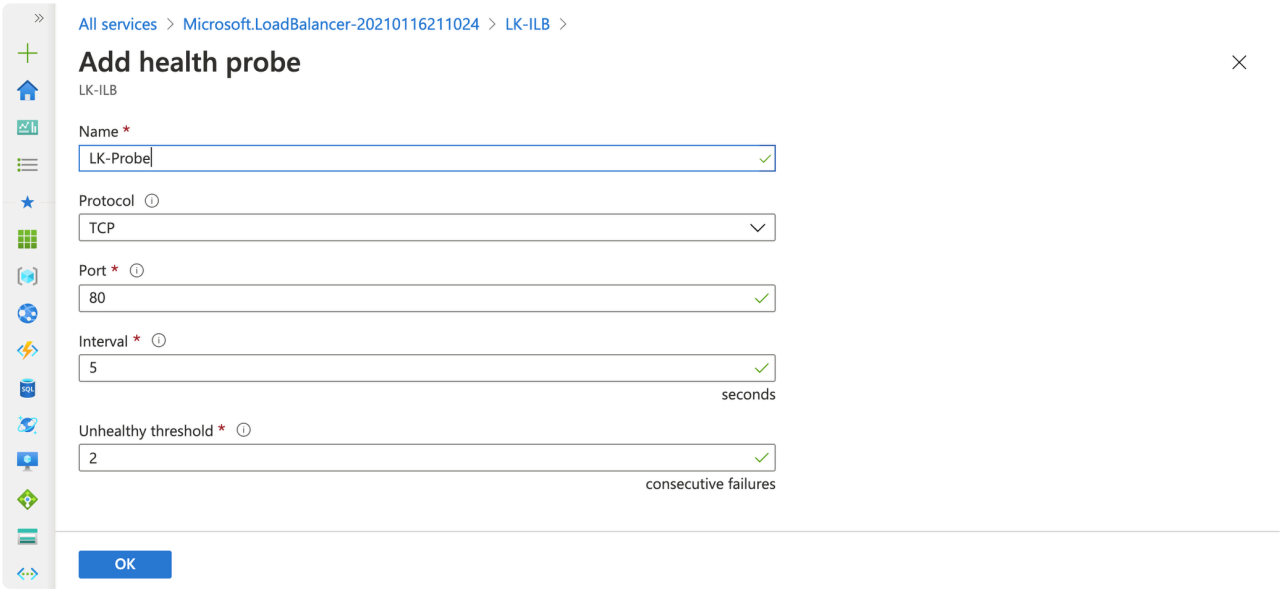
- Backend pools (step 3)
- Health probes (step 4)
- Load Balancing Rules (step 5)




3. Go to the Backend pools page and click “Add”. Once the Backend Pool Page is open, select node-a and node-b from the list of available virtual machines and provide a name for the pool (for example, LK-BackendPool).

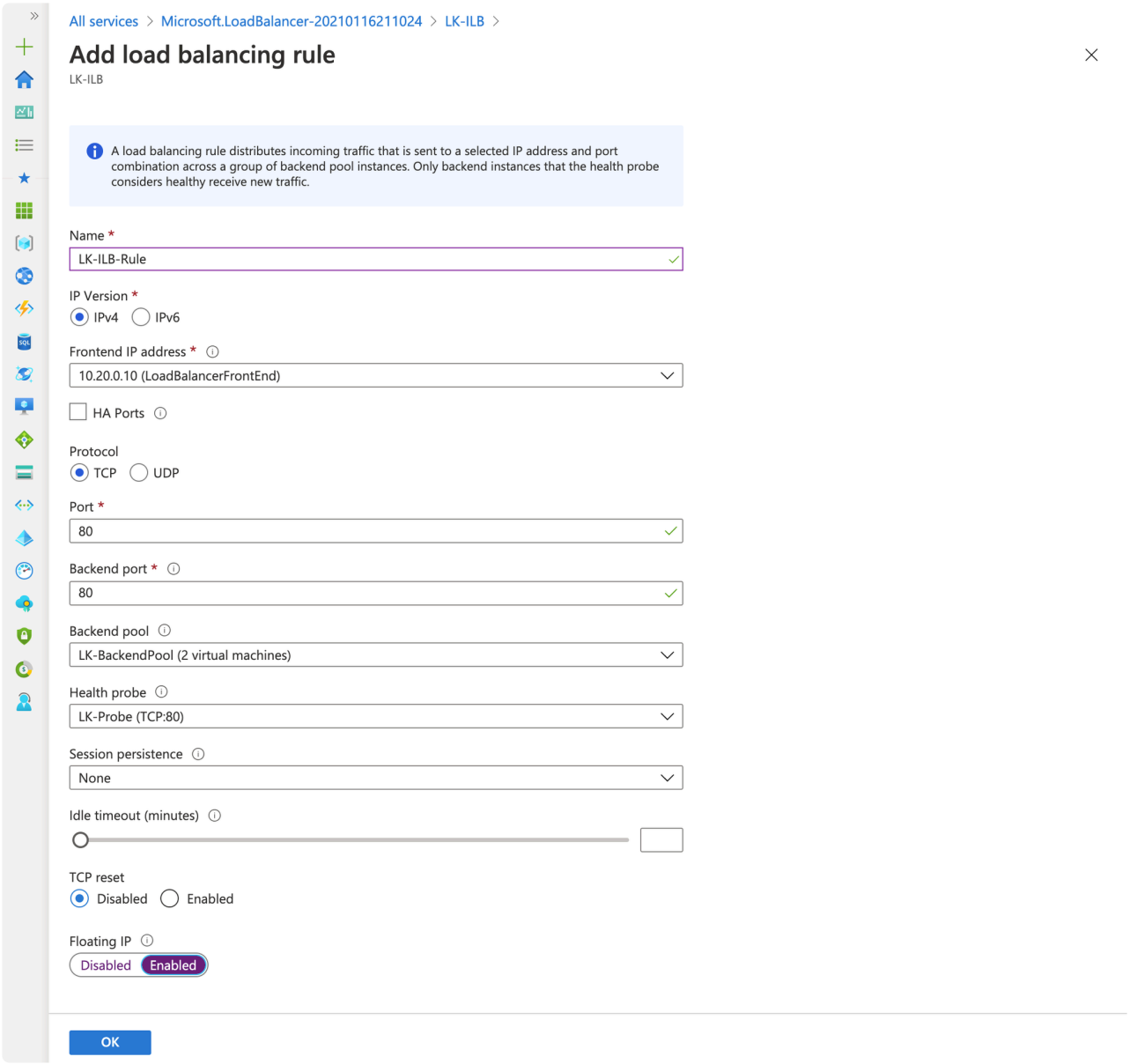



4. Go to the Health Probes page and click “Add”. Select the TCP port number used by the application (e.g., 80 if we are going to protect `httpd`), then provide a name for the probe (for example, LK-Probe).




5. Go to the Load Balancing Rule page and click “Add” to create a new rule. Enter the following values:

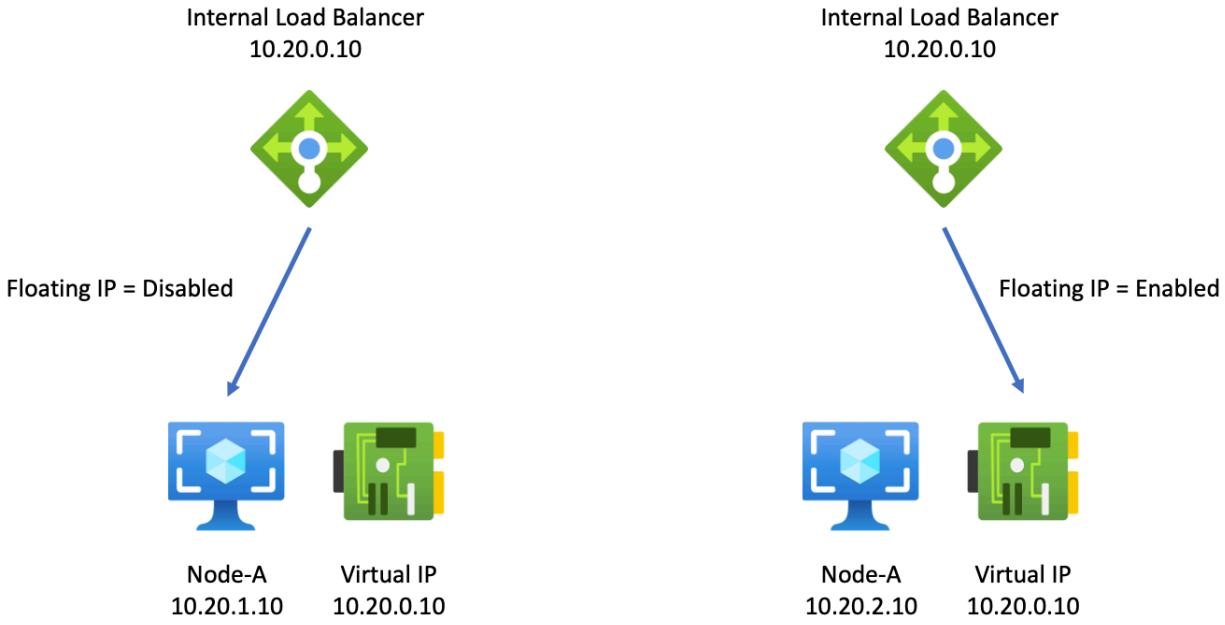
| Item | Value |
|---|---|
| Name | LK-ILB-Rule |
| Frontend IP address | 10.20.0.10 |
| Protocol | TCP (assuming your app uses TCP) |
| Port # | Select the Port # in use by the application |
| Backend Port | Same as Port |
| Backend Pool | LK-BackendPool |
| Health Probe | LK-Probe |
| Session Persistence | None |
| Floating IP | Enabled (you can also choose Disabled). See following section for details. |




# The Difference between Floating IP Options

The difference between Floating IP options determines where the Internal Load Balancer sends packets to.

- If Disabled is selected, the Load Balancer sends packets to the node’s primary IP.
- If Enabled is selected, the Load Balancer sends packets to the Virtual IP.

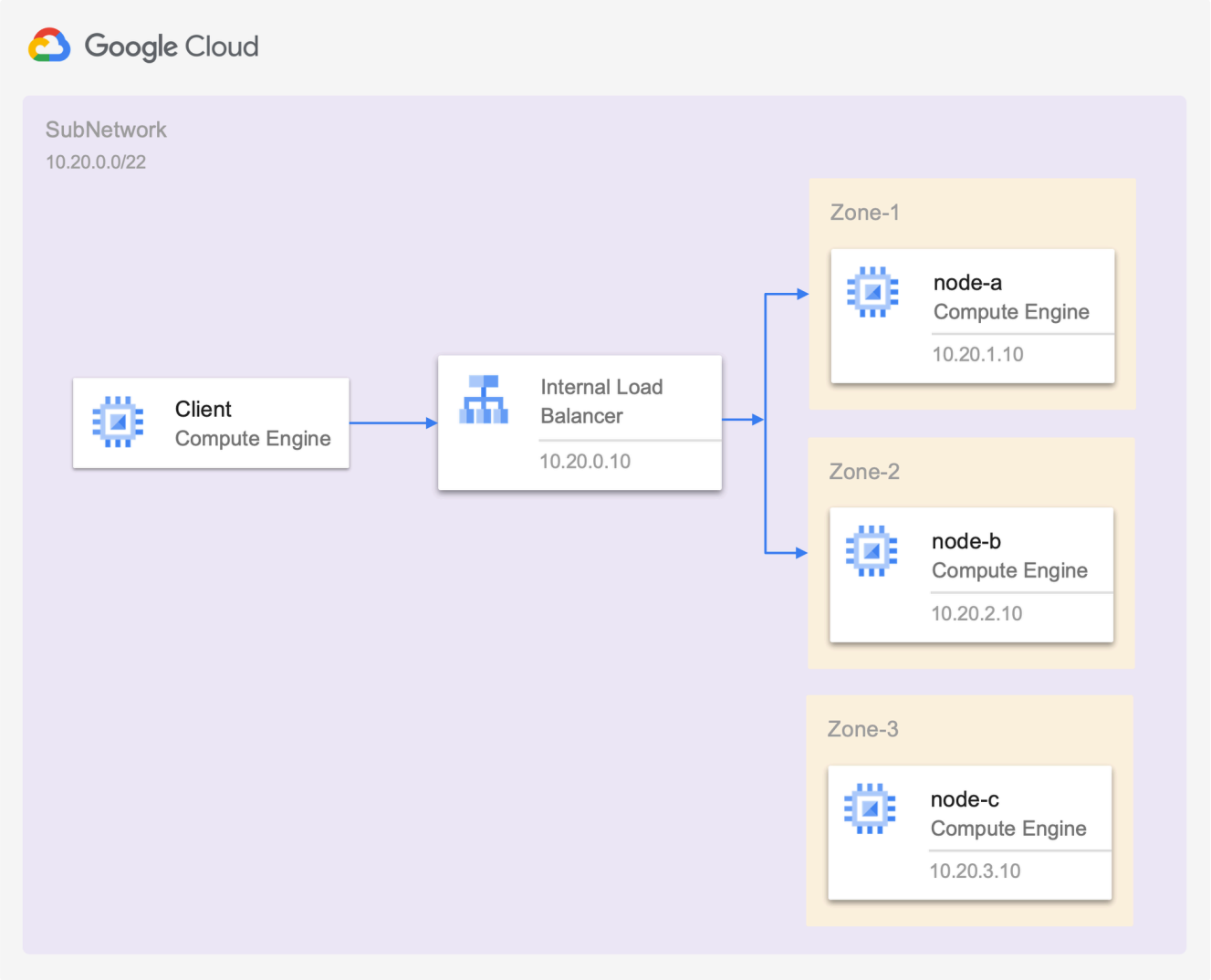



Because the Virtual IP is attached to the active node, it appears to be the same operation. However, if the response from the active node is required to be sent from the Virtual IP, “Enabled” should be selected. This is a more natural way for LifeKeeper to handle these nodes (in the same way as in an on-premise environment). If the application listens only to the primary IP address (for example, `10.20.1.10`), the Floating IP parameter should be set to “Disabled”.

**Note:** The Health Probe checks the status of the node by sending requests to the primary IP address (for example, 10.20.1.10). This is true regardless of the “Floating IP” option value.

# 11.2.7.2.5. GCP – Using an Internal Load Balancer

Here we will discuss how to switch between nodes using GCP's Internal Load Balancer by creating an Internal Load Balancer that routes traffic to the active node. Clients connect to the Frontend IP address provided by Internal Load Balancer. The Internal Load Balancer regularly checks the health of each VM in the backend pool using a user-defined "Health Check Probe" function, and then routes client requests to the active node.



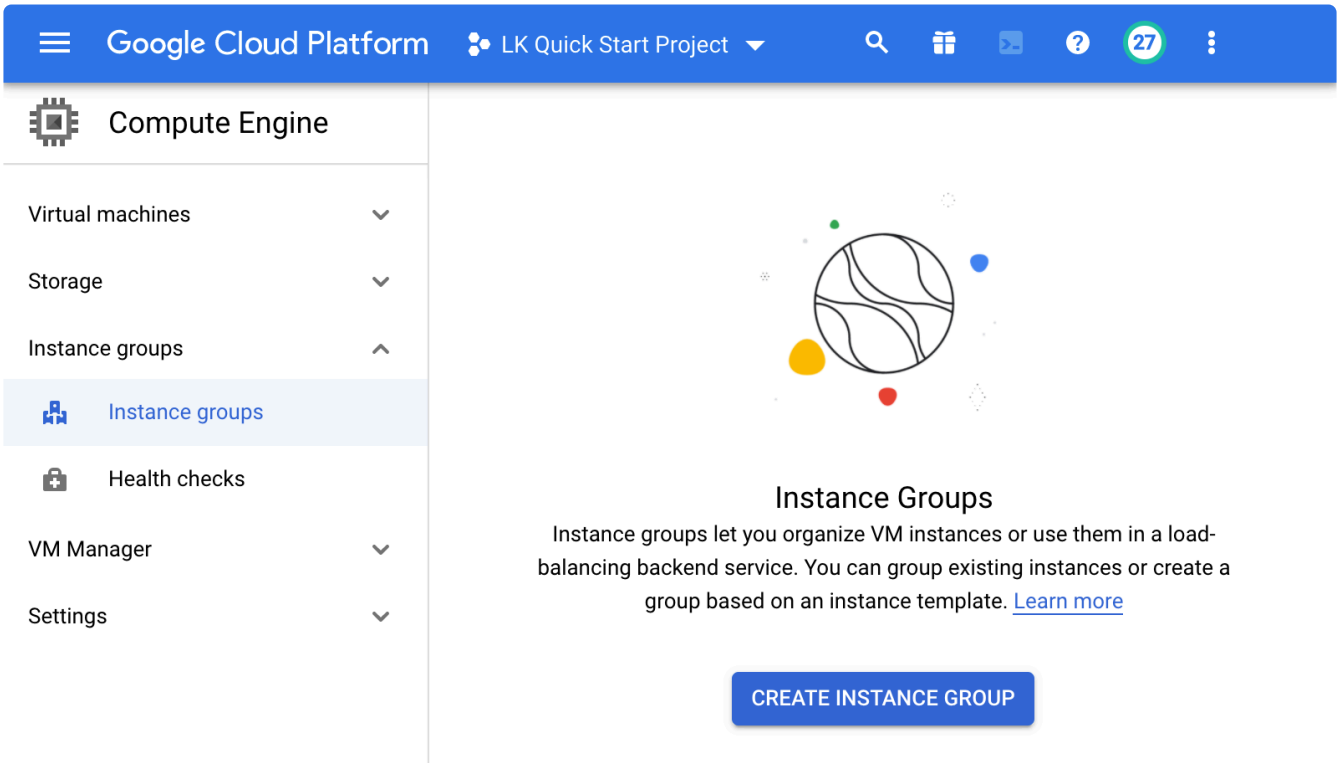

## Create Instance Groups

In order to use an Internal Load Balancer, the first step is to create Instance Groups containing each of our nodes. Create Instance Groups with the following parameters:

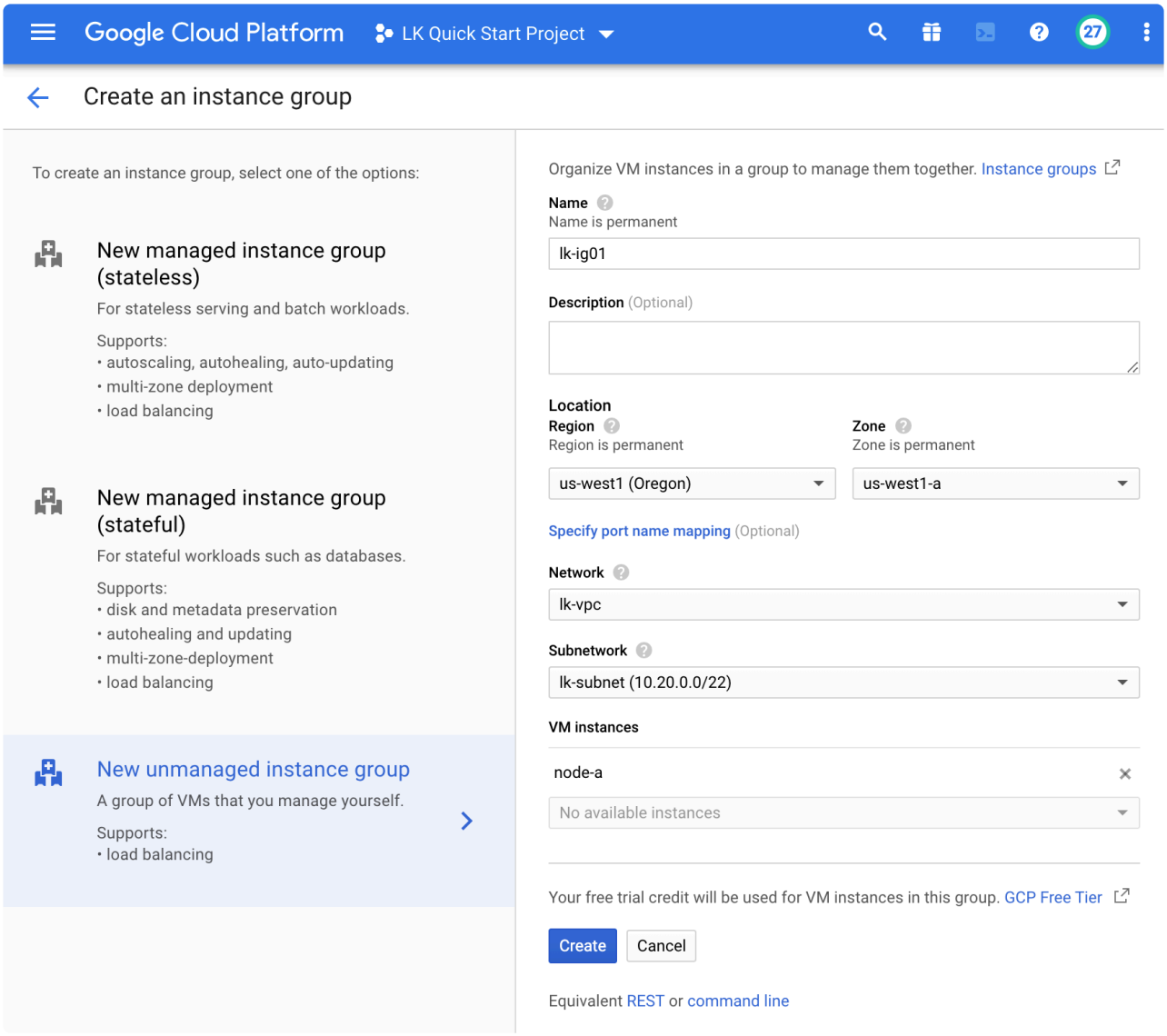
| Name | Region | Zone | Network | Subnetwork | VM Instance |
|---|---|---|---|---|---|
| lk-ig01 | us-west1 | a | lk-vpc | lk-subnet | node-a |

| | | | | | |
|---|---|---|---|---|---|
| lk-ig02 | | b | | | node-b |

1. Select “Compute Engine” > “Instance Group” from the navigation menu.

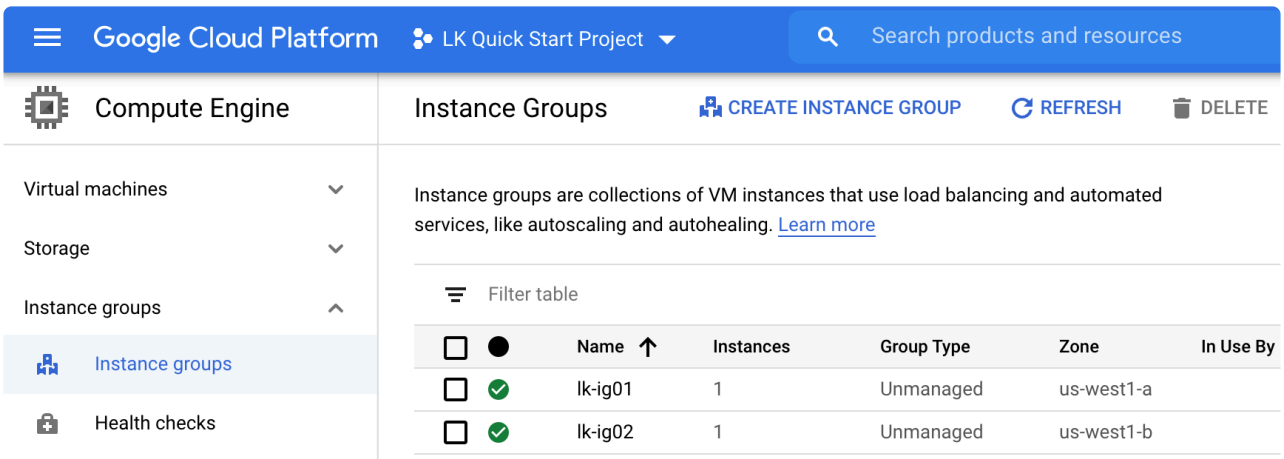



2. Select “CREATE INSTANCE GROUP”, “New Unmanaged instance group”, then enter the parameters listed in the table.




3. Create `lk-ig02` following the same steps.

4. Two Instance groups are now created.

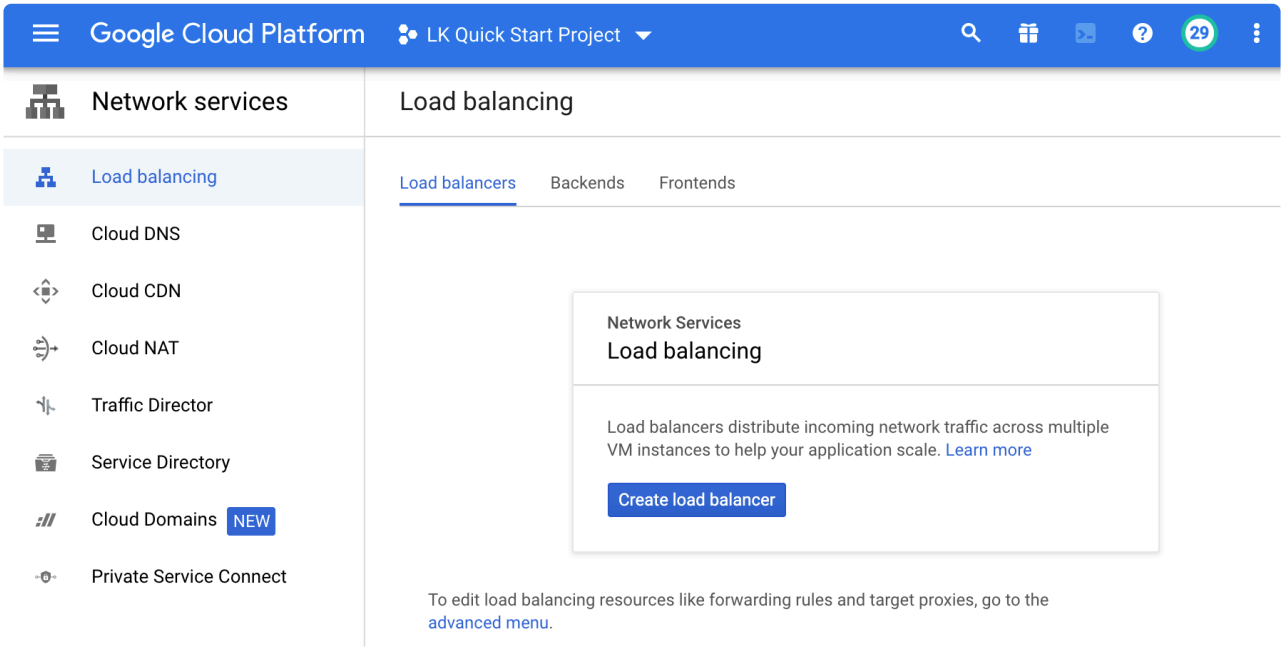


# Create an Internal Load Balancer

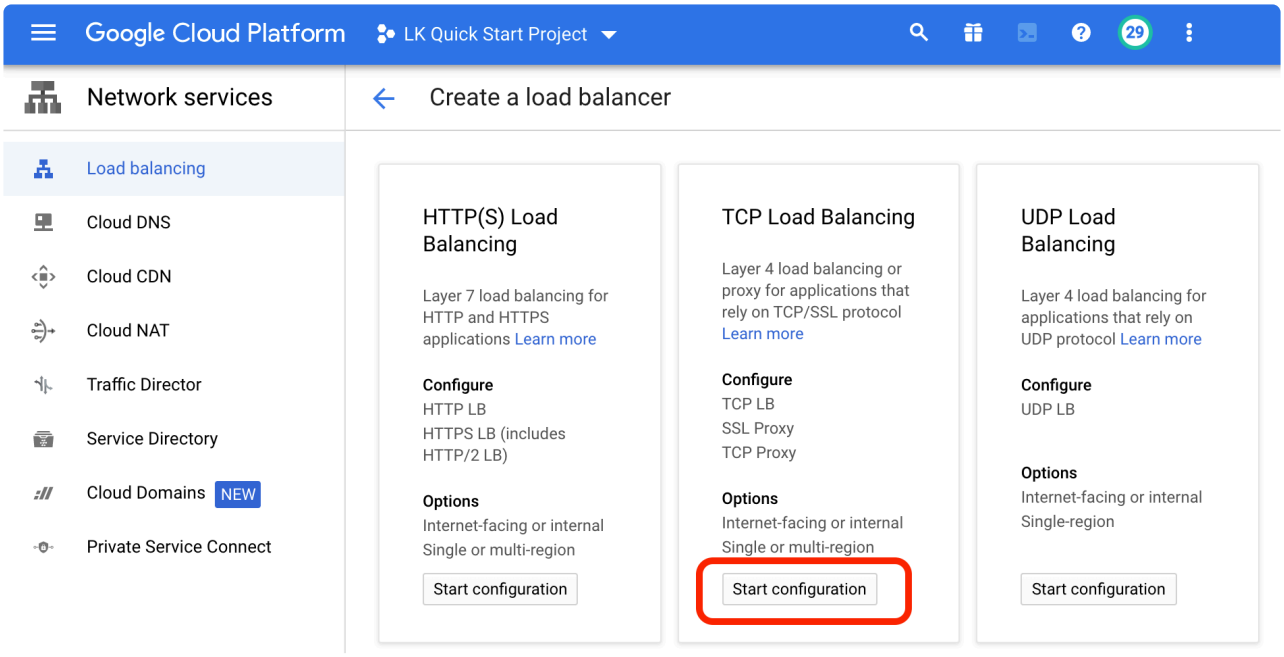
> ! Before configuring a load balancer, it is important to check which port the application listens to (for example, port 80 for http). To ensure that the application is running, the Internal Load Balancer periodically checks the status of the application (this is referred to as a “health check probe”). It then sends incoming requests to the active node. The following sections use port 80 as the example (which would be appropriate when setting up a health probe on a set of backend targets running web servers), but you should modify the port based on the application that you are protecting.

An internal load balancer distributes traffic to the active node and can be created using the following steps. In order to configure the load balancer, start the application on node-a (to ensure the load balancer is working before configuring it through LifeKeeper).

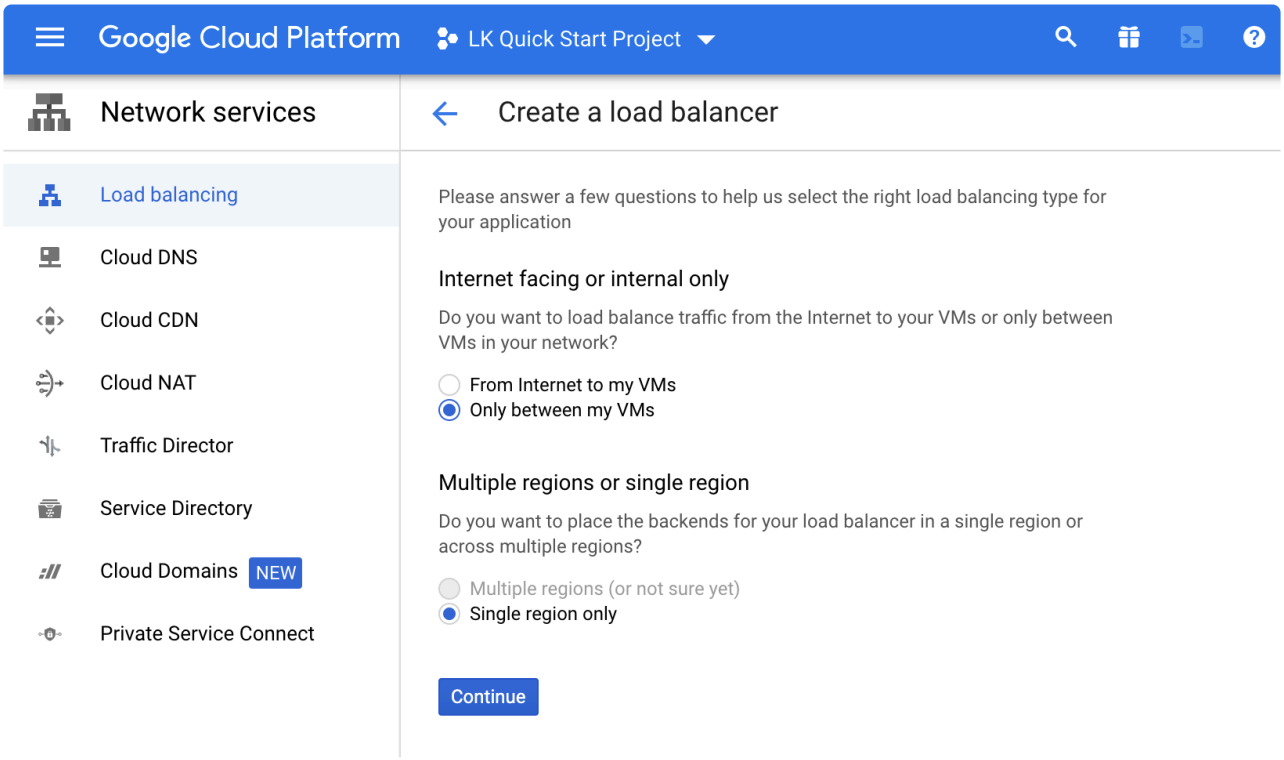
1. On the navigation menu, select “Network Service” > “Load Balancing”.



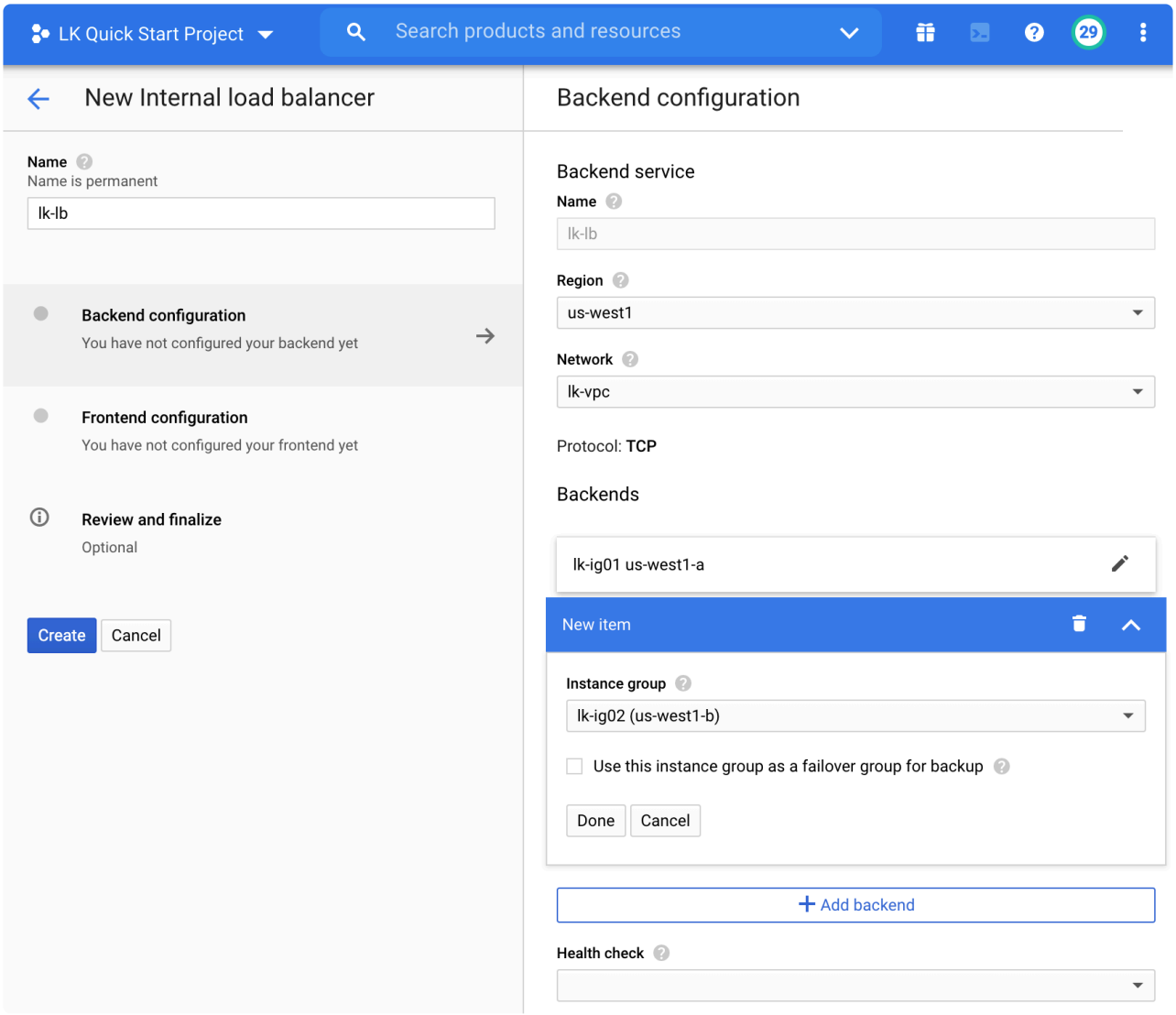

2. Click “Start Configuration” under “TCP Load Balancing”.



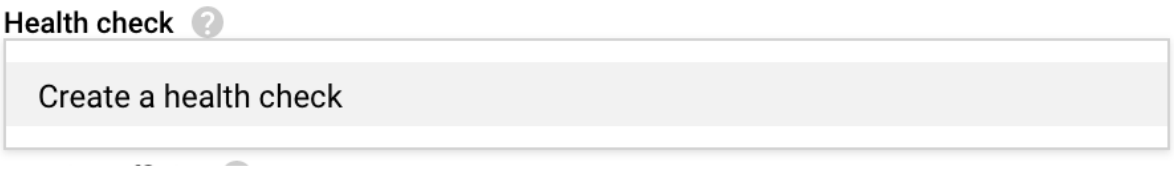

3. Select “Only between my VMs” (meaning Internal Load Balancer), and Single region only. Click “Continue”.



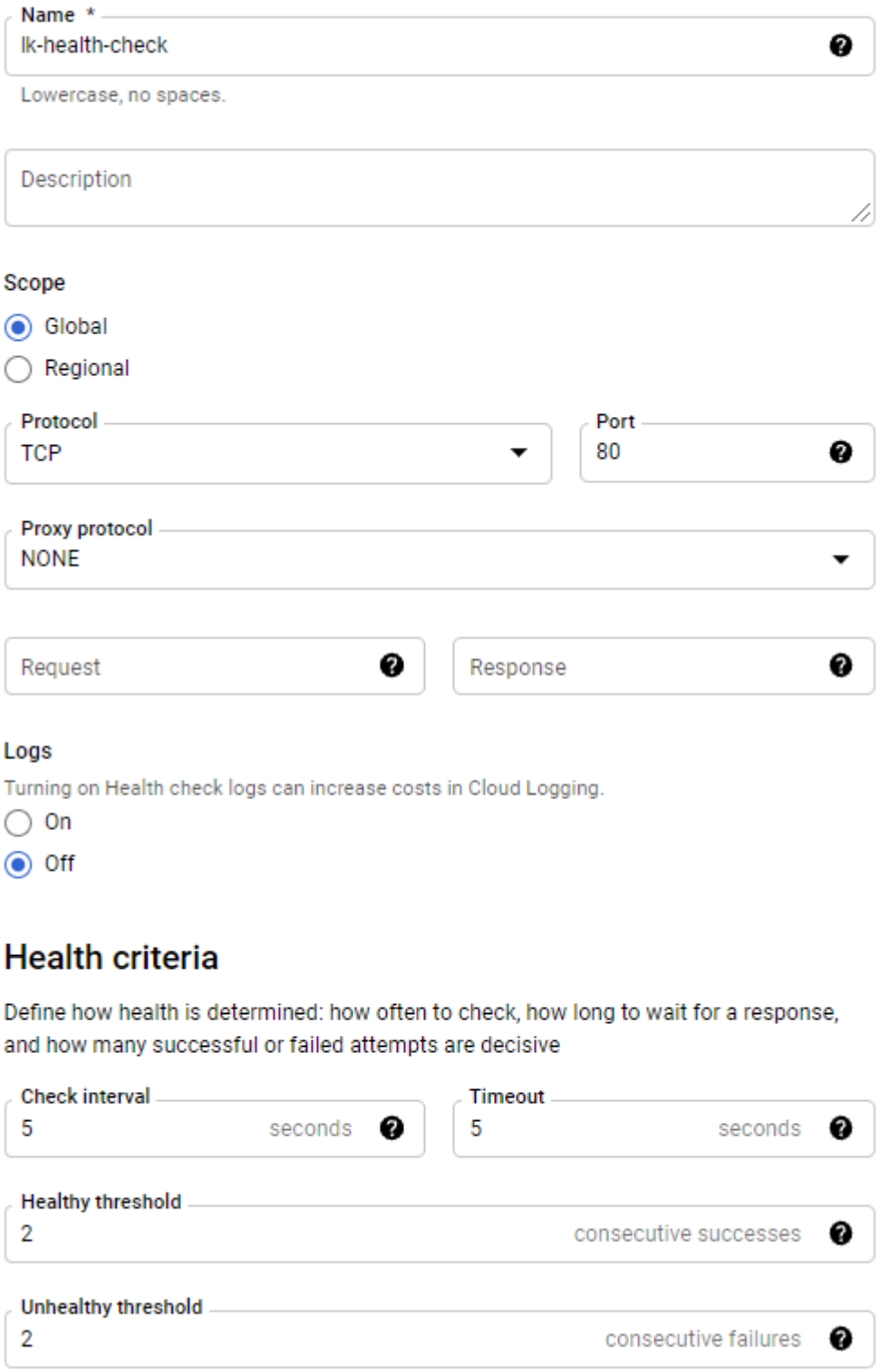

4. Specify the name as `lk-lb`, then configure the backend parameters. Select Region, Network as `lk-vpc`, then select two instance groups (`lk-ig01` and `lk-ig02`) from the dropdown menu to be added to the backend pool.



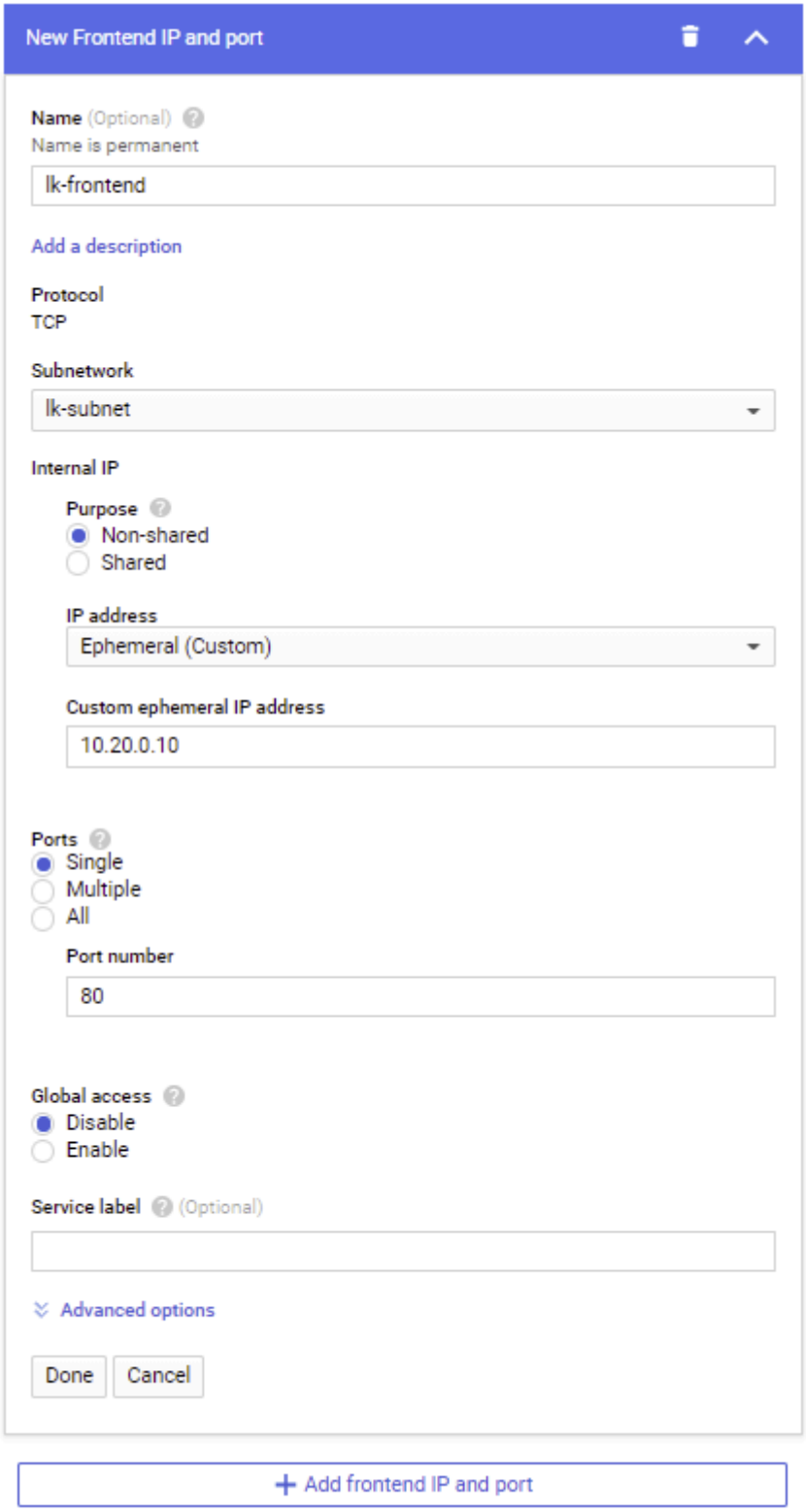

5. Move on to “Health check” located at the bottom of the Backend configuration page. Select “Create a health check”.



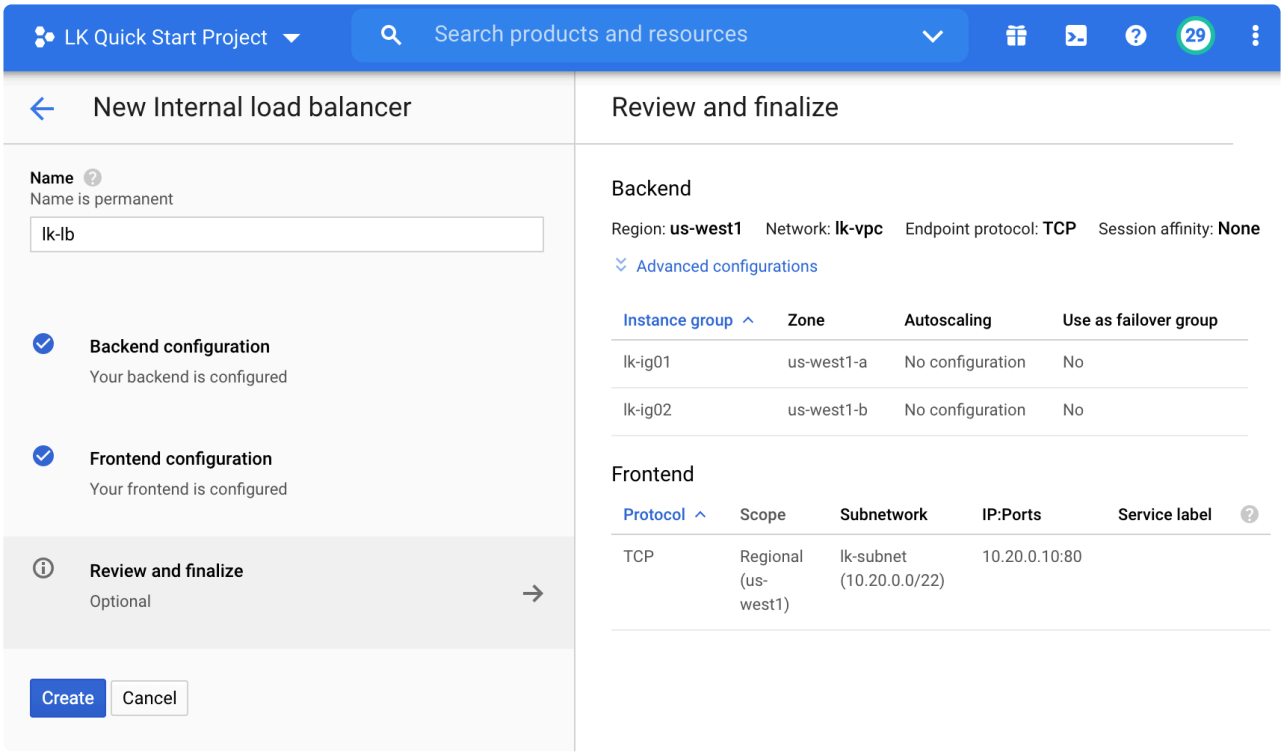

6. The health check configuration screen appears. Select the name `lk-health-check`, select Port (make sure it matches with the service you are going to use), then review the other parameters and click “Save and continue”.



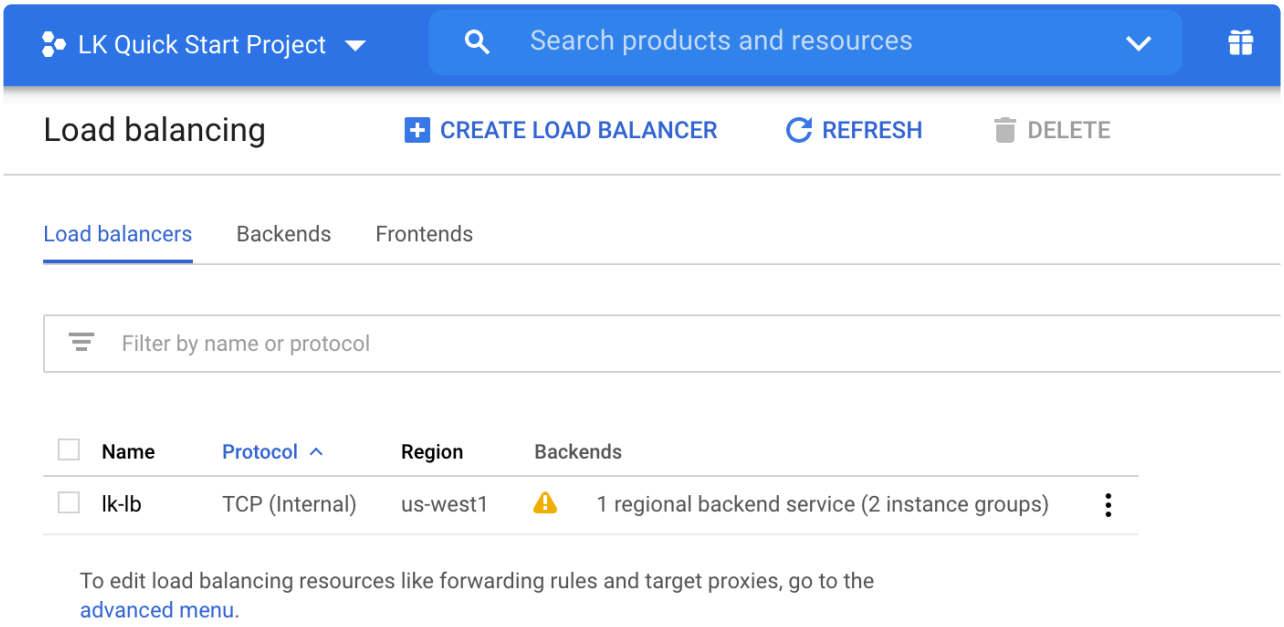

7. Move to the “Frontend configuration” tab. Provide the name and subnetwork for the frontend. In this example we will use the name `lk-frontend` and select the `lk-subnet` subnetwork. Under “IP address”, choose “Ephemeral (Custom)” and specify the custom ephemeral IP address to be used by the load balancer frontend. In this example we will use IP address `10.20.0.10`. Under “Ports”, select either “Single”, “Multiple”, or “All” as appropriate, based on the application that traffic is being routed to, and enter any required ports. In this example we will forward traffic on TCP port 80.




8. Move on to the “Review and Finalize” tab and click “Create”.




9. Once the Load Balancer is created, the status will be displayed. As the application (such as httpd) runs on only one of these nodes, you may see the icon, but this behavior is expected.




10. Now the Internal Load Balancer is configured. Once you install the application to protect on node-a, you can connect to it through the frontend IP (10.20.0.10) of the ILB you have just configured. The following example shows how to check the current target for http traffic via the ILB.

```
[gcp-user@node-c ~]$ curl http://10.20.0.10/
<html>
<body>
Test Page from node-a
</body>
</html>
```

# Enabling Load Balancer Traffic from Backend Servers

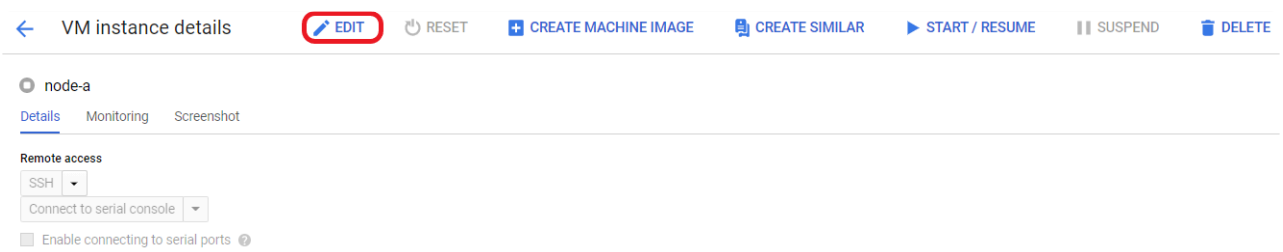
In certain configurations, two applications which are hosted on different backend servers of an internal load balancer may need to communicate through the frontend IP of the load balancer itself. This happens, for example, when using internal load balancers to manage floating IP failover for ASCS and ERS instances in an SAP AS ABAP deployment.

As a consequence of how internal load balancers are implemented in Google Cloud, traffic that is sent from a backend server of a load balancer to the frontend IP of the load balancer will always be routed back to the same backend server that it originated from, regardless of whether it is considered healthy or not by the load balancer's health checks. See the **Traffic is sent to unexpected backend VMs** section of Troubleshooting Internal TCP/UDP Load Balancing for more details.

The following steps describe a workaround for this behavior. These steps must be performed on all VM's which will be added as backend targets for the load balancer.

> ! This workaround applies only to VMs which were created using the Google Cloud "RedHat Enterprise Linux 8.2 for SAP Applications" boot image. Due to networking differences between boot images, the following script may cause unexpected behavior on VMs running different operating systems.

1. In the Google Cloud Console, click the VM instance and select **Edit** from the top menu.




2. In the **Custom metadata** section, add a new key-value pair with key "startup-script". Add the contents of the following script to the value field:

```
#!/bin/bash

nic0_mac="$(curl -H "Metadata-Flavor:Google" \
http://169.254.169.254/computeMetadata/v1/instance/network-interfaces/0/mac)"
```

```
for nic in $(ls /sys/class/net); do
        nic_addr="$(cat /sys/class/net/$nic/address)"
        if [ "$nic_addr" == "$nic0_mac" ]; then
                nic0_name="$nic"
                break
        fi
done

if [ -n $nic0_name ]; then
        if ! grep -q "net.ipv4.conf.${nic0_name}.accept_local=1" /etc/sysctl.c
onf; then
                echo "net.ipv4.conf.${nic0_name}.accept_local=1" >> /etc/sysct
l.conf
                sysctl -p
        fi

        ip rule add pref 0 from all iif ${nic0_name} lookup local
        ip rule del from all lookup local
        ip route add local 127.0.0.0/8 dev lo proto kernel \
                scope host src 127.0.0.1 table main
        ip route add local 127.0.0.1 dev lo proto kernel \
                scope host src 127.0.0.1 table main
        ip route add broadcast 127.0.0.0 dev lo proto kernel \
                scope link src 127.0.0.1 table main
        ip route add broadcast 127.255.255.255 dev lo proto kernel \
                scope link src 127.0.0.1 table main
fi
```

3. Click **Save** at the bottom of the page to save the changes.

4. Once the changes have been saved successfully, reboot the VM to allow the changes to take effect.

See the **Test GenLB Resource Switchover and Failover** section of Responding to Load Balancer Health Checks for details on how to test this behavior.

# 11.2.7.2.6. Responding to Load Balancer Health Checks

The **LifeKeeper Generic Application Recovery Kit for Load Balancer Health Checks (“GenLB Recovery Kit”)** may be used as part of a LifeKeeper resource hierarchy to help route load balancer traffic to the cluster node where a particular resource is currently in-service. This is achieved by maintaining a listener on a user-specified TCP port on the cluster node where the resource is in-service.

## Install the GenLB Recovery Kit

The GenLB Recovery Kit is only supported on SIOS Protection Suite for Linux version 9.5.1 and later, and an rpm installation file can be obtained from the same FTP directory where the SIOS Protection Suite for Linux installation media was downloaded. The filename format is `steeleye-lkHOTFIX-Gen-LB-PL-7172-x.x.x-xxxx.x86_64.rpm`. No additional SIOS licenses are required in order to create GenLB resources.

1. Install the rpm:

```
[root@node-a ~]# rpm -ivh steeleye-lkHOTFIX-Gen-LB-PL-7172-x.x.x-xxxx.x86_64.r
pm
Verifying...                          ########################## [100%]
Preparing...                          ########################## [100%]
Updating / installing... 1:steeleye-lkHOTFIX-Gen-LB-PL-7172-# [100%]
```

2. Verify that the GenLB resource action scripts have been successfully installed to the /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172 directory.

```
[root@node-a ~]# ls -l /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/
total 12
-r-x------ 1 root root 2579 Jan 01 00:00 quickCheck
-r-x------ 1 root root 1734 Jan 01 00:00 remove.pl
-r-x------ 1 root root 3909 Jan 01 00:00 restore.pl
```

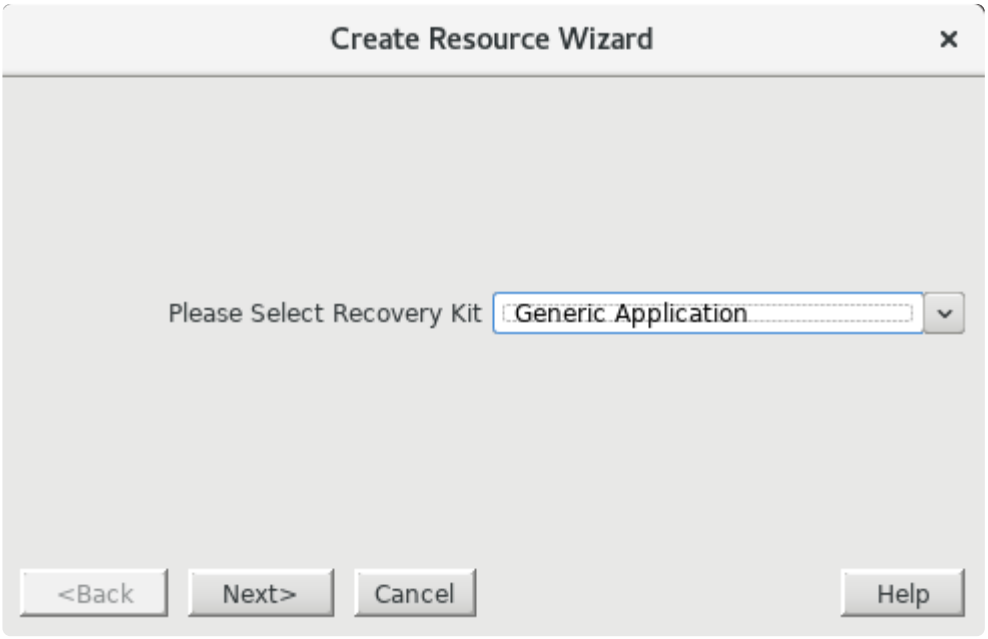
## Create a GenLB Resource

In this example we will create a sample GenLB resource on server node-a listening on TCP port 54321.

> **!** Make sure to modify the server, port, and/or resource tag name as appropriate for the load balancer health check that this resource is responding to. For example, if you have set up a health check which is probing TCP port 41098 on each backend server, then the GenLB resource will contain “41098” in its application info field rather than the example value “54321” given here. When choosing a port to use for a GenLB resource, any open port in the range 1024-65535 may be used.

**Note:** In order for the resource to come in-service successfully, the load balancer must be actively sending health check probes to the server during the resource creation process. This means that the load balancer and health checks must be fully configured before the GenLB resource is created. See the subsections under Switching Between Nodes in a Cloud Environment for more details on how to create internal load balancers in Microsoft Azure and Google Cloud environments.

1. In the LifeKeeper GUI, click to open the **Create Resource Wizard**. Select the “Generic Application” Recovery Kit.



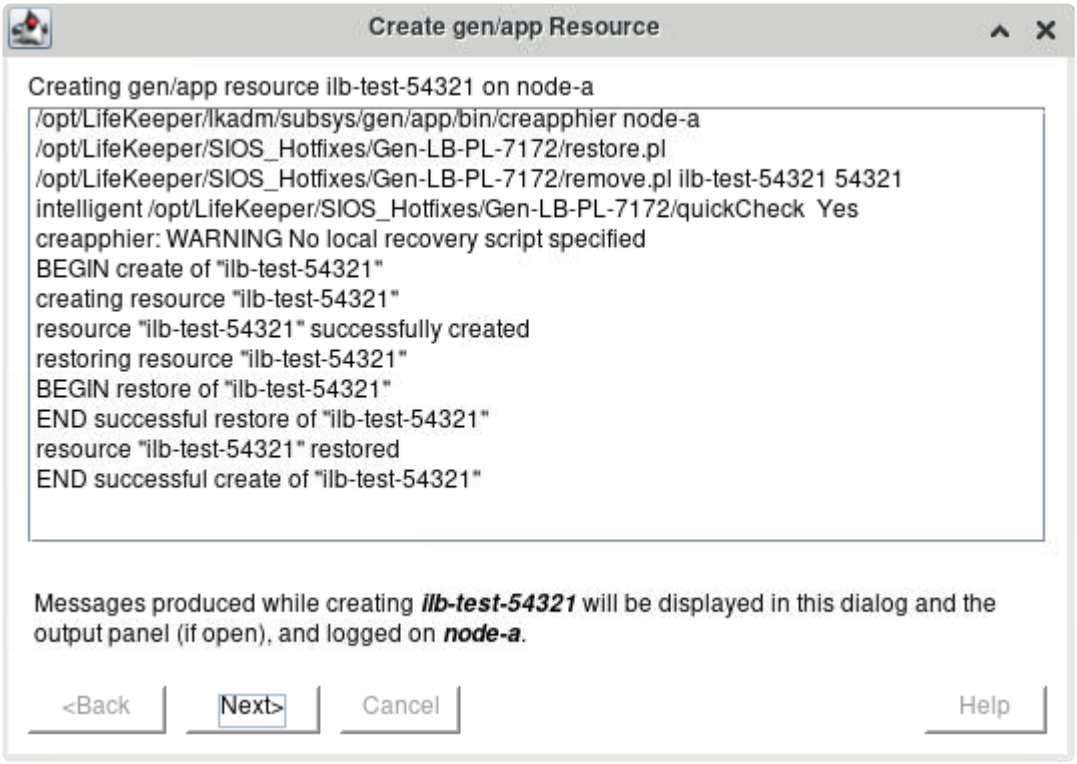

2. Enter the following values into the **Create Resource Wizard** and click **Create** when prompted.

   The icon indicates that the default option is chosen. The “Application Info” field has the format “<TCP Port> <Response>”. The response is optional, and no whitespace is allowed in the response text.

| Field | Value |
|---|---|
| Switchback Type | Intelligent |
| Server | node-a |
| Restore Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/restore.pl |
| Remove Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/remove.pl |
| QuickCheck Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/quickCheck |
| Local Recovery Script | None (Empty) |
| Application Info | 54321 |

| | |
|---|---|
| Bring Resource In Service | Yes |
| Resource Tag | ilb-test-54321 |

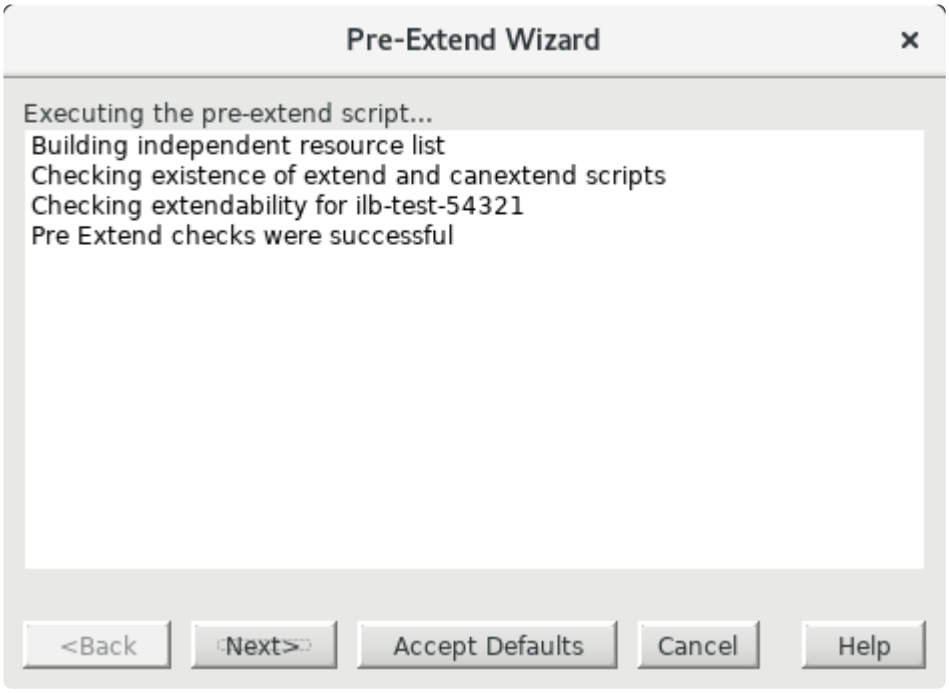
Once the resource has been created and brought in-service successfully, click **Next>** to proceed to the **Pre-Extend Wizard**.




3. Enter the following values into the **Pre-Extend Wizard**. The ✓ icon indicates that the default option is chosen.

| Field | Value |
|---|---|
| Target Server | node-b ✓ |
| Switchback Type | Intelligent ✓ |
| Template Priority | 1 ✓ |
| Target Priority | 10 ✓ |

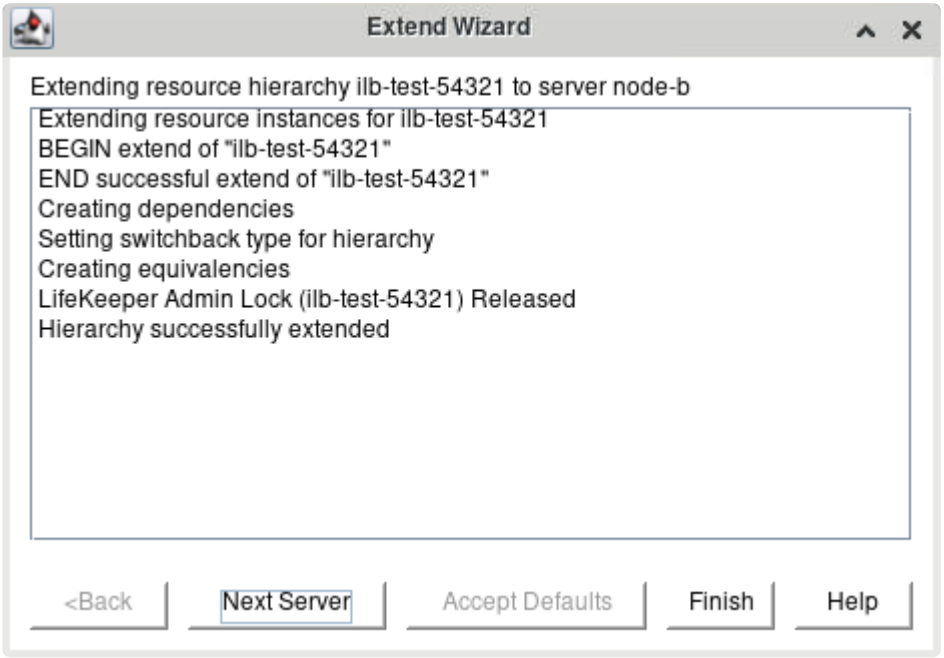
Once the pre-extend checks have passed, click **Next>** to proceed to the **Extend gen/app Resource Hierarchy Wizard**.




4. Enter the following values into the **Extend gen/app Resource Hierarchy Wizard** and click **Extend** when prompted. The icon indicates that the default option is chosen.

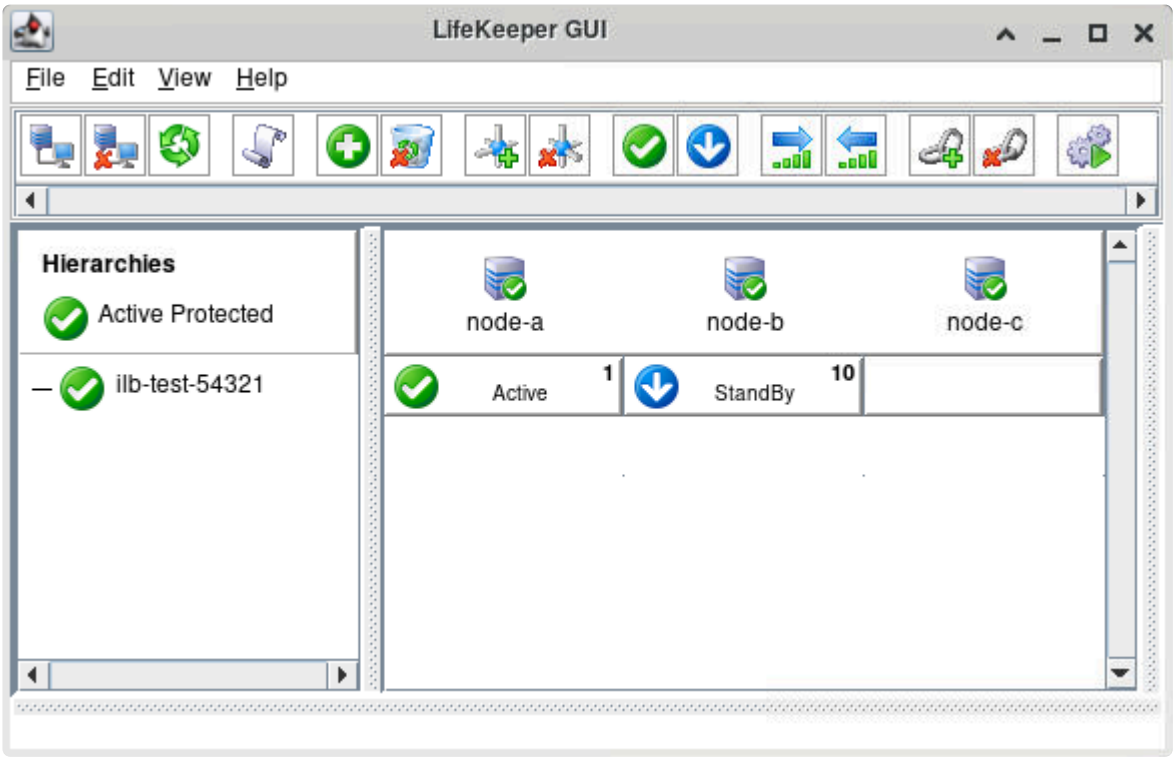
| Field | Value |
|---|---|
| Resource Tag | ilb-test-54321 |
| Application Info | 54321 |

Once the resource has been extended successfully, click **Finish**.




5. Back in the LifeKeeper GUI, we see that the newly created **ilb-test-54321** resource is Active on

node-a and Standby on node-b. In this state, a TCP load balancer with a TCP health check on port 54321 will treat node-a as healthy and node-b as unhealthy, causing all load balancer traffic to be routed to node-a. When placed in a resource hierarchy with a protected application, this resource will ensure that load balancer traffic is always routed to the server on which the application is currently running.




## Test GenLB Resource Switchover and Failover

In this section we will assume that we have created an internal load balancer with node-a and node-b as backend targets which has the following properties:

- Front-end internal IP: 10.20.0.10
- TCP health check on port 54321

and that the **ilb-test-54321** GenLB resource that was created in the previous section is currently Active on node-a.

For convenience we will set up a temporary Apache web server that will simply return the hostname of each server. Execute the following commands **on both node-a and node-b**. Adjust the commands accordingly (e.g., to use `zypper install`) if installing on a SLES server.

```
# yum install -y httpd
# systemctl start httpd
# echo $(hostname) > /var/www/html/index.html
```

Before continuing, verify that traffic is allowed on TCP port 80 for node-a and node-b.

We will now test the switchover and failover capabilities of the **ilb-test-54321** GenLB resource.

1. With the **ilb-test-54321** resource Active on node-a and Standby on node-b, verify the output of the following command on each server.

```
[root@node-a ~]# curl http://10.20.0.10
node-a
[root@node-b ~]# curl http://10.20.0.10
node-a
```

> **Note:** If the curl command returns 'node-b' when run on node-b and the servers are located in Google Cloud, see the **Enabling Load Balancer Traffic From Backend Servers** section of GCP – Using an Internal Load Balancer. Once the workaround from that section has been implemented, return here to verify the expected behavior.

2. Execute the following command on node-a:

   ```
   [root@node-a ~]# while true; do curl http://10.20.0.10; sleep 1; done
   ```

   and initiate a switchover of the **ilb-test-54321** resource to node-b. Once the switchover has completed successfully, use Ctrl-C (SIGINT) to terminate the running command on node-a.

   The output of the command should be similar to:

   ```
   …
   node-a
   node-a
   node-a
   [switchover occurs]
   node-b
   node-b
   node-b
   …
   ```

   In particular, the load balancer should cleanly stop routing traffic to node-a before beginning to route it to node-b. If the output near the switchover point looks like the following:

   ```
   …
   node-a
   [switchover occurs]
   node-b
   node-a
   node-b
   node-a
   node-b
   node-a
   node-b
   ```

```
node-b
node-b
…
```

then you may need to edit the health check properties to decrease the time between health check probes and/or decrease the minimum number of unsuccessful health check probes before a backend instance is marked unhealthy and removed from the load balancer pool.

3. With the **ilb-test-54321** resource Active on node-b, execute the following command on node-a:

```
[root@node-a ~]# while true; do curl http://10.20.0.10; sleep 1; done
```

and forcefully reboot node-b to initiate a failover of the **ilb-test-54321** resource back to node-a:

```
[root@node-b ~]# echo b > /proc/sysrq-trigger
```

After the failover has completed successfully, use Ctrl-C (SIGINT) to terminate the running command on node-a.

The output of the command on node-a should be similar to:

```
…
node-b
node-b
node-b
[failover occurs]
node-a
node-a
node-a
…
```

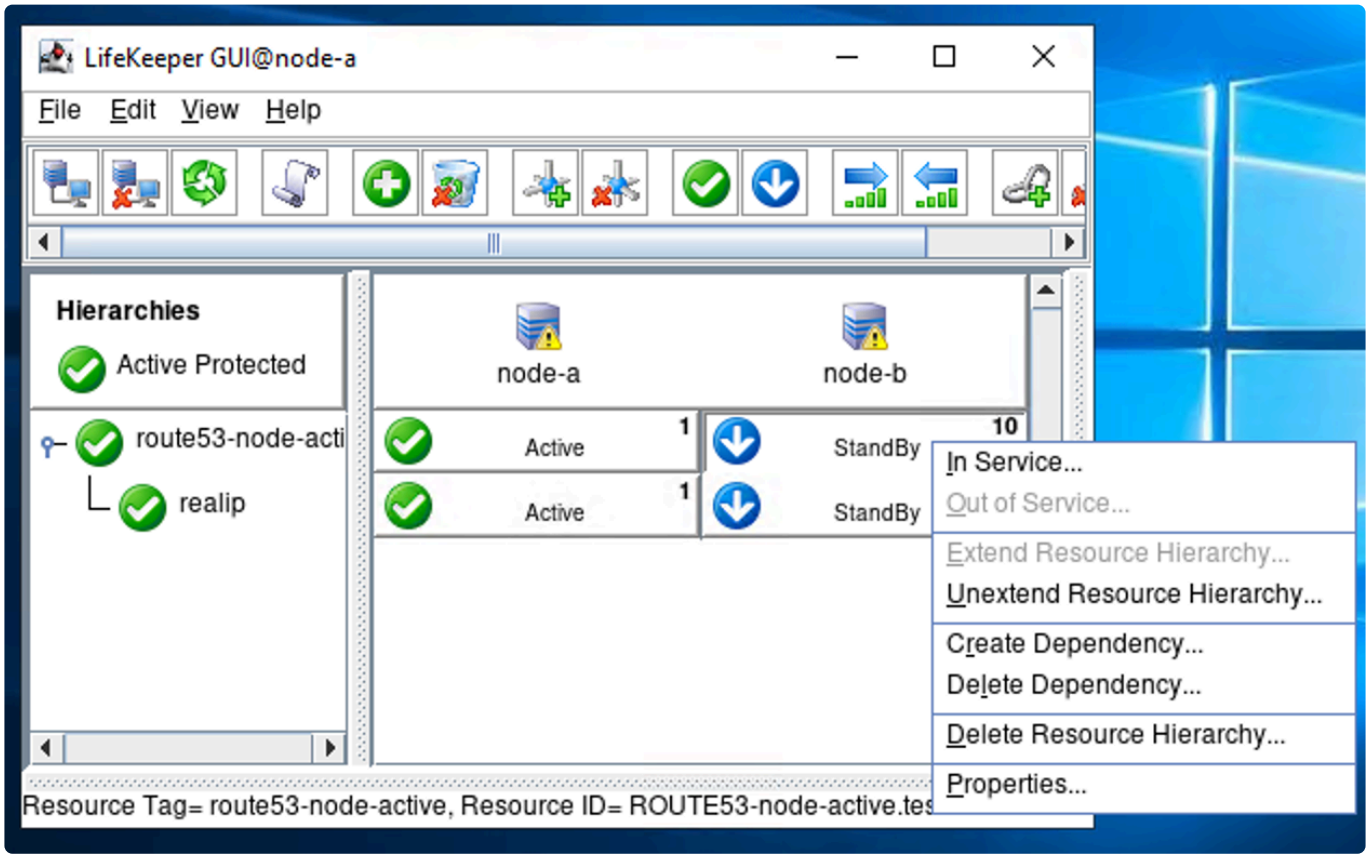
At this point basic verification of the GenLB resource behavior is complete. Execute additional tests as necessary to verify the interaction between the GenLB resource and your protected application on switchover and failover. Once finished testing the GenLB resource functionality, the temporary Apache web servers may be removed by executing the following commands on both node-a and node-b:

```
# systemctl stop httpd
# rm -f /var/www/html/index.html
# yum remove -y httpd
```

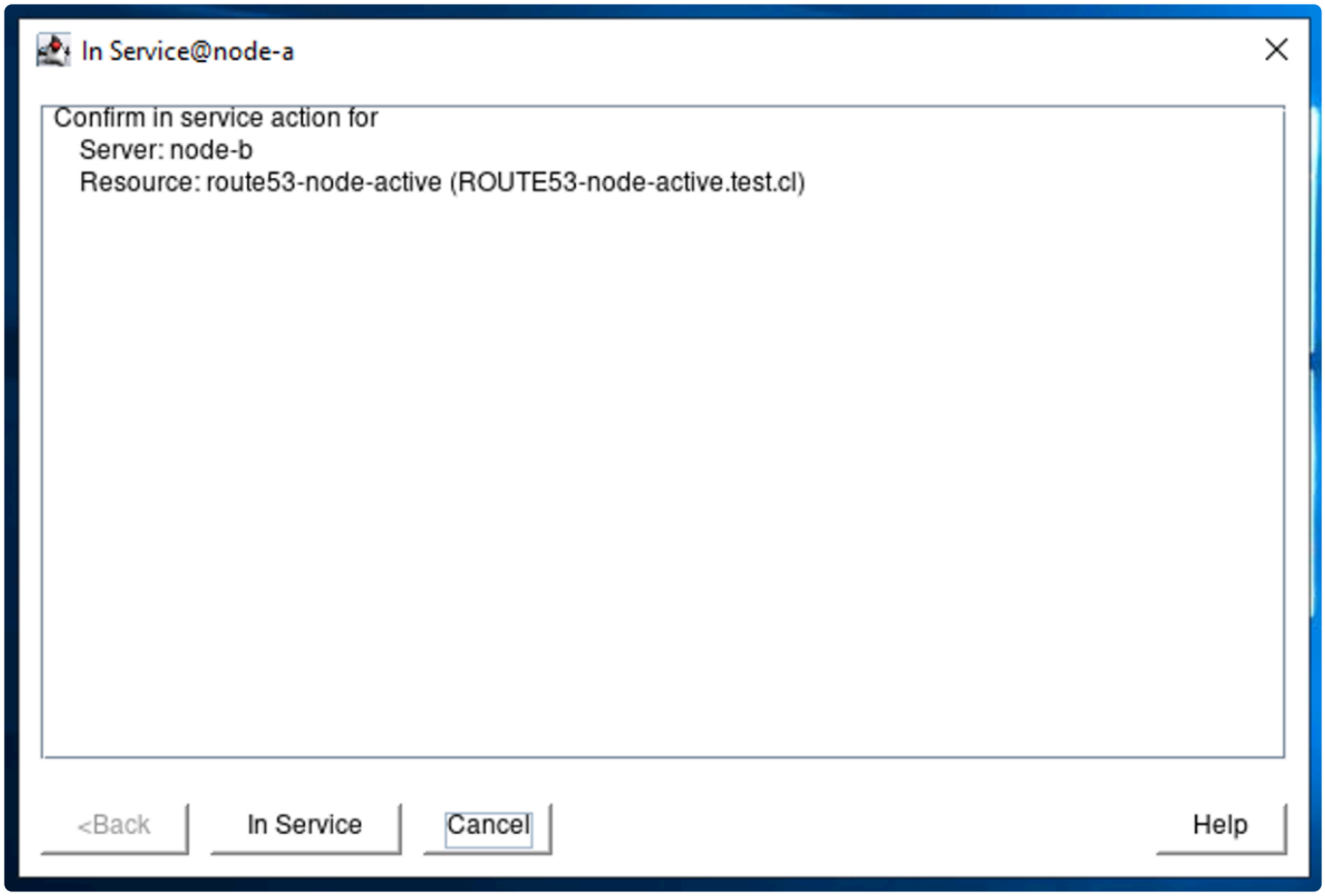
# 11.2.7.3. Switch to Standby Node to Confirm Switchover is Working

It is recommended to test each cluster resource as it is created. This helps to identify misconfigurations or other issues early in the cluster creation process.

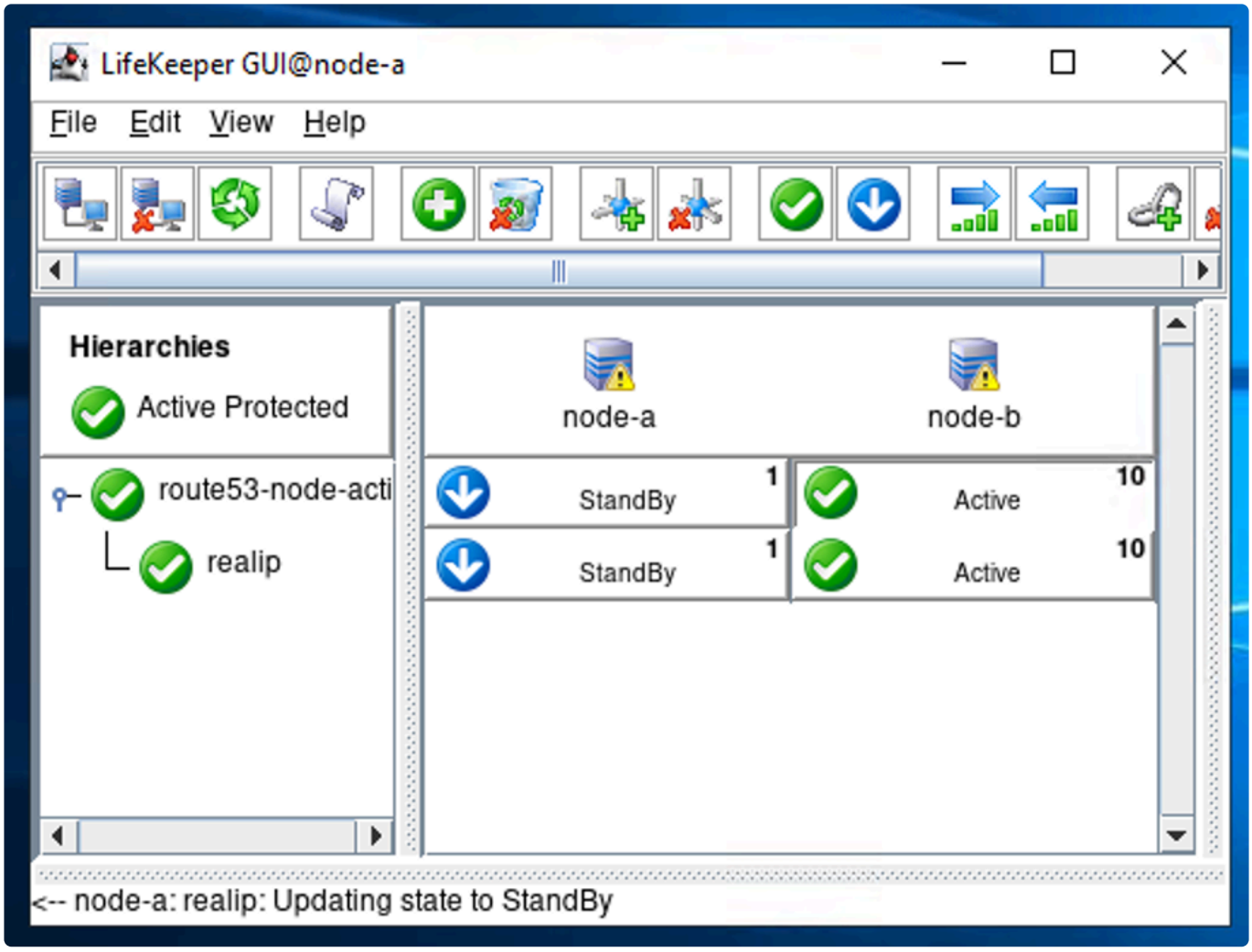
In the LifeKeeper User Interface, right-click a resource on the standby node and select “In Service…” from the context menu. This forces a switchover to the standby node.




Once the switchover is confirmed, LifeKeeper will switch the resource (including any dependent resources) to the standby node.




Once the switchover has completed successfully, node-b will be listed as the active node.




Switch back to node-a once you confirm that the switchover works.

# 11.2.7.4. How to Create Data Replication of a File System

As discussed in How Does Data Replication between Nodes Work, DataKeeper creates a NetRAID Device which works as RAID1 device.

This guide uses the following parameters as examples. Replace these parameters based on the local environment that DataKeeper is being installed on.

| Item | Value |
|---|---|
| Disk Device Name for Replication | /dev/xvdb (the second storage device attached)<br><br>**Note:** The Storage Device name may vary based on your environment. |
| Mount point | /datakeeper |

1. See one the following topics to prepare a disk for replication.

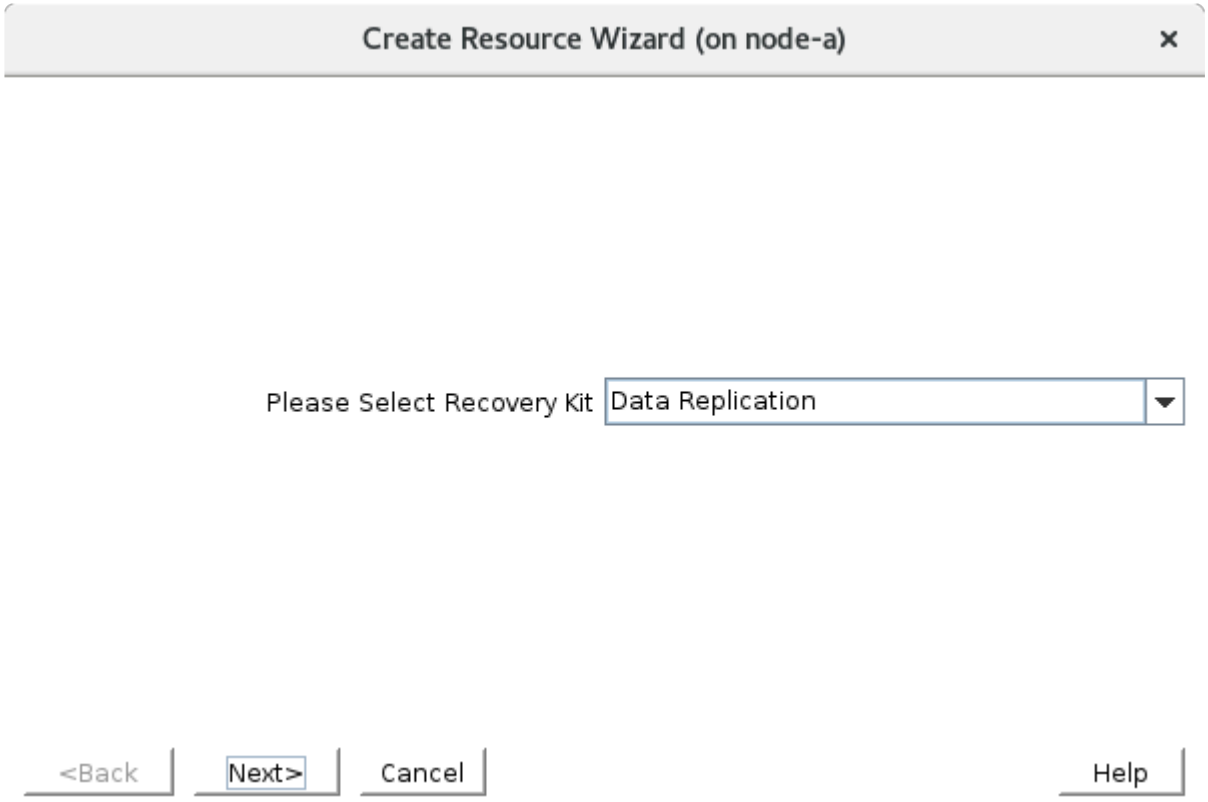
- How to Prepare Disks for Replication on AWS
- How to Prepare Disks for Replication on Azure
- How to Prepare Disks for Replication on GCP

2. Mount the disk on the primary node (node-a).

Be sure to check the name of the storage device to mount (such as `/dev/xvdb1` or `/dev/sdc1`).

```
# mkdir /datakeeper
# mount /dev/xvdb1 /datakeeper
```

3. On the LifeKeeper User Interface, define a new resource. Select to start the Create Resource Wizard @ node-a. Select “Data Replication” as the Recovery Kit.

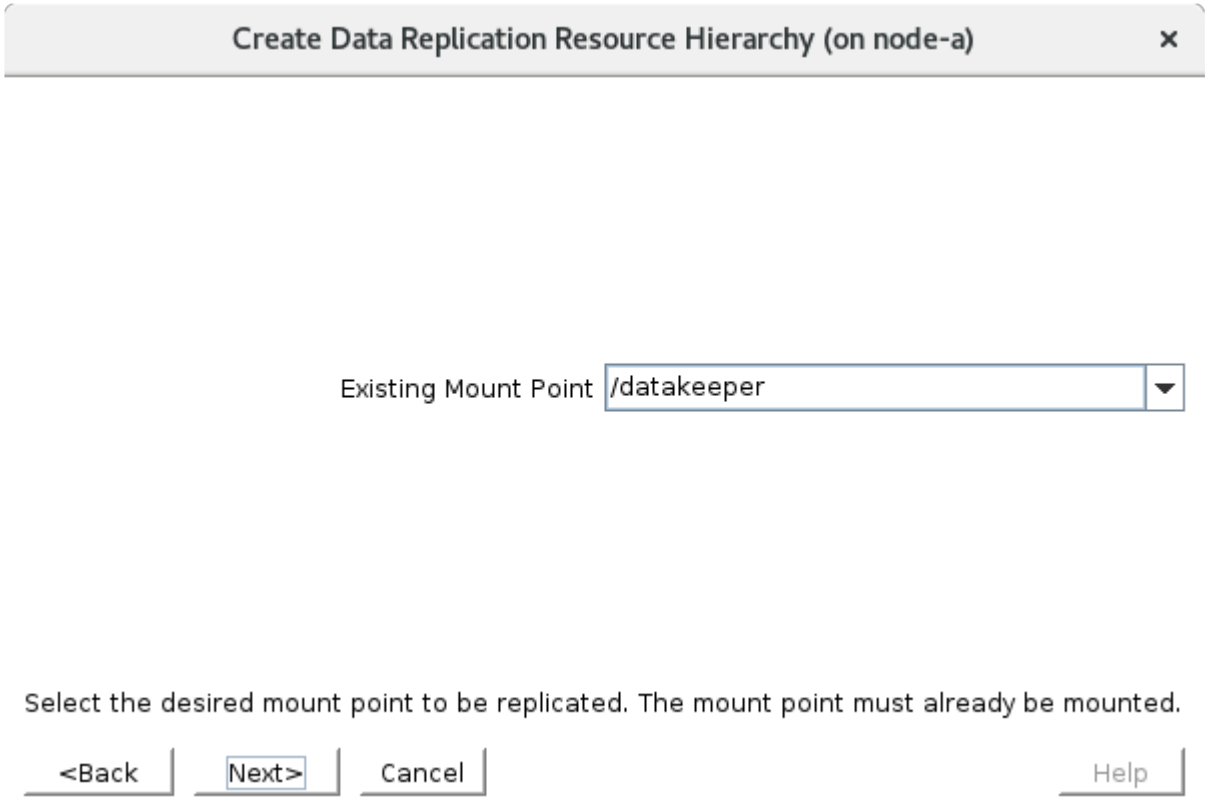



4. Enter the following values. Select the default values.

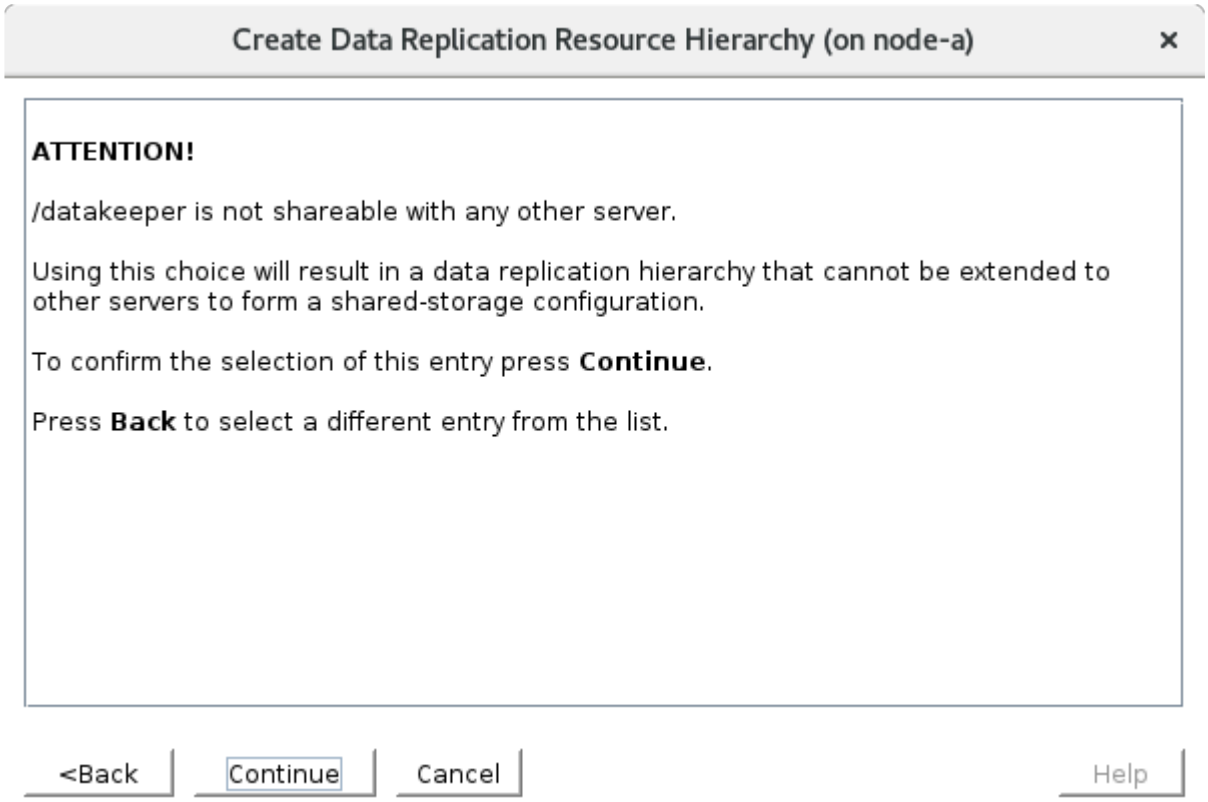
> **Note:** For items with a checkmark ( ) review the default values and use the value suggested.

| Field | Value |
|---|---|
| Switchback Type | intelligent |
| Server | node-a |
| Hierarchy Type | Replicate Existing Filesystem |

5. Select Mount Point `/datakeeper`. The wizard will have already scanned the system and will show `/datakeeper` as the candidate for this field.




6. The wizard asks for confirmation of the selection. Select ‘Continue’ to move on.

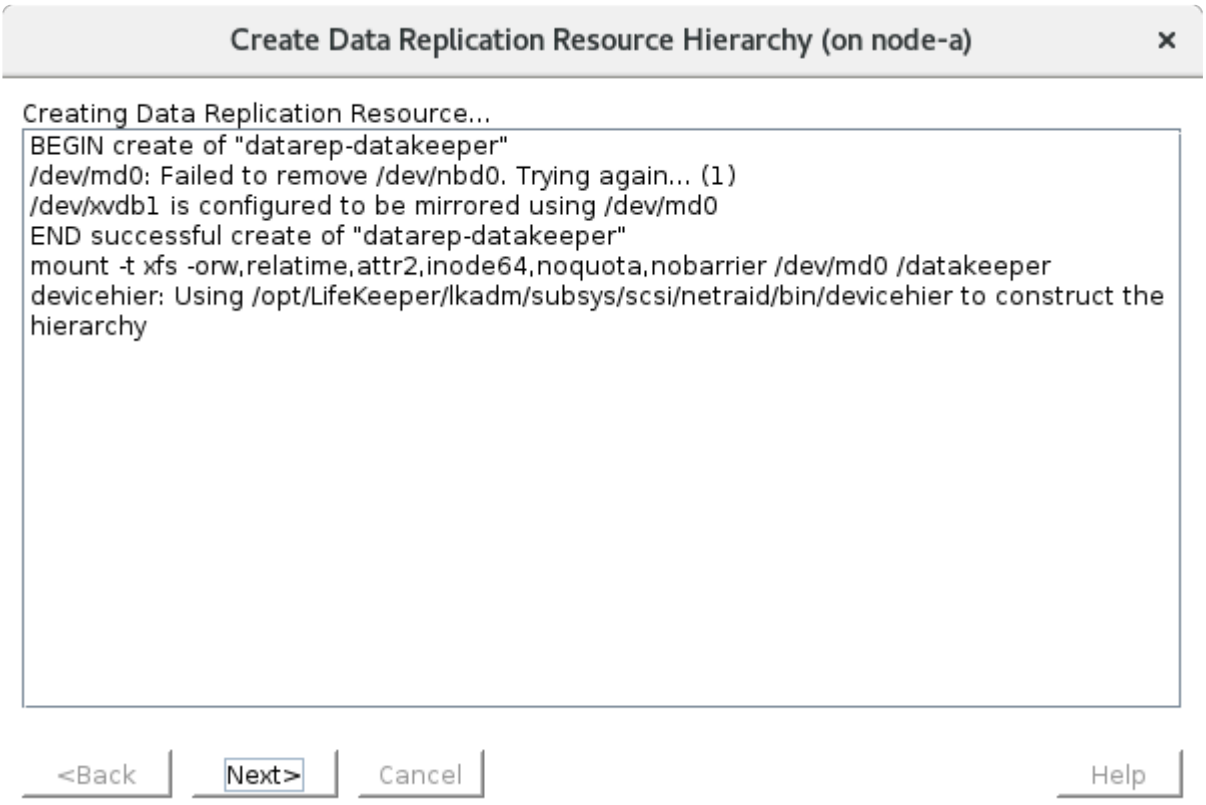



7. Enter the following values in the fields. Select the default values.

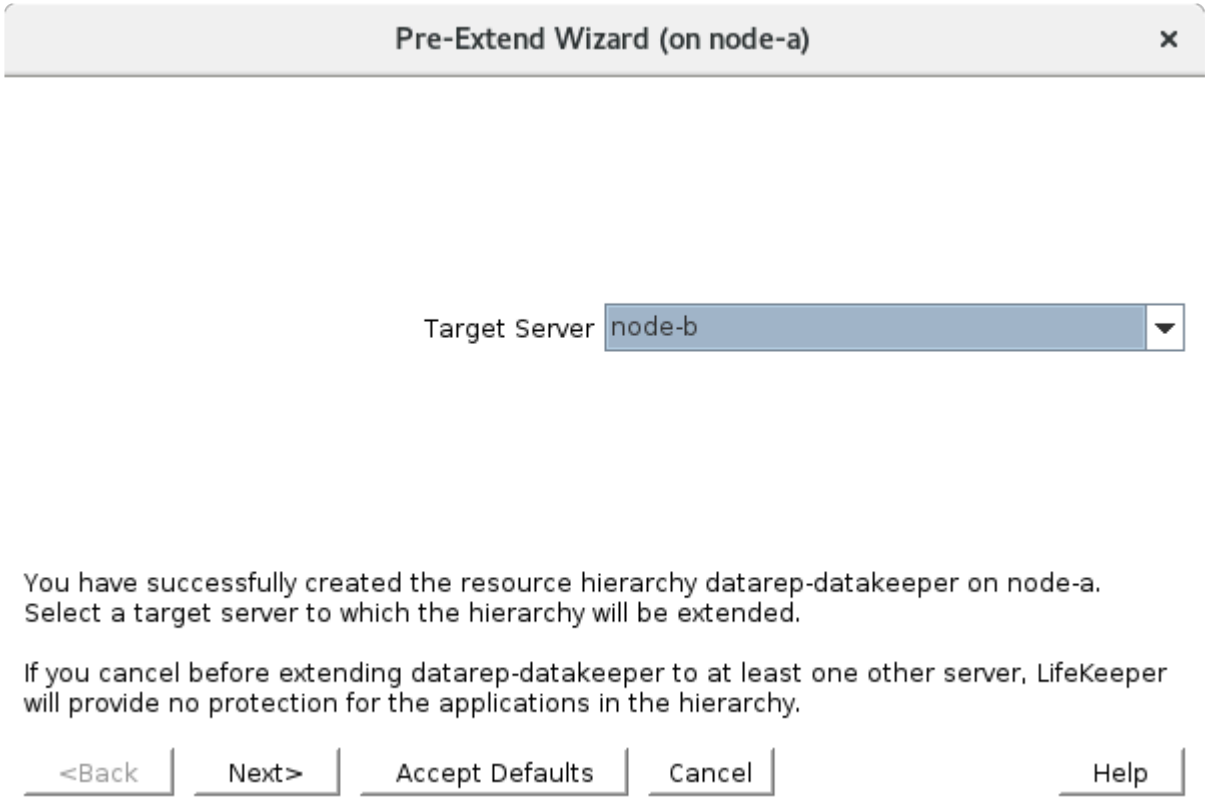
| Field | Value |
|---|---|

| | |
|---|---|
| Data Replication Resource Tag | datarep-datakeeper |
| File System Resource Tag | /datakeeper |
| Bitmap File | /opt/LifeKeeper/bitmap__datakeeper |
| Enable Asynchronous Replication? | no |

8. The ‘Create Data Replication Resource Hierarchy’ wizard displays these values. Select “Next” to continue.



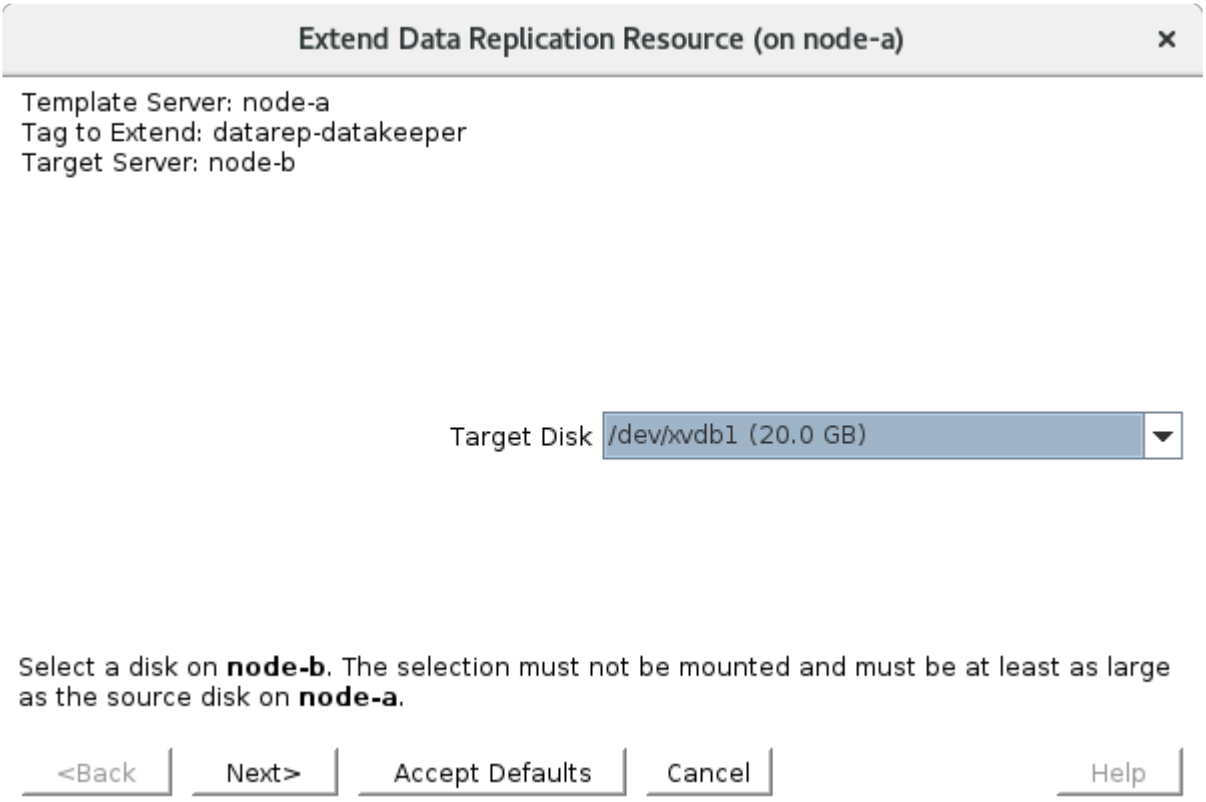

9. The next step is the “Pre-Extend Wizard”. Select the default values.




| Field | Value |
|---|---|
| Switchback Type | intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

Once the checks in the ‘Pre-Extend Wizard’ have passed, select ‘Next’ to continue.

10. On the “Extend gen/filesys Resource Hierarchy” wizard, the first choice is selecting the disk as the standby node. Select the disk from the dropdown list.

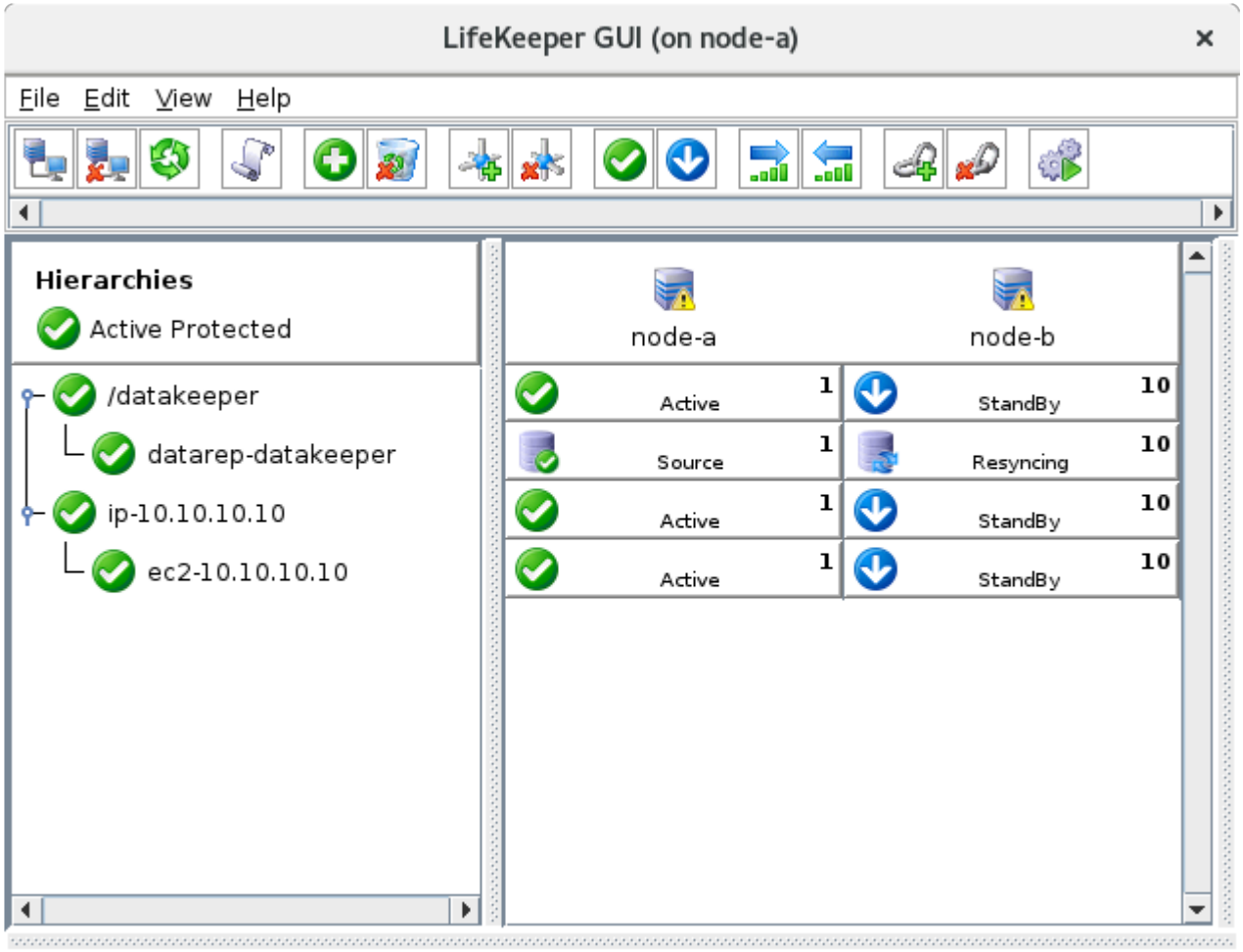



11. On the following pages, select the default values.

| Field | Value |
|---|---|
| Mount point | /datakeeper |
| Root Tag | /datakeeper |
| Target Disk | /dev/xvdb1 (20G) |
| Data Replication Resource Tag | datarep-datakeeper |
| Bitmap File | /opt/LifeKeeper/bitmap__datakeeper |
| Replication Path | 10.20.1.10/10.20.2.10 |

Once the ‘Extend Wizard’ successfully completes, select ‘Finish’.

Now the disk is configured to replicate the data. As seen below, a full initial sync of the data is required. Once the data is synced, the label will change to “Target”.


LifeKeeper GUI (on node-a)
File Edit View Help
Hierarchies
Active Protected
/datakeeper
datarep-datakeeper
ip-10.10.10.10
ec2-10.10.10.10
node-a
node-b
Active 1
StandBy 10
Source 1
Resyncing 10
Active 1
StandBy 10
Active 1
StandBy 10

# 11.2.7.4.1. How to Prepare Disks for Replication on AWS

> The following steps must be performed on each node.

Before setting up disk replication using DataKeeper, the disks must be prepared for replication. Please note that this tutorial assumes the availability of an empty disk for each node, both of equal size.

This guide uses the following parameters as examples. You will need to replace these parameters based on the local environment DataKeeper is being installed on.

| Items | Value |
|---|---|
| Disk Device Name for Replication | /dev/xvdb (as the second disk) |
| Mount point | /datakeeper |

## Create `/datakeeper` Folder

```
# mkdir /datakeeper
```

> **Note**: After performing `mkdir /datakeeper`, **change ownership to oracle.oinstall** so that Oracle
> setup can actually create database files. **Run command**: `sudo chown oracle.oinstall /datakeeper`

## Check the Available Disks on node-a

Confirm the disk to mount using `lsblk`. As seen in the screenshot below, the storage device `xvdb` is not mounted to the instance yet.

```
[root@node-a media]# lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    10G  0 disk
├─xvda1 202:1    0     1M  0 part
└─xvda2 202:2    0    10G  0 part /
xvdb    202:16   0     8G  0 disk
loop0     7:0    0 376.3M  0 loop /media
```

# Use `gdisk` to Create a Partition

Use `gdisk /dev/xvdb` to create a partition on this disk.

Type these commands in the following order:

- `n` to create a new partition
- `<Enter>` to select default (Partition number)
- `<Enter>` to select default (First Sector)
- `<Enter>` to select default (Last Sector)
- `<Enter>` to select default (HEX code)
- `w` to write down the changes
- `y` to confirm the operation

```
# gdisk /dev/xvdb
GPT fdisk (gdisk) version 0.8.10

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries.

Command (? for help): n
Partition number (1-128, default 1):
First sector (34-41943006, default = 2048) or {+-}size{KMGTP}:
Last sector (2048-41943006, default = 41943006) or {+-}size{KMGTP}:
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): Y
OK; writing new GUID partition table (GPT) to /dev/xvdb.
The operation has completed successfully.
```

# Create a File System to the Partition using `mkfs`

```
# mkfs -t xfs /dev/xvdb1
meta-data=/dev/xvdb1             isize=512    agcount=4, agsize=1310655 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=5242619, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

# 11.2.7.4.2. How to Prepare Disks for Replication on Azure

If working in Azure, review the following document describing how to attach a second disk to an instance.

https://docs.microsoft.com/en-us/azure/virtual-machines/linux/attach-disk-portal

## Create `/datakeeper` Folder

```
# mkdir /datakeeper
```

> **Note**: After performing `mkdir /datakeeper`, **change ownership to oracle.oinstall** so that Oracle
> setup can actually create database files. **Run command**: `sudo chown oracle.oinstall /datakeeper`

## Check the Available Disks on node-a

Confirm the disk to mount using `lsblk`. As seen in the screenshot below, the storage device `sdc` is not mounted to the instance yet.

```
[root@node-a ~]# lsblk
NAME              MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                 8:0    0   64G  0 disk
├─sda1              8:1    0  500M  0 part /boot/efi
├─sda2              8:2    0  500M  0 part /boot
├─sda3              8:3    0    2M  0 part
└─sda4              8:4    0   63G  0 part
  ├─rootvg-tmplv  253:0    0    2G  0 lvm  /tmp
  ├─rootvg-usrlv  253:1    0   10G  0 lvm  /usr
  ├─rootvg-optlv  253:2    0    2G  0 lvm  /opt
  ├─rootvg-homelv 253:3    0    1G  0 lvm  /home
  ├─rootvg-varlv  253:4    0    8G  0 lvm  /var
  └─rootvg-rootlv 253:5    0    2G  0 lvm  /
sdb                 8:16   0    4G  0 disk
└─sdb1              8:17   0    4G  0 part /mnt/resource
sdc                 8:32   0    8G  0 disk
```

> Follow the Azure document above to mount it to the instance.

Use `parted` to prepare the partition, then use `mkfs` to prepare the file system.

```
[root@node-a ~]# parted /dev/sdc --script mklabel gpt mkpart xfspart xfs 0% 100%
[root@node-a ~]# mkfs.xfs /dev/sdc1
meta-data=/dev/sdc1              isize=512    agcount=4, agsize=524160 blks
         =                       sectsz=4096  attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=2096640, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=4096  sunit=1 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@node-a ~]# partprobe /dev/sdc1
```

Run `lsblk` again, now `/dev/sdc1` should be available to mount.

```
[root@node-a ~]# lsblk
NAME              MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                 8:0    0   64G  0 disk
├─sda1              8:1    0  500M  0 part /boot/efi
├─sda2              8:2    0  500M  0 part /boot
├─sda3              8:3    0    2M  0 part
└─sda4              8:4    0   63G  0 part
  ├─rootvg-tmplv  253:0    0    2G  0 lvm  /tmp
  ├─rootvg-usrlv  253:1    0   10G  0 lvm  /usr
  ├─rootvg-optlv  253:2    0    2G  0 lvm  /opt
  ├─rootvg-homelv 253:3    0    1G  0 lvm  /home
  ├─rootvg-varlv  253:4    0    8G  0 lvm  /var
  └─rootvg-rootlv 253:5    0    2G  0 lvm  /
sdb                 8:16   0    4G  0 disk
└─sdb1              8:17   0    4G  0 part /mnt/resource
sdc                 8:32   0    8G  0 disk
└─sdc1              8:33   0    8G  0 part
```

Now you are ready to create the Data Replication Resource using LifeKeeper / DataKeeper.

# 11.2.7.4.3. How to Prepare Disks for Replication on GCP

## Create `/datakeeper` Folder

```
# mkdir /datakeeper
```

> **Note**: After performing `mkdir /datakeeper`, **change ownership to oracle.oinstall** so that Oracle
> setup can actually create database files. **Run command**: `sudo chown oracle.oinstall /datakeeper`

## Check the Available Disks on node-a

Confirm the disk to mount using `lsblk`. As seen in the screenshot below, the storage device `sdc` is not mounted to the instance yet.

```
[root@node-a ~]# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda      8:0    0    20G  0 disk
├─sda1   8:1    0   200M  0 part /boot/efi
└─sda2   8:2    0  19.8G  0 part /
sdb      8:16   0    10G  0 disk
```

> Follow the Azure document above to mount it to the instance.

Use `parted` to prepare the partition, then use `mkfs` to prepare the file system.

```
[root@node-a ~]# parted /dev/sdb --script mklabel gpt mkpart xfspart xfs 0% 100%
[root@node-a ~]# mkfs.xfs /dev/sdb1
meta-data=/dev/sdb1              isize=512    agcount=4, agsize=655232 blks
         =                       sectsz=4096  attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=2620928, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=4096  sunit=1 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@node-a ~]# partprobe /dev/sdb1
```

Run `lsblk` again, now `/dev/sdc1` should now be available to mount.

```
[root@node-a ~]# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda      8:0    0    20G  0 disk
├─sda1   8:1    0   200M  0 part /boot/efi
└─sda2   8:2    0  19.8G  0 part /
sdb      8:16   0    10G  0 disk
└─sdb1   8:17   0    10G  0 part
```

Now you are ready to create the Data Replication Resource using LifeKeeper / DataKeeper.

# 11.2.7.5. How to Protect Other Resources (Databases or Applications)

This section includes the basic steps to create various resources such as databases and applications. Each resource can be created independently (unless noted otherwise).

- Protecting an Oracle Resource
- Protecting MSSQL Using Quick Service Protection
- Protecting a PostgreSQL Resource
- Protecting an NFS Resource
- Protecting an SAP Resource
- Protecting an SAP HANA Resource

This guide uses different types of IP Resources to demonstrate different patterns. Different IP resource tags may be seen including:
- ip-10.10.10.10
- ip-10.20.0.10
- ip-10.20.10.100
- realip

Although different tags (based on how IP resources are configured) may be seen, these differences don’t affect how each application or service (such as Oracle) are protected.

# 11.2.7.5.1. Protecting an Oracle Resource (non-PDB)

This section outlines the steps to protect Oracle Resources (non-PDB).

To protect Oracle resources, the data needs to be replicated across nodes using DataKeeper (unless shared storage or SAN devices are used).

The following table outlines the location of each piece.

| Items | Location |
|---|---|
| Oracle 19c (ORACLE_HOME) | /u01/app/oracle/product/19.3.0/dbhome_1 |
| Database (SID: ORCL) | /datakeeper/oradata/ORCL |

As discussed earlier in How to Create Data Replication of a File System, this guide uses `/datakeeper` to replicate data between nodes. Therefore, the data on `/datakeeper/oradata/ORCL` is also replicated between nodes.

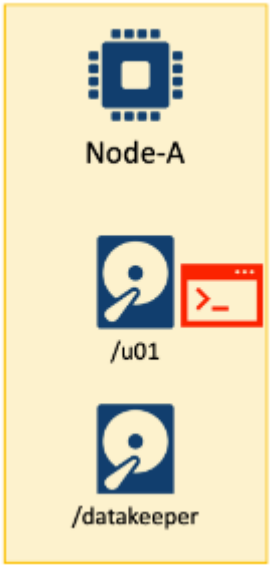
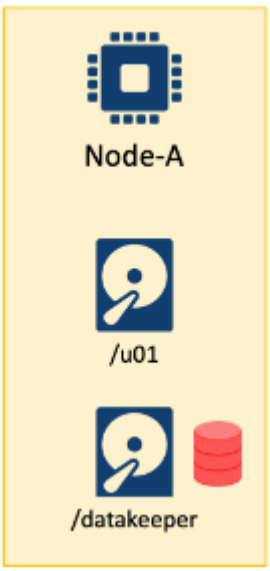
Please note that this guide uses computing resources with the following devices attached (Oracle resources require more space than other resources even for evaluation purposes).

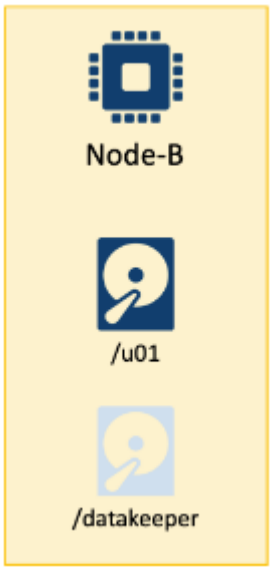
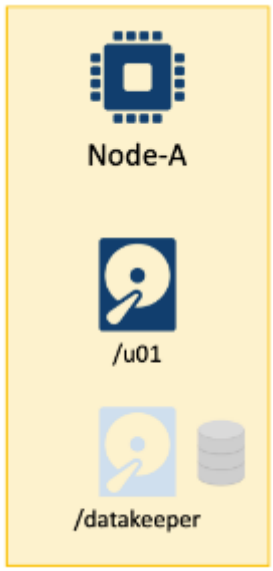
| Resource | Required Size |
|---|---|
| Boot Disk (such as /dev/sda/) | 30 Gb |
| Data Disk (such as /dev/sdb/, configured and mounted at `/datakeeper`) | 20 Gb |
| Memory | 2 Gb |

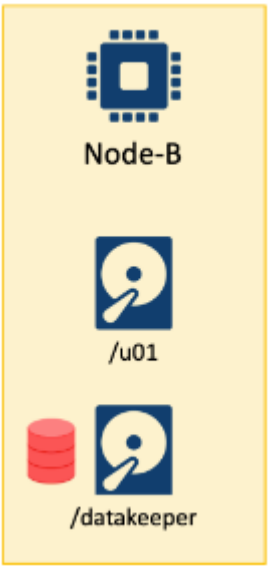
## Configure the Oracle Resource

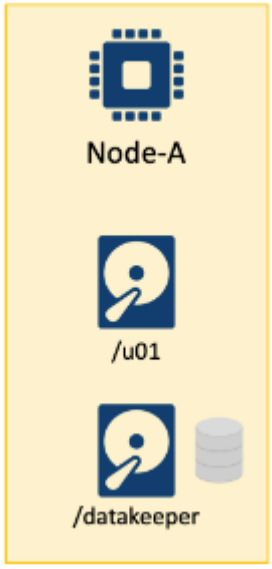
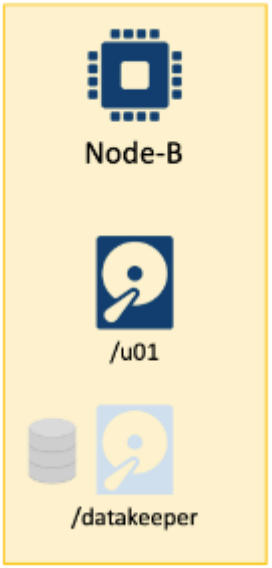
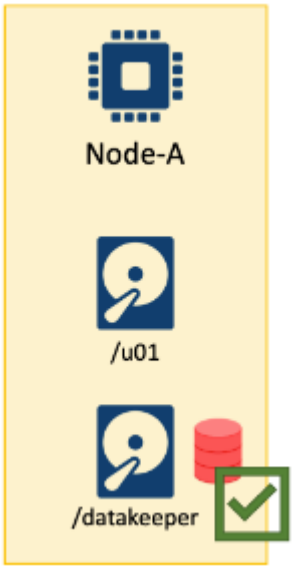
The following table outlines the general steps to configure an Oracle Resource. The red ‘stacked disk’ shaped icons indicate a node that has Oracle Database instances at the time of each step. The grey ‘stacked disk’ icons indicate Oracle Databases that are not running.

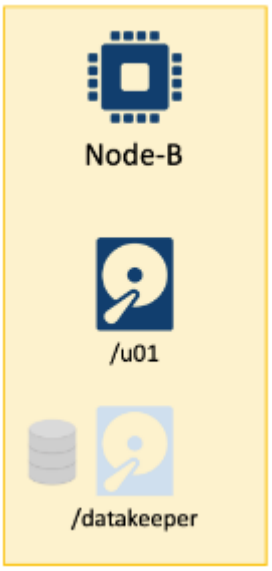
Also, the grey “storage” icon indicates that the storage `/datakeeper` is not available for the node.

| | |
|---|---|
| **Install Oracle 19c on both nodes**<br><br>• Install Oracle on Your System |  <br> |
| **Create Oracle Database on Node-A**<br><br>• Review How to Confirm if the Data Storage is Available on a Node. The data storage should be available on node-a.<br><br>• Create an Oracle Database on node-a<br><br>• Stop Oracle Instance on node-a |  |

<table>
<tr><td></td><td> 
</td></tr>
<tr><td>Create Oracle Database on Node-B<br><br>• Switchover (the data storage) to node-b<br><br>• Rename /datakeeper/oradata/ORCL<br><br>• Create an Oracle Database on node-b<br><br>• Stop Oracle Instance on node-b</td><td> 


</td></tr>
</table>

| | |
|---|---|
| **Configure Oracle LISTENER on both nodes**<br>• Switchover (the data storage) back to node-a<br>• Update Config File for Oracle Listener on Both Nodes |  <br>  |
| **Complete Configuration using LifeKeeper**<br>• Start Database and Listener on node-a<br>• Configure resources on LifeKeeper<br>° Configure Oracle LISTENER Resource<br>° Configure Oracle Resource<br>• Test Switchover of the Oracle Resource |  |

| | |
|---|---|
| |  Node-B<br>/u01<br>/datakeeper |

# 11.2.7.5.1.1. Install Oracle

Install Oracle 19c on both nodes.

Install Oracle on a Linux system.

- Refer to the Database Installation Guide for Linux

Install Oracle 19c at the following location:

```
/u01/app/oracle/product/19.3.0/dbhome_1
```

The location should be specified in the environment variable `ORACLE_HOME`. Once you start the Oracle installer, specify the following values on the installer wizard pages.

| Step # | Item | Value ( ✓ indicates accepting default value) |
|---|---|---|
| 1 | Install Option | Set Up Software Only |
| 2 | Type of database installation | Single Instance database installation |
| 3 | Database Edition | Standard Edition 2 |
| 4 | Oracle Base | `/u01/app/oracle` ✓ |
| 5 | Inventory Location | `/u01/app/oraInventory` ✓ |
| 6 | Privileges Operating System Groups | ✓ |
| 7 | Root Script Execution Configuration | Enter “root” user credential |
| 8 | Perform Prerequisite Checks | Fix if the wizard indicates important issue ✓ |
| 9 | Summary | Select “Install” ✓ |
| 10 | Install Product | Wait for install ✓ |
| 11 | Finish | Install should complete with no errors ✓ |

# 11.2.7.5.1.2. Create an Oracle Database (non-PDB)

Be sure to perform these steps on the nodes indicated in the documentation. Also, confirm if the data storage is available on the node.

1. On node-a, start installer as follows:

```
$ su - oracle
Password:
Last login: Sun Dec 20 04:33:22 UTC 2020 on pts/1
$ cd $ORACLE_HOME
$ ./bin/dbca
```

2. Once the installer starts, complete the following steps:

| Step # | Item | Value ( ✓ indicates accepting default value) |
|---|---|---|
| 1 | Select Operation | Create a database ✓ |
| 2 | Creation Mode | Advanced Configuration |
| 3 | Deployment Type | Database Type: Oracle Single Instance Database<br>Template: General Purpose or Transaction Processing |
| 4 | Database Identification | orcl ✓ |
| 5 | Storage Option | Storage Type: File System<br>Location: `/datakeeper/oradata` |
| 6 | Fast Recovery Option | Select nothing ✓ |
| 7 | Network Configuration | Create a new Listener (Name: LISTENER1) |
| 8 | Configuration Option | Leave as default values ✓ |

| | | |
|---|---|---|
| 9 | Management Option | Leave as default values |
| 10 | User Credentials | Select “Use the same administrative password for all accounts” and then Specify Password |
| 11 | Creation Option | Create database |
| 12 | Summary | Confirm the values |
| 13 | Progress Page | Watch for progress |
| 14 | Finish | Confirm the result |

# 11.2.7.5.1.3. Stop the Oracle Instance

Ensure these steps are completed on the correct node.

## Shutdown the Database

To shutdown the database, execute `sqlplus` and run the `shutdown immediate` command as follows:

```
[oracle]$ $ORACLE_HOME/bin/sqlplus / AS SYSDBA

SQL*Plus: Release 19.0.0.0.0 - Production on Sun Dec 20 05:50:40 2020
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Standard Edition 2 Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

## Confirm the Database can be Restarted Later

The following is not required. However, it is recommended that the database be restarted using the `startup` command and then shut down with the `shutdown immediate` command.

```
SQL> startup
ORACLE instance started.

Total System Global Area  771748536 bytes
Fixed Size             8901304 bytes
Variable Size                490733568 bytes
Database Buffers     268435456 bytes
Redo Buffers                   3678208 bytes
Database mounted.
Database opened.
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Standard Edition 2 Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
```

# 11.2.7.5.1.4. Rename /datakeeper/oradata/ORCL

So far the following tasks have been completed:

1. Created the database on Node-A
2. Switched over the `datarep-datakeeper` resource to Node-B

In step 1, the information about the newly created ORCL database was created in `/u01/app/oracle/product/19.3.0/dbhome_1/`.

Now we will create the same ORCL database on Node-B so the information about the ORCL database will be stored on Node-B in `/u01/app/oracle/product/19.3.0/dbhome_1/`. The database files themselves are located in /datakeeper/oradata/ORCL.

On Node-B, these files can be seen as shown below:

```
[ec2-user@node-b ~]$ su - oracle
Password:
Last login: Sun Dec 20 03:58:10 UTC 2020 on pts/0
[oracle@node-b ~]$ ls -al /datakeeper/oradata/ORCL/
total 2477920
drwxr-x--- 2 oracle oinstall       200 Dec 20 04:15 .
drwxrwxr-x 3 oracle oinstall        18 Dec 20 04:11 ..
-rw-r----- 1 oracle oinstall  10600448 Dec 20 05:15 control01.ctl
-rw-r----- 1 oracle oinstall  10600448 Dec 20 05:15 control02.ctl
-rw-r----- 1 oracle oinstall 209715712 Dec 20 05:15 redo01.log
-rw-r----- 1 oracle oinstall 209715712 Dec 20 05:15 redo02.log
-rw-r----- 1 oracle oinstall 209715712 Dec 20 05:15 redo03.log
-rw-r----- 1 oracle oinstall 555753472 Dec 20 05:15 sysaux01.dbf
-rw-r----- 1 oracle oinstall 943726592 Dec 20 05:15 system01.dbf
-rw-r----- 1 oracle oinstall  33562624 Dec 20 04:31 temp01.dbf
-rw-r----- 1 oracle oinstall 356524032 Dec 20 05:15 undotbs01.dbf
-rw-r----- 1 oracle oinstall   5251072 Dec 20 05:15 users01.dbf
```

Rename the ORCL folder so that creation of the ORCL database on Node-B can be completed successfully.

```
[oracle@node-b ~]$ cd /datakeeper/oradata/
[oracle@node-b oradata]$ mv ORCL/ ORCL_Backup/
[oracle@node-b oradata]$ ls -al
total 0
drwxrwxr-x 3 oracle oinstall  25 Dec 20 05:26 .
drwxr-xr-x 3 root   root      21 Dec 20 02:24 ..
drwxr-x--- 2 oracle oinstall 200 Dec 20 04:15 ORCL_Backup
```

# 11.2.7.5.1.5. Update Config File for Oracle Listener on Both Nodes

The following steps should be executed on both nodes.

## Check the Listener Status

Check the current status of the Listener with following command:

```
$ cd $ORACLE_HOME/bin/
$ ./lsnrctl status LISTENER1
```

## Stop Listener if it is Running

```
$ cd $ORACLE_HOME/bin/
$ ./lsnrctl stop LISTENER1
```

## Edit the Configuration File

The configuration file is located at `/u01/app/oracle/product/19.3.0/dbhome_1/network/admin/listener.ora`.

The content of this file should be similar to the example shown below:

```
# listener.ora Network Configuration File: /u01/app/oracle/product/19.3.0/dbho
me_1/network/admin/listener.ora
# Generated by Oracle configuration tools.
LISTENER1 =
  (DESCRIPTION_LIST =
      (DESCRIPTION =
           (ADDRESS = (PROTOCOL = TCP)(HOST = node-a)(PORT = 1521))
           (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
       )
   )
```

See Creating an IP Resource to confirm which IP address was previously set.

Also, an additional section for the `SID_LIST_LISTENER1` resource is needed. The updated file should look similar to the example below:

```
# listener.ora Network Configuration File: /u01/app/oracle/product/19.3.0/dbho
me_1/network/admin/listener.ora
# Generated by Oracle configuration tools.
SID_LIST_LISTENER1 =
     (SID_LIST =
     (SID_DESC =
             (SID_NAME = orcl)
       )
 )
LISTENER1 =
  (DESCRIPTION_LIST =
      (DESCRIPTION =
           (ADDRESS = (PROTOCOL = TCP)(HOST = 10.10.10.10)(PORT = 1521))
           (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
       )
   )
```

# 11.2.7.5.1.6. Start Database and Listener on node-a

The following steps should be completed on node-a.

As discussed in previous steps, start the database with `sqlplus` and the listener with `lsnrctl`.

Start both processes on node-a so that LifeKeeper can see these processes as the corresponding resources are created.

## Start the Database

```
[oracle@node-a ~]$ $ORACLE_HOME/bin/sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Sun Dec 20 05:58:05 2020
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area  771748536 bytes
Fixed Size         8901304 bytes
Variable Size       490733568 bytes
Database Buffers    268435456 bytes
Redo Buffers          3678208 bytes
Database mounted.
Database opened.
SQL>EXIT
```

## Start the Listener

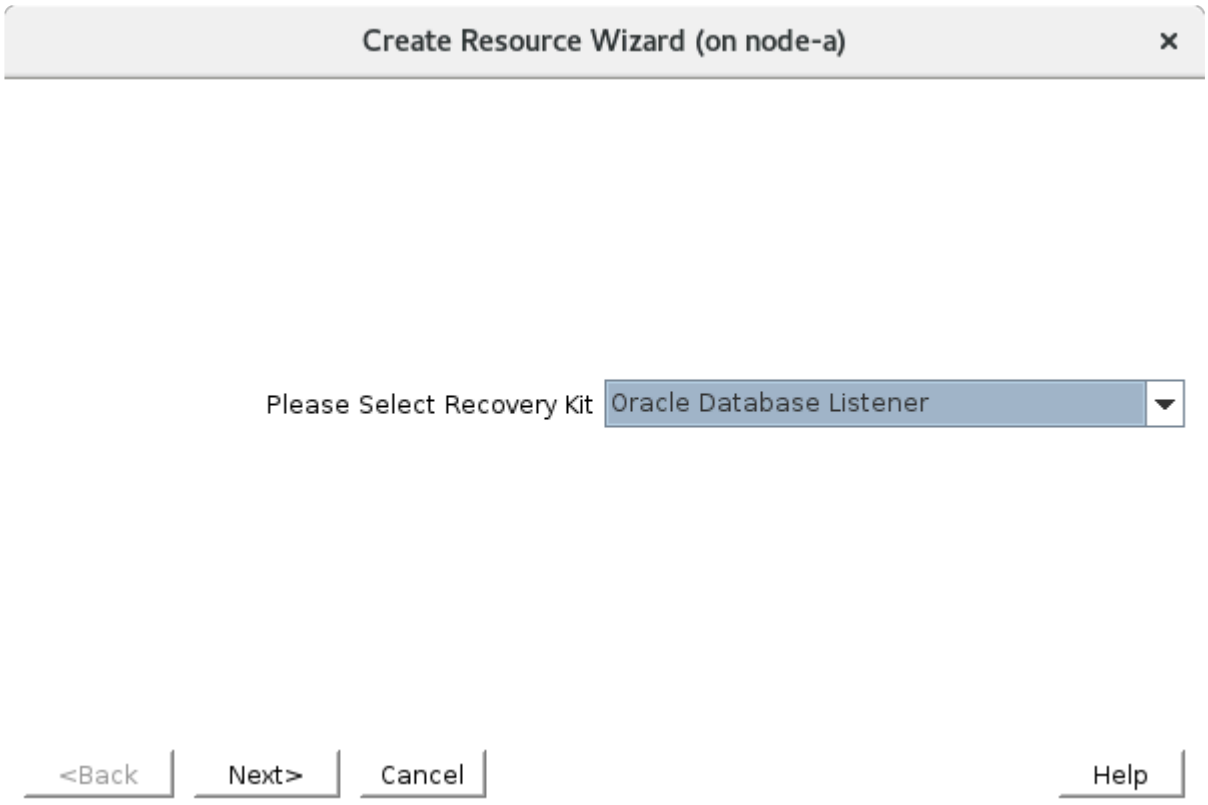
```
$ cd $ORACLE_HOME/bin/
$ ./lsnrctl start LISTENER1
```

# 11.2.7.5.1.7. Configure Oracle LISTENER Resource

As discussed in Update Config File for Oracle LISTENER on Both Nodes, the name of the Listener to protect is `LISTENER1`.

> **Note**: If using a version **previous to 9.5.2**, do not specify the ip address for the listener.

1. In the LifeKeeper User Interface, define a new resource. Select ⊕ to start the Create Resource Wizard (on node-a). Select “Oracle Database Listener” as the Recovery Kit.

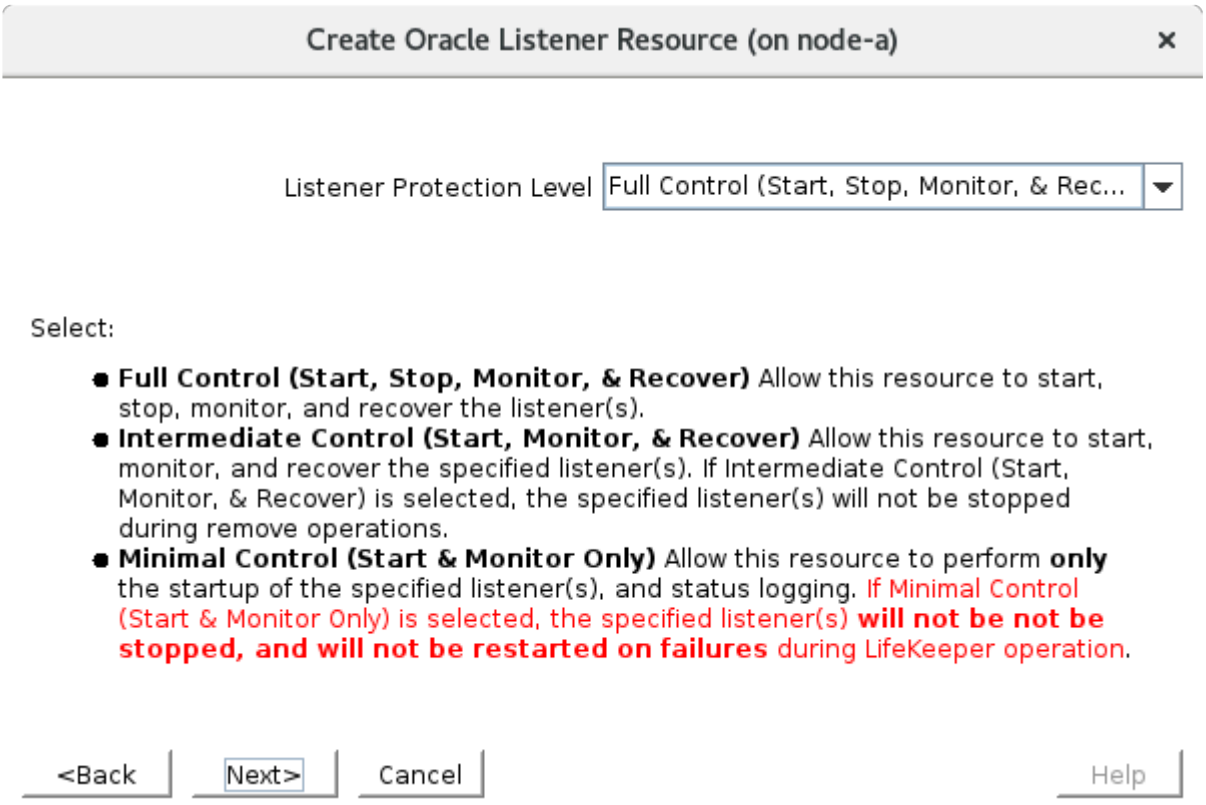



2. Enter the following values. Select the default values.

| Field | Value |
|---|---|
| Switchback Type | intelligent ✓ |
| Server | node-a ✓ |
| Listener Configuration File Path | /u01/app/oracle/product/19.3.0/dbhome_1/network/admin/listener.ora ✓ |
| Listener Name(s) | LISTENER1 ✓ |

| | |
|---|---|
| Listener Executable(s) | /u01/app/oracle/product/19.3.0/dbhome_1/bin/lsnrctl |

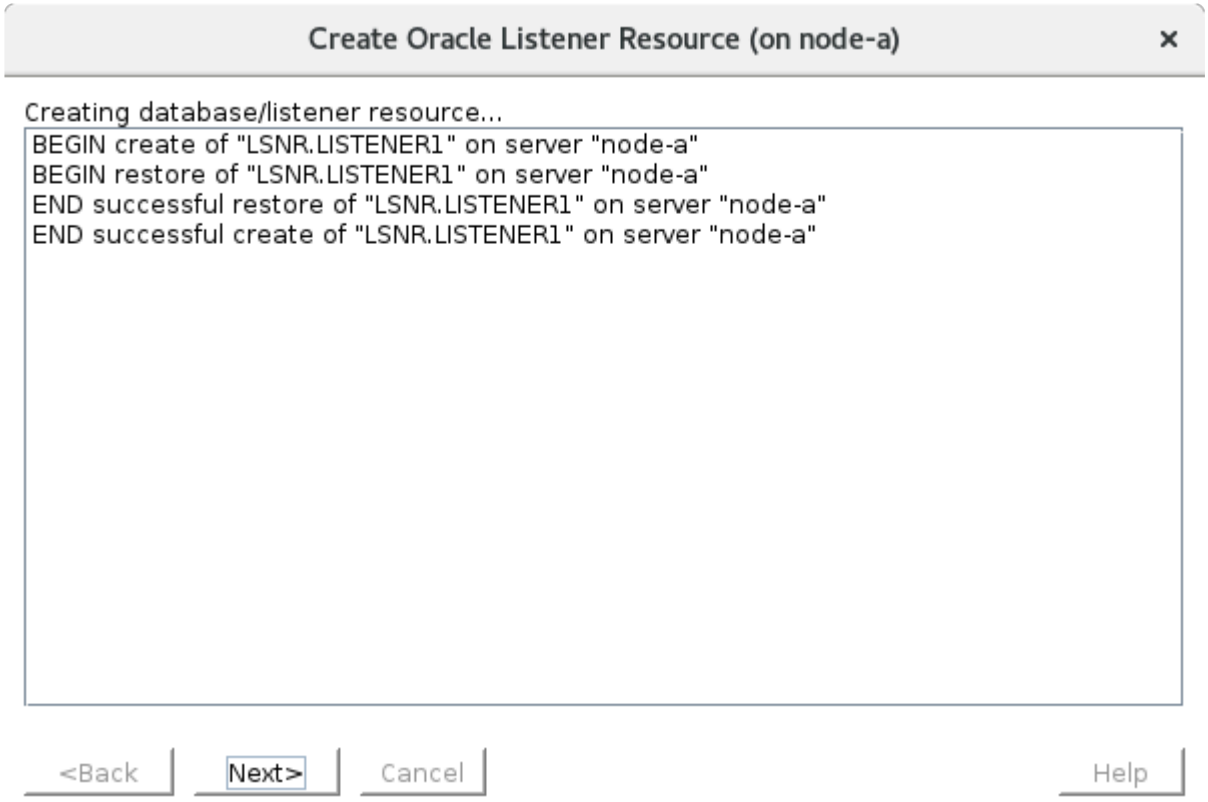
3. Select Listener Protection Level. Select “Full Control” as the wizard recommends.



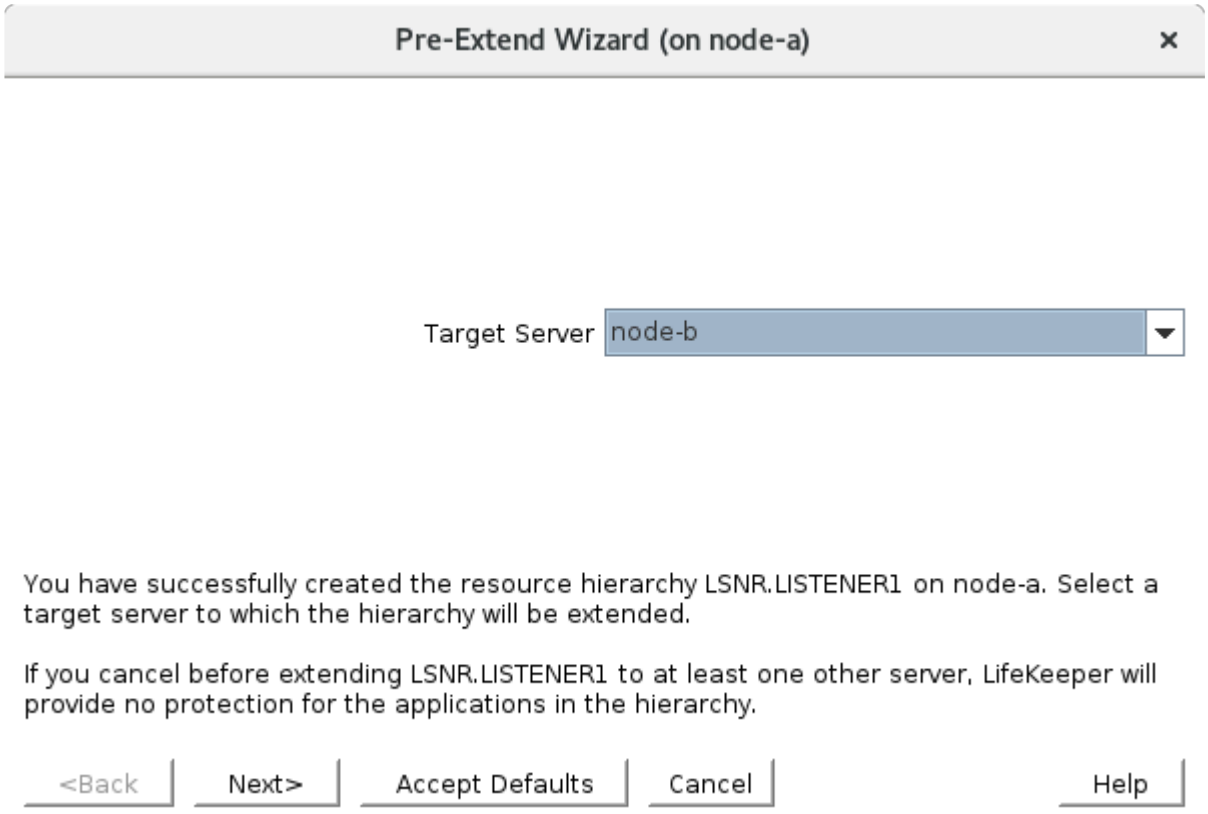

4. Enter the following values. Select the default values.

| Field | Value |
|---|---|
| Listener Recovery Level | Standard (On) |
| IP address Name(s) | ip-10.10.10.10 |
| Listener Tag | LSNR.LISTENER1 |

5. The wizard reviews all values specified. Select “Next >” to continue.



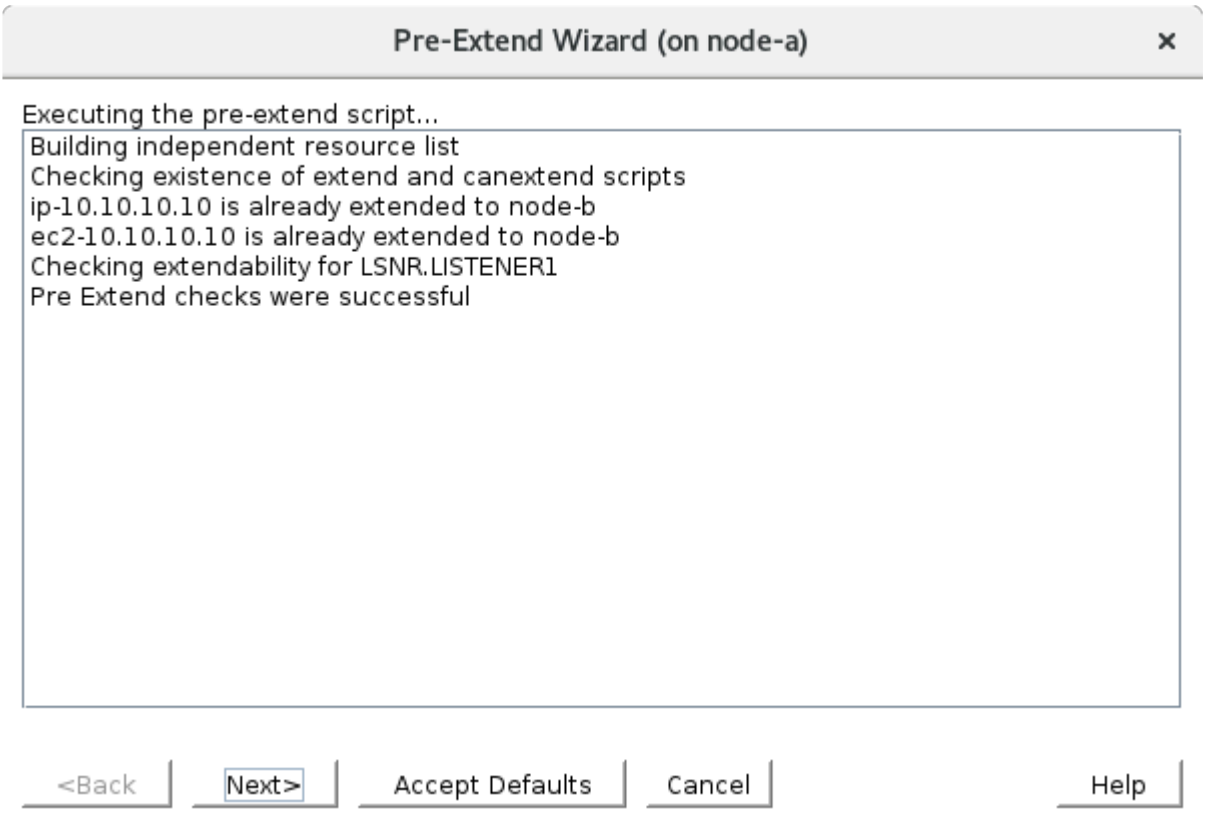

6. The next step is the “Pre-Extend Wizard”. Select the default values.



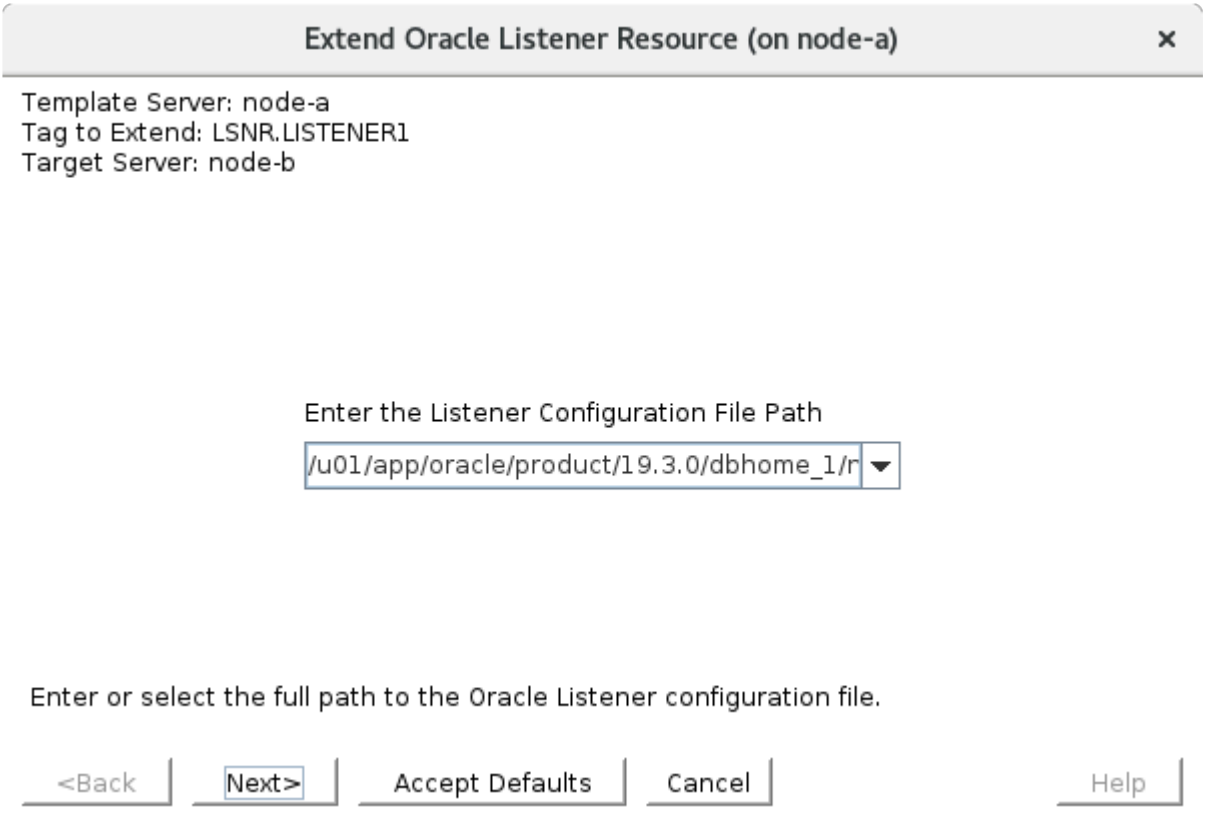

| Field | Value |
|---|---|
| Switchback Type | intelligent |

| | |
|---|---|
| Template Priority | 1 |
| Target Priority | 10 |

Once the ‘Pre-Extend Wizard’ checks are passed, select ‘Next’ to continue.



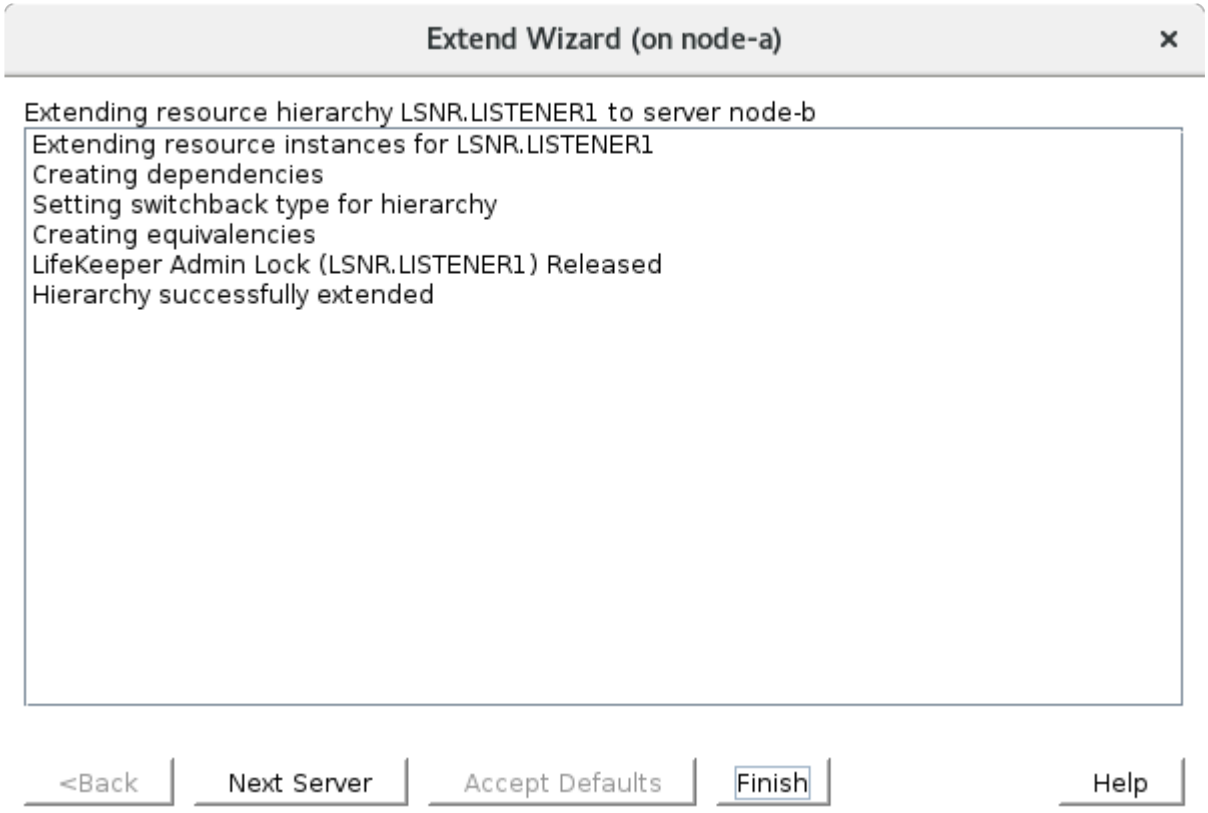

7. On the “Extend Oracle Listener Resource” wizard, the first choice is selecting the Listener Configuration File Path. This is the same as step #2, use the path detected by the wizard.



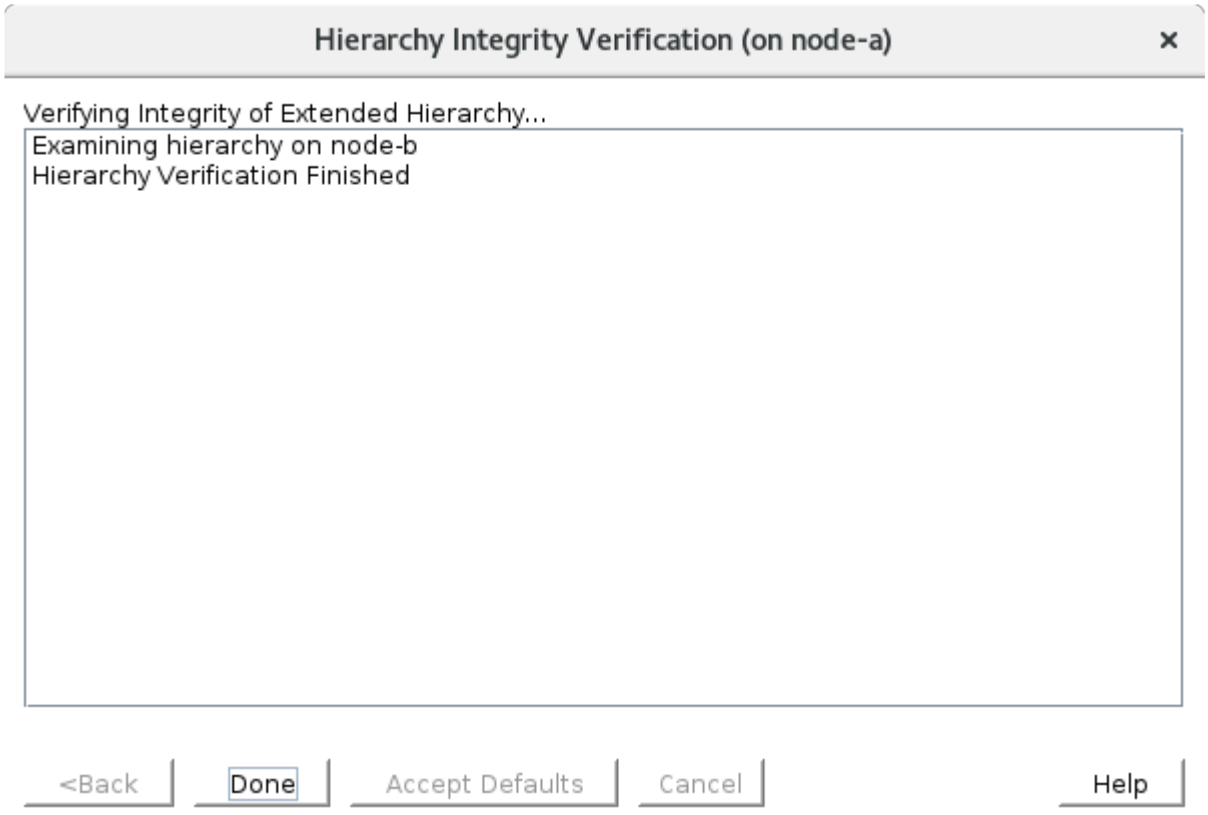

8. Enter the following values. Select the default values.

| Field | Value |
|---|---|
| Listener Executable(s) | /u01/app/oracle/product/19.3.0/dbhome_1/bin/lsnrctl |
| Listener Tag | LSNR.LISTENER1 |

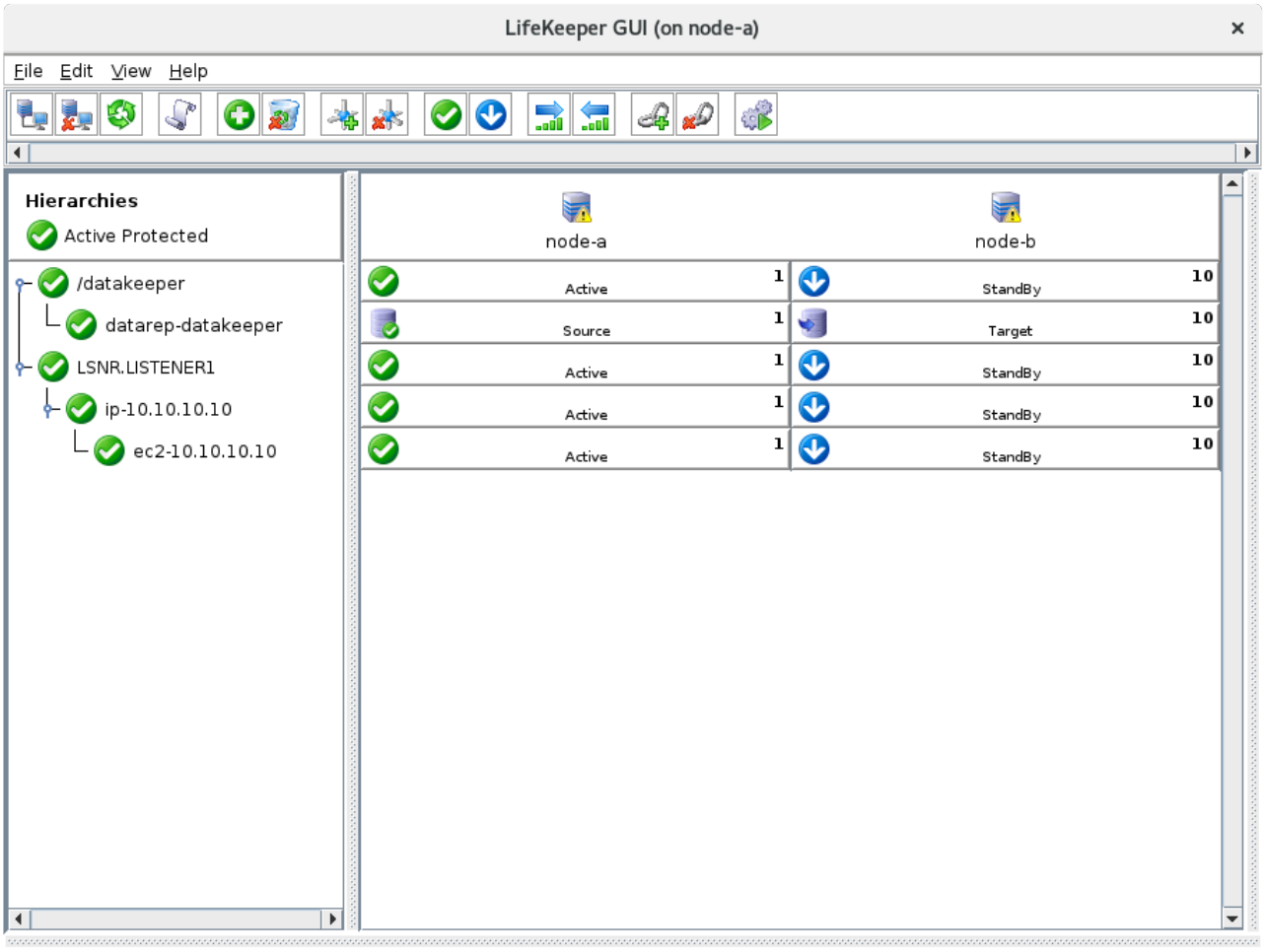
9. The Extend Wizard reviews the values and extends the Listener Resource to Node-B. Once the configuration is completed, select “Finish”.




10. The Hierarchy is now verified. Select “Done” once the verification completes.



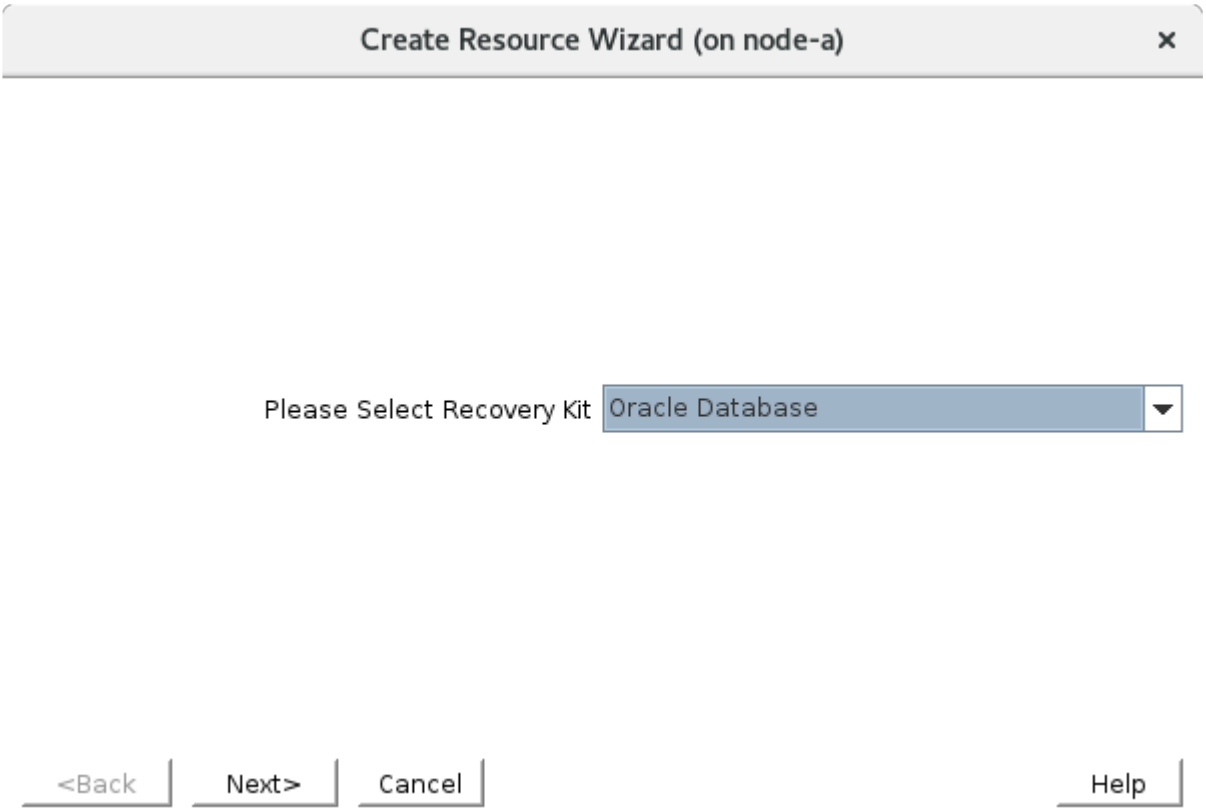

11. Now the Oracle Listener Resource is defined. The Oracle Listener Resource depends on the ip-10.10.10.10 resource as shown below.


LifeKeeper GUI (on node-a)
File Edit View Help
Hierarchies
Active Protected
/datakeeper
datarep-datakeeper
LSNR.LISTENER1
ip-10.10.10.10
ec2-10.10.10.10
node-a
node-b
Active 1
StandBy 10
Source 1
Target 10
Active 1
StandBy 10
Active 1
StandBy 10
Active 1
StandBy 10

# 11.2.7.5.1.8. Configure the Oracle Resource

As discussed in Create an Oracle Database (non-PDB), the name of Oracle Instance to protect is `ORCL`.

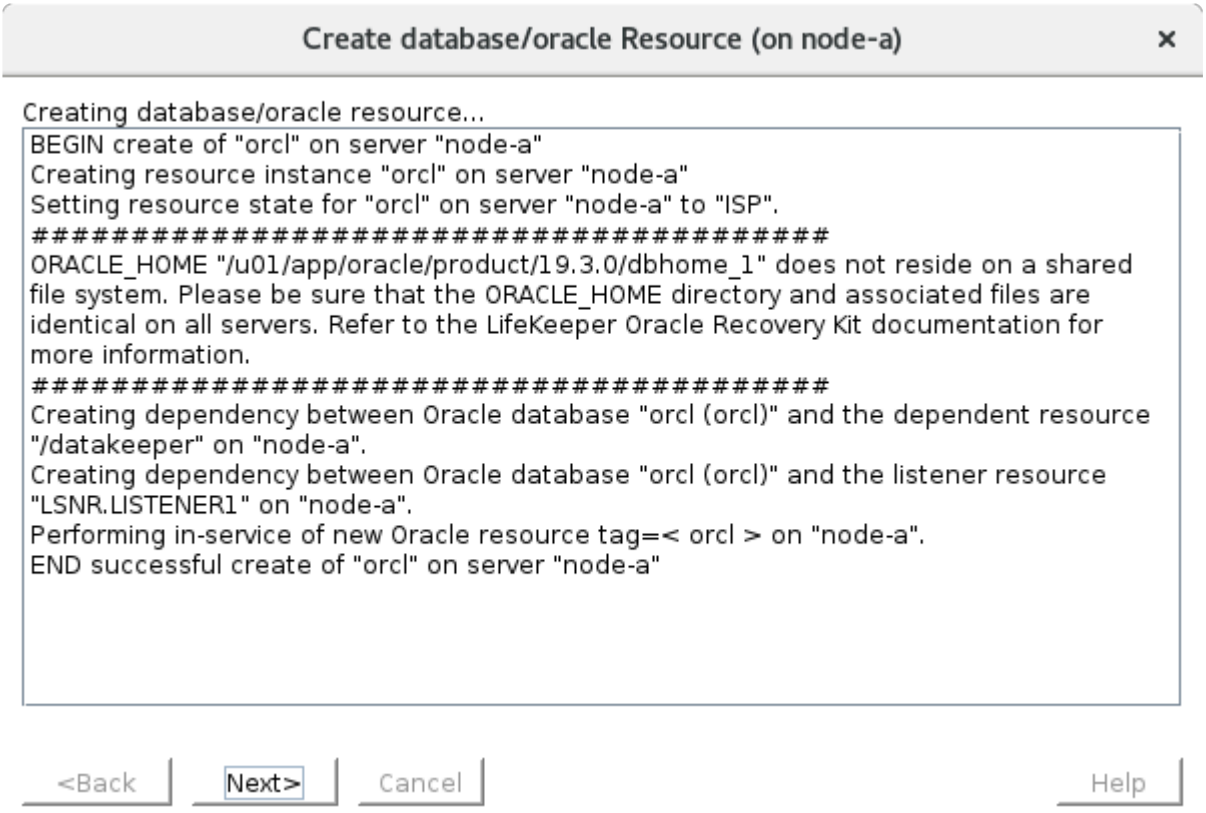
1. In the LifeKeeper User Interface, define a new resource. Select to start the Create Resource Wizard (on node-a). Select “Oracle Database” as the Recovery Kit.



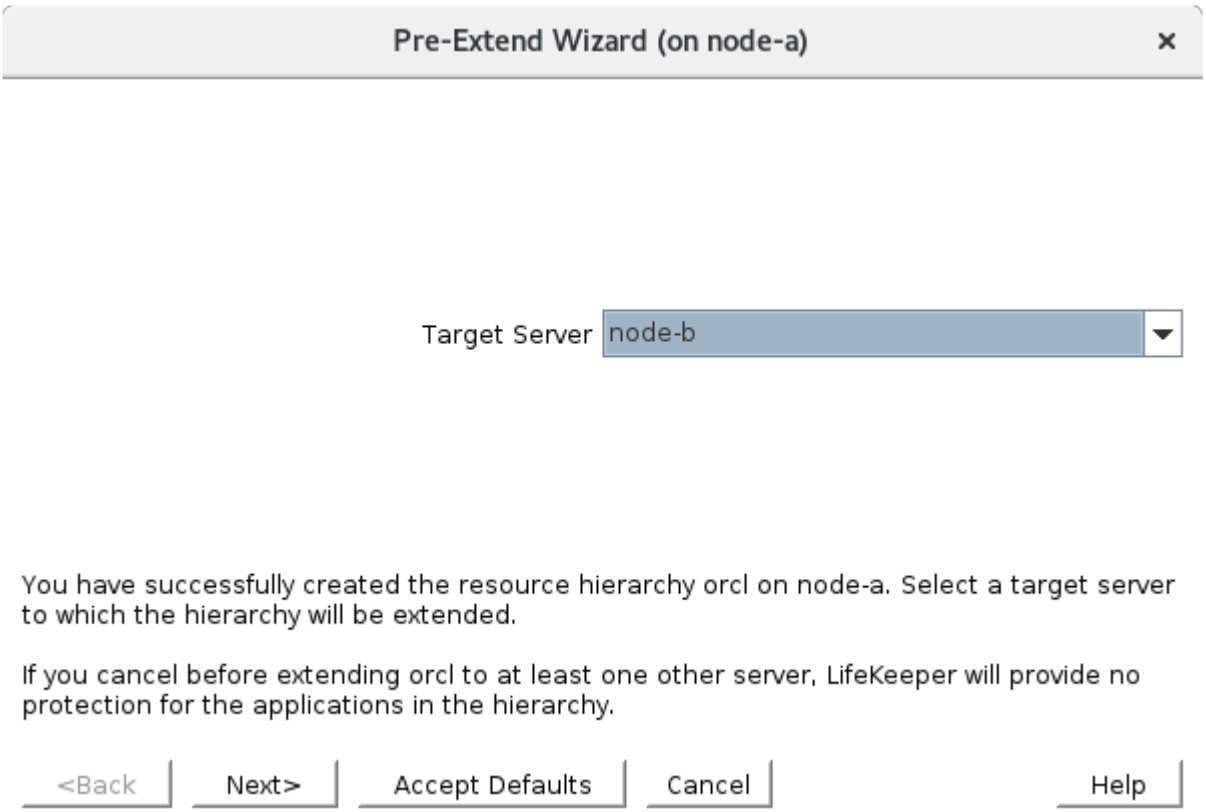

2. Enter the following values. Select the default values.

| Field | Value |
|---|---|
| Switchback Type | intelligent |
| Server | node-a |
| Oracle SID | ORCL |
| Username | <Leave blank> |
| Oracle Listener | LSNR.LISTENER1 |
| Database Tag | ORCL |

3. The wizard reviews the values. Once this is complete, click “Next >” to continue.



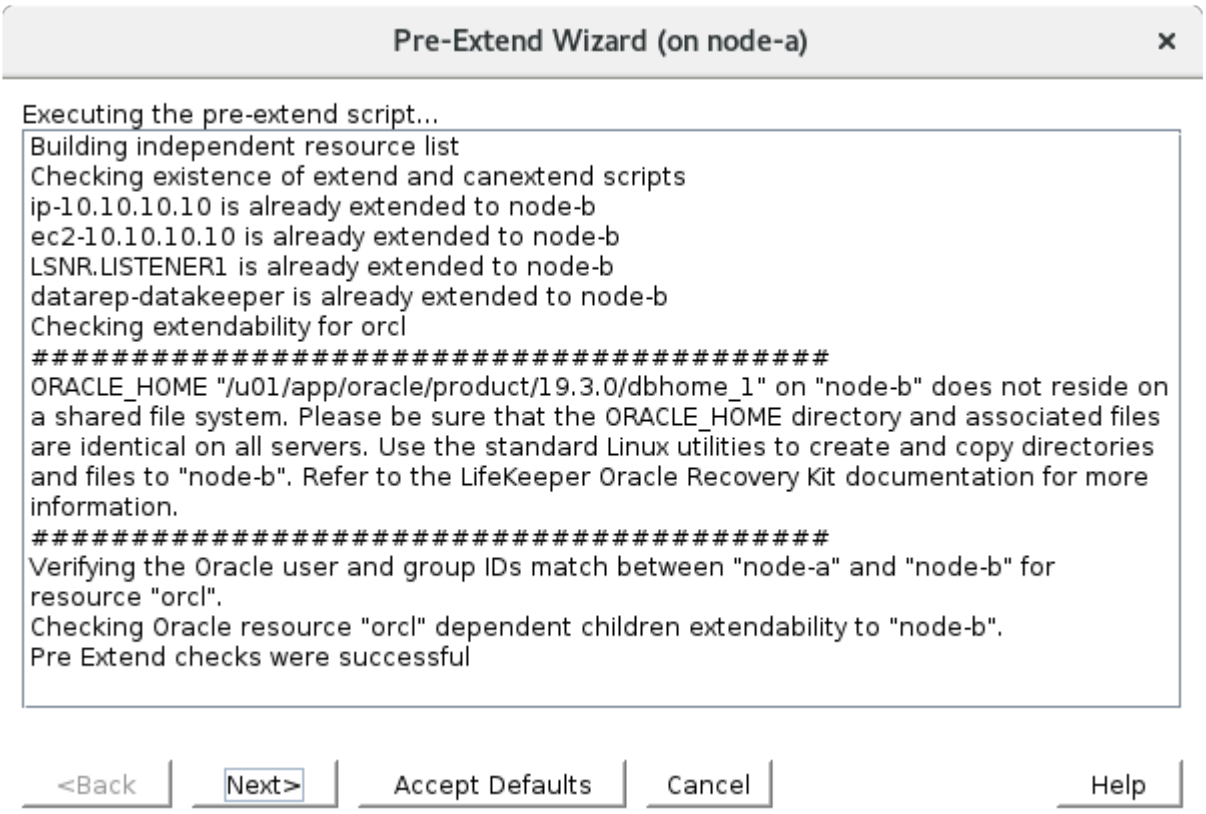

4. The next step is the “Pre-Extend Wizard”. Select the default values.



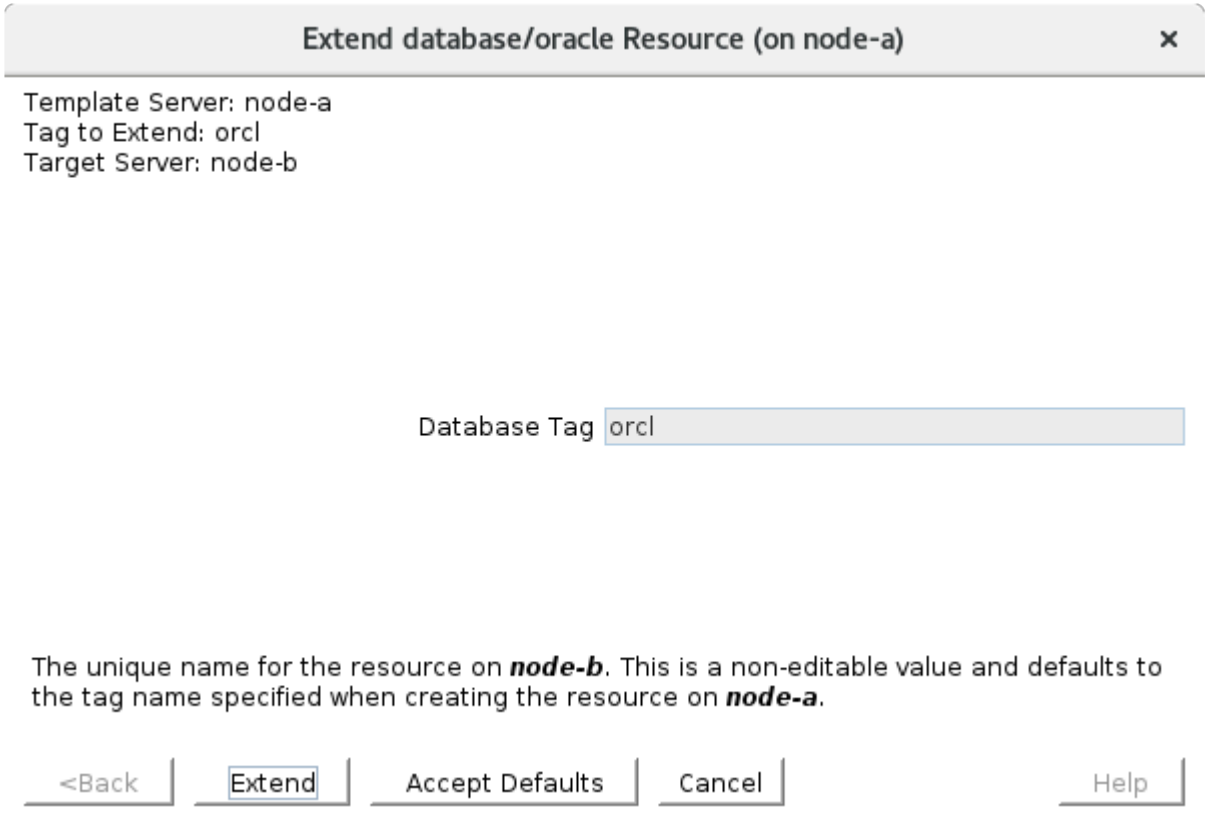

| Field | Value |
|---|---|
| Switchback Type | intelligent |

| | |
|---|---|
| Template Priority | 1 |
| Target Priority | 10 |

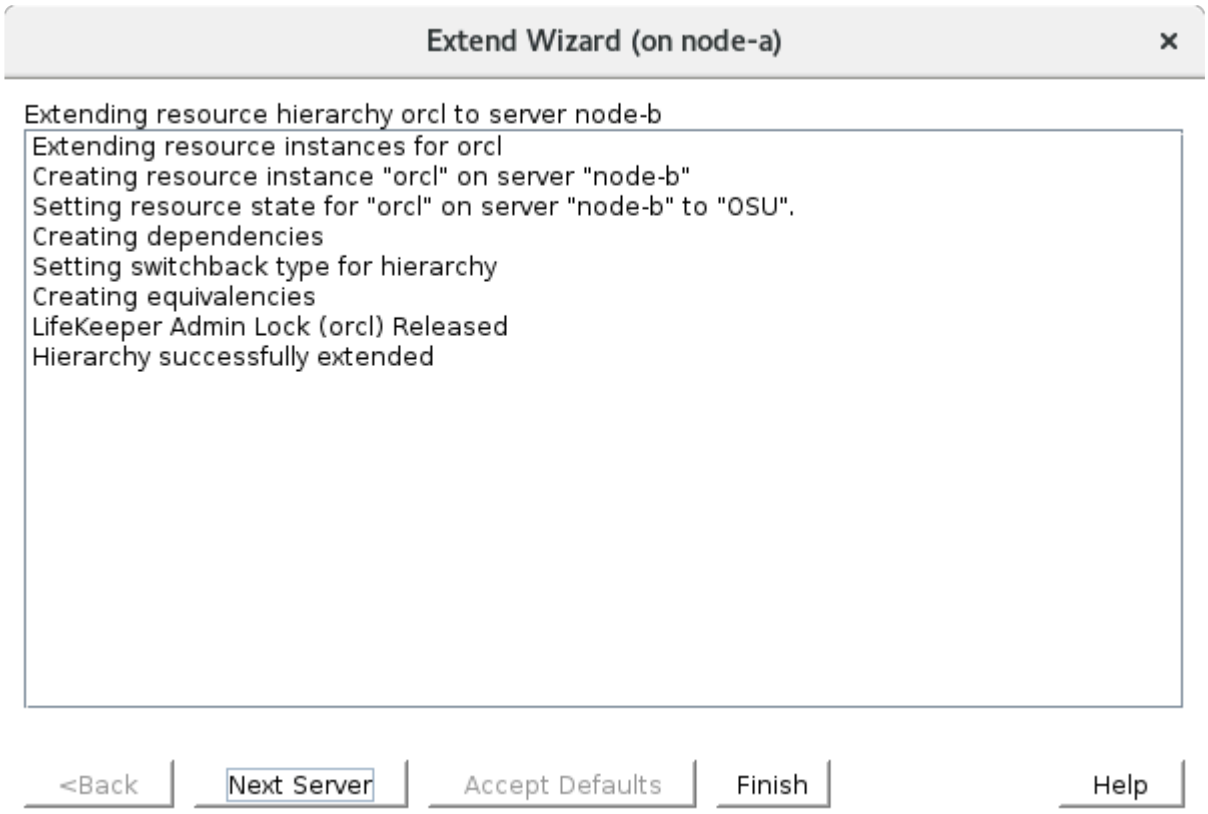
Once the ‘Pre-Extend Wizard’ checks pass, click ‘Next’ to continue.



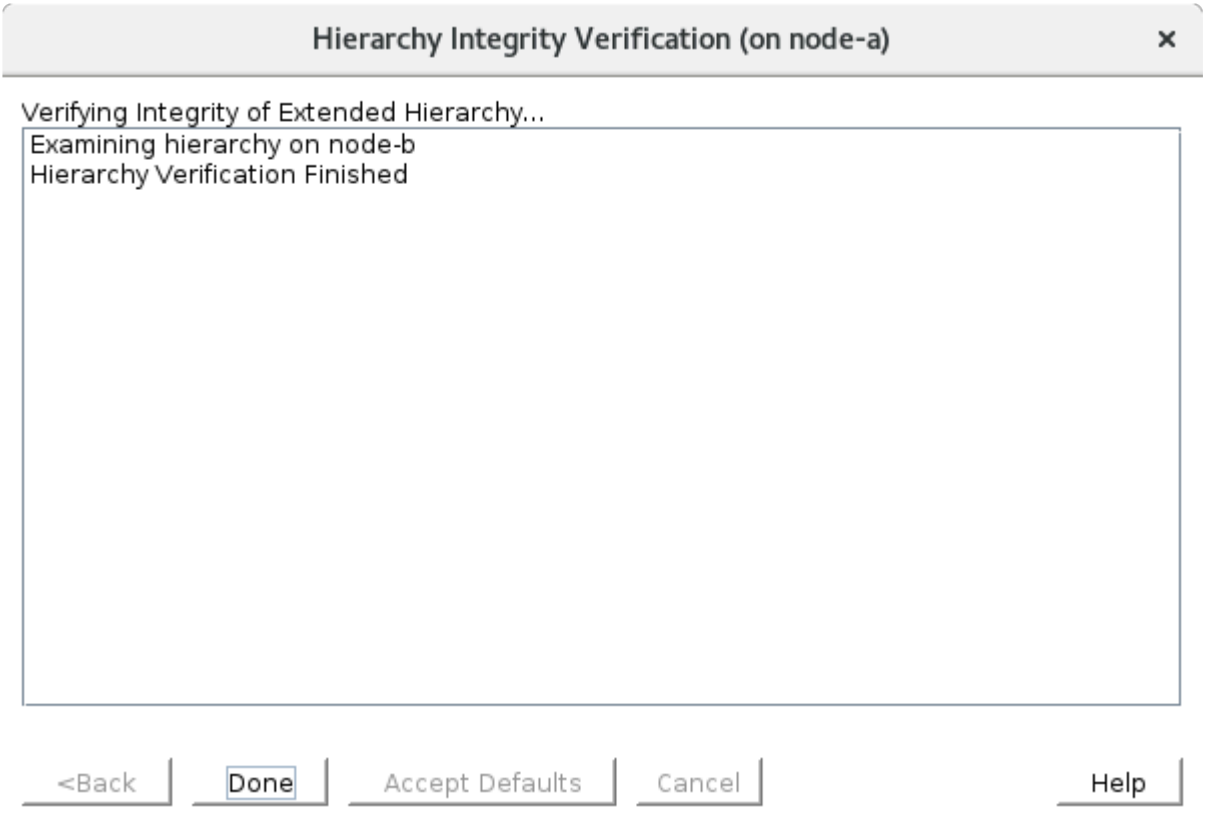

5. On the “Extend database/oracle Resource” wizard, the first choice is selecting the Database Tag. Confirm the default value and click “Extend”.



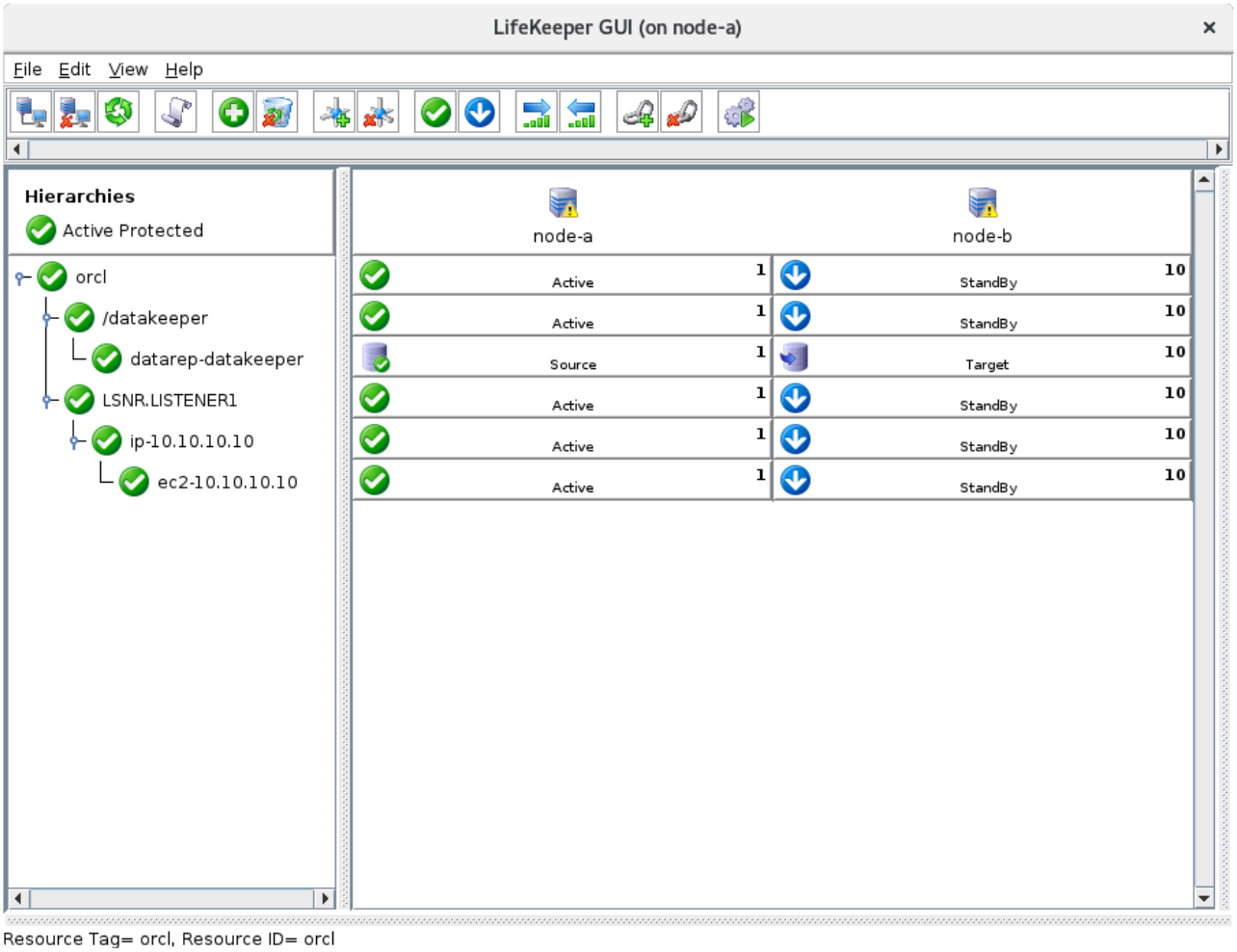

6. Once the Extend Wizard configures the Oracle resource, click “Finish”.




7. The Hierarchy is now checked and complete, click “Done”.




8. Return to the LifeKeeper GUI. The “orcl” resource depends on the /datakeeper resource and LSNR.LISTENER as shown below.



# 11.2.7.5.1.9. Test Switchover of the Oracle Resource

## Understanding the Status from Command Line Tools

Before testing the switchover, review the status of each node.

### Understand the Status of the File System

The easiest way to check the status is using `df`.

```
[oracle@node-b dbhome_1]$ df
Filesystem     1K-blocks     Used Available Use% Mounted on
devtmpfs          916888        0    916888   0% /dev
tmpfs            7340032   450560   6889472   7% /dev/shm
tmpfs             940184    17480    922704   2% /run
tmpfs             940184        0    940184   0% /sys/fs/cgroup
/dev/xvda2      31444972 17247248  14197724  55% /
tmpfs             188040        0    188040   0% /run/user/1000
/dev/md0        20960236  4997824  15962412  24% /datakeeper
tmpfs             188040        0    188040   0% /run/user/54321
```

The /datakeeper resource is attached to the node and is mounted as shown here.

### Understand the Status of the Listener and Oracle Processes

We can use the `ps` command to review the status of each node. If the processes are running they will be listed as shown below.

```
$ ps afx | grep -v grep | grep tnslsnr
13806 ?        Ssl    0:00 /u01/app/oracle/product/19.3.0/dbhome_1/bin/tnslsn
r LISTENER1 -inherit
$ ps afx | grep -v grep | grep ora_pmon
23091 ?        Ss     0:00 ora_pmon_orcl
```

### Understand the Status of the IP Resource

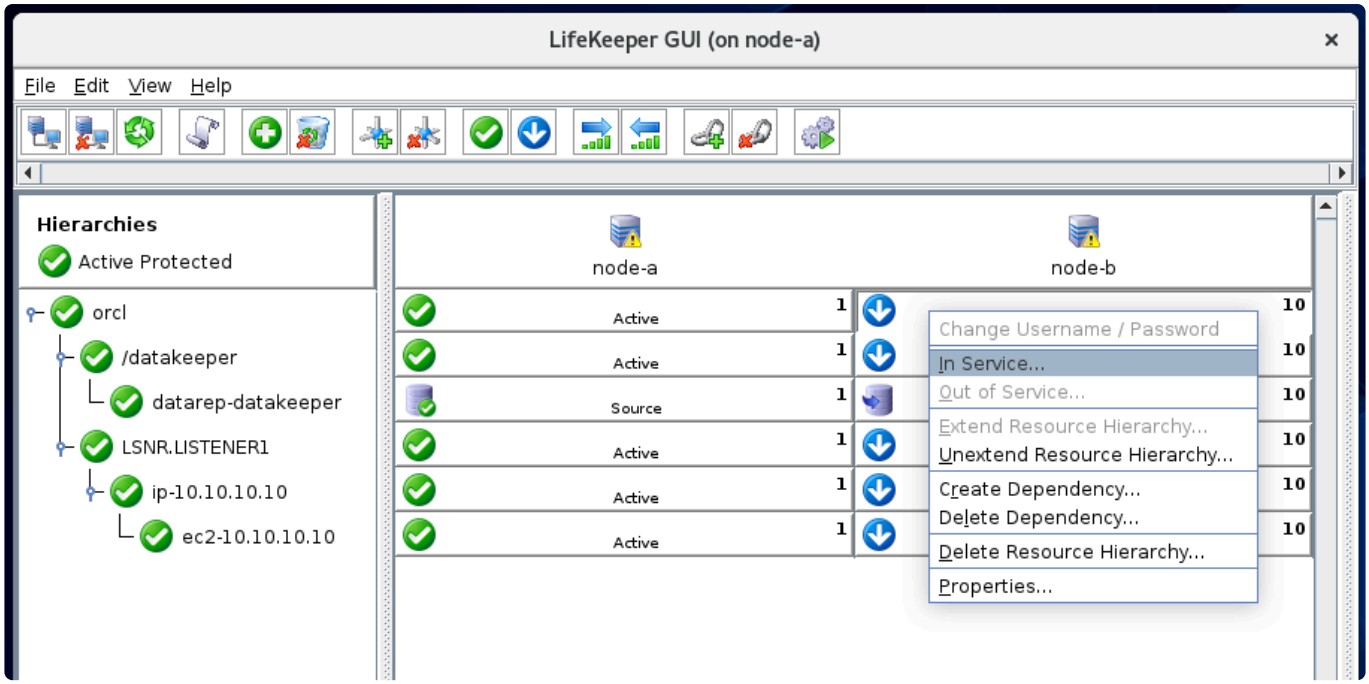
As discussed in How a Client Connects to the Active Node, if the IP resource is active, the virtual IP address can be found on the `ip` command as follows (see 10.10.10.10).

```
[oracle@node-b dbhome_1]$ ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP
group default qlen 1000
```

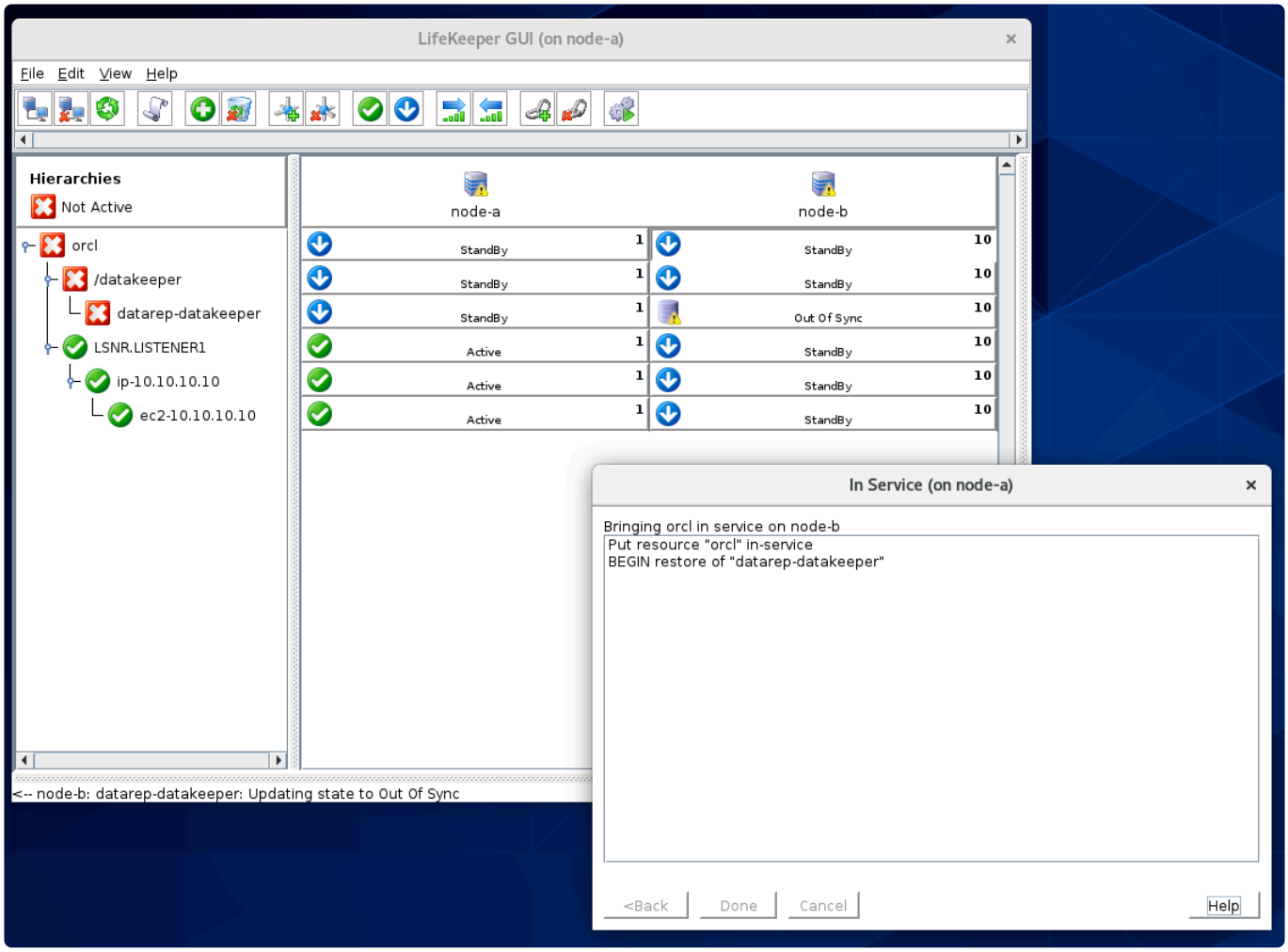
```
    link/ether 02:a8:0c:57:53:0b brd ff:ff:ff:ff:ff:ff
    inet 10.20.10.12/24 brd 10.20.10.255 scope global noprefixroute dynamic et
h0
       valid_lft 3585sec preferred_lft 3585sec
    inet 10.10.10.10/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::a8:cff:fe57:530b/64 scope link
       valid_lft forever preferred_lft forever
```

## Switching the Resource Between Nodes

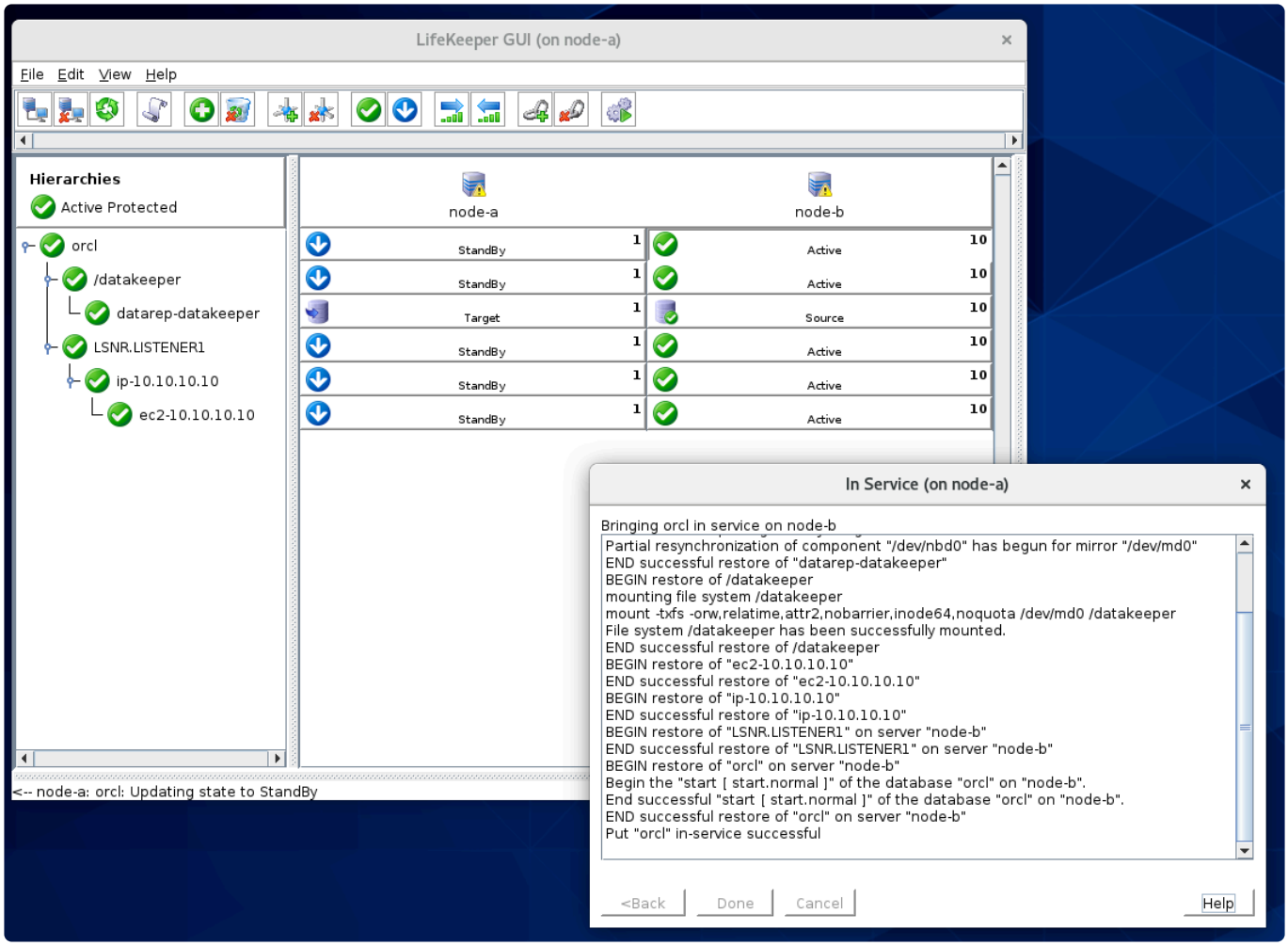
Show the context menu on the “orcl” resource on node-b (standby node) and select “In Service”.




First, the resource and all child dependencies will be transitioned to “StandBy” on node-a.




Once all resources are stopped on node-a, the dependencies and the “orcl” resource will be started on node-b.


LifeKeeper GUI (on node-a)
File Edit View Help
Hierarchies
Active Protected
orcl
/datakeeper
datarep-datakeeper
LSNR.LISTENER1
ip-10.10.10.10
ec2-10.10.10.10
node-a
node-b
StandBy 1
Active 10
StandBy 1
Active 10
Target 1
Source 10
StandBy 1
Active 10
StandBy 1
Active 10
StandBy 1
Active 10
<-- node-a: orcl: Updating state to StandBy
In Service (on node-a)
Bringing orcl in service on node-b
Partial resynchronization of component "/dev/nbd0" has begun for mirror "/dev/md0"
END successful restore of "datarep-datakeeper"
BEGIN restore of /datakeeper
mounting file system /datakeeper
mount -txfs -orw,relatime,attr2,nobarrier,inode64,noquota /dev/md0 /datakeeper
File system /datakeeper has been successfully mounted.
END successful restore of /datakeeper
BEGIN restore of "ec2-10.10.10.10"
END successful restore of "ec2-10.10.10.10"
BEGIN restore of "ip-10.10.10.10"
END successful restore of "ip-10.10.10.10"
BEGIN restore of "LSNR.LISTENER1" on server "node-b"
END successful restore of "LSNR.LISTENER1" on server "node-b"
BEGIN restore of "orcl" on server "node-b"
Begin the "start [ start.normal ]" of the database "orcl" on "node-b".
End successful "start [ start.normal ]" of the database "orcl" on "node-b".
END successful restore of "orcl" on server "node-b"
Put "orcl" in-service successful
<Back
Done
Cancel
Help

# 11.2.7.5.2. Protecting MSSQL Using Quick Service Protection

This section outlines the steps to protect Microsoft SQL Server 2017 in a Linux environment.

> ✻ This document uses Microsoft SQL Server 2017 as an example. You can apply the same steps on Microsoft SQL Server 2012 R2 through Microsoft SQL Server 2019.

Although LifeKeeper doesn't have a specific Application Recovery Kit for MSSQL, LifeKeeper can still protect MSSQL as a general service. This feature is called Quick Service Protection (QSP) and you can protect MSSQL as well as other services running in a Linux environment.

To protect database resources, the data needs to be replicated across nodes using DataKeeper (unless you use shared storage or a SAN device).

The following table outlines the location of each component.

| Items | Location |
|---|---|
| Master DataBase Files | /datakeeper/mssql/data |
| Master Log File | /datakeeper/mssql/xlog |

As discussed earlier in How to Create Data Replication of a File System, this guide uses `/datakeeper` to replicate data between nodes. Therefore, the data on `/datakeeper/mssql/data` is also replicated between nodes (this is the same for `/datakeeper/mssql/xlog` as well).

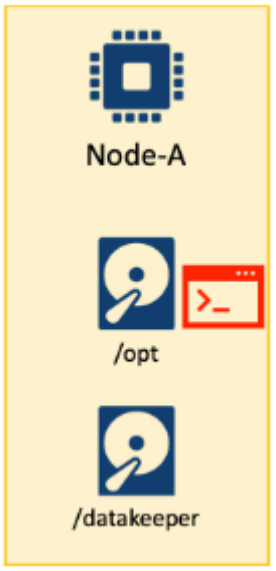
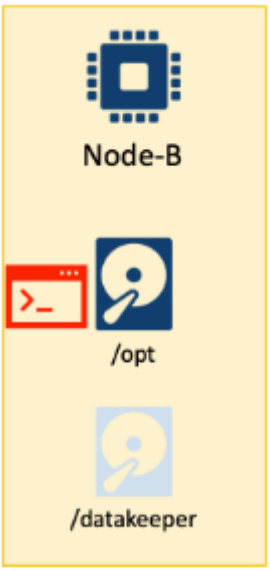
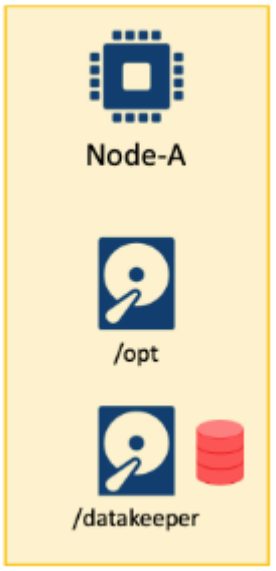
Please note that this guide uses the following computing resources (Microsoft SQL Server resource requires more memory than other resources even for evaluation purposes).

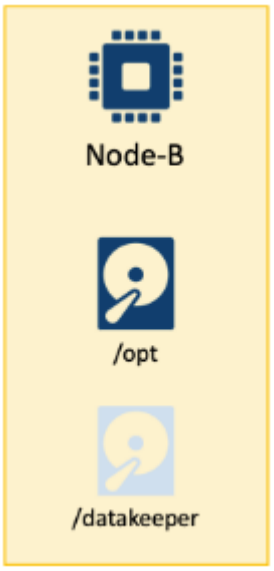
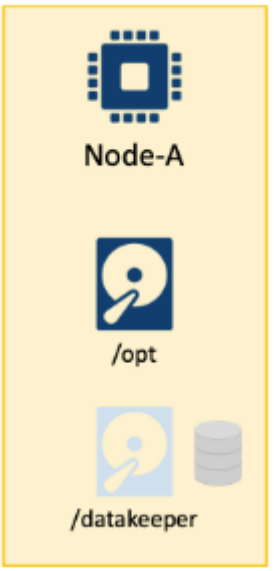
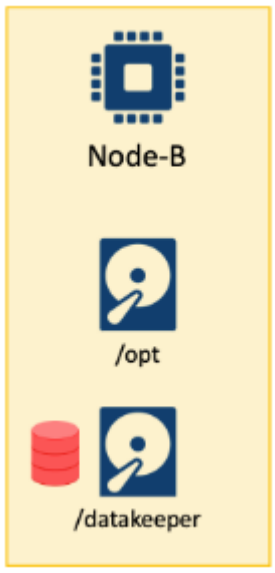
| Resource | Required Size |
|---|---|
| Memory | 4 GiB |

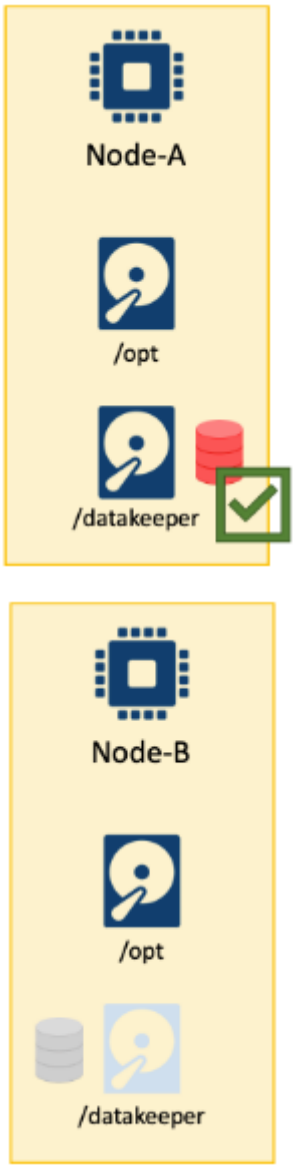
## Configure the MSSQL Resource

The following table outlines the general steps to configure a MSSQL Resource. The red 'stacked disk' shaped icons indicate a node that has the database instances at the time of each step. The grey 'stacked disk' icons indicate a database that is not running.

Also, the grey "storage" icon indicates that the replicated storage `/datakeeper` is not available for the node.

| | |
|---|---|
| **Install MSSQL 2017 on both nodes**<br><br>• Install MSSQL 2017 on a System |  <br>  |
| **Create Database on Node-A**<br><br>• Review How to Confirm if the Data Storage is Available on a Node. The data storage should be available on node-a.<br><br>• Relocate Master Database and Log Files to Replicated Storage on node-a |  |

| | |
|---|---|
| |  |
| **Create Database on Node-B**<br>• Switchover (the data storage) to node-b<br>• Rename Folders Under /datakeeper/MSSQL<br>• Relocate Master Database and Log Files to Replicated Storage on node-b |  <br>  |

| | |
|---|---|
| **Complete Configuration using LifeKeeper**<br><br>• Switchover (the data storage) back to node-a<br><br>• Configure Resources with LifeKeeper<br><br>• Customize LocalRecovery Parameter on Both Nodes<br><br>• Update Dependency between Resources |  |

# 11.2.7.5.2.1. Install MSSQL 2017

Install MSSQL on both nodes.

1. Refer to Microsoft's Installation guidance for SQL Server on Linux guide. It contains instructions for different versions of operating systems as well as MSSQL servers.

2. Download the Microsoft SQL Server 2017 Red Hat repository configuration file.

```
# curl -o /etc/yum.repos.d/mssql-server.repo
https://packages.microsoft.com/config/rhel/8/mssql-server-2017.repo
```

3. Run the following commands to install SQL Server:

```
# yum install -y mssql-server
```

4. Configure the MSSQL server.

```
# /opt/mssql/bin/mssql-conf setup
```

5. The server is now installed. Check the status of MSSQL server with the following command.

```
# systemctl status mssql-server
```

6. There are a few more steps before starting MSSQL Server. The MSSQL server is now installed successfully. Complete these steps on both nodes.

# 11.2.7.5.2.2. Relocate Master Database and Log Files to Replicated Storage

As discussed earlier, the data should be stored on the replicated file system. Therefore, we use the following locations for each component.

| Items | Location |
|---|---|
| Master DataBase Files | /datakeeper/mssql/data |
| Master Log File | /datakeeper/mssql/xlog |

> The following steps should be carried out on node-a **AND** node-b.
>
> There are some additional steps that need to be done in between. Please review the summary of the overall steps in Protecting MSSQL Using Quick Service Protection.

First, create directories on node-a.

```
# mkdir -p /datakeeper/mssql/data
# mkdir -p /datakeeper/mssql/xlog
# chown mssql -R /datakeeper/mssql/
# chgrp mssql -R /datakeeper/mssql/
```

Relocate the Master database to the newly created directories with the `/opt/mssql/bin/mssql-conf` tool as follows:

```
# /opt/mssql/bin/mssql-conf set filelocation.masterlogfile /datakeeper/mssql/xlog/mastlog.ldf
SQL Server needs to be restarted in order to apply this setting. Please run
'systemctl restart mssql-server.service'.

# /opt/mssql/bin/mssql-conf set filelocation.masterdatafile /datakeeper/mssql/data/master.mdf
SQL Server needs to be restarted in order to apply this setting. Please run
'systemctl restart mssql-server.service'.
```

Start the MSSQL server.

```
# systemctl restart mssql-server.service
# systemctl status mssql-server.service
● mssql-server.service - Microsoft SQL Server Database Engine
   Loaded: loaded (/usr/lib/systemd/system/mssql-server.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2020-12-30 18:55:54 UTC; 6s ago
     Docs: https://docs.microsoft.com/en-us/sql/linux
 Main PID: 15506 (sqlservr)
   CGroup: /system.slice/mssql-server.service
           ├─15506 /opt/mssql/bin/sqlservr
           └─15508 /opt/mssql/bin/sqlservr
```

Once the database is started, you will see some files created on the folders specified earlier.

```
# ls -al /datakeeper/mssql/data
total 51136
drwxr-xr-x 2 mssql mssql      137 Dec 30 18:56 .
drwxr-xr-x 6 mssql mssql       62 Dec 30 18:48 ..
-rw-rw---- 1 mssql mssql  4194304 Dec 30 18:56 master.mdf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:56 modellog.ldf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:56 model.mdf
-rw-rw---- 1 mssql mssql 14090240 Dec 30 18:56 msdbdata.mdf
-rw-rw---- 1 mssql mssql   524288 Dec 30 18:56 msdblog.ldf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:56 tempdb.mdf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:56 templog.ldf
# ls -al /datakeeper/mssql/xlog
total 2048
drwxr-xr-x 2 mssql mssql      25 Dec 30 18:55 .
drwxr-xr-x 6 mssql mssql      62 Dec 30 18:48 ..
-rw-rw---- 1 mssql mssql 2097152 Dec 30 18:56 mastlog.ldf
```

Now the instance can be stopped.

```
# systemctl stop mssql-server.service
```

# 11.2.7.5.2.3. Rename Folders Under /dataKeeper/MSSQL

So far the following tasks have been completed:

1. Created the database on node-a
2. Switched over the `datarep-datakeeper` resource to node-b

In step 1, the information about the newly created mssql database was created in `/var/opt/mssql/mssql.conf`.

Now we will create the same mssql database on node-b so the information about the mssql database will be stored on node-b in `/var/opt/mssql/mssql.conf`. The database files themselves are located in `/datakeeper/mssql/`.

On node-b, these files can be seen as shown below:

```
[root@node-b ~]# cd /datakeeper/mssql/
[root@node-b mssql]# ls -al data
total 51136
drwxr-xr-x 2 mssql mssql      137 Dec 30 18:38 .
drwxr-xr-x 6 mssql mssql       62 Dec 30 18:36 ..
-rw-rw---- 1 mssql mssql  4194304 Dec 30 18:38 master.mdf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:38 modellog.ldf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:38 model.mdf
-rw-rw---- 1 mssql mssql 14090240 Dec 30 18:38 msdbdata.mdf
-rw-rw---- 1 mssql mssql   524288 Dec 30 18:38 msdblog.ldf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:38 tempdb.mdf
-rw-rw---- 1 mssql mssql  8388608 Dec 30 18:38 templog.ldf
[root@node-b mssql]# ls -al xlog_bak/
total 2048
drwxr-xr-x 2 mssql mssql      25 Dec 30 18:38 .
drwxr-xr-x 6 mssql mssql      62 Dec 30 18:36 ..
-rw-rw---- 1 mssql mssql 2097152 Dec 30 18:38 mastlog.ldf
```

Rename these folders so that the database can be created successfully on node-b.

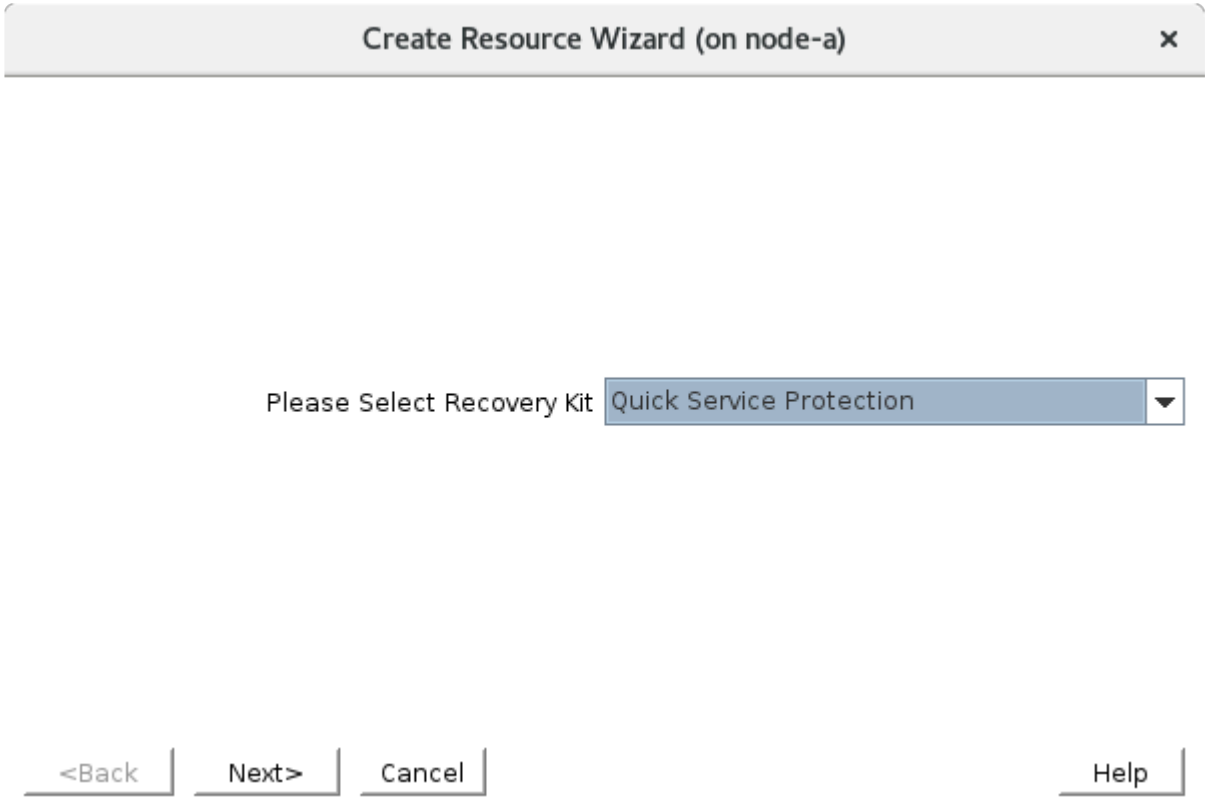
```
[root@node-b mssql]# mv data data_bak
[root@node-b mssql]# mv xlog xlog_bak
[root@node-b mssql]# ls -al
total 0
drwxr-xr-x 4 mssql mssql  38 Dec 30 18:48 .
drwxr-xr-x 5 root  root   57 Dec 30 18:36 ..
drwxr-xr-x 2 mssql mssql 137 Dec 30 18:38 data_bak
drwxr-xr-x 2 mssql mssql  25 Dec 30 18:38 xlog_bak
```

# 11.2.7.5.2.4. Configure MSSQL Resource

Now that the master database has been relocated to replicated storage, the MSSQL server resources are ready to be protected using LifeKeeper. Before configuring it with LifeKeeper, start MSSQL on node-a.

```
[root@node-a ~]# systemctl start mssql-server.service
[root@node-a ~]# systemctl status mssql-server.service
● mssql-server.service - Microsoft SQL Server Database Engine
   Loaded: loaded (/usr/lib/systemd/system/mssql-server.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2020-12-30 18:59:02 UTC; 1s ago
     Docs: https://docs.microsoft.com/en-us/sql/linux
 Main PID: 17146 (sqlservr)
   CGroup: /system.slice/mssql-server.service
           ├─17146 /opt/mssql/bin/sqlservr
           └─17148 /opt/mssql/bin/sqlservr
```

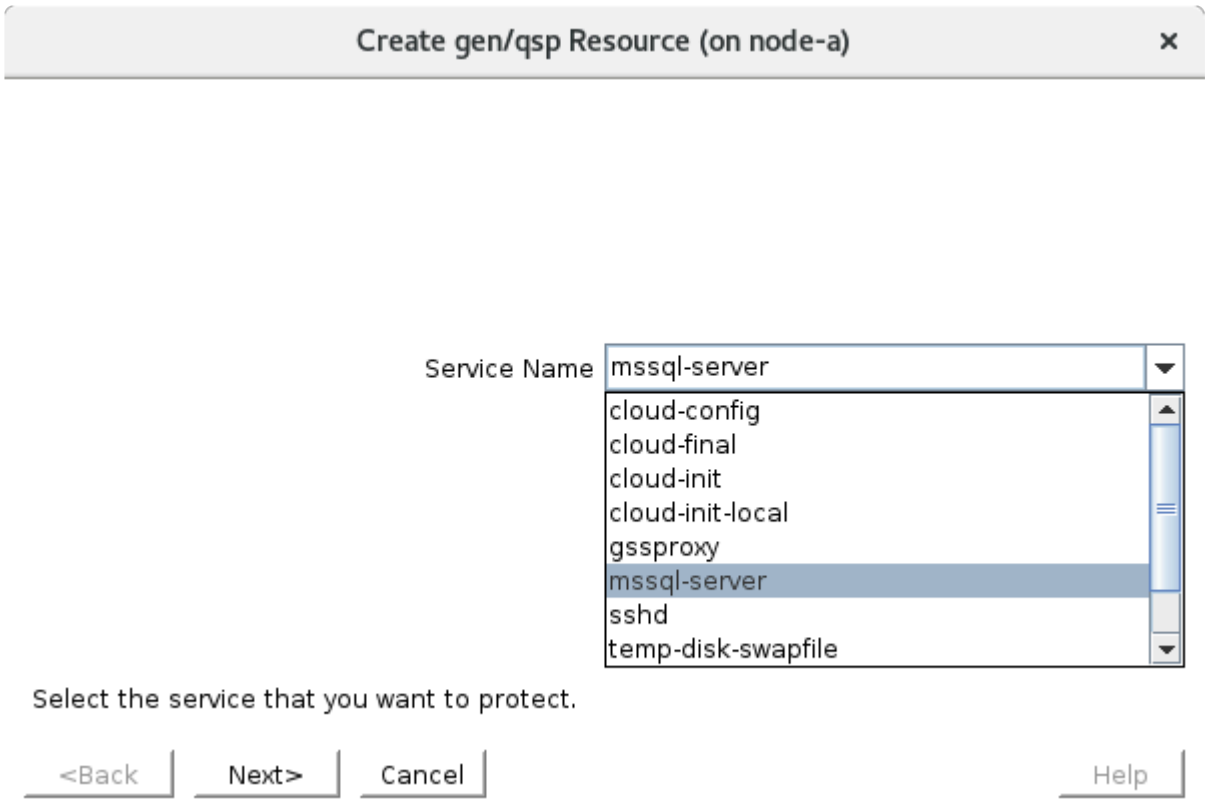
1. In the LifeKeeper User Interface, define a new resource. Click the icon to start the Create Resource Wizard (on node-a). Select “Quick Service Protection” as the Recovery Kit.




2. Enter the following values. Select the default values.

| Field | Value |
|---|---|
| Switchback Type | intelligent |
| Server | node-a |

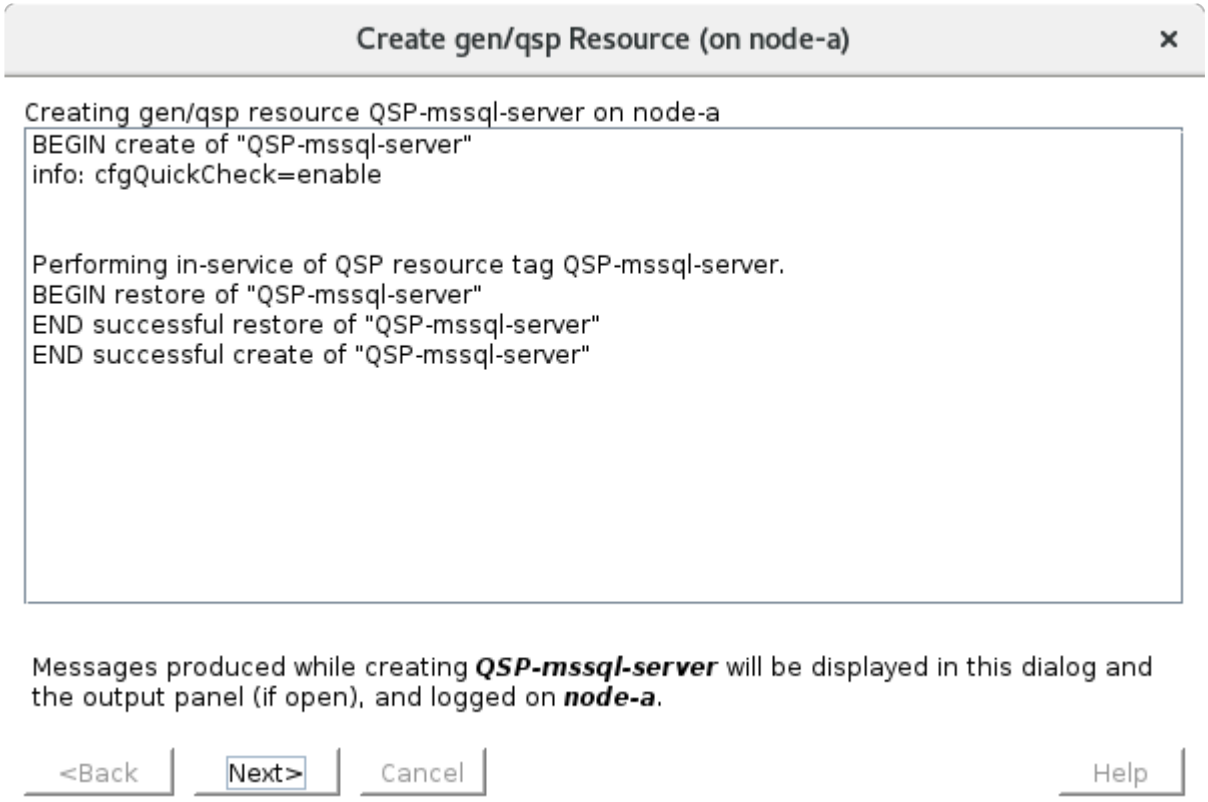
3. Select “mssql-server” for the Service Name from the dropdown list.



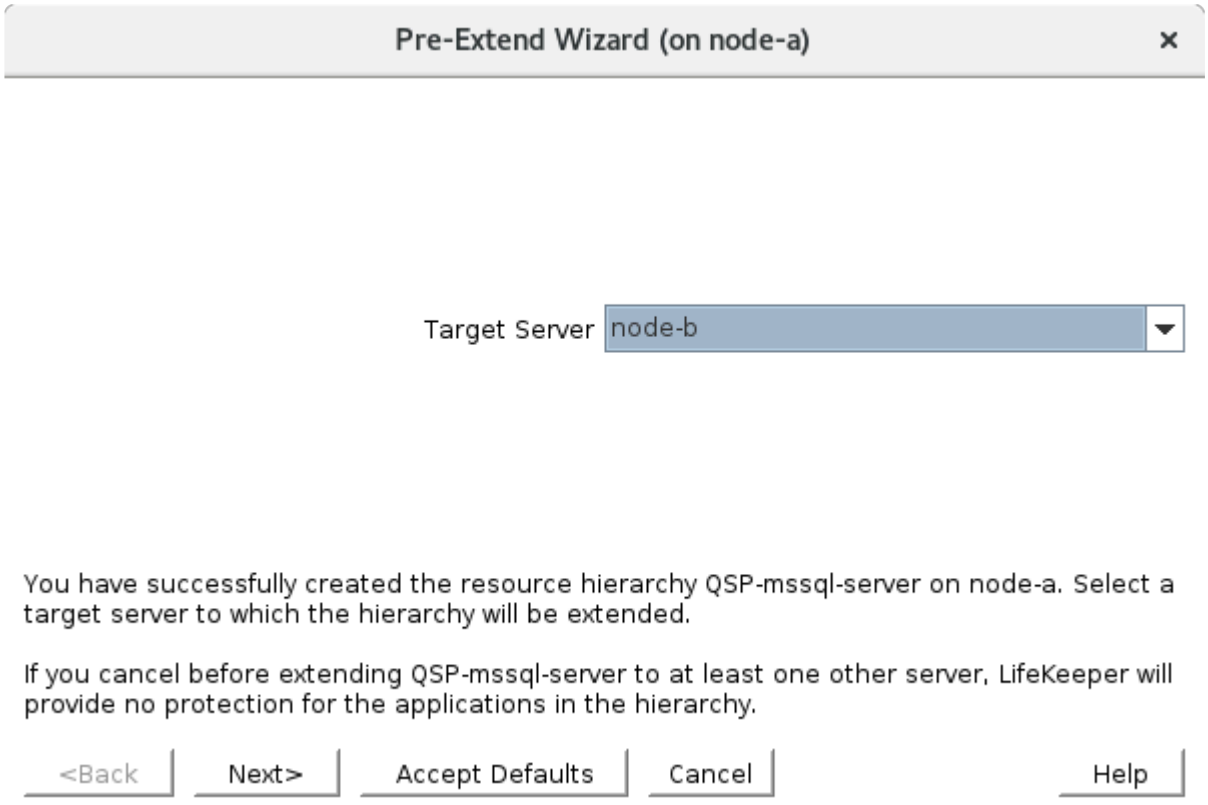

4. Enter the following values. Select the default values.

| Field | Value |
|---|---|
| Enable or Disable Monitoring | enable |
| Resource Tag | QSP-mssql-server |

5. The wizard reviews the values. Once it completes, click “Next >” to continue.



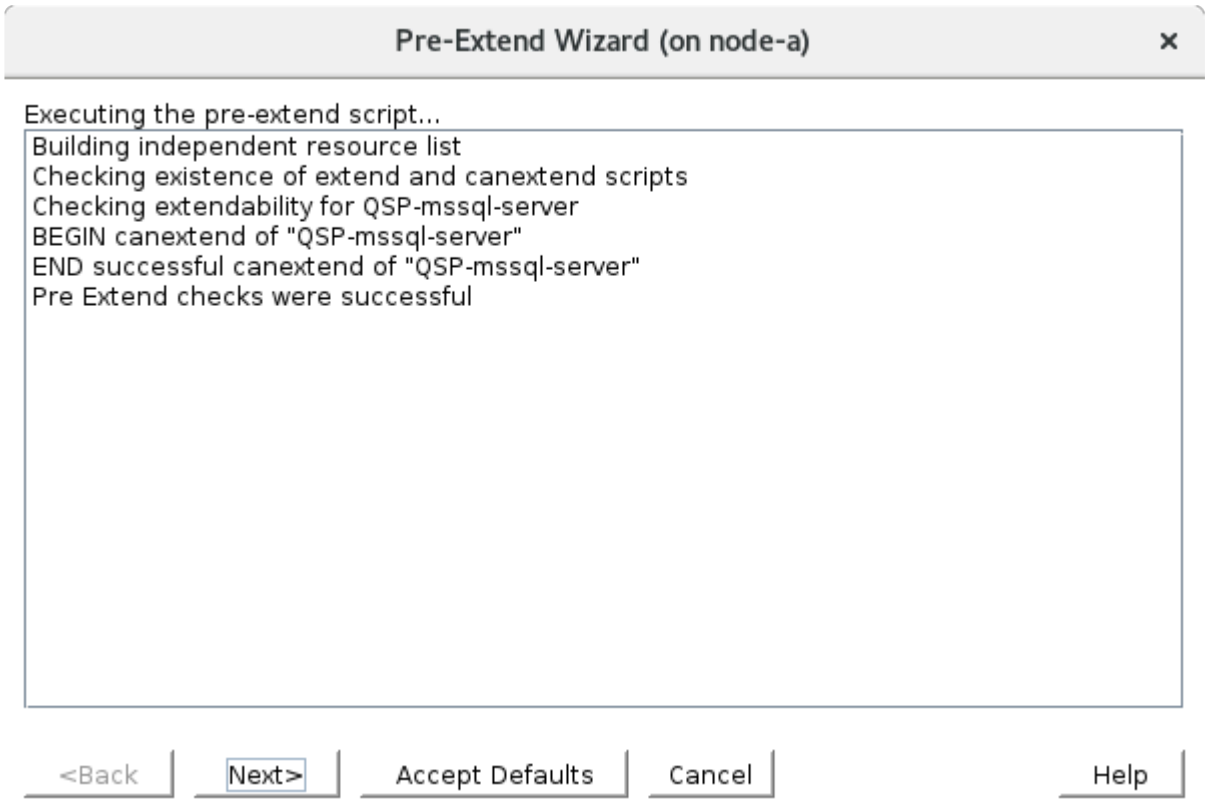

6. The next step is the “Pre-Extend Wizard”. Select the default values.



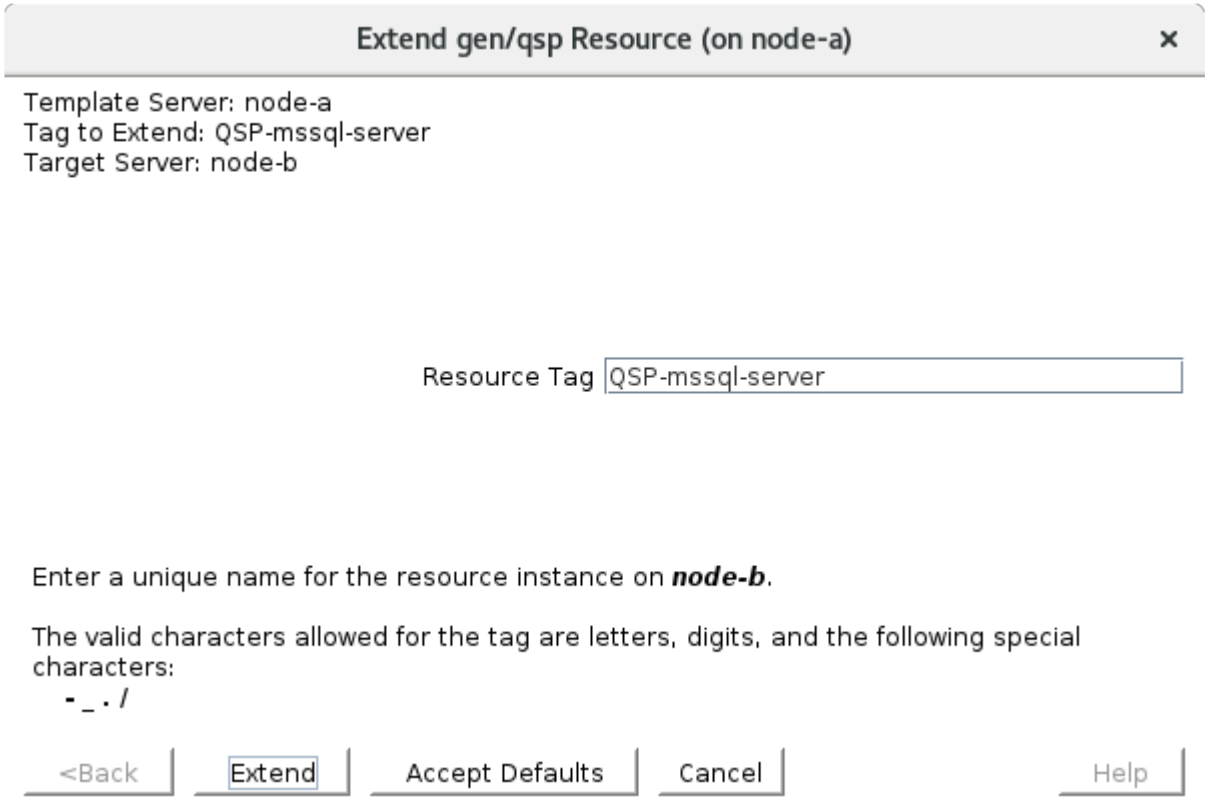

| Field | Value |
|---|---|
| Switchback Type | intelligent |

| | |
|---|---|
| Template Priority | 1 |
| Target Priority | 10 |

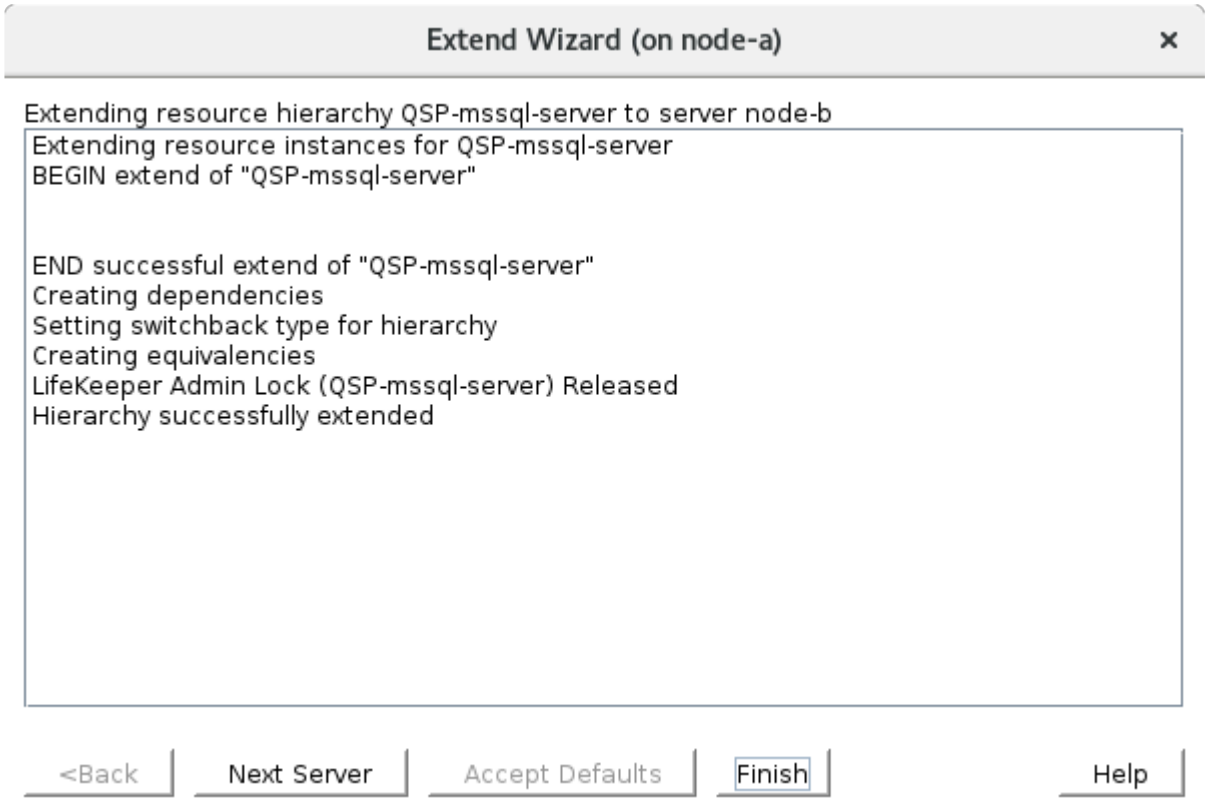
Once the checks on the ‘Pre-Extend Wizard’ have completed and passed, click ‘Next’ to continue.



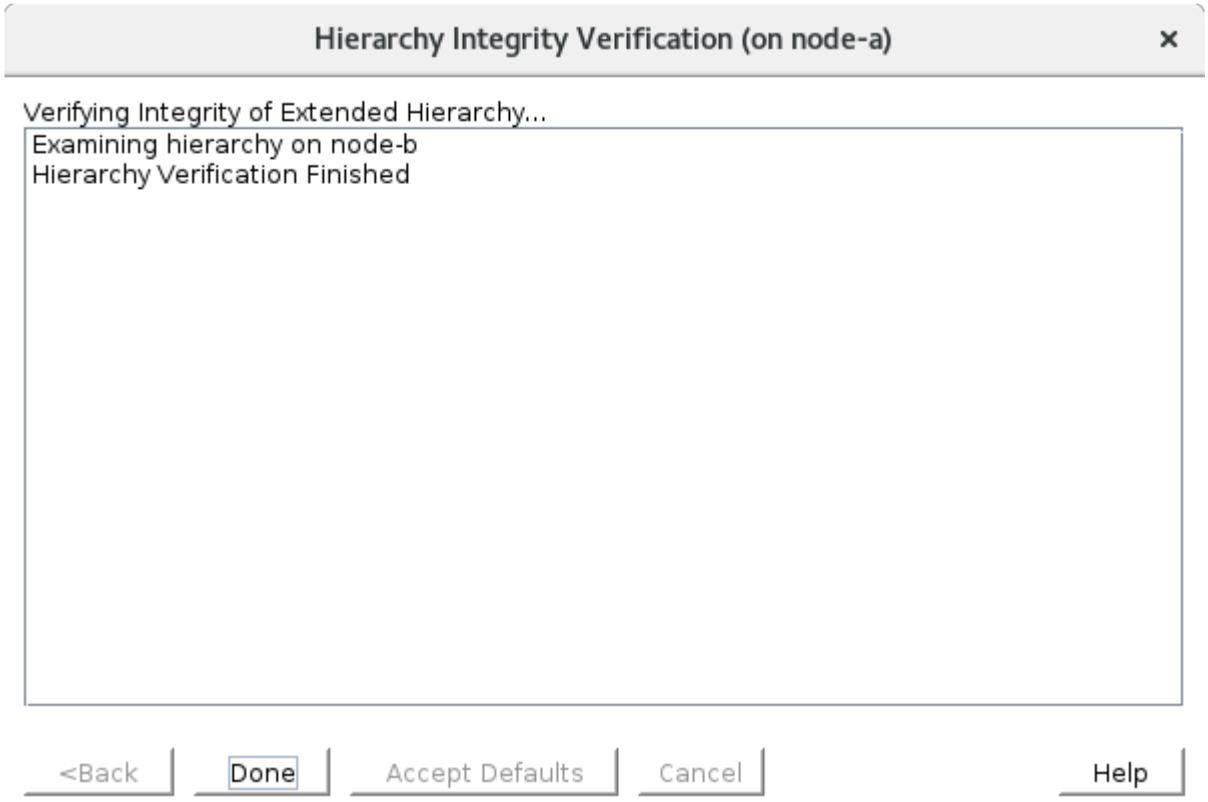

7. On the “Extend gen/qsp Resource” wizard, the first choice is selecting the Database Tag. Confirm the default value and click “Extend”.



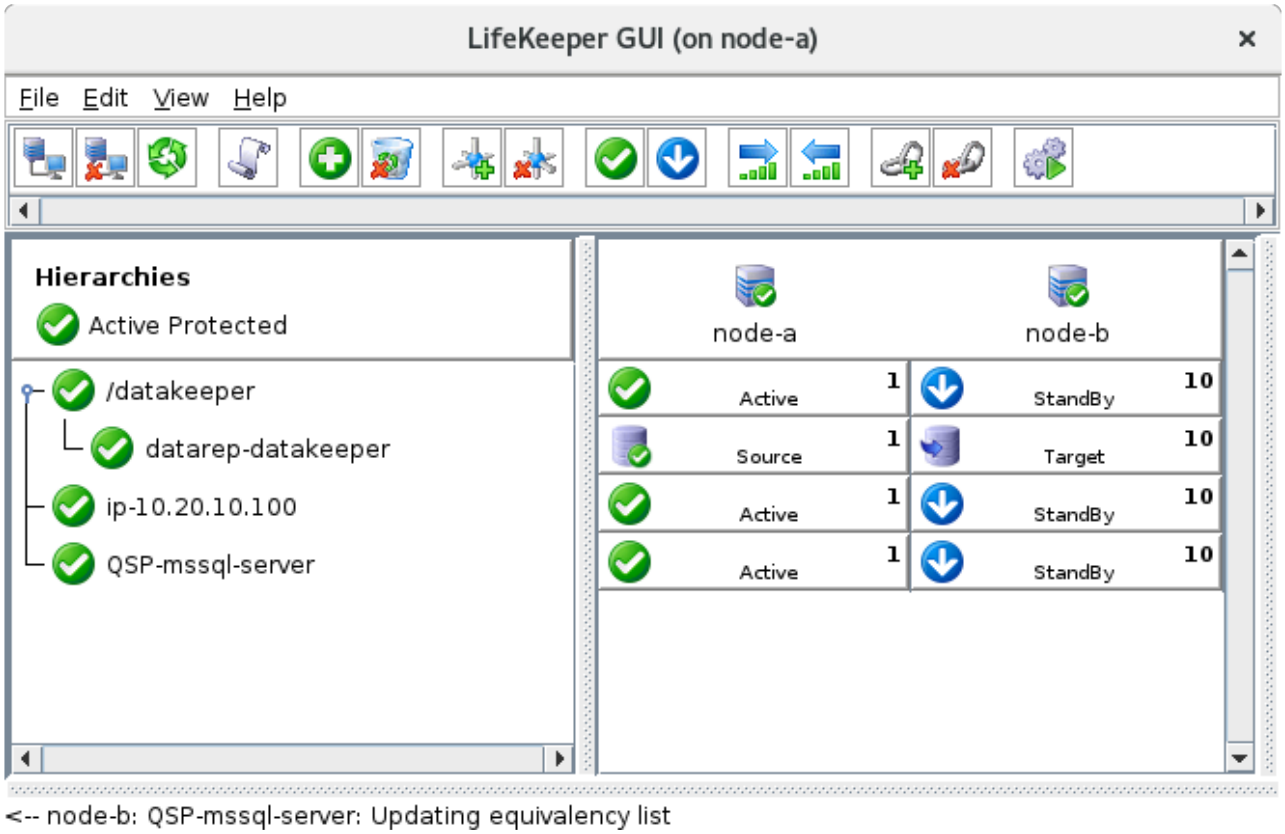

8. Once the Extend Wizard configures the gen/qsp resource, click “Finish” to complete.




9. The hierarchy has now been successfully created and verified. Click “Done” to complete.




10. Return to the LifeKeeper GUI. The “QSP-mssql-server” resource is defined as shown below.



# 11.2.7.5.2.5. Customize LocalRecovery Parameter on Both Nodes

The Quick Service Protection Wizard completes most of the configuration. However, we still need to customize some parameters.

> ✻ Execute this command on both node-a **AND** node-b.

First, we'll review some policy parameters using the `lkpolicy` tool as follows:

```
# /opt/LifeKeeper/bin/lkpolicy -g -v
Server-level Policies:
    Failover: On
    LocalRecovery: On
Resource-level Policies:
    QSP-mssql-server:
        Failover: On
        LocalRecovery: On    <--- This needs to be updated
    /datakeeper:
        Failover: On
        LocalRecovery: On
    datarep-datakeeper:
        Failover: On
        LocalRecovery: On
    ip-10.20.10.100:
        Failover: On
        LocalRecovery: On
```

The `LocalRecovery` parameter for QSP-mssql-server is currently set to On and needs to be changed to Off. Update it with the following command.

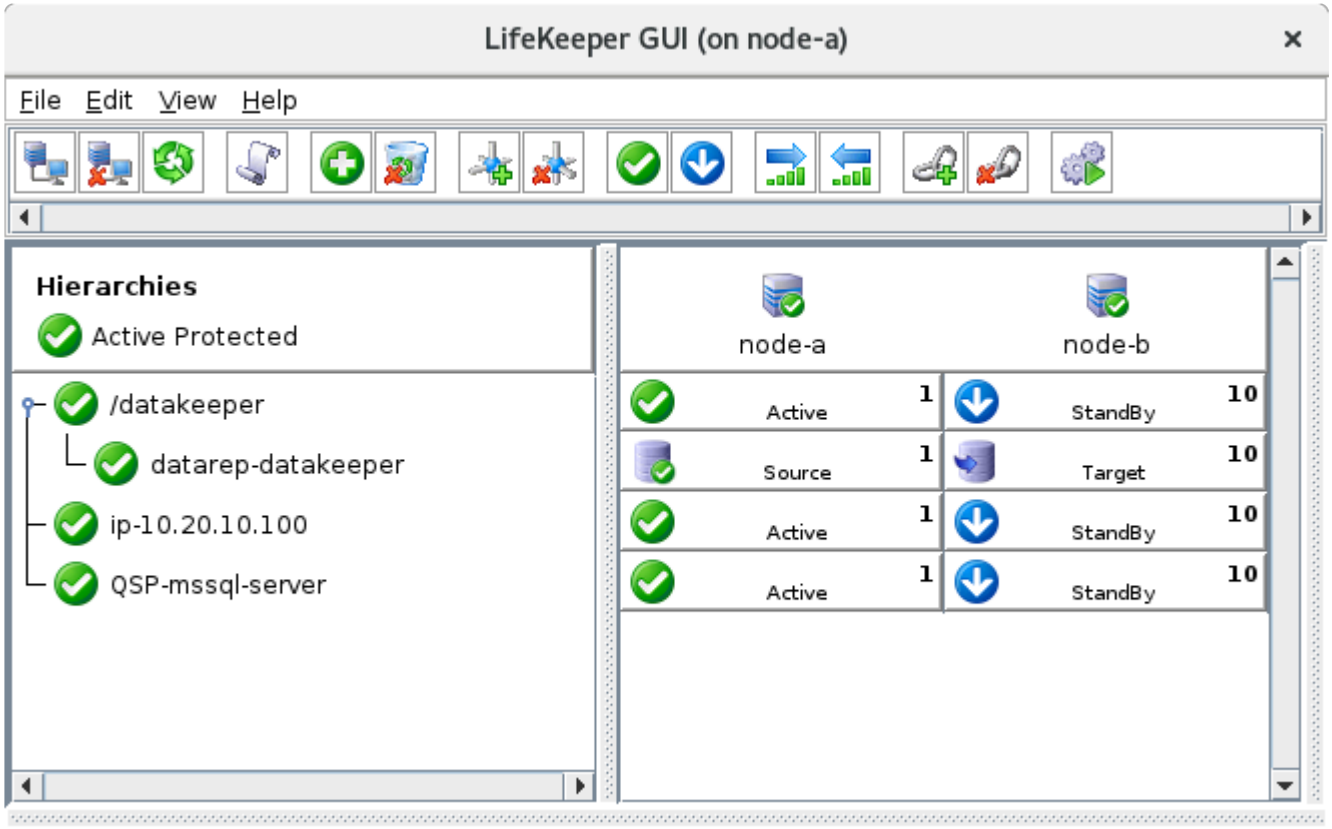
```
# /opt/LifeKeeper/bin/lkpolicy -s LocalRecovery -E tag="QSP-mssql-server"
Policy Disabled
```

Once the value is updated, confirm it as follows:

```
# /opt/LifeKeeper/bin/lkpolicy -g -v
Server-level Policies:
    Failover: On
    LocalRecovery: On
Resource-level Policies:
    QSP-mssql-server:
        Failover: On
        LocalRecovery: Off
    /datakeeper:
        Failover: On
        LocalRecovery: On
    datarep-datakeeper:
        Failover: On
        LocalRecovery: On
    ip-10.20.10.100:
        Failover: On
        LocalRecovery: On
```

# 11.2.7.5.2.6. Update Dependency between Resources

At this point, the LifeKeeper GUI shows the MSSQL related resources as shown below:



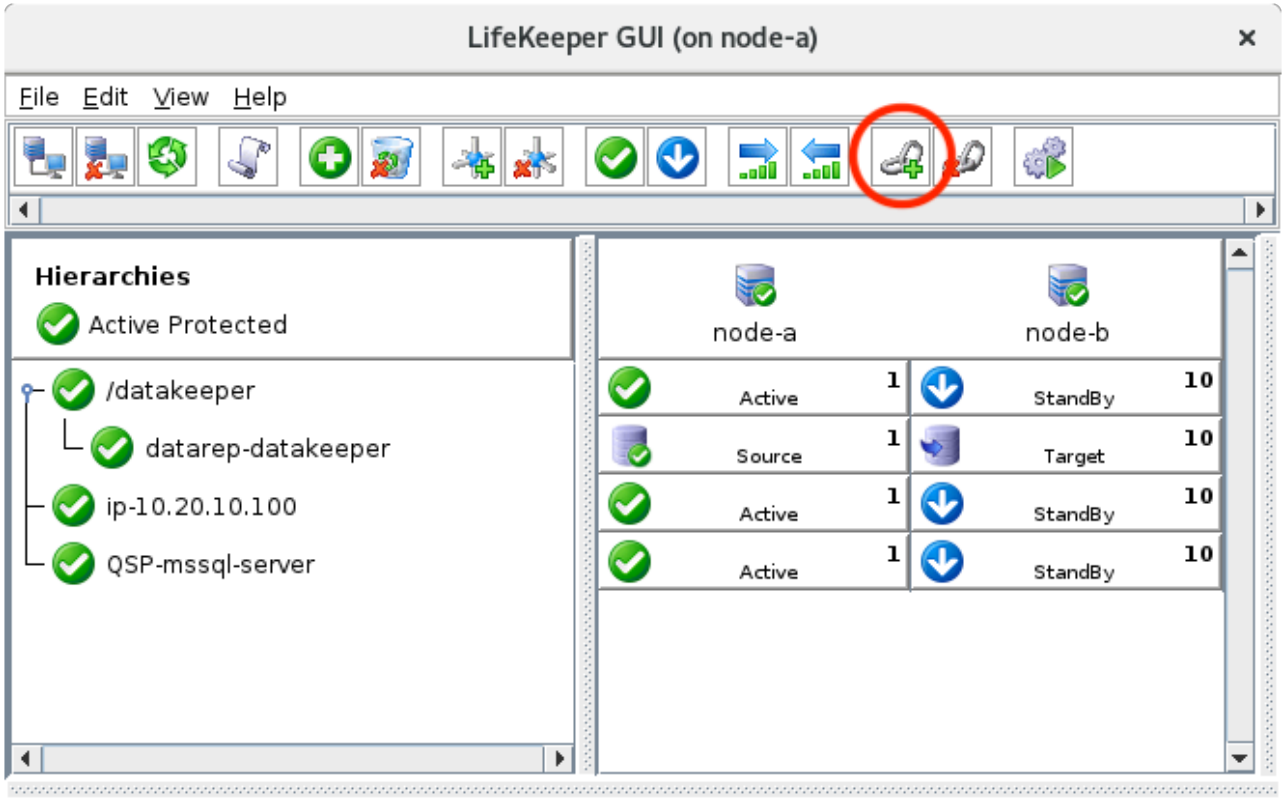

<-- node-b: QSP-mssql-server: Updating equivalency list

The QSP-mssql-server resource depends on the following resources:

- ip-10.20.10.100 (to connect to the database, we need the virtual IP to connect to)
- /datakeeper (to make the database work, we need to mount the underlying filesystem)

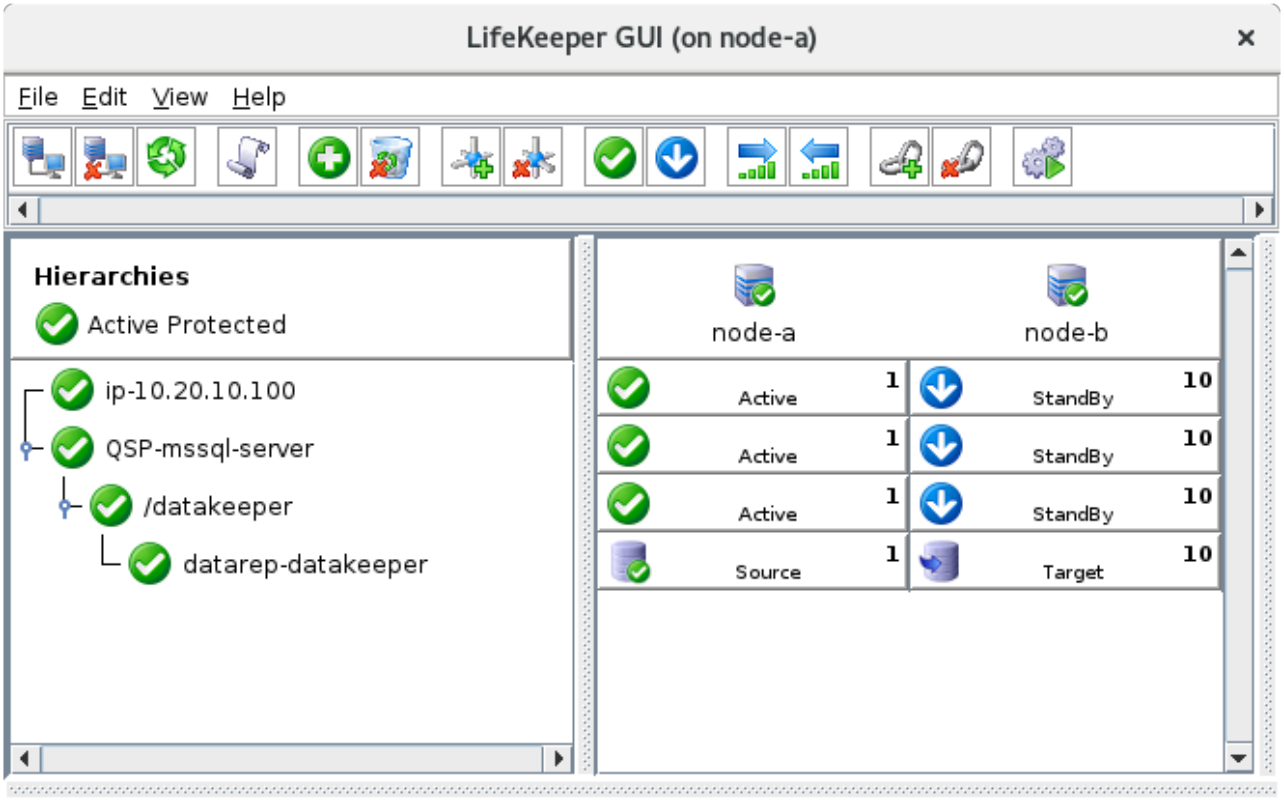
This section explains how to define the dependency between resources.

1. Select “Create Dependency”.

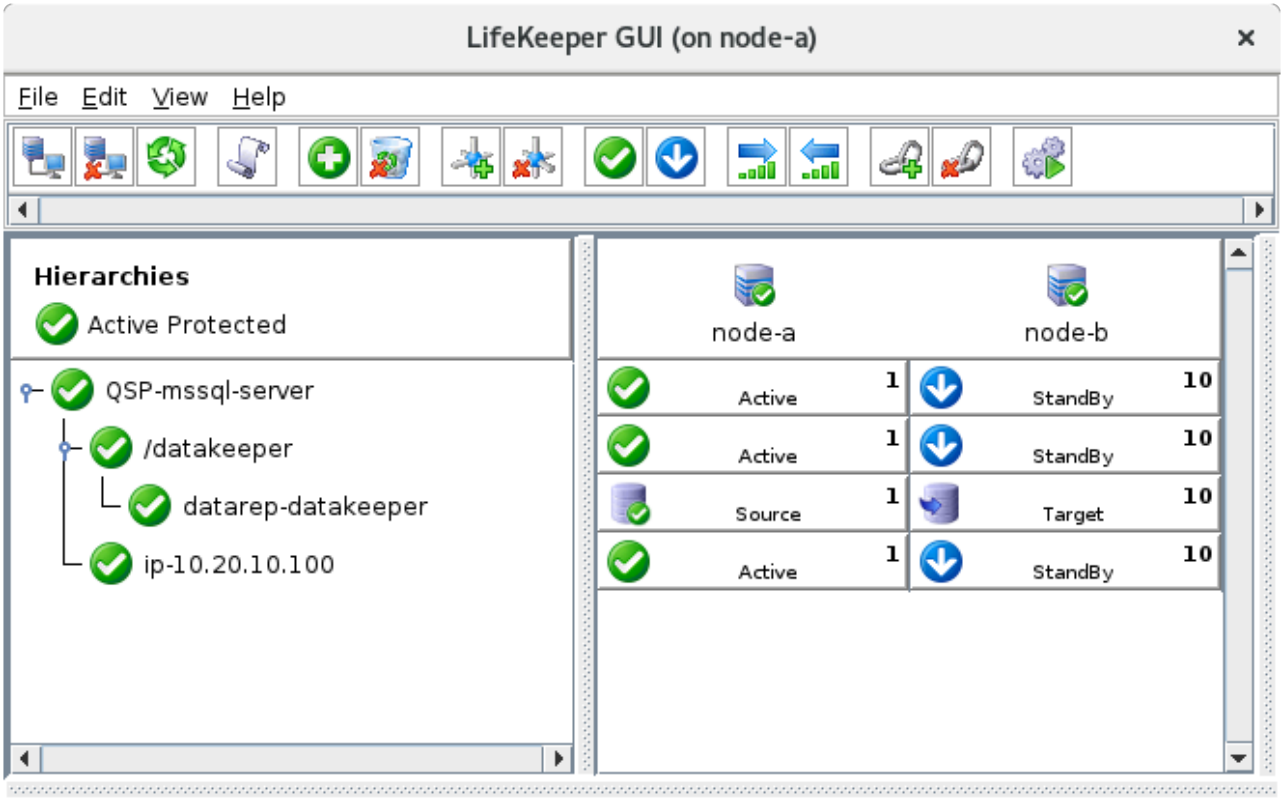



2. Select “node-a” as the server.
3. Select “QSP-mssql-server” as the Parent Resource Tag.
4. Select “datakeeper” as the Child Resource Tag.

Once the dependency is defined, the LifeKeeper GUI shows the relationship as follows:




5. Repeat the steps with “ip-10.20.10.100”. Now the dependencies are defined.




<-- node-b: ip-10.20.10.100: Adding parent: QSP-mssql-server

# 11.2.7.5.3. Protecting a PostgreSQL Resource

> ! This section uses datakeeper/pgsql/data/ as the location for Postgres data. In the steps below, this directory will be created on the replicated storage. If a different directory structure is created, be sure to account for the specific environment variances throughout the remainder of the steps described below.

Before configuring Postgres 12, please follow the instructions at Install Postgres 12 to Linux Nodes.

> ✻ Once PostgreSQL is installed on both the primary and standby nodes, execute the following steps on the primary node **only**.

## Create a Directory to Store Data on the Primary Node

```
[root@node-a]# mkdir -m 700 -p /datakeeper/pgsql/data
[root@node-a]# chown postgres:postgres -R /datakeeper/pgsql
```

## Initialize Postgres Database Server on the Primary Node

In this instance, `initdb` is located at `/usr/pgsql-12/bin/initdb`. Use the location of `initdb` to construct the parameters here.

```
[root@node-a ~]# su postgres -c "/usr/pgsql-12/bin/initdb -D /datakeeper/pgsql/data"
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.

(snip)

Success. You can now start the database server using:

    /usr/pgsql-12/bin/pg_ctl -D /datakeeper/pgsql/data -l logfile start
```

> ✻ Check the following locations to be used during the following steps.

| Item | Location |
|---|---|
| Location of Postgres binaries | `/usr/pgsql-12/bin/` |

| | |
|---|---|
| Data | /datakeeper/pgsql/data |

## Start the Postgres Process on the Primary Node

```
[root@node-a ~]# su postgres -c "/usr/pgsql-12/bin/pg_ctl start -D /datakeepe
r/pgsql/data"
waiting for server to start....2020-12-30 03:40:20.925 UTC [6346] LOG:  starti
ng PostgreSQL 12.5 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 4.8.5 2015062
3 (Red Hat 4.8.5-39), 64-bit
2020-12-30 03:40:20.926 UTC [6346] LOG:  listening on IPv6 address "::1", por
t 5432
2020-12-30 03:40:20.926 UTC [6346] LOG:  listening on IPv4 address "12
7.0.0.1", port 5432
2020-12-30 03:40:20.932 UTC [6346] LOG:  listening on Unix socket "/var/run/po
stgresql/.s.PGSQL.5432"
2020-12-30 03:40:20.946 UTC [6346] LOG:  listening on Unix socket "/tmp/.s.PGS
QL.5432"
2020-12-30 03:40:20.960 UTC [6346] LOG:  redirecting log output to logging col
lector process
2020-12-30 03:40:20.960 UTC [6346] HINT:  Future log output will appear in dir
ectory "log".
 done
server started
```

Check the following locations to be used during the following steps.

| Item | Location |
|---|---|
| Unix socket | /tmp/.s.PGSQL.5432 |

## Protecting PostgreSQL Resource with LifeKeeper

1. Click in the LifeKeeper user interface.
2. The Create Resource Wizard at node-a will appear. Select PostgreSQL Database as the Recovery Kit.
3. Select the following parameters.

| Item | Location |
|---|---|
| Switchback Type | intelligent |

| | |
|---|---|
| Server | node-a |
| PostgreSQL Executable Location | `/usr/pgsql-12/bin/` |
| PostgreSQL Client Executable Location | /usr/pgsql-12/bin/psql |
| PostgreSQL Administration Executable Location | /usr/pgsql-12/bin/pg_ctl |
| PostgreSQL Data Directory | `/datakeeper/pgsql/data` |
| PostgreSQL Port | 5432 |
| PostgreSQL Socket Path | `/tmp/.s.PGSQL.5432` |
| Enter Database Administrator User | postgres |
| PostgreSQL Logfile | /tmp/pgsql-5432.lk.log |
| PostgreSQL Database Tag | pgsql-5432 |

The wizard checks these values. Once “pgsql-5432” is successfully created on node-a, continue to the next steps.

4. Select the following values in the Pre-Extend Wizard @ node-a.

| Item | Location |
|---|---|
| Target Server | node-b |
| Switchback Type | intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

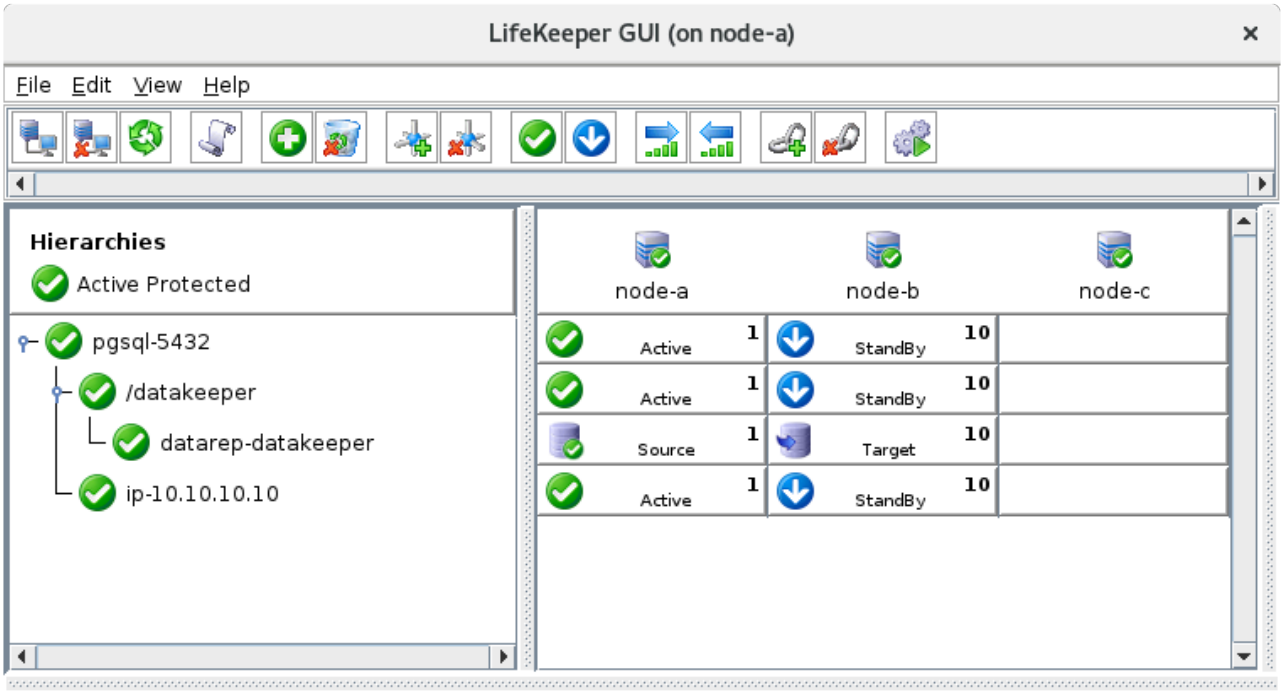
The wizard checks these values. Once the Pre-Extend check is complete, continue to the next steps.

5. Select the following values in the Extend database/pgsql Resource @ node-a wizard.

| Field | Value |
|---|---|
| PostgreSQL Executable Location | `/usr/pgsql-12/bin/` |
| PostgreSQL Database Tag | pgsql-5432 |

Now the PostgreSQL resource is defined on LifeKeeper. The wizard automatically defines a

dependency between the PostgreSQL resource and the replicated disk (`/datakeeper`).




<-- node-b: ip-10.10.10.10: Adding parent: pgsql-5432

# 11.2.7.5.3.1. Install Postgres 12 on Linux Nodes

This section outlines the basic steps required to install Postgres 12 on Linux nodes.

✻ Postgres must be installed on both the active and standby nodes.

- Instructions for installing Postgres are available at https://www.postgresql.org/download/.

By selecting the Linux distribution, version, and architecture, customized installation instructions will be provided. Follow the instructions up to the point immediately before installing the product. Because `initdb` should be called differently between node-a and node-b, those steps are discussed on the parent page.

In the case of RedHat 8 or CentOS 8, execute the following sections:

```
# Install the repository RPM:
# dnf install -y https://download.postgresql.org/pub/repos/yum/reporpms/EL-8-x
86_64/pgdg-redhat-repo-latest.noarch.rpm
# Disable the built-in PostgreSQL module:
# dnf -qy module disable postgresql
# Install PostgreSQL:
# dnf install -y postgresql12-server
```

! Please stop at the step before executing `initdb`.

## Set Password to the postgres User

```
1 # passwd postgres
```

## Change Directory to Store Data to Under /datakeeper/pgsql/data

```
# cp -p /var/lib/pgsql/.bash_profile /var/lib/pgsql/.bash_profile.org
# vi /var/lib/pgsql/.bash_profile
# cat /var/lib/pgsql/.bash_profile
[ -f /etc/profile ] && source /etc/profile
PGDATA=/datakeeper/pgsql/data
export PGDATA
# If you want to customize your settings,
# Use the file below. This is not overridden
# by the RPMS.
[ -f /var/lib/pgsql/.pgsql_profile ] && source /var/lib/pgsql/.pgsql_profile
```

# 11.2.7.5.4. Protecting an NFS Resource

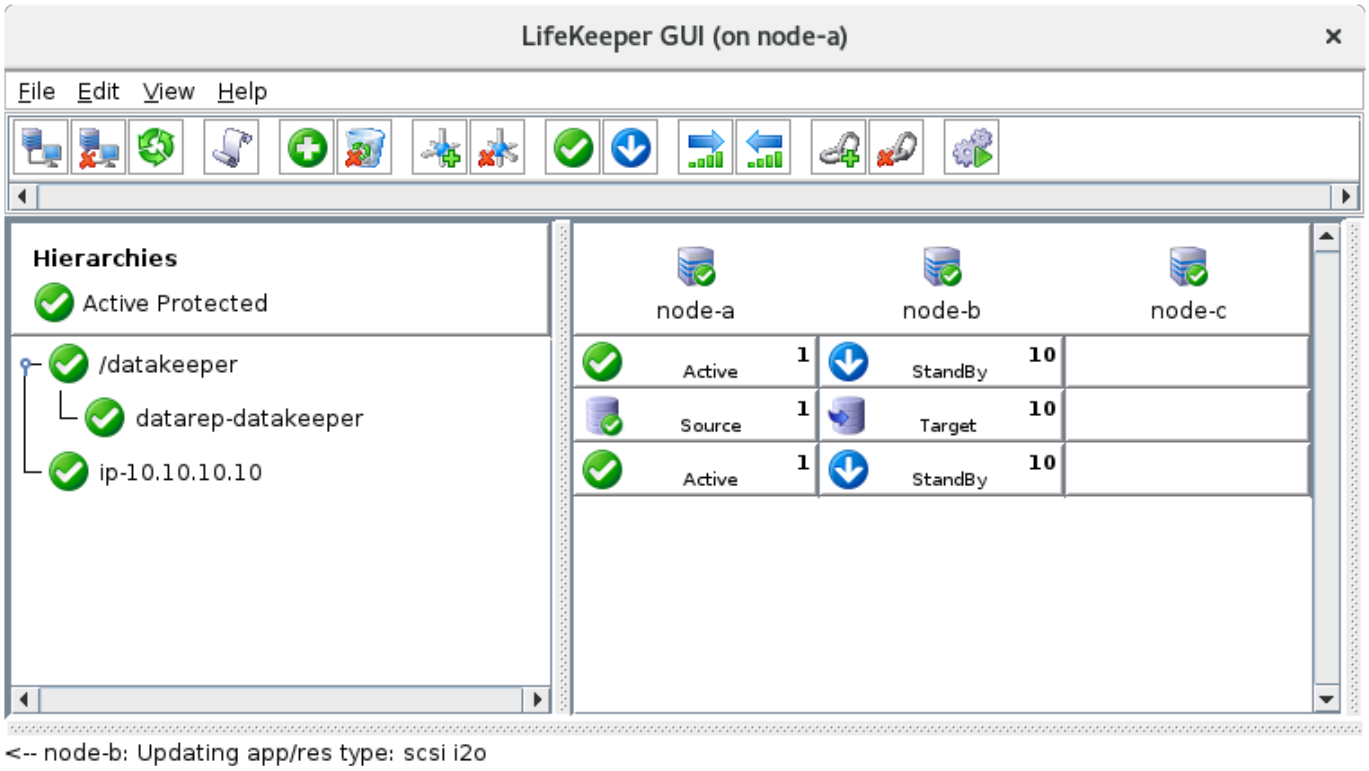
This section describes the steps required to protect a cluster of NFS servers.

## Before Beginning

This section assumes that the following 2 resources are configured on the cluster.

- IP Resource
- Data Replication at `/datakeeper`

The LifeKeeper GUI should look like the following prior to installing the NFS software.




<-- node-b: Updating app/res type: scsi i2o

## Install NFS Server Software

> Install `nfs-utils` on both nodes.

Install `nfs-utils` on both node-a and node-b.

```
# yum -y install nfs-utils
(snip)
Complete!
```

# Configure the NFS Server

> ✻ The following steps should be performed on node-a **only**.

Edit `/etc/exports` to define the NFS export. We are using the shared storage (`/datakeeper`) that is protected by DataKeeper. Create `/datakeeper/nfs/data` and use it as the location of the export.

Once the `/etc/exports` file is edited it should look like the following:

```
[root@node-a ~]# mkdir -p /datakeeper/nfs/data
[root@node-a ~]# vi /etc/exports
[root@node-a ~]# cat /etc/exports
/datakeeper/nfs/data 10.20.0.0/22(rw,no_root_squash)
```

Start the NFS process on node-a.

```
[root@node-a ~]# systemctl start rpcbind nfs-server
```

# Confirm Access to the NFS Server from a Client

> ✻ The following steps should be done on a client machine.

On the client machine (e.g., node-c), complete the following steps to install the NFS software. Mount the NFS export from node-a to a local folder (e.g., mount to `/local/nfsclient`).

```
[root@node-c ~]# yum -y install nfs-utils
(snip)
Complete!
[root@node-c /]# mkdir -p /local/nfsclient
[root@node-c ~]# mount -t nfs -o nfsvers=4 node-a:/datakeeper/nfs/data /local/nfsclient
```

Now the content on the NFS server should be visible. Disconnect (unmount) from the NFS server to configure the NFS cluster.

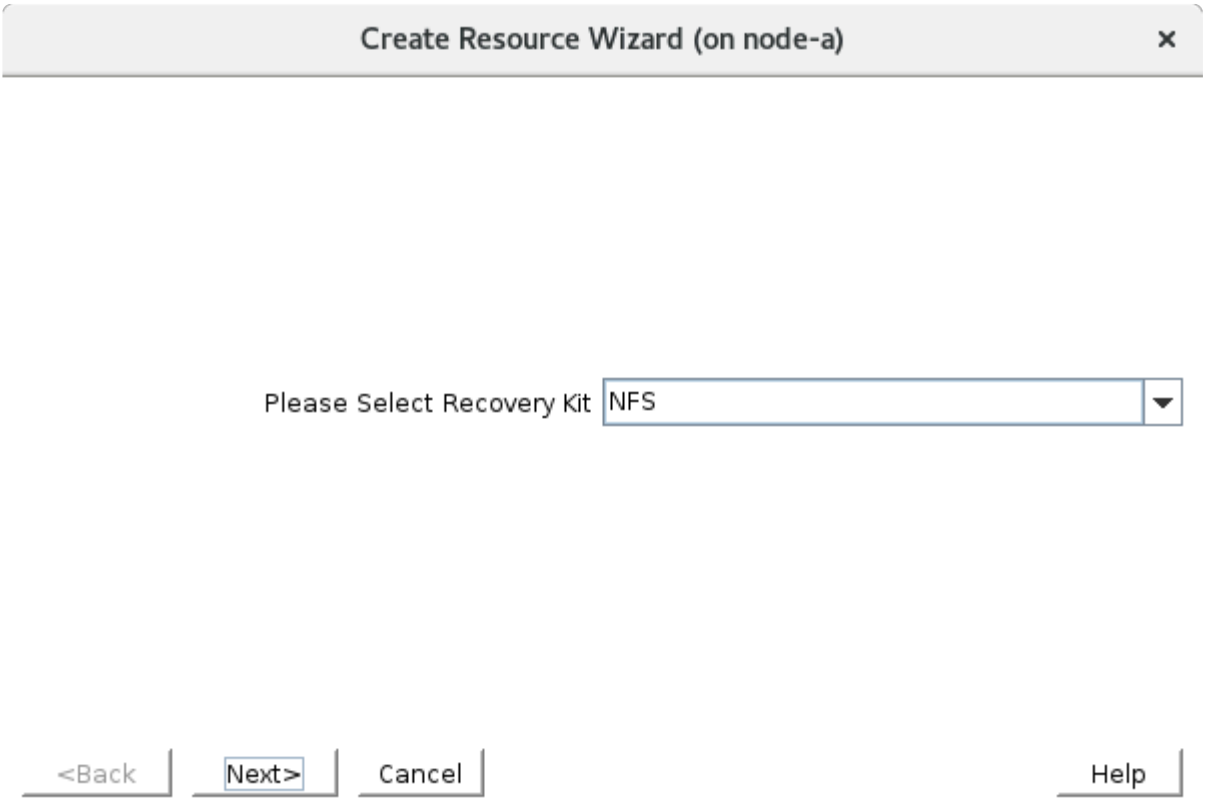
```
[root@node-c ~]# umount /local/nfsclient
```

# Protecting an NFS Resource using LifeKeeper

Complete the follow these steps on node-a.

1. Select on the LifeKeeper user interface.

2. The Create Resource Wizard at node-a screen will appear. Select the NFS Recovery Kit as the Recovery Kit.




3. Select the following parameters.

| Item | Location |
|---|---|
| Switchback Type | intelligent |
| Server | node-a |
| Export Point | /datakeeper/nfs/data |
| IP Tag | ip-10.10.10.10 |
| NFS Tag | nfs-/datakeeper/nfs/data |

The wizard checks these values. Once “nfs-/datakeeper/nfs/data” is successfully created on node-

a, continue to the next steps.

4. Select the following values on Pre-Extend Wizard @ node-a.

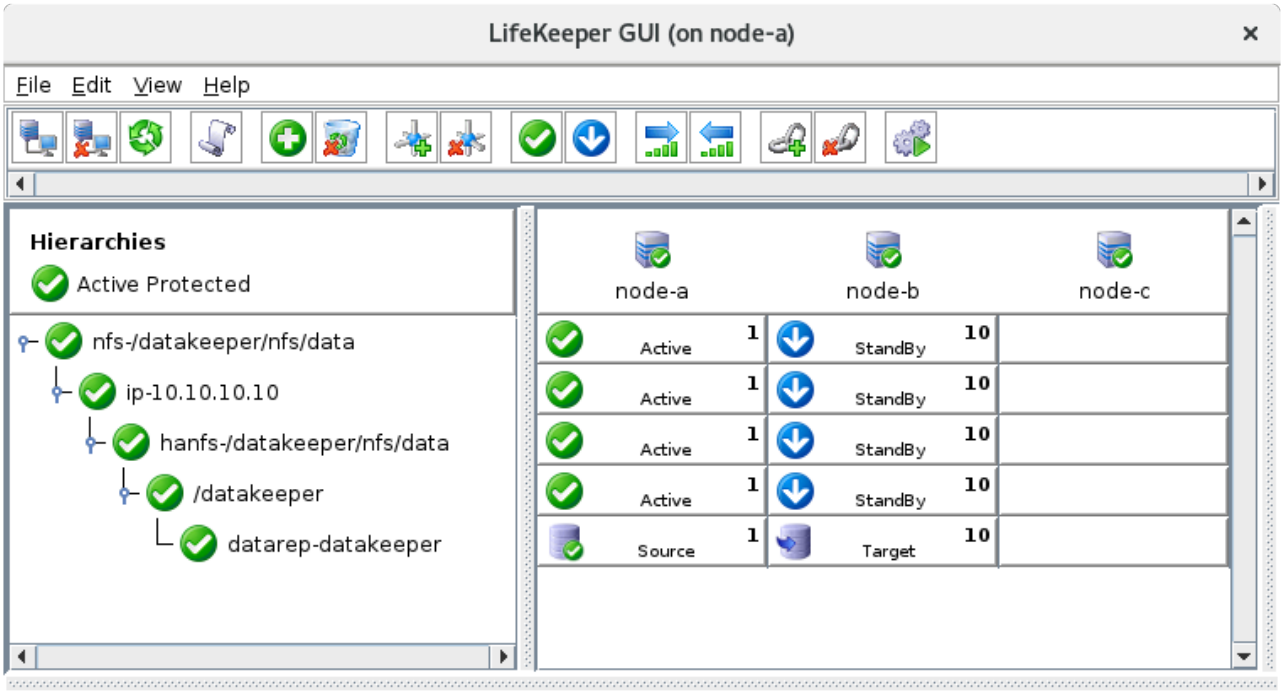
| Field | Value |
|---|---|
| Target Server | node-b |
| Switchback Type | intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

The wizard checks these values. Once the Pre-Extend Check is completed, move on to the next steps.

5. Select the following values on the Extend gen/nfs Resource Hierarchy @ node-a wizard.

| Field | Value |
|---|---|
| NFS Tag | nfs-/datakeeper/nfs/data |

The NFS resource hierarchy is now defined in LifeKeeper. The wizard automatically defines a dependency between the NFS resource and required resources (IP resource and `/datakeeper` resource).



# 11.2.7.5.5. Protecting SAP Resources

In this section we will deploy an SAP S/4HANA 1909 AS ABAP environment and use the LifeKeeper SAP Recovery Kit to protect the ABAP SAP Central Services (ASCS) and corresponding Enqueue Replication Server (ERS) instances. While it is possible to protect a redundant Primary Application Server (PAS) instance using LifeKeeper, we will instead take the approach of creating an external application server pool consisting of one PAS instance and one Additional Application Server (AAS) instance spread across two servers.

Since the PAS and AAS instances will not be protected by LifeKeeper, this guide only details the installation and configuration of the highly-available Central Services (ASCS/ERS) cluster. This section also does not describe the configuration of a highly-available database cluster (e.g., using SAP HANA). Please see the section of the guide relevant to the database that is being installed and adapt it to fit your planned SAP landscape.

The details of the Central Services cluster nodes that will be created are given below. The instance provisioning described in this guide gives the **minimum** required specifications to run the applications in an evaluation context, and is not generally sufficient to handle productive workloads. Consult the documentation provided by SAP and your cloud provider for best practices when provisioning VM's in a production environment. For example, see Hardware Requirements for SAP System Hosts.

- node-a (Primary ASCS instance host, Standby ERS instance host)
  - Private IP: 10.20.1.10
  - vCPU's: 2
  - Memory: 4GB
  - SAP NFS shares mounted
  - SAP instances ASCS10 and ERS20 installed under SID 'SPS'
  - SIOS Protection Suite for Linux installed with quorum/witness functionality enabled, SAP Recovery Kit installed
  - [GCP] GenLB Recovery Kit installed

- node-b (Primary ERS instance host, Standby ASCS instance host)
  - Private IP: 10.20.2.10
  - vCPU's: 2
  - Memory: 4GB
  - SAP NFS shares mounted
  - SAP instances ASCS10 and ERS20 installed under SID 'SPS'
  - SIOS Protection Suite for Linux installed with quorum/witness functionality enabled, SAP Recovery Kit installed
  - [GCP] GenLB Recovery Kit installed

- node-c (Quorum/Witness node)
  - Private IP: 10.20.3.10
  - vCPU's: 1
  - Memory: 2GB
  - SIOS Protection Suite for Linux installed with quorum/witness functionality enabled

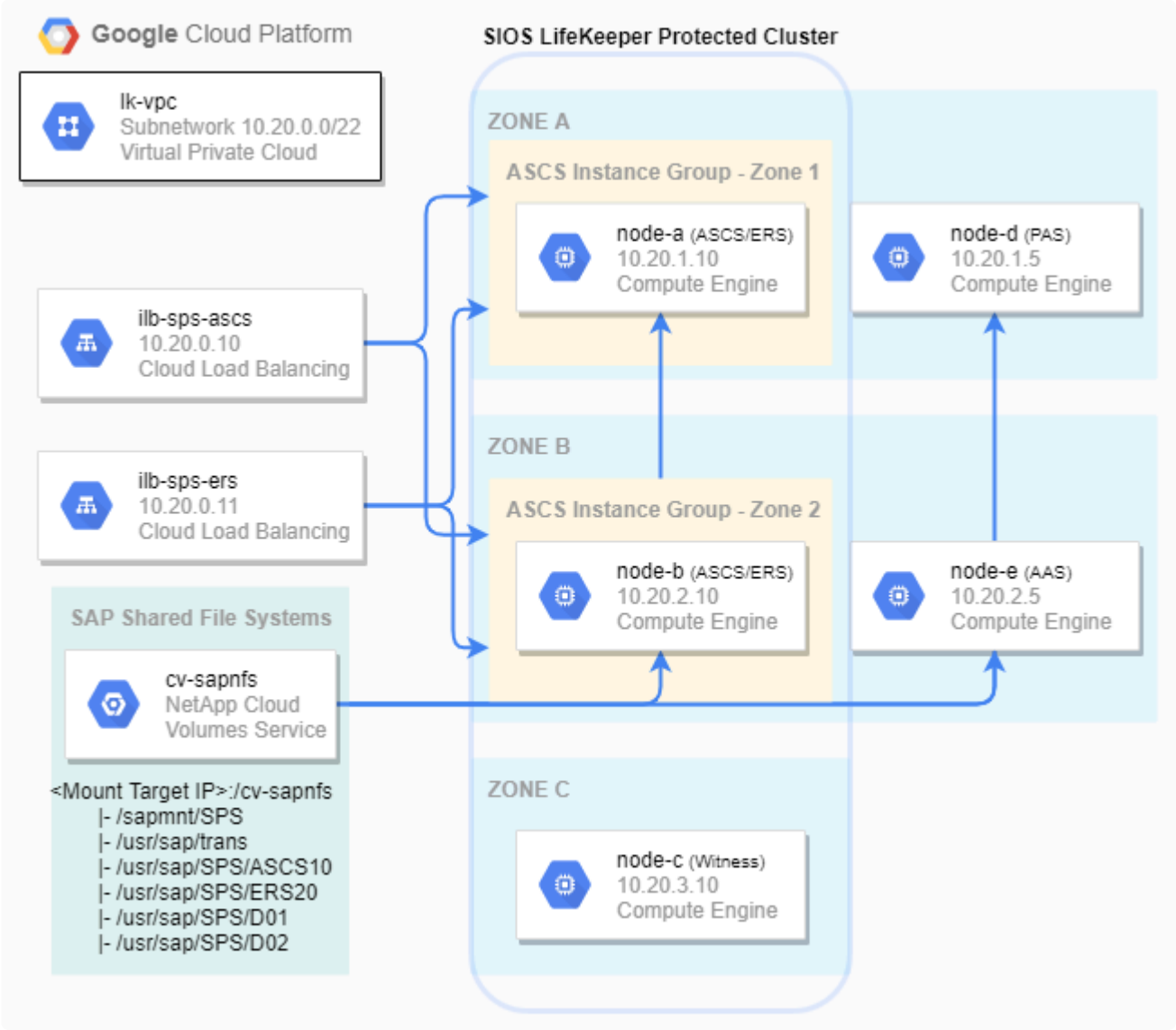
- node-d (Primary Application Server host)
  - Private IP: 10.20.1.5
  - vCPU's: 2
  - Memory: 4GB
  - SAP NFS shares mounted
  - SAP PAS instance D01 installed under SID 'SPS'

- node-e (Additional Application Server host)
  - Private IP: 10.20.2.5
  - vCPU's: 2
  - Memory: 4GB
  - SAP NFS shares mounted
  - SAP AAS instance D02 installed under SID 'SPS'

- ASCS10 virtual hostname: sps-ascs

- ERS20 virtual hostname: sps-ers

- NFS shared file systems:
  - /sapmnt/SPS
  - /usr/sap/trans
  - /usr/sap/SPS/ASCS10
  - /usr/sap/SPS/ERS20




This section assumes that the following prerequisite steps have already been completed:

1. Three VM instances (node-a, node-b, and node-c) have been created using the networking conventions described earlier in the guide. Firewall rules are in place to allow inter-node communication as well as SSH connections. See Configuring Network Components and Creating Instances for details.

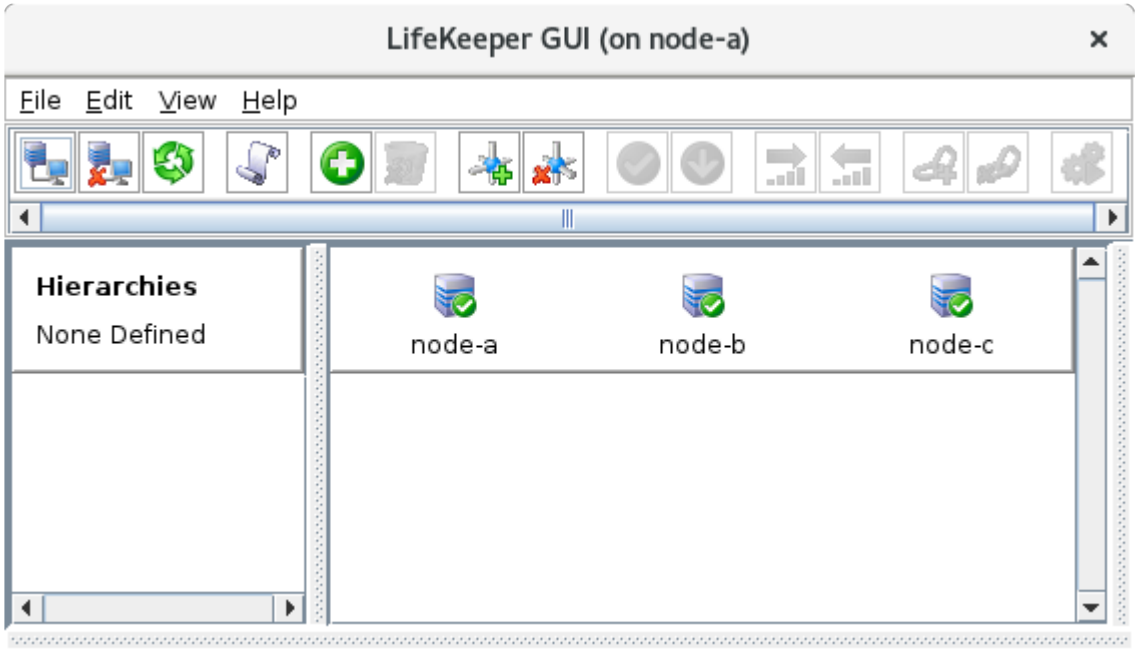
> ! If deploying in Google Cloud, the “RedHat Enterprise Linux 8.2 for SAP Applications” boot image must be used in order to successfully follow the steps in this guide. Due to networking differences between boot images, the workaround script described in the **Enabling Load Balancer Traffic from Backend Servers** section of GCP – Using an Internal Load Balancer may cause unexpected behavior on VMs running different operating systems.

2. All three VM instances have been configured to run SIOS Protection Suite for Linux. In particular, SELinux is disabled, the firewall is disabled, the /etc/hosts file on each node contains entries to resolve each node’s hostname to its private IP, and the root user has the same password on each node. See Configure Linux Nodes to Run SIOS Protection Suite for Linux for details.

3. SIOS Protection Suite for Linux has been installed on all three nodes with quorum/witness functionality enabled. The SAP Recovery Kit has been installed on node-a and node-b. On node-c (the witness node), no additional recovery kits beyond the core LifeKeeper installation are required. All necessary SIOS licenses have been installed on each node. See Install SIOS Protection Suite for Linux for details.

4. LifeKeeper communication paths have been defined between all pairs of cluster nodes. Note that this requires creation of **three** bi-directional communication paths (node-a ↔ node-b, node-a ↔ node-c, node-b ↔ node-c). See Login and Basic Configuration Tasks for details.

After completing all of these tasks, the LifeKeeper GUI should resemble the following image.



# 11.2.7.5.5.1. Create ASCS and ERS Virtual IPs

As of SAP AS ABAP release 7.53 (ABAP Platform 1809) and the introduction of the Enqueue Standalone Framework v2 (ENSAv2), the ASCS and ERS instances communicate with each other via associated virtual hostnames. The virtual host names sps-ascs and sps-ers will be configured to resolve to the current hosts of the ASCS10 and ERS20 instances, respectively.

The recommended implementation of these virtual IP addresses varies by cloud platform. Please follow the steps provided in:

- GCP – Creating ASCS and ERS Internal Load Balancers

# 11.2.7.5.5.1.1. GCP – Creating ASCS and ERS Internal Load Balancers

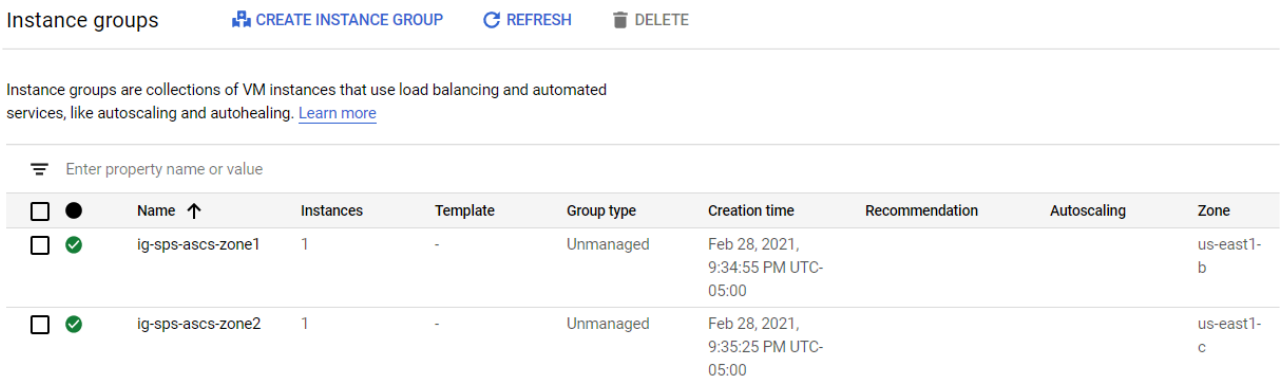
In Google Cloud, TCP Internal Load Balancers are used to facilitate failover of the ASCS and ERS virtual IP's. The frontend of each load balancer is assigned an ephemeral IP from the subnet that it operates in. In this example, we will be using the following IP addresses for load balancers corresponding to each virtual host:

- ASCS10 instance virtual host:
  - Virtual hostname: sps-ascs
  - Internal TCP load balancer: ilb-sps-ascs
  - IP address: 10.20.0.10

- ERS20 instance virtual host:
  - Virtual hostname: sps-ers
  - Internal TCP load balancer: ilb-sps-ers
  - IP address: 10.20.0.11

Following the steps provided in GCP – Using an Internal Load Balancer and Responding to Load Balancer Health Checks, create and configure the following Google Cloud and LifeKeeper resources.

- Create two unmanaged instance groups: **ig-sps-ascs-zone1**, containing node-a, and **ig-sps-ascs-zone2**, containing node-b.




2. Create two TCP Internal Load Balancers, **ilb-sps-ascs** and **ilb-sps-ers**, with the following properties. The ✓ icon indicates that the default option is chosen.

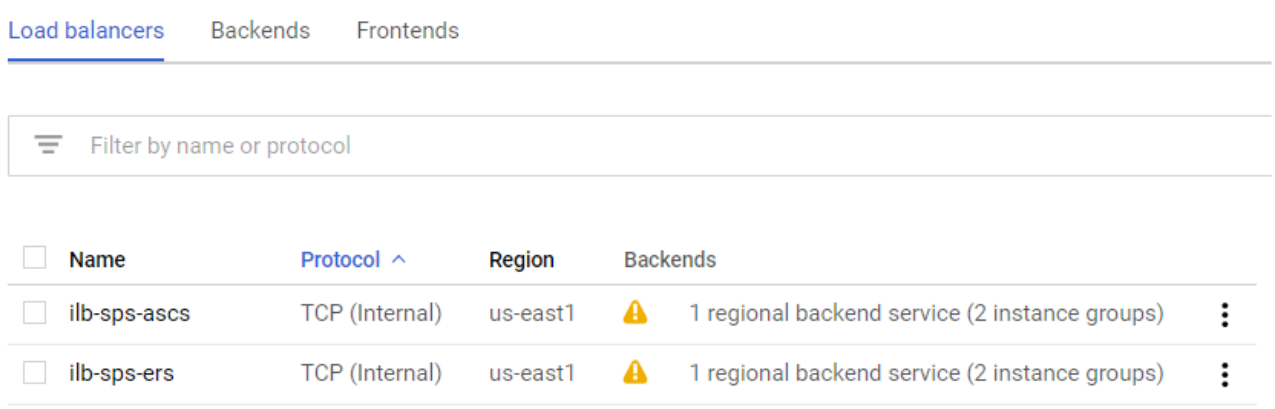
| Name | ilb-sps-ascs |
|---|---|
| **Backend configuration** | |
| Region | <Deployment region> (e.g., us-east1) |
| Network | lk-vpc |
| Backends | ig-sps-ascs-zone1<br>ig-sps-ascs-zone2 |

| **Health Check** | |
|---|---|
| Name | hc-sps-ascs |
| Protocol | TCP |
| Port | 51098 (This must agree with the corresponding LifeKeeper GenLB resource created later) |
| Proxy protocol | NONE |
| Request | Leave empty |
| Response | Leave empty |
| Check interval | 2 seconds |
| Timeout | 2 seconds |
| Healthy threshold | 2 consecutive successes |
| Unhealthy threshold | 3 consecutive failures |
| Connection draining timeout | 10 seconds |
| **Frontend configuration** | |
| Name | fe-sps-ascs |
| Subnetwork | lk-subnet |
| Purpose | Non-shared |
| IP address | Ephemeral (Custom) |
| Custom ephemeral IP address | 10.20.0.10 |
| Ports | All |
| Global access | Disable |
| Service label | Leave empty |

| **Name** | **ilb-sps-ers** |
|---|---|
| **Backend configuration** | |
| Region | <Deployment region> (e.g., us-west1) |
| Network | lk-vpc |
| Backends | ig-sps-ascs-zone1 |

| | |
|---|---|
| | ig-sps-ascs-zone2 |
| **Health Check** | |
| Name | hc-sps-ers |
| Protocol | TCP |
| Port | 51098 (This must agree with the corresponding LifeKeeper GenLB resource created later) |
| Proxy protocol | NONE |
| Request | Leave empty |
| Response | Leave empty |
| Check interval | 2 seconds |
| Timeout | 2 seconds |
| Healthy threshold | 2 consecutive successes |
| Unhealthy threshold | 3 consecutive failures |
| Connection draining timeout | 10 seconds |
| **Frontend configuration** | |
| Name | fe-sps-ers |
| Subnetwork | lk-subnet |
| Purpose | Non-shared |
| IP address | Ephemeral (Custom) |
| Custom ephemeral IP address | 10.20.0.11 |
| Ports | All |
| Global access | Disable |
| Service label | Leave empty |

Once created, the **Network services** → **Load balancing** page in the Google Cloud Console will show the two load balancers.




3. Implement the startup script workaround described in the **Enabling Load Balancer Traffic from Backend Servers** section of GCP – Using an Internal Load Balancer to allow node-a and node-b to communicate through the frontend IP addresses of the load balancers. Reboot node-a and node-b for the changes to take effect.

4. Following the steps in Responding to Load Balancer Health Checks, install the LifeKeeper Generic Application Recovery Kit for Load Balancer Health Checks (“GenLB Recovery Kit”) and create two LifeKeeper GenLB resources, **ilb-sps-ascs** and **ilb-sps-ers**, with the following properties. Notice that **ilb-sps-ascs** is created on node-a and extended to node-b, while **ilb-sps-ers** is created on node-b and extended to node-a.

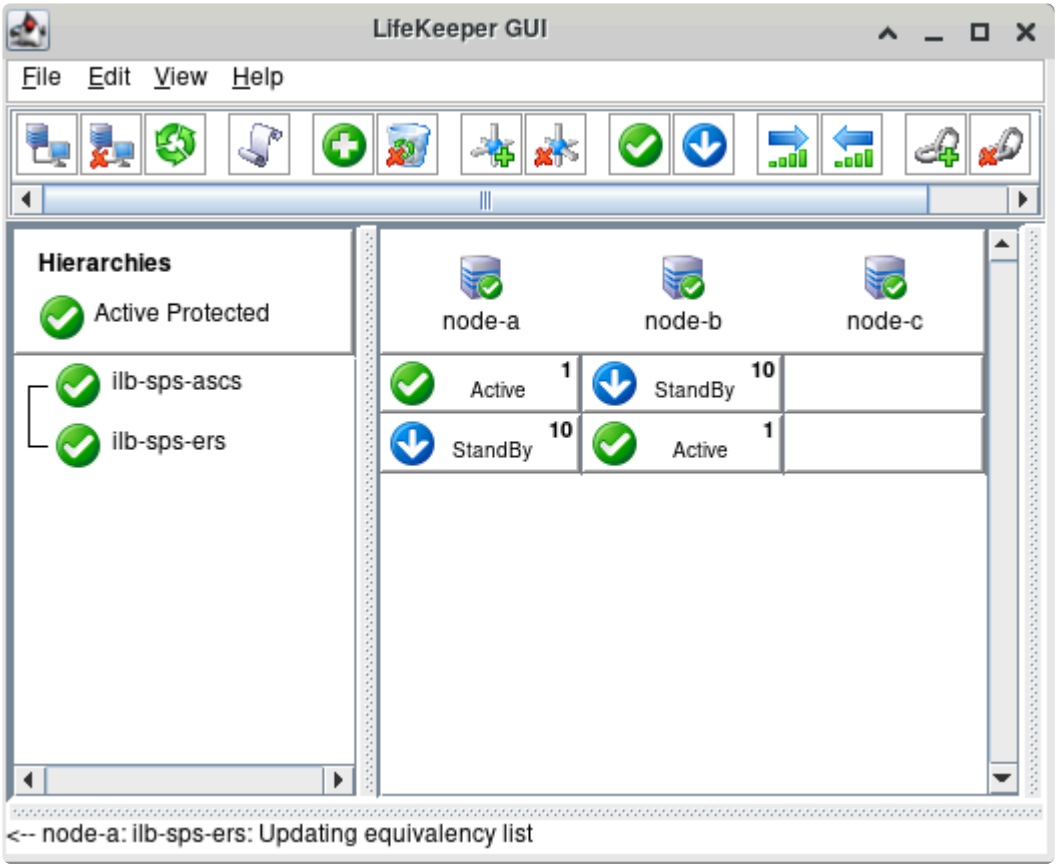
   The ✓ icon indicates that the default option is chosen.

| Field | Value |
|---|---|
| **Create Resource Wizard** | |
| Switchback Type | intelligent ✓ |
| Server | node-a ✓ |
| Restore Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/restore.pl |
| Remove Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/remove.pl |
| QuickCheck Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/quickCheck |
| Local Recovery Script | None (Empty) |
| Application Info | 51098 |
| Bring Resource In Service | Yes |
| Resource Tag | ilb-sps-ascs |
| **Pre-Extend Wizard** | |
| Target Server | node-b ✓ |
| Switchback Type | intelligent ✓ |

| | |
|---|---|
| Template Priority | 1 |
| Target Priority | 10 |
| **Extend gen/app Resource Hierarchy Wizard** | |
| Resource Tag | ilb-sps-ascs |
| Application Info | 51098 |

| Field | Value |
|---|---|
| **Create Resource Wizard** | |
| Switchback Type | intelligent |
| Server | node-b |
| Restore Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/restore.pl |
| Remove Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/remove.pl |
| QuickCheck Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/quickCheck |
| Local Recovery Script | None (Empty) |
| Application Info | 52098 |
| Bring Resource In Service | Yes |
| Resource Tag | ilb-sps-ers |
| **Pre-Extend Wizard** | |
| Target Server | node-a |
| Switchback Type | intelligent |
| Template Priority | 1 |
| Target Priority | 10 |
| **Extend gen/app Resource Hierarchy Wizard** | |
| Resource Tag | ilb-sps-ers |
| Application Info | 52098 |

The resources will appear in the LifeKeeper GUI resource pane once they have been created and extended successfully.




5. Add the following entries to /etc/hosts on node-a and node-b to allow resolution of each virtual hostname to the frontend IP address of the corresponding load balancer:

```
10.20.0.10 sps-ascs
10.20.0.11 sps-ers
```

6. Test switchover and failover of the GenLB resources as described in the **Test GenLB Resource Switchover and Failover** section of Responding to Load Balancer Health Checks. Correct any issues found or tune the parameters of the load balancer health checks as required to achieve successful operation.

# 11.2.7.5.5.2. Create SAP NFS Shared File Systems

In this section we will create highly-available NFS shared file systems which will be mounted on each node hosting an SAP instance. Typical file systems to be shared between systems in a highly-available SAP AS ABAP environment include:

- /sapmnt/<SID>
- /usr/sap/trans
- /usr/sap/<SID>/ASCS<InstNum>
- /usr/sap/<SID>/ERS<InstNum>

See Setting up File Systems for a High-Availability System for more details.

The recommended implementation for creating mount targets for the NFS shares varies by cloud platform. Please follow the steps provided in:

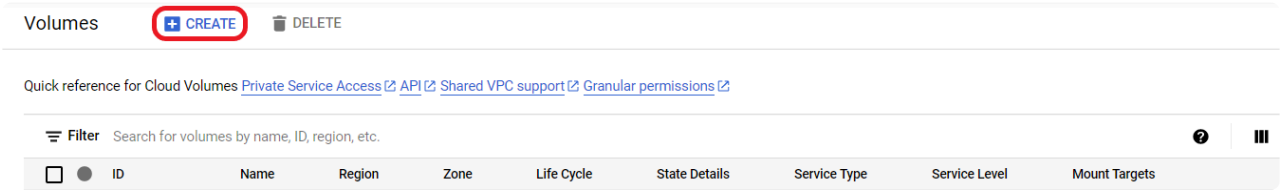
- GCP – Creating SAP NFS Shared File Systems

# 11.2.7.5.5.2.1. GCP – Creating SAP NFS Shared File Systems

In this section we will set up a highly-available NFS mount target using the NetApp Cloud Volumes Service for Google Cloud, which is a cloud-native third-party service that integrates with Google Cloud to provide zonal high-availability for shared file systems. Complete the following steps to create a Cloud Volume, mount it on each cluster node, and prepare the shared file system for installation of the SAP instances.

1. Follow the steps provided in the Quickstart for Cloud Volumes Service to purchase and enable the Cloud Volumes Service in your Google Cloud project.

2. Once the Cloud Volumes Service has been successfully enabled, navigate to the **Cloud Volumes** → **Volumes** page and click **Create** at the top to create a new Cloud Volume.

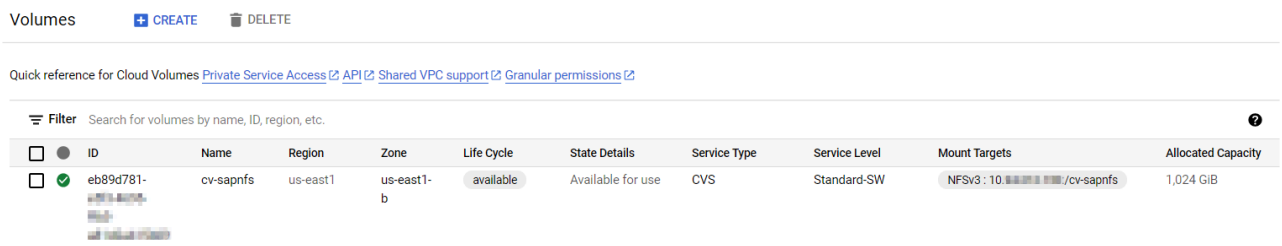



3. Enter the following parameters and click **Save** at the bottom of the page to create the Cloud Volume. The ✓ icon indicates that the default option is chosen.

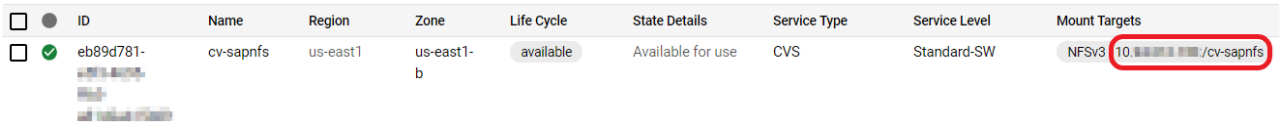
| Field | Value |
|---|---|
| Volume Name | cv-sapnfs |
| Service Type | CVS ✓ |
| Region | <Deployment region> (e.g., us-east1) |
| Zone | <Zone> (e.g., us-east1b) |
| Volume Path | cv-sapnfs |
| Service Level | Standard-SW ✓ |
| Volume Details | 1024 GiB |
| Protocol Type | NFSv3 ✓ |
| Make snapshot directory visible | Unchecked ✓ |
| Shared VPC configuration | Unchecked ✓ |
| Use Custom Address Range | Unchecked ✓ |

| | |
|---|---|
| Export Rule 1 | Allow ‘Read & Write’ on 10.20.0.0/22<br><br>**Note:** We are exposing the NFS export to the entire lk-subnet subnetwork for simplicity, but the scope of the export rules can be made more fine-grained to only allow access from the cluster nodes which are mounting the NFS share. |
| Allow automatic snapshots | Unchecked |

The **cv-sapnfs** Cloud Volume will appear on the **Cloud Volumes** → **Volumes** page once it has been successfully provisioned.




4. Note the **Mount Target** for the NFS export. This will be used to mount the shared file system on the cluster nodes. In the following steps we will use <Mount Target IP> to refer to the IP address of the NFS server listed here.




5. Execute the following commands on both node-a and node-b to create the /sapnfs mount point for the cv-sapnfs NFS share. Add the given mount entry to /etc/fstab on both nodes (replacing <Mount Target IP> with the IP obtained from the Cloud Volumes page in the previous step) so that each cluster node mounts the shared file system at boot.

```
# mkdir /sapnfs
# echo "<Mount Target IP>:/cv-sapnfs        /sapnfs        nfs        nfsver
s=3,proto=tcp,rw,sync,bg        0 0" >> /etc/fstab
# mount -a
```

Verify that the file system has been mounted successfully on both nodes.

```
# df -h | grep sapnfs
<Mount Target IP>:/cv-sapnfs  1.0T  1.2G 1023G   1% /sapnfs
```

**Note:** This step should also be performed on the servers which will host the PAS and AAS instances.

6. Execute the following command on node-a to create the subdirectories within /sapnfs which will contain the shared SAP files.

```
[root@node-a ~]# mkdir -p /sapnfs/{sapmnt/SPS,usr/sap/trans,usr/sap/SPS/{ASCS1
0,ERS20}}
```

Verify that the subdirectories were created successfully.

```
[root@node-a ~]# du -a /sapnfs
0       /sapnfs/sapmnt/SPS
0       /sapnfs/sapmnt
0       /sapnfs/usr/sap/trans
0       /sapnfs/usr/sap/SPS/ASCS10
0       /sapnfs/usr/sap/SPS/ERS20
0       /sapnfs/usr/sap/SPS
0       /sapnfs/usr/sap
0       /sapnfs/usr
0       /sapnfs
```

7. Execute the following commands on both node-a and node-b to create soft links from the SAP file systems to their corresponding subdirectories under /sapnfs.

```
# mkdir -p /sapmnt /usr/sap/SPS
# ln -s /sapnfs/sapmnt/SPS /sapmnt/SPS
# ln -s /sapnfs/usr/sap/SPS/ASCS10 /usr/sap/SPS/ASCS10
# ln -s /sapnfs/usr/sap/SPS/ERS20 /usr/sap/SPS/ERS20
```

Verify that the soft links have been created successfully on each node.

```
# ls -l /sapmnt /usr/sap/SPS
/sapmnt:
total 0
lrwxrwxrwx 1 root root 01 Jan  1 00:00 SPS -> /sapnfs/sapmnt/SPS

/usr/sap/SPS:
total 0
lrwxrwxrwx 1 root root 01 Jan  1 00:00 ASCS10 -> /sapnfs/usr/sap/SPS/ASCS10
lrwxrwxrwx 1 root root 01 Jan  1 00:00 ERS20 -> /sapnfs/usr/sap/SPS/ERS20
```

Execute similar commands on the PAS and AAS instance hosts to create soft links from /sapmnt/SPS and /usr/sap/trans to their corresponding subdirectories under /sapnfs.

The shared file systems are now prepared for installation of the SAP instances.

# 11.2.7.5.5.3. Install SAP Instances

Complete the following steps to install the ASCS10 and ERS20 instances on node-a and node-b.

## Prepare for Installation

1. On node-a and node-b, increase the available swap space to greater than 1MiB to satisfy the SAP system requirements. In this example we will create a 1GiB swap file to increase the available swap space.

```
# mkdir /swap
# dd if=/dev/zero of=/swap/swapfile.img bs=1024 count=1M
# mkswap /swap/swapfile.img
# echo "/swap/swapfile.img        swap        swap        sw        0 0" >> /e
tc/fstab
# swapon /swap/swapfile.img
```

2. Ensure that the tcsh package is installed on both node-a and node-b. Modify this command (e.g., to use `zypper install`) if installing on a SLES server.

```
# yum install -y tcsh
```

3. On node-a and node-b, create a /sap-install directory which will contain the SAP installation files.

```
# mkdir /sap-install
```

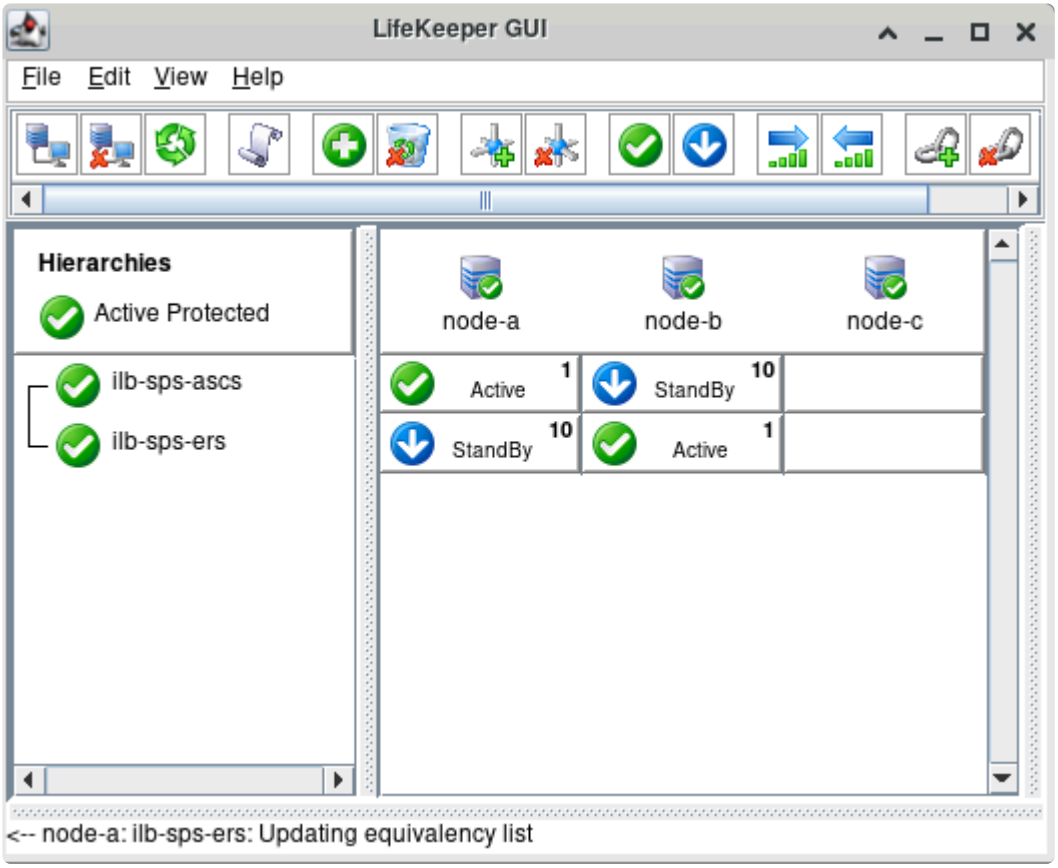
4. Download SAPCAR, SAP Software Provisioning Manager 2.0 (SWPM20SPxx_x-xxxx.SAR), and all necessary packages (e.g., SAPEXE_xx-xxxx.SAR, SAPHOSTAGENTxx_xx-xxxx.SAR, etc.) to the /sap-install directory on both nodes. Make sure that SAPCAR is executable.

```
# chmod +x /sap-install/SAPCAR
```

5. Use SAPCAR to extract the files for SWPM 2.0.

```
# /sap-install/SAPCAR -xvf SWPM20SPxx_x-xxxx.SAR
```

Before installing the SAP instances, verify that all shared file systems described in Create SAP NFS Shared File Systems are mounted on node-a and node-b and that the LifeKeeper resources providing virtual hostname failover for the ASCS and ERS instances are in-service on node-a and node-b, respectively.




## Install the ASCS Instance

Complete the following steps to install the ASCS10 instance on node-a.

1. If installing from a remote system, add a firewall rule to allow inbound traffic on TCP port 4237 (the port which exposes the sapinst graphical interface) on node-a.

2. Execute the following command on node-a to initiate sapinst with the virtual hostname associated to the ASCS instance:

```
[root@node-a ~]# /sap-install/sapinst SAPINST_USE_HOSTNAME=sps-ascs
```

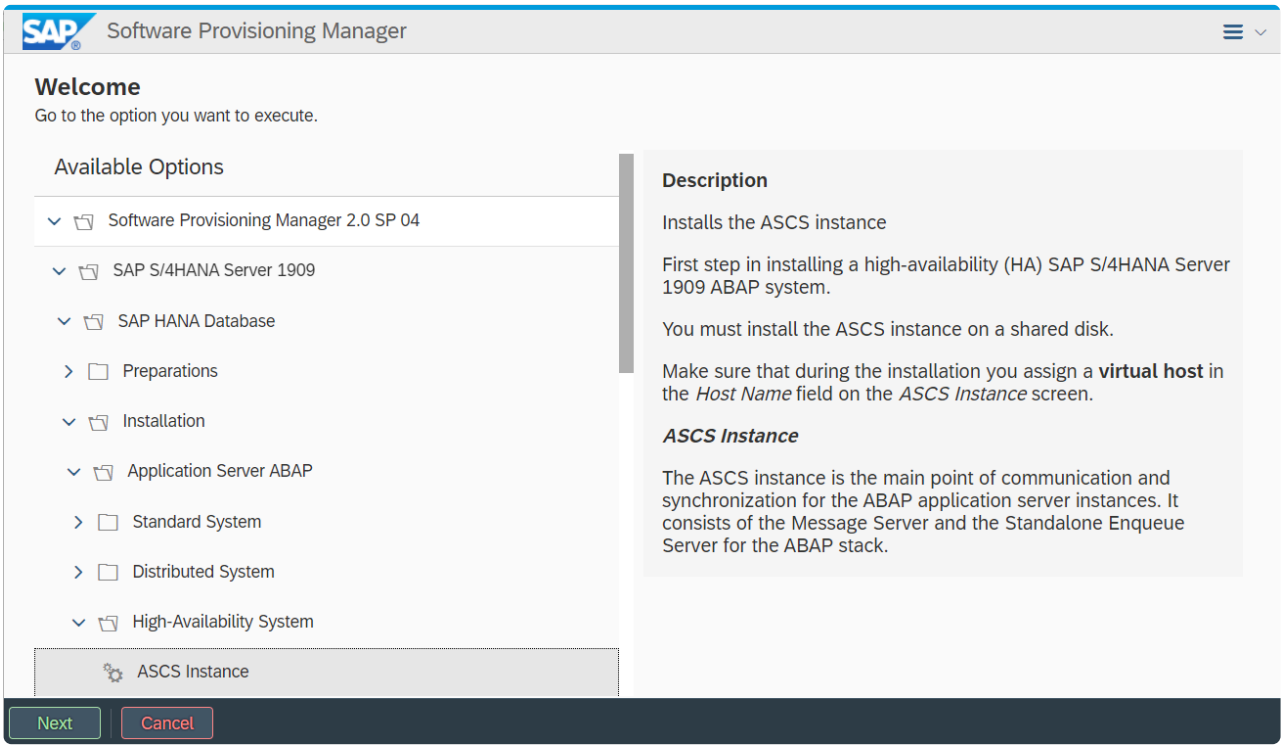
> **Note:** The `SAPINST_REMOTE_ACCESS_USER=<user>` parameter may be used to allow login to the sapinst graphical interface by a non-root user.

> **Note:** If sapinst displays an error due to a missing dt_socket transport library, install the libnsl package with `yum -y install libnsl`.

3. To access the sapinst graphical interface, use a web browser to navigate to https://<node-a Public IP>:4237/sapinst/docs/index.html.

4. Select **Software Provisioning Manager 2.0 SP XX → SAP S/4HANA Server 1909 → SAP HANA**

**Database → Installation → Application Server ABAP → High-Availability System → ASCS Instance** and click **Next**.




5. Provide the following parameters when installing the ASCS instance. The ✓ icon indicates that the default option is chosen.

| Field | Value |
|---|---|
| SAP System ID (SAPSID) | SPS |
| SAP Mount Directory | /sapmnt ✓ |
| Password for All Users | <SAP User Password> |
| Password of SAP System Administrator | <SAP Admin Password> |
| User ID | Leave empty ✓ |
| Group ID of sapsys | Leave empty ✓ |
| Specify path to SAPEXE.SAR | /sap-install |
| Specify path to SAPHOSTAGENT.SAR | /sap-install |
| ASCS Instance Number | 10 |
| ASCS Instance Host Name | sps-ascs ✓ |
| ABAP Message Server Port | 3610 ✓ |
| Internal ABAP Message Server Port | 3910 ✓ |

| | |
|---|---|
| Yes, clean up operating system users | Click |

6. Review the parameters and proceed with the installation.

> **Note:** If the installation fails due to insufficient permissions on the SAP file systems, modify the permissions as described in the next section (**Modify SAP File System Permissions**) and retry the installation.

7. Execute the following command on node-a to clean up SAP installation files:

```
[root@node-a ~]# rm -rf /root/.sapinst
```

## Modify SAP File System Permissions

Execute the following commands on node-a to set the correct permissions on the SAP file systems:

```
[root@node-a ~]# chown -R spsadm /sapnfs/usr/sap/SPS /sapmnt/SPS /usr/sap/SPS
[root@node-a ~]# chgrp -R sapsys /sapnfs /sapmnt /usr/sap
```

## Install the ERS Instance

Complete the following steps to install the ERS20 instance on node-b.

1. If installing from a remote system, add a firewall rule to allow inbound traffic on TCP port 4237 (the port which exposes the sapinst graphical interface) on node-b.

2. Execute the following command on node-b to initiate sapinst with the virtual hostname associated to the ERS instance:

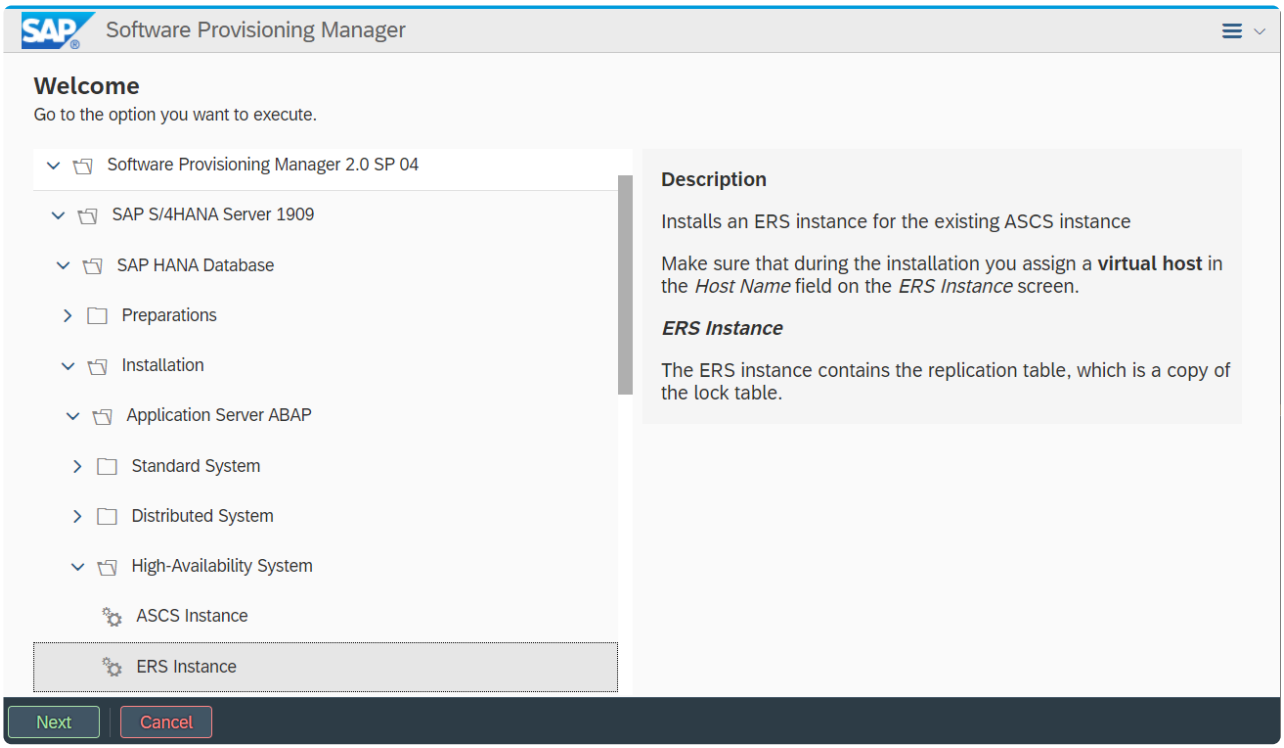
```
[root@node-b ~]# /sap-install/sapinst SAPINST_USE_HOSTNAME=sps-ers
```

> **Note:** The `SAPINST_REMOTE_ACCESS_USER=<user>` parameter may be used to allow login to the sapinst graphical interface by a non-root user.

> **Note:** If sapinst displays an error due to a missing dt_socket transport library, install the libnsl package with `yum install -y libnsl`.

3. To access the sapinst graphical interface, use a web browser to navigate to https://<node-b Public IP>:4237/sapinst/docs/index.html.

4. **Select Software Provisioning Manager 2.0 SP XX → SAP S/4HANA Server 1909 → SAP HANA Database → Installation → Application Server ABAP → High-Availability System → ERS Instance** and click **Next**.




5. Provide the following parameters when installing the ERS instance. The ✓ icon indicates that the default option is chosen.

| Field | Value |
|---|---|
| Profile Directory | /sapmnt/SPS/profile ✓ |
| Password of SAP System Administrator (spsadm) | <spsadm User Password> |
| Specify path to SAPHOSTAGENT.SAR | /sap-install |
| Password of SAP System Administrator (sapadm) | <spsadm User Password> |
| User ID | Leave empty ✓ |
| Group ID of sapsys | Use default value ✓ |
| Name of the ASCS Instance to be Replicated | ASCS10 ✓ |
| Number of the ASCS Instance to be Replicated | 10 ✓ |
| Number of the ERS Instance | 20 ✓ |
| ERS Instance Host | sps-ers ✓ |
| Yes, clean up operating system users | Click |

6. Review the parameters and proceed with the installation.

> **Note:** If the installation fails due to insufficient permissions on the SAP file systems, modify the permissions as described in the **Modify SAP File System Permissions** section and retry the installation.

7. Execute the following command on node-b to clean up SAP installation files:

```
[root@node-b ~]# rm -rf /root/.sapinst
```

## Modify /usr/sap/sapservices on Both Nodes

Since we only installed the ASCS10 instance on node-a and only installed the ERS20 instance on node-b, the /usr/sap/sapservices file is incomplete on each node. Modify this file on both node-a and node-b so that it contains entries for both the ASCS10 and ERS20 instances.

```
# vi /usr/sap/sapservices
# cat /usr/sap/sapservices
#!/bin/sh
LD_LIBRARY_PATH=/usr/sap/SPS/ASCS10/exe:$LD_LIBRARY_PATH; export LD_LIBRARY_PA
TH; /usr/sap/SPS/ASCS10/exe/sapstartsrv pf=/usr/sap/SPS/SYS/profile/SPS_ASCS1
0_sps-ascs -D -u spsadm
LD_LIBRARY_PATH=/usr/sap/SPS/ERS20/exe:$LD_LIBRARY_PATH; export LD_LIBRARY_PAT
H; /usr/sap/SPS/ERS20/exe/sapstartsrv pf=/usr/sap/SPS/SYS/profile/SPS_ERS20_sp
s-ers -D -u spsadm
```

## Restart the ASCS Instance to Enable Enqueue Replication

Now that the ERS instance has been installed on node-b, the ASCS instance and its corresponding SAP Start Service process must be restarted on node-a in order to enable enqueue replication. Execute the following commands on node-a to restart the ASCS instance and verify that enqueue replication has been enabled.

1. 
```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 10 -function RestartSer
vice SPS"
15.03.2021 00:29:23
RestartService
OK
```

2. 
```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 10 -function StopWait 6
0 2"
15.03.2021 00:29:44
```

```
Stop
OK
```

3. 
```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 10 -function StartWait
60 2"
15.03.2021 00:30:07
Start
OK
```

4. 
```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 10 -function EnqGetStat
istic | grep replication_state"
replication_state: GREEN
```

## Modify /etc/default/LifeKeeper on Both Nodes

Add the following entries to /etc/default/LifeKeeper on both node-a and node-b:

```
NFS_RPC_PROTOCOL=tcp
SAP_NFS_CHECK_DIRS=/sapnfs
```

The `SAP_NFS_CHECK_DIRS` entry allows LifeKeeper to check the availability of the Cloud Volume mount target exporting the **cv-sapnfs** NFS share before performing SAP-related operations, and the `NFS_RPC_PROTOCOL` entry tells LifeKeeper to use the TCP protocol when checking availability of the NFS server.

## Verify Successful ASCS and ERS Installation

Execute the following commands to verify that the ASCS10 and ERS20 instances are running successfully on node-a and node-b, respectively.

1. 
```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 10 -function GetProcess
List"
01.01.2021 00:00:00
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
msg_server, MessageServer, GREEN, Running, 2021 01 01 00:00:00, 0:00:0
0, 8420
enq_server, Enqueue Server 2, GREEN, Running, 2021 01 01 00:00:00, 0:0
0:00, 8428
```

2. 
```
[root@node-b ~]# su - spsadm -c "sapcontrol -nr 20 -function GetProcess
```

```
List"
01.01.2021 00:00:00
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
enq_replicator, Enqueue Replicator 2, GREEN, Running, 2021 01 01 00:0
0:00, 0:00:00, 25263
```

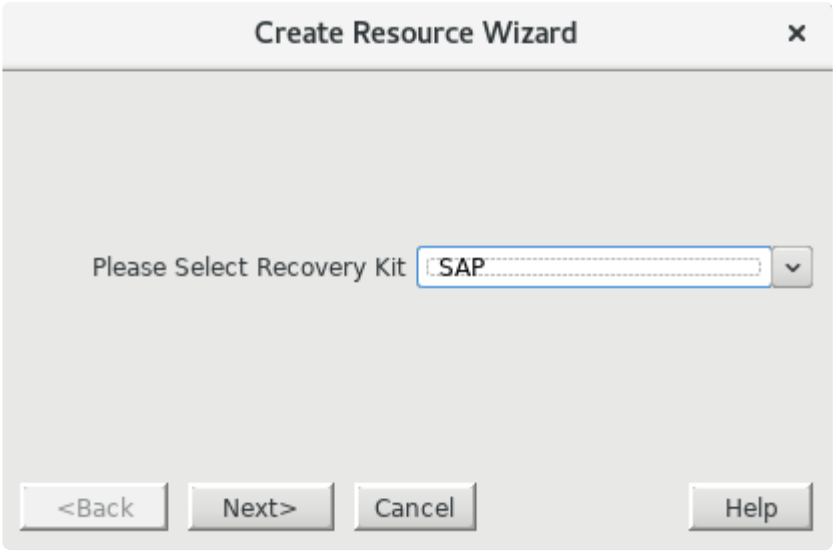
## Install PAS and AAS Instances

Once the database (e.g., SAP HANA) has been installed and configured for high-availability, use SWPM 2.0 to install one Primary Application Server (PAS) and one Additional Application Server (AAS) instance on two external application server nodes spread across two availability zones. As application server redundancy is provided by the existence of the AAS instance, the application server instances do not require LifeKeeper protection. Note that the previously created NFS shares containing the /sapmnt/SPS and /usr/sap/trans file systems must be mounted on both the PAS and AAS instance hosts before installation. Please consult SAP documentation for details on the installation of the PAS and AAS instances.

# 11.2.7.5.5.4. Create LifeKeeper SAP Resources

## Create the ASCS Resource Hierarchy

1. In the LifeKeeper GUI, click the icon to open the **Create Resource Wizard**. Select the “SAP” Recovery Kit.

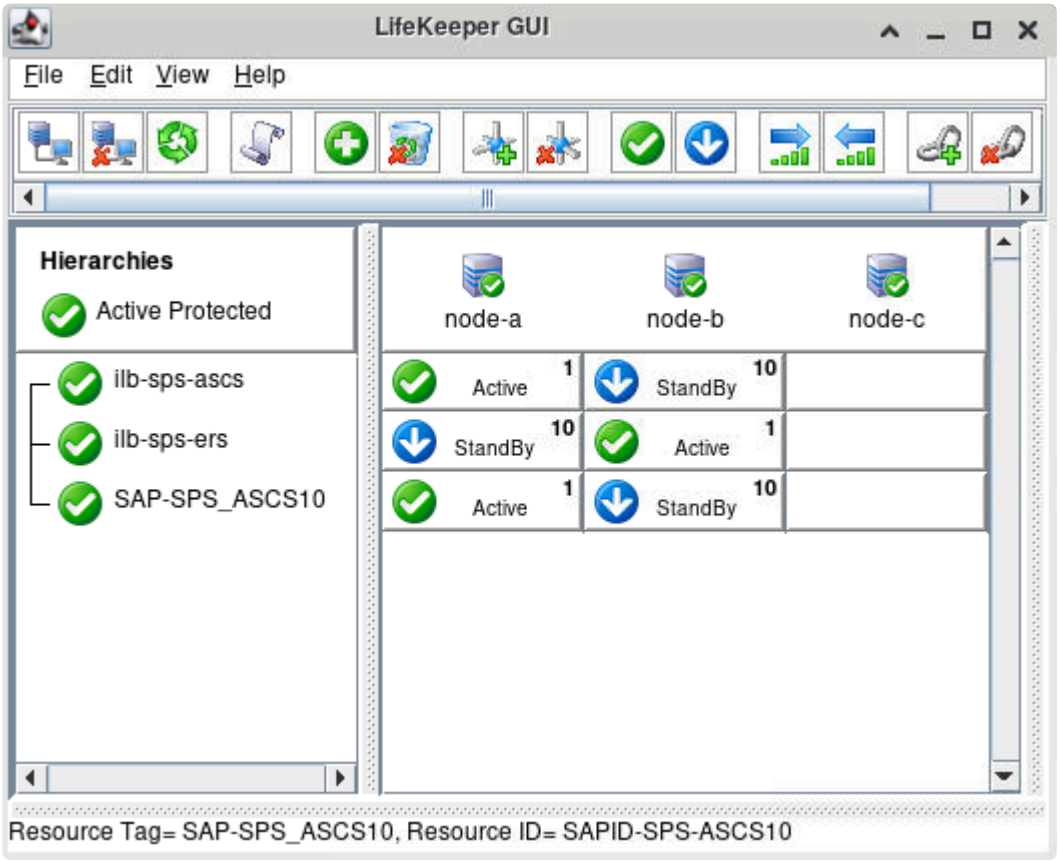



2. Enter the following values to create and extend a LifeKeeper resource (**SAP-SPS_ASCS10**) to protect the ASCS10 instance on node-a and node-b. Notice that the resource is being created on node-a and extended to node-b. Also note that the resulting resource should not be extended to node-c, the witness node. The icon indicates that the default option is chosen.

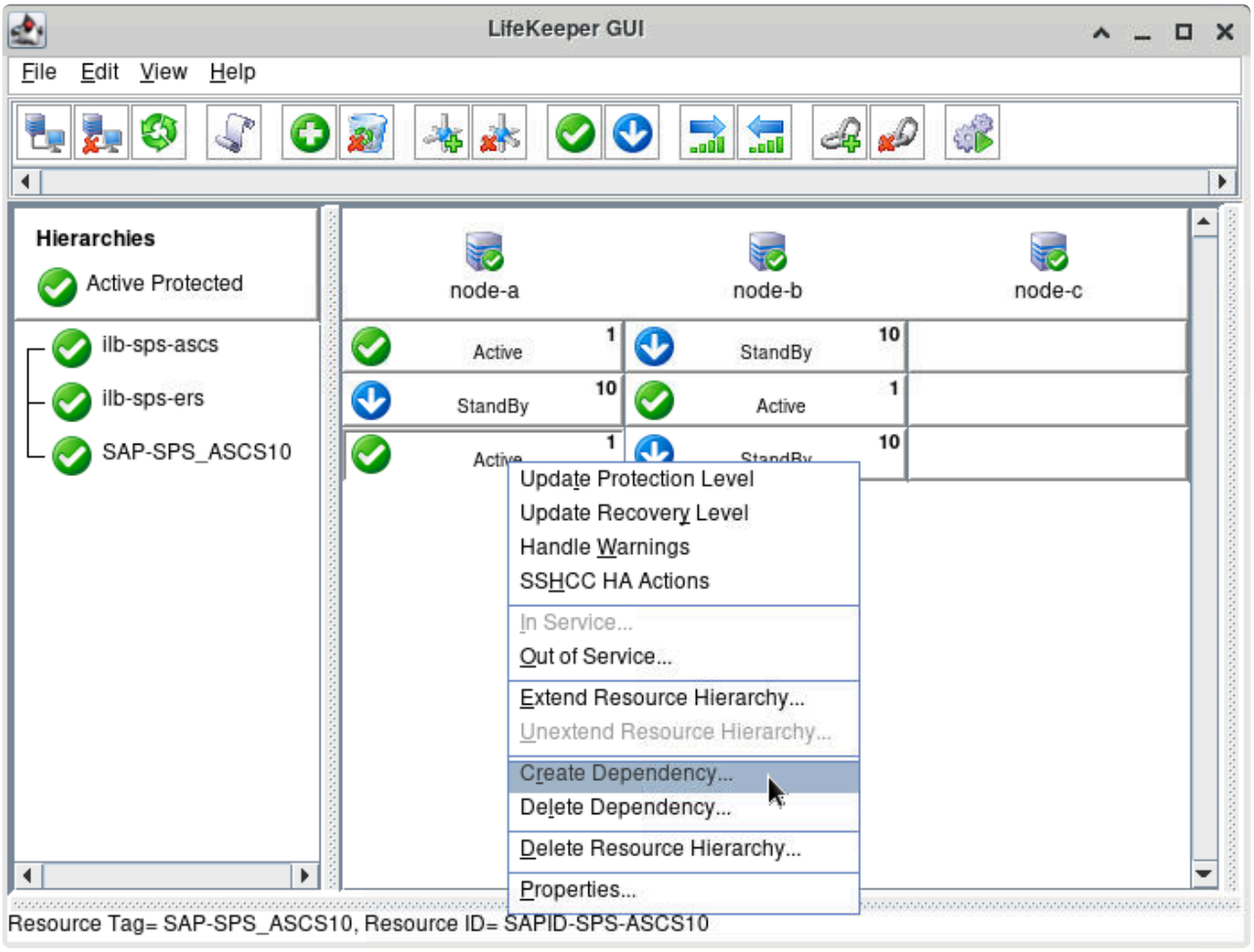
| Field | Value |
|---|---|
| **Create Resource Wizard** | |
| Switchback Type | intelligent |
| Server | node-a |
| SAP SID | SPS |
| SAP Instance for SPS | ASCS10 |
| IP child resource | none |
| Automate dependent filesystem creation | no |
| Dependent filesystem resource | none |

| | |
|---|---|
| SAP Tag | SAP-SPS_ASCS10 |
| **Pre-Extend Wizard** | |
| Target Server | node-b |
| Switchback Type | intelligent |
| Template Priority | 1 |
| Target Priority | 10 |
| **Extend gen/app Resource Hierarchy Wizard** | |
| Root Tag | SAP-SPS_ASCS10 |

Once the **SAP-SPS_ASCS10** resource has been successfully created and extended to node-b, the LifeKeeper resource panel should look similar to the following. Note that the names of the resources providing failover functionality for the ASCS and ERS virtual IP's/hostnames may differ according to your chosen cloud platform depending on which subsection of Create ASCS and ERS Virtual IPs was followed.

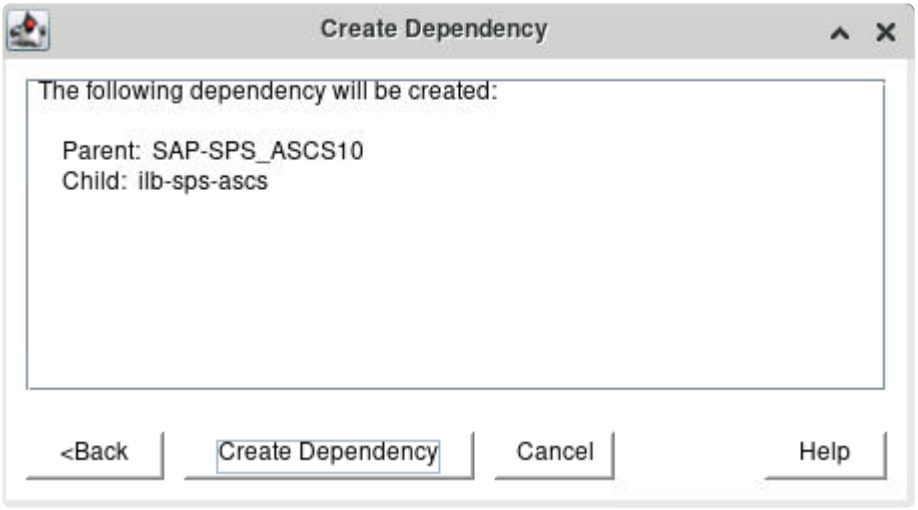



3. Right-click on the **SAP-SPS_ASCS10** resource and select **Create Dependency…** from the drop-down menu.

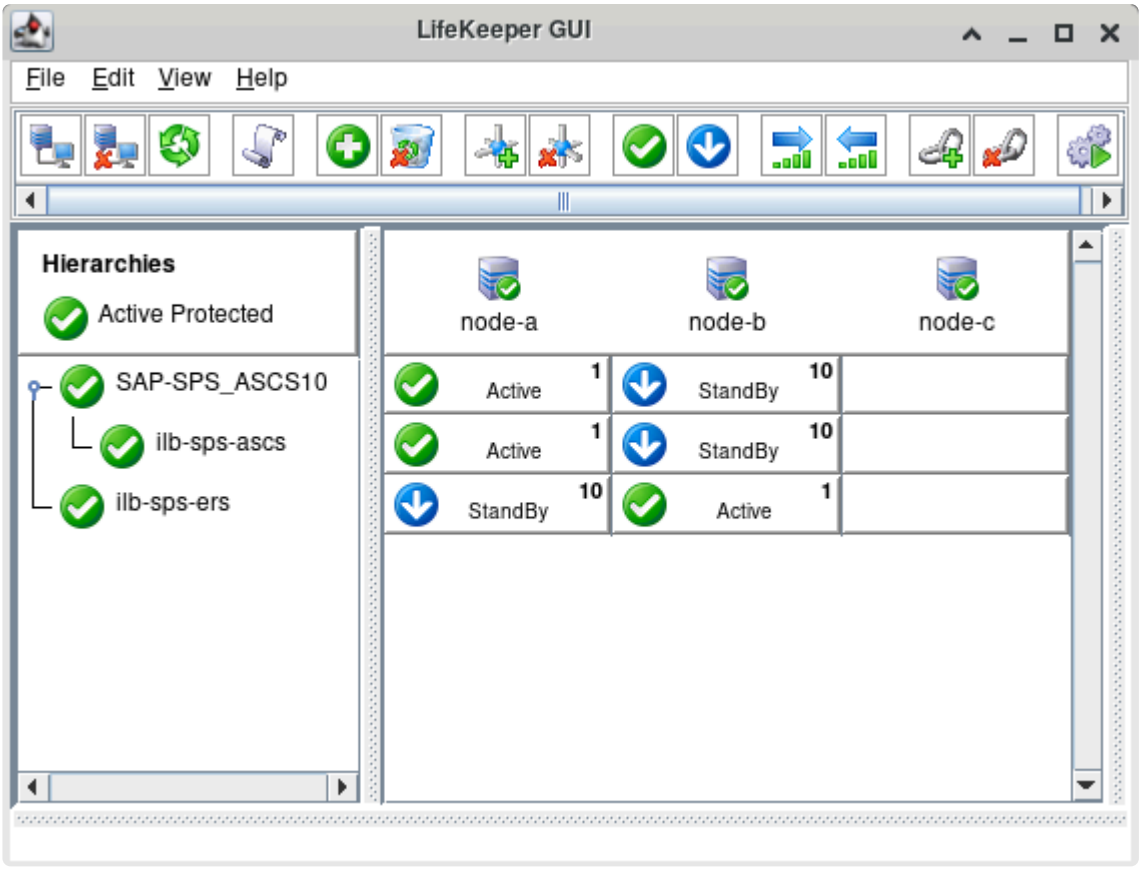



For **Child Resource Tag**, specify the root resource of the resource hierarchy being used to failover the ASCS virtual IP/hostname. The precise resource name used in this step will depend on resources created in the subsection of Create ASCS and ERS Virtual IPs corresponding to your chosen cloud platform.

Click **Next>** to continue, then click **Create Dependency** to create the dependency.

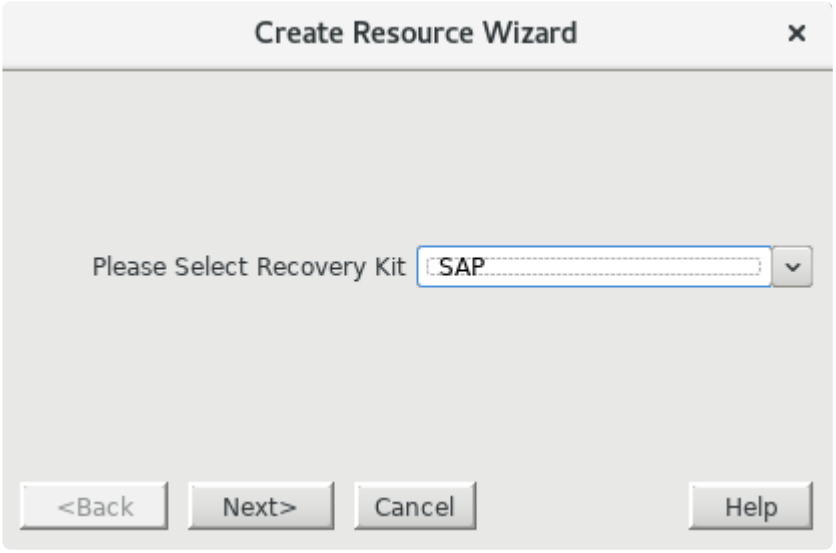



The resulting hierarchy will look similar to the following:




## Create the ERS Resource Hierarchy

1. In the LifeKeeper GUI, click the icon to open the **Create Resource Wizard**. Select the “SAP” Recovery Kit.

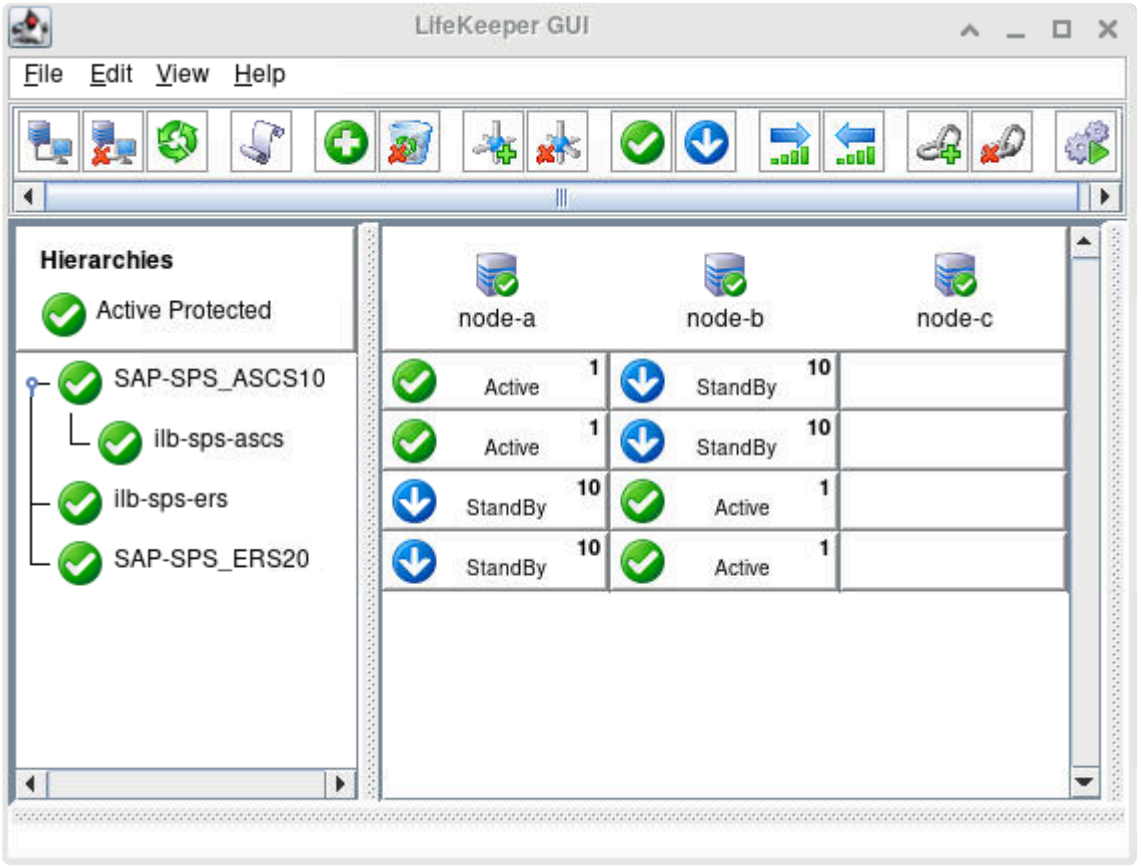



2. Enter the following values to create and extend a LifeKeeper resource (**SAP-SPS_ERS20**) to protect the ERS20 instance on node-a and node-b. Notice that this resource is being created on node-b and extended to node-a. Also note that the resulting resource should not be extended to node-c, the witness node. The ✓ icon indicates that the default option is chosen.

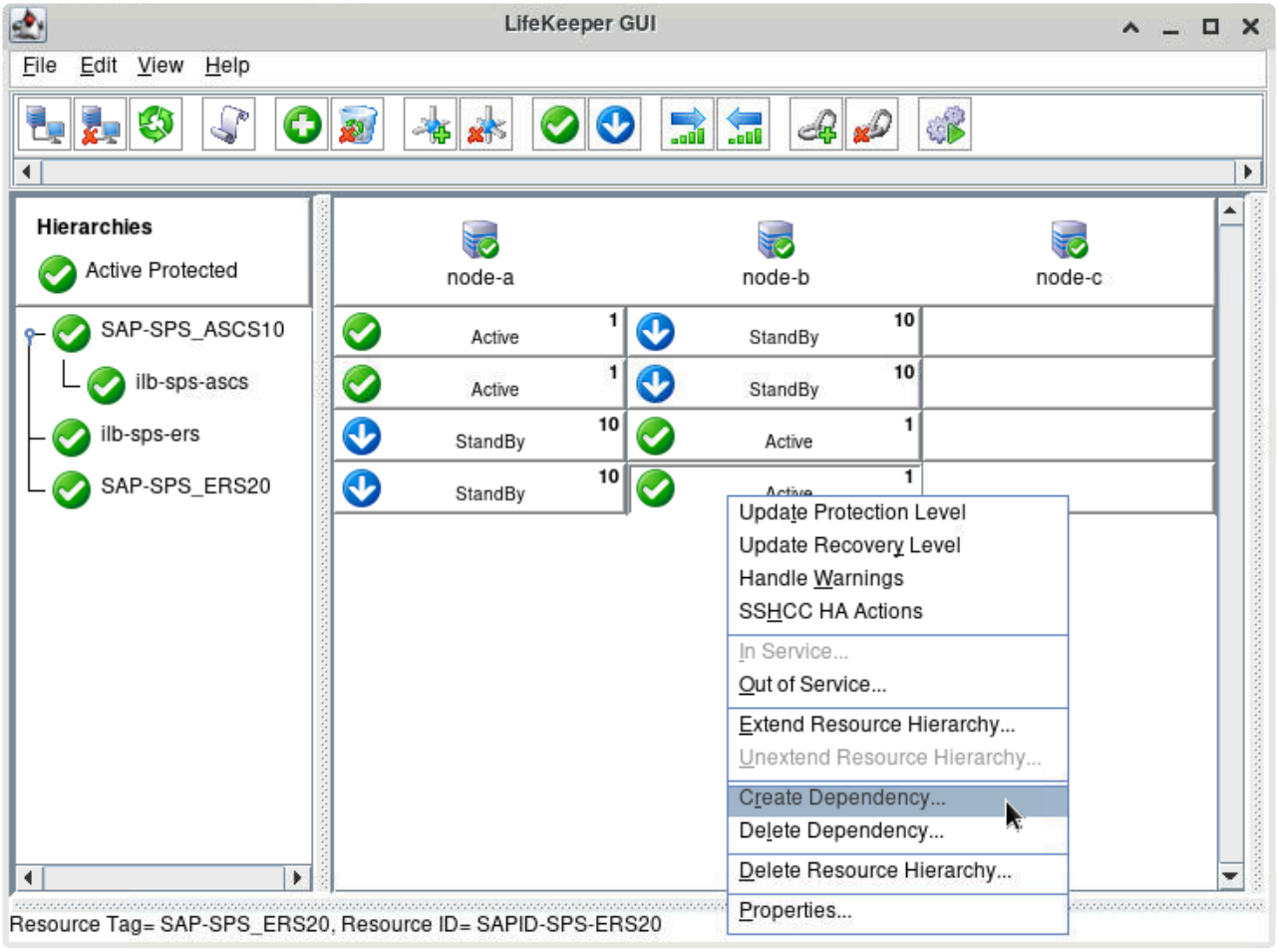
| Field | Value |
|---|---|
| **Create Resource Wizard** | |
| Switchback Type | intelligent ✓ |
| Server | node-b |
| SAP SID | SPS ✓ |
| SAP Instance for SPS | ERS20 ✓ |
| IP child resource | none ✓ |
| Automate dependent filesystem creation | no ✓ |
| Dependent filesystem resource | none ✓ |
| SAP Tag | SAP-SPS_ERS20 ✓ |
| **Pre-Extend Wizard** | |
| Target Server | node-a ✓ |
| Switchback Type | intelligent ✓ |
| Template Priority | 1 ✓ |
| Target Priority | 10 ✓ |

| Extend gen/app Resource Hierarchy Wizard | |
|---|---|
| Root Tag | SAP-SPS_ERS20 |

Once the **SAP-SPS_ERS20** resource has been successfully created and extended to node-a, the LifeKeeper resource panel should look similar to the following. Note that the names of the resources providing failover functionality for the ASCS and ERS virtual IP's/hostnames may differ according to your chosen cloud platform depending on which subsection of Create ASCS and ERS Virtual IPs was followed.

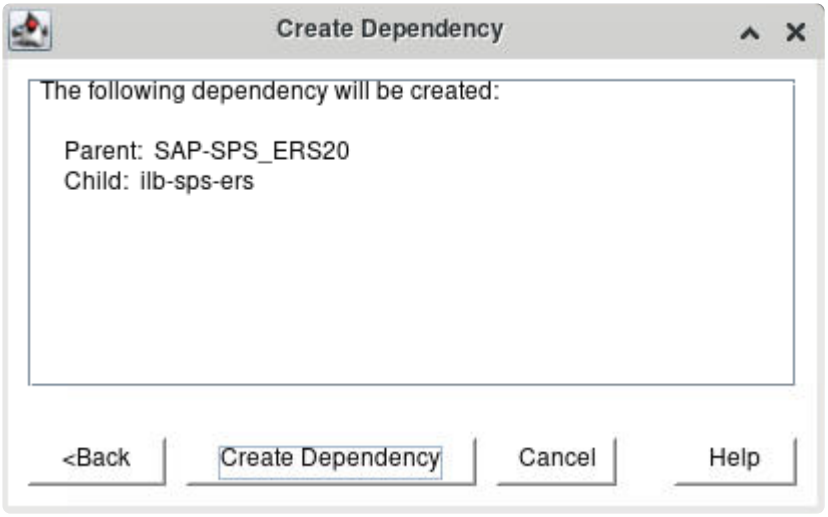



3. Right-click on the **SAP-SPS_ERS20** resource and select **Create Dependency…** from the drop-down menu.

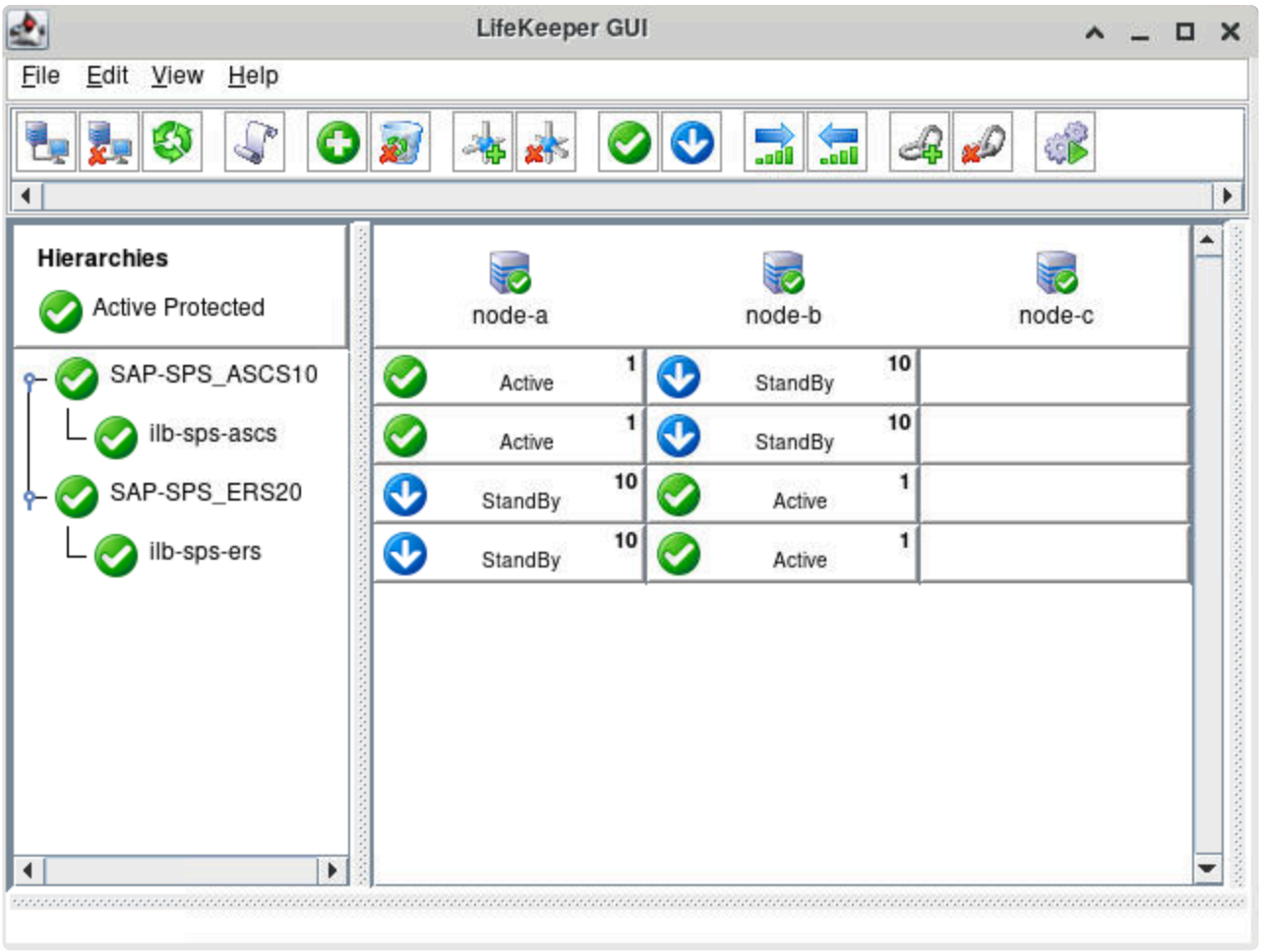



For **Child Resource Tag**, specify the root resource of the resource hierarchy being used to failover the ERS virtual IP/hostname. The precise resource name used in this step will depend on resources created in the subsection of Create ASCS and ERS Virtual IPs corresponding to your chosen cloud platform.

Click **Next>** to continue, then click **Create Dependency** to create the dependency.




The resulting hierarchy will look similar to the following:

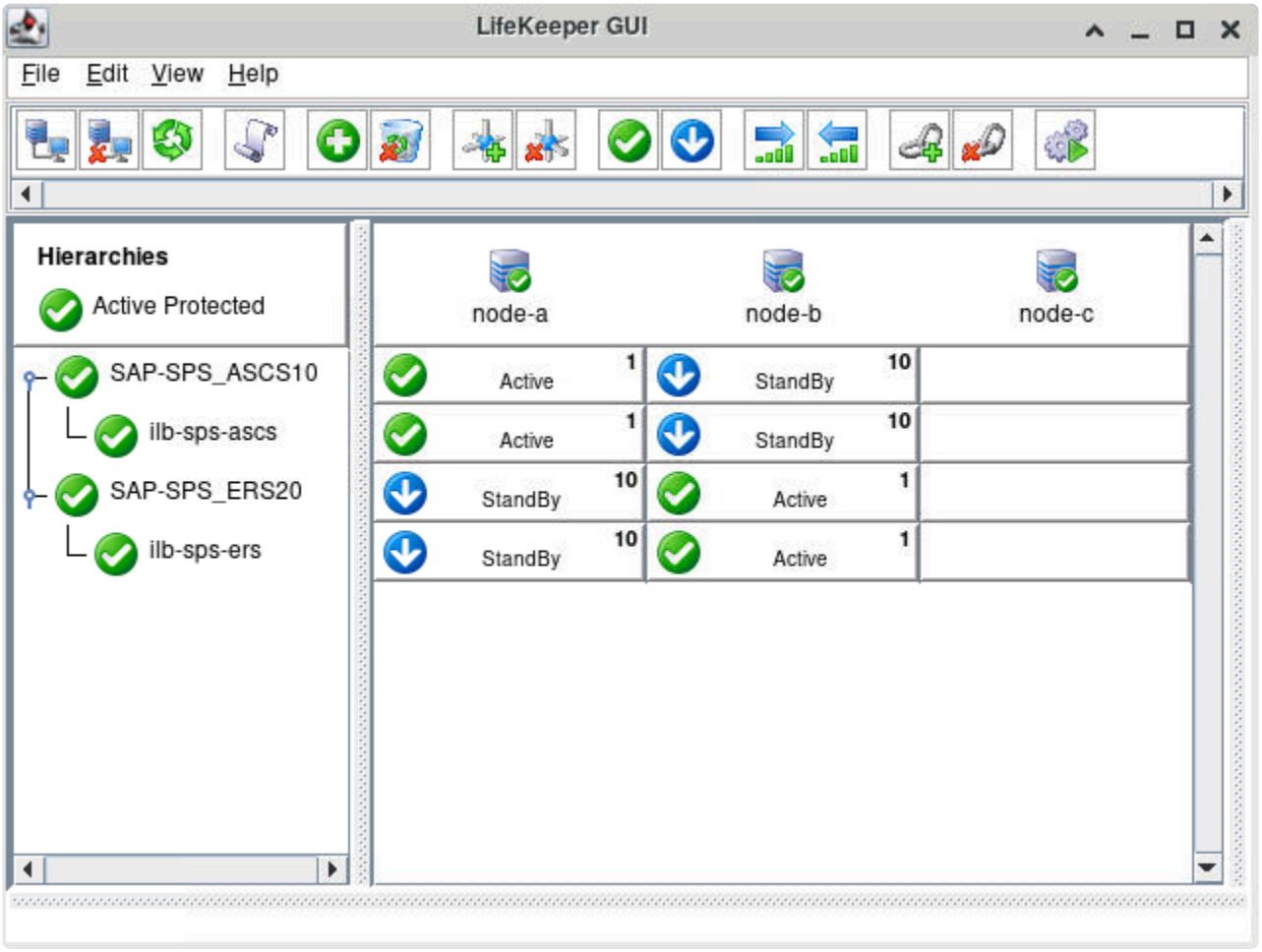



The ASCS10 and ERS20 instances have now been successfully protected by LifeKeeper.

# 11.2.7.5.5.5. Test Switchover and Failover

In this section we will perform basic tests to verify the expected behavior of the **SAP-SPS_ASCS10** and **SAP-SPS_ERS20** resource hierarchies on switchover and failover. It is important to test that the enqueue server process in the ASCS10 instance is able to successfully recover the enqueue lock table from the ERS20 instance after switchover or failover.

1. Verify that the **SAP-SPS_ASCS10** resource state is currently Active on node-a and Standby on node-b, and that the **SAP-SPS_ERS20** resource state is currently Active on node-b and Standby on node-a.




2. Execute the following commands to verify that the ASCS10 and ERS20 instances are running successfully on node-a and node-b, respectively:

```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 10 -function GetProcessList"
04.03.2021 20:24:12
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
msg_server, MessageServer, GREEN, Running, 2020 12 21 16:53:00, 1755:31:12, 11
497
enq_server, Enqueue Server 2, GREEN, Running, 2020 12 21 16:53:00, 1755:31:1
```

```
2, 11498

[root@node-b ~]# su - spsadm -c "sapcontrol -nr 20 -function GetProcessList"
04.03.2021 20:24:22
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
enq_replicator, Enqueue Replicator 2, GREEN, Running, 2021 02 22 16:55:17, 24
3:29:05, 30028
```

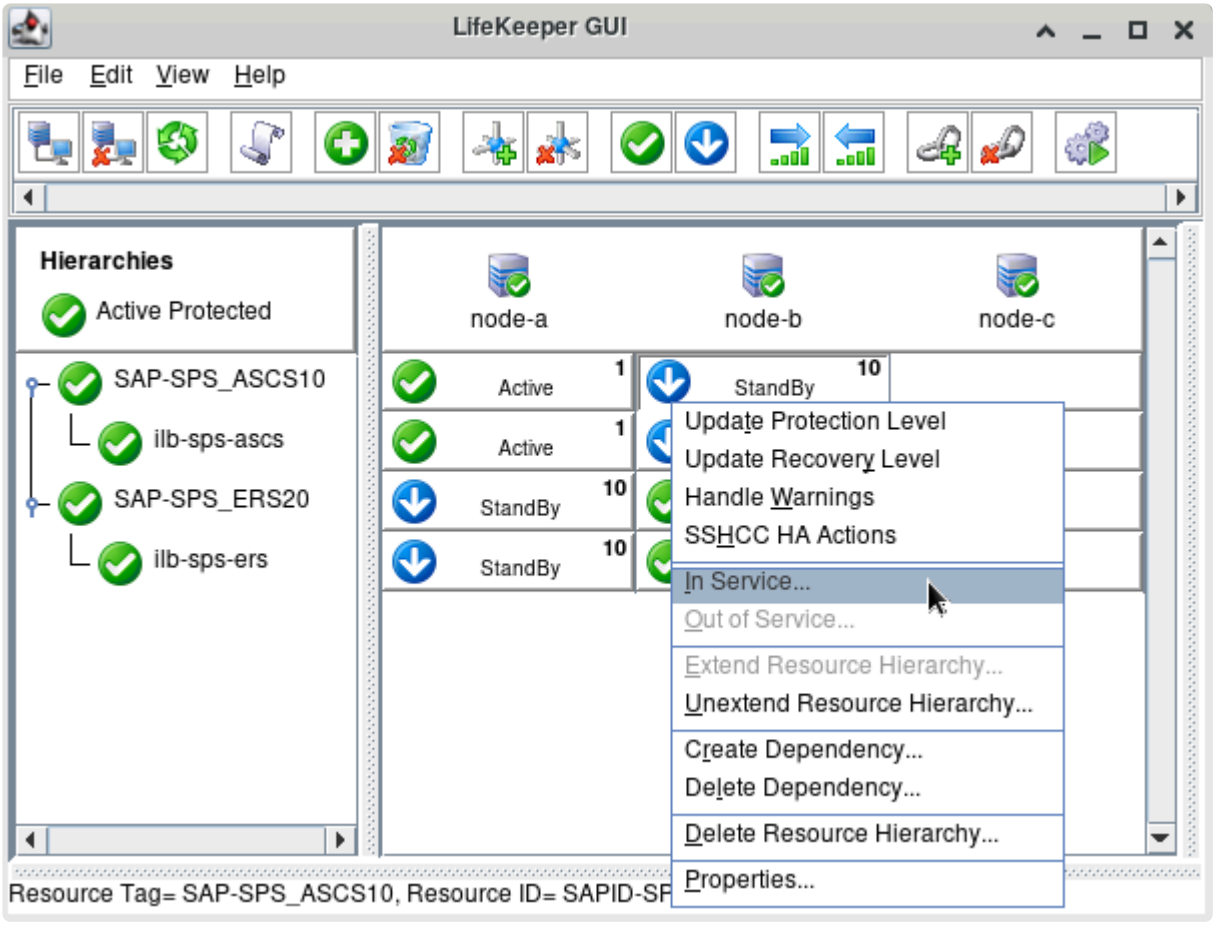
3. Execute the following command on node-a to write 100 exclusive non-cumulative locks labeled 0-99 to the lock table maintained by the enqueue server.

```
[root@node-a ~]# su - spsadm -c "enq_admin --set_locks=100:X:DIAG::TAB:%u p
f=/usr/sap/SPS/SYS/profile/SPS_ASCS10_sps-ascs"
Enqueue Server 2

2021-03-04 20:32:16; OK; 'Set Locks'; Response=41496 usec
============================================================
```

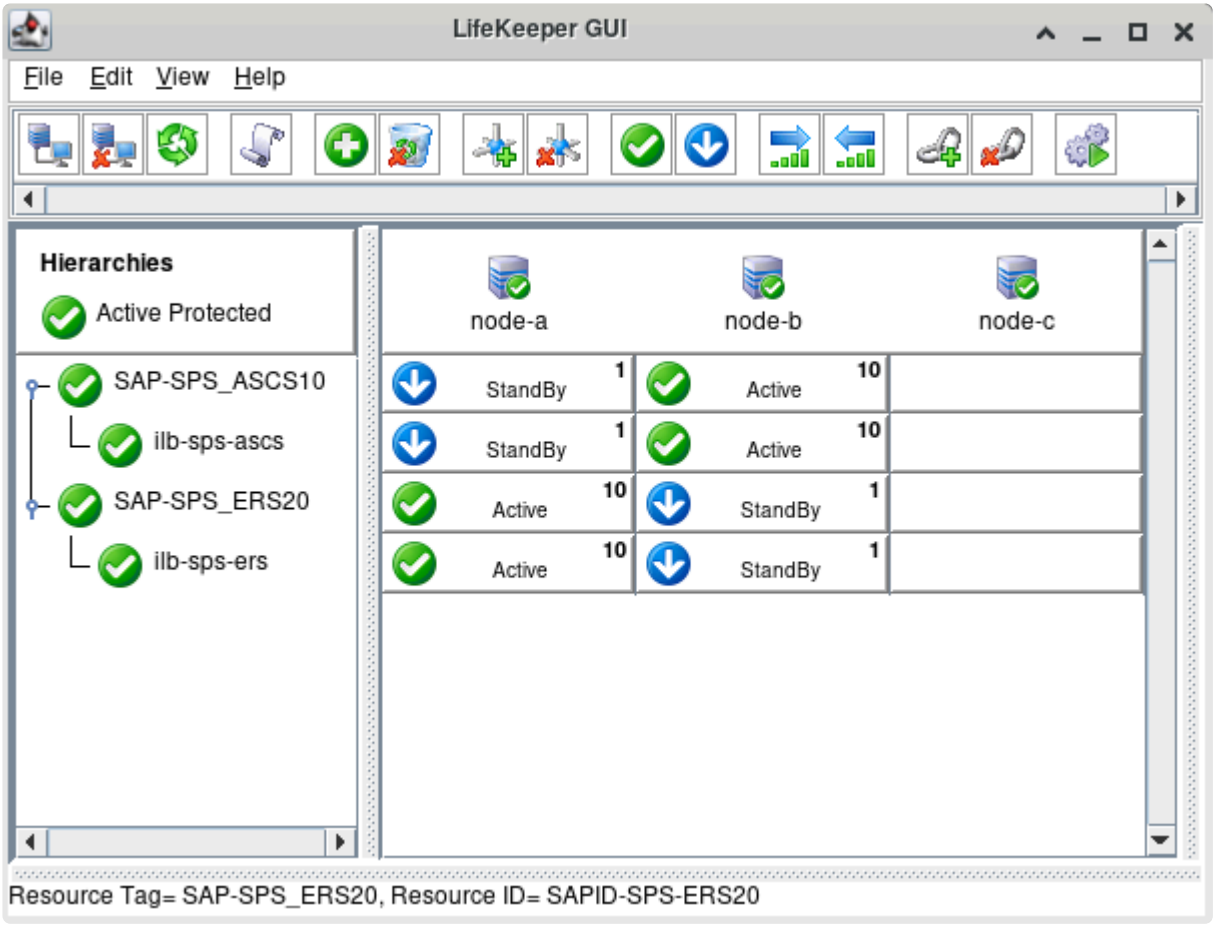
Execute the following commands to verify that the locks have been successfully stored in the lock table on node-a and replicated to the enqueue replication server on node-b:

```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 10 -function EnqGetStatistic"
| grep locks_now
locks_now: 100

[root@node-b ~]# su - spsadm -c "sapcontrol -nr 20 -function EnqGetStatistic"
| grep locks_now
locks_now: 100
```

4. Perform a switchover of the ASCS resource hierarchy by right-clicking the **SAP-SPS_ASCS10** resource on node-b and choosing the **In-Service…** operation. Click **In Service** to begin the switchover.




Once the ASCS resource hierarchy has successfully come in-service on node-b and the enqueue server process has obtained the copy of the backup enqueue lock table from the enqueue replication server process, LifeKeeper will automatically relocate the **SAP-SPS_ERS20** resource to node-a to provide lock table redundancy across cluster nodes.




5. Once LifeKeeper has relocated the SAP-SPS_ERS20 resource back to node-a, execute the following commands to verify that the ASCS10 and ERS20 instances are running successfully on node-b and node-a, respectively, and that they both still hold the 100 locks written in step 3.

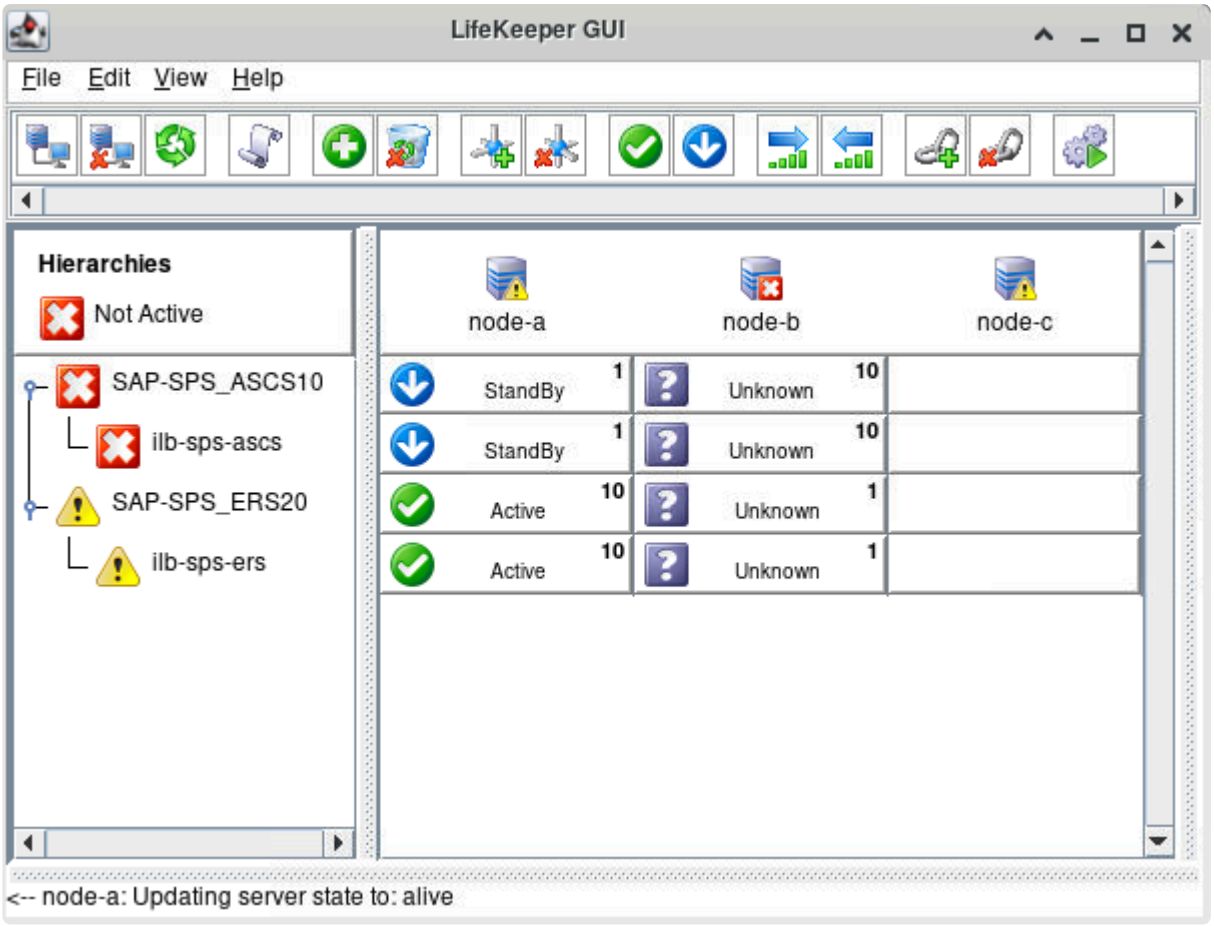
```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 20 -function GetProcessList"
04.03.2021 20:58:57
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
enq_replicator, Enqueue Replicator 2, GREEN, Running, 2021 03 04 20:57:34, 0:0
1:23, 21967

[root@node-a ~]# su - spsadm -c "sapcontrol -nr 20 -function EnqGetStatistic"
| grep locks_now
locks_now: 100

[root@node-b ~]# su - spsadm -c "sapcontrol -nr 10 -function GetProcessList"
04.03.2021 20:56:56
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
msg_server, MessageServer, GREEN, Running, 2021 03 04 20:54:47, 0:02:09, 17074
enq_server, Enqueue Server 2, GREEN, Running, 2021 03 04 20:54:47, 0:02:09, 17
075
```

```
[root@node-b ~]# su - spsadm -c "sapcontrol -nr 10 -function EnqGetStatistic"
| grep locks_now
locks_now: 100
```

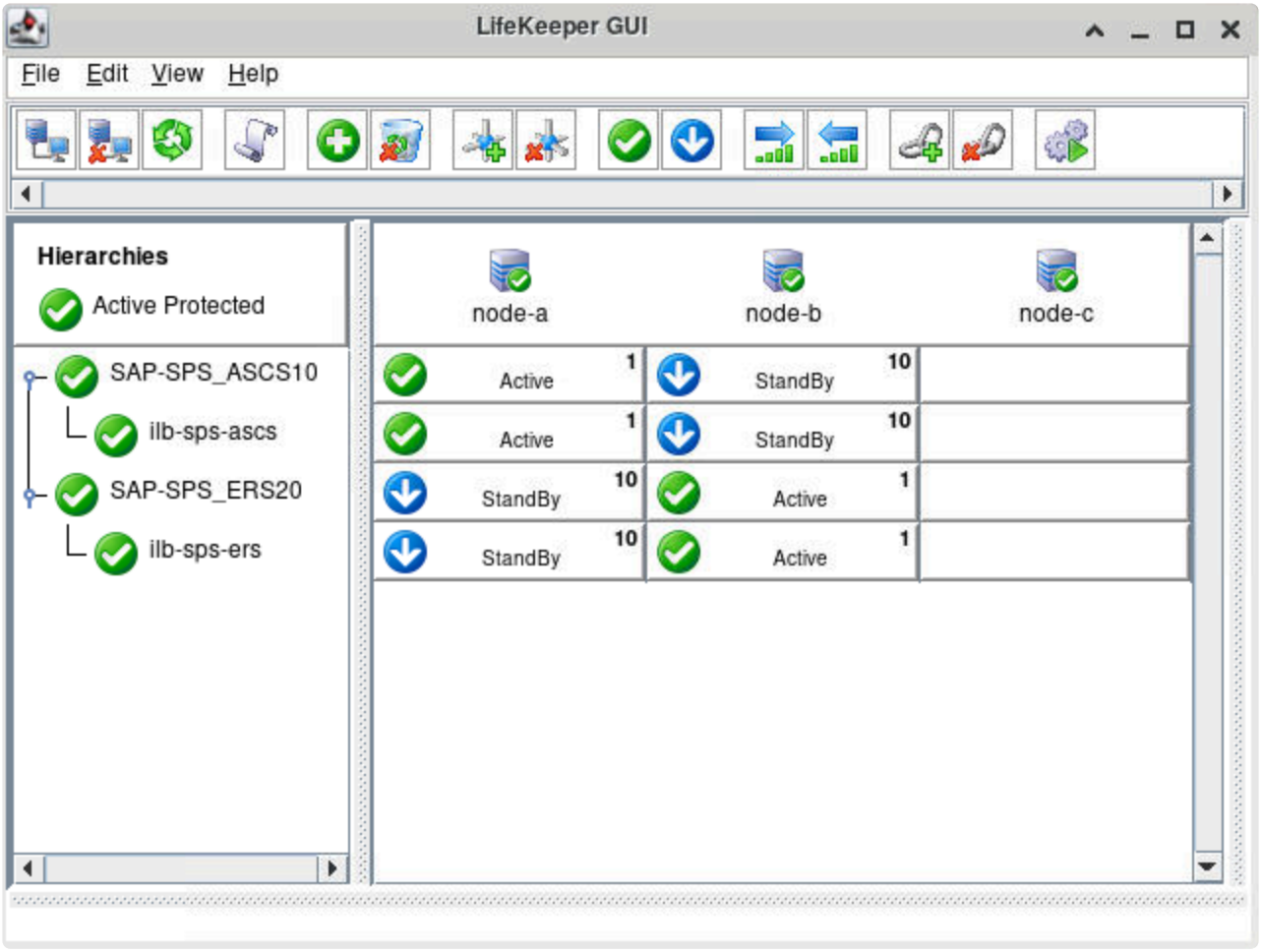
6. Execute the following command to forcefully reboot node-b:

```
[root@node-b ~]# echo b > /proc/sysrq-trigger
```

Once LifeKeeper has detected that node-b has been powered off, the status of node-b updates to “Unknown” in the LifeKeeper GUI and LifeKeeper initiates failover of the **SAP-SPS_ASCS10** resource hierarchy back to node-a.




Once node-b is back online, LifeKeeper will automatically relocate the **SAP-SPS_ERS20** resource hierarchy back to node-b. Once this process is complete, the **SAP-SPS_ASCS10** and **SAP-SPS_ERS20** resource hierarchies will be back in-service on node-a and node-b, respectively.




7. Execute the sapcontrol commands given in steps 2 and 3 again to verify the expected state on each node.

8. Execute the following command on node-a to release the 100 locks that were written in step 3:

```
[root@node-a ~]# su - spsadm -c "enq_admin --release_locks=100:X:DIAG::TAB:%u
pf=/usr/sap/SPS/SYS/profile/SPS_ASCS10_sps-ascs"
Enqueue Server 2

2021-03-04 21:10:22; OK; 'Release Locks'; Response=36883 usec
============================================================
```

We have now verified the basic switchover and failover functionality of the ASCS and ERS resource hierarchies.

# 11.2.7.5.6. Protecting SAP HANA Resources

In this section we will deploy a highly-available SAP HANA 2.0 SPS04 cluster. Our configuration will use SAP HANA System Replication to replicate data from the active database host to the standby database host and will use the LifeKeeper SAP HANA Recovery Kit to handle monitoring, recovery, and failover the HDB database instance.

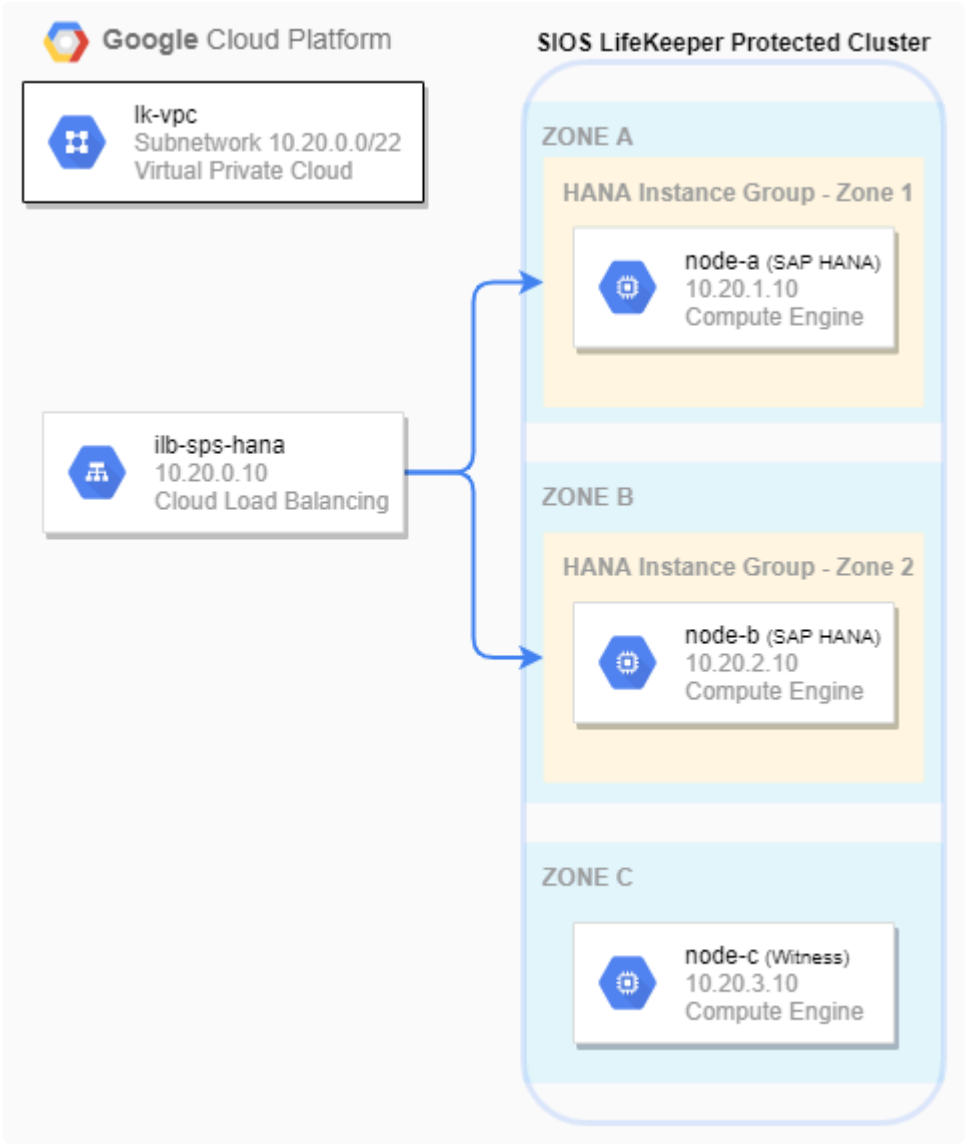
The details of the SAP HANA cluster nodes that will be created are given below. The instance provisioning described in this guide gives the **minimum** required specifications to run the applications in an evaluation context, and is not generally sufficient to handle productive workloads. Consult the documentation provided by SAP and your cloud provider for best practices when provisioning VM's in a production environment. For example, see Hardware Requirements for SAP System Hosts.

- node-a (Primary database host)
  - Private IP: 10.20.1.10
  - vCPU's: 8
  - Memory: 32GB
  - SAP HANA 2.0 SPS04 database instance HDB00 installed under SID 'SPS'
  - SAP HANA System Replication parameters:
    - Site name: SiteA
    - Replication mode: primary
    - Operation mode: primary
  - SIOS Protection Suite for Linux installed with quorum/witness functionality enabled, SAP HANA Recovery Kit installed
  - [GCP] GenLB Recovery Kit installed

- node-b (Standby database host)
  - Private IP: 10.20.2.10
  - vCPU's: 8
  - Memory: 32GB
  - SAP HANA 2.0 SPS04 database instance HDB00 installed under SID 'SPS'
  - SAP HANA System Replication parameters:
    - Site name: SiteB
    - Replication mode: sync
    - Operation mode: logreplay
  - SIOS Protection Suite for Linux installed with quorum/witness functionality enabled, SAP HANA Recovery Kit installed
  - [GCP] GenLB Recovery Kit installed

- node-c (Quorum/Witness node)
  - Private IP: 10.20.3.10
  - vCPU's: 1
  - Memory: 2GB
  - SIOS Protection Suite for Linux installed with quorum/witness functionality enabled

- SAP HANA virtual hostname: sps-hana




This section assumes that the following prerequisite steps have already been completed:

1. Three VM instances (node-a, node-b, and node-c) have been created using the networking conventions described earlier in the guide. Firewall rules are in place to allow inter-node communication as well as SSH connections. See Configuring Network Components and Creating Instances for details.

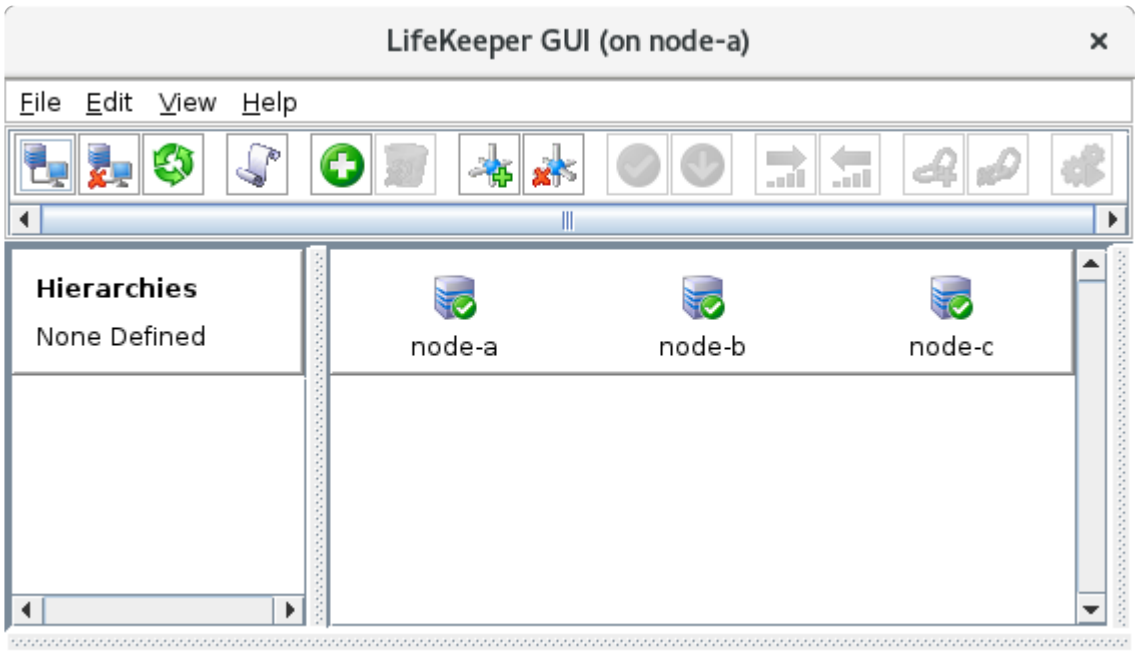
> ! If deploying in Google Cloud, the “RedHat Enterprise Linux 8.2 for SAP Applications” boot image must be used in order to successfully follow the steps in this guide. Due to networking differences between boot images, the workaround script described in the **Enabling Load Balancer Traffic from Backend Servers** section of GCP – Using an Internal Load Balancer may cause unexpected behavior on VMs running different operating systems.

2. All three VM instances have been configured to run SIOS Protection Suite for Linux. In particular, SELinux is disabled, the firewall is disabled, the /etc/hosts file on each node contains entries to resolve each node’s hostname to its private IP, and the root user has the same password on each node. See Configure Linux Nodes to Run SIOS Protection Suite for Linux for details.

3. SIOS Protection Suite for Linux has been installed on all three nodes with quorum/witness functionality enabled. The SAP Recovery Kit has been installed on node-a and node-b. On node-c (the witness node), no additional recovery kits beyond the core LifeKeeper installation are required. All necessary SIOS licenses have been installed on each node. See Install SIOS Protection Suite for Linux for details.

4. LifeKeeper communication paths have been defined between all pairs of cluster nodes. Note that this requires creation of **three** bi-directional communication paths (node-a ↔ node-b, node-a ↔ node-c, node-b ↔ node-c). See Login and Basic Configuration Tasks for details.

After completing all of these tasks, the LifeKeeper GUI should resemble the following image.



# 11.2.7.5.6.1. Create SAP HANA Primary Database Virtual IP

The virtual host name sps-hana will be configured to resolve to the current primary database host.

The recommended implementation of the virtual IP address varies by cloud platform. Please follow the steps provided in the appropriate section.

- GCP – Create SAP HANA Primary Database Load Balancer

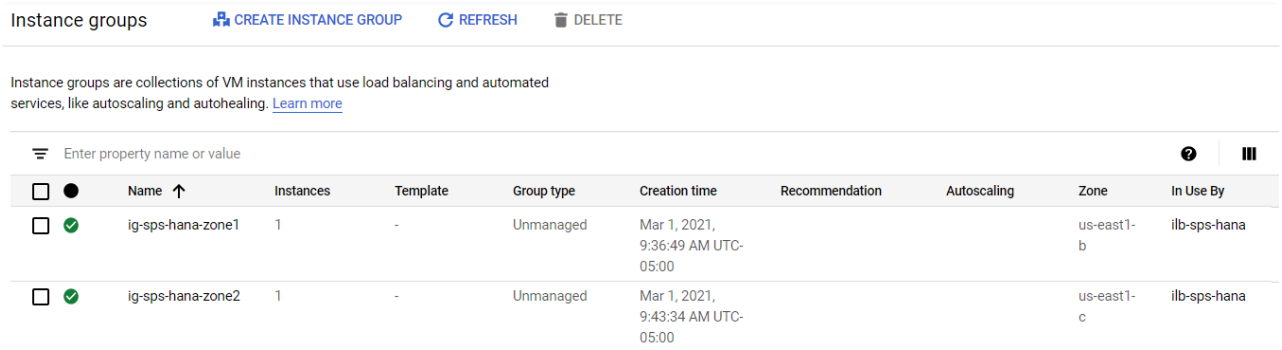
# 11.2.7.5.6.1.1. GCP – Create SAP HANA Primary Database Load Balancer

In Google Cloud, a TCP Internal Load Balancer is used to facilitate failover of the SAP HANA virtual IP. The frontend of the load balancer is assigned an ephemeral IP from the subnet that it operates in. In this example, we will be using the following IP address for load balancer corresponding to the sps-hana virtual hostname:

- SAP HANA virtual host:
  - Virtual hostname: sps-hana
  - Internal TCP load balancer: ilb-sps-hana
  - IP address: 10.20.0.10

Following the steps provided in GCP – Using an Internal Load Balancer and Responding to Load Balancer Health Checks, create and configure the following Google Cloud and LifeKeeper resources.

1. Create two unmanaged instance groups: **ig-sps-hana-zone1**, containing node-a, and **ig-sps-hana-zone2**, containing node-b.

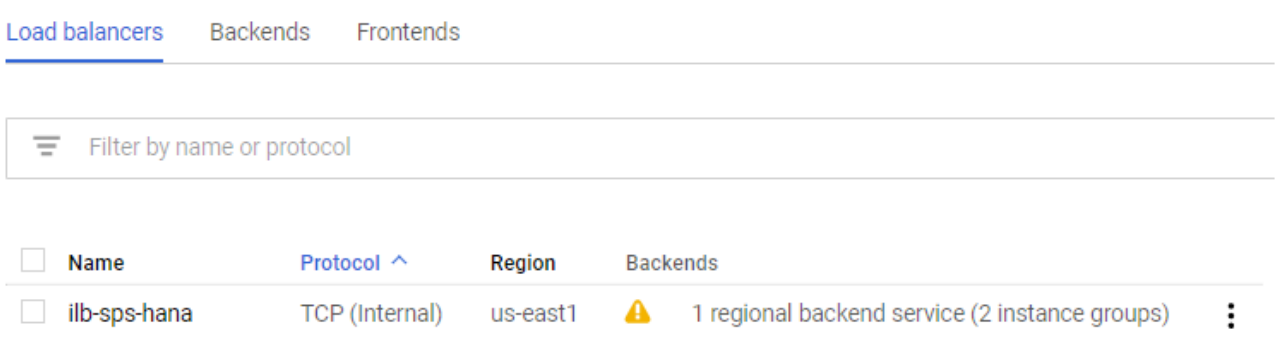



2. Create a TCP Internal Load Balancer, **ilb-sps-hana**, with the following properties. The ✓ icon indicates that the default option is chosen.

| Name | ilb-sps-hana |
|---|---|
| **Backend configuration** | |
| Region | <Deployment region> (e.g., us-east1) |
| Network | lk-vpc |
| Backends | ig-sps-hana-zone1<br>ig-sps-hana-zone2 |
| **Health Check** | |
| Name | hc-sps-hana |
| Protocol | TCP ✓ |

| | |
|---|---|
| Port | 50098 (This must agree with the corresponding LifeKeeper GenLB resource created later) |
| Proxy protocol | NONE |
| Request | Leave empty |
| Response | Leave empty |
| Check interval | 2 seconds |
| Timeout | 2 seconds |
| Healthy threshold | 2 consecutive successes |
| Unhealthy threshold | 3 consecutive failures |
| Connection draining timeout | 10 seconds |
| **Frontend configuration** | |
| Name | fe-sps-hana |
| Subnetwork | lk-subnet |
| Purpose | Non-shared |
| IP address | Ephemeral (Custom) |
| Custom ephemeral IP address | 10.20.0.10 |
| Ports | All |
| Global access | Disable |
| Service label | Leave empty |

Once created, the **Network services** → **Load balancing** page in the Google Cloud Console will show the load balancer.



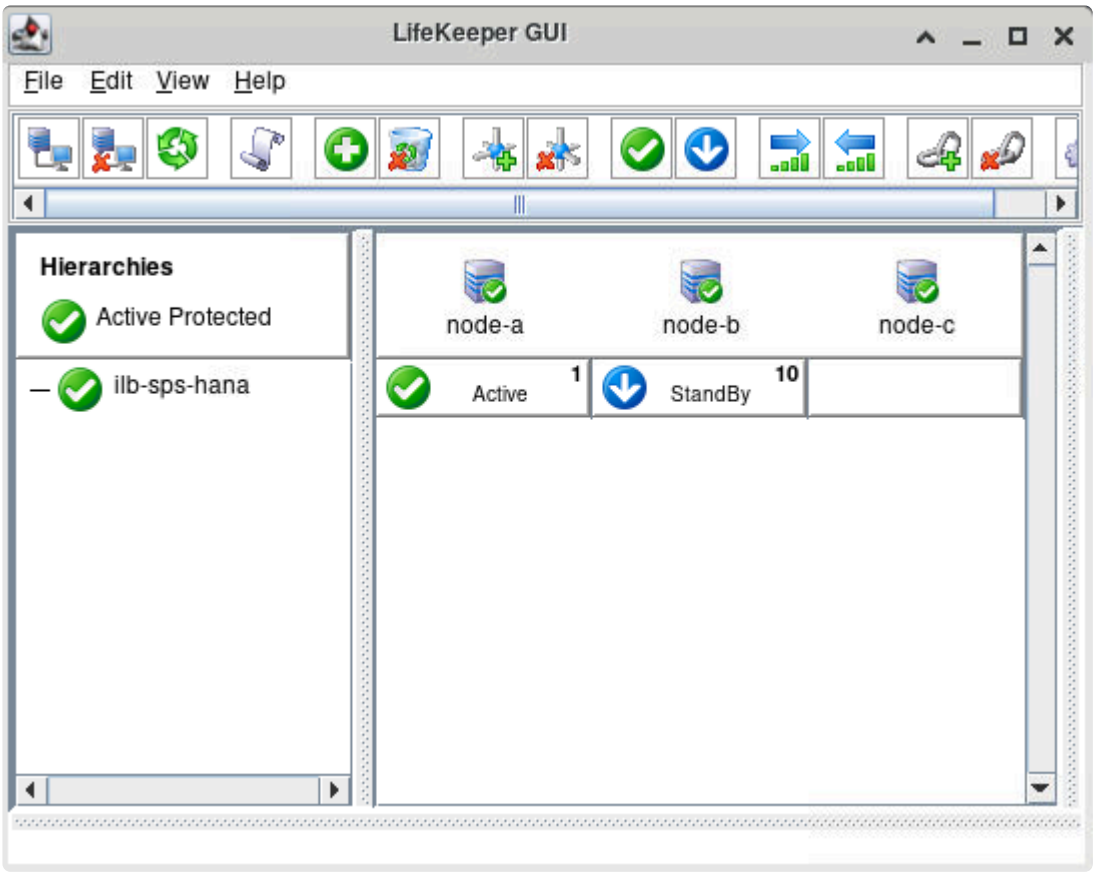
3. [Optional] Implement the startup script workaround described in the **Enabling Load Balancer Traffic from Backend Servers** section of GCP – Using an Internal Load Balancer to allow node-a and node-b to communicate through the frontend IP address of the load balancer. Reboot node-a and node-b for the changes to take effect. This step is optional for a SAP HANA deployment since inter-backend communication through the frontend IP of the load balancer is not required in this configuration, but it may make it easier to test the functionality of the **ilb-sps-hana** GenLB resource in the next step from within the cluster nodes themselves.

4. Following the steps in Responding to Load Balancer Health Checks, install the LifeKeeper Generic Application Recovery Kit for Load Balancer Health Checks (“GenLB Recovery Kit”) and create a LifeKeeper GenLB resource, **ilb-sps-hana**, with the following properties. The ✓ icon indicates that the default option is chosen.

| Field | Value |
|---|---|
| **Create Resource Wizard** | |
| Switchback Type | intelligent ✓ |
| Server | node-a ✓ |
| Restore Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/restore.pl |
| Remove Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/remove.pl |
| QuickCheck Script | /opt/LifeKeeper/SIOS_Hotfixes/Gen-LB-PL-7172/quickCheck |
| Local Recovery Script | None (Empty) |
| Application Info | 50098 |
| Bring Resource In Service | Yes |
| Resource Tag | ilb-sps-hana |
| **Pre-Extend Wizard** | |
| Target Server | node-b ✓ |
| Switchback Type | intelligent ✓ |
| Template Priority | 1 ✓ |
| Target Priority | 10 ✓ |
| **Extend gen/app Resource Hierarchy Wizard** | |
| Resource Tag | ilb-sps-hana ✓ |
| Application Info | 50098 ✓ |

The resource will appear in the LifeKeeper GUI resource pane once it has been created and

extended successfully.




5. Add the following entry to /etc/hosts on node-a and node-b to allow resolution of the virtual hostname to the frontend IP address of the corresponding load balancer:

   ```
   10.20.0.10 sps-hana
   ```

6. Test switchover and failover of the GenLB resources as described in the **Test GenLB Resource Switchover and Failover** section of Responding to Load Balancer Health Checks. Correct any issues found or tune the parameters of the load balancer health checks as required to achieve successful operation.

# 11.2.7.5.6.2. Attach Disks for SAP HANA File Systems

1. Create and attach three additional disks to node-a and node-b to support the installation of SAP HANA. The device names used in this example may vary depending on your environment. Note that these disk sizes are used for evaluation purposes only. Consult the relevant documentation from SAP, your cloud provider, and any third-party storage provider when provisioning resources in a production environment.

| Device Name | Size | Mount Point |
|---|---|---|
| /dev/sdb | 20GB | /hana/log |
| /dev/sdc | 30GB | /hana/data |
| /dev/sdd | 60GB | /hana/shared |

2. Execute the following commands on both node-a and node-b to create a /dev/sdb1 partition with an xfs file system.

```
[root@node-a ~]# parted /dev/sdb --script mklabel gpt mkpart xfspart xfs 0% 10
0%

[root@node-a ~]# mkfs.xfs /dev/sdb1
meta-data=/dev/sdb1    isize=512   agcount=4, agsize=1310592 blks
         =             sectsz=4096 attr=2, projid32bit=1
         =             crc=1       finobt=1, sparse=1, rmapbt=0
         =             reflink=1
data     =             bsize=4096  blocks=5242368, imaxpct=25
         =             sunit=0     swidth=0 blks
naming   =version 2    bsize=4096  ascii-ci=0, ftype=1
log      =internal log bsize=4096  blocks=2560, version=2
         =             sectsz=4096 sunit=1 blks, lazy-count=1
realtime =none         extsz=4096  blocks=0, rtextents=0

[root@node-a ~]# partprobe /dev/sdb1
```

   Repeat these commands for /dev/sdc and /dev/sdd on both node-a and node-b.

3. Execute the following command to create the mount points for the SAP HANA file systems:

```
[root@node-a ~]# mkdir -p /hana/{log,data,shared}
```

4. Use the `blkid` command to view the UUID's for each partition.

```
[root@node-a ~]# blkid | sort
/dev/sdb1: UUID="f058a8de-4e17-4c1d-97a3-a2672aff58dd" TYPE="xfs" PARTLABEL="x
fspart" PARTUUID="f780893d-bd61-4beb-89df-fb23ea04f16e"
/dev/sdc1: UUID="c0ab7218-ec24-4a6f-b883-37e505ec806c" TYPE="xfs" PARTLABEL="x
fspart" PARTUUID="34c47333-7174-468c-b8f0-a3491cd0fe0c"
/dev/sdd1: UUID="057f4f63-8fb3-486b-82b5-49fbbd4914cc" TYPE="xfs" PARTLABEL="x
fspart" PARTUUID="0c9cc7d3-9622-4b3f-8949-e744a02af466"
```

5. Referencing the UUID's found in step 3, add the following entries to /etc/fstab on node-a and node-b to allow the file systems to be mounted at boot. Replace the sample UUID's given in this example with the UUID's of the appropriate partitions on your systems.

```
UUID=f058a8de-4e17-4c1d-97a3-a2672aff58dd       /hana/log      xfs     default
s        0 0
UUID=c0ab7218-ec24-4a6f-b883-37e505ec806c       /hana/data       xfs     defau
lts        0 0
UUID=057f4f63-8fb3-486b-82b5-49fbbd4914cc       /hana/shared    xfs     defaul
ts        0 0
```

6. Execute the following command on node-a and node-b to mount the SAP HANA file systems on both nodes:

```
mount -a
```

The file systems are now prepared for SAP HANA installation.

# 11.2.7.5.6.3. Install SAP HANA and Configure System Replication

## Install SAP HANA on node-a

1. Create a /sap-install directory on node-a which will contain the SAP HANA installation files.

```
mkdir /sap-install
```

2. Download the SAP HANA installation files to /sap-install on node-a.

3. Extract the SAP HANA installation files. In this example we are installing SAP HANA 2.0 SPS04 Platform Edition.

```
[root@node-a ~]# unzip /sap-install/hana-2-platform-edition-sp4.zip -d /sap-in
stall/
```

4. Run the hdblcm executable to begin the installation.

```
[root@node-a ~]# /sap-install/DATA_UNITS/HDB_LCM_LINUX_X86_64/hdblcm
```

5. Enter the following parameters to complete a basic installation of the SAP HANA Database Server.

   Select other optional components during installation as required for your deployment. The ✓ icon indicates that the default option is chosen.

| Field | Value |
|---|---|
| Enter selected action index | 1 (install) |
| Components to install | 2 (HDB Server) |
| Enter Installation Path | /hana/shared ✓ |
| Enter Local Host Name | node-a ✓ |
| Do you want to add hosts to the system? (y/n) | n ✓ |
| Enter SAP HANA System ID | SPS |
| Enter Instance Number | 00 ✓ |
| Enter Local Host Worker Group | default ✓ |
| Select Usage / Enter Index | 4 (custom) ✓ |

| | |
|---|---|
| Enter Location of Data Volumes | /hana/data/SPS |
| Enter Location of Log Volumes | /hana/log/SPS |
| Restrict maximum memory allocation? | n |
| Enter Certificate Host Name for Host ‘node-a’ | node-a |
| Enter System Administrator (spsadm) Password | <spsadm User Password> |
| Confirm System Administrator (spsadm) Password | <spsadm User Password> |
| Enter System Administrator Home Directory | /usr/sap/SPS/home |
| Enter System Administrator Login Shell | /bin/sh |
| Enter System Administrator User ID | 1001 (or default value) |
| Enter System Database User (SYSTEM) Password | <SYSTEM DB User Password> |
| Confirm System Database User (SYSTEM) Password | <SYSTEM DB User Password> |
| Restart system after machine reboot? | n |
| Do you want to continue? | y |

6. Once the installation completes, execute the following command on node-a to verify that the HDB00 instance is running successfully:

```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
04.03.2021 23:42:13
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2021 03 04 23:38:10, 0:04:03, 23804
hdbcompileserver, HDB Compileserver, GREEN, Running, 2021 03 04 23:38:41, 0:0
3:32, 24044
hdbnameserver, HDB Nameserver, GREEN, Running, 2021 03 04 23:38:11, 0:04:02, 2
3822
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2021 03 04 23:38:41, 0:03:3
2, 24047
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2021 03 04 23:39:24, 0:0
2:49, 24454
hdbindexserver, HDB Indexserver-SPS, GREEN, Running, 2021 03 04 23:38:41, 0:0
3:32, 24093
hdbxsengine, HDB XSEngine-SPS, GREEN, Running, 2021 03 04 23:38:41, 0:03:32, 2
4096
```

# Install SAP HANA on node-b

1. Repeat the installation steps from the previous section section on node-b, replacing “node-a” by “node-b” wherever it appears. The parameters (including the user ID of the spsadm user and the group ID of the sapsys group) must be exactly the same in both installations.

2. Once the installation completes, execute the following command on node-b to verify that the HDB00 instance is running successfully.

```
[root@node-b ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
04.03.2021 23:44:09
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2021 03 04 23:38:16, 0:05:53, 10785
hdbcompileserver, HDB Compileserver, GREEN, Running, 2021 03 04 23:38:46, 0:0
5:23, 11111
hdbnameserver, HDB Nameserver, GREEN, Running, 2021 03 04 23:38:16, 0:05:53, 1
0804
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2021 03 04 23:38:46, 0:05:2
3, 11114
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2021 03 04 23:39:29, 0:0
4:40, 11438
hdbindexserver, HDB Indexserver-SPS, GREEN, Running, 2021 03 04 23:38:47, 0:0
5:22, 11159
hdbxsengine, HDB XSEngine-SPS, GREEN, Running, 2021 03 04 23:38:47, 0:05:22, 1
1162
```

# Back up the SYSTEMDB and SPS Databases on Both Nodes

Execute the following commands on both node-a and node-b to create backups of the SYSTEMDB database and the SPS tenant database, replacing <SYSTEM User Password> with the password for the SYSTEM database user:

```
# su - spsadm -c "hdbsql -i 00 -u SYSTEM -p <SYSTEM User Password> -d SystemD
B \"BACKUP DATA USING FILE ('/hana/shared/SPS/HDB00')\""
0 rows affected (overall time 7390.598 msec; server time 7388.081 msec)

# su - spsadm -c "hdbsql -i 00 -u SYSTEM -p <SYSTEM User Password> -d SPS \"BA
CKUP DATA USING FILE ('/hana/shared/SPS/HDB00')\""
0 rows affected (overall time 5104.380 msec; server time 5102.842 msec)
```

# Copy the PKI ssfs KEY and DAT Files

1. Execute the following command to stop the HDB00 instance on node-b.

```
[root@node-b ~]# su - spsadm -c “sapcontrol -nr 00 -function StopSystem HDB”
04.03.2021 23:55:10
StopSystem
OK
```

2. Verify that the HDB00 instance has been successfully stopped on node-b.

```
[root@node-b ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
04.03.2021 23:55:51
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GRAY, Stopped, , , 10785
```

3. Execute the following command to copy the /usr/sap/SPS/SYS/global/security/rsecssfs/data/SSFS_SPS.DAT and /usr/sap/SPS/global/security/rsecssfs/key/SSFS_SPS.KEY files from node-a to node-b.

```
[root@node-a ~]# scp -r /usr/sap/SPS/SYS/global/security/rsecssfs/ root@node-b:/usr/sap/SPS/SYS/global/security/
SSFS_SPS.DAT                  100% 2960     2.8MB/s   00:00
SSFS_SPS.KEY                  100%  187   190.9KB/s   00:00
```

# Configure SAP HANA System Replication

1. Execute the following command on node-a to enable system replication using site name SiteA.

```
[root@node-a ~]# su - spsadm -c “hdbnsutil -sr_enable --name=SiteA”
nameserver is active, proceeding ...
successfully enabled system as system replication source site
done.
```

2. Execute the following command to register node-b as a secondary system replication site using site name SiteB.

```
[root@node-b ~]# su - spsadm -c "hdbnsutil -sr_register --remoteHost=node-a --remoteInstance=00 --replicationMode=sync --operationMode=logreplay --name=SiteB"
adding site ...
nameserver node-a:30001 not responding.
```

```
collecting information ...
updating local ini files ...
done.
```

3. Execute the following command to start the HDB00 instance on node-b.

```
[root@node-b ~]# su - spsadm -c "sapcontrol -nr 00 -function StartSystem HDB"
05.03.2021 00:11:51
StartSystem
OK
```

4. Execute the following commands to verify that the HDB00 instance is running successfully on node-b and that the replication and operation modes are primary/primary and sync/logreplay on node-a and node-b, respectively.

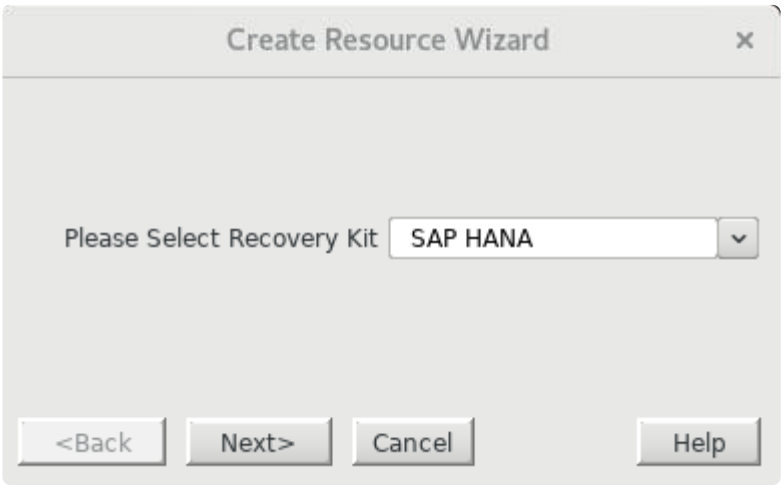
```
[root@node-b ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
05.03.2021 00:13:36
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2021 03 05 00:11:52, 0:01:44, 15816
hdbcompileserver, HDB Compileserver, GREEN, Running, 2021 03 05 00:11:57, 0:0
1:39, 15940
hdbindexserver, HDB Indexserver-SPS, GREEN, Running, 2021 03 05 00:11:57, 0:0
1:39, 15997
hdbnameserver, HDB Nameserver, GREEN, Running, 2021 03 05 00:11:52, 0:01:44, 1
5835
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2021 03 05 00:11:57, 0:01:3
9, 15943
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2021 03 05 00:12:02, 0:0
1:34, 16217
hdbxsengine, HDB XSEngine-SPS, GREEN, Running, 2021 03 05 00:11:57, 0:01:39, 1
6000


[root@node-b ~]# su - spsadm -c "hdbnsutil -sr_state"
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~

online: true

mode: sync
operation mode: logreplay
site id: 2
site name: SiteB
```

```
is source system: false
is secondary/consumer system: true
has secondaries/consumers attached: false
is a takeover active: false
active primary site: 1

primary masters: node-a

Host Mappings:
~~~~~~~~~~~~~~

node-b -> [SiteB] node-b
node-b -> [SiteA] node-a

Site Mappings:
~~~~~~~~~~~~~~
SiteA (primary/primary)
    |---SiteB (sync/logreplay)

Tier of SiteA: 1
Tier of SiteB: 2

Replication mode of SiteA: primary
Replication mode of SiteB: sync

Operation mode of SiteA: primary
Operation mode of SiteB: logreplay

Mapping: SiteA -> SiteB
done.
```

Now that the HDB00 instance is running on node-a and node-b and both nodes are registered in SAP HANA System Replication, we are ready to create the SAP HANA resources in LifeKeeper.

# 11.2.7.5.6.4. Create LifeKeeper SAP HANA Resource

1. In the LifeKeeper GUI, click the icon to open the **Create Resource Wizard**. Select the “SAP HANA” Recovery Kit.

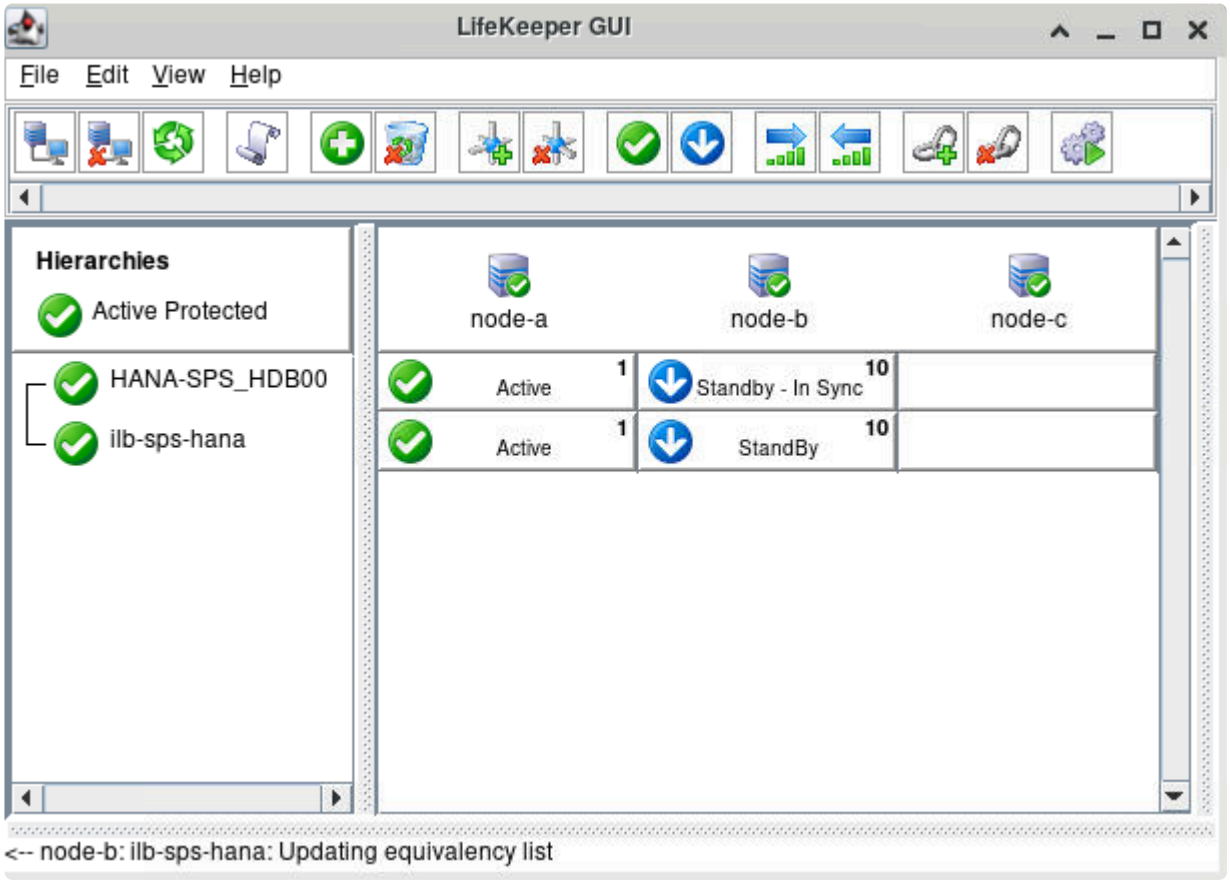



2. Enter the following values to create and extend a LifeKeeper resource (**HANA-SPS_HDB00**) to protect the HDB00 database instance on node-a and node-b. Note that the resulting resource should not be extended to node-c, the witness node. The icon indicates that the default option is chosen.

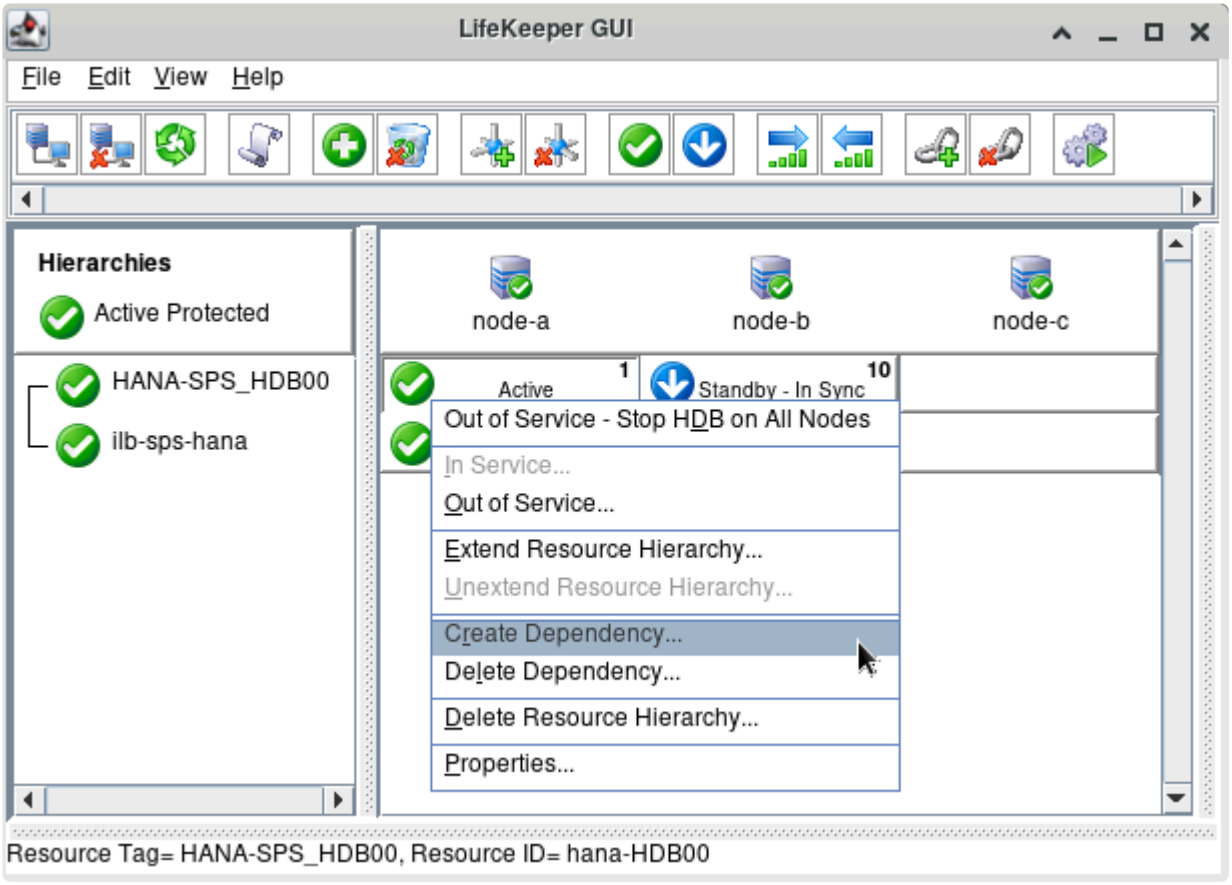
| Field | Value |
|---|---|
| **Create Resource Wizard** | |
| Switchback Type | intelligent |
| Server | node-a |
| SAP HANA SID | SPS |
| SAP HANA Instance for SPS | HDB00 |
| IP child resource | none |
| HANA Tagt | HANA-SPS_HDB00 |
| **Pre-Extend Wizard** | |
| Target Server | node-b |
| Switchback Type | intelligent |

| | |
|---|---|
| Template Priority | 1 |
| Target Priority | 10 |
| **Extend gen/app Resource Hierarchy Wizard** | |
| Root Tag | HANA-SPS_HDB00 |

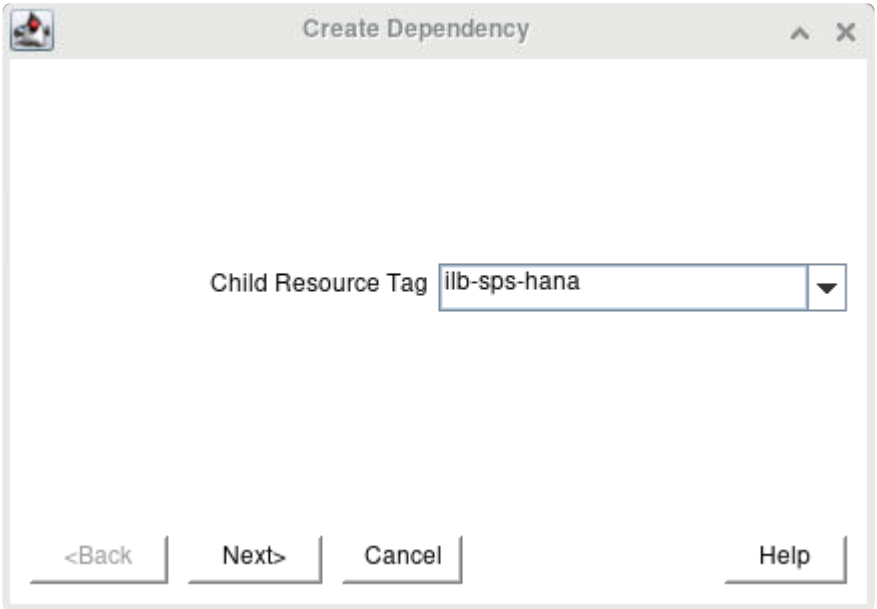
Once the **HANA-SPS_HDB00** resource has been successfully created and extended to node-b, the LifeKeeper resource panel should look similar to the following. Note that the names of the resources providing failover functionality for the SAP HANA virtual IP/hostname may differ according to your chosen cloud platform depending on which subsection of Create SAP HANA Primary Database Virtual IP was followed.



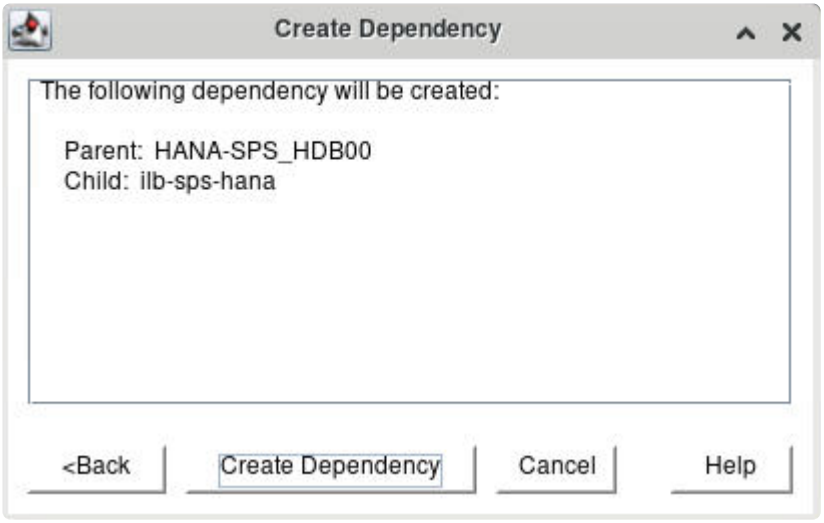

3. Right-click on the **SAP-SPS_HDB00** resource and select **Create Dependency...** from the drop-down menu.



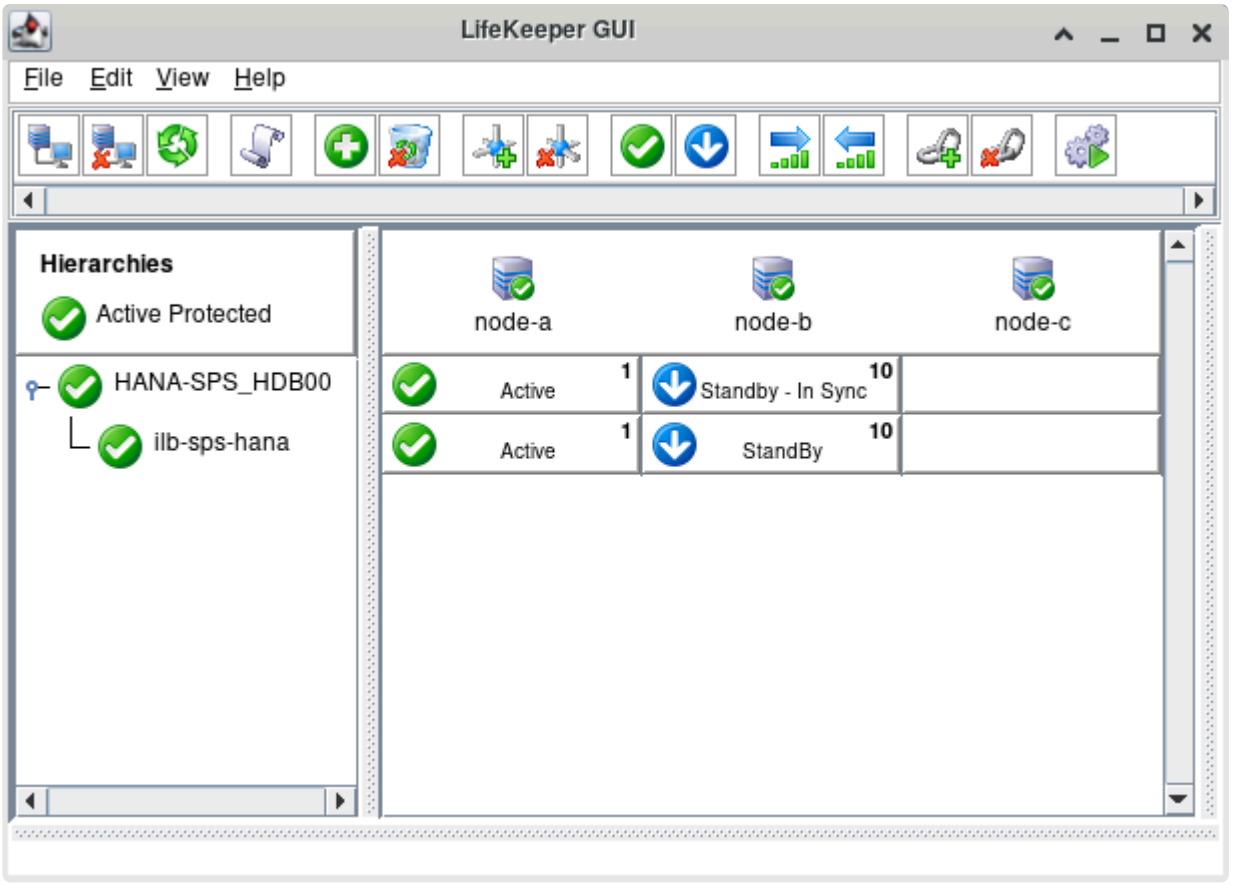

For **Child Resource Tag**, specify the root resource of the resource hierarchy being used to failover the SAP HANA virtual IP/hostname. The precise resource name used in this step will depend on resources created in the subsection of Create SAP HANA Primary Database Virtual IP corresponding to your chosen cloud platform.




Click **Next>** to continue, then click **Create Dependency** to create the dependency.




The resulting hierarchy will look similar to the following:

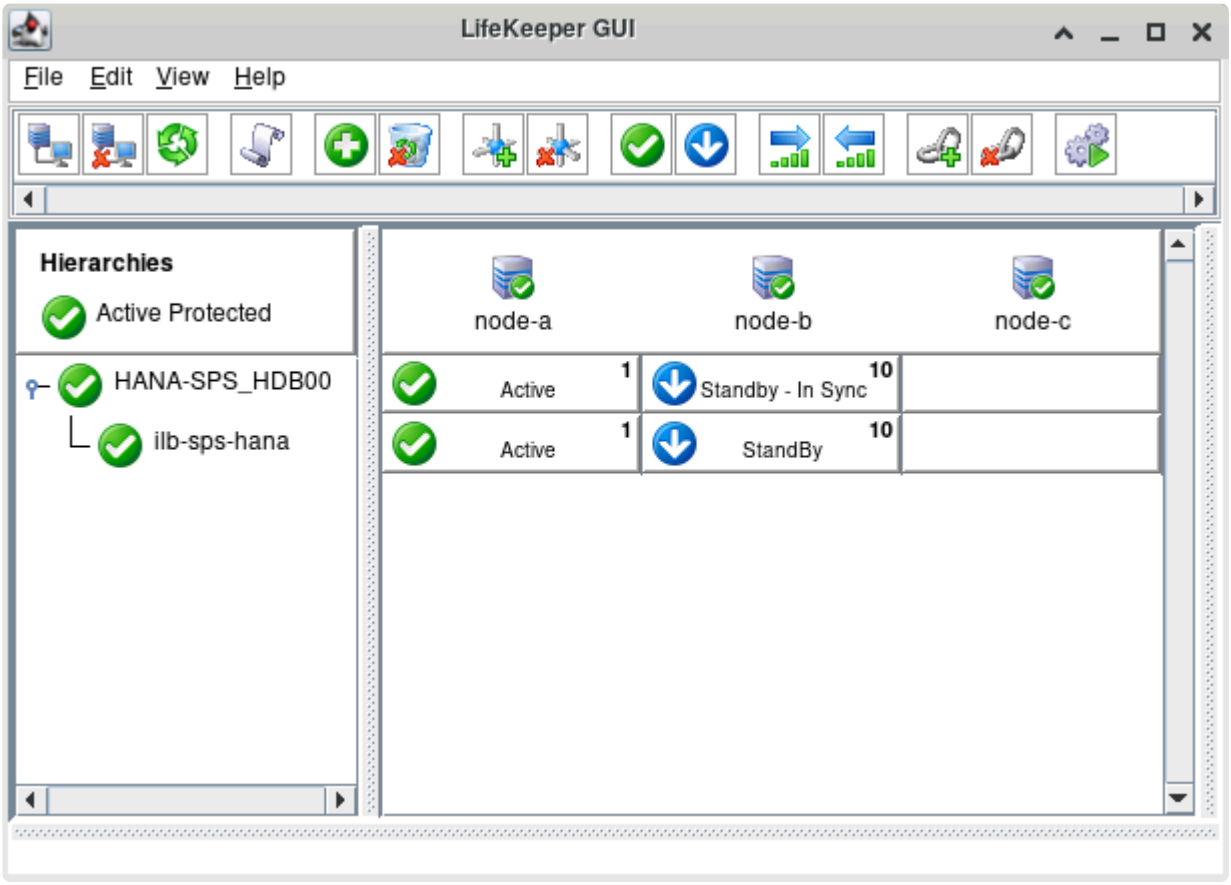



SAP HANA database instance HDB00 has now been successfully protected by LifeKeeper.

# 11.2.7.5.6.5. Test Switchover and Failover

In this section we will perform basic tests to verify the expected behavior of the **HANA-SPS_HDB00** resource hierarchy on switchover and failover.

1. Verify that the **HANA-SPS_HDB00** resource state is currently “Active” on node-a and “Standby – In-Sync” on node-b.



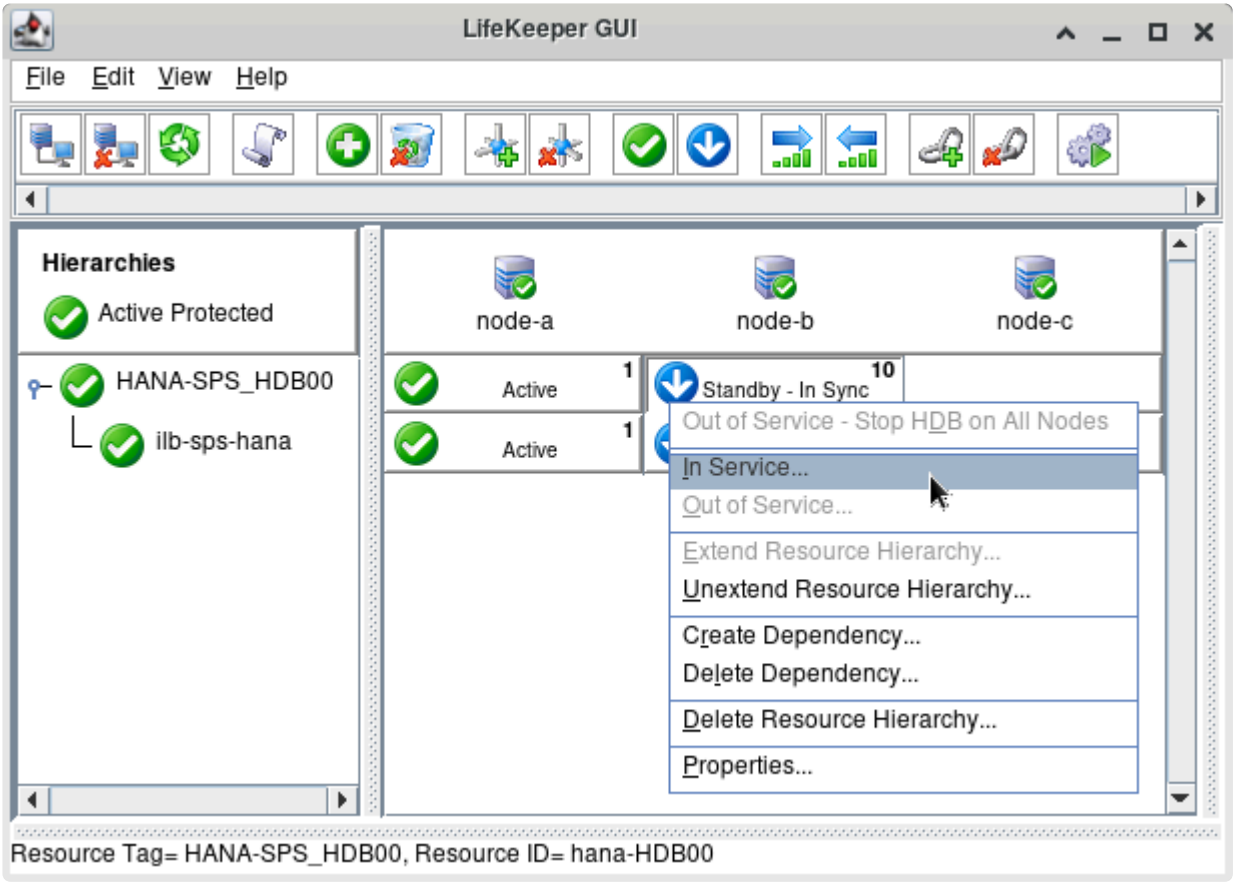

2. Execute the following commands on node-a and node-b to verify that the HDB00 instance is running successfully on both nodes, node-a and node-b are both registered appropriately in SAP HANA System Replication, and that a connection to the primary database can be established through the virtual hostname sps-hana. The replication and operation modes should be primary/ primary and sync/logreplay on node-a and node-b, respectively.

```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
04.03.2021 21:31:50
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2021 03 04 19:44:27, 1:47:23, 19236
hdbcompileserver, HDB Compileserver, GREEN, Running, 2021 03 04 19:45:46, 1:4
6:04, 19618
hdbindexserver, HDB Indexserver-SPS, GREEN, Running, 2021 03 04 19:45:47, 1:4
6:03, 19666
```

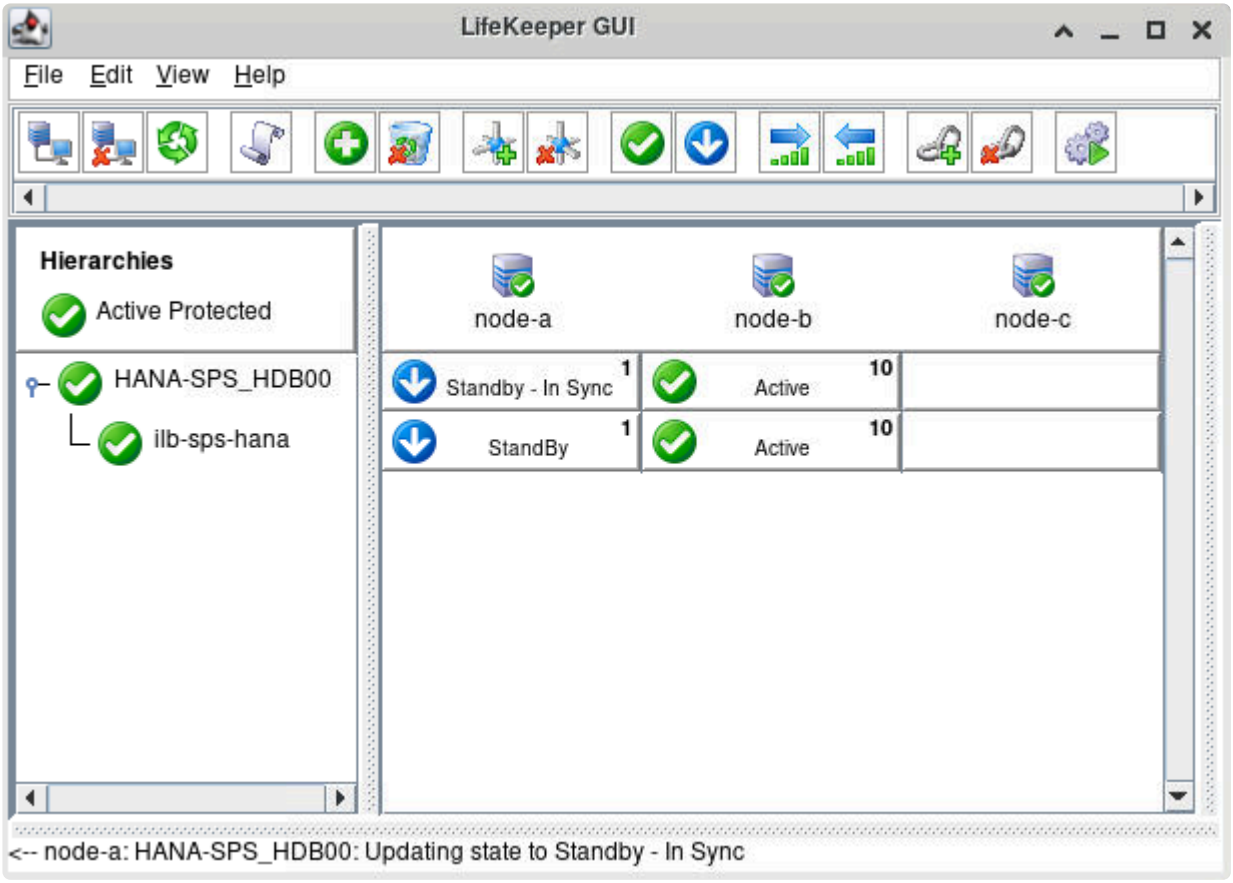
```
hdbnameserver, HDB Nameserver, GREEN, Running, 2021 03 04 19:44:27, 1:47:23, 1
9254
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2021 03 04 19:45:46, 1:46:0
4, 19621
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2021 03 04 19:46:15, 1:4
5:35, 20241
hdbxsengine, HDB XSEngine-SPS, GREEN, Running, 2021 03 04 19:45:47, 1:46:03, 1
9669

[root@node-a ~]# su - spsadm -c "hdbnsutil -sr_state"
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~

online: true

mode: primary
operation mode: primary
site id: 1
site name: SiteA

is source system: true
is secondary/consumer system: false
has secondaries/consumers attached: true
is a takeover active: false

Host Mappings:
~~~~~~~~~~~~~~

node-a -> [SiteB] node-b
node-a -> [SiteA] node-a

Site Mappings:
~~~~~~~~~~~~~~
SiteA (primary/primary)
    |---SiteB (sync/logreplay)

Tier of SiteA: 1
Tier of SiteB: 2


Replication mode of SiteA: primary
Replication mode of SiteB: sync

Operation mode of SiteA: primary
Operation mode of SiteB: logreplay

Mapping: SiteA -> SiteB
```

```
done.

[root@node-b ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
04.03.2021 21:38:10
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2021 03 04 19:48:50, 1:49:20, 13383
hdbcompileserver, HDB Compileserver, GREEN, Running, 2021 03 04 19:49:00, 1:4
9:10, 13747
hdbindexserver, HDB Indexserver-SPS, GREEN, Running, 2021 03 04 19:49:00, 1:4
9:10, 13794
hdbnameserver, HDB Nameserver, GREEN, Running, 2021 03 04 19:48:51, 1:49:19, 1
3401
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2021 03 04 19:49:00, 1:49:1
0, 13750
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2021 03 04 19:49:42, 1:4
8:28, 15167
hdbxsengine, HDB XSEngine-SPS, GREEN, Running, 2021 03 04 19:49:00, 1:49:10, 1
3797

[root@node-b ~]# su - spsadm -c "hdbsql -n sps-hana -i 00 -u SYSTEM -p <SYSTE
M user password> -d SPS '\s'"
host           : sps-hana:30013
sid            : SPS
dbname         : SPS
user           : SYSTEM
kernel version: 2.00.046.00.1581325702
SQLDBC version:        libSQLDBCHDB 2.04.182.1579711187
autocommit     : ON
locale         : en_US.UTF-8
input encoding: UTF8
sql port       : node-a:30015
```

3. Perform a switchover of the SAP HANA resource hierarchy by right-clicking the **HANA-SPS_HDB00** resource on node-b and choosing the **In-Service…** operation. Click **In Service** to begin the switchover.




Allow time for the switchover to complete and for the resource states to switch to “Standby – In Sync” on node-a and “Active” on node-b. Note that it may take up to two minutes for the resource state to transition from “Standby – HDB Running” to “Standby – In-Sync” after the standby database instance is running.




4. Execute the following commands on node-a and node-b to verify that the HDB00 instance is running successfully on both nodes, node-a and node-b are both registered appropriately in SAP HANA System Replication, and that a connection to the primary database can be established through the virtual hostname sps-hana. The replication and operation modes should now be sync/logreplay and primary/primary on node-a and node-b, respectively.

```
[root@node-a ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
04.03.2021 21:50:20
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2021 03 04 21:43:51, 0:06:29, 156845
hdbcompileserver, HDB Compileserver, GREEN, Running, 2021 03 04 21:43:58, 0:0
6:22, 156970
hdbindexserver, HDB Indexserver-SPS, GREEN, Running, 2021 03 04 21:43:58, 0:0
6:22, 157101
hdbnameserver, HDB Nameserver, GREEN, Running, 2021 03 04 21:43:52, 0:06:28, 1
56864
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2021 03 04 21:43:58, 0:06:2
2, 156973
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2021 03 04 21:44:36, 0:0
5:44, 157429
hdbxsengine, HDB XSEngine-SPS, GREEN, Running, 2021 03 04 21:43:58, 0:06:22, 1
57104
```

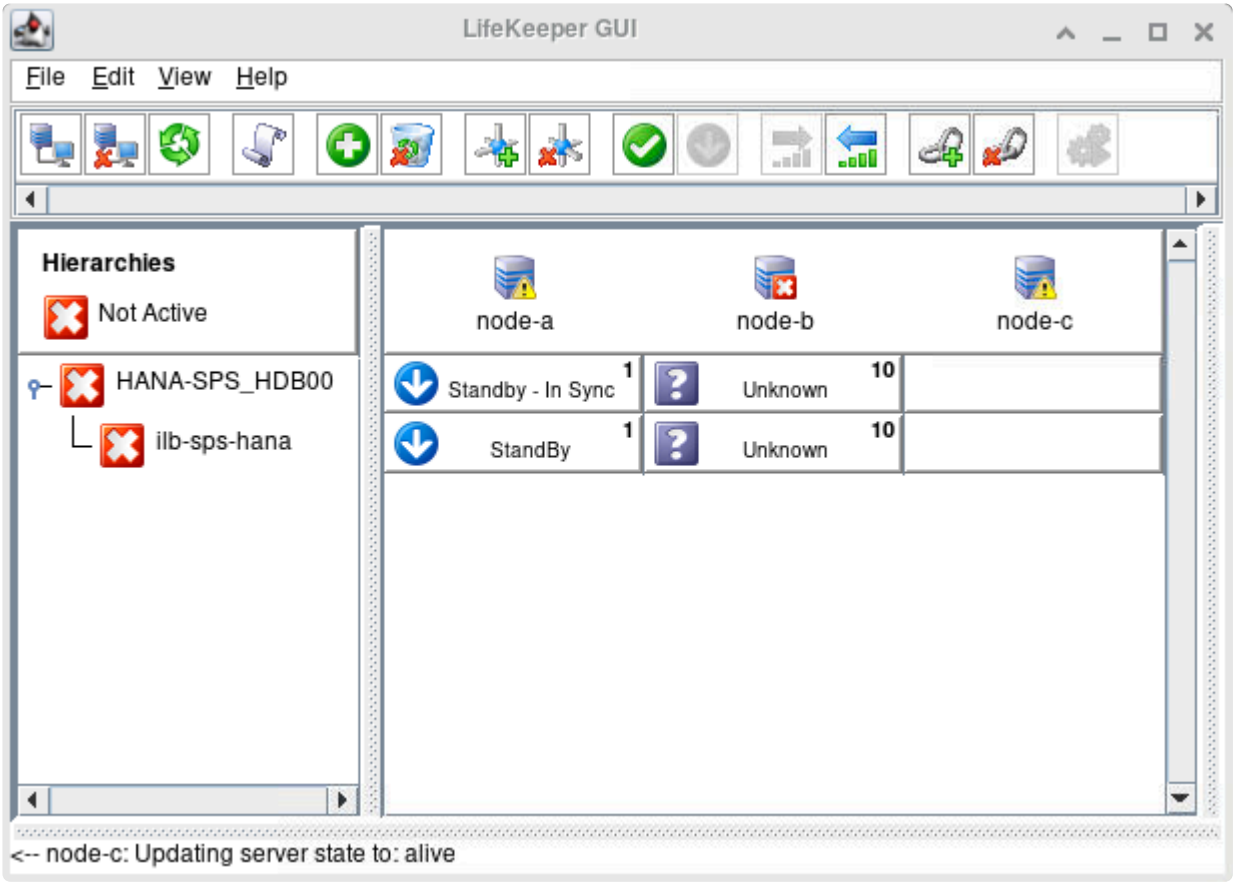
```
[root@node-a ~]# su - spsadm -c "hdbsql -n sps-hana -i 00 -u SYSTEM -p <SYSTE
M user password> -d SPS '\s'"
host           : sps-hana:30013
sid            : SPS
dbname         : SPS
user           : SYSTEM
kernel version: 2.00.046.00.1581325702
SQLDBC version:        libSQLDBCHDB 2.04.182.1579711187
autocommit     : ON
locale         : en_US.UTF-8
input encoding: UTF8
sql port       : node-b:30015

[root@node-b ~]# su - spsadm -c "sapcontrol -nr 00 -function GetProcessList"
04.03.2021 21:52:36
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2021 03 04 19:48:50, 2:03:46, 13383
hdbcompileserver, HDB Compileserver, GREEN, Running, 2021 03 04 19:49:00, 2:0
3:36, 13747
hdbindexserver, HDB Indexserver-SPS, GREEN, Running, 2021 03 04 19:49:00, 2:0
3:36, 13794
hdbnameserver, HDB Nameserver, GREEN, Running, 2021 03 04 19:48:51, 2:03:45, 1
3401
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2021 03 04 19:49:00, 2:03:3
6, 13750
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2021 03 04 19:49:42, 2:0
2:54, 15167
hdbxsengine, HDB XSEngine-SPS, GREEN, Running, 2021 03 04 19:49:00, 2:03:36, 1
3797

[root@node-b ~]# su - spsadm -c "hdbnsutil -sr_state"
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~

online: true

mode: primary
operation mode: primary
site id: 2
site name: SiteB

is source system: true
is secondary/consumer system: false
has secondaries/consumers attached: true
is a takeover active: false
```

```
Host Mappings:
~~~~~~~~~~~~~~

node-b -> [SiteB] node-b
node-b -> [SiteA] node-a

Site Mappings:
~~~~~~~~~~~~~~
SiteB (primary/primary)
    |---SiteA (sync/logreplay)

Tier of SiteB: 1
Tier of SiteA: 2

Replication mode of SiteB: primary
Replication mode of SiteA: sync

Operation mode of SiteB: primary
Operation mode of SiteA: logreplay

Mapping: SiteB -> SiteA
done.
```

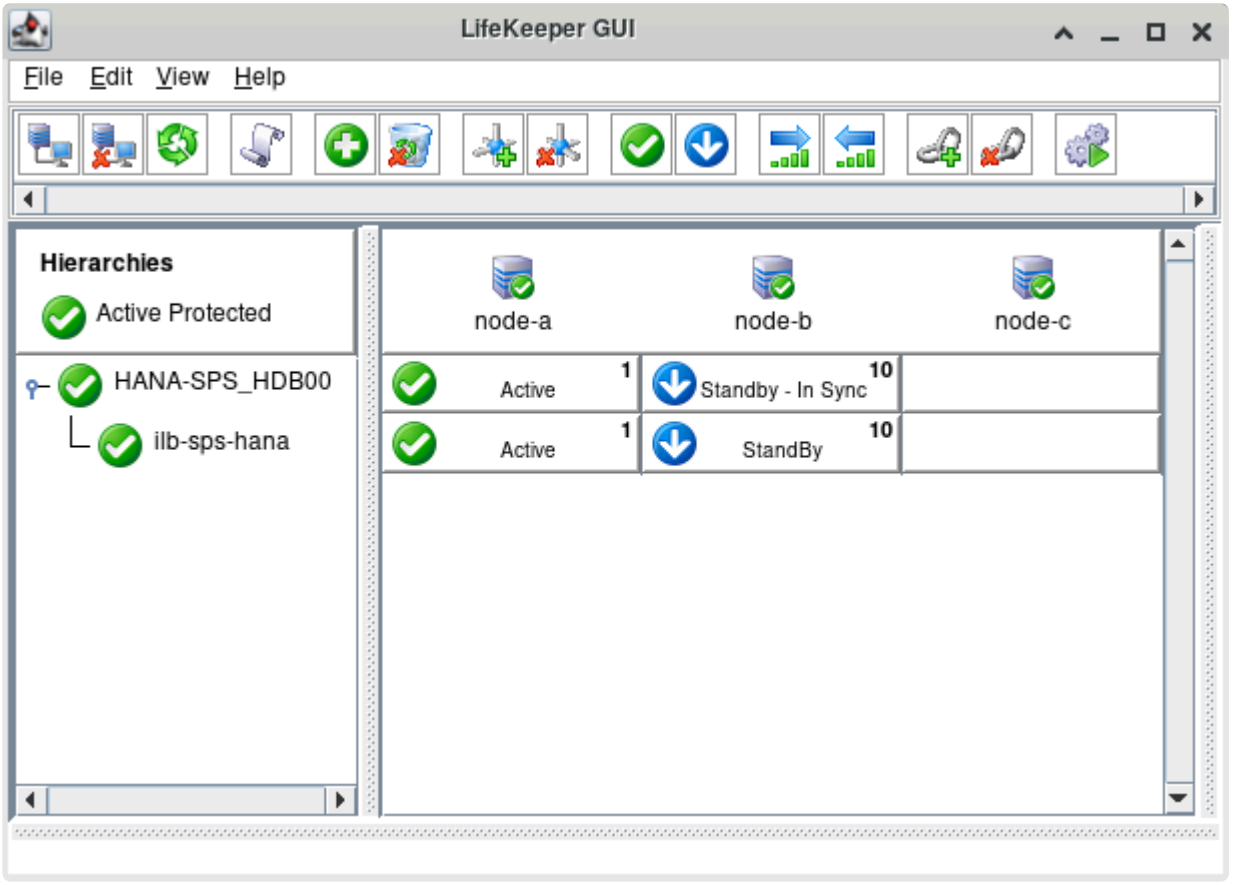
5. Execute the following command to forcefully reboot node-b:

```
[root@node-b ~]# echo b > /proc/sysrq-trigger
```

Once LifeKeeper has detected that node-b has been powered off, the status of node-b updates to “Unknown” in the LifeKeeper GUI and LifeKeeper initiates failover of the **HANA-SPS_HDB00** resource hierarchy back to node-a.




Once node-b is back online, LifeKeeper will automatically re-register it as a secondary replication site in SAP HANA System Replication and restart it. Once this process is complete, the resource states will transition back to “Active” on node-a and “Standby – In-Sync” on node-b.



6. Execute the commands given in step 2 again to verify that the database is functioning as expected on both nodes.

We have now verified the basic switchover and failover functionality of the SAP HANA resource hierarchy.

# 11.2.7.6. Common Tasks

This following topics contain common tasks across different resources.

- How to Confirm if the Data Storage is Available on a Node
- Switchover the Data Storage to the Other Node

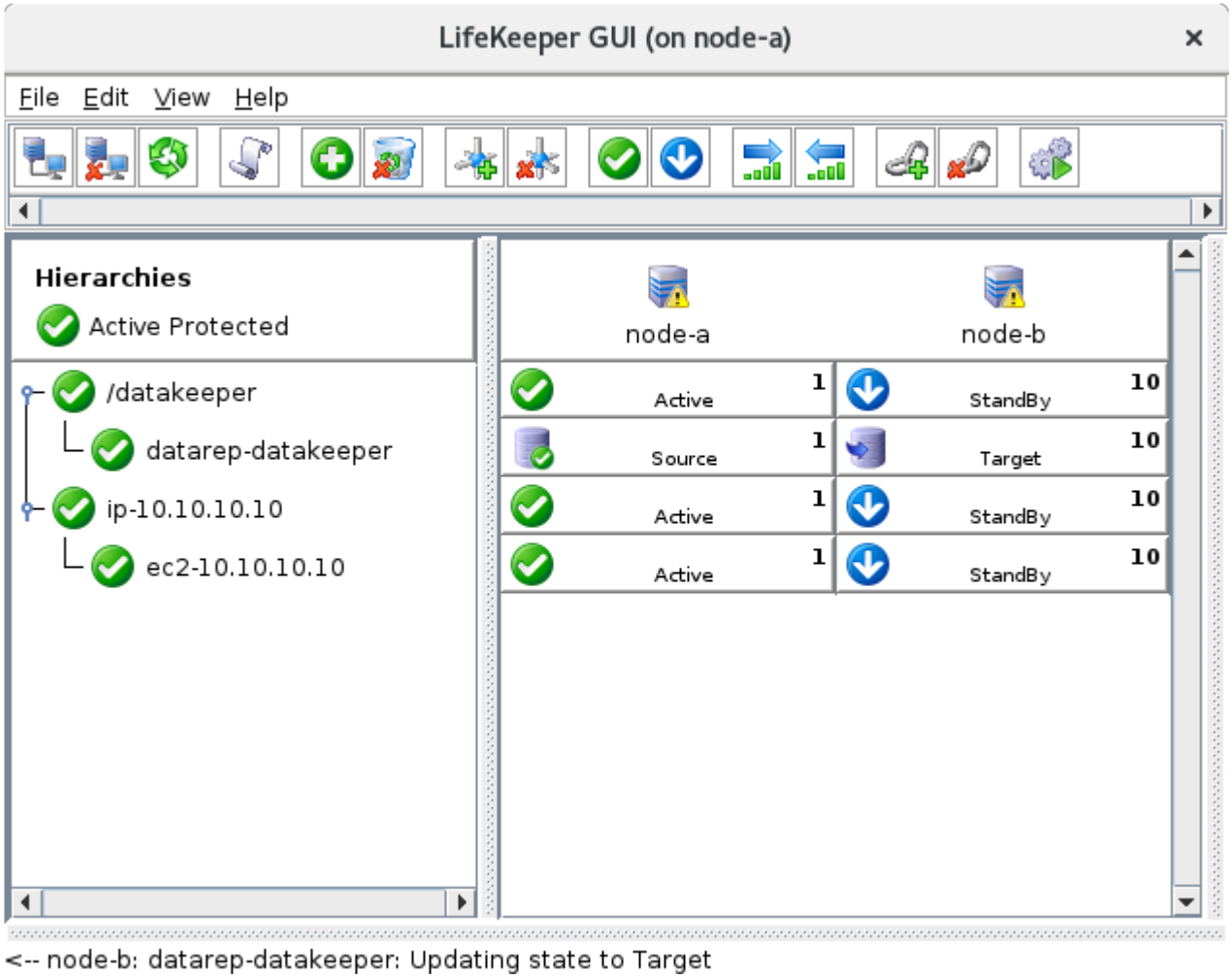
# 11.2.7.6.1. How to Confirm if the Data Storage is Available on a Node

During these steps we will be switching data storage between nodes. Here are two options to check whether the data storage (per this guide, located at /datakeeper) is available in a node.

- View the output from the `df` command.

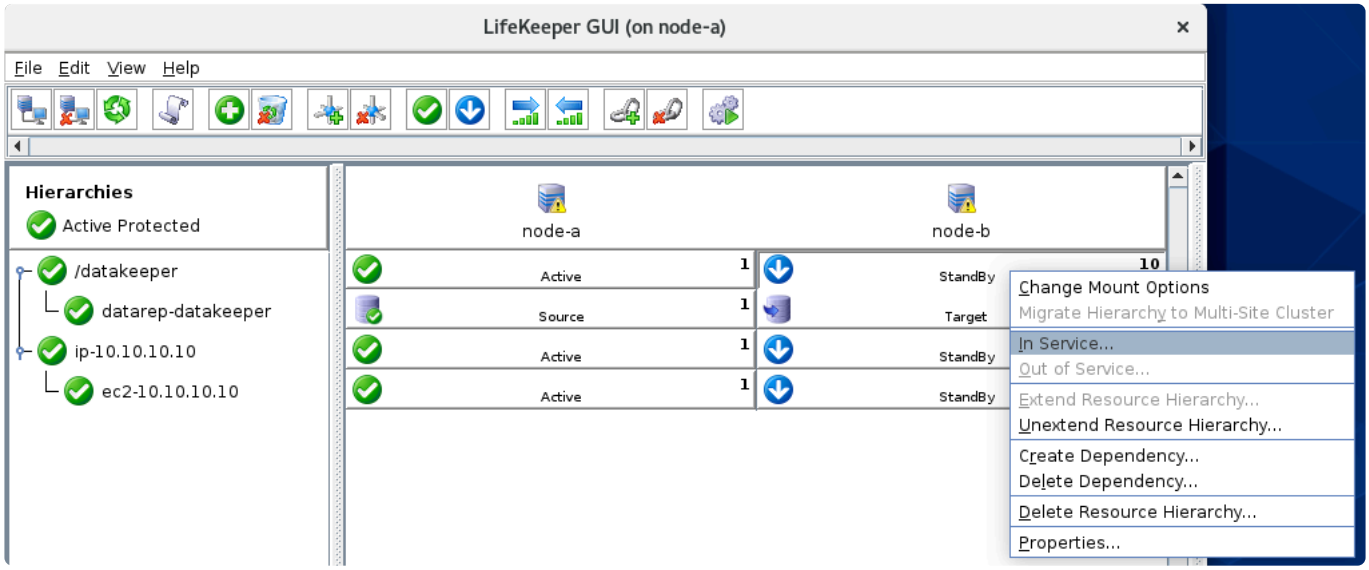
```
$ df
Filesystem     1K-blocks     Used Available Use% Mounted on
devtmpfs          916888        0    916888   0% /dev
tmpfs            7340032        0   7340032   0% /dev/shm
tmpfs             940184    17424    922760   2% /run
tmpfs             940184        0    940184   0% /sys/fs/cgroup
/dev/xvda2      31444972 17183608  14261364  55% /
tmpfs             188040        0    188040   0% /run/user/1000
/dev/md0        20960236  2510944  18449292  12% /datakeeper
```

- Check the LifeKeeper GUI to see the status of the `datarep-datakeeper` resource. The node status should be “Source” and the other node should be labeled as “Target”.

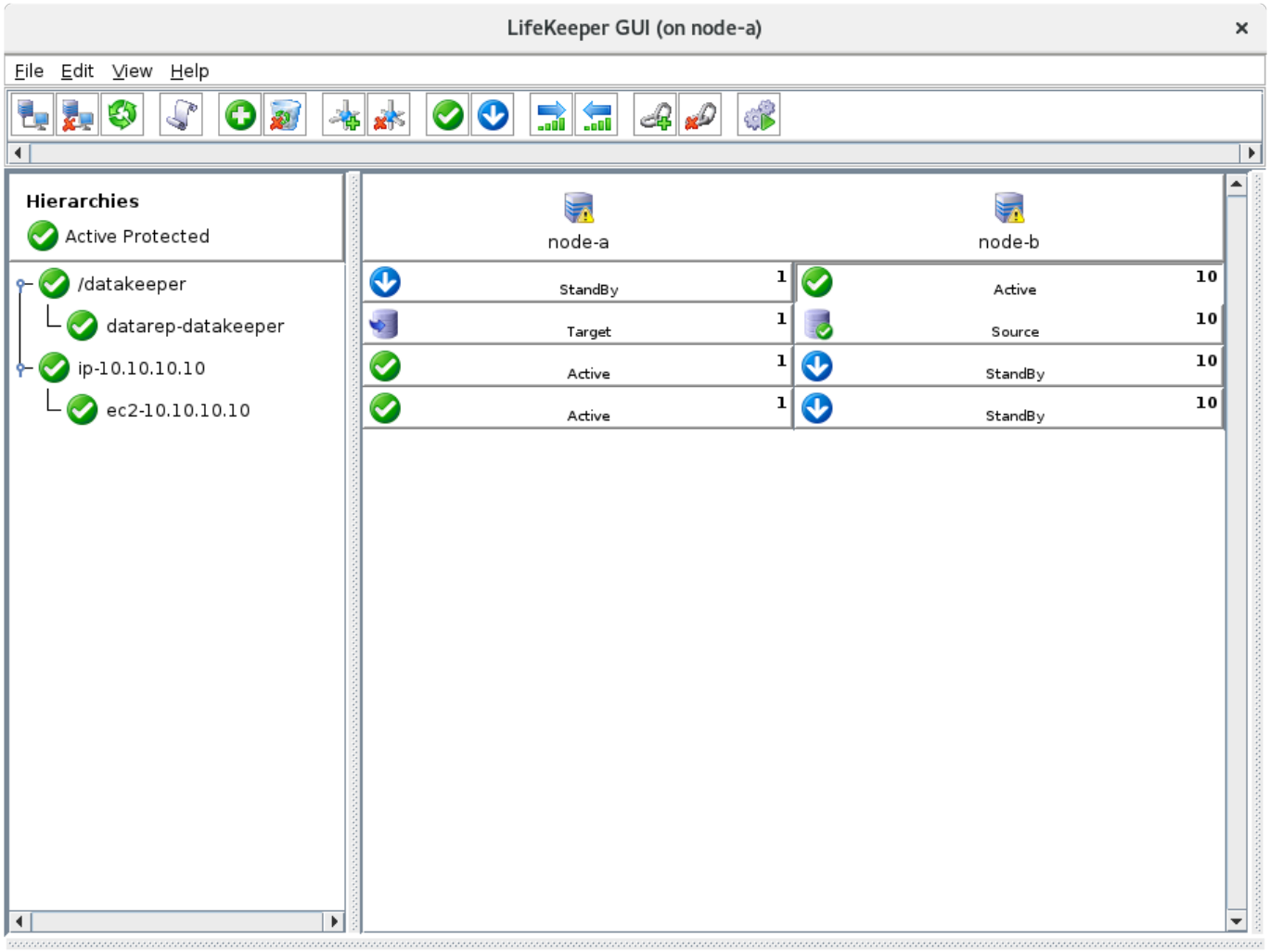


# 11.2.7.6.2. Switchover the Data Storage to the Other Node

To switchover the disk to the other node using the LifeKeeper GUI, select the `/datakeeper` resource on the standby node, then select “In Service…”.




Once the `datarep-datakeeper` resource becomes active on the other node, the Source & Target labels are switched.




<-- node-a: datarep-datakeeper: Updating state to Target

# 12. Quick Start Guides

AWS Direct Connect Quick Start Guide Protection Suite for Linux in the AWS Cloud (SAP)

SAP HANA Recovery Kit

AWS VPC Peering Connections Quick Start Guide

DataKeeper for Linux Evaluation Guide

MySQL Cluster with Data Replication (i.e.

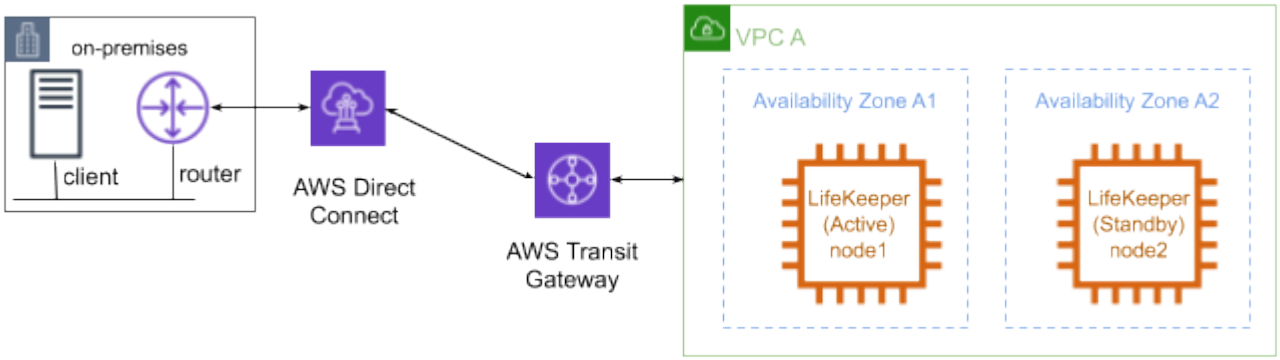
PostgreSQL Cluster with Shared Storage

Apache/MySQL Cluster Using Both Shared and Replicated Storage

# 12.1. AWS Direct Connect クイックスタートガイド

## 目的

AWS Transit Gateway のリリースにより、AWS Direct Connect を利用したオンプレミス環境（下図の On-premises）から AWS Transit Gateway を経由して、下図の VPC A 内に配置された HA クラスターノードへ接続する構成で、Recovery Kit for EC2 のルートテーブルシナリオが利用可能になりました。




本資料は、LifeKeeper for Linux で VPC 外から接続する構成を構築するための要件や基本操作を解説するものです。

なお、本資料は LifeKeeper や Amazon Web Service(以下 AWS)の基本的な設定や操作、技術的な詳細情報を解説するものではありません。 本構成の前提となる LifeKeeper や AWS に関する用語･操作･技術情報等につきましては、関連のマニュアルやユーザーサイト等であらかじめご確認ください。

> 注記:「Amazon Web Services」、「Powered by Amazon Web Services」のロゴ、「AWS」、「Amazon EC2」、「EC2」、「Amazon Elastic ComputeCloud」、「Amazon Virtual Private Cloud」、「AWS Direct Connect」、「AWS Identity and Access Management」、「AWS Transit Gateway」および「Amazon VPC」は、米国その他の国における Amazon.com, Inc. またはその関連会社の商標です。

# 12.1.1. 利用のための要件

本構成を利用するためには、環境を準備する段階で満たすべきいくつかの要件があります。以下に AWS 環境とその上に作成するインスタンスに関する要件をまとめます。

## AWS 環境の要件

サービスを提供するための基盤となる環境を AWS 上に作成します。本構成を利用するための要件は以下の通りです。

### Amazon Virtual Private Cloud (VPC)

- VPC を AWS 内に設定する必要があります。
- プライマリ用インスタンスを配置するサブネットとスタンバイ用インスタンスを配置するサブネットを、それぞれ異なる Availability Zone(AZ) に作成する必要があります。

### Amazon Elastic Compute Cloud (EC2)

- インスタンスが 2つ以上必要です。
- プライマリ用インスタンスとスタンバイ用インスタンスがそれぞれ異なる AZ で起動するように構成する必要があります。
- インスタンスは、Elastic Network Interface (ENI) に接続されます。
- インスタンスは、LifeKeeper のインストール要件を満たす必要があります。
- AWS Command Line Interface (AWS CLI) を全ての EC2 インスタンスにインストールする必要があります。インストール方法は、「AWS Command Line Interface のインストール 」を参照してください。
- https を使用して Amazon EC2 Web サービスエンドポイントの URL (EC2 URL) に、および http を使用して Amazon EC2 メタデータの URL (http://169.254.169.254/) にアクセス可能でなければなりません。

### AWS Identity and Access Management (IAM)

LifeKeeper が AWS を操作するために、以下のアクセス権を持った IAM ユーザーもしくは IAM ロールが必要です。EC2 インスタンスの root ユーザーからアクセスできるように EC2 の IAM ロール を設定するか、AWS CLI の設定 を適切に行ってください。

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress

- ec2:DescribeRouteTables
- ec2:ReplaceRoute

### AWS Transit Gateway

- クラスターノードを配置した VPC と、クライアントを配置したオンプレミス環境とは、Virtual Private Gateway で接続するのではなく、AWS Transit Gateway を経由して接続する必要があります。
- AWS Transit Gateway を作成する際に Default route table association と Default route table propagation を有効にしてください。
- VPC との接続は、Transit Gateway Attachment を作成することで行ってください。
- AWS Direct Connect との接続は Direct Connect Gateway の Gateway association の設定で、作成した AWS Transit Gateway を選択することで行なってください。 この際に、 Allowed prefixes には、クラスターノードを配置した VPC のネットワークアドレスと、仮想 IP アドレスの両方を設定してください。

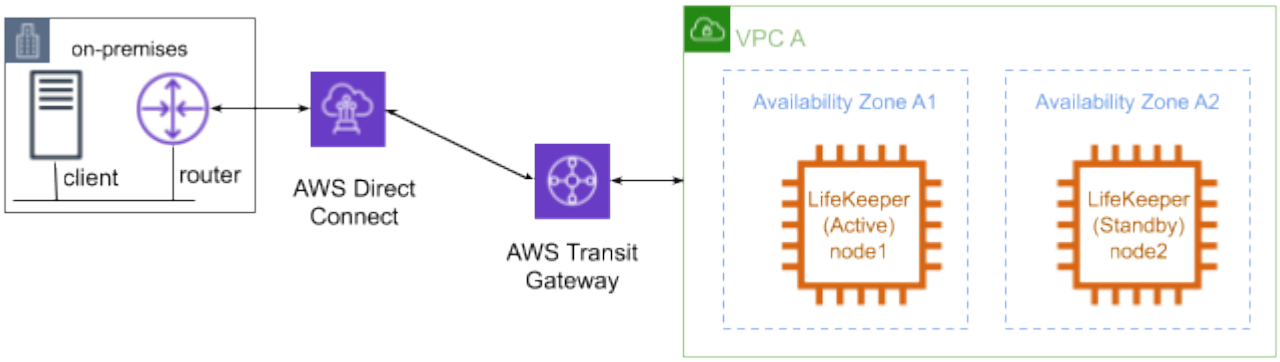
## LifeKeeper ソフトウェアの要件

各サーバーに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。本構成で必要な Application Recovery Kit (ARK) は以下の通りです。具体的な LifeKeeper の要件については、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート を参照してください。

- LifeKeeper IP Recovery Kit
- LifeKeeper Recovery Kit for EC2

# 12.1.2. AWS Direct Connect 構築手順

以下の構成図の様な環境を構築するための一般的な手順を解説します。



# 12.1.2.1. AWS Direct Connect 準備

「AWS Direct Connect の要件」を満たす環境を構築してください。それぞれのインスタンスに LifeKeeper をインストールして、node1/node2 間にコミュニケーションパスを作成してください。

# 12.1.2.2. Direct Connect リソースの作成

## IP リソース作成

仮想 IP リソースを作成します。IP リソースのアドレスは、VPC で管理している CIDR ブロック外である必要があります。

## EC2 リソース作成

EC2リソースを作成します。リソース作成時に要求される IP リソースは、上述の「**IP リソース作成**」で作成したリソースを指定してください。リソース作成時に要求される EC2 リソースタイプは Route Table (Backend Cluster) を指定してください。

## 保護するサービスのリソース作成

保護するサービスのためのリソースを作成してください。リソース作成に IP リソースが要求される場合は、上述の「**IP リソース作成**」で作成したリソースを指定してください。親リソースが保護するサービスのリソース、子リソースが EC2 リソースとなるようにリソースの依存関係を設定してください。

# 12.1.2.3. ルートテーブルの設定

ルートテーブルを下記のように設定してください。

- クラスターノードを配置した VPC もしくはサブネットのルートテーブルに、オンプレミス環境のネットワークへの経路情報を追加してください。

| 送信先 | ターゲット |
|---|---|
| オンプレミス環境のネットワークアドレス | 作成した Transit Gateway |

- Transit Gateway のルートテーブルに、仮想 IP アドレスへの経路情報を追加してください。

| 送信先 | ターゲット |
|---|---|
| 仮想 IP アドレス | クラスターノードを配置した VPC |

- オンプレミス環境のクライアントやルーターの経路情報を適切に設定し、クラスターノードを配置した VPC のネットワークアドレスや仮想 IP アドレス宛のパケットの送信先が Direct Connect となるようにしてください。

設定が完了したら、クライアントからクラスタサーバーのプライベートアドレスや仮想 IP アドレスにアクセスできることを確認してください。

# 12.1.3. 本構成における設定および運用上の留意点

## LifeKeeper I-O フェンシングの利用を検討してください

AWS 環境では共有ディスク環境が構成できないため、スプリットブレイン問題を防ぐために SCSI リザーブ方式を採ることができません。これを踏まえて、本構成ではより安全に運用できるように、LifeKeeper の I/O フェンシング機能の Quorum / Witness Server もしくは STONITH の利用をご検討ください。

特に、Quorum モードの TCP_REMOTE 設定を利用すれば、別途 Quorum サーバを立てずに I/Oフェンシング機能を実装できるため、クラウド環境においては利用しやすいと考えられます。利用方法については以下の URL をご確認ください。

- Quorom/Witness
- STONITH

## 既知の問題とトラブルシューティング

LifeKeeper for Linux v9.4.1 リリース時点での情報はありません。

# 12.2. Microsoft Azure Quick Start Guide

This guide provides the steps to deploy LifeKeeper for virtual machines and increase their availability. It is based on actual work procedures to help you use SIOS Protection Suite for Linux (LifeKeeper) with Microsoft Azure (Azure) Resource Manager. It will guide you through the steps to configure the following examples of a cluster environment on an Azure VM.

- LifeKeeper cluster (2-node cluster) *Multi-NIC configuration
- Data mirror type shared data area with DataKeeper replication
- IP resources, Oracle resources

**Disclaimer:** This guide outlines a general SIOS Protection Suite for Linux configuration in a Microsoft Azure environment. The setting values used in this guide are listed with an excerpt of the items required for explanation. Items that are not listed are assumed to be default values or arbitrary values. You can configure additional Azure infrastructure options by replacing the setting values described in this guide according to your actual requirements.

## Future Compatibility

The information contained in this document is based on the results of validation performed on Azure as of March 2020. We do not guarantee any changes in the specifications of Azure and LifeKeeper in the future. Please refer to the most recent documentation and make the appropriate settings if necessary.

## Restrictions

- Multi-NIC configurations are supported only on RHEL7 and CentOS7 where “Predictable Network Interface Name” can be used.
- RHEL5, CentOS5, RHEL6, CentOS6 only support single NIC configurations, not multi-NIC configurations.
- Active/Active configurations are not supported.
- Services that use multiple ports or configurations that provide multiple services are not supported.

## Terms

- **Internal Load Balancing or ILB** – This is a function that enables load balancing between virtual machines that reside inside Azure cloud services or virtual networks with limited regions.
- **[VIP protected with IP resources]** – A virtual IP address created with LifeKeeper.
- **[VIP configured by ILB]** – A virtual IP address of the Internal Load Balancing created in Azure.
- **[Azure private IP address]** – A private address used in Azure Virtual Network.

# 12.2.1. Microsoft Azure Overview

Azure is a public cloud service provided by Microsoft that allows users to use hardware, networks, storage (e.g. disks), server operating system (e.g. Windows Server and Linux), middleware (e.g. web servers and RDBMS), groupware and server applications, and application runtime environments (.Net Framework, etc.) on demand through network.

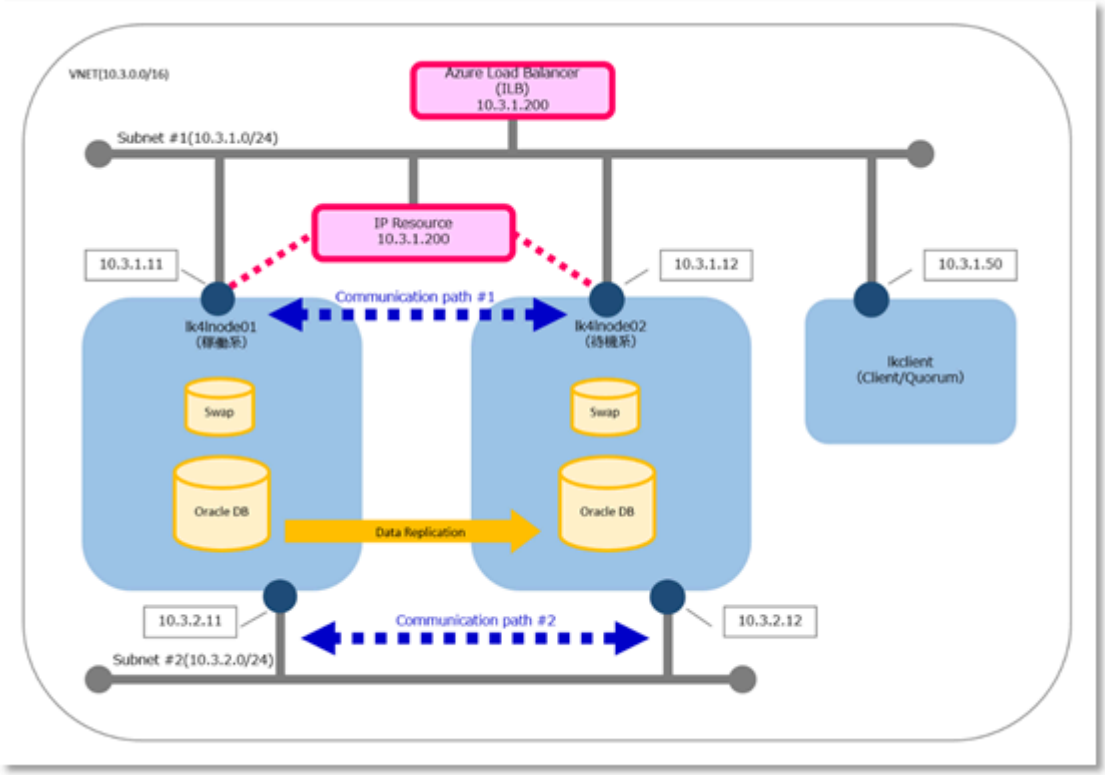
Azure enables a user to easily create, deploy and manage server applications over the Internet in Microsoft-managed data centers around the world. It allows you to use computing resources such as memory and disks when you need them without having to own servers or infrastructure. The pricing is different by resource and is based on an hourly pay-as-you-go system. The web browser-based management portal allows users to operate intuitively.

Azure offers services in four forms:

- Virtual Machines (VMs)
- Websites
- Mobile Services
- Cloud Services

# 12.2.2. 構成情報

本構成では LifeKeeper を使って、以下の構成のアクティブ/スタンバイクラスターを構築します。マルチ NIC 構成の Azure VM に LifeKeeper を導入します。




## サーバー構成

| | |
|---|---|
| 仮想マシンサイズ | クラスターノード: Standard_A1 (1 コア, 1.75 GB メモリ)<br><br>Client/Witness Server: Basic_A0<br><br>(0.25 コア, 0.75GB メモリ)<br><br>*注1 |
| データディスク | 30GiB (Oracle DB 用)<br><br>10GiB (swap 領域用)<br><br>(/dev/disk/cloud/azure_resourceはデータディスクとして使用できません ※注2)<br><br>*注1 |

| | |
|---|---|
| **[IP リソースが保護する VIP]** | 10.3.1.200 |
| **[ILB に設定する IP]** | 10.3.1.200 |
| **ILB が転送するポート** | 1521 |
| **ILB の負荷分散先ホストおよびポート** | lk4lnode01/lk4lnode02 の各 1521 ポート |
| **[Azure のプライベート IP アドレス]** | クラスターノード (稼働系) 10.3.1.11 / 10.3.2.11<br>クラスターノード (待機系) 10.3.1.12 / 10.3.2.12<br>クライアント兼 Witness サーバー 10.3.1.50 |

※注１：Oracle のインストール要件（メモリは１GB、Oracle データベース・インストールディスクサイズ、Swap エリアサイズ）に合うよう、インスタンスサイズおよび仮想マシンのディスクを用意してください。

※注２：ほとんどの Azure VM は一時ディスク (/dev/disk/cloud/azure_resource) を含んでいますが、これは管理された Disk ではありません。一時ディスクは、アプリケーションとプロセスのための短期ストレージを提供し、ページまたはスワップファイルのようなデータのみを格納することを意図しています。一時ストレージは自動的に /mnt にマウントされますが、以下の例では /mnt/resource にマウントされています。azure_resource は /dev/sdb デバイスノードであることが多いですが、設定によっては別のデバイスノードになることもあります。この一時ディスクは、DataKeeper で使用されるストレージのような LifeKeeper で保護されたデバイスとして使用するのには適していません。詳細については、/mnt/DATALOSS_WARNING_README.txt を参照してください。

## ソフトウェア構成

| | |
|---|---|
| **OS** | RedHat Enterprise Linux 7.5 64bit |
| **LifeKeeper** | SIOS Protection Suite for Linux v9.3.2 |
| **Oracle** | Oracle Database 12c Enterprise Edition |

# 監視対象

| | |
|---|---|
| **監視対象** | IP リソースが保護する VIP / ファイルシステム / DataReplication / Oracle DB / Oracle Listener |

## ネットワーク構成

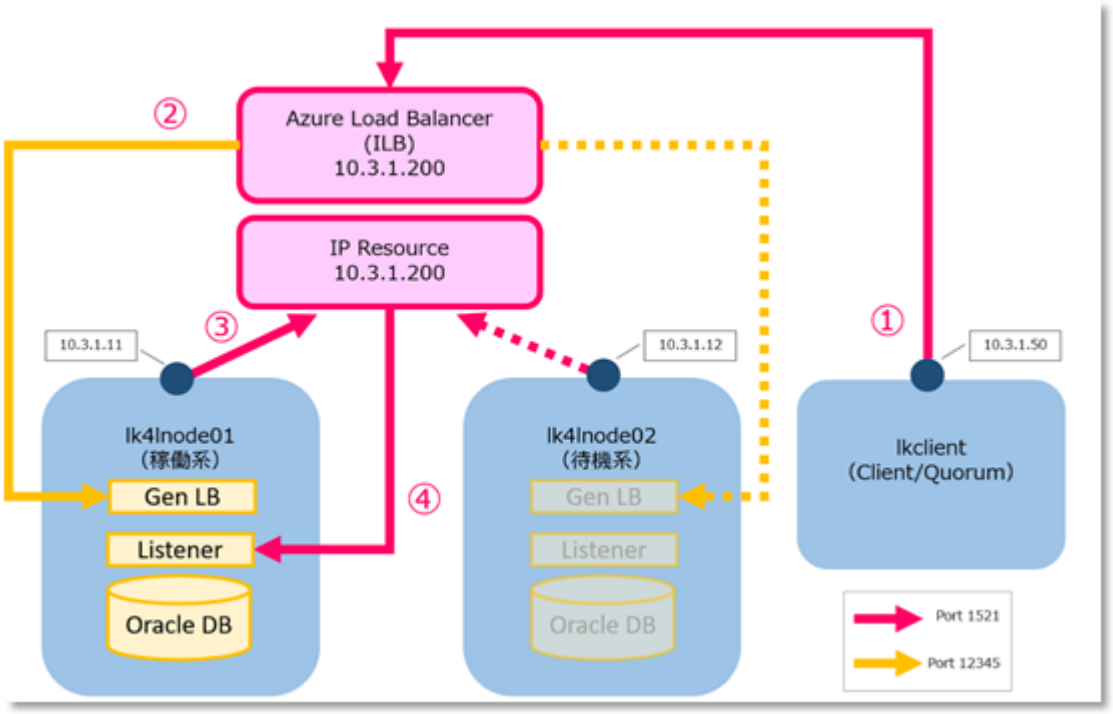
Azure では、仮想マシン（VM）間での通信を行えるようにする為、“仮想ネットワーク”（VNET）を作成します。VNET は、サブネットを指定する事で、このサブネット内で VM間での通信が行えるようにしています。

Azure で VM にネットワークを構成する場合、あらかじめ VNET を作成し、VM にこの VNET を指定するのが一般的です。VNET を使用すると、VM に仮想プライベートネットワーク (VPN) を提供できます。オプションとして、VPN をオンプレミス環境に接続して、ハイブリッドソリューションやクロスプレミスソリューションの実現もできます。VNET は管理ポータルサイトや Azure PowerShell を通して、DNS および IP アドレス範囲の構成を含むネットワークトポロジーを制御できます。

# 12.2.3. LifeKeeper-Specific Configurations in Azure

## Internal Load Balancer (ILB)

Azure is not capable of recognizing [VIP protected with IP resources] with the VNET. Because of this, network communication using [VIP protected with IP resources], which is usually assumed by LifeKeeper for Linux, cannot be performed. Therefore, LifeKeeper introduces ILB as follows with the [VIP set by ILB] set as a network communication path.




1. In order for the Oracle Client to connect to the Oracle Listener, start the connection to 10.3.1.200 (port 1521) via ILB.
2. In the ILB, 10.3.1.11 and 10.3.1.12 are registered in the load balancing destination (backend pool) with no port settings. Packets received by the ILB for 10.3.1.200 (port 1521) are forwarded to both the active and standby nodes.
3. Due to the LifeKeeper specifications, it is necessary to specify [VIP protected with IP resources] for the Oracle Listener resource and an IP resource for receiving needs to be created. Since the LifeKeeper IP resource (10.3.1.200) is in-service only on the active node, only the active node (10.3.1.11) receives requests.
4. As a result, the connection request from the Oracle Client is received by the active Oracle Listener (10.3.1.200: port 1521).

## Address Conversion

LifeKeeper uses PING to external hosts to monitor IP resources, but due to the Azure's specifications, PING with virtual IP as the source cannot be performed. However, by changing the packet source

information as described below, PING can be enabled and network failures between external hosts can be detected.

1. From the IP resource, start ping with the VIP protected by the IP resource as an origin.
2. Replace the PING origin address from 10.3.1.200 (VIP protected by the IP resource) to 10.3.1.11 (Azure private IP address).
3. After replacing the origin, send the packet to the PING destination. If it is replaced with [Azure Private IP address], the response will be returned if the network is reachable.

## Quorum/Witness

In an Azure environment, DataKeeper replication disks are used as an alternative to cluster shared disks. In a cluster that uses DataKeeper, when the network fails such that the communication path is completely disconnected, it may result in a split brain due to its mechanism. To avoid this, configure I/O Fencing using the Quorum/Witness functionality.

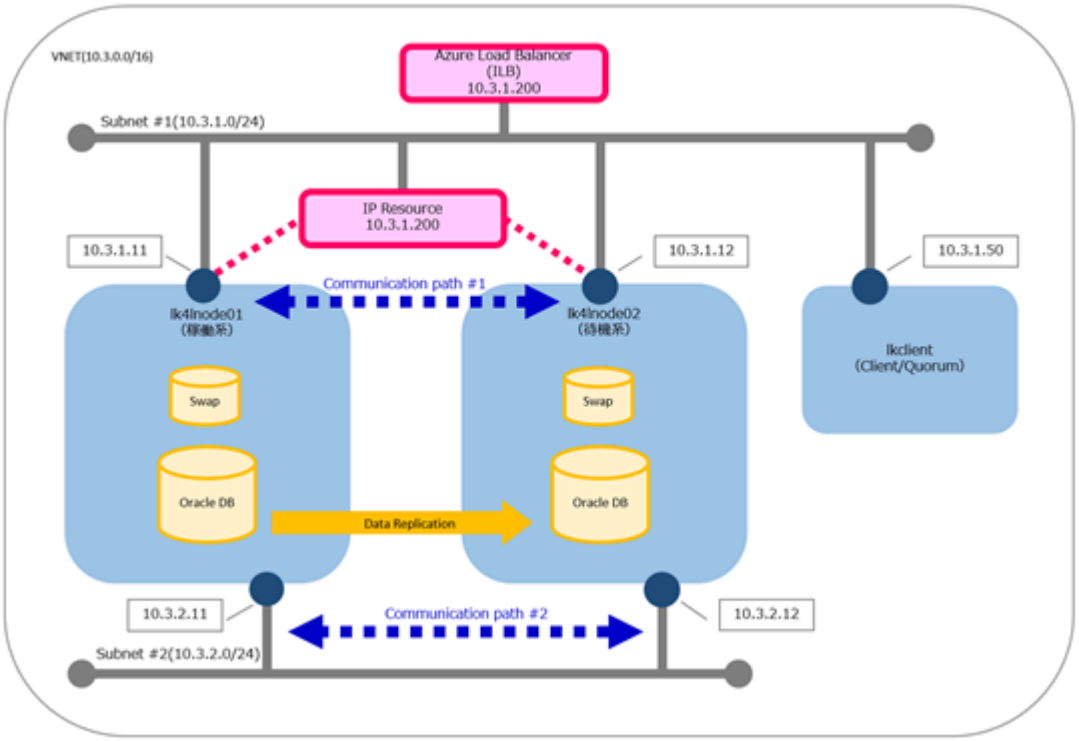
In this verification, we have confirmed successful operations in TCP Remote mode. Refer to Quorum/Witness for more details.

# 12.2.4. Building a Virtual Machine and Starting the OS

## Creating a Virtual Machine and Starting the OS

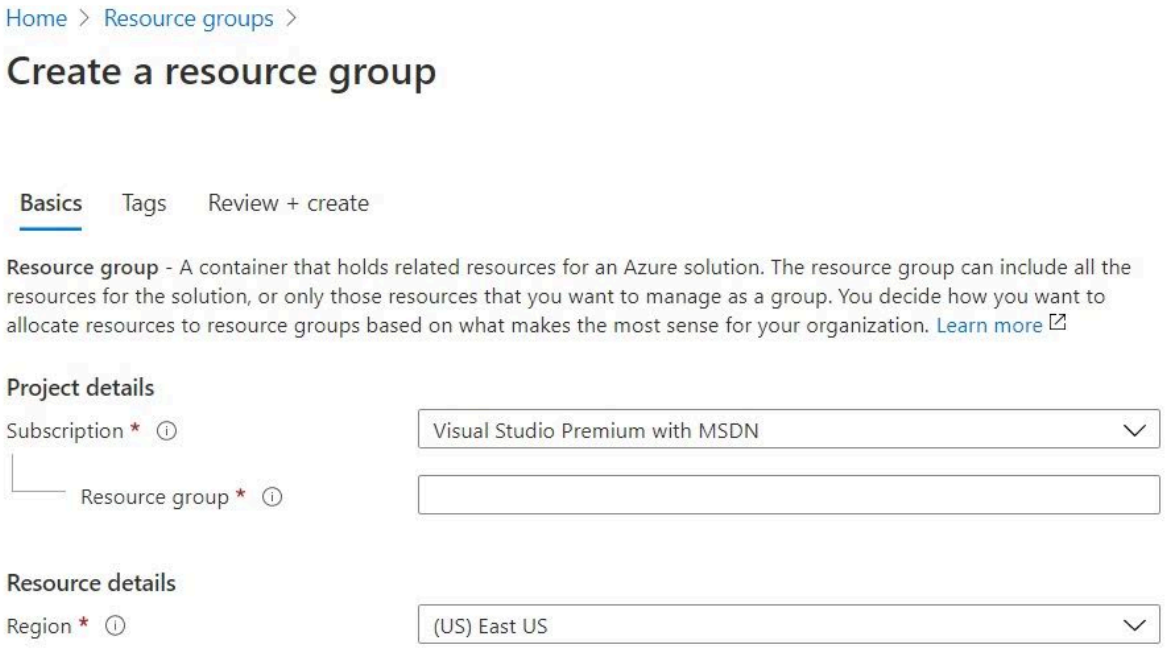
When using LifeKeeper with Azure it is necessary to make the following three settings specific to the Azure environment:

- Use internal load balancing (ILB) to enable connections to Oracle resources protected by LifeKeeper
- For communication using a virtual IP address, configure address conversion
- Setup Quorum to avoid split brain



# 12.2.4.1. Creating a Resource Group

1. Select **Azure Portal** > **Resource groups** > **Add** to create a resource group to be used in this environment. Enter the required values and click **Review + Create**.

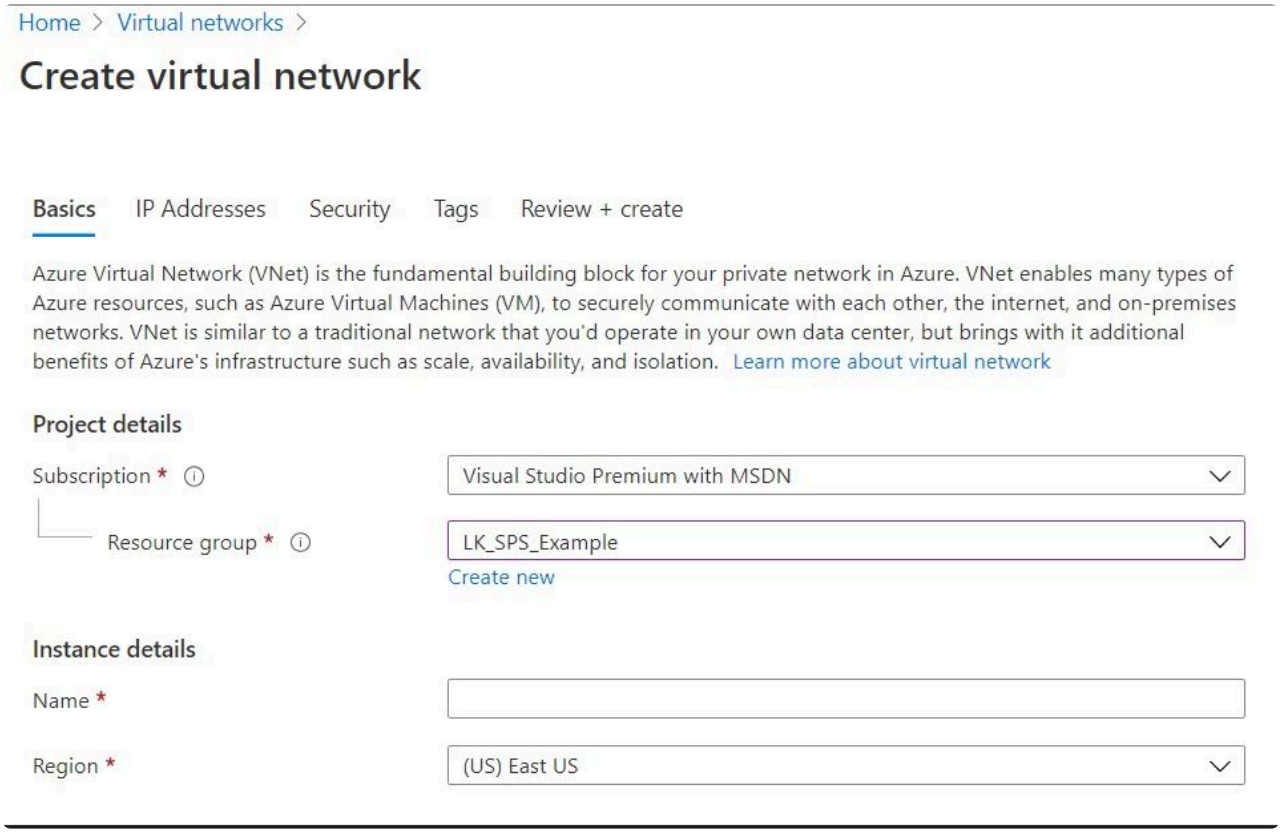



2. Review the details and click **Create**.

# 12.2.4.2. Creating a Virtual Network

Create a virtual network to use in this environment. Create one address space and two subnets, one of these subnets is for services and the other is for DataKeeper replication.

1. Select **Azure Portal** > **Virtual networks** > **Add** to create the virtual network and one subnet at a time for this environment.

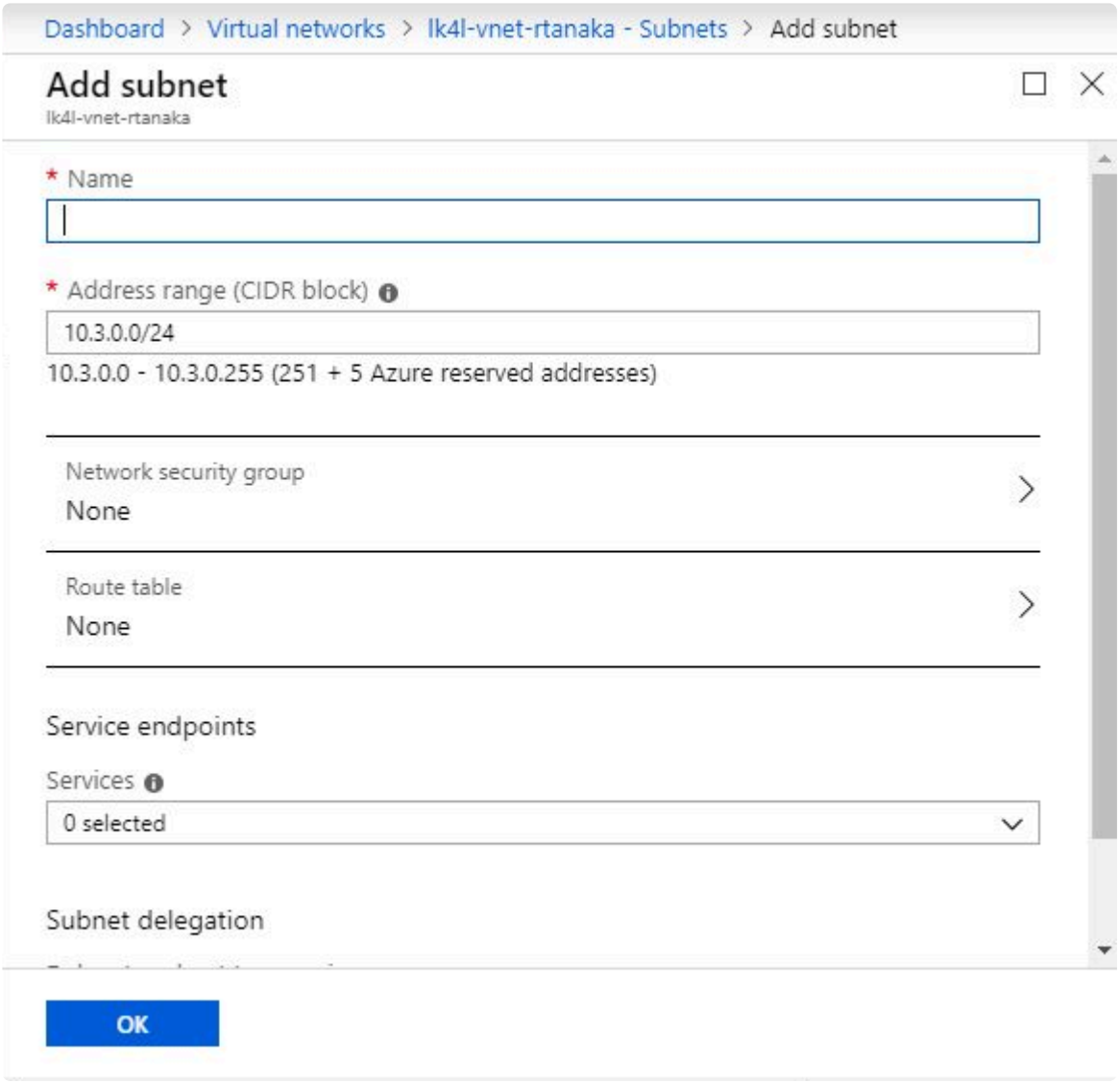



   Enter the required information and click **Create**.

   Use the following values:

| Item | Value to be entered or selected |
|---|---|
| Name | lk4l-vnet |
| Address space | 10.3.0.0/16 |
| **Subnet** | |
| Name | lk4l-nw01 |
| Address range | 10.3.1.0/24 |
| Firewall | Disabled (default) |

2. Select **Azure Portal** > **Virtual networks** > (**Virtual network name**) > **Subnets** > **Add** to add a second subnet for your virtual network.




Enter the required information and click **OK**.

Use the following values:

| Item | Value to be entered or selected |
|---|---|
| Name | lk4l-nw02 |
| Address range | 10.3.2.0/24 |

# 12.2.4.3. Creating a Virtual Machine

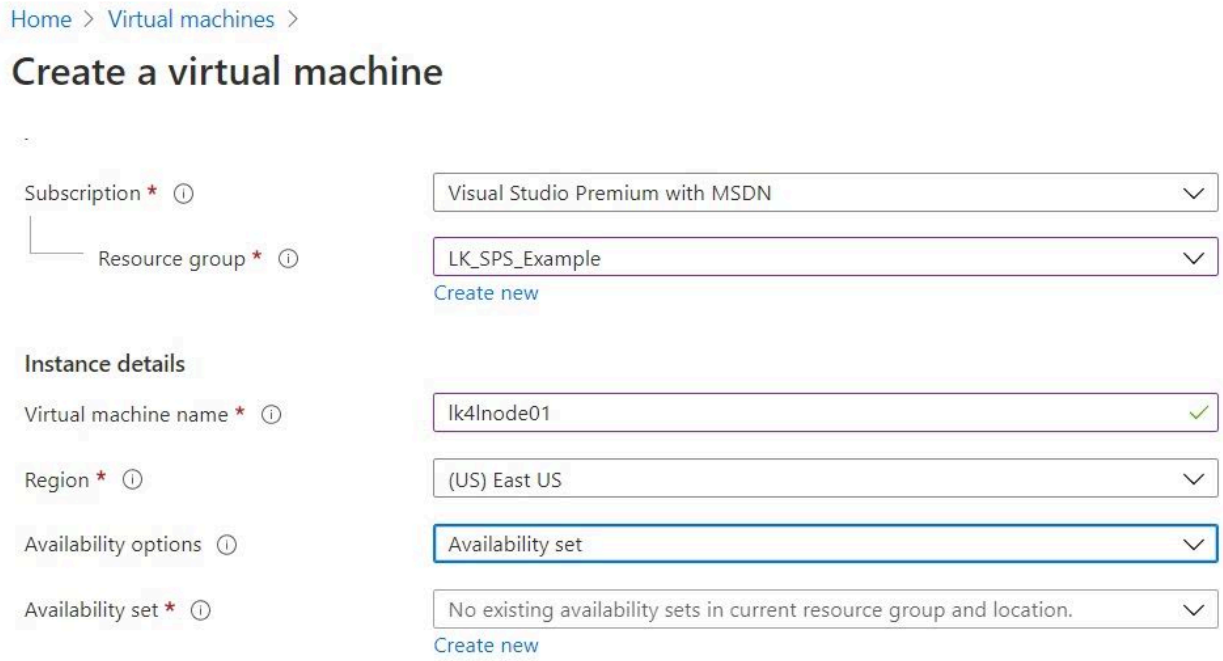
Create a virtual machine to use in this environment.

The following three virtual machines are created.

- Cluster Node (Active)
- Cluster Node (Standby)
- Client and Witness Server

# 12.2.4.3.1. Creating a Cluster Node (Active)

1. Select **Azure Portal** > **Virtual machines** > **Add** and enter the basics settings.



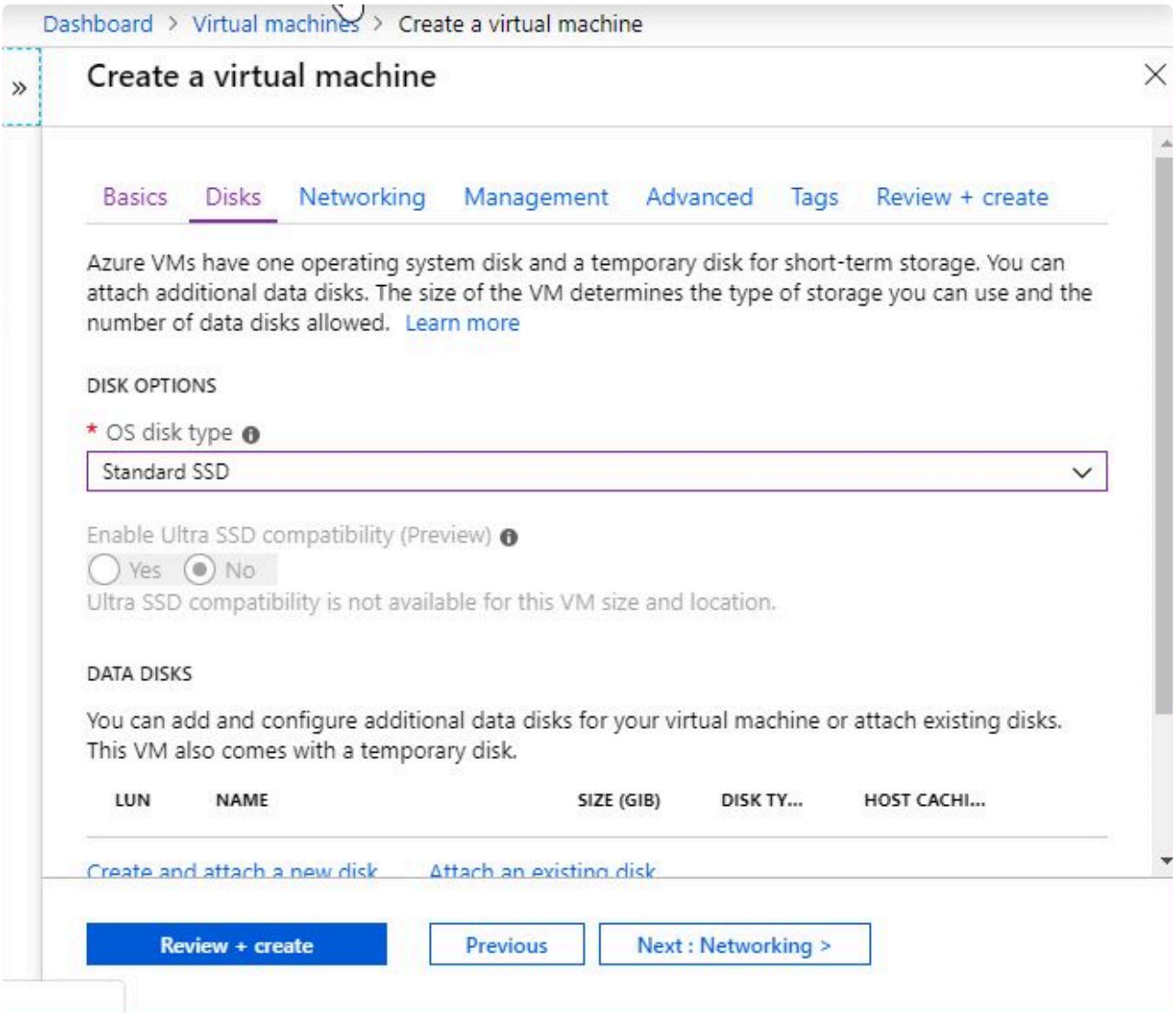

Use the following values:

| **Basics** | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| **INSTANCE DETAILS** | | |
| Virtual machine name | lk4lnode01 | |
| Availability options | Availability set | *Note 1 |
| | (create new)lk4lcluster | |
| Image | Red Hat Enterprise Linux 8.1 | *Note 2 |
| Size | Standard A1 | *Note 3 *Note 4 |
| **ADMINISTRATIVE ACCOUNT** | | |
| Authentication type | Password | |
| Username | lkadmin | |
| Password | XXXXXXXX | |
| **INBOUND PORT RULES** | | |
| Public inbound ports | Allow selected ports | |
| Select inbound ports | SSH | |

- **Note 1:** Availability Sets are a prerequisite for LifeKeeper clusters on Azure and must be set up.
- **Note 2:** Select the OS version supported by LifeKeeper to be installed. Refer to the Support Matrix OS versions supported by LifeKeeper.
- **Note 3:** Be sure to select an instance size for the Availability Set. Availability Sets are not available for some instance sizes.
- **Note 4:** Be sure to select an instance size for the load balancing rules that can be applied. Load balancing rules are not available for some instance sizes.

2. Once the required values are entered click **Next:Disks**.
3. Configure the disk settings. In this case, a disk for Oracle DB (30GiB) and a disk for swap area (10GiB) are required in addition to the OS disk.

   Click **Create and attach a new disk** to assign additional disks.



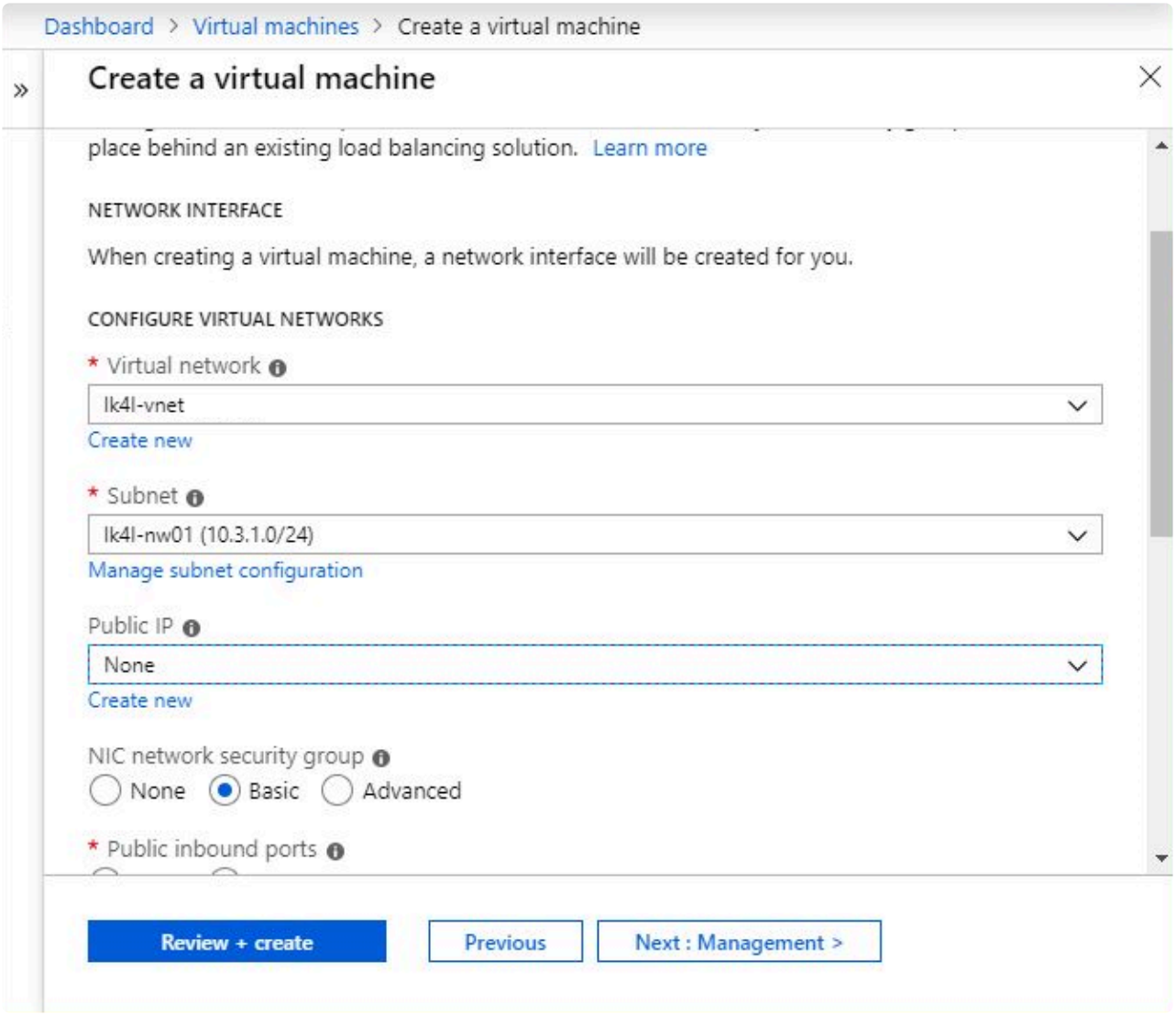
Use the following values:

| Disks | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| **DISK OPTIONS** | | |
| OS Disk type | Standard SSD | |
| **DATA DISKS** | | |
| LUN | 0 | |
| NAME | (create and attach a new disk)<br><br>lklnode01_DataDisk_0 | |
| Size(GB) | 30 | |
| DISK TYPE | Standard SSD | |
| HOST CACHING | None | *Note 1 |
| **DATA DISKS** | | |
| LUN | 1 | |
| NAME | (create and attach a new disk)<br><br>lklnode01_DataDisk_1 | Any value |
| Size(GB) | 10 | |
| DISK TYPE | Standard SSD | |
| HOST CACHING | None | *Note1 |

- **Note 1:** In this case, the cache is disabled for database use and should be configured appropriately according to your requirements.

4. Once the required values are entered click **Next:Networking**.

5. Enter the network settings.




At this time, configure the first subnet that you created in the previous step for the virtual machine. This setting attaches a single network interface. The second network interface is configured after the virtual machine is created.

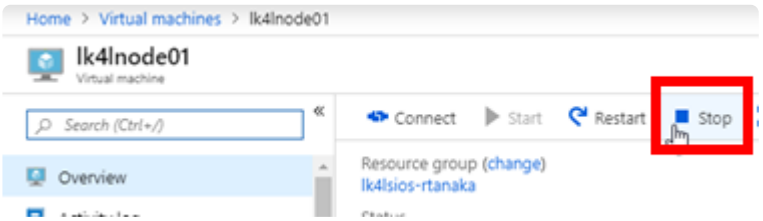
Use the following values:

| **Networking** | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| Virtual network | lk4l-vnet | |
| Subnet | lk4l-nw01(10.3.1.0/24) | *Note 1 |
| Public IP | (create new)<br><br>Name:lk4lnode01pubip<br><br>SKU:Basic<br><br>Assignment:Static | *Note 2 |

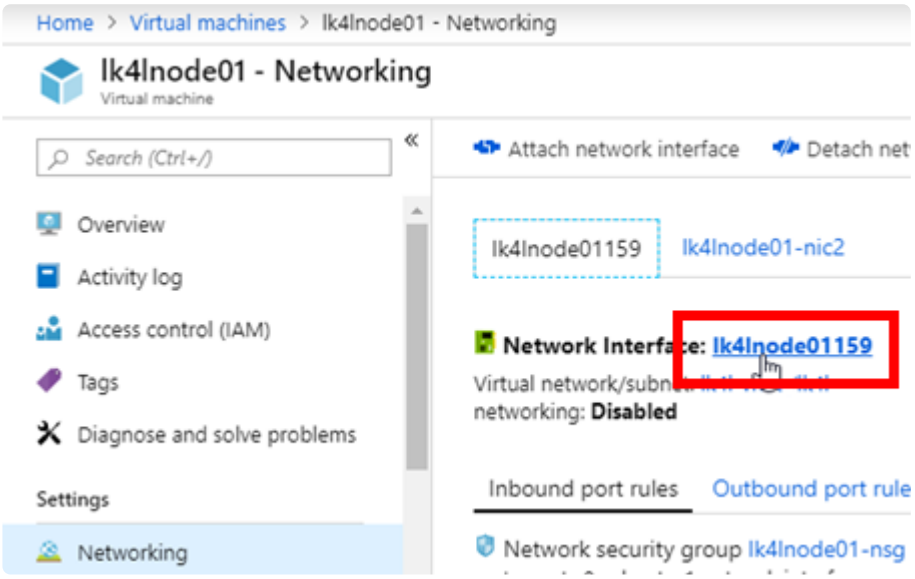
| | | |
|---|---|---|
| NIC network security group | Basic | |
| Public inbound ports | Allow selected ports | |
| Selected inbound ports | SSH | |

- **Note 1:** Configure the first subnet that you created in the previous step for the virtual machine.
- **Note 2:** In this procedure the Public IP has been set to access via the Internet.

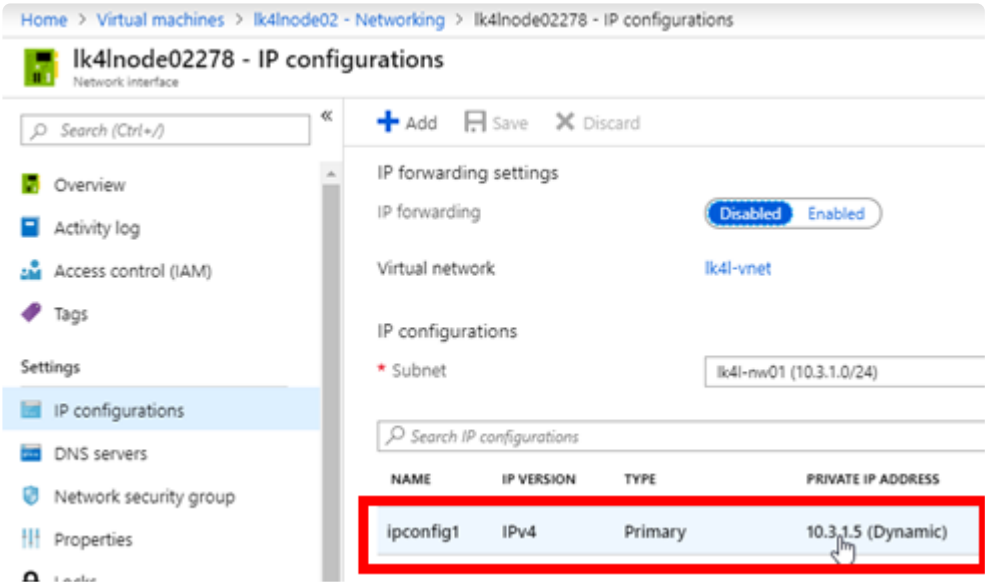
6. Configure administrative settings, advanced settings, and tag settings if necessary, then click **Review + Create**.
7. Review the details and click **Create**. When deployed with the above configuration, Red Hat Enterprise Linux starts with the installation completed.
8. Stop the virtual machine to change the network configuration.



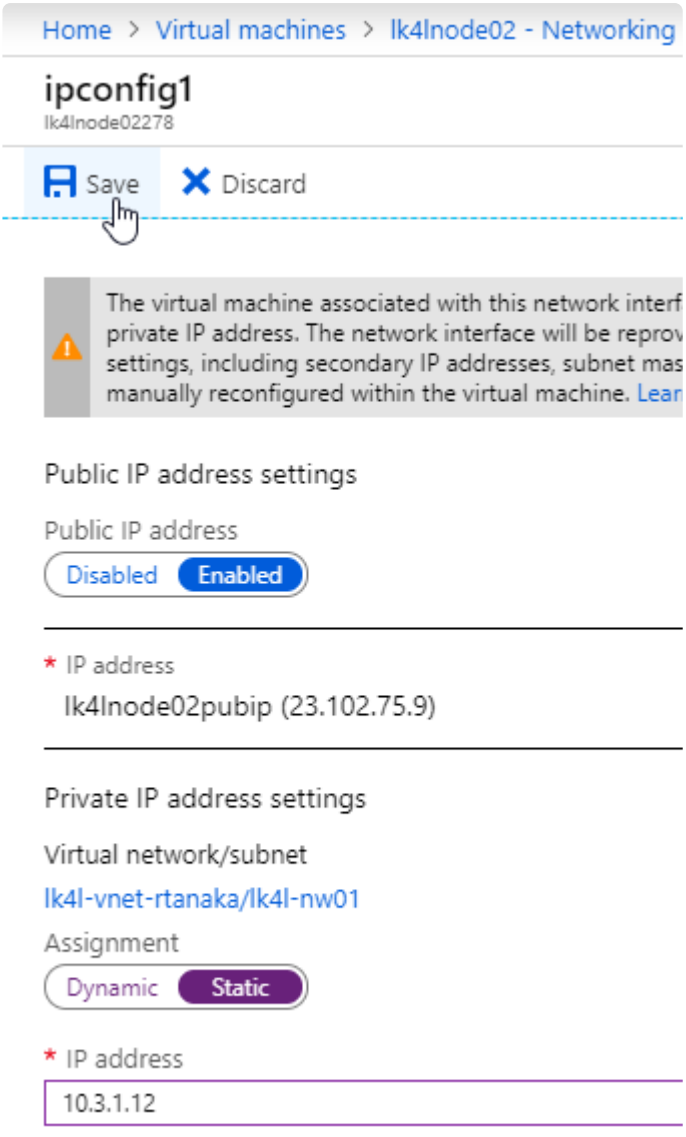

9. Change the first network interface to a static IP address. Click **Azure Portal** > **Virtual machines** > **(virtual machine name)** > **Networking** > **(first interface name)**.




10. Click **IP configurations** > **ipconfig1**. At this point the IP address is set to Dynamic.




11. Change the IP address setting to static.

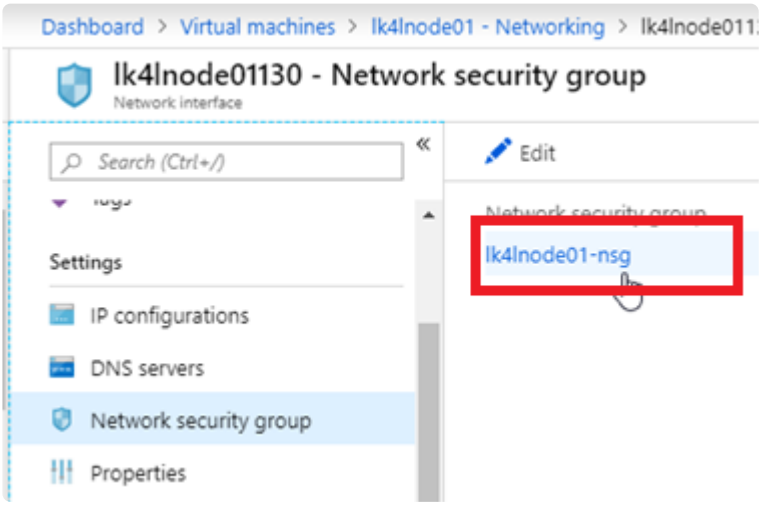



Use the following values:

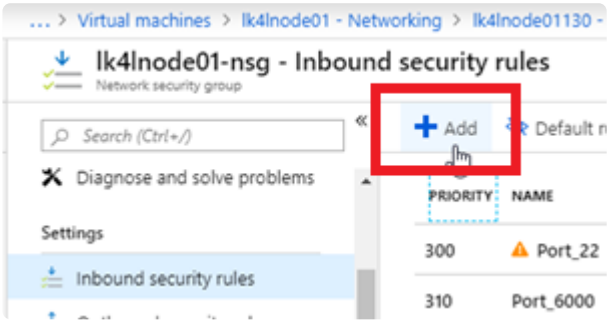
| Item | Value to be entered or selected | Notes |
|---|---|---|
| Assignment | Static | |
| IP address | 10.3.1.11 | |

12. Click **Save** to save the settings.

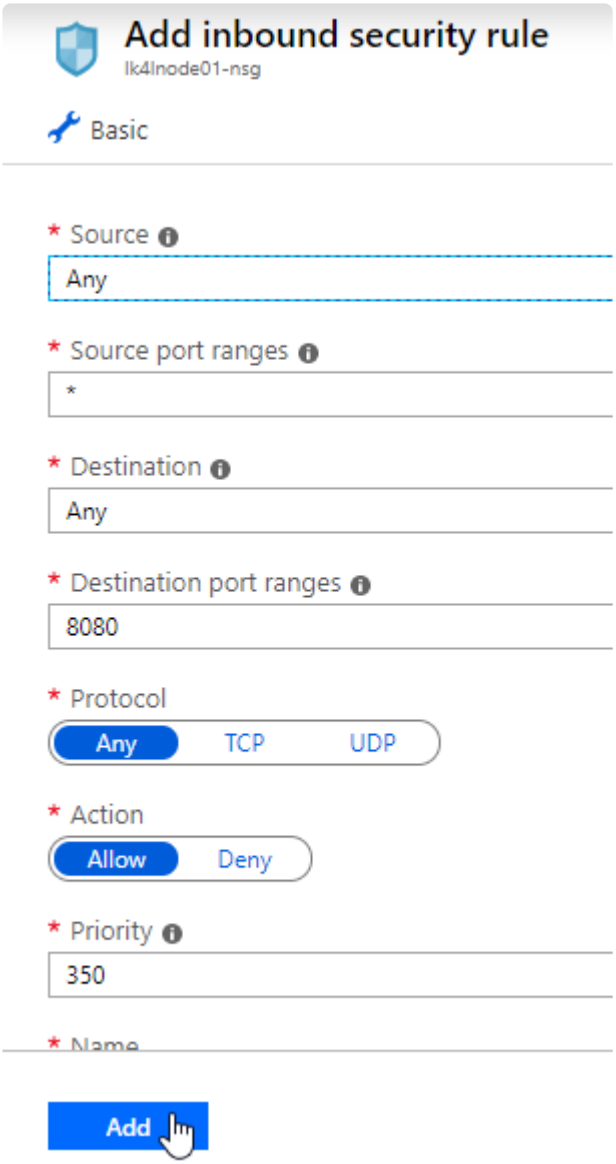
13. Edit the network security group to allow the Oracle listener service to communicate using TCP Port 1521. Click **Azure Portal** > **Virtual machines** > **(virtual machine name)** > **Networking** > **(first interface name)** > **Network security group** > **(security group name)**.




14. Click **Inbound security rules** > **Add**.




15. Add inbound rules.



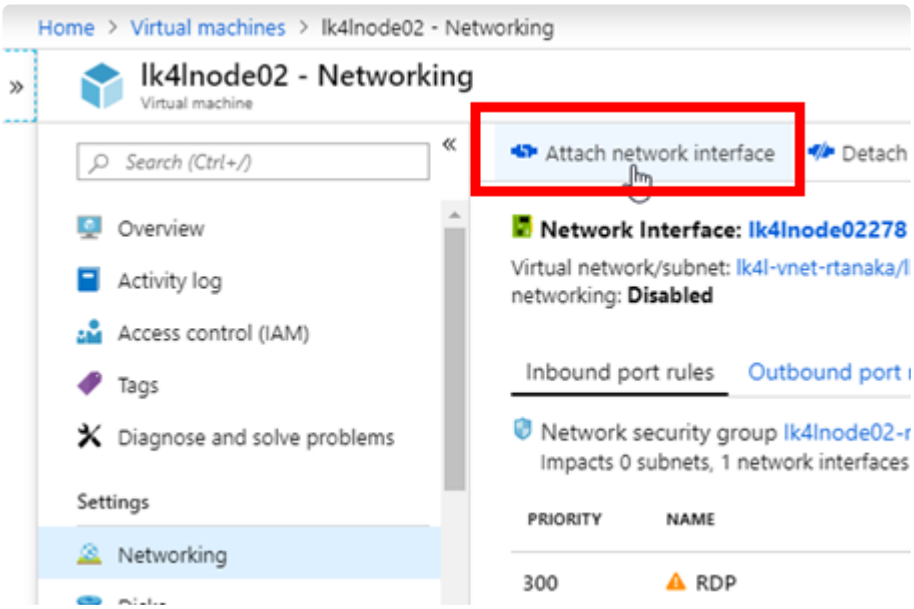

Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Source | Any | |
| Source port ranges | • | |
| Destination | Any | |
| Destination port ranges | 1521 | *Note 1 |
| Protocol | TCP | |
| Action | Allow | |
| Name | Port_1521 | |

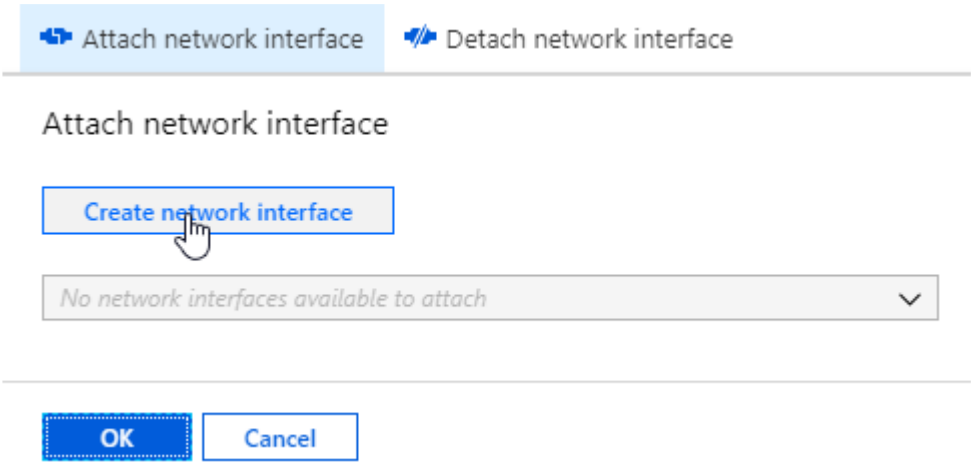
- **Note 1:** Since the Oracle listener uses port1521, communication to the relevant port is permitted.

16. Click **Add** to save your settings.

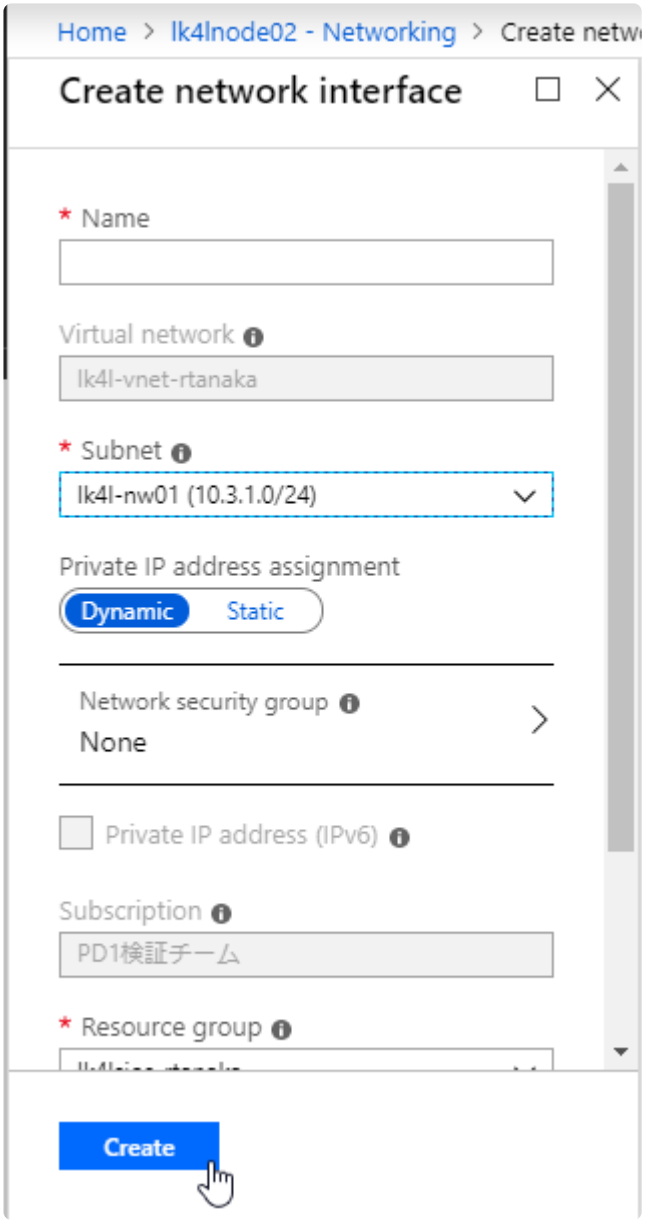
17. Next, add a second network interface. Click **Azure Portal** > **Virtual machines** > **(Virtual machine name)** > **Networking** > **Attach network interface**.




18. Click **Create network interface**.




19. After entering the required values, click **Create**.



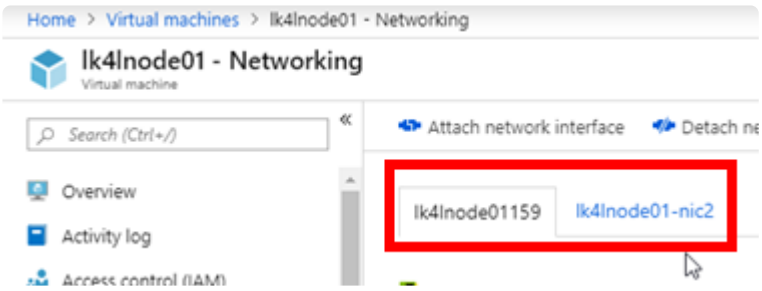

Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Name | lk4lnode02-nic2 | |
| Subnet | 10.3.2.0/24 | *Note 1 |
| Private IP address assignment | Static | |
| Private IP address | 10.3.2.11 | |
| Network security group | None | |

- **Note 1:** Set the second subnet created in the previous step for the virtual machine.

20. Click **OK**.

21. Make sure that the two interfaces are attached.




22. Start the virtual machine.

# 12.2.4.3.2. Creating a Cluster Node (Standby)

1. Follow the same procedure to create a virtual machine that was used for the cluster nodes (standby node).

   Use the following values for basic settings:

| **Basics** | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| **INSTANCE DETAILS** | | |
| Virtual machine name | lk4lnode02 | |

   All other settings should be the same as the active node.

   Use the following values for disk settings:

| **Disks** | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| **DATA DISKS** | | |
| LUN | 0 | |
| NAME | (create and attach a new disk)<br>lklnode02_DataDisk_0 | |
| **DATA DISKS** | | |
| LUN | 1 | |
| NAME | (create and attach a new disk)<br>lklnode02_DataDisk_1 | Any value |

   All other settings should be the same as the active node.

   Use the following values for network settings:

| **Networking** | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| Public IP | (create new) | *Note 1 |

| | | |
|---|---|---|
| | Name:lk4lnode02pubip<br><br>SKU:Basic<br><br>Assignment:Static | |

**Note 1:** In this procedure the Public IP has been set to access via the Internet.

All other settings should be the same as the active node.

2. Make administrative settings, advanced settings, and tagging if necessary, and then click **Review + Create**.
3. Review the details and click **Create**.
4. Since the network configuration will be changed after this, stop the virtual machine.
5. Change the first network interface to a static IP address. Click **Azure Portal** > **Virtual machines** > **(Virtual machine name)** > **Networking** > **(First interface name)**.
6. Click **IP configurations** > **ipconfig1**. At this point the IP address is set as Dynamic.
7. Set the IP address to Static.

   Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Assignment | Static | |
| IP address | Static 10.3.1.12 | |

8. Click **Save** to save the settings.
9. Next, add a second network interface. Click **Azure Portal** > **Virtual machines** > **(Virtual machine name)** > **Networking** > **Attach network interface**.
10. Click **Create network interface**.
11. Click **Create** after entering the required values.

    Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Name | lk4lnode02-nic2 | |
| Subnet | 10.3.2.0/24 | *Note 1 |

| | | |
|---|---|---|
| Private IP address assignment | Static | |
| Private IP address | 10.3.2.12 | |
| Network security group | None | |

- **Note 1:** Set the second subnet created in the previous step for the virtual machine.

12. Click **OK**.
13. Make sure that the two interfaces are attached.
14. Start the virtual machine.

# 12.2.4.3.3. Creating a Client and Witness Server

1. Follow the same procedure that was used for the cluster nodes (standby node).

   Use the following values for basic settings:

| **Basics** | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| **INSTANCE DETAILS** | | |
| Virtual machine name | lk4lclient | |
| Availability options | No Infrastructure redundancy required | *Note 1 |
| Image | Red Hat Enterprise Linux 7.5 | |
| Size | Basic AO | |
| **INBOUND PORT RULES** | | |
| Public inbound parts | Allow selected ports | |
| Select inbound ports | SSH | |

- **Note 1:** It is not necessary to set the Availability Set except for the cluster nodes. If the client is used for Quorum you should consider including it in the Availability Set for high availability.

   After entering the required values, click **Next: Disks**.

2. Enter the disk settings.

   A disk in not added at this time. Click **Next: Networking**.

3. Enter the network settings.

   Use the following values:

| **Networking** | | |
|---|---|---|
| **Item** | **Value to be entered or selected** | **Notes** |
| Virtual network | lk4l-vnet | |
| Subnet | lk4l-nw01(10.3.1.0/24) | *Note 1 |
| Public IP | (create new) | *Note 2 |

| | | |
|---|---|---|
| | Name:lk4lclientpubip<br><br>SKU:Basic<br><br>Assignment:Static | |
| NIC network security group | Basic | |
| Public inbound ports | Allow selected ports | |
| Selected inbound ports | SSH | |

- **Note 1:** Set the first subnet created in the previous step for the virtual machine.
- **Note 2:** In this procedure, the Public IP is set because it is accessed via the Internet.

4. Make administrative settings, advanced settings, and tagging if necessary, and then click **Review + Create**.
5. Review the details and click **Create**.

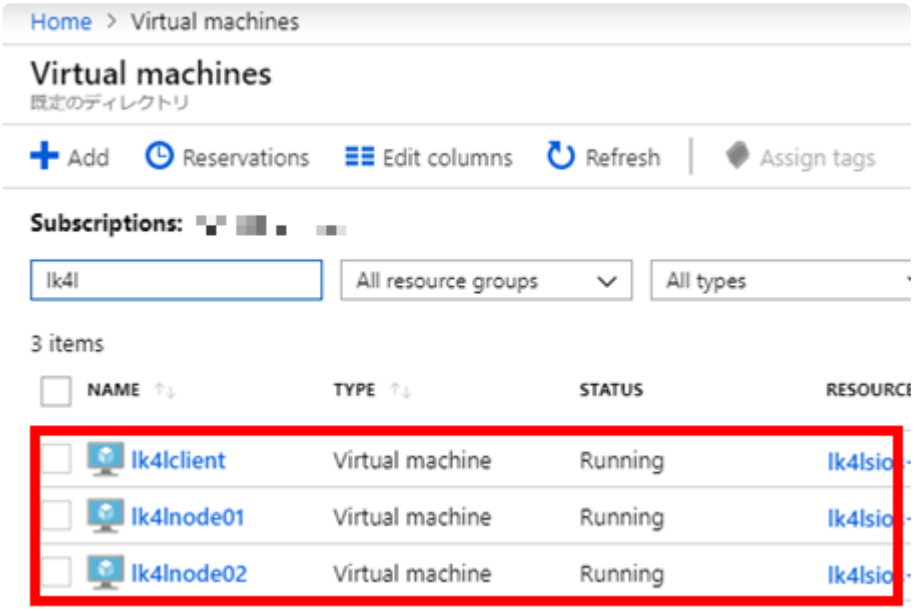
   After the virtual machine is started, change the first network interface to a static IP address. Click **Azure Portal** > **Virtual machines** > **(virtual machine name)** > **(Networking)** > **(first interface name)**.
6. Click **IP configurations** > **ipconfig1**. At this point the IP address is set to Dynamic.
7. Change the setting of the IP address to Static.

   Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Assignment | Static | |
| IP address | 10.3.1.50 | |

8. Click **Save** to save the settings.

   Now three virtual machines have been created.


Home > Virtual machines
Virtual machines
既定のディレクトリ
Add
Reservations
Edit columns
Refresh
Assign tags
Subscriptions:
lk4l
All resource groups
All types
3 items
NAME
TYPE
STATUS
RESOURCE
lk4lclient
Virtual machine
Running
lk4lnode01
Virtual machine
Running
lk4lnode02
Virtual machine
Running

# 12.2.4.4. Creating a Load Balancer

On Azure the virtual IP created by LifeKeeper cannot be used for communication. A load balancer is required for the virtual IP to work.

There are two types of Azure load balancers:

- External load balancer
- Internal load balancer

Here we will use the internal load balancer. The IP address of this internal load balancer is also used as the virtual IP address that will be set later. It will become the access point of the protected services.

1. Select **Azure Portal** > **Load balancers** > **Add** to create a load balancer used in this environment. Enter the required values and click **Review + Create**.

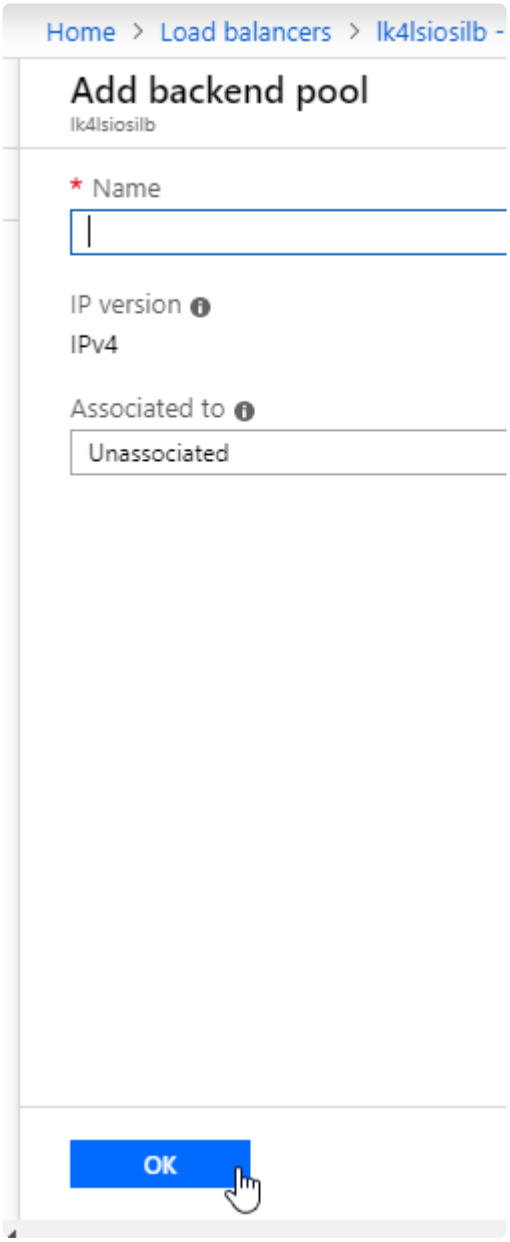
   Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| **INSTANCE DETAILS** | | |
| Name | lk4lsiosilb | |
| Type | Internal | *Note 1 |
| SKU | Basic | |
| **PUBLIC IP ADDRESS** | | |
| Virtual network | lk4l-vnet | |
| Subnet | lk4l-nw01(10.3.1.0/24) | |
| IP address assignment | Static | |
| Private IP address | 10.3.1.200 | *Note 2 |

- **Note 1:** Be sure to select “Internal” to create an internal load balancer.
- **Note 2:** This value is also used as the value of the virtual IP address that will be set later. It becomes the access point of the protected services.

2. Review the details and click **Create**.
3. Next, add two virtual machines for the created cluster to the backend pool of the internal load balancer. Select **Azure Portal** > **Load balancers** > **(load balancer name)** > **Backend pools** > **Add**.

lk4l
All resource gr
1 items
NAME
RESOURC
lk4lsiosilb
lk4lsios
Home > Load balancers > lk4lsiosilb - Backend pools
lk4lsiosilb - Backend pools
Load balancer
Search (Ctrl+/)
Add
Refresh
Search backend addres
Access control (IAM)
Tags
Diagnose and solve problems
VIRTUAL MACHINE
No results.
Settings
Frontend IP configuration
Backend pools



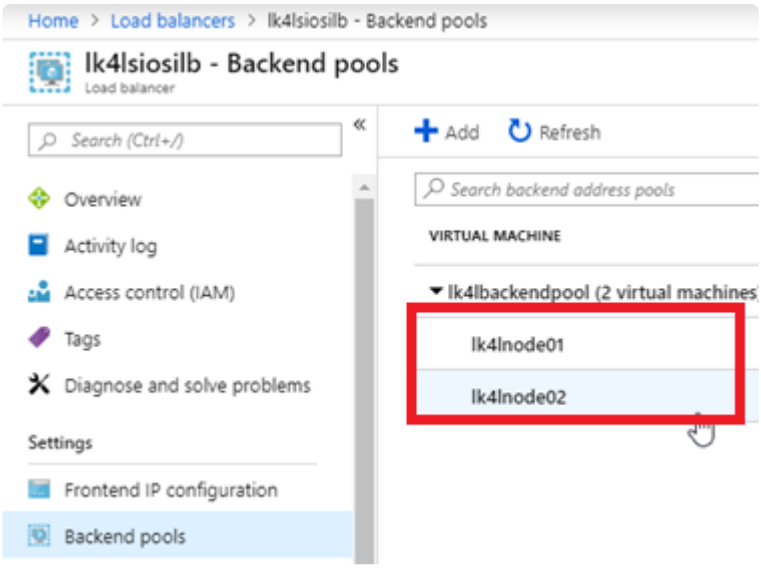

Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Name | lk4lbackendpool | |
| Associated to | Availability set | |
| Availability set | lk4lcluster | *Note 1 |
| **Target #1** | | |
| Target virtual machine | lk4lnode01 | |
| Network IP configuration | ipconfig1(10.3.1.11) | *Note 2 |
| **Target #2** | | |
| Target virtual machine | lk4lnode02 | |
| Network IP configuration | ipconfig1(10.3.1.12) | *Note 2 |

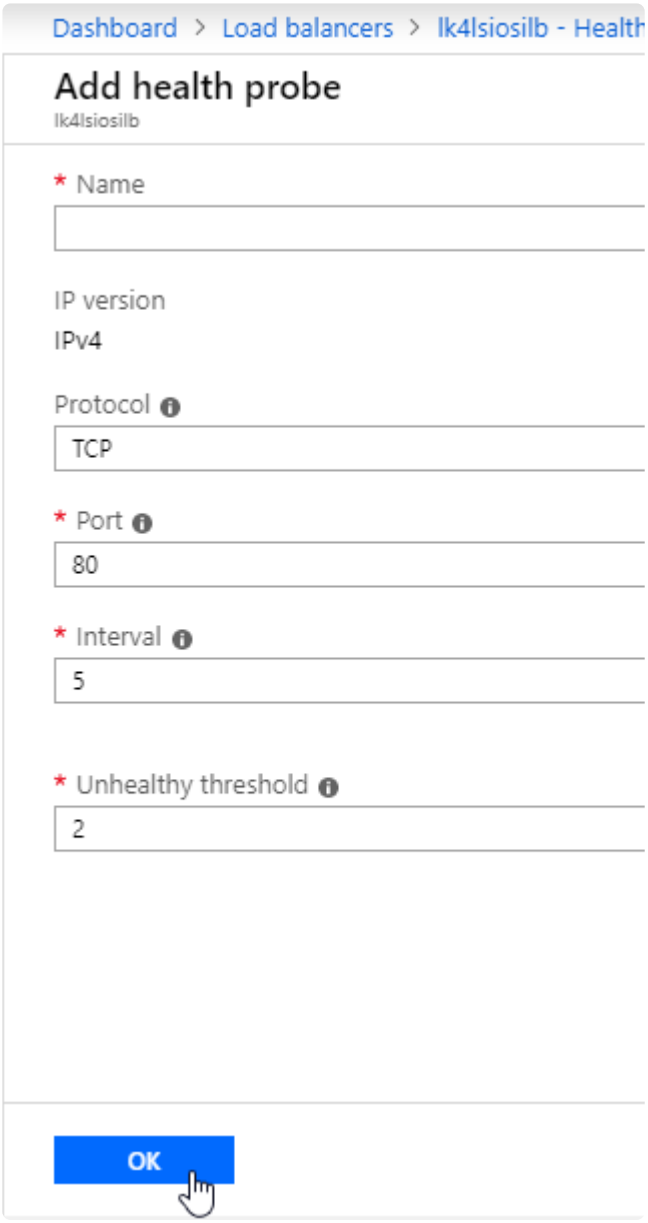
- **Note 1:** Specify the created Availability Set.
- **Note 2:** Associate the backend pool with the primary interface of the virtual machine in the created Availability set.

Enter the required values and click **OK**.

4. Verify that the backend pool has been created.




5. Configure the probe. Click **Azure Portal** > **Load balancers** > **(load balancer name)** > **Health Probes** > **Add**.

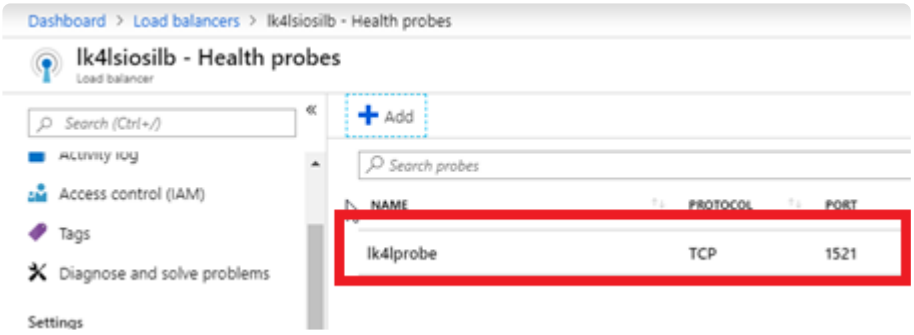



Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Name | lk4lprobeport | |
| Protocol | TCP | *Note 1 |
| Port | 1521 | *Note 1 |

- **Note 1:** For the probe, specify the protocol and port used by the Oracle listener.

6. Enter the required values and click **OK**.
7. Verify that the probe has been created.




8. Configure the load balancing rules. Click **Azure Portal** > **Load balancers** > **(load balancer name)** > **Load balancing rules** > **Add**.

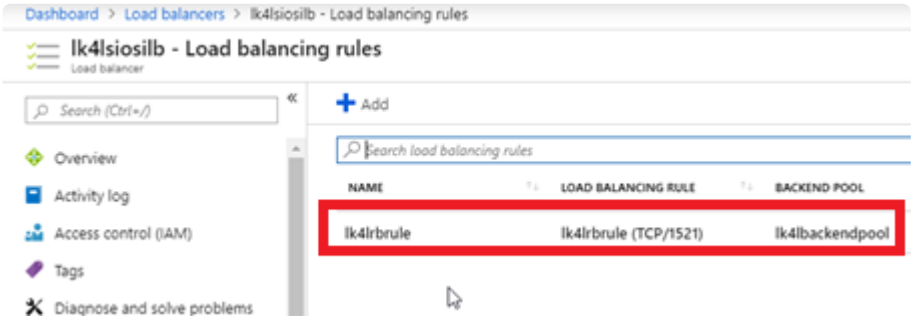
Use the following values:

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Name | lk4lrbrule | |

| | | |
|---|---|---|
| Protocol | TCP | |
| Port | 1521 | *Note 1 |
| Backend port | 1521 | *Note 1 |
| Backend pool | lk4lbackendpool | |
| Health probe | lk4lprobe | |
| Floating IP address | Enabled | *Note 2 |

- **Note 1:** For the port for load balancing, specify the port used by the Oracle listener.
- **Note 2:** Since the communication from the client to the cluster must be a packet communication to an IP resource (VIP), the Floating IP address must be enabled.

9. Enter the required values and click **OK**.
10. Verify that the load balancing rules are created.




Now the load balancer has been configured.

# 12.2.4.5. Configuring the OS

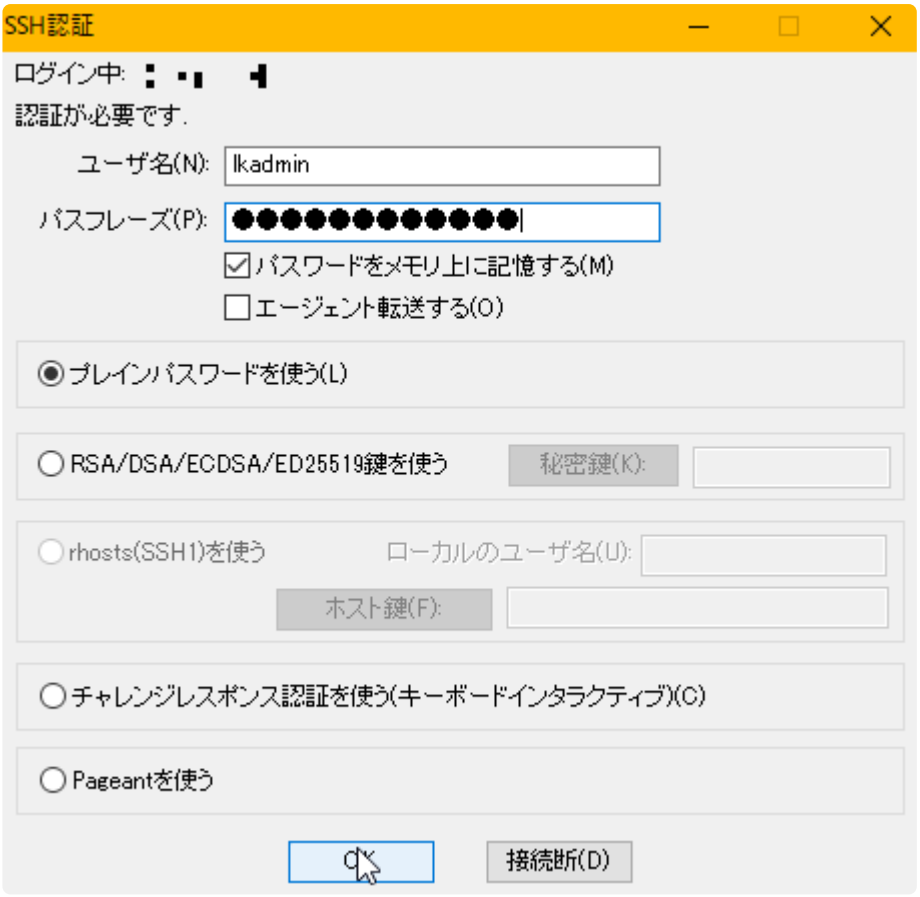
Prior to installing LifeKeeper, configure the OS for the three virtual machines that you have created.

## Logging In to a Virtual Machine

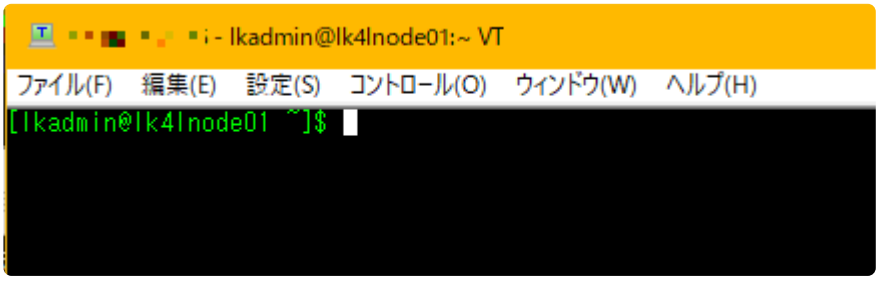
Perform the following steps on **both cluster nodes and clients**.

1. Select **Azure Portal** > **Virtual machines** > **Overview** > **Connect** and review the information required for the connection.

2. Using terminal software (Tera Term etc.) on the local PC, SSH to the public IP of the virtual machine with the account created when creating the virtual machine.




3. Log in to the virtual machine.



4. Set it up for subsequent use by root. Change to root privileges.

```
$ sudo su -
```

5. Set the root password.

```
# passwd
```

6. Allow root login.

```
# cp /etc/ssh/sshd_config /etc/ssh/sshd_config.org
# vi /etc/ssh/sshd_config
< omitted >
PermitRootLogin yes    //Uncomment
< omitted >
#systemctl restart sshd
```

7. Launch another window with the terminal software and confirm that you can log in as root (**do not close the terminal window that you have been using**).

8. After successfully logging in, perform all subsequent tasks as root.

# Fixing NIC Addresses

The NIC address is fixed to prevent it from being updated when the Azure infrastructure is updated. Make the following settings on **both cluster nodes**.

1. Display the NIC information of the virtual machine and take a note of each interface name and MAC address.

```
#ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.3.1.11  netmask 255.255.255.0  broadcast 10.3.1.255
        inet6 fe80::20d:3aff:fe50:42bf  prefixlen 64  scopeid 0x20<link>
        ether 00:0d:3a:50:42:bf  txqueuelen 1000  (Ethernet)
<omitted>
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.3.2.11  netmask 255.255.255.0  broadcast 10.3.2.255
        inet6 fe80::e3c9:ec16:62a:bbd0  prefixlen 64  scopeid 0x20<link>
        ether 00:0d:3a:51:fc:01  txqueuelen 1000  (Ethernet)
<omitted>
```

2. By default, there is only a network configuration file for eth0, so create a network configuration file for eth1.

```
#cp /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth1
```

3. Edit the eth0 and eth1 configuration files. For HWADDR, use the value you took a note in the

previous step.

```
#vi /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
ONBOOT=yes
<omitted>
HWADDR=00:0d:3a:50:42:bf      //Add

#vi /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1    //change
ONBOOT=yes
<omitted>
HWADDR=00:0d:3a:51:fc:01      //Add
```

# SELinux

Disable SELinux. Make the following settings on **both cluster nodes**.

1. Open the following file.

```
# vi /etc/selinux/config
```

2. Change the following parameters from “enforcing” to “disabled” and save.

```
SELINUX=disabled
```

3. Reboot the virtual machine for the settings to take effect.
4. Confirm that SELinux is disabled.

```
# getenforce
Disabled
```

# Firewall

Since LifeKeeper uses a specific port, refer to Running LifeKeeper with a Firewall to change the OS firewall configuration.

Disable the firewall. Configure the following settings on **both cluster nodes**.

- Execute the following commands.

```
# systemctl stop firewalld.service
# systemctl disable firewalld.service
```

# Address Conversion

Configure the following settings at the same time.

- To enable communication to the VIP, the destination address of the packet for port 1521 where the load is balanced by ILB is converted to [VIP protected with IP resources].
- Convert the source address of the ICMP packet to [Private IP address of Azure] so that the monitoring process of IP resources can be performed successfully.

Make the following settings on **both cluster nodes**.

1. Install the IP tables package.

```
# yum -y install iptables-services
```

2. Enable IP tables.

```
#systemctl start iptables.service
#systemctl enable iptables.service
```

3. Make sure the IP tables are enabled.

```
#systemctl list-unit-files | grep iptables
iptables.service                                enabled
```

4. Configure the IP tables.

   Execute the following command on the active node (lk4lnode1):

```
# iptables -t nat -A PREROUTING -p tcp --dport 1521 -j DNAT --to-destination
10.3.1.200:1521
# iptables -t nat -A POSTROUTING -p icmp -s 10.3.1.200 -j SNAT --to-source 10.3.1.11
# service iptables save
```

   Execute the following command on the standby node (lk4lnode02):

```
# iptables -t nat -A PREROUTING -p tcp --dport 1521 -j DNAT --to-destination
10.3.1.200:1521
# iptables -t nat -A POSTROUTING -p icmp -s 10.3.1.200 -j SNAT --to-source 10.3.1.12
# service iptables save
```

5. Confirm that the ip table settings have been added. The following is an example of the active node.

```
#cat /etc/sysconfig/iptables
<omitted>
:POSTROUTING ACCEPT [16:960]
-A PREROUTING -p tcp -m tcp --dport 1521 -j DNAT --to-destination 10.3.1.200:1521
-A POSTROUTING -s 10.3.1.200/32 -p icmp -j SNAT --to-source 10.3.1.11
<omitted>
```

6. Reboot the virtual machine for the settings to take effect.

# GUI Connection Settings

By default, Azure virtual machines do not have a GUI environment installed. We need to set up an X11 environment on the cluster nodes and a GUI environment on the client to use the LifeKeeper GUI.

Make the following settings on **both cluster nodes**.

7. Install the GUI package. Execute the following command.

```
# yum -y groupinstall “X11”
```

8. After installing the package, execute the following command.

```
# systemctl set-default graphical.target
```

9. Next, set the permission for intra-subnet communication. Edit the following file and add an entry before [REJECT].

```
# vi /etc/sysconfig/iptables
<omitted>
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [4710:957955]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A INPUT -s 10.3.1.0/24 -p tcp -j ACCEPT      //Add
-A INPUT -s 10.3.2.0/24 -p tcp -j ACCEPT      //Add
-A INPUT -j REJECT --reject-with icmp-host-prohibited
<omitted>
```

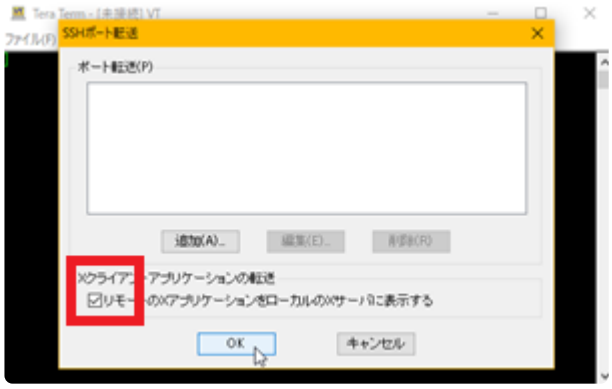
10. Reboot the virtual machine for the settings to take effect.

```
#  reboot
```

11. Configure the local PC. Install the X Server software (such as Xming) on your local PC and start it.

12. Next, configure the X 1 1 forwarding in the local PC terminal software (such as TeraTerm).

The setting method varies depending on the terminal software you are using. The following is an example of the configuration in TeraTerm.




# Name Resolution

Configure the following settings on **both cluster nodes and clients**.

1. Register the information of the cluster node (active) and cluster node (standby) clients and three Witness servers in */etc/hosts*.

```
#vi /etc/hosts
10.3.1.11 lk4lnode01    //Add
10.3.1.12 lk4lnode02    //Add
10.3.1.50 lkclient    //Add
```

2. Verify that each of them can communicate with each other with host name.

```
[root@lk4lnode01 ~]# ping lk4lnode02 -c 4
[root@lk4lnode01 ~]# ping lk4lclient -c 4
```

```
[root@lk4lnode02 ~]# ping lk4lnode01 -c 4
[root@lk4lnode02 ~]# ping lk4lclient -c 4
```

```
[root@lk4lclient ~]# ping lk4lnode01 -c 4
[root@lk4lclient ~]# ping lk4lnode02 -c 4
```

# Checking the Kernel Version

Check the kernel version of RHEL.

```
# uname -r
3.10.0-862.11.6.el7.x86_64
# cat /etc/redhat-release
Red Hat Enterprise Linux Server release 7.5 (Maipo)
```

For each distribution that supports LifeKeeper, refer to the SIOS Protection Suite for Linux Support Matrix.

Now you are ready to install LifeKeeper.

# 12.2.5. Building an HA Cluster with LifeKeeper

Install LifeKeeper on the virtual machine OS and build an HA cluster.

## Installing and Launching LifeKeeper

1. Upload the LifeKeeper installation image and license file on the virtual machine using the scp command etc. in advance.

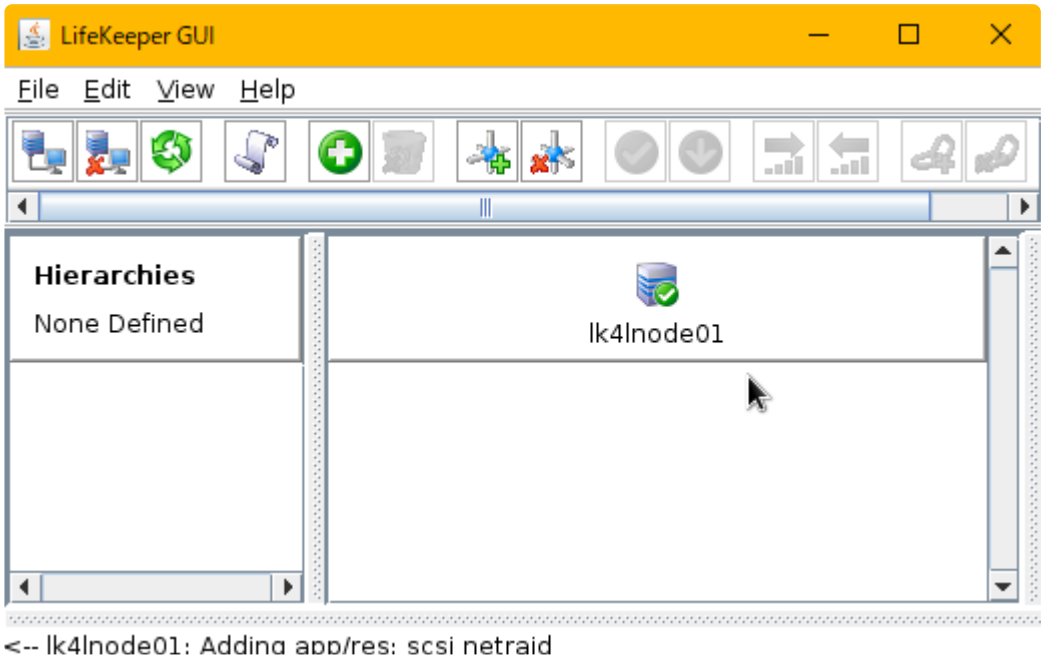
2. Install LifeKeeper on **both cluster nodes**. For installation instructions, refer to the SIOS Protection Suite for Linux Installation Guide.

- Select Use Quorum / Witness Functions.
- For LifeKeeper Authentication, enter root.
- In the Recovery Kit Selection Menu, select LifeKeeper Oracle RDBMS Recovery Kit and DataKeeper for Linux.
- Select LifeKeeper Startup After Install.

3. After installing, add the LifeKeeper command path to your shell environment variables.

```
# vi /root/.bash_profile
PATH=$PATH:/opt/LifeKeeper/bin MANPATH=$MANPATH:/opt/LifeKeeper/man
export PATH MANPATH
# source /root/.bash_profile
```

4. Start the LifeKeeper GUI.



# 12.2.5.1. Configuring Quorum/Witness

Configure Quorum/Witness on the client to prevent split brain. In the following example, the TCP Remote mode is used as Quorum mode and none as Witness mode.

1. Configure Quorum mode and Witness mode. Edit the following files on **both cluster nodes**.

```
#vi /etc/default/LifeKeeper
<omitted>
QUORUM_MODE=tcp_remote      //edit
<omitted>
QUORUM_HOSTS=10.3.1.50      //edit
<omitted>
QUORUM_TIMEOUT_SECS=20    //Edit the timeout as needed
<omitted>
WITNESS_MODE=none  //edit
<omitted>
```

2. Restart LifeKeeper.

```
#lkstop
#lkstart
```

Refer to Quorum Witness and Quorum Parameters List.

# 12.2.5.2. Creating a Communication Path

LifeKeeper performs live monitoring of the nodes by checking the heartbeat response through all the configured communication paths. If the heartbeat communication does not receive a predetermined number of responses, it is determined that the communication path is disconnected, and the status is set to DEAD. If the status of all communication paths is DEAD, LifeKeeper determines it as a node failure and initiates a failover.

Since two or more communication paths between servers are recommended with LifeKeeper, configure them as follows.

| Type | Priority | Active | Standby |
|---|---|---|---|
| TCP | 1 | 10.3.1.11 | 10.3.1.12 |
| TCP | 2 | 10.3.2.11 | 10.3.2.12 |

Refer to Creating a Communication Path for details.

# 12.2.5.3. Disable Broadcast Ping

In an Azure virtual machine environment, the response to Broadcast Ping cannot be obtained immediately after creating the IP resource. Therefore, disable the alive monitoring with Broadcast Ping on both cluster nodes.

Edit the following file.

```
# vi /etc/default/LifeKeeper
<omitted>
NOBCASTPING=1          //Change from an initial value 0
```

# 12.2.5.4. Creating IP Resources

Create an IP resource with the LifeKeeper GUI.

1. Select **Create Resource Hierarchy** from the LifeKeeper GUI administration screen.

2. Follow the Create Resource wizard. Use the following values.

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Select Recovery Kit | IP | |
| Switchback Type | Intelligent | |
| IP Resource | 10.3.1.200 | *Note 1 |
| Netmask | 255.255.255.0 | |
| Network Interface | eth0 | |
| IP Resource Tag | ip-10.3.1.200 | |

- **Note 1:** Use the same value as the configured Azure load balancer frontend IP.

When extending, use the following values.

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Target Server | lk4lnode02 | |
| Switchback Type | Intelligent | |
| Template Priority | 1 | |
| Target Priority | 10 | |
| IP Resource | 10.3.1.200 | *Note 1 |
| Netmask | 255.255.255.0 | |
| Network Interface | eth0 | |
| IP Resource Tag | ip-10.3.1.200 | |

- **Note 1:** Use the same value as the configured Azure load balancer frontend IP.

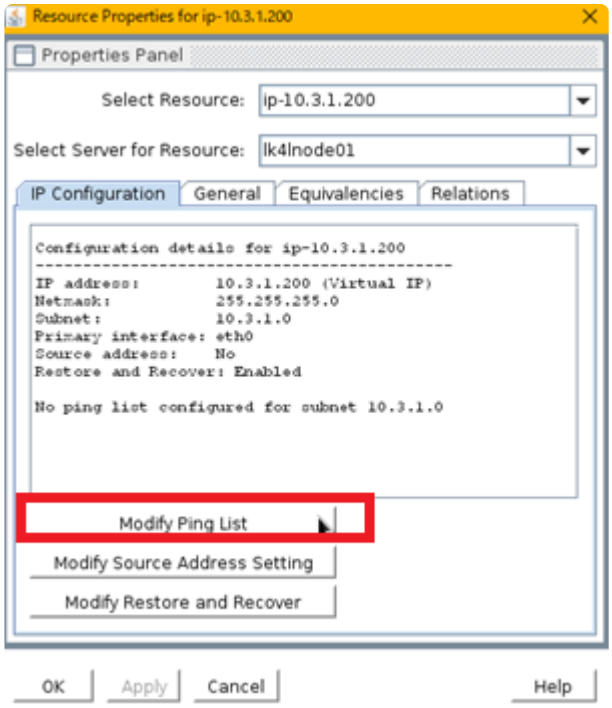
3. IP resource is created.

Creating an IP Resource Hierarchy

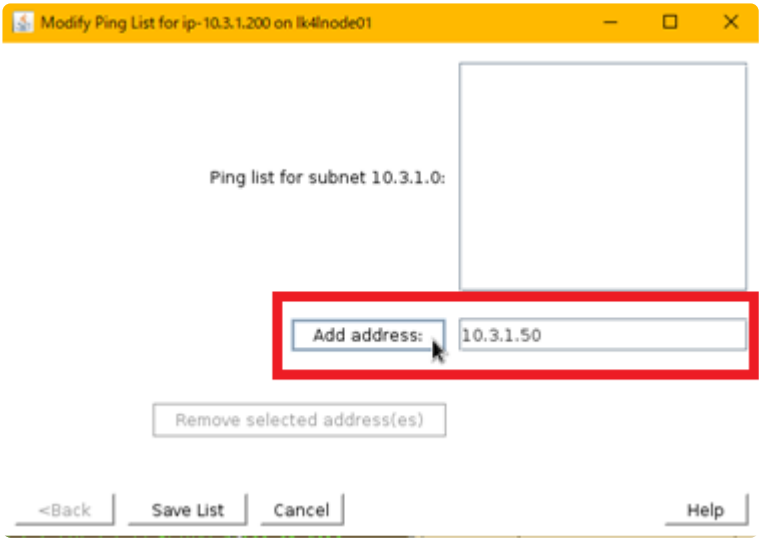
In this environment, since the monitoring by Broadcast Ping is disabled, we will perform live monitoring using Unicast Ping.

**Example**

Open the IP resource property screen and click **Modify Ping List**.




Enter the IP address to set in the Ping List and click **Add address**.




Click **Save List** to save the settings.

Verify that you can get a ping response from the Ping List address you configured.

```
# ping -I 10.3.1.200 10.3.1.50
```

# 12.2.5.5. Creating a Data Replication Resource Hierarchy

This document describes how to create a data replication resource.

## Configuring a Disk

First, configure the disk to be used by the data replication resource.

1. Create a directory for an oracle database mount point on both nodes of the cluster.

```
# mkdir /mnt/ORA
```

2. On both nodes, verify that the disk (the disk specified when creating the instance) is attached.

```
#parted -l
(omitted)
/dev/sdc: unrecognised disk label
Model: Msft Virtual Disk (scsi)
Disk /dev/sdc: 33.3GB
(omitted)
Model: Msft Virtual Disk (scsi)
Disk /dev/sdd: 11.8GB
(omitted)
```

3. Create a disk partition for Oracle on both nodes.

```
#parted /dev/sdc
GNU Parted 3.1
Using /dev/sdc
…… (omitted)
```

4. Check the created partition.

```
# parted -l
(omitted)
Model: Msft Virtual Disk (scsi)
Disk /dev/sdc: 32.2GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name  Flags
 1      1049kB  32.2GB                       ora

Model: Msft Virtual Disk (scsi)
Disk /dev/sdd: 10.7GB
Sector size (logical/physical): 512B/512B
Partition Table: loop
Disk Flags:

Number  Start   End     Size    File system     Flags
 1      0.00B   10.7GB  10.7GB  linux-swap(v1)
```

5. Create a mount point for the file system for Oracle on both cluster nodes.

```
# mkdir /mnt/ORA
```

6. Next, configure DataKeeper to recognize the device.

   When performing data replication using DataKeeper for Linux on Azure, since the ID used to identify the storage with the standard configurations cannot be obtained, you need to create a GUID Partition (GPT) and assign a unique ID to the partition or use an LVM. Refer to Troubleshooting for more information.

   Now the disk is configured.

## Creating a Data Replication Resource Hierarchy

Next, create a Data Replication resource with the LIfeKeeper GUI.

1. Start the LifeKeeper GUI with the following command on the primary node.

```
# /opt/LifeKeeper/bin/lkGUIapp &
```

2. Click the **Create Resource Hierarchy** icon to start creating a resource.

3. The Create Resource Wizard appears. Select **Data Replication** for the Recovery Kit from the pull-down menu and click **Next**.

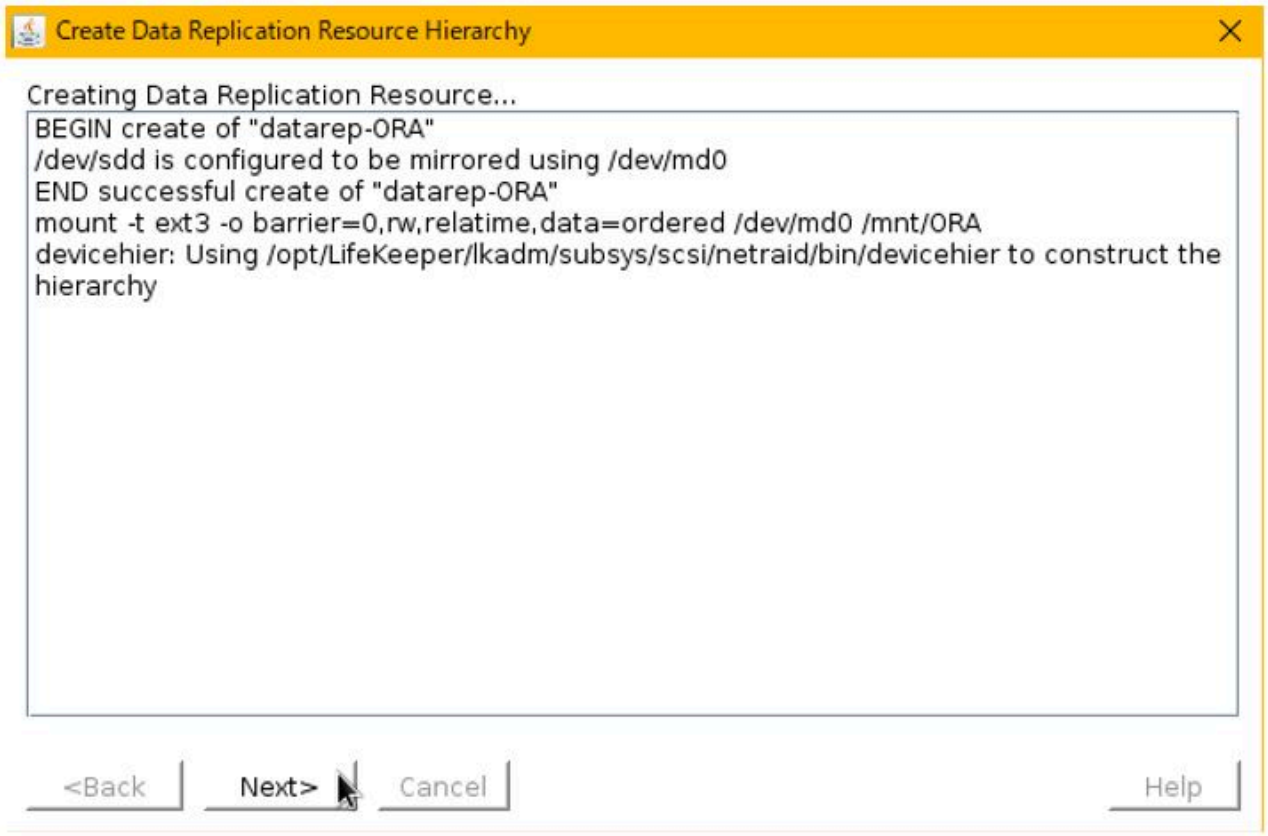
4. In the Create Resource Wizard, enter the following values.

| Item | Value to be entered or selected | Notes |
|---|---|---|
| Switchback Type | Intelligent | |
| Server | lk4lnode01 | |
| Hierarchy Type | Replicate New Filesystem | |
| Source DIsk | /dev/sdc1 | *Note 1 |
| New Mount Point | /mnt/ORA | *Note 2 |
| New Filesystem Type | xfs | |
| Data Replication Resource Tag | datarep-ORA | |
| File System Resource Tag | /mnt/ORA | |
| Bitmap File | /opt/LifeKeeper/bitmap_mnt_ORA | *Note 3 |
| Enable Asynchronous Replication? | no | |

- **Note 1:** After this, “ATTENTION! <Device name> is not shareable with any other server” is displayed, you can ignore it because it is always displayed when creating a replication.
- **Note 2:** Specify the mount point that has been created.
- **Note 3:** Although the default path is used this time, consider specifying a faster private area to improve performance.

5. Creation of the data replication resource will begin.

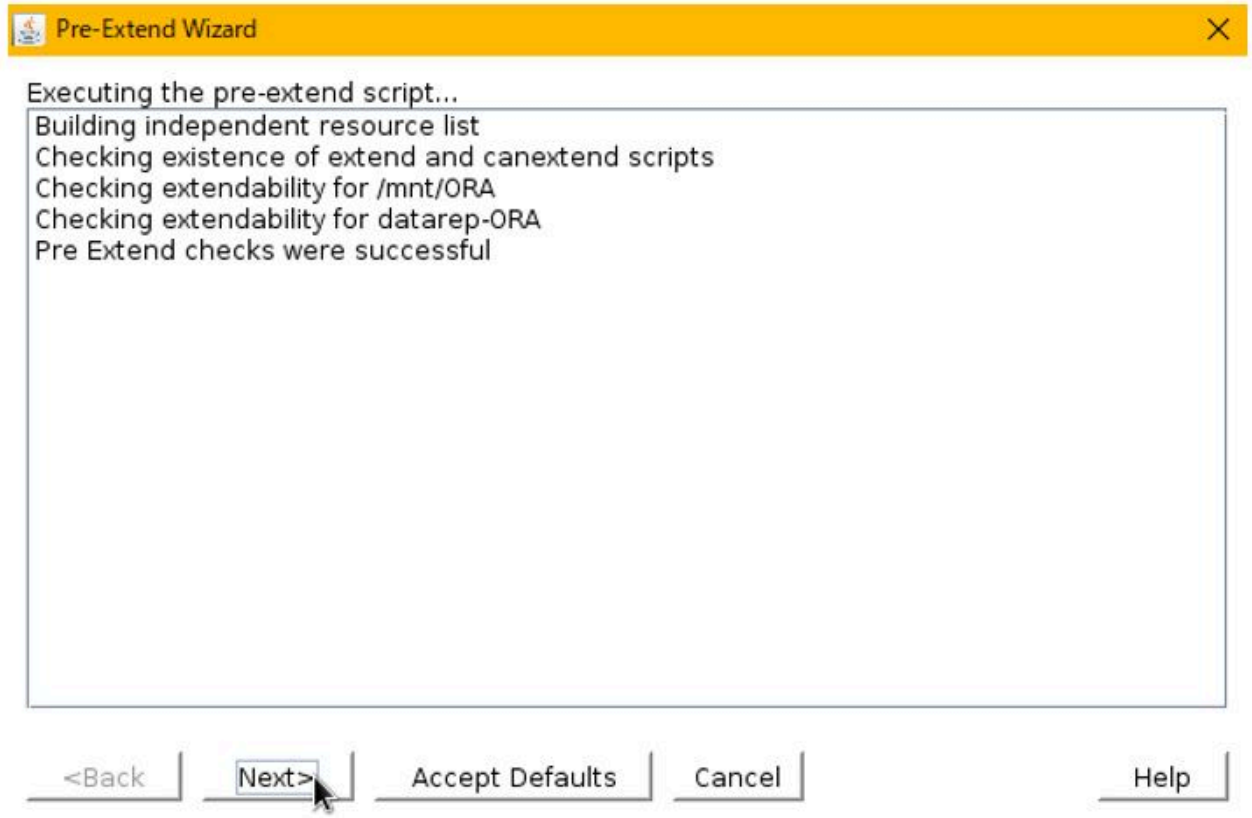
6. “End of successful Create of…” is displayed on the screen to confirm it is successful. Click **Next** to go to the Pre-Extend Wizard.




7. Enter the following values.

| Item | Value to be entered or selected |
|---|---|
| Target Server | lk4lnode02 |
| Switchback Type | Intelligent |
| Template Priority | 1 |
| Target Priority | 10 |

8. Pre-Extend is started. It is successful if “Pre Extend checks were successful” is displayed. Click **Next**.




9. In the Extend Resource Wizard, enter the following values.

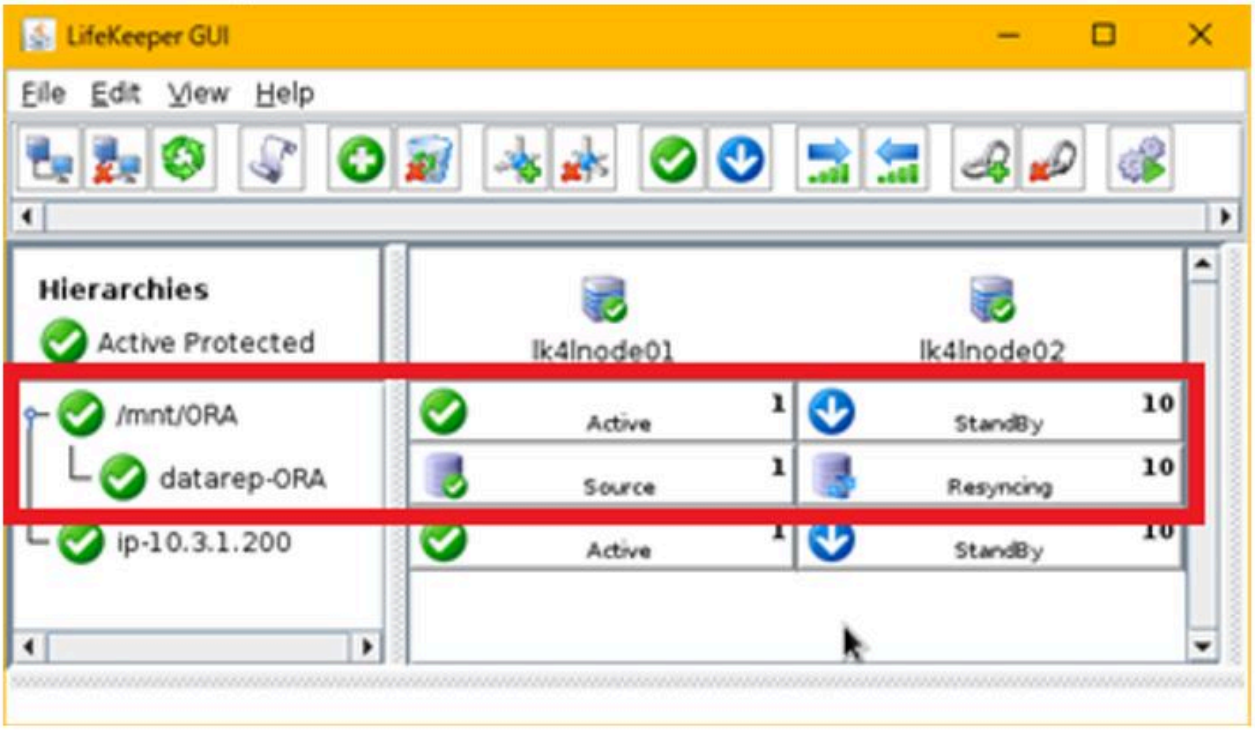
| Item | Value to be entered or selected | Notes |
|---|---|---|
| Mount Point | /mnt/ORA | |
| Root Tag | /mnt/ORA | |
| Target Disk | /dev/sdc1 | |
| Data Replication Resource Tag | datarep-ORA | |
| Bitmap File | /opt/LifeKeeper/bitmap_mnt_ORA | |
| Replication Path | 10.3.2.11/10.3.2.12 | *Note 1 |

*Note1: Specify the communication path on the second subnet.

10. Extending starts. If “Hierarchy successful extended” is displayed, it is successful. Click **Finish**.

11. Click **Done** to exit the wizard.

12. Data replication resource is created.




*Data replication resource is not created for the swap area.

Refer to Creating a DataKeeper Resource Hierarchy for details.

# 12.2.5.6. Creating an Oracle Resource Hierarchy

Set up the environment for installing Oracle Database 12c. For details on the prerequisites for installing Oracle, please refer to the Oracle documentation.

Make the following settings on **both cluster nodes**.

1. Create a swap area.

   At least 3GB of swap space is required to install Oracle. Configure the swap area using the 10GB of disk (*/dev/sdd*) assigned to the instance.

```
# mkswap /dev/sdd
# swapon /dev/sdd
# free
              total        used        free      shared  buff/cache   available
Mem:        1706084      189248     1348300        9184      168536     1300564
Swap:      12582904           0    12582904
# blkid | grep /dev/sdd
/dev/sdd: UUID="268e56bd-c64f-4798-9197-e411a497ba11" TYPE="swap" //write down the UUID
# vi /etc/fstab
# /etc/fstab
# Created by anaconda on Fri Mar 23 17:41:14 2018
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=50a9826b-3a50-44d0-ad12-28f2056e9927 /       xfs     defaults        0 0
UUID=268e56bd-c64f-4798-9197-e411a497ba11 swap swap defaults 0 0   ←追加
```

2. Register the RHEL option repository. Add a repository containing the required package “compat-libstdc++-33”.

```
# yum-config-manager --enable rhui-REGION-rhel-server-extras rhui-REGION-rhel-server-optional
```

3. Install the package.

```
# yum -y install binutils compat-libcap1 compat-libstdc++-33 compat-libstdc++-33.i686 gcc gcc-c++ glibc glibc.i686 glibc-devel glibc-devel.i686 ksh libgcc libgcc.i686 libstdc++ libstdc++.i686 libstdc++-devel libstdc++-devel.i686 libaio libaio.i686 libaio-devel libaio-devel.i686 libXext libXext.i686 libX11 libX11.i686 libxcb libxcb.i686 libXi libXi.i686 make sysstat
```

4. Edit the kernel parameters.

**Note:** Symbols and blanks may be changed to different characters due to Word and Exel specifications and an error may occur when you execute the command using copy and paste. Do not add a blank at the end. The command may fail because the variable is recognized as a string instead of a number.

```
# MEMTOTAL=$(free -b | sed -n '2p' | awk '{print $2}')
# SHMMAX=$(expr $MEMTOTAL / 2)
# SHMMNI=4096
# PAGESIZE=$(getconf PAGE_SIZE)
# cat >> /etc/sysctl.conf << EOF
>fs.aio-max-nr = 1048576
>fs.file-max = 6815744
>kernel.shmmax = $SHMMAX
>kernel.shmall = `expr ¥( $SHMMAX / $PAGESIZE ¥) ¥* ¥( $SHMMNI / 16 ¥)`
>kernel.shmmni = $SHMMNI
>kernel.sem = 250 32000 100 128
>net.ipv4.ip_local_port_range = 9000 65500
>net.core.rmem_default = 262144
>net.core.rmem_max = 4194304
>net.core.wmem_default = 262144
>net.core.wmem_max = 1048576
>EOF
```

5. Apply to the kernel.

```
#   sysctl -p
```

6. Create a dedicated user group for Oracle and configure the system environment.

```
# i=54321; for group in oinstall dba backupdba oper dgdba kmdba; do groupadd -g $i
$group; i=`expr $i + 1` ;done
# useradd -u 1200 -g oinstall -G dba,oper,backupdba,dgdba,kmdba -d /home/oracle
```

7. Set the password for the Oracle user.

```
# passwd oracle
Changing password for user oracle.
New password:
Retype new password:
Passwd: all authentication tokens updated successfully.
```

8. Verify that */mnt/ORA* is protected by DK and mounted on the primary node. (This will only be done on the primary node)

```
# df
Filesystem      1K-blocks      Used Available Use% Mounted on
<<中 omitted>>
/dev/md0          31864752   7396580   22842880  25% /mnt/ORA
<<後 omitted>>
#lcdstatus -e
lk4lnode02    LSNR.LISTENER    dblistener-11079     ISP        1   lk4lnode01
lk4lnode02     /mnt/ORA          /mnt/ORA              ISP        1   lk4lnode01
lk4lnode02      datarep-ORA    /dev/sdc              ISP        1   lk4lnode01
lk4lnode02     ip-10.3.1.200        ISP        1   lk4lnode01
<<後 omitted>>
```

9. Change directory creation, access rights, and owner. (Perform this only on the primary node)

```
# mkdir -p /mnt/ORA/app/oracle
# chown -R oracle:oinstall /mnt/ORA/app
# chmod -R 775 /mnt/ORA
```

10. Add it in */etc/pam.d/login* around line 14.

```
# vi /etc/pam.d/login
session     required      pam_selinux.so open
session     required      pam_namespace.so
session     required      pam_limits.so     //Add
session     optional      pam_keyinit.so force revoke
```

11. Add it to the last line of */etc/security/limits.conf*.

```
# vi /etc/security/limits.conf
oracle   soft   nproc    2047    //Add
oracle   hard   nproc    16384    //Add
oracle   soft   nofile   1024    //Add
oracle   hard   nofile   65536    //Add
oracle   soft   stack    10240    //Add
oracle   hard   stack    32768    //Add
```

12. Set up the environment for Oracle. The work is done with the “oracle” user created in the previous step.

```
# su oracle
$ vi ~/.bash_profile
// Add the following to the last line
umask 022
export ORACLE_BASE=/mnt/ORA/app/oracle
export ORACLE_HOME=/mnt/ORA/app/oracle/product/12.1.0/dbhome_1
export PATH=$PATH:$ORACLE_HOME/bin
export ORACLE_SID=＜SID name＞
export NLS_LANG=Japanese_Japan.AL32UTF8

$source ~/.bash_profile
```

Write down the ORACLE_SID you set up here for use in later steps. In this example it is “lkoracle”.

13. Copy the root xauth settings to the oracle user. (This is done only on the primary node)

```
# xauth list
# su oracle
$ xauth add <DisplayName> <ProtocolName> <Hexkey>
```

14. Verify that the xauth output for the oracle user is the same as root.

```
$ xauth list
```

15. Create a directory for installation. (This is done only on the primary node)

```
$ mkdir /home/oracle/tmp
```

Now you are ready to install.

# 12.2.5.6.1. Installing Oracle DB

Perform the following steps on the cluster node (primary). For Oracle installation, please refer to the Oracle documentation.

1. Download the Oracle 12c installation package and unzip it to the appropriate directory under */home/oracle*.

2. Switch to the oracle user with su oracle

3. A directory named “database” is created under the unzipped directory. The directory will launch the installer, which is located:

```
$./runInstaller &
```

4. Configure Security Update. Enter the registered email address and support password for Oracle and click **Next**. This is not mandatory.

5. Installation option. Select **Install Database Software only** and click **Next**.

6. Grid Installation options. Select **Single Instance Database** installation and click **Next**.

7. Product Language. Select your **preferred language** and click **Next**.

8. Database Edition. Select **Enterprise Edition** and click **Next**.

9. Installation Location. If it is
   Oracle base = /mnt/ORA/app/oracle
   Software Location = /mnt/ORA/app/oracle/product/12.1.0/dbhome_1
   click **Next**.

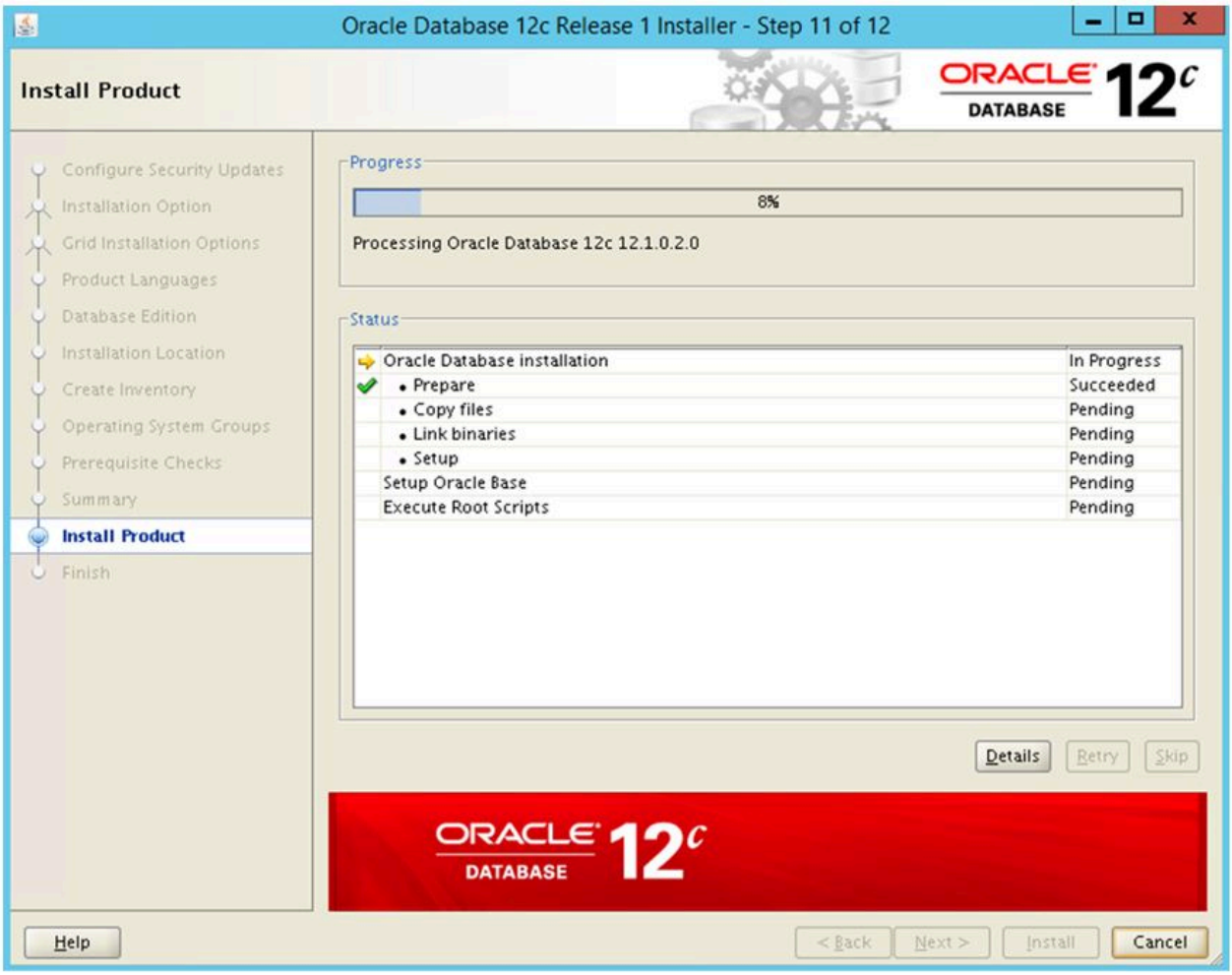
10. Create Inventory. Use the **default value** and click **Next**.

11. Product list checks. Use the **default value** and click **Next**.

12. Operating System Group. Use the **default value** and click **Next**.

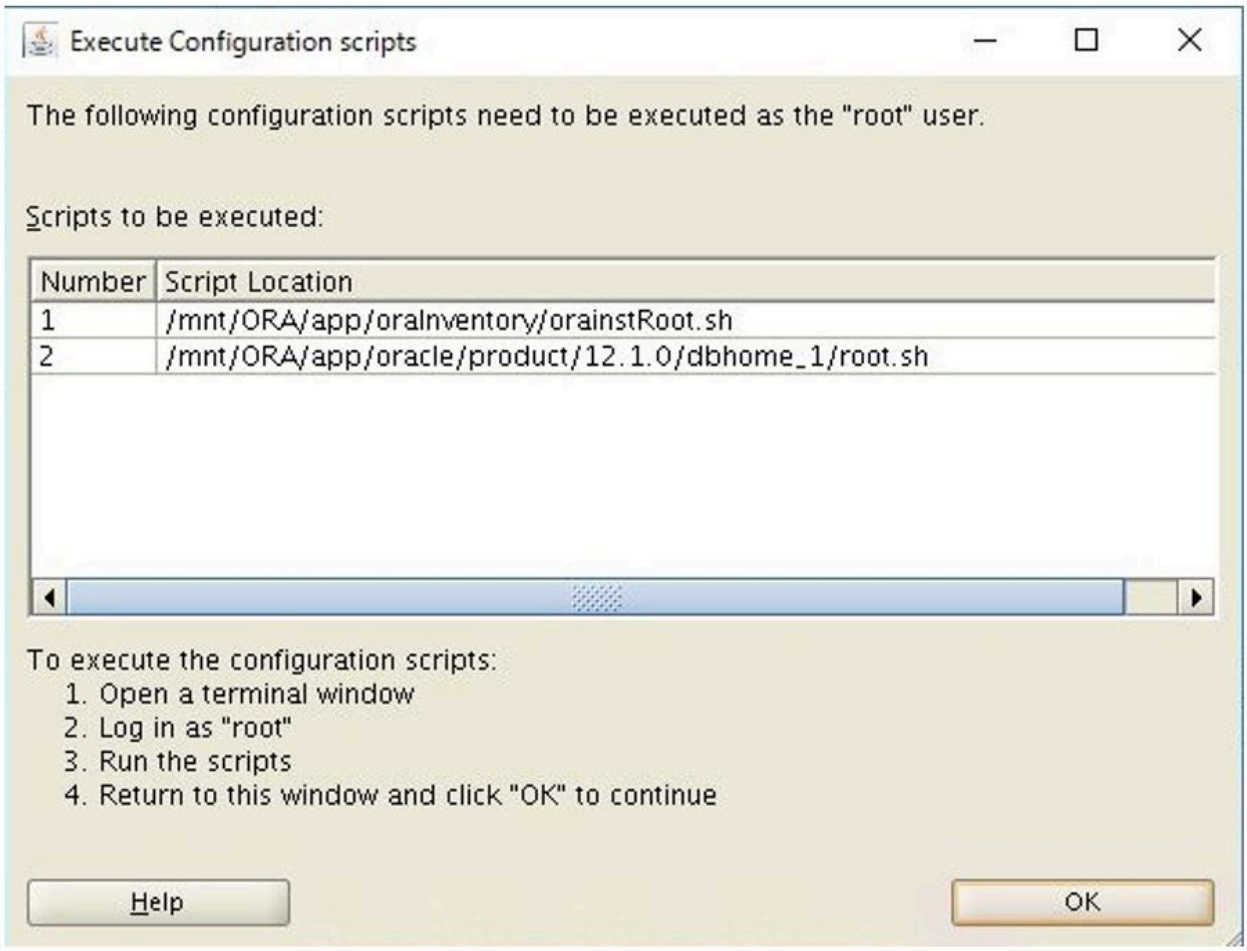
13. Summary. If there is no problem, run **Install**.

14. Installation is started.




15. The Execute Configuration scripts window will open. Follow the instructions and run the script as root.

```
# /mnt/ORA/app/oraInventory/orainstRoot.sh
Changing permissions of /mnt/ORA/app/oraInventory.
Adding read,write permissions for group.
Removing read,write,execute permissions for world.

Changing groupname of /mnt/ORA/app/oraInventory to oinstall.
The execution of the script is complete.
# /mnt/ORA/app/oracle/product/12.1.0/dbhome_1/root.sh
Performing root user operation.

The following environment variables are set as:
    ORACLE_OWNER= oracle
    ORACLE_HOME=  /mnt/ORA/app/oracle/product/12.1.0/dbhome_1

Enter the full pathname of the local bin directory: [/usr/local/bin]:
   Copying dbhome to /usr/local/bin ...
   Copying oraenv to /usr/local/bin ...
   Copying coraenv to /usr/local/bin ...


Creating /etc/oratab file...
Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.
```



When the script is completed, click **OK**.

16. Click **Close** to complete the installation.

# 12.2.5.6.2. Configuring a Listener

Configure a listener. On the cluster node (primary), make the following settings:

1. As the oracle user, execute the following command.

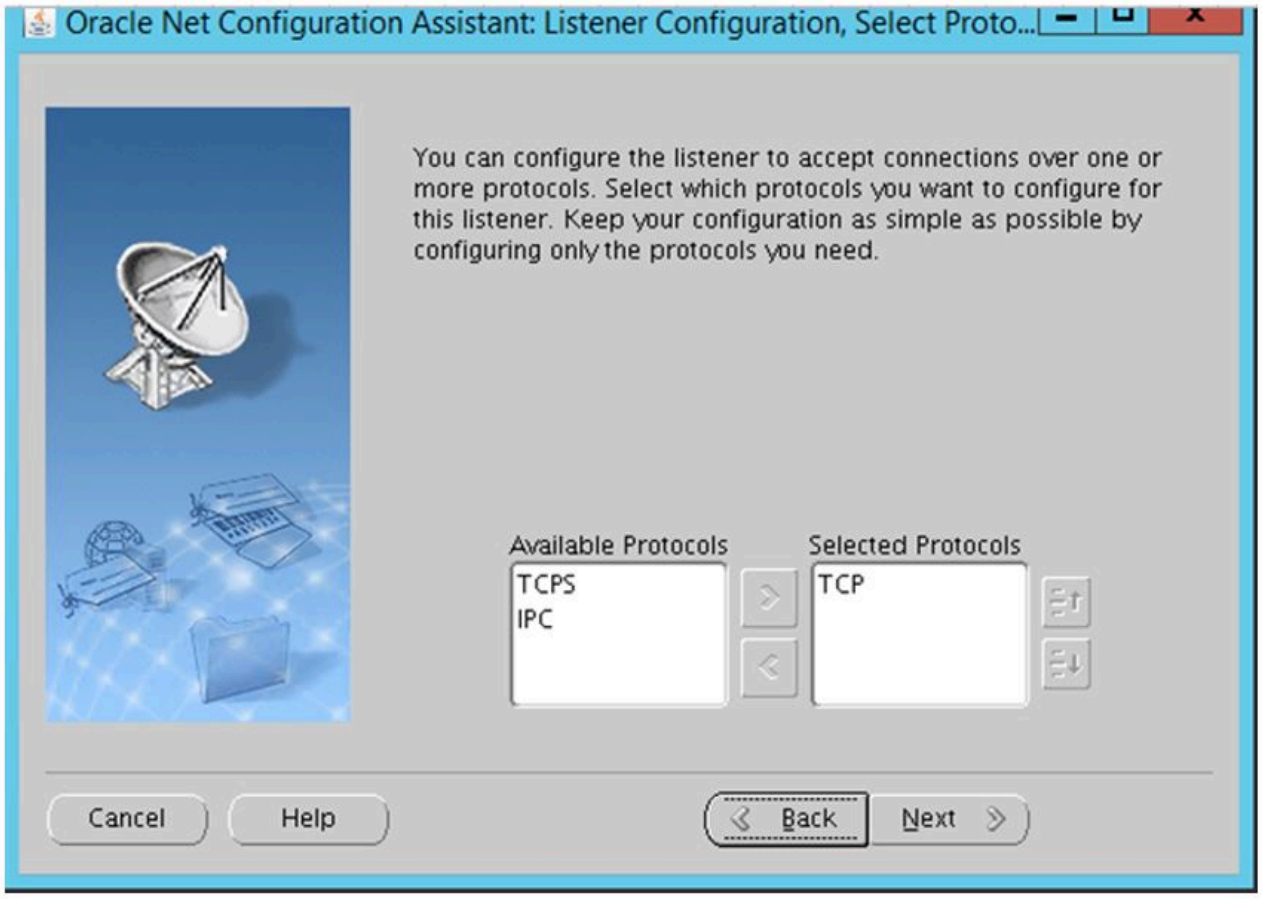
```
$ /mnt/ORA/app/oracle/product/12.1.0/dbhome_1/bin/netca &
```

2. Select **Listener Configuration** and click **Next**.

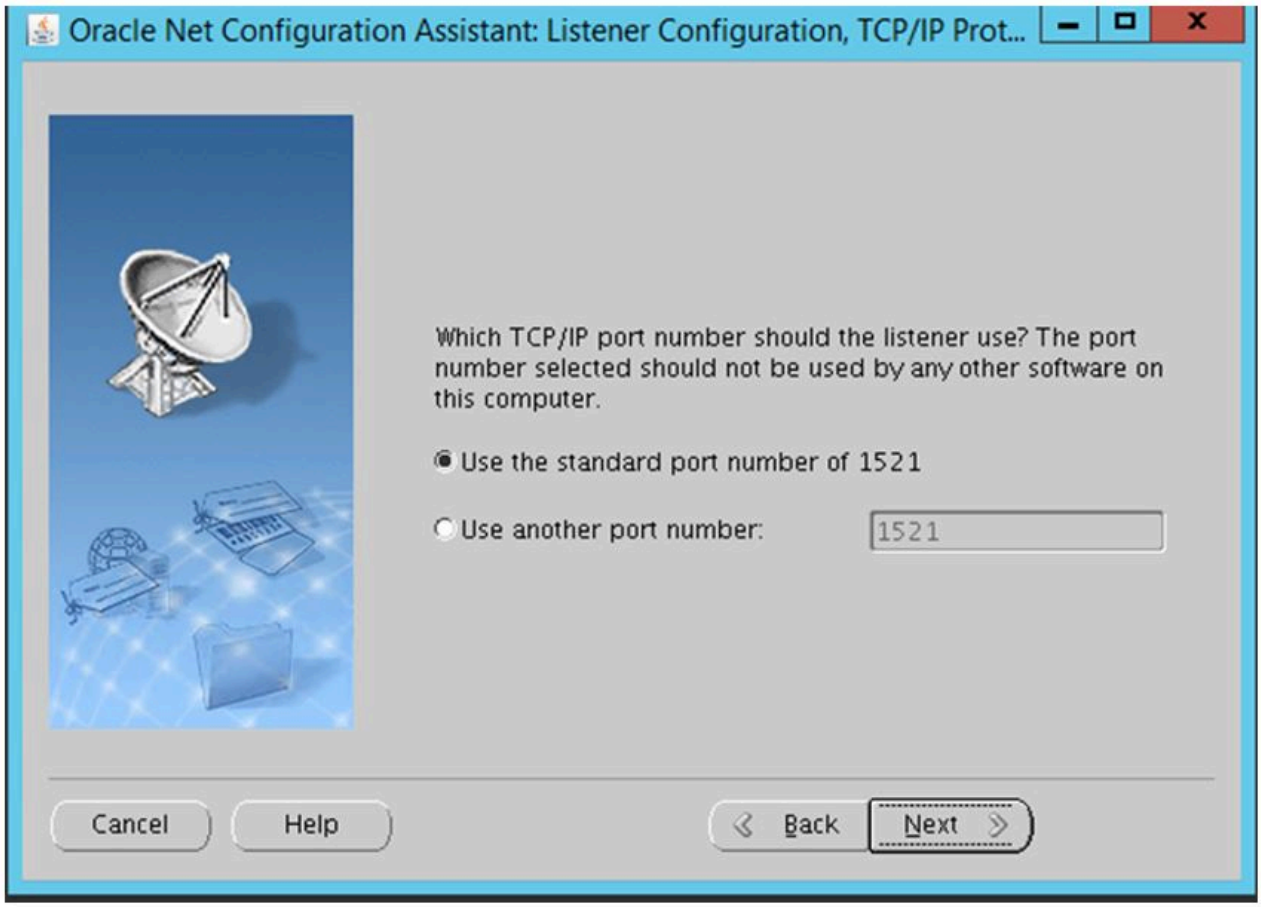
3. Select **Add** and click **Next**.

4. Specify a name for the Listener and click **Next**.

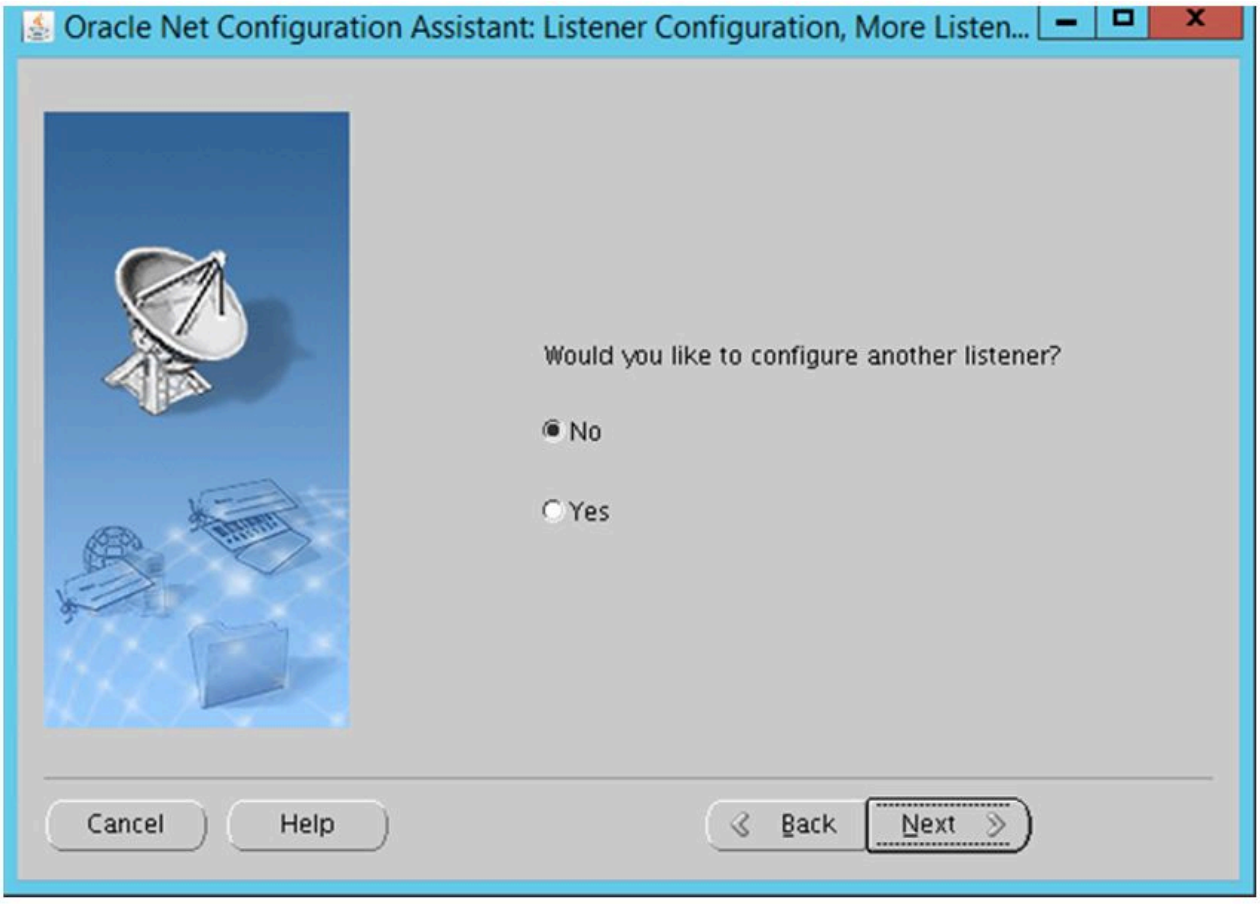
5. Click **Next**.



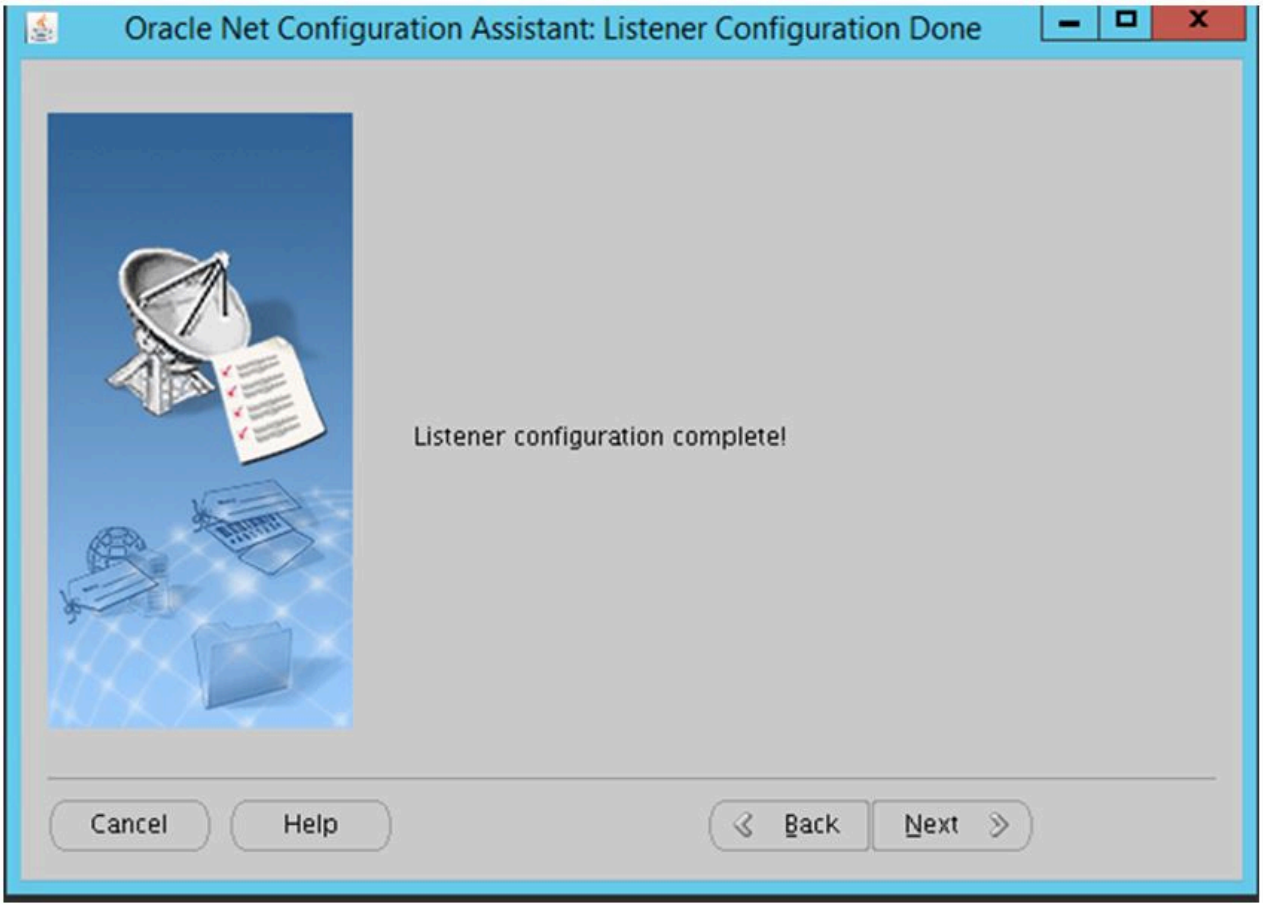

6. Select the default **port 1521** and click **Next**.




7. Complete the Listener configuration and click **Next**.



8. The Listener configuration is completed. Click the **[X]** at the top right of the dialog to complete.




The Listener configuration is now completed.

# 12.2.5.6.3. Creating the DB

Configure the following settings on the cluster node (primary).

1. Execute the following command as an oracle user:

```
$ /mnt/ORA/app/oracle/product/12.1.0/dbhome_1/bin/dbca &
```

2. Database operation. Select **Create Database** and click **Next**.
3. Creation Mode. Select **Advanced Mode** and click **Next**.
4. Database Template. Select **General Purpose** or **Transaction Processing** and click **Next**.
5. Database Identification. Set the **database name** and **SID** as follows:
   Global Database Name = lkoracle
   SID = lkoracle
7. Management Options. Use the **default value** and click **Next**.
8. Database Credentials. Select **Use the Same Administrative Password for All Accounts** to register the password.
9. Network Configuration. Since the Listener has been created, use the **default value** and click **Next**.
10. Storage Locations. Use the **default value** and click **Next**.
11. Database Options. Use the **default value** and click **Next**.
12. Initialize Parameters. Use the **default value** and click **Next**.
13. Creation Options. Use the **default value** and click **Next**.
14. If the settings are correct, click **Finish** to start creating the database.
15. The database is created.
16. Next, modify */etc/oratab*. Note that this setting will be undone in a later step.

```
#   vi /etc/oratab
oracle:/mnt/ORA/app/oracle/product/12.1.0/dbhome_1:Y ←change N to Y
```

17. Copy */etc/oratab* from the primary node to the secondary node.

    The installation of Oracle DB is now completed.

    *Installation of Oracle DB on the secondary node is not necessary.

# 12.2.5.6.4. Configuring Oracle

The configuration file must be modified before creating the Listener resource since the Listener is being accessed by the virtual IP address.

On the cluster node (primary), configure the following settings.

1. Modify *$ORACLE_HOME/network/admin/listener.ora*.

```
$vi $ORACLE_HOME/network/admin/listener.ora
<<中 omitted>>
        SID_LIST_LISTENER =
          (SID_LIST =
            (SID_DESC =
              (SID_NAME = LKORACLE)
            )
          )

        LISTENER =
          (DESCRIPTION_LIST =
            (DESCRIPTION =
              (ADDRESS = (PROTOCOL = TCP)(HOST = 10.3.1.200)(PORT = 1521))
            )
    )
```

2. Modify *$ORACLE_HOME/network/admin/tnsnames.ora*.

```
$vi $ORACLE_HOME/network/admin/tnsnames.ora
LISTENER_LKORACLE =
  (ADDRESS = (PROTOCOL = TCP)(HOST = 10.3.1.200)(PORT = 1521))


LKORACLE =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = 10.3.1.200)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = lkoracle)
    )
  )
```

3. Create a password file to be used for login via the Listener. Refer to the official Oracle documentation for the format of the password file.

```
$orapwd file=orapwlkoracle force=y ignorecase=y password=XXXXXXXX
```

4. Check the connection to the database via the Listener.

```
$ sqlplus sys/XXXXXXXX@10.3.1.200:1521/lkoracle as sysdba
SQL*Plus: Release 12.1.0.2.0 Production on 木 5 月 23 07:52:41 2019

Copyright (c) 1982, 2014, Oracle.  All rights reserved.



Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
に接続されました。
SQL> select instance_name from v$instance ;

INSTANCE_NAME
----------------------------------------------------
lkoracle
```

# 12.2.5.6.5. Creating an Oracle Database Listener Resource Hierarchy

First, create an Oracle Database Listener resource.

1. Click the **Create Resource Hierarchy** icon to start creating a resource.

2. The Create Resource Wizard appears. Select **Oracle Database Listener** for the Recovery Kit from the pull-down menu and click **Next**.

3. In the Create Resource wizard, enter the following values.

| No. | Item | Value to be entered or selected |
|---|---|---|
| 1 | Switchback Type | Intelligent |
| 2 | Server | lk4lnode01 |
| 3 | Listener Configuration File Path | /mnt/ORA/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.ora |
| 4 | Listener Name(s) | LISTENER |
| 5 | Listener Executable(s) | /mnt/ORA/app/oracle/product/12.1.0/dbhome_1/bin |
| 6 | Listener Protection Level | Full Control(Start,Stop,Monitor,& Recover) |

| | | |
|---|---|---|
| 7 | Listener Recovery Level | Standard, (On) |
| 8 | IP Address Name | ip-10.3.1.200 |
| 9 | Listener Tag | LSNR.LISTENER |

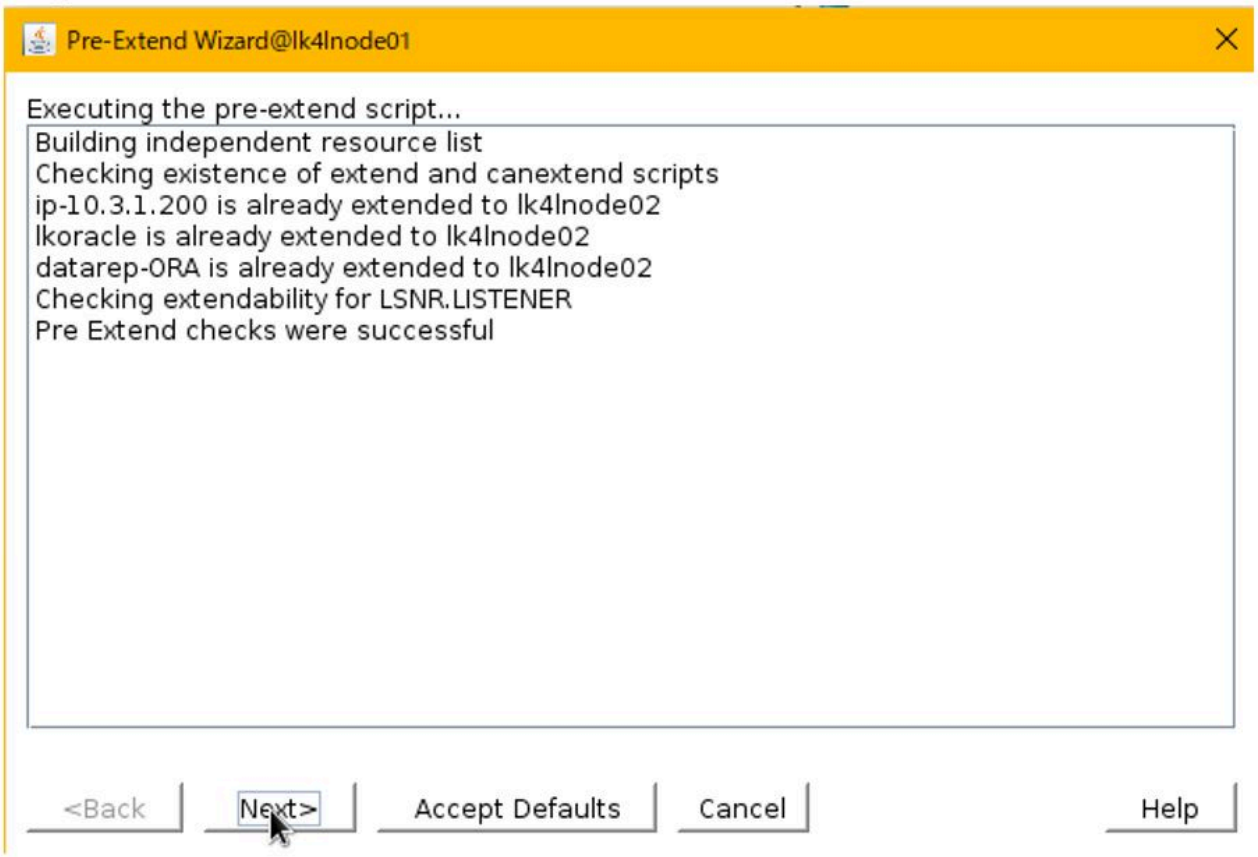
4. Creation of the Oracle Database Listener resource begins.

5. It is successful when “End successful create of …” is displayed. Click **Next** to proceed to the Pre-Extend wizard.

6. Enter the following values.

| No. | Item | Value to be entered or selected |
|---|---|---|
| 1 | Target Server | lk4lnode02 |
| 2 | Switchback Type | Intelligent |
| 3 | Template Priority | 1 |
| 4 | Target Priority | 10 |

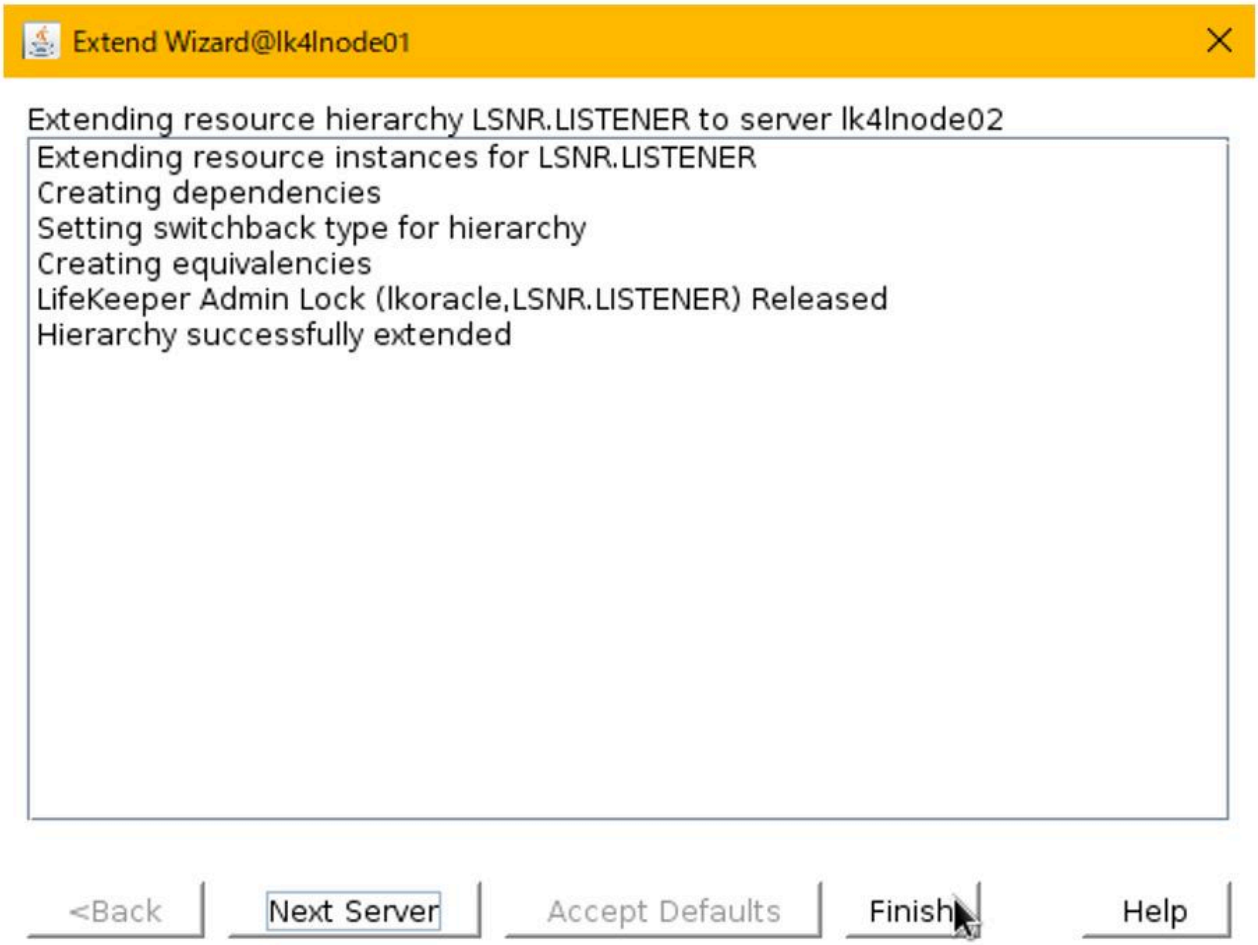
7. Pre-Extend is started. It is successful when “Pre Extend checks were successful” is displayed. Click **Next**.




8. In the Extend Oracle Listener Resource Wizard, enter the following values.

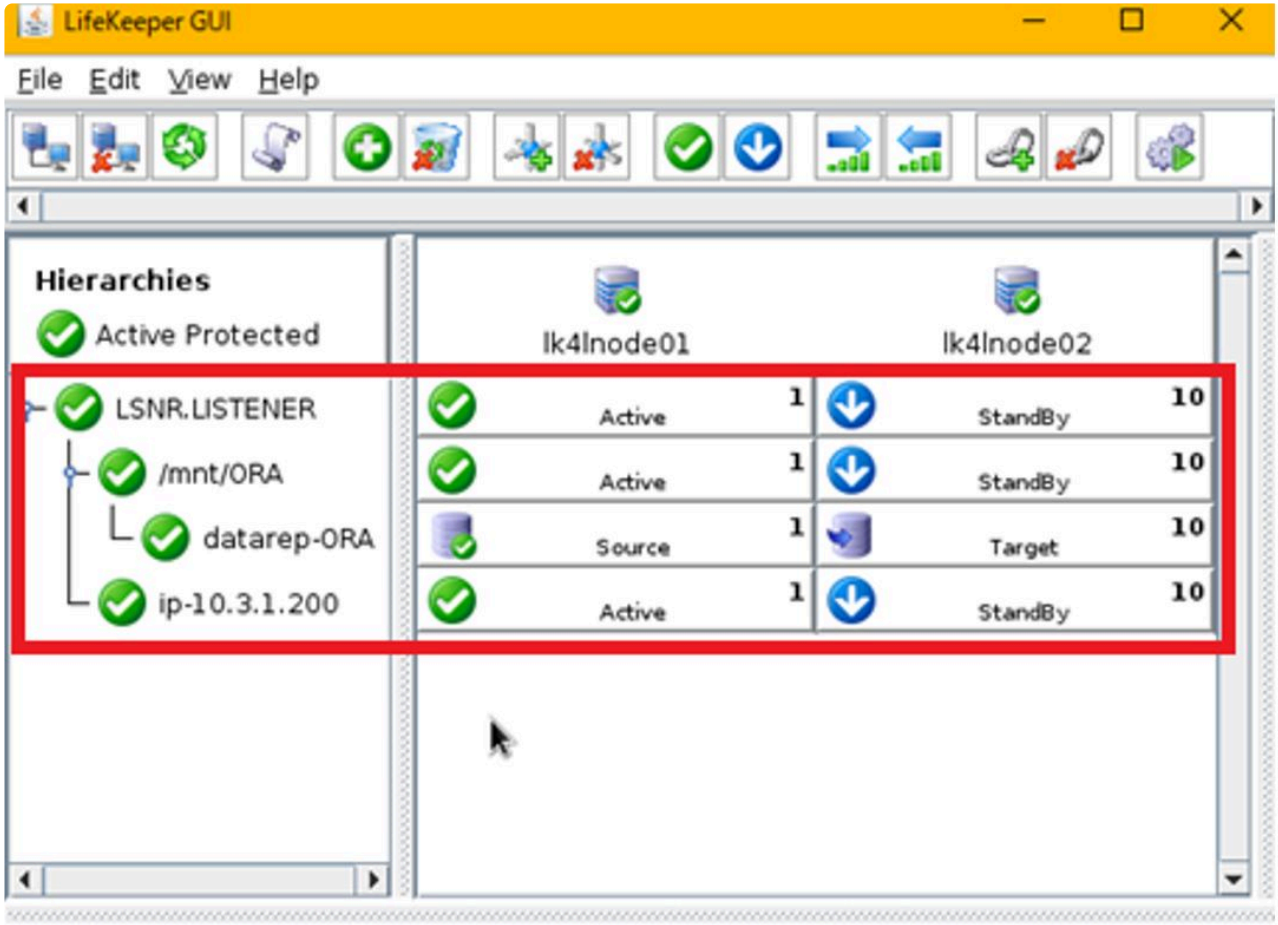
| No. | Item | Value to be entered or selected |
|---|---|---|
| 1 | Enter the Listener Configuration File Path | /mnt/ORA/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.ora |
| 2 | Enter the Path to the Listener Executable(s) | /mnt/ORA/app/oracle/product/12.1.0/dbhome_1/bin |
| 3 | Listener Tag | LSNR.LISTENER |

9. Extend is started. It is successful when “Hierarchy successfully extended” is displayed. Click **Finish**.




10. Press **Done** to exit the wizard.

11. Oracle Database Listener resource is created.



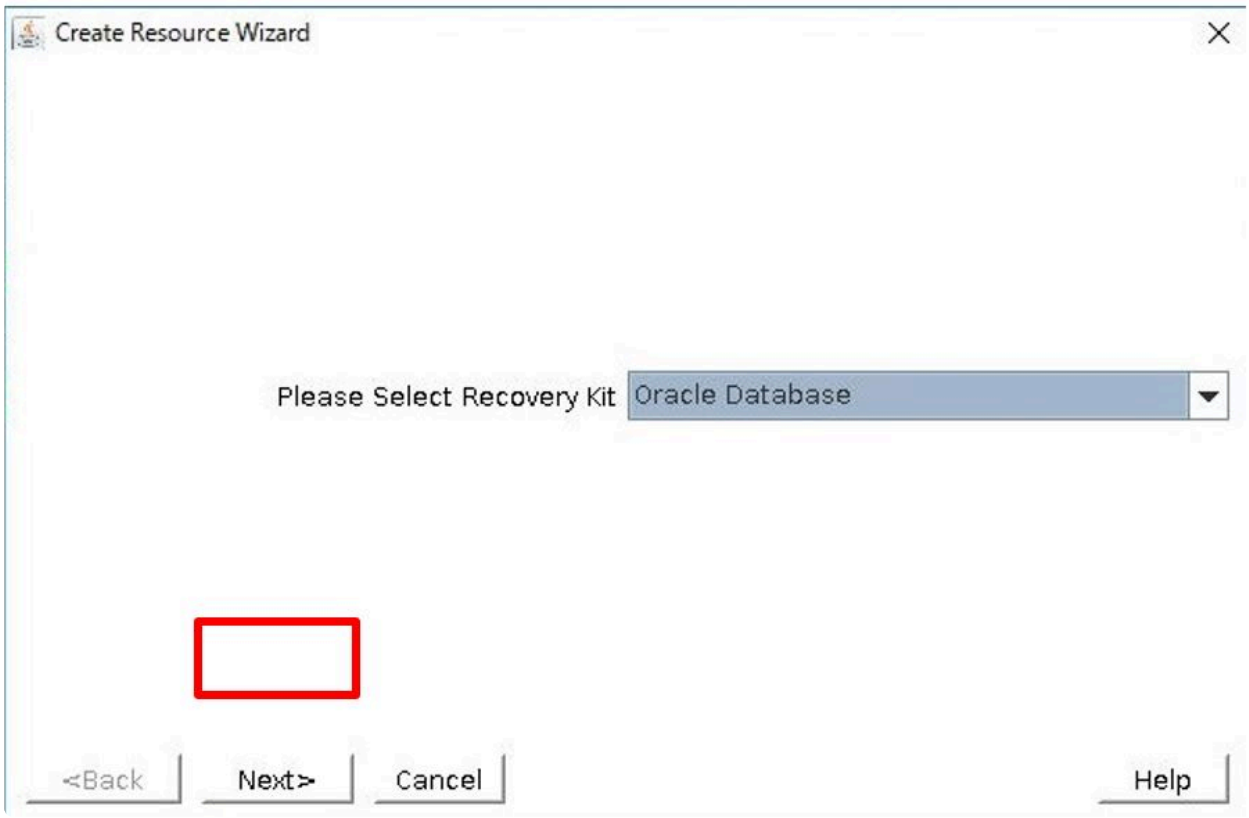

Refer to Creating an Oracle Resource Hierarchy for more information.

# 12.2.5.6.6. Creating an Oracle Resource Hierarchy

Next, create a resource for the Oracle database.

1. Click the Create Resource Hierarchy icon to start creating a resource.

2. The Create Resource Wizard appears. Select **Oracle Database** for the Recovery Kit from the pull-down menu and click **Next**.



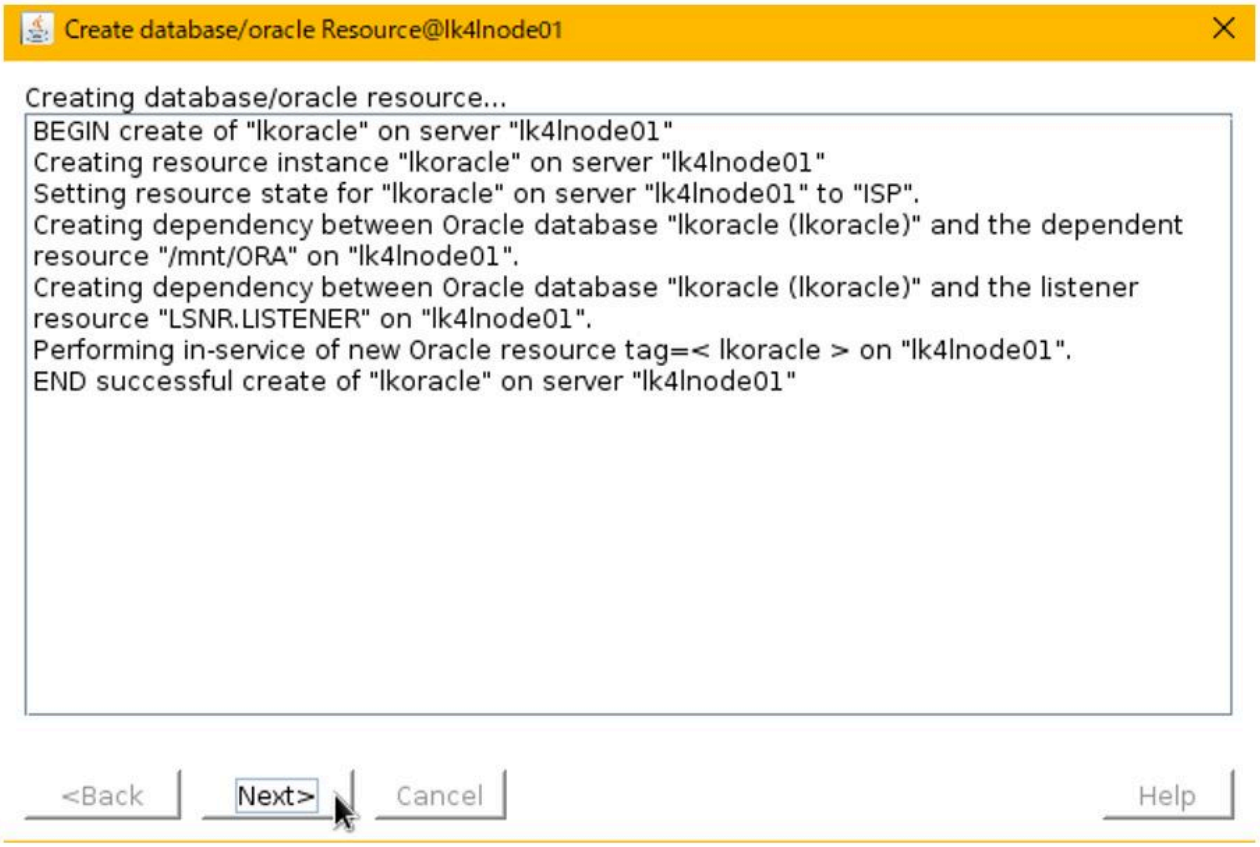

3. In the Create Resource Wizard, enter the following values.

| No. | Item | Value to be entered or selected |
|---|---|---|
| 1 | Switchback Type | Intelligent |
| 2 | Server | lk4lnode01 |
| 3 | Oracle _ SID for Database | lkoracle |
| 4 | User Name | oracle |
| 5 | Password | << Password specified when creating Oracle DB >> |

| | | |
|---|---|---|
| 6 | Select the Oracle Listener | LSNR.LISTENER |
| 7 | Database Tag | lkoracle |

4. The Oracle Database resource creation starts.

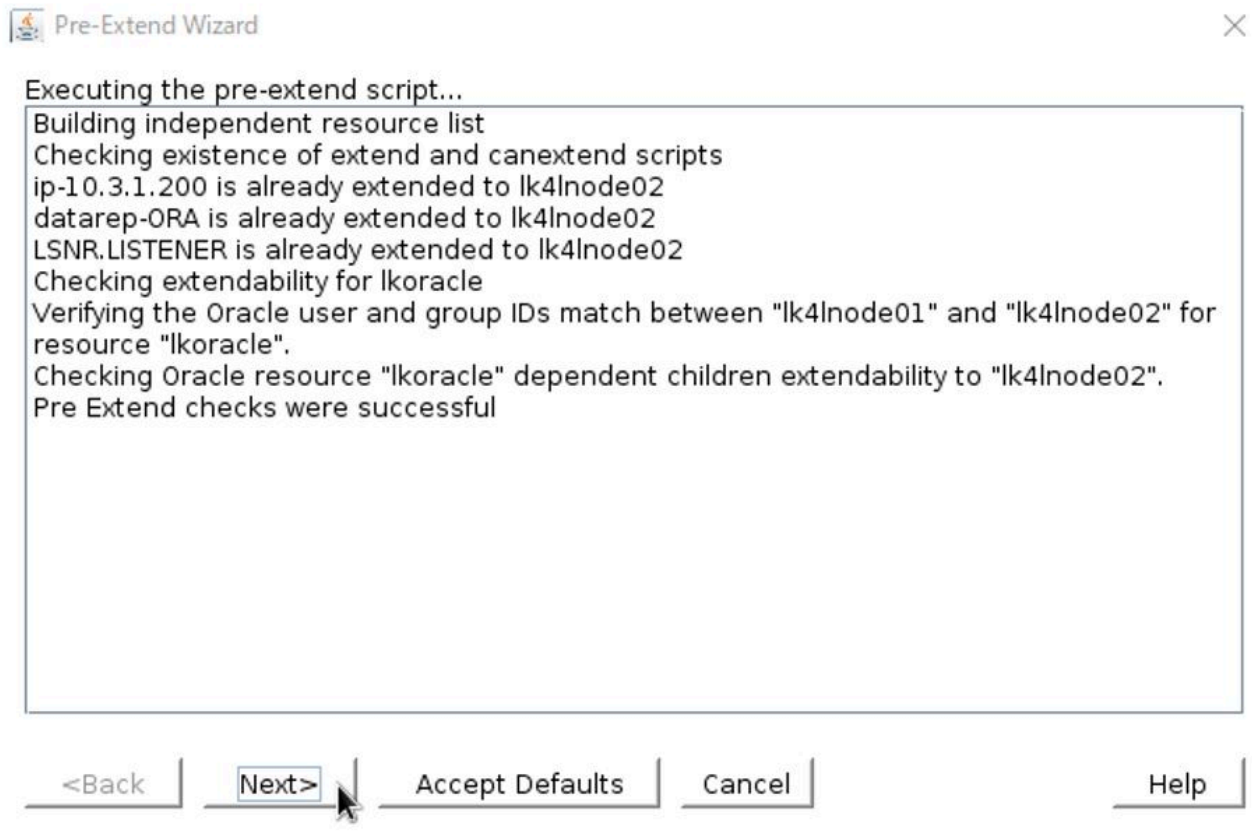
5. It is successful when “End successful Create of …” is displayed. Click **Next** to proceed to the Pre-Extend wizard.



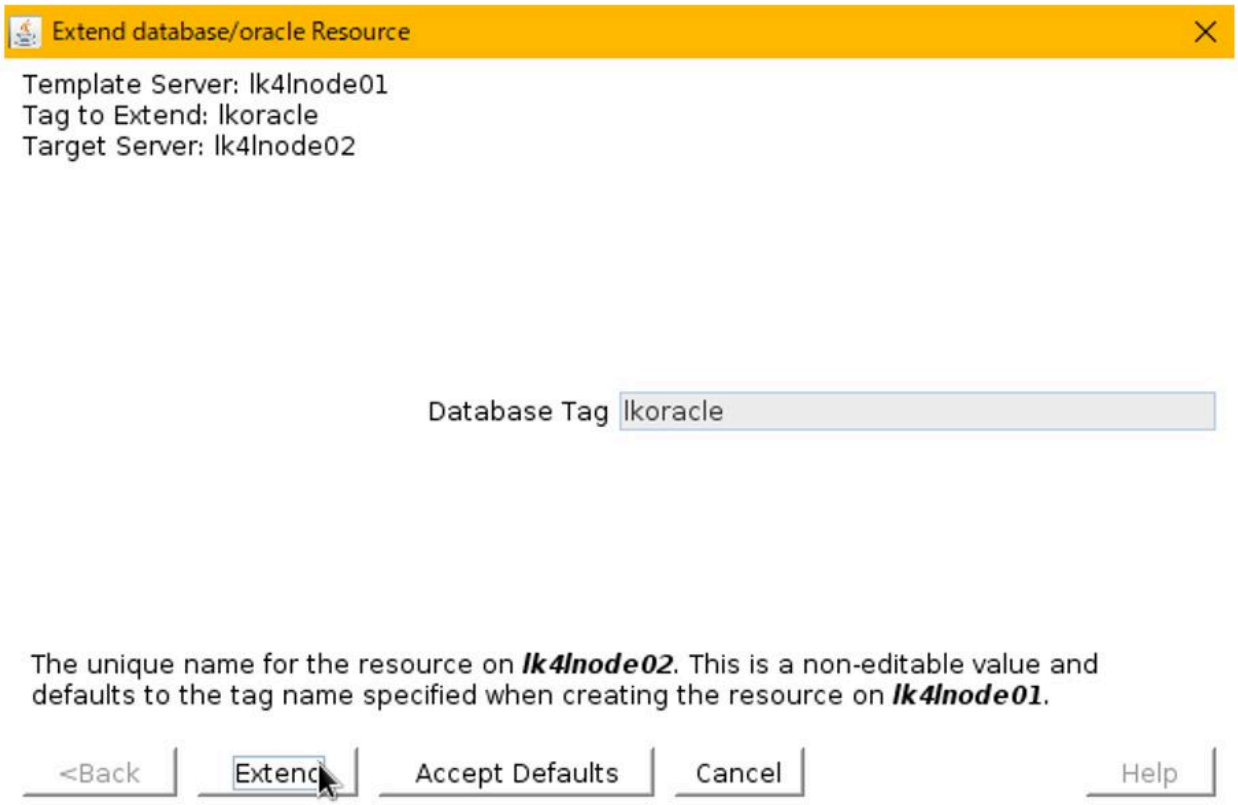

6. Enter the following values.

| No. | Item | Value to be entered or selected |
|---|---|---|
| 1 | Target Server | lk4lnode02 |
| 2 | Switchback Type | Intelligent |
| 3 | Template Priority | 1 |
| 4 | Target Priority | 10 |

7. The Pre-Extend is started. It is successful when “Pre Extend checks were successful” is displayed. Click **Next** to proceed.




8. When the Extend database/oracle Resource wizard appears, click **Extend**.

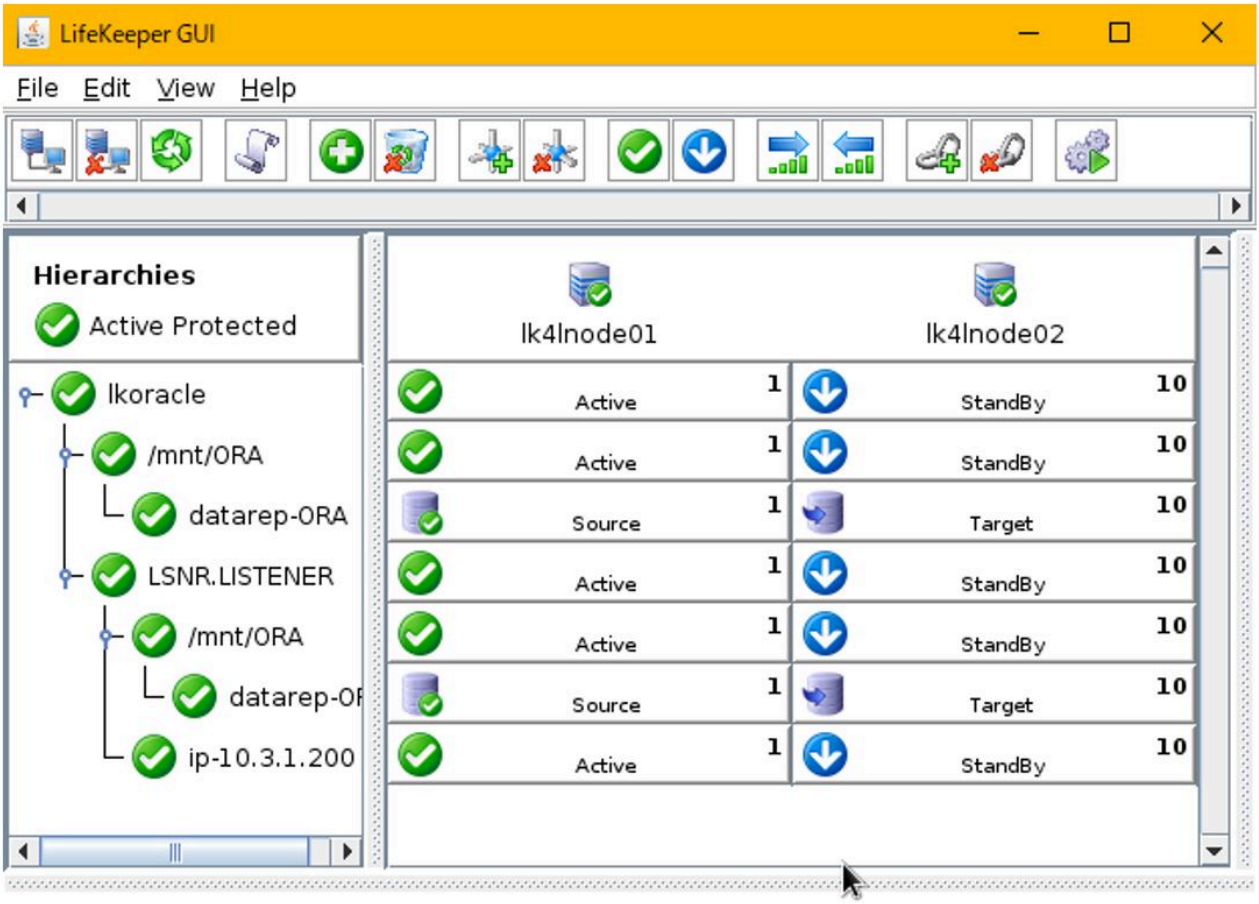



9. Extend is started. It is successful when “Hierarchy successfully extended” is displayed. Click **Finish**.

10. Click **Done** to exit the wizard.

11. The Oracle Database resource is created.

    Below is the created resource tree.




12. Undo the modifications in *etc/oratab*.

```
#   vi /etc/oratab
oracle:/mnt/ORA/app/oracle/product/12.1.0/dbhome_1:N ← change Y to N
```

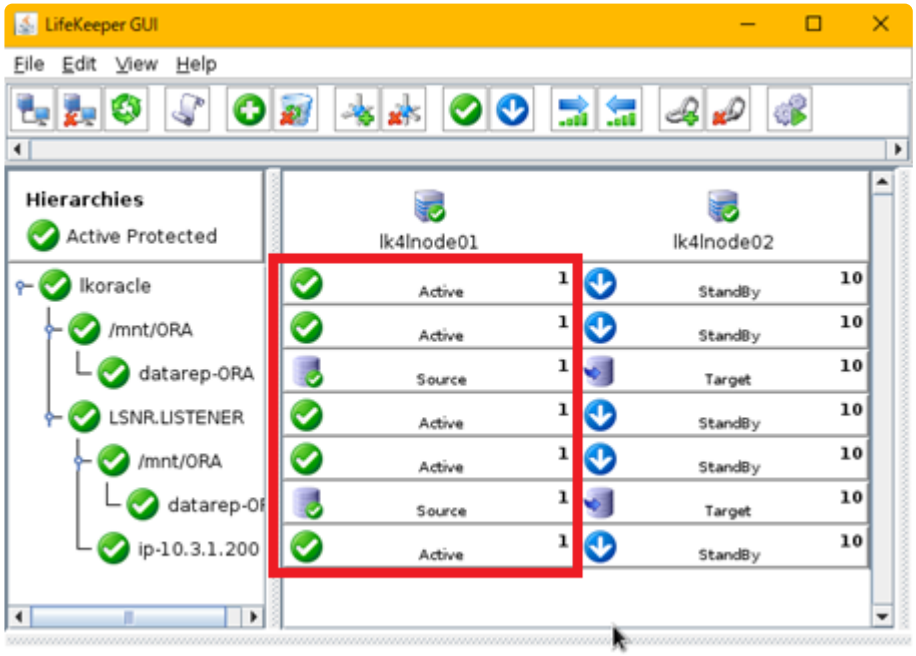
Refer to Creating an Oracle Resource Hierarchy for more information.

# 12.2.5.7. Connectivity Check

Check the connection to the database from the remote client.

1. **Install** the Oracle client software on the client.

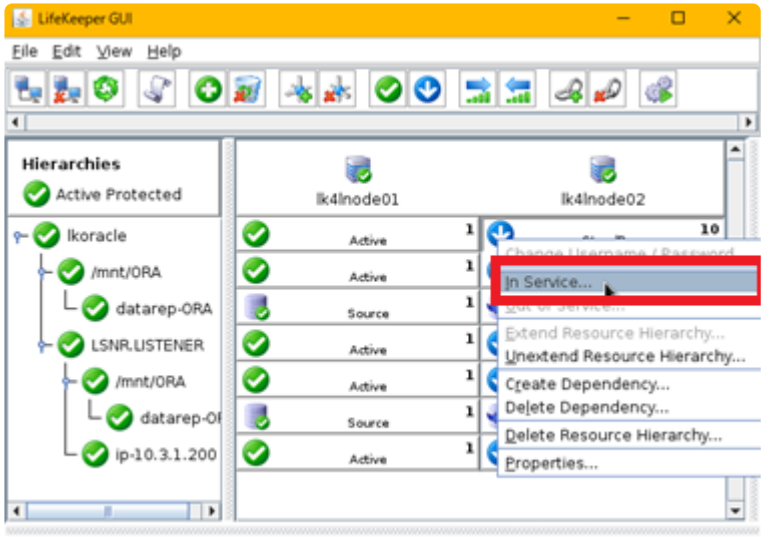
2. Ensure that the resource is active on the primary node.

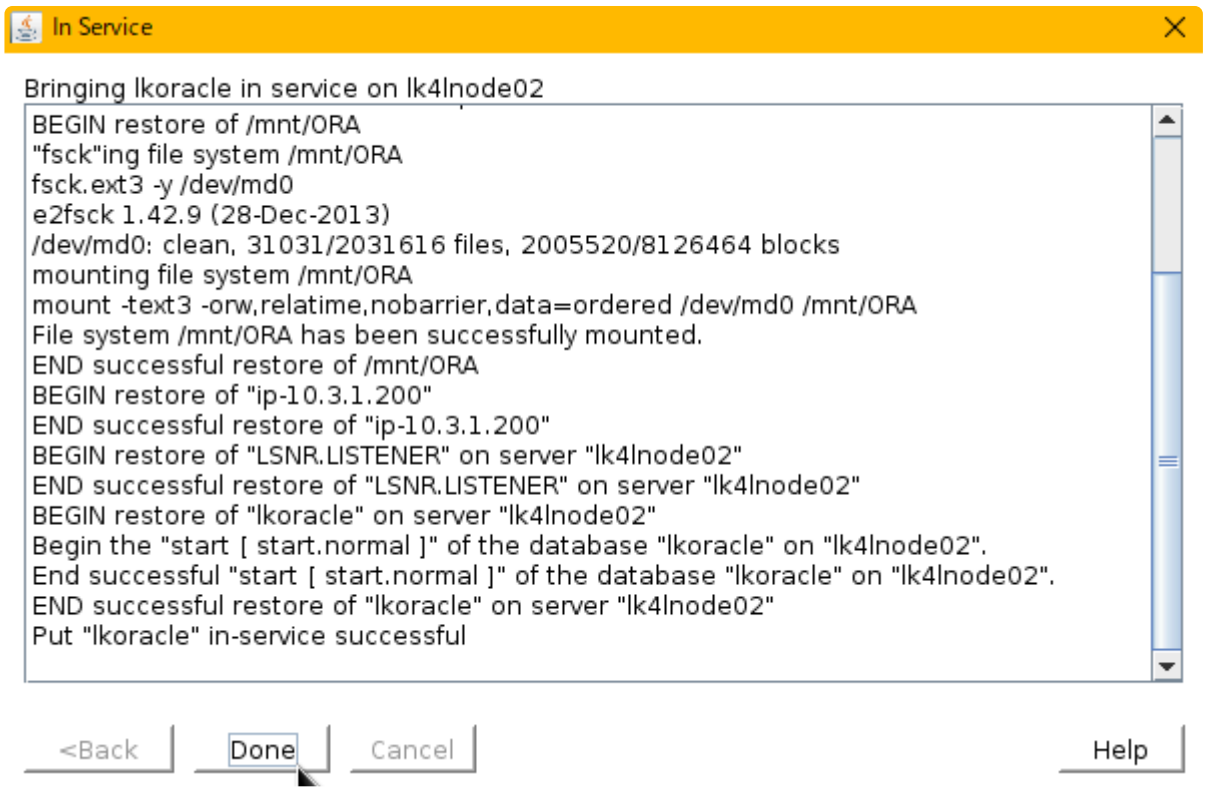



3. Connect to the database via the listener from the remote client using the virtual IP address for connection.

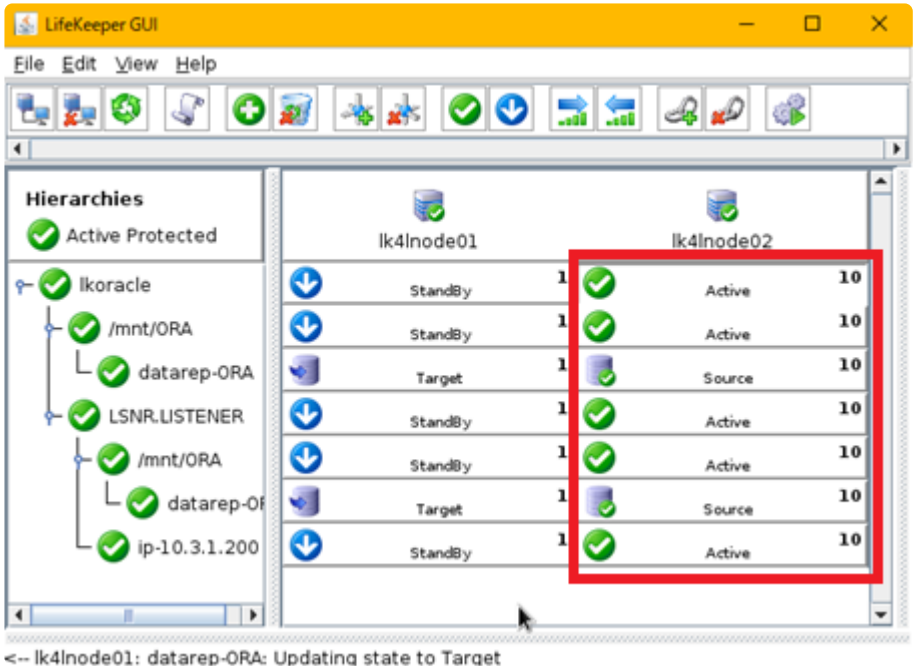
```
# hostname
lk4lclient
# sqlplus sys/XXXXXXXX@10.3.1.200:1521/lkoracle as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Sat May 25 17:21:32 2019
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL> select instance_name from v$instance ;

INSTANCE_NAME
----------------
lkoracle
```

4. Switch over the Oracle Database resource from the LifeKeeper GUI. Right-click the **Oracle Database** resource and select **In Service**.




5. The switchover is completed.



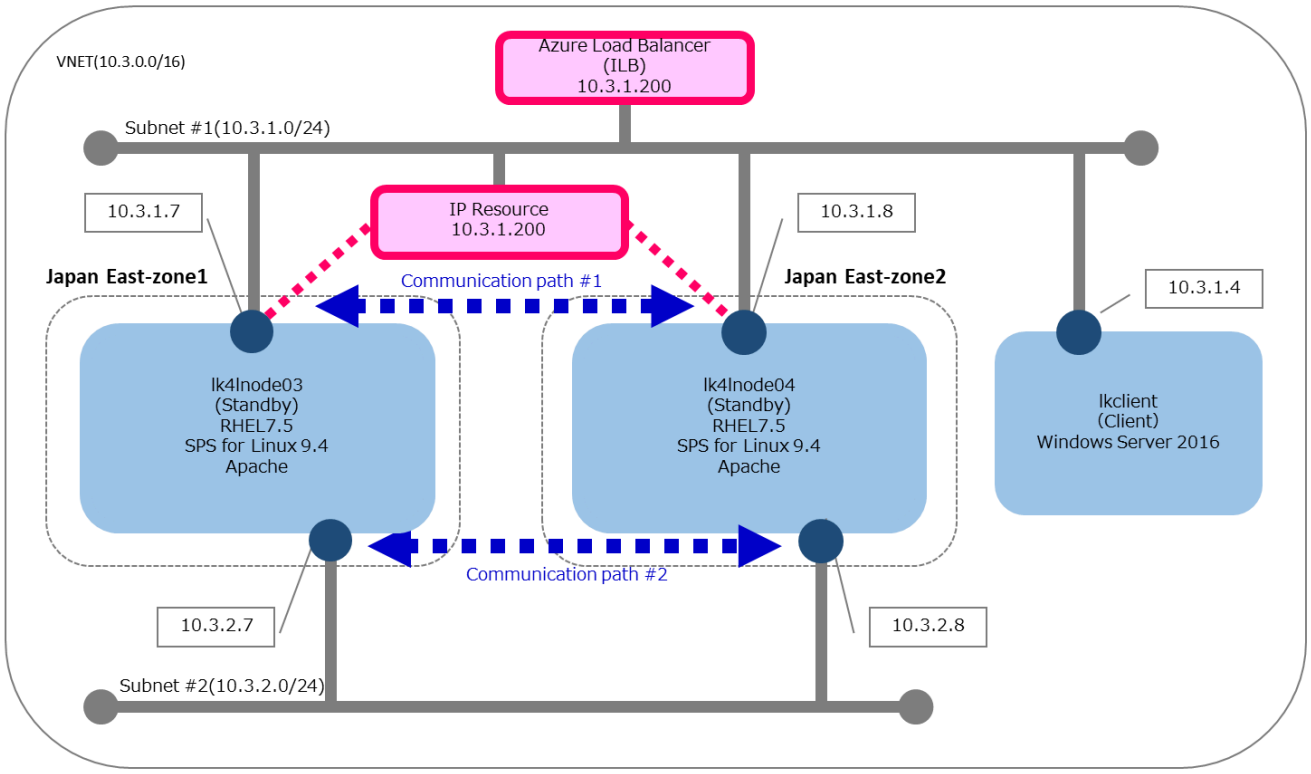

6. Connect from the remote client via the listener as follows (access with the cluster host name).

```
# hostname
lk4lclient
# sqlplus sys/XXXXXXXX@10.3.1.200:1521/lkoracle as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Sat May 25 17:21:32 2019
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL> select instance_name from v$instance ;

INSTANCE_NAME
----------------
lkoracle
```

7. It was confirmed that the same IP address (virtual IP address) can be used to connect to Oracle before and after switching.

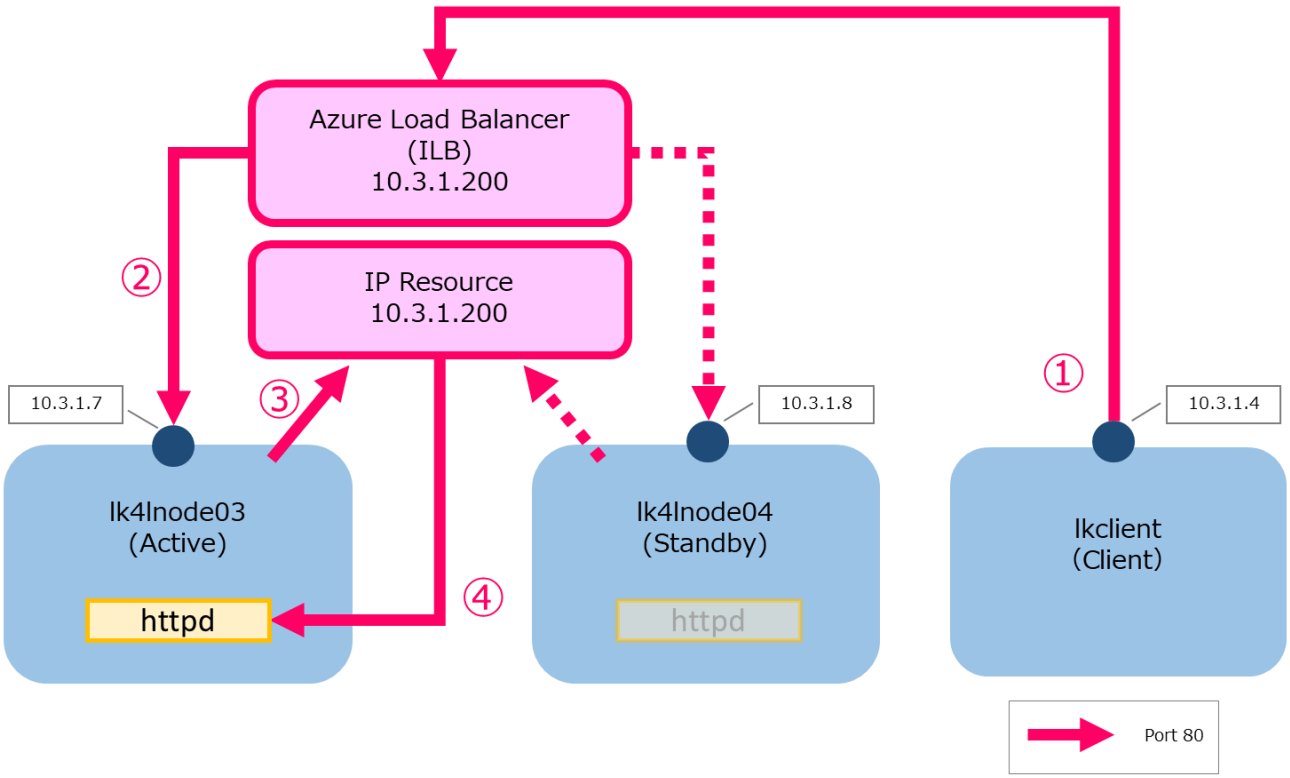
# 12.2.6. Availability Zone (High Availability Zone)

## Configuration for the verification




The application to be protected is Apache and we checked the connection from the client PC with a web browser. The active server (lk4lnode03) and the standby server (lk4lnode04) belong to different zones. The http request from the client (lkclient) is redirected to the active server virtual IP (10.3.1.200) via the ILB probing at 80.

# Overview of client redirection with the ILB




The outline of connecting steps from the client PC is as follows.

1. In order for the client to connect to httpd, start the connection to 10.3.1.200 (port 80) via the ILB.

2. In the ILB 10.3.1.7 (active node: lk4lnode03) and 10.3.1.8 (standby node: lk4lnode04) are registered in the load balancing destination (backend pool) without port setting. The packets received by the ILB for 10.3.1.200 (port 80) are forwarded to both the active and standby nodes.

3. Due to LifeKeeper specifications, it is necessary to specify [VIP protected with IP resources] for Apache resources. Create an IP resource (10.3.1.200) for receiving. Since the LifeKeeper IP resource (10.3.1.200) is in service only on the active node, only the active node (10.3.1.7) receives requests.

4. As a result, httpd (10.3.1.200: port80) of the active node receives the connection request from the client.

# 12.2.6.1. Azure Configuration

To use the Availability Zone, configure it when you create the VM as follows.

1. For the Region, select (Asia Pacific) Japan East.*1

2. Select Availability Zone in Availability options.

3. In Availability Zone, select 1, 2, or 3. The active node and the standby node should be in different zones: active (Zone 1) and standby (Zone 2).

*1 If you select a region such as Japan West where only the Availability Set is available, you cannot select “Availability Zone” in Availability options.

Since the protected application used for verification is Apache, set port 80 to probe in the ILB settings. Other configurations are the same as the one using the Availability Set.

The environment used to verify the configuration using the Availability Zone is as follows.

1. Network, load balancer

**[Network]**
VNet: 10.3.0.0/16
Subnet:

Subnet1: 10.3.1.0/24
Subnet2: 10.3.2.0/24

**[Load Balancer]**
Location: Japan East
SKU: Standard
Availability Zone: Zone redundant
FrontendIP:

Type: Private
Subnet: 10.3.1.0/24
IP Address: 10.3.1.200(Static)

BackendPool :

lk4lnode03
lk4lnode04

Health Probe:

Protocol: TCP

Port: 22

Load Balancing Rule:

Port: 80
Backend Port: 80

Inbound NAT rules：

0 rules

Session persistence: None
Floating IP(Direct Server Return): Enable

2. Virtual server

**[Active Node]**

Name: lk4lnode03
Operating system: Linux (RHEL 7.5)
Size: Standard A1 v2 (1 vcpus, 2 GiB memory)
Network Interface:

Private IP: 10.3.1.7
Public IP: 52.156.45.126

Installed Software:

httpd: httpd-2.4.6-80.el7_5.1.x86_64
iptables: iptables-1.4.21-24.1.el7_5.x86_64
LifeKeeeper: SPS for Linux v9.5.0 Build 108

IP tables:

```
...

-A PREROUTING -p tcp -m tcp – –dport 80 -j DNAT – –to-destination 10.3.1.200:80

-A POSTROUTING -s 10.3.1.200/32 -p icmp -j SNAT – –to-source 10.3.1.8

...

:OUTPUT ACCEPT [4188:806503]

-A INPUT -s 10.3.0.0/16 -p tcp -j ACCEPT

...
```

httpd.conf:

```
...

Listen 10.3.1.200:80•••

...

ServerName 10.3.1.200:80

...
```

**[Standby Node]**
Name: lk4lnode04
Network Interface:

Private IP: 10.3.1.8
Public IP: None

IP tables:

```
...

-A PREROUTING -p tcp -m tcp – –dport 80 -j DNAT – –to-destination 10.3.1.200:80

-A POSTROUTING -s 10.3.1.200/32 -p icmp -j SNAT – –to-source 10.3.1.7

...

:OUTPUT ACCEPT [4188:806503]

-A INPUT -s 10.3.0.0/16 -p tcp -j ACCEPT

...
```

Same as the active node for others.

**[Client Node]**
Name: lkclient
Operating System: Windows (Windows Server 2012 Datacenter)
Size: Basic A1 (1 vcpus, 1.75 GiB memory)
Network Interface:

Private IP: 10.3.1.4
Public IP: 104.41.171.59

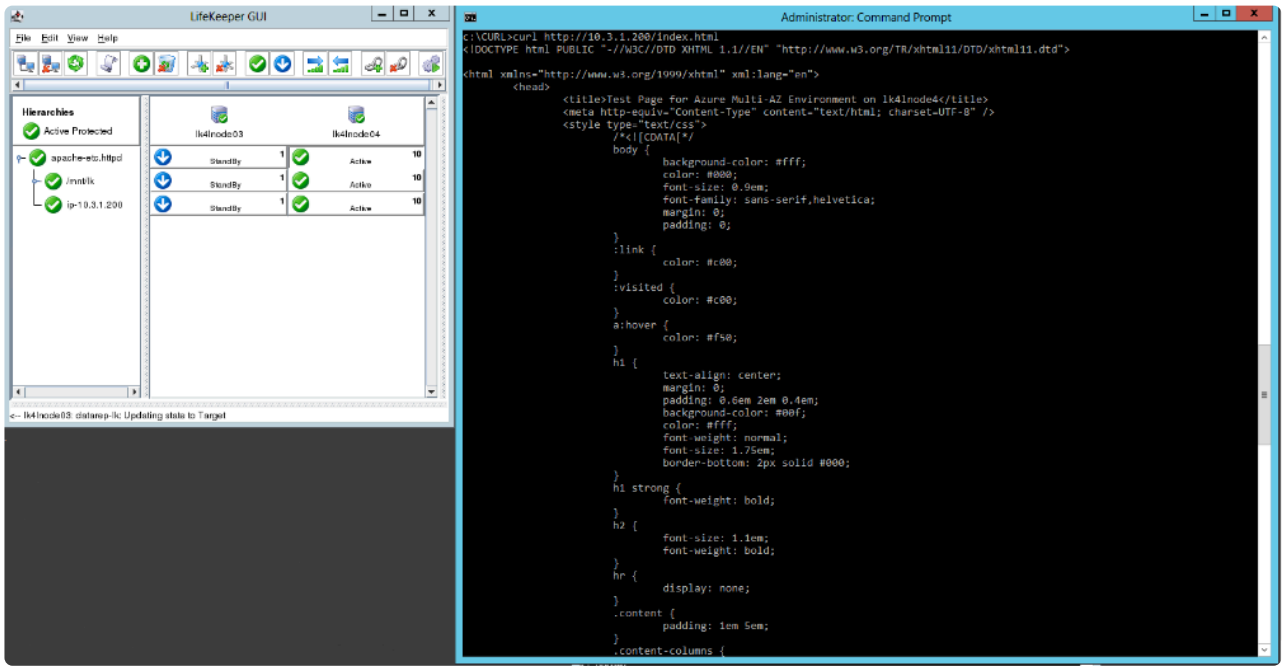
Installed Software:

Google Chrome： Version 80.0.3987.149 (Official Build) (64-bit)
Xming ：6.9.0.31
Tera Team： 4.102
WinSCP： 5.17.2

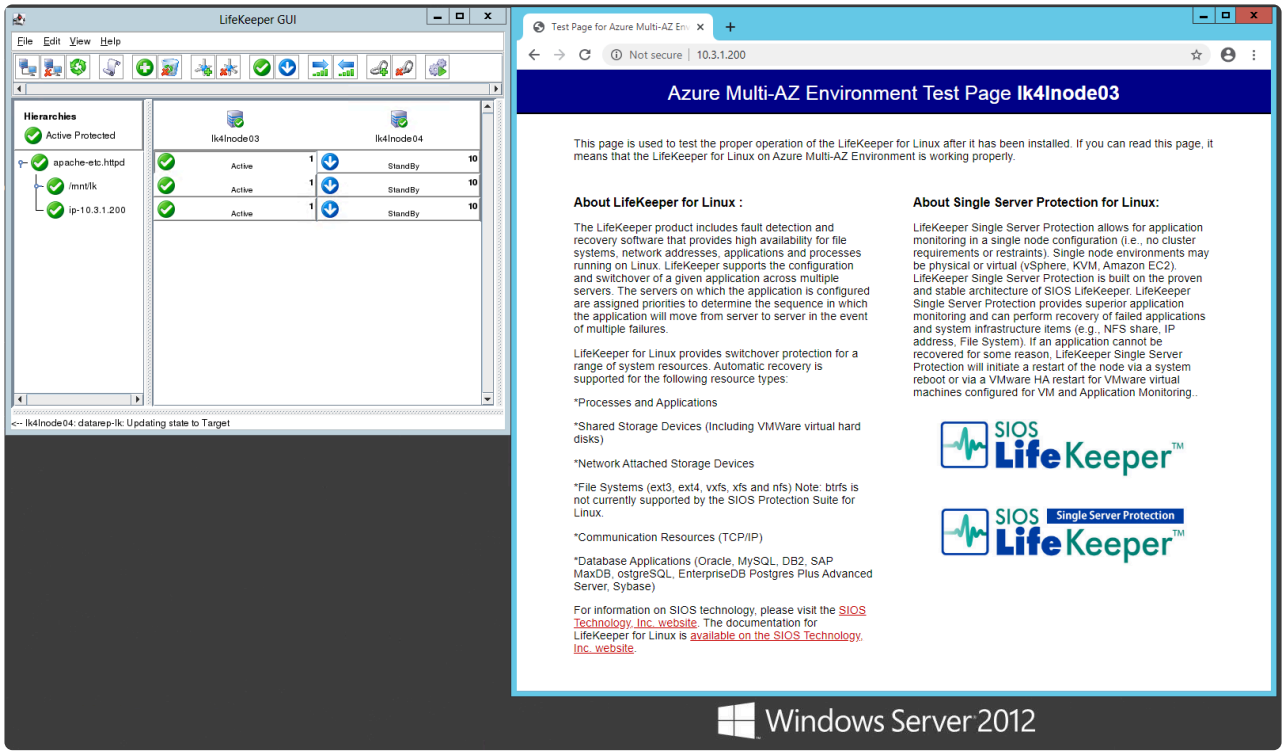
# 12.2.6.2. Checking the Client Redirection

We checked the access from lkclient using the following two methods and confirmed that we can connect to the virtual IP using both methods.

1. From the terminal of Windows Server2012 (client), accessed http://10.3.1.200/ with curl (command line based http access tool) and confirmed that the index.html of the active node is displayed. We confirmed that the index.html of the standby system was displayed along with the switchover.




2. On the client web browser, enter http://10.3.1.200/ to access the web page and confirm that index.html is displayed (clear the cache each time you check). We confirmed that the index.html of the switchover destination is displayed along with the switchover.


LifeKeeper GUI
File Edit View Help
Hierarchies
Active Protected
apache-etc.httpd
/mnt/lk
ip-10.3.1.200
lk4lnode03
lk4lnode04
Active
StandBy
lk4lnode04: datarep-lk: Updating state to Target
Test Page for Azure Multi-AZ Env
Not secure | 10.3.1.200
Azure Multi-AZ Environment Test Page lk4lnode03
This page is used to test the proper operation of the LifeKeeper for Linux after it has been installed. If you can read this page, it means that the LifeKeeper for Linux on Azure Multi-AZ Environment is working properly.
About LifeKeeper for Linux :
The LifeKeeper product includes fault detection and recovery software that provides high availability for file systems, network addresses, applications and processes running on Linux. LifeKeeper supports the configuration and switchover of a given application across multiple servers. The servers on which the application is configured are assigned priorities to determine the sequence in which the application will move from server to server in the event of multiple failures.
LifeKeeper for Linux provides switchover protection for a range of system resources. Automatic recovery is supported for the following resource types:
*Processes and Applications
*Shared Storage Devices (Including VMWare virtual hard disks)
*Network Attached Storage Devices
*File Systems (ext3, ext4, vxfs, xfs and nfs) Note: btrfs is not currently supported by the SIOS Protection Suite for Linux.
*Communication Resources (TCP/IP)
*Database Applications (Oracle, MySQL, DB2, SAP MaxDB, ostgreSQL, EnterpriseDB Postgres Plus Advanced Server, Sybase)
For information on SIOS technology, please visit the SIOS Technology, Inc. website. The documentation for LifeKeeper for Linux is available on the SIOS Technology, Inc. website.
About Single Server Protection for Linux:
LifeKeeper Single Server Protection allows for application monitoring in a single node configuration (i.e., no cluster requirements or restraints). Single node environments may be physical or virtual (vSphere, KVM, Amazon EC2). LifeKeeper Single Server Protection is built on the proven and stable architecture of SIOS LifeKeeper. LifeKeeper Single Server Protection provides superior application monitoring and can perform recovery of failed applications and system infrastructure items (e.g., NFS share, IP address, File System). If an application cannot be recovered for some reason, LifeKeeper Single Server Protection will initiate a restart of the node via a system reboot or via a VMware HA restart for VMware virtual machines configured for VM and Application Monitoring..
SIOS LifeKeeper™
SIOS Single Server Protection LifeKeeper™
Windows Server 2012

# 12.2.7. References and Acknowledgements

**SIOS Technology:**

SIOS Technical Documentation

SIOS Protection Suite for Linux 9.5.0 Technical Documentation

Oracle Recovery Kit Administration Guide

**External websites:**

Microsoft Azure
https://azure.microsoft.com/en-us/

Microsoft Azure Linux Virtual Machine Document
https://docs.microsoft.com/en-us/azure/virtual-machines/linux/

Microsoft Azure Blog
https://azure.microsoft.com/en-us/blog/

High Availability for a file share using WSFC, ILB and 3rd-party Software SIOS DataKeeper (used as a reference in ILB settings)
http://azure.microsoft.com/blog/2014/11/11/high-availability-for-a-file-share-using-wsfc-ilb-and-3rd-party-software-sios-datakeeper/

**Acknowledgements**

Microsoft Japan approved the use of Microsoft Azure for the verification. We appreciate their great support in building and configuring the environment and creating the test cases.

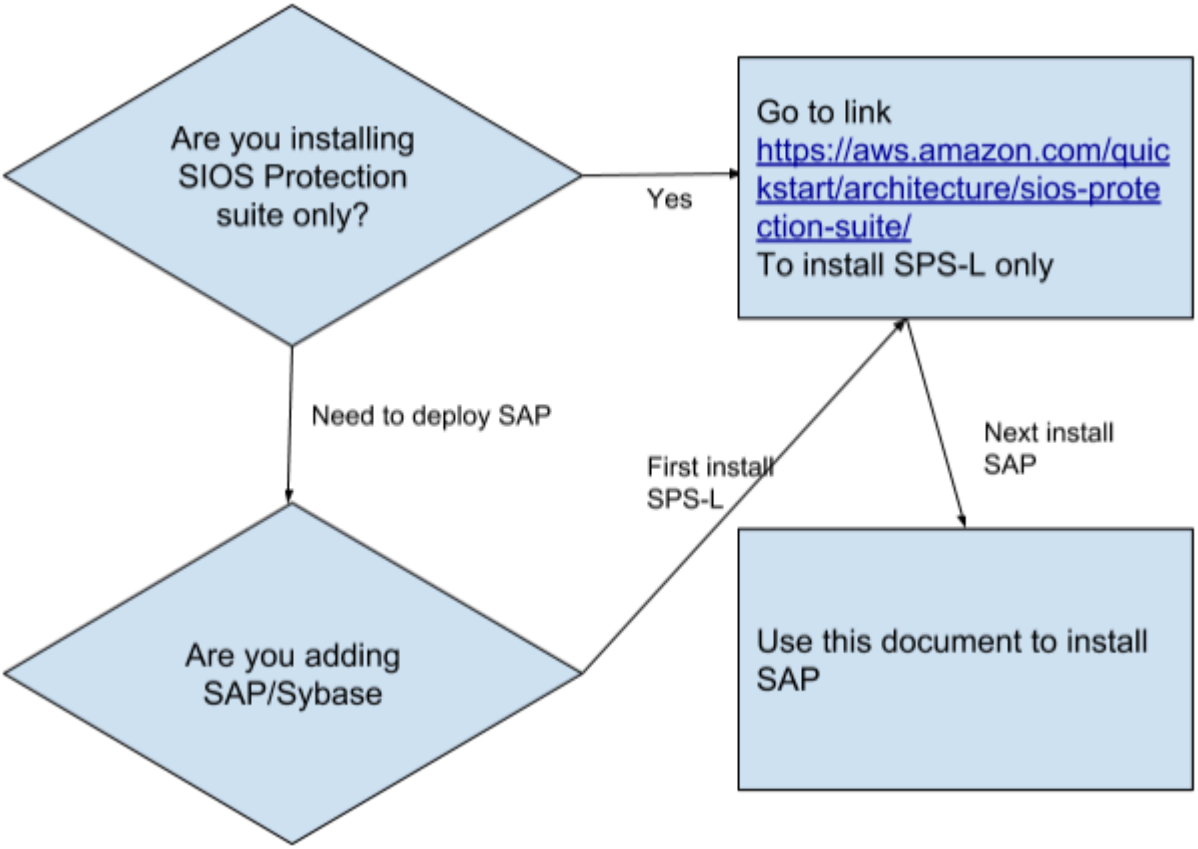
Microsoft Azure Website
https://azure.microsoft.com/en-us/

# 12.3. SIOS Protection Suite for Linux in the AWS Cloud (SAP)

## 概要

本書は、SIOS Protection Suite for Linux のインストール（SPS）のユーザーガイドです。

SPS for SAP 環境のインストール方法を理解するには、以下のマトリックスに従ってください。




注記：図中のリンクを こちら にも記載します。

# 12.3.1. SPS 上で SAP を設定する際の追加手順

SPS 上で SAP を設定するには、以下の手順に従ってください。

## ステップ 1. デプロイメントのテスト

SPS-L ノードに接続するには、Windows JumpBox に接続する必要があります。Windows マシンに接続するには、リモートデスクトップ端末に接続してください。

AWS コンソールで、作成された Windows JumpBox ノードを選択し、 **[Action]** の次に **[Connect]** をクリックします。これでリモートデスクトッププログラムをダウンロードして接続できます。また、マシンにログインするために必要なパスワードを複合化する必要があります。

Windows マシンに接続したら、PuTTY と VNC Viewer をダウンロードすることを推奨します。以下のサイトからダウンロードしてください。

- **Putty** – www.putty.org
- **VNC Viewer** – https://www.realvnc.com/en/connect/download/viewer/

これで、PuTTY を使用して各ノードのプライベート IP アドレスに接続することができます。また、同じプライベート IP アドレスを使用しているノードに接続するために VNC Viewer を使用することもできます。ノードへは Windows JumpBox の外ではアクセスできませんが、ノードは NAT ゲートウェイを使用してインターネットにアクセスできます。（ **注記：** NAT ゲートウェイに問題がある場合は、セキュリティグループのルール/メインルートを確認してください。）

いずれかのノードに接続したら、先ほどのテンプレートで作成したパスワードを使用して root で su を実行し、プログラム vncserver を実行します。これにより、VNC Viewer を使用してグラフィカルインターフェース内のそのノードに接続できます。

**VNC サーバーをインストールするためのコードスニペット**

次のオプションを指定して、vncserver コマンドを root として実行します。

パスワードを入力し、確認のために繰り返します
読み取り専用パスワードを **No** に設定します

オプションで /root/.vnc/config を編集し、以下の記述を追加します。
securitytypes=none
vncserver -kill:1

VNC へのアクセスは ipv4:5901 です。5901 は指定されたポート番号です。

デスクトップを右クリックして **Open Terminal** をクリックし、LifeKeeper GUI に接続するコマンド **/opt/LifeKeeper/bin/lkGUIapp** を入力します。事前に設定したパスワードで root としてログインしてください。2 つのノードが接続されていることがわかります。

この時点で、基本的な LifeKeeper 2 ノードがセットアップされました。LifeKeeper を使用して、SAP のインストールと SAP サービスの保護を続行します。

# ステップ 2. 仮想 IP の設定

ノード上で SAP が設定されたので、引き続き LifeKeeper を設定して SAP サービスとファイルシステムを保護します。

## Amazon AWS Elastic Compute Cloud (EC2) のセットアップ

AWS Command Line Interface（CLI）を各ノードにインストールする必要があります。詳細は、 AWS Command Line Interface のインストール を参照してください。すべての EC2 インスタンスは、HTTP および HTTPS を使用して Amazon EC2 サービスエンドポイントにアクセスできる必要があります。Amazon EC2 インスタンスのメタデータを取得するには、HTTP プロトコルを使用して IP アドレス 169.254.169.254 にアクセスしてください。

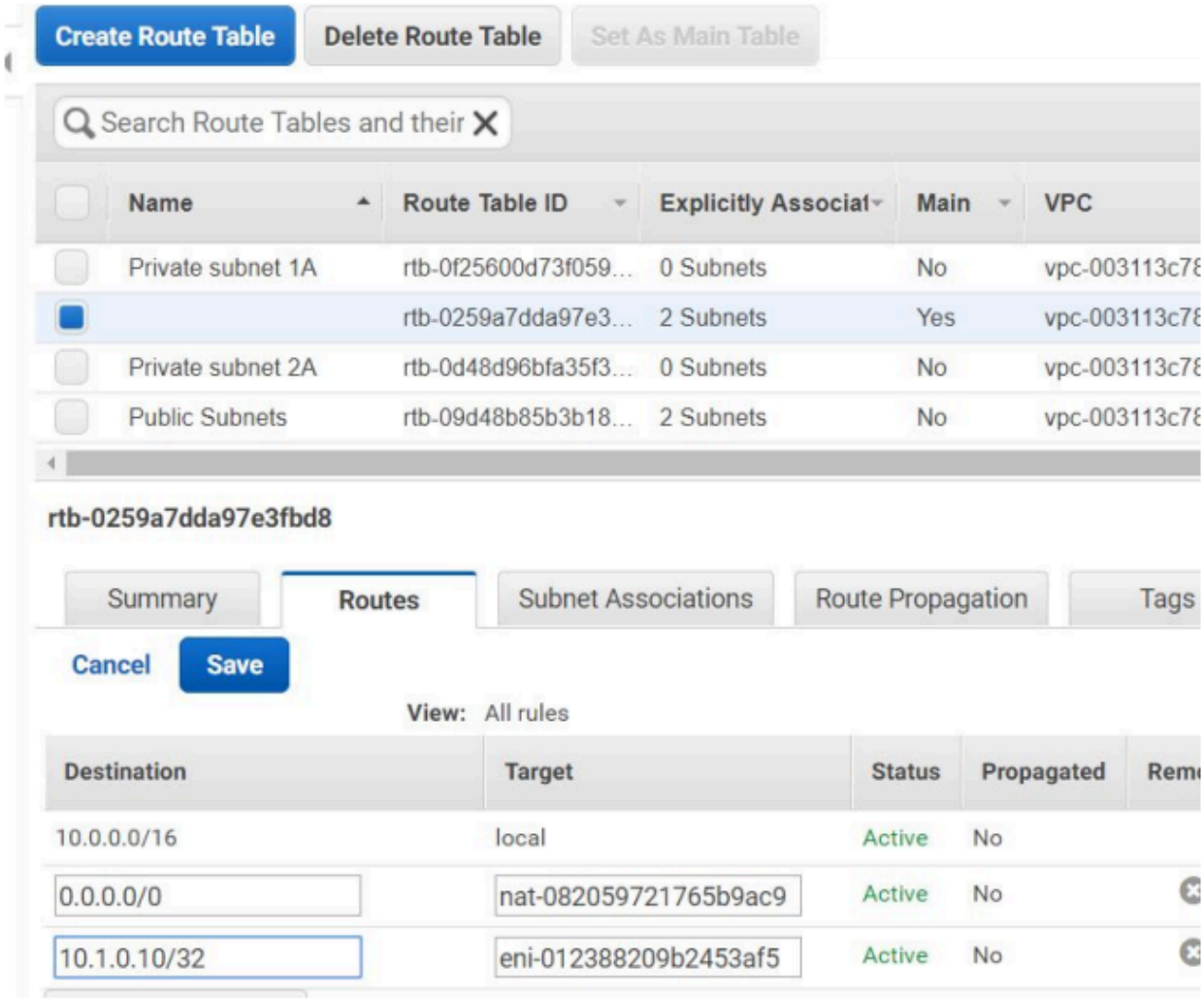
LifeKeeper が AWS を操作するためには、以下のアクセス権を持つ IAM ユーザーまたは IAM ロールが必要です。EC2 インスタンスの root ユーザーからアクセスできるように、 EC2 IAM ロール を設定するか、AWS CLIを適切に設定してください 。

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress
- ec2:DescribeRouteTables
- ec2:ReplaceRoute

## 仮想 IP リソースの作成

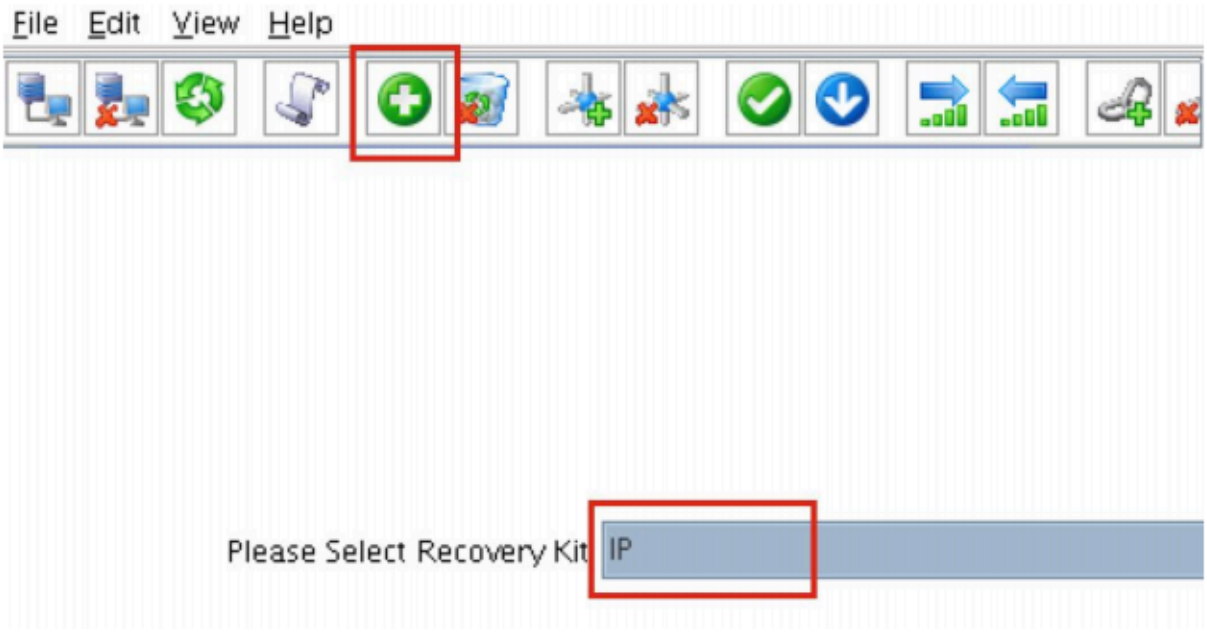
IP アドレスを決定します。IP アドレスは、ノードの現在の IP の CIDR ブロック範囲外の IP アドレスである必要があります。IP アドレスは、ノードの VPC ルートテーブルに配置してください。

以下の図では eni- ネットワークアダプタを使用して IP アドレスを 10.1.0.10/32 に設定し、ノードの 1 つに関連付けました。




/etc/default/LifeKeeper を編集して NOBCASTPING=1 を設定し、続行する前にブロードキャスト ping を無効にします。

**緑色のプラスアイコン** をクリックして、新規リソースを作成します。




ウィザードに従い、以下のように選択して IP リソースを作成します。

Select Recovery Kit: IP

Switchback Type: Intelligent

IP Resource: 10.1.0.10

Netmask: 255.255.255.0

Network Interface: eth0

IP Resource Tag: ip-10.1.0.10

以下のように選択して IP リソースを拡張します。

Switchback Type: Intelligent

Template Priority: 1

Target Priority: 10

IP Resource: 10.1.0.10

Netmask: 255.255.255.0

Network Interface: eth0

IP Resource Tag: ip-10.1.0.10

クラスターは次のようになり、ミラーリソースと IP リソースの両方が作成されます。

## ステップ 3. SAP のセットアップ

SAP ソフトウェアをダウンロードし、ノード上でセットアップします。SAP マーケットプレイスにアクセスして、各ノードに SAP ソフトウェアをダウンロードできます。

SAP のセットアップにはいくつかの選択肢があります。どれを実装するかは、コスト、経験、RAS（信頼性、可用性、サービサビリティ）などのさまざまな要因に依存します。

- NFS なしの ASCS の構成
- NFS を使用している ASCS + ERS 構成

各構成にはメリットとデメリットがあります。現場で SAP エキスパートと作業するか、SIOS プロフェッ

ショナルサービスに加入して、お使いの環境に最適なものを決定することをお勧めします。

**注記：** 今後作成するドキュメントでは、EFS（AWS Elastic File System）と Cloudwatch を使用したオールインワンの HANA のインストールについて詳しく説明します。自動インストールのクイックスタートスクリプトやインストールを管理するための SAP Landscape Manager（LaMa）の使用も計画しています。

# 12.3.2. NFS なしの ASCS 構成

NFS なしで ASCS をセットアップする手順は次の通りです。

- 一般的なセットアップのステップ概要
- SAP のインストール
- SAP リソース階層の作成

# 12.3.2.1. 一般的なセットアップのステップ手順

1. 仮想 IP を作成します。（先のステップで実施済み）
2. EC2 リソースを作成し、仮想 IP の依存関係として作成します。（先のステップで実施済み）
3. 「仮想 IP」に基づいた「仮想ホスト名」のノード 1 に SAP をインストールします。
4. ノード 1 で stopsap を実行します。
5. LifeKeeper GUI を使用して仮想 IP をノード 2 に対して「In Service」にし、「仮想 IP」に基づいてノード 2 の「仮想ホスト名」に SAP をインストールします。
6. ノード 2 で stopsap を実行し、両方のノード上でプロファイルファイルを変更します。（以下参照）
7. LifeKeeper GUI を使用して、仮想 IP をノード 1 で「In Service」にします。
8. SAP コンサルタントのアドバイスに従って、SAP に必要なマウントポイントの複製リソースを作成します。（先のステップで実施済み）
9. ノード 1 で Startsap を実行し、SAP が正常に動作していることを確認します。
10. /etc/default/LifeKeeper で、両方のノードの最後に次のエントリを追加します。

    SAP_EXPERTMODE=1

    SAP_NFS_CHECK_IGNORE=1

    SAP_DB_CHECK_IGNORE=1

11. SPS セットアッププログラムを再実行して、SAP Recovery Kit を追加します 。

    ./setup

    Recovery Kit Selection Menu から SAP Recovery Kit をインストールしてください。矢印キーを使用して使用可能な Recovery Kit のメニューから SAP Recovery Kit を選択し、<スペース>で決定し、<Enter> を押してインストールを続行します。

12. SAP Recovery Kit ガイド に従って SAP リソースを作成します。本書では簡素化した手順を説明します。

# 12.3.2.2. SAP のインストール

1. ASCS は、先のインストール手順でホストファイルに追加された「仮想 IP」に基づく「仮想ホスト名」に基づいてインストールする必要があります。まだ完了していない場合は、SAP のインストール前に必ず実行してください。

- ◦ SAP（具体的には ASCS または SCS）をインストールする場合、SAPINST_USE_HOSTNAME=vip オプションを指定する必要があります。 これは ERS では必要ありません。（ **注記：** SAPINST_USE_HOSTNAME 仮想 IP アドレスは後で LifeKeeper で SAP リソースを作成する際に使用するので、書き留めておいてください。）

  **注記：** `sapinst SAPINST_USE_HOSTNAME=vip` を指定してください。 **vip** は、ノード間で浮動する仮想IPです。

  以下を実行します。

  ./sapinst SAPINST_USE_HOSTNAME= *{hostname}*

- ◦ 7 段階にわけて、 **Core サービス** を作成して開始する必要があります。 *jdbcconnect.jar* で権限エラーが発生した場合は、 */sapmnt/STC/exe/uc/linuxx86_64* に移動し、このディレクトリでも *jdbcconnect.jar* を書き込み可能にしてください（chmod 777 —-）。

ASCS プロファイルは、/usr/sap、sapmnt を含むローカルマウントポイント、またはご利用の環境のSAP ファイルに必要なその他のローカルマウントポイントを指す必要があります。

エンキューサーバーおよびエンキューレプリケータープロセスが sapstart ユーティリティによって自動的に再起動されるのを防ぐために、ASCS および ERS インスタンスプロファイルを変更する必要があります。インスタンスプロファイルを更新した後、これらのインスタンスのSAP Start Serviceを再起動して、変更を有効にする必要があります。「 ASCS および ERS インスタンスプロファイル設定の変更 」に記載されている手順を実行し、その後このページに戻ってセットアッププロセスを続行してください。

2. ノード 1 で sapstop を実行して SAP を起動します。

3. LifeKeeper GUI を使用して、ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、ノード 2 を選択して IP をノード 2 に切り替えます。

4. ステップ 1 を繰り返してノード 2 に SAP をインストールし、正しく実行できることを確認します。

5. ノード 2 で sapstop を実行します。

6. LifeKeeper GUI を使用して ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、node1 を選択して IP をノード 1 に戻します。

7. ノード 1 で sapstart を実行し、正しく実行できることを確認します。

# 12.3.2.3. SAP リソース階層の作成

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。ドロップダウンメニューから、 **[Create Resource Hierarchy]** 選択します。

   クラスター内で認識されたすべてのインストール済みのリカバリーキットがドロップダウンリストで表示されます。ドロップダウンリストから **SAP** を選択します。

   **[Next]** をクリックします。

   いずれかのダイアログボックスで **[Back]** ボタンが有効になっている場合は、前のダイアログボックスに戻ることができます。これは、以前に入力した情報を修正する必要のあるエラーが発生した場合に便利です。

   階層の作成中に **[Cancel]** をクリックすると、LifeKeeper は作成プロセス全体をキャンセルします。

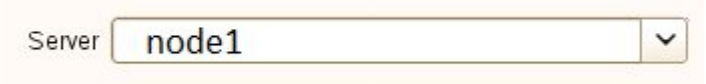
2. **スイッチバックタイプ** を選択します。これは、バックアップサーバーへのフェールオーバー後にサービスに復帰したときに SAP インスタンスをこのサーバーにどのようにスイッチバックするかを指示します。 **Intelligent** または **Automatic** のいずれかを選択できます。 **Intelligent** スイッチバックでは、インスタンスをプライマリ/元のサーバーに戻すために管理者の介入が必要です。 **Automatic** スイッチバック とは、プライマリサーバーがオンラインに戻り、LifeKeeper のコミュニケーションパスを再確立するとすぐにスイッチバックが実行されることを意味します。

   スイッチバックタイプは必要に応じて、 **[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。

   **[Next]** をクリックします。

3. SAP PAS、ASCS または SCS を配置するサーバーを選択します（通常、これをプライマリまたはテンプレートサーバーと呼びます）。クラスター内のすべてのサーバーがドロップダウンリストボックスに含まれています。

4. **SAP SID** を選択します。これは、保護対象の SAP PAS、ASCS、または SCS システムのシステム識別子です。

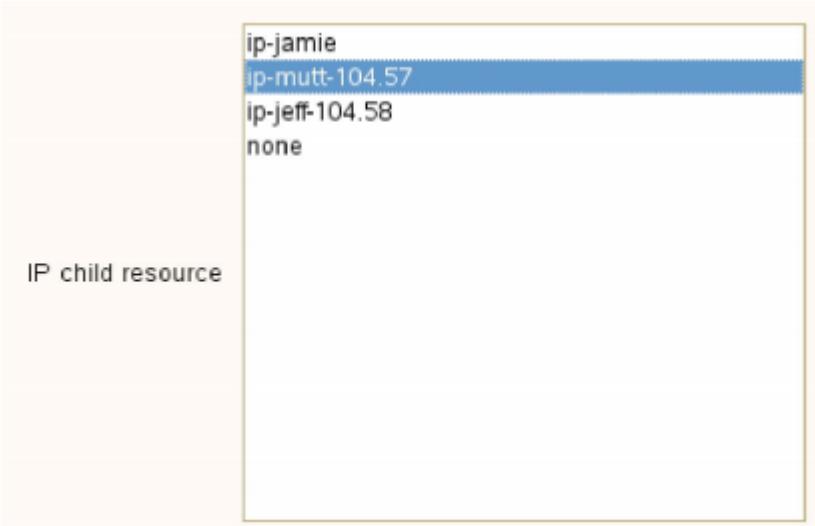
**[Next]** をクリックします。

5. 保護対象の SID の SAP インスタンス名（例：ASCS<*No.*>）（コアインスタンスが先頭）を選択します。

**[Next]** をクリックします。

**注記：** 保護およびリカバリーレベルのカスタマイズに関連する追加の画面が表示される場合があります。

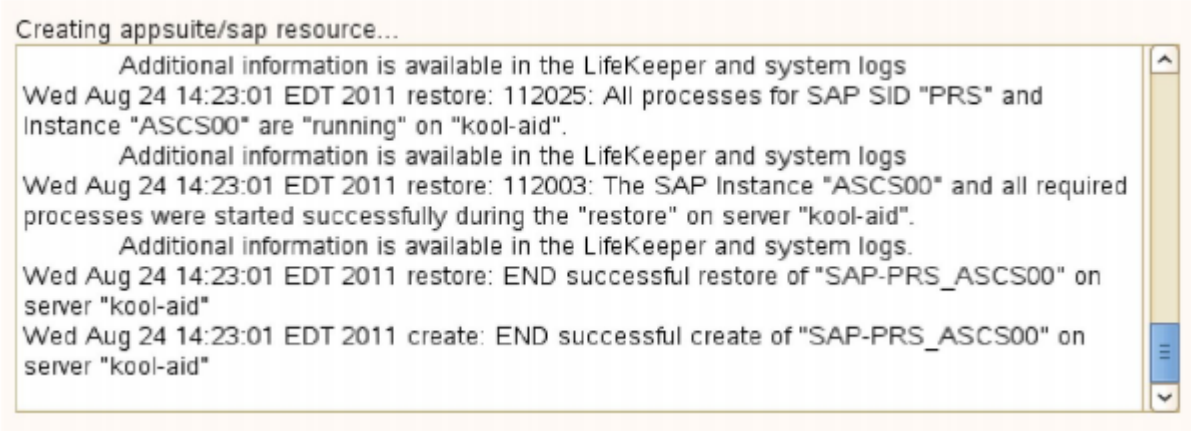
6. **IP の子リソース** を選択します。これは通常、SAP インストール時に指定された仮想ホスト IP アドレス（SAPINST_USE_HOSTNAME）またはフェールオーバーに必要な IP アドレスのいずれかです。




7. **SAP Tag** を選択または入力します。これは LifeKeeper が SAP の階層に付与するタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。デフォルトのタグは SAP-<SID>_<ID> です。

**[Create]** をクリックすると、 **SAP リソースの作成ウィザード** が SAP リソースを作成します。

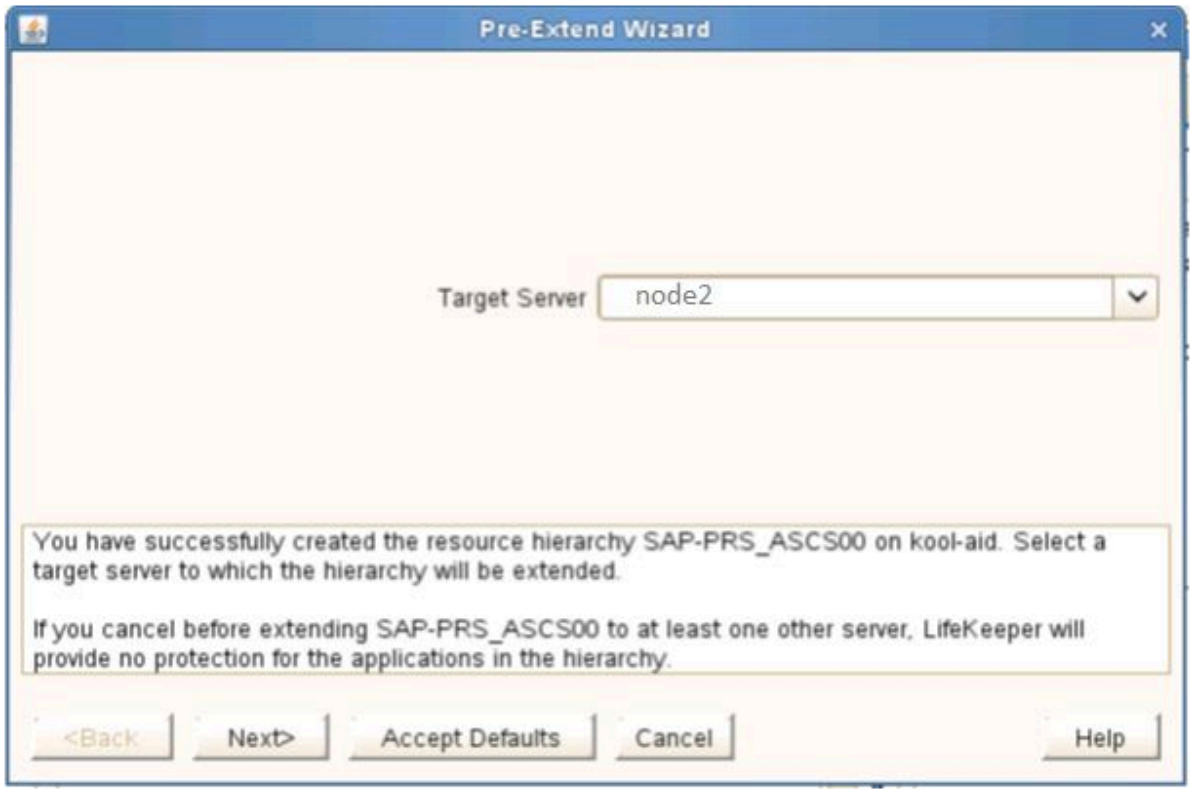
8. この時点で情報ボックスが表示され、LifeKeeper は SAP リソース階層を作成するための有効なデータが提供されているかどうかを検証します。LifeKeeper が問題を検出すると、情報ボックスに ERROR と表示されます。検証が成功すると、リソースが作成されます。情報ボックスに表示される SAP 起動スクリプトから出力されるエラーまたはメッセージが表示される場合もあります。




**[Next]** をクリックします。

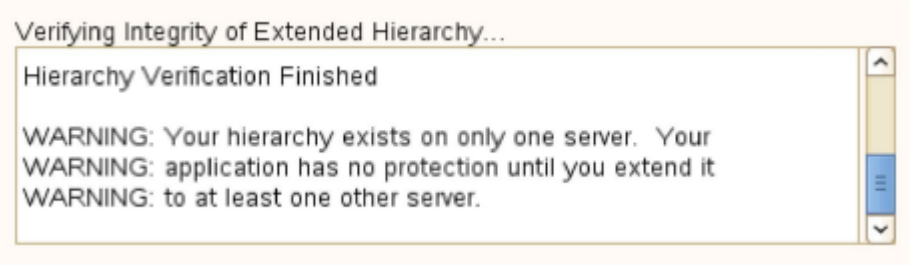
9. SAP リソース階層を正常に作成したことを示す別の情報ボックスが表示されます。そして、LifeKeeper 保護下に置くためにその階層をクラスター内の別のサーバーに **拡張** する必要があります。

**[Next]** をクリックすると、LifeKeeper はこのセクションで後述する ***Pre-Extend Wizard*** を起動します。

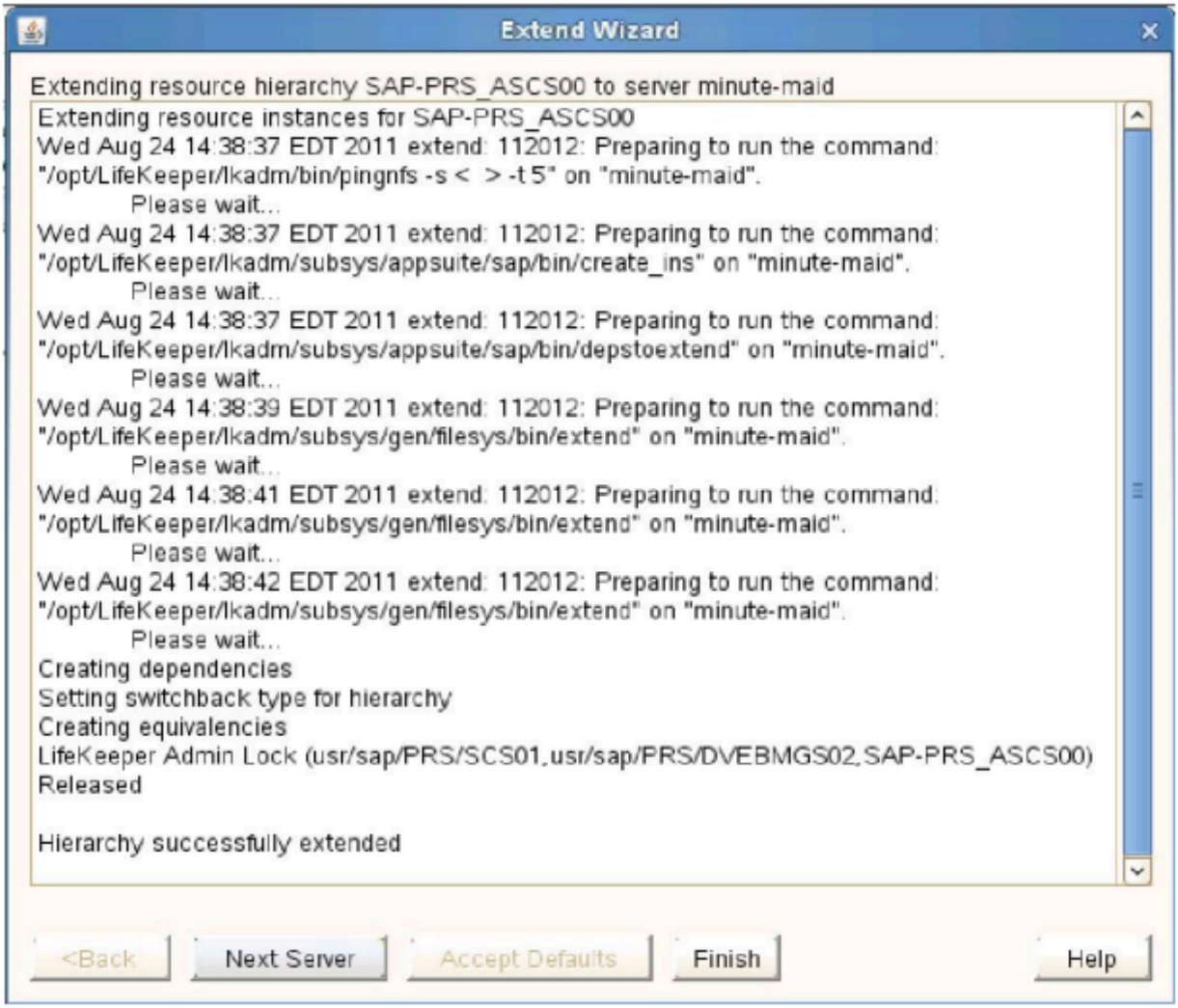



ここで **[Cancel]** をクリックすると、後で SAP リソース階層を LifeKeeper の保護下に置くにはこ

のステップに戻って SAP リソース階層を別のサーバーに拡張する必要があることを警告するダイアログボックスが表示されます。




10. 階層が正常に拡張された ことを示す **[Extend Wizard]** ダイアログが表示されます。 **[Finish]** をクリックします。




**[Hierarchy Integrity Verification]** ダイアログが表示されます。階層の検証が終了したら、 **[Done]** をクリックして **[Create Resource Hierarchy]** メニューを終了します。

**Hierarchy with the Core as the Top Level**

SIOS Protection Suite を使用して PAS および AAS サーバーを保護することはできますが、ほとんどのお客様は、それらを追加の HA なしで独立したスタンバイサーバーとして使用しているだけです。このガイドでは保護手順については説明しないので、詳細と手順については SAP Recovery Kit を参照してください。

# 12.3.3. NFS を使用している ASCS + ERS 構成

NFS を使用して ASCS + ERS をセットアップする手順は次の通りです。

- 一般的なセットアップのステップ概要
- SAP のインストール
- NFS のセットアップ
- NFS リソース階層の作成
- SAP リソース階層の作成
- ERS リソースの作成

# 12.3.3.1. 一般的なセットアップのステップ手順

1. 仮想 IP を作成し（先のステップでノード 1 で実施済み）、拡張します（先のステップで実施済み）。

2. EC2 リソースを作成し、仮想 IP の依存関係を作成します（先のステップで実施済み）。

3. 「仮想 IP」に基づく「仮想ホスト名」のノード 1 に SAP をインストールします。

4. ノード 1 で stopsap を実行します。

5. LifeKeeper GUI を使用して仮想 IP をノード 2 で「In-Service」にし、「仮想 IP」に基づく「仮想ホスト名」のノード 2 に SAP をインストールします。

6. ノード 2 で Stopsap を実行し、両方のノードでプロファイルファイルを変更します（下記参照）。

7. LifeKeeper GUI を使用して仮想 IP をノード 1 で「In-Service」に戻します。

8. SAP コンサルタントのアドバイスに従って、SAP に必要なマウントポイントのレプリケーションリソースを作成します（先のステップで実施済み）。

9. ノード 1 で startsap を実行し、SAP が正常に稼働していることを確認します。

10. 両方のノードの /etc/default/LifeKeeper で、次のエントリを最後に追加します。

    SAP_EXPERTMODE=1

    SAP_NFS_CHECK_IGNORE=1

    SAP_DB_CHECK_IGNORE=1

11. SPS セットアッププログラムを再実行して、SAP Recovery Kit を追加します。

    次のコマンドを使用して sps.img ファイルをマウントします（前述の手順に従ってダウンロードしてください）：

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

    PATH：イメージへのパス

    IMAGE_NAME：イメージ名

    MOUNT_POINT：マウントする場所へのパス

    マウントされた sps.img ディレクトリに移動し、次のように入力します：

    ./setup

Recovery Kit Selection Menu から SAP Recovery Kit をインストールしてください。SAP の Recovery Kit を選択するには、矢印キーを使用して選択して <スペースバー>を押して決定し、<Enter>を押してインストールを続行します。

12. NFS サーバーをセットアップします。

13. ファイルシステムを SAP サーバーにコピーし、冗長化とフェールオーバーのために、ファイルシステムにレプリケーションリソースを作成します。

14. NFS Recovery Kit のガイド に従って、NFS リソースを作成します。

    こちら では簡略化した手順を説明します。

15. SAP Recovery Kit のガイド に従って、SAP リソースを作成します。

    こちら では簡略化した手順を説明します。

# 12.3.3.2. SAP のインストール

1. ASCS と ERS は、「仮想 IP」の「仮想ホスト名」に基づいてインストールする必要があります。これは、以前のインストール手順でホストファイルに追加されています。まだ完了していない場合は、SAP のインストール前に必ず実行してください。

- ◦ SAPinst を起動する前に SAPinst プロパティ SAPINST_USE_HOSTNAME を設定し、必要な仮想ホスト名を指定します。（**注記：** SAPINST_USE_HOSTNAME の仮想 IP アドレスをメモしておいてください。LifeKeeper で SAP リソースを作成する際に後で使用します。）

  **注記：** `sapinst SAPINST_USE_HOSTNAME=vip` を指定してください。 **vip** は、ノード間で浮動する仮想IPです。

  以下を実行します。

  ./sapinst SAPINST_USE_HOSTNAME= *{hostname}*

- ◦ 7 段階にわけて、 **Core サービス** を作成して開始する必要があります。 *jdbcconnect.jar* で権限エラーが発生した場合は、 */sapmnt/STC/exe/uc/linuxx86_64* に移動し、このディレクトリでも *jdbcconnect.jar* を書き込み可能にしてください（chmod 777 —-）。

エンキューレプリケーションは、SAP のドキュメントおよびとベストプラクティスに基づいて構成し、動作を確認する必要があります。

2. ASCS および ERS プロファイルは、/usr/sap、sapmnt を含むローカルマウントポイント、またはご利用の環境の SAP ファイルに必要なその他のローカルマウントポイントを指す必要があります。実際のファイルは、インストールおよび構成後に NFS マウントポイントに移動されます。

   エンキューサーバーおよびエンキューレプリケータープロセスが sapstart ユーティリティによって自動的に再起動されるのを防ぐために、ASCS および ERS インスタンスプロファイルを変更する必要があります。インスタンスプロファイルを更新した後、これらのインスタンスのSAP Start Serviceを再起動して、変更を有効にする必要があります。「 ASCS および ERS インスタンスプロファイル設定の変更 」に記載されている手順を実行し、その後このページに戻ってセットアッププロセスを続行してください。

3. ノード 1 で SAP の sapstop を実行します。

4. LifeKeeper GUI を使用して、ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、ノード 2 を選択して IP をノード 2 に切り替えます。

5. ステップ 1 を繰り返してノード 2 に SAP をインストールし、正しく実行できることを確認します。

6. ノード 2 で SAP の sapstop を実行します。

7. LifeKeeper GUI を使用して、ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、ノード 1 を選択して IP をノード 1 に切り替えます。

8. ノード 1 で SAP の sapstop を実行し、正しく実行できることを確認します。

# 12.3.3.3. NFS のセットアップ

SIOS 製品をインストールする前に、両方のクラスターノードに NFS サーバーをインストールしておく必要があります。

お使いの SAP 設計での SAP 要件に基づいて NFS エクスポートを作成します。以下は参考例であり、お使いの SAP 環境を表すものではありません。

LifeKeeper は inode を使用して NFS 共有情報を管理します。したがって、すべての NFS 共有には固有の inode が必要です。すべてのファイルシステムの root ディレクトリは同じ inode を持つので、LifeKeeper で保護するためには、NFS 共有は root から少なくとも 1 つ下のディレクトリレベルでなければなりません。

たとえば、上記の情報を参照すると、 */usr/sap/trans* ディレクトリが SAP サーバー上で NFS 共有である場合、共有ストレージデバイス上に */trans* ディレクトリが作成され、共有ストレージデバイスを */usr/sap* としてマウントすることが求められます。 しかし、 */usr/sap* 配下にあるすべてのファイルをこの配列で必要な共有ストレージに置くことは必ずしも望ましいとは限りません。

この問題を回避するには、NFS 共有のディレクトリを含むすべての共有ファイルシステムをマウントするための */exports* ディレクトリツリーを作成するか、SAP ディレクトリと */exports* ディレクトリの間にソフトリンクを作成することを推奨します。または、NFS 共有ディレクトリをローカルで NFS マウントすることを推奨します。（ **注記：** ここで */exports* と記載しているディレクトリの名前は、ユーザーの設定よって異なる場合があります。ただし、わかりやすくするために、本書では全体を通じてこのディレクトリを */exports* と記載します。）

たとえば、SAP プライマリサーバーでのこの例のディレクトリとリンク/マウントは、次のようになります。

| **For the */usr/sap/trans* share** | |
|---|---|
| **Directory** | **Notes** |
| */trans* | created on share file system and shared through NFS |
| */exports/usr/sap* | mounted to / (on shared file system) |
| */user/sap/trans* | soft linked to */exports/usr/sap/trans* |

<sapmnt>/<SAPSID> 共有のディレクトリとリンクは次のようになります。

| **For the <sapmnt>/<SAPSID> share** | |
|---|---|
| **Directory** | **Notes** |
| /<SAPSID> | created on shared file systems and shared through NFS |
| */exports/sapmnt* | mounted to / (on shared file system) |
| <sapmnt>/<SAPSID> | NFS mounted to *<virtual SAP server>:/exports/sapmnt/<SAPSID>* |

## ローカル NFS マウント

LifeKeeper 環境における SAP の推奨ディレクトリ構造では、1 つ以上の SAP システムディレクトリに対してローカルにマウントされた NFS 共有が必要です。

ローカルにマウントされた NFS 共有のいずれかの NFS エクスポートポイントが使用できなくなった場合、エクスポートポイントが再び利用可能になるのを待っている間にシステムがハングすることがあり、システムの再起動を含め、多くのシステム操作が正しく機能しなくなります。SAP クラスター用の NFS サーバーは LifeKeeper で保護されている必要があり、ローカルマウントポイントが存在する間は手動でサービスを停止しないでください。

## <INST>ディレクトリの場所

/usr/sap/<SAPSID> パスは NFS 共有ではないため、ファイルシステムのルートディレクトリにマウントできます。/usr/sap/<SAPSID> パスには、SYS サブディレクトリと、サーバー上で実行できる各 SAP インスタンスの <INST> サブディレクトリが含まれています。構成によっては <INST> ディレクトリは 1 つしかないので、共有ファイルシステムの /usr/sap/<SAPSID> の下に置いても問題ありません。

ただし他の構成では、バックアップサーバーにローカル AS インスタンスが含まれている可能性があります。このローカル AS インスタンスの <INST> ディレクトリは常時利用できるわけではなく、共有ファイルシステム上に置くべきではありません。この問題を解決するには、ある構成については、ローカルサーバーに配置する必要のある AS の /usr/sap/<SAPSID>、/usr/sap/<SAPSID>/SYS および /usr/sap/<SAPSID>/<AS-INST> の代わりに、PAS、ASCS または SCS の /usr/sap/<SAPSID>/<INST>、/usr/sap/<SAPSID>/<ASCS-INST> または /usr/sap/<SAPSID>/<SCS-INST> ディレクトリを共有ファイルシステムにマウントすることを推奨します。

たとえば、次のディレクトリとマウントポイントを ABAP + Java 構成用に作成する必要があります。

| Directory | Notes |
|---|---|
| *usr/sap/<SAPSID>/DVEBMS<**No.**>* | mounted to / (Replicated non-NFS file system ) |
| *usr/sap/<SAPSID>/SCS<**No.**>* | mounted to / (Replicated non-NFS file system ) |
| *usr/sap/<SAPSID>/ERS<**No.**>*<br><br>*(for SCS instance)* | should be locally mounted on all cluster nodes or mounted from a NAS share (*should not be mounted on shared storage*) |
| *usr/sap/<SAPSID>/ASCS<**Instance No.** >* | mounted to / (Replicated from non-NFS file system) |
| *usr/sap/<SAPSID>/ERS<**No.**>* | should be locally mounted on all cluster nodes or mounted from a NAS share (*should not be mounted on shared storage*) |

| | |
|---|---|
| *(for ASCS instance)* | |
| *usr/sap/<SAPSID>AS<**Instance No.** >* | created for AS backup server |

## NFS のマウントとファイルシステムの移動

メイン SAP ファイルシステムのマウントポイントが作成されたら、必要に応じてマウントします。この時点で、これらの手順に進む前にすべての SAP サービスを停止します。

**mount /dev/sap/sapmnt /exports/sapmnt**

**mount /dev/sap/saptrans /exports/saptrans**

## データの NFS への移動

1. */etc/exports* を編集し、SAP のメインディレクトリのマウントポイントを挿入します。

```
/exports/sapmnt * (rw,sync,no_root_squash)

/exports/saptrans * (rw,sync,no_root_squash)
```

## NFS エクスポートの例

各 <nfsvip>を、その NFS 共有で使用される適切な仮想 IP に置き換えます。SAP システムの設計に応じて、異なる仮想 IP を使用して異なるファイルシステムを共有できます。

# more /etc/exports

/exports/sapmnt 10.2.0.69(rw,sync,all_squash,anonuid=0,anongid=1001)

/exports/sapmnt 10.2.0.11(rw,sync,all_squash,anonuid=0,anongid=1001)

/exports/usr/sap/<instance name>/ASCS01 10.2.0.69(rw,sync,all_squash,anonuid=0,anongid=1001)

/exports/sap/<instance name>/ASCS01
10.2.0.11(rw,sync,all_squash,anonuid=0,anongid=1001)

# more /etc/fstab

#

# /etc/fstab

# Created by anaconda on Mon Nov 9 20:20:10 2015

#

# Accessible filesystems, by reference, are maintained under '/dev/disk'

# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info

#

UUID=367df610-4210-4a5a-8c8d-51ddf499fc17 / xfs defaults 0 0

/dev/xvdb swap swap defaults 0 0

/dev/xvdc /tmp xfs nodev,nosuid,noexec,relatime 0 0

/dev/xvdp1 /var xfs defaults 0 0

/dev/xvdp2 /var/log xfs defaults 0 0

/dev/xvdp3 /var/log/audit xfs defaults 0 0

/dev/xvdp4 /home xfs defaults,nodev 0 0

/tmp /var/tmp none bind,nodev,nosuid 0 0

/dev/xvdj /usr/sap xfs defaults 0 0

/dev/xvdg /exports/usr/sap/P4G/ASCS01 xfs defaults 0 0

/dev/xvdh /usr/sap/P4G/D00 xfs defaults 0 0

/dev/xvdi /sapcd xfs defaults 0 0

/dev/xvdk /exports/sapmnt xfs defaults 0 0

**<nfsvip>**:/exports/usr/sap/P4G/ASCS01 /usr/sap/**<instance name>**/ASCS01 nfs nfsvers=3,proto=udp,rw,sync,bg 0 0

**<nfsvip>**:/exports/sapmnt /sapmnt nfs nfsvers=3,proto=udp,rw,sync,bg 0 0

**<nfsvip>**:/exports/usr/sap/P4G/ASCS01 /usr/sap/PG4/ASCS01 nfs nfsvers=3,proto=udp,rw,sync,bg 0 0 (**Note:** This ERS entry will only be present if using an ERSv2 configuration with a shared ERS filesystem.)

2. `systemctl start nfs-server.service` コマンドを使用して NFS サーバーを起動します。NFS サーバーがすでにアクティブな場合は、"`exportfs -va`" を実行してこれらのマウントポイ

ントをエクスポートします。

3. ノード 1 および 2 の両方で、次のマウントコマンドを実行します（**udp の使用方法に注意してください。これはフェールオーバーとリカバリーで重要です**）。これにより、NFS 共有をマウントできるようになります。

   mount {virtual ip}:/exports/sapmnt/<PG4> /sapmnt/<PG4> -o rw,sync,bg,udp

   mount {virtual ip}:/exports/saptrans /usr/sap/trans -o rw,sync,bg,udp

4. ノード 1 から、必要なファイルシステムを、/usr/sap および/sapmnt またはその他の必要なファイルから、NFS サーバーからノード 1 にマウントされた NFS マウントポイントにコピーします。

5. SAP にログインし、SAP を起動します（su の後に stcadm を実行します）。

   ```
   startsap sap{No.}
   ```

6. すべてのプロセスが開始されていることを確認してください。

   ```
   ps -ef | grep en.sap (2 processes)

   ps -ef | grep ms.sap (2 processes)

   ps -ef | grep dw.sap (17 processes)
   ```

   「SAP Logon」または「SAP GUI for Windows」は、SAP が提供する Windows クライアントです。プログラムは SAP のダウンロードサイトからダウンロードできます。仮想 IP アドレスは **[Properties]** ページで「アプリケーションサーバー」として使用できます。これにより、仮想 IP が存在するプライマリマシンへの接続がアクティブになります。

7. まだ実行していない場合は、NFS 共有マウントポイントにデータレプリケーションクラスターリソースを作成して、ノード 1 からノード 2 にデータを複製します。

# 12.3.3.4. NFS リソース階層の作成

複数の NFS 共有の複数のリソース階層を作成するには、以下の手順を繰り返す必要があります。プライマリサーバーからリソースインスタンスを作成するには、次の手順を実行してください。

1. LifeKeeper GUI メニューから **[Edit]** → **[Server]** を選択し、ドロップダウンメニューから **[Create Resource Hierarchy]** を選択します。

   すでに選択されている内容を変更する場合や、NFS リソース階層の作成中にエラーメッセージが表示された場合は、 **[Back]** ボタンを使用して選択内容を変更するか、または修正します（ **[Back]** ボタンが有効な場合）。

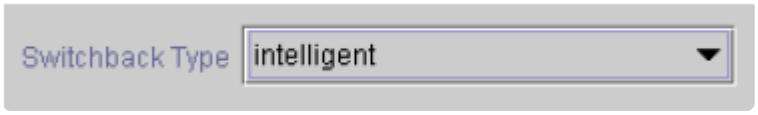
   ダイアログボックスに、クラスター内にインストールされ認識されているすべての Recovery Kit の一覧がドロップダウンメニューで表示されます。ドロップダウンメニューから **NFS** を選択します。

   **[Next]** をクリックして次のダイアログボックスに進みます。

   **注記：** 階層の作成中に **[Cancel]** ボタンをクリックすると、LifeKeeper は作成プロセス全体をキャンセルします。

2. **スイッチバックタイプ** を選択します。これは、バックアップサーバーへのフェールオーバー後にサービスに戻るときに、NFS インスタンスがプライマリサーバーに戻される方法を指示します。 **Intelligent** または **Automatic** のいずれかを選択してください。 **Intelligent** スイッチバック では、インスタンスをプライマリ/元のサーバーに戻す際に管理者の介入が必要です。 **Automatic** スイッチバック では、プライマリサーバーがオンラインに戻って LifeKeeper 保護下に入るとすぐにスイッチバックが行われます。

   スイッチバックタイプは、必要に応じて、 **[Resource Properties]** ダイアログボックスの **[General]** タブから後で変更できます。

3. NFS リソースを作成する **サーバー** を選択します（通常、これはプライマリまたはテンプレートサーバーと呼ばれます）。クラスター内のすべてのサーバーがドロップダウンメニューに含まれています。

   **[Next]** をクリックして次のダイアログボックスに進みます。

4. **Export Point** ダイアログには、次の基準を満たしている NFS ファイルシステムのエクスポートポイントのドロップダウンリストが表示されます。

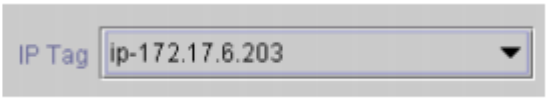
- エクスポートポイントは NFS によってエクスポートされている。
- エクスポートポイントは共有ドライブ上にある。
- 基礎となるファイルシステムが LifeKeeper で保護されている場合、そのファイルシステムは In-Service 状態で、 **サーバー** ダイアログで選択したサーバーで優先度の最も高いファイルシステムである必要があります。

- NFSv4 の基準：
  - NFS v4 のルートエクスポートがすでに保護されている場合は、選択肢はありません（v4 が 1 つのみで存在する必要があります。v4 と v2/v3 が混在すると保護できません）。
  - NFS v2/v3 がすでに保護されている場合は、選択肢に NFS v4 はリストされません。
  - 何も保護されていない場合、リストには v2/v3 と v4 の両方が含まれます。

ドロップダウンリストから、保護する NFS エクスポートポイントを選択します。

**[Next]** をクリックして次のダイアログボックスに進みます。

5. **[IP Tag]** ダイアログには、LifeKeeper 保護下にあり、NFS リソースが作成されているサーバー上で In-Service 状態の仮想 IP アドレスに対応するタグのドロップダウンリストが表示されます。クライアントが保護された NFS ファイルシステムにアクセスするために使用する仮想 IP アドレスの タグ を選択します。




**注記：** この時点で、LifeKeeper は利用可能な保護された IP リソースがあることを確認します。また、NFS リソース階層を作成するために有効なデータが提供されているかどうか検証します。LifeKeeper がこれらの検証のいずれかで問題を検出すると、ERROR ボックスが画面に表示されます。ディレクトリパスは有効であるのに NFS の設定自体にエラーがある場合、一時停止してこれらのエラーを修正し、階層作成を続行することができます。必要な LifeKeeper の IP リソースを作成するために一時停止することもできます。

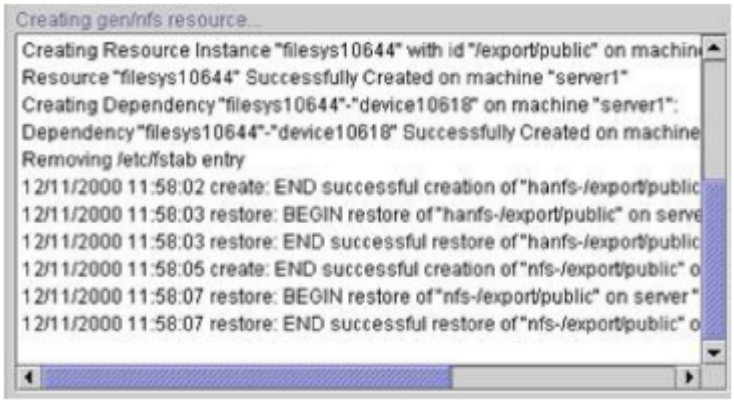
**注記：** 仮想 IP アドレスの依存関係を持つ他の LifeKeeper Recovery Kit を使用している場合は、NFS リソース用に別の仮想 IP アドレスを作成する必要があります。そうしないと、仮想 IP リソースがバックアップサーバーにフェールオーバーした場合、その IP リソースに依存するすべてのリソースが同時にフェールオーバーされてしまいます。

**[Next]** をクリックして次のダイアログボックスに進みます。

6. **NFS タグ** を選択または入力します。これは NFS 階層に与えられたタグ名です。デフォルトを選択

するか、独自のタグ名を入力することができます。

**[Create]** ボタンをクリックすると、 **リソース作成ウィザード**が NFS リソースを作成します。




**[Next]** をクリックすると、LifeKeeper は階層の拡張で説明している ***Pre-Extend Wizard*** を起動します。

**注記：** この時点で NFS リソース階層が正常に作成されている必要があります。ただし、新しい NFS インスタンスが正しく起動できなかったことを示すエラーメッセージが表示されることがあります。新しい NFS 階層は、別のシステムに拡張する前に起動する（In-Service 状態にする）必要があります。

起動に失敗すると階層が削除されることがありますが、そうでない場合は、この時点で一時停止し、表示されるエラーメッセージに基づいて問題を修正することができます。エラーを修正できない場合はキャンセル以外に選択肢はなく、リソース作成がキャンセルされます。

階層の拡張に進む前に、新しい階層を In-Service にしてください。

*** 各 NFS 共有の追加のリソース階層を作成するには、上記の手順を繰り返してしてください。

**注記：** RHEL 7.1 以降および SLES12 SP1 以降で NFS リソースを作成した後、nfs-server.service の自動起動を無効にしてください。NFS リソースの起動時に rpcbind.service を実行する必要があるため、 rpcbind.service を自動的に起動するように設定します。

# 12.3.3.5. SAP リソース階層の作成

1. 1. LifeKeeper GUI メニューから、 **[Edit]** → **[Server]** を選択します。 ドロップダウンメニューから、 **[Create Resource Hierarchy]** を選択します。

   クラスター内にインストールされ認識されているすべての Recovery Kit 一覧のドロップダウンリストのあるダイアログボックスが表示されます。 ドロップダウンリストから **SAP** を選択します。

   **[Next]** をクリックします。

   いずれかのダイアログボックスで **[Back]** ボタンが有効になっている場合は、前のダイアログボックスに戻ることができます。 これは、以前に入力した情報を修正する必要のあるエラーが発生した場合に特に役立ちます。

   階層の作成中に **[Cancel]** をクリックすると、LifeKeeper は作成プロセス全体をキャンセルします。

2. スイッチバックタイプ を選択します。 これは、バックアップサーバーへのフェールオーバー後にサービスに復帰したときに SAP インスタンスをこのサーバーに戻す方法を指示します。 **Intelligent** または **Automatic** のいずれかを選択できます。 **Intelligent** スイッチバック では、インスタンスをプライマリサーバー/元のサーバーに戻すために管理者の介入が必要です。 **Automatic** スイッチバック では、プライマリサーバーがオンラインに戻り、LifeKeeper のコミュニケーションパスが再確立されるとすぐにスイッチバックが行われます。

   スイッチバックタイプは、後で **[Resource Properties]** ダイアログボックスの **[General]** タブから変更できます。

   **[Next]** をクリックします。

3. SAP PAS、ASCS または SCS を配置するサーバーを選択します（通常、これをプライマリまたはテンプレートサーバーと呼びます）。 クラスター内のすべてのサーバーがドロップダウンリストボックスに含まれています。

4. **SAP SID** を選択します。 これは、保護対象の SAP PAS、ASCS、または SCS システムのシステム識別子です。

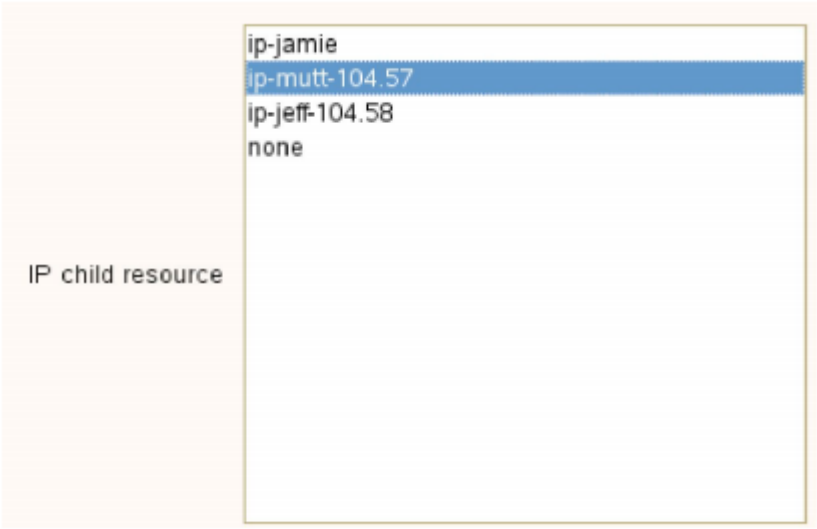
**[Next]** をクリックします。

5. 保護対象の SID の SAP インスタンス名（例：ASCS<*No.*>）（コアインスタンスが先頭）を選択します。

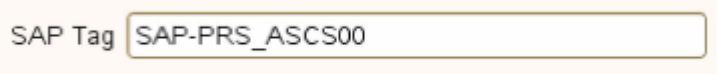
**[Next]** をクリックします。

**注記：** 保護およびリカバリーレベルのカスタマイズに関連して、追加の画面が表示されることがあります。

6. **IP の子リソース** を選択します。これは通常、SAP インストール時に指定された仮想ホスト IP アドレス（SAPINST_USE_HOSTNAME）またはフェールオーバーに必要な IP アドレスのいずれかです。



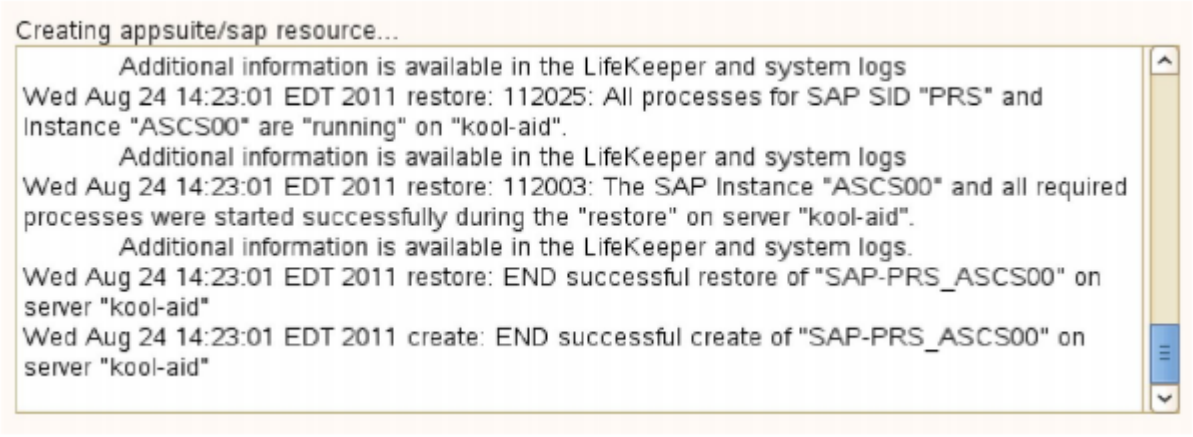

7. **SAP Tag** を選択または入力します。これは LifeKeeper が SAP 階層に付与するタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。デフォルトのタグは *SAP-<SID>_<ID>* です。




**[Create]** をクリックすると、 **SAP リソース作成ウィザード** が SAP リソースを作成します。

8. ここで情報ボックスが表示され、LifeKeeper は SAP リソース階層を作成するために有効なデータが提供されているかどうか検証します。LifeKeeper が問題を検出すると、ERROR が情報ボックスに

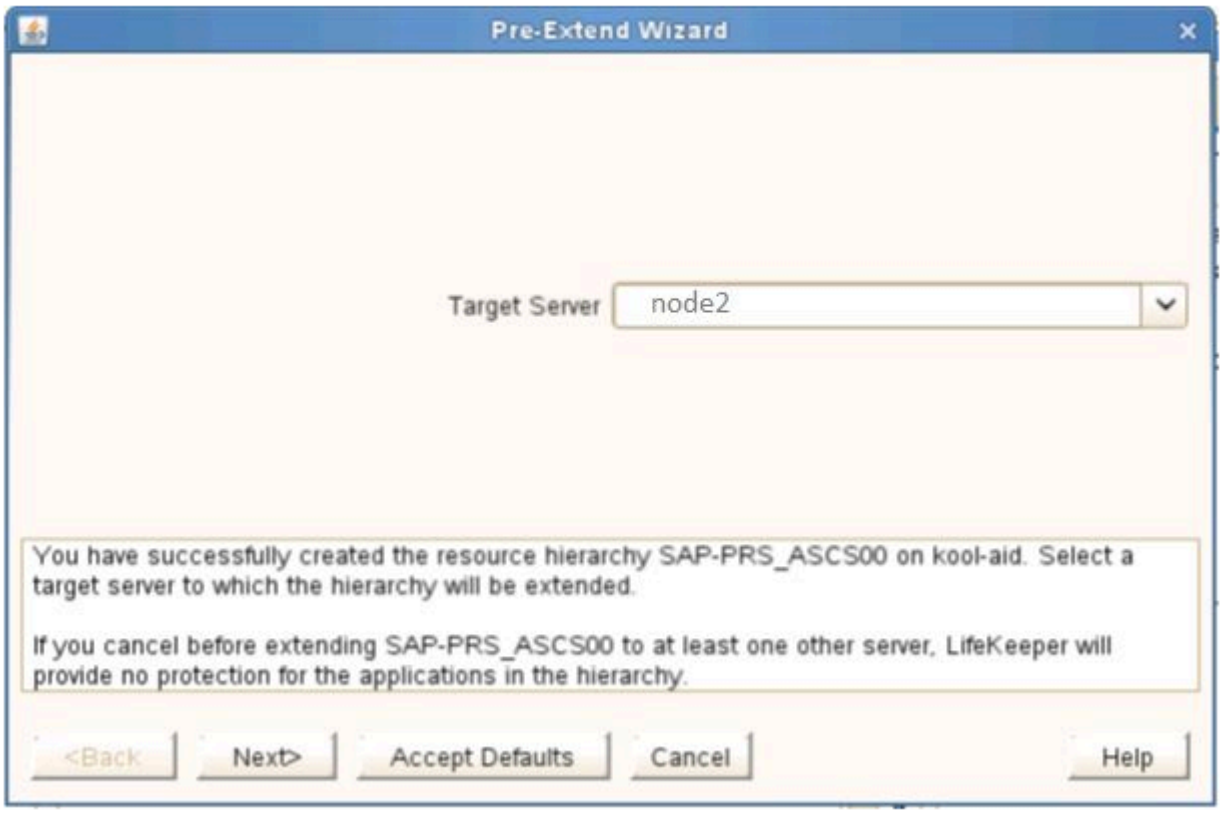
表示されます。検証が成功すると、リソースが作成されます。情報ボックスに表示される SAP 起動スクリプトからエラーまたはメッセージが出力される場合もあります。



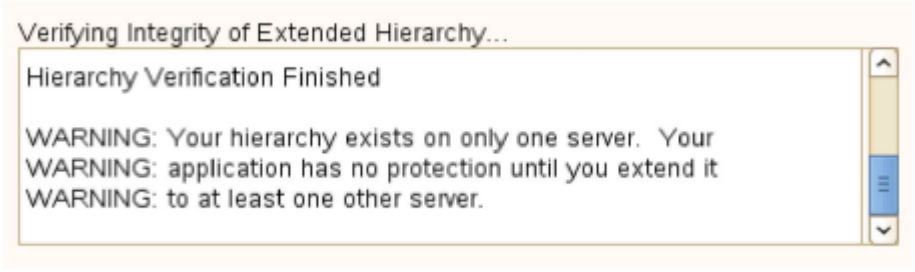

**[Next]** をクリックします。

9. SAP リソース階層が正常に作成されたことを示す別の情報ボックスが表示されます。その階層を LifeKeeper 保護下に置くためには、クラスター内の別のサーバーに **拡張** する必要があります。

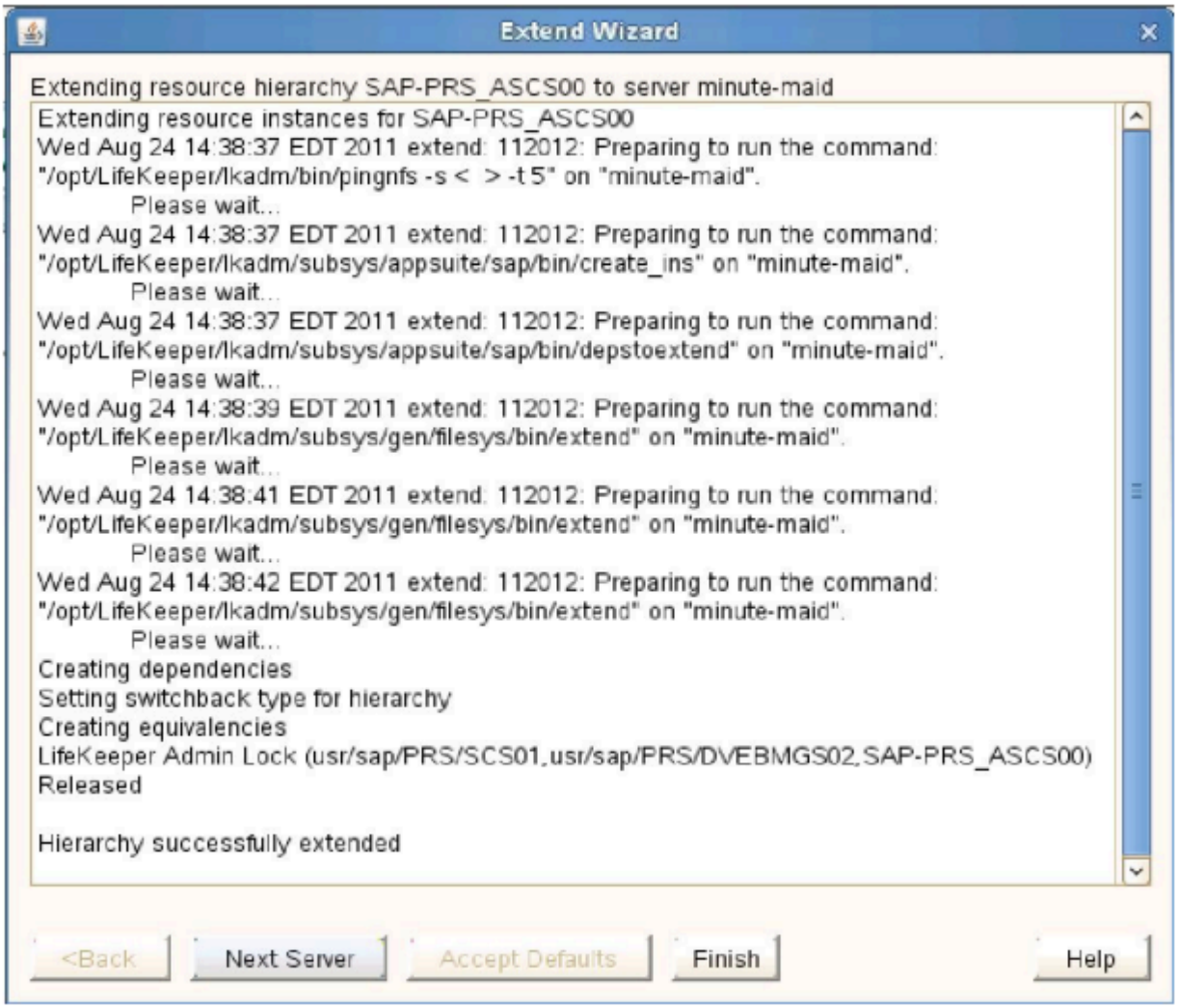
   **[Next]** をクリックすると、LifeKeeper は ***[Pre-Extend Wizard]*** を起動します。このウィザードについては、本セクションで後述します。




ここで **[Cancel]** をクリックすると別のダイアログボックスが表示され、いずれかの時点でここに戻り、SAP リソース階層を別のサーバーに拡張して LifeKeeper の保護下に置く必要があることが警告されます。




10. 階層が正常に拡張された ことを示す ***Extend Wizard*** ダイアログが表示されます。 **[Finish]** をクリックしてください。




**[Hierarchy Integrity Verification]** ダイアログが表示されます。 階層の検証が終了したら、 **[Done]** をクリックして、 **[Create Resource Hierarchy]** メニューを終了します。

## Hierarchy with the Core as the Top Level

# 12.3.3.6. ERS リソースの作成

ERS リソースは、コアインスタンス（セントラルサービスインスタンス）またはエンキューサーバープロセスの単一障害点に対する追加の保護を提供します。コアインスタンス（セントラルサービスインスタンス）で障害が発生して再起動されると、ERS リソースはロックテーブルとトランザクションの現在のステータスを取得します。その結果、エンキューサーバーに障害が発生した場合でもトランザクションや更新は失われず、SAP システムのサービスは継続されます。

この ERS リソースを作成するには、以下の手順を実行してください。

1. この同じ SAP SID について上記の手順を繰り返し、 **[ERS instance]** を選択して、ERS リソースを作成します。

2. その後、 **[Dependent Instances]** を選択するよう要求されます。上記で作成した **コアリソース** を選択し、 **[Next]** をクリックします。

3. 画面の指示に従って **リソース階層を拡張** してください。

4. 「 **Hierarchy Successfully Extended** 」と表示されたら、 **[Finish]** をクリックします。

5. **[Done]** を選択してください。

   **注記：** エンキューレプリケーションサーバ (ERS) はクラスターの各ノードで In-Service（ISP）になります。ただし、ERS のアーキテクチャと機能では、インスタンスの実際のプロセスはバックアップノードで実行する必要があります。これにより、スタンバイサーバーは、プライマリサーバーおよびプライマリエンキューサーバーインスタンスのロックテーブル情報の完全なコピーを保持できるようになります。エンキューサーバーを実行しているプライマリサーバーで障害が発生すると、ERS プロセスが現在実行されているバックアップサーバー上の SIOS Protection Suite によって再起動されます。ERS に格納されているロックテーブル（レプリケーションテーブル）は復旧中のエンキューサーバープロセスに転送され、そこから新しいロックテーブルが作成されます。このプロセスが完了すると、アクティブなレプリケーションサーバーは非アクティブになります（エンキューサーバーへの接続が切断され、レプリケーションテーブルが削除されます）。SIOS Protection Suite は、新しいバックアップノード（以前はプライマリノード）上の、今まで非アクティブだった ERS プロセスを再起動します。ERS プロセスがアクティブになると、エンキューサーバーに接続してレプリケーションテーブルを作成します。ERS プロセスがアクティブになると、エンキューサーバーに接続してレプリケーションテーブルを作成します。

   ERS プロセスおよび SAP アーキテクチャの機能の詳細については、 http://help.sap.com にアクセスし、 **Enqueue Replication Service** を検索してください。

**Hierarchy with ERS as Top Level**

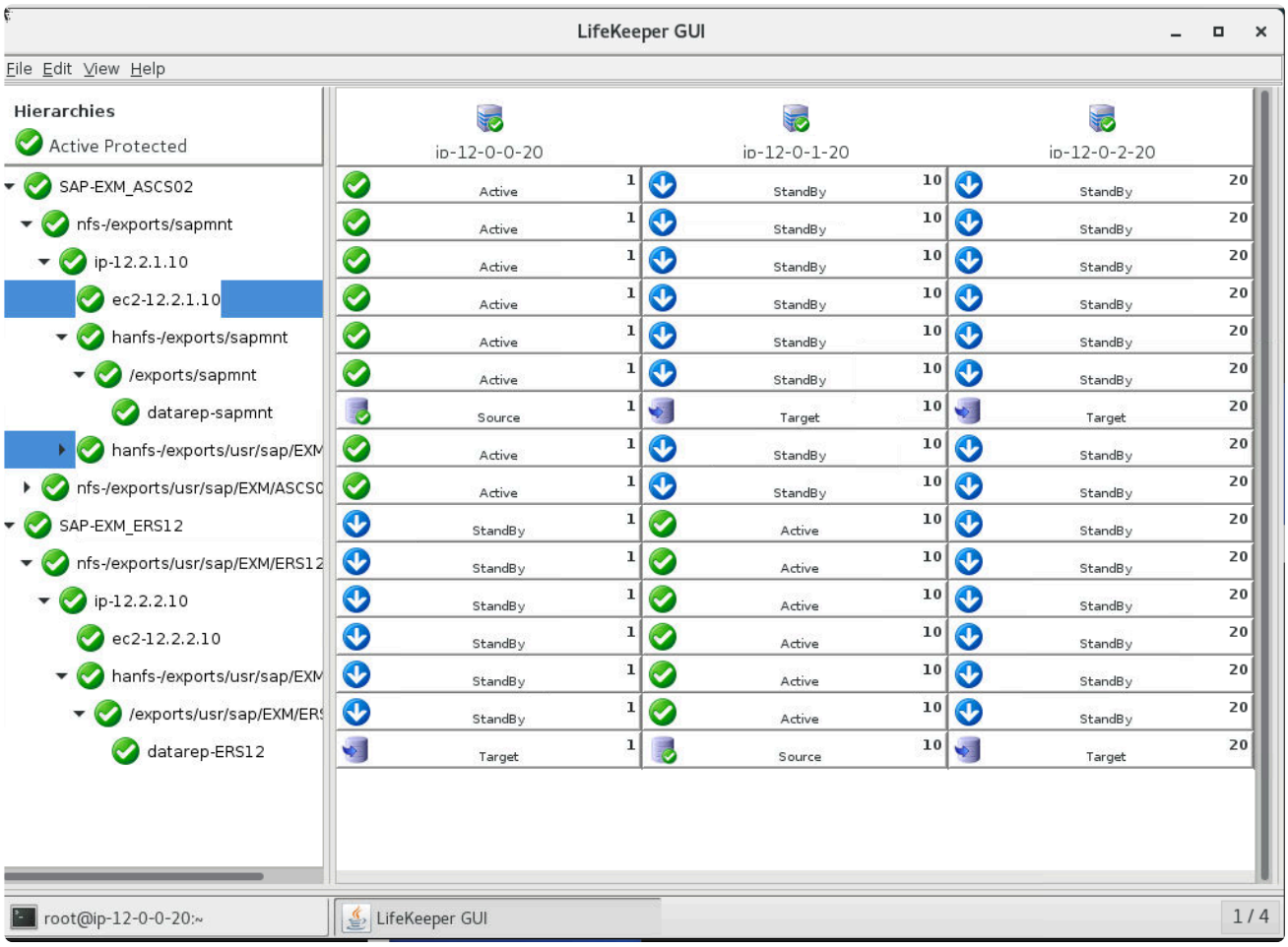
SIOS Protection Suite を使用して PAS および AAS サーバーを保護することはできますが、ほとんどのお客様は、それらを追加の HA なしで独立したスタンバイサーバーとして使用しているだけです。このガイドでは保護手順については説明しないので、詳細と手順については SAP Recovery Kit を参照してください。

# 12.3.3.7. ENSAv2/ERSv2使用時にASCS/ERSに回避動作を強制する

ERSv2は、ASCSリソースがアクティブでないクラスター内のノードでアクティブ（In-service）になります。ERS quickCheckは、ERSとASCSが同じノードでアクティブであり、別のノードが利用可能な場合、ERS階層を自動的に転送します。スイッチオーバーまたはフェイルオーバー後に同じノードでASCSとERSの両方がアクティブ（In-service）になる状況を回避するために、gen/appターミナルリーフリソースを作成して、対応するリソース階層がアクティブ（In-service）ではないノードにIn-service状態を自動的にルーティングできます。このターミナルリーフノードの作成を容易にするために、新しいユーティリティ /opt/LifeKeeper/bin/create_terminal_leaf (1M) が提供されています。

回避ターミナルリーフを作成するために、ユーティリティは2つのパラメーター、ASCSおよびERS階層ルートリソースタグを使用します。2つの階層は、クラスター内のすべてのノードに完全に拡張し、クラスター内のノードでIn-serviceにする必要があります。ユーティリティがクラスター内のノードで実行されている限り、ユーティリティが実行されているノードで階層をIn-serviceにする必要はありません。tagが回避する適切なルートノードである場合、ターミナルリーフノードには「avoid_<tag>」という名前が付けられます。ターミナルリーフノードは、階層の各ブランチの子依存関係としてアタッチされます。

たとえば、SAP-EXM_ASCS02およびSAP-EXM_ERS12をルートノードとして使用する構成では、次のようになります。

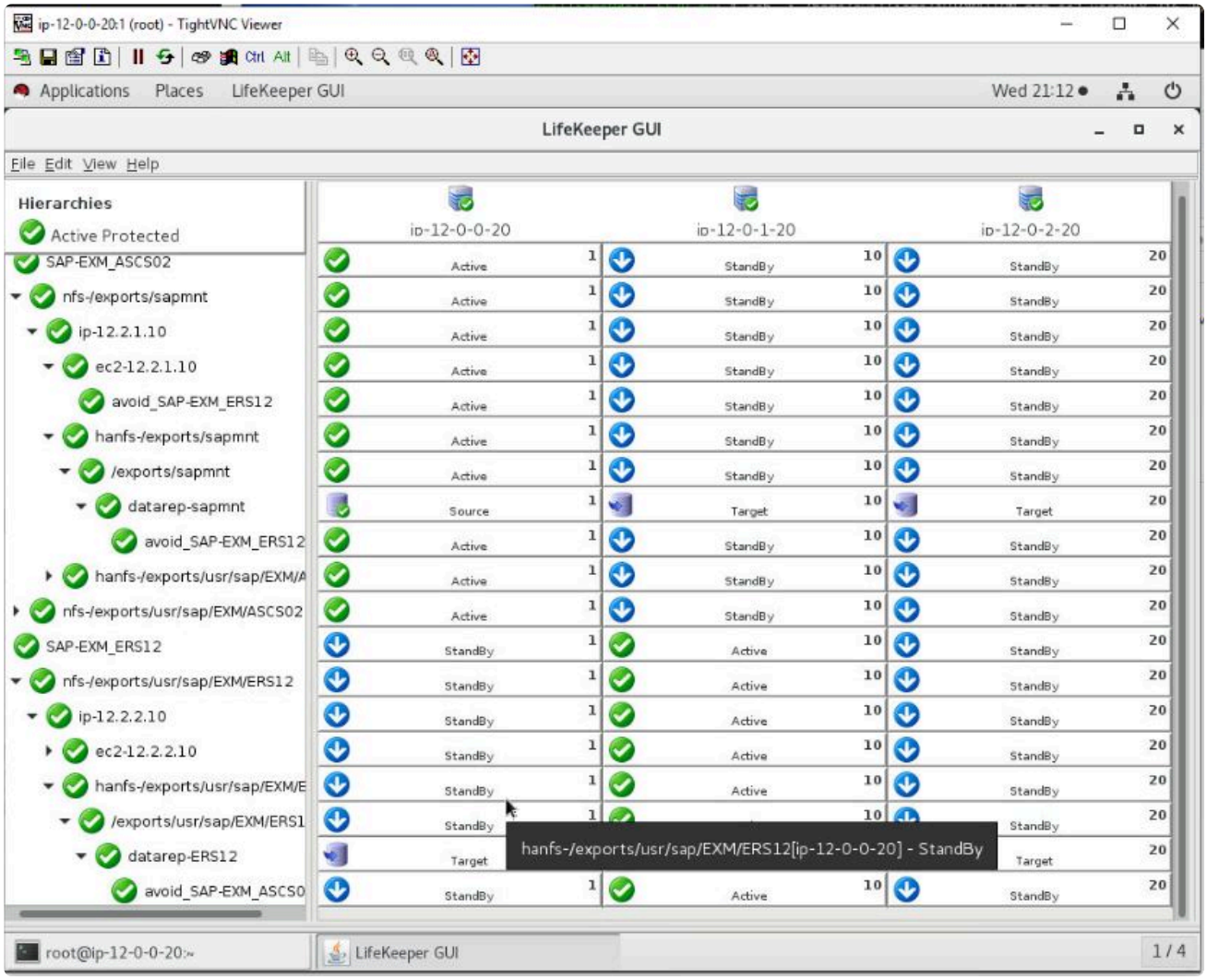



/opt/LifeKeeper/bin/create_terminal_leaf SAP-EXM_ASCS02 SAP-EXM_ERS12を実行することにより、

適切なターミナルリーフノードが作成されます。

```
[root@ip-12-0-2-20 ~]# /opt/LifeKeeper/bin/create_terminal_leaf SAP-EXM_ASCS02 SAP-EXM_ERS12
Create avoidance terminal leaf resource for root hierarchy 'SAP-EXM_ASCS02' and 'SAP-EXM_ERS12'.
creapphier: WARNING No quickCheck script specified
creapphier: WARNING No local recovery script specified
BEGIN create of "avoid_SAP-EXM_ERS12"
creating resource "avoid_SAP-EXM_ERS12"
resource "avoid_SAP-EXM_ERS12" successfully created
restoring resource "avoid_SAP-EXM_ERS12"
BEGIN restore of "avoid_SAP-EXM_ERS12"
Attempting to avoid resource avoid_SAP-EXM_ASCS02. Since the resource was not found on ip-12-0-0-20, res
tore of avoid_SAP-EXM_ERS12 is successful on ip-12-0-0-20.
END successful restore of "avoid_SAP-EXM_ERS12"
resource "avoid_SAP-EXM_ERS12" restored
END successful create of "avoid_SAP-EXM_ERS12"
Extending resource instances for avoid_SAP-EXM_ERS12
Creating dependencies
Setting switchback type for hierarchy
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ERS12) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ERS12"
END successful extend of "avoid_SAP-EXM_ERS12"
Extending resource instances for avoid_SAP-EXM_ERS12
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ERS12) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ERS12"
END successful extend of "avoid_SAP-EXM_ERS12"
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCS02'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCS02'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCS02'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
creapphier: WARNING No quickCheck script specified
creapphier: WARNING No local recovery script specified
BEGIN create of "avoid_SAP-EXM_ASCS02"
creating resource "avoid_SAP-EXM_ASCS02"
resource "avoid_SAP-EXM_ASCS02" successfully created
restoring resource "avoid_SAP-EXM_ASCS02"
BEGIN restore of "avoid_SAP-EXM_ASCS02"
Attempting to avoid resource avoid_SAP-EXM_ERS12. Since the resource is not ISP on ip-12-0-1-20, restore
 of avoid_SAP-EXM_ASCS02 is successful on ip-12-0-1-20.END successful restore of "avoid_SAP-EXM_ASCS02"
resource "avoid_SAP-EXM_ASCS02" restored
END successful create of "avoid_SAP-EXM_ASCS02"
Extending resource instances for avoid_SAP-EXM_ASCS02
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ASCS02) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ASCS02"
END successful extend of "avoid_SAP-EXM_ASCS02"
Extending resource instances for avoid_SAP-EXM_ASCS02
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ASCS02) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ASCS02"
END successful extend of "avoid_SAP-EXM_ASCS02"
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
[root@ip-12-0-2-20 ~]#
```

ターミナルリーフノードがアタッチされました。




avoid_SAP-EXM_ASCS02リソース（SAP-EXM_ERS12階層内）がノードでIn-serviceであり、クラスター内に利用可能な別のノードがある場合、avoid_SAP-EXM_ERS12リソースはSAP-EXM_ASCS02階層がそのノード上でIn-serviceになることを許可しません。

次の場合、 ノードは利用できません。

1. ノードが応答していない場合。
2. LifeKeeperがそのノードで実行されていない場合。
3. ノードでローカルリカバリーに失敗した場合。失敗したかどうかは、/opt/LifeKeeper/bin/flg_list の出力から '!volatile!recover_fail_<tag>'フラグを確認することにより判断できます。

フラグを作成することにより、特定のシステムで回避リーフを無効にできます。

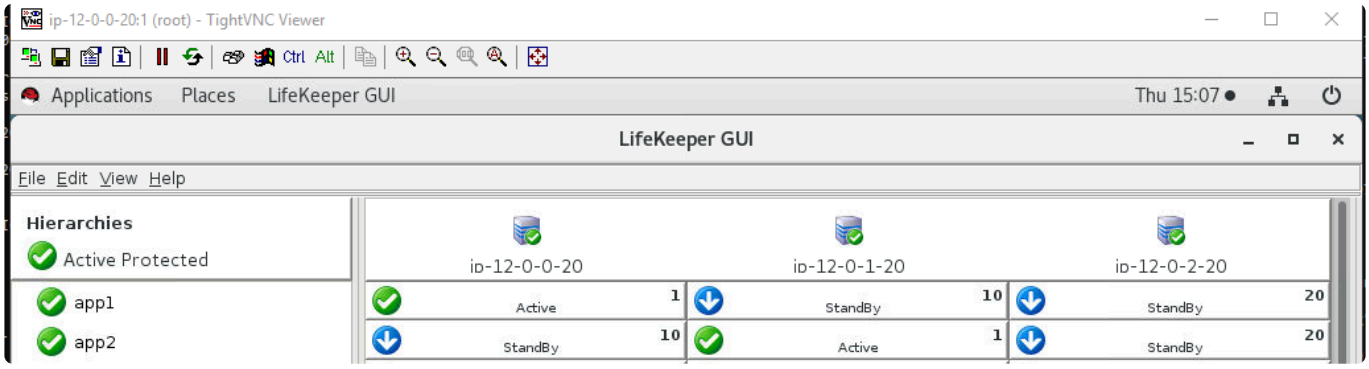
1. "ignore_avoidance_leaf" – リソースの回避リーフチェックを無効にします。つまり、回避リーフは常にIn-serviceになります。
2. "ignore_<tag>" – 特定の<tag>を無効にして常にIn-serviceになるようにしますが、他の回避リーフはIn-serviceを回避します。

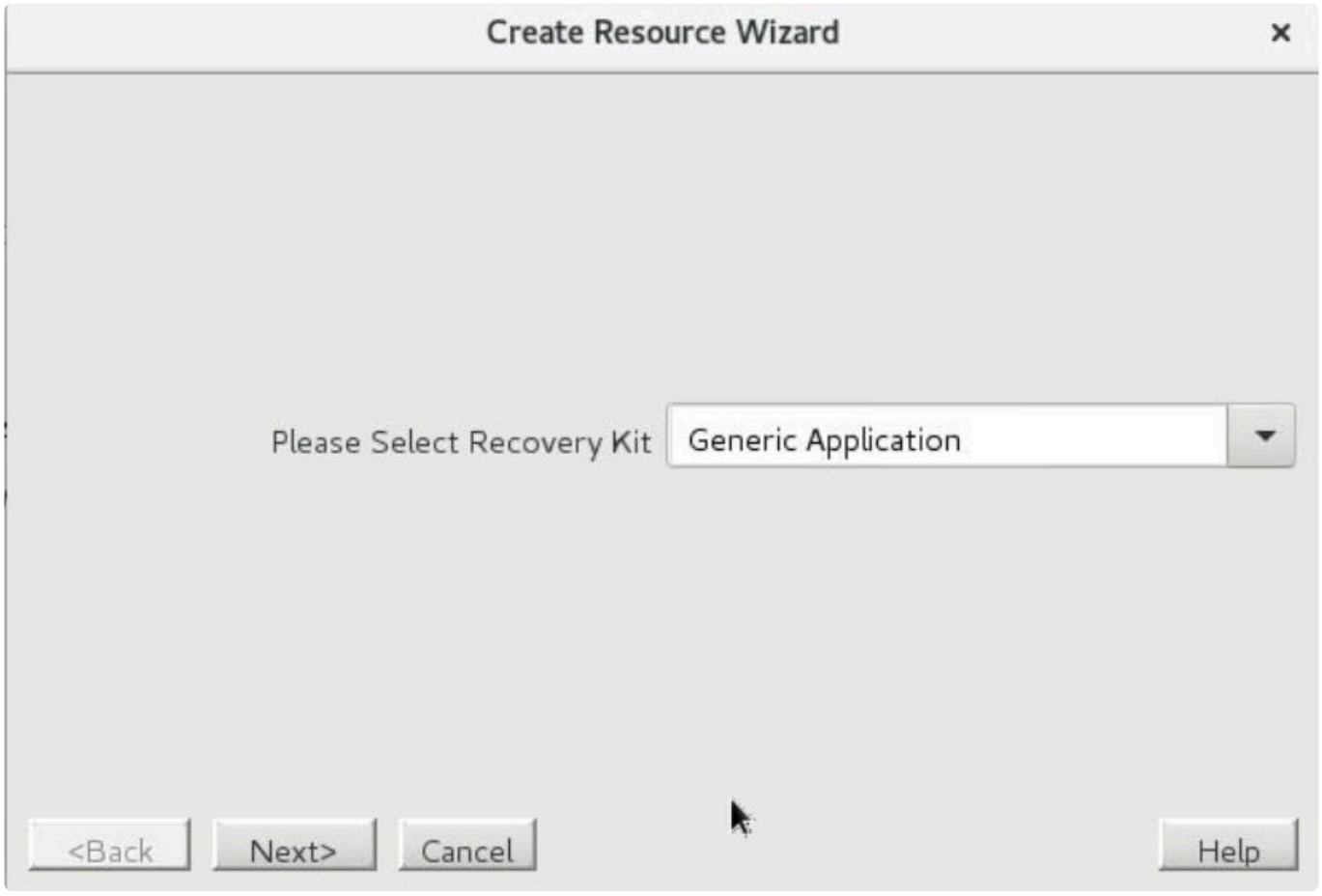
注記 ：回避リーフを無効にするための上記のフラグは、SAP quickCheckがASCSで実行されていることを検出した場合、SAP quickCheckがERSを移行することには影響しませ

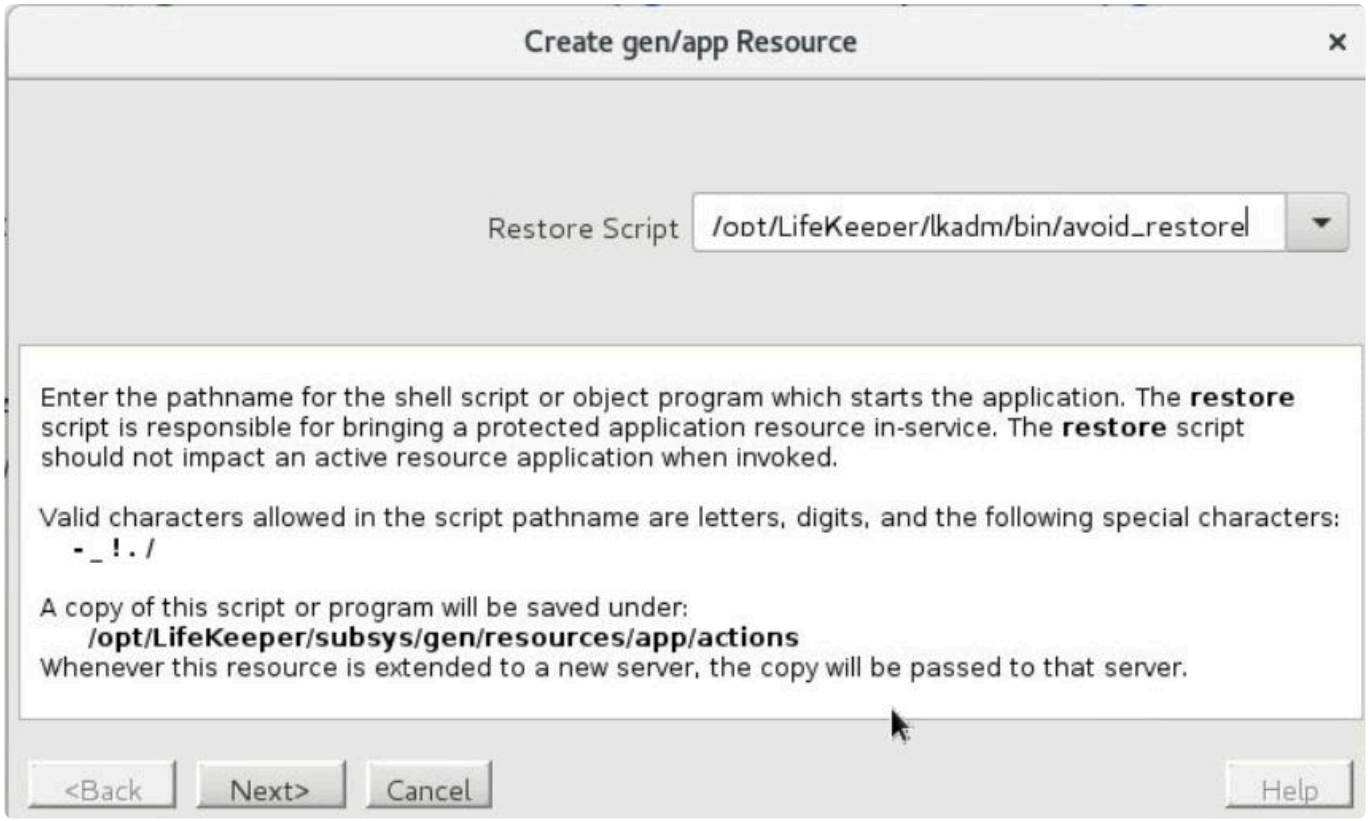
ん。 フラグ "sap_no_ers_relocation_<tag>" は、<tag>がERSリソースタグの場合、quickCheckがERSを再配置できないようにします。

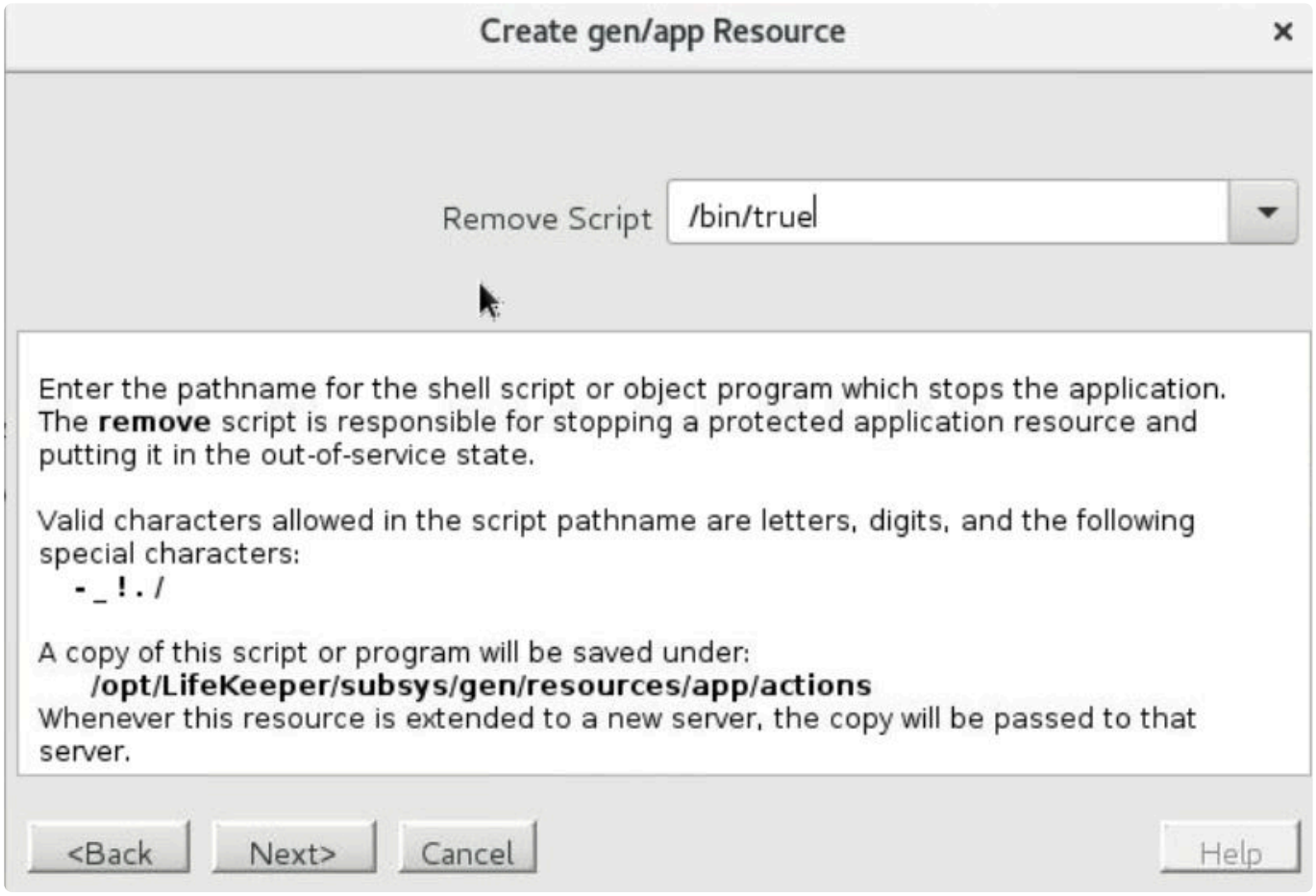
# GUIを使用して回避ターミナルリーフノードを作成する

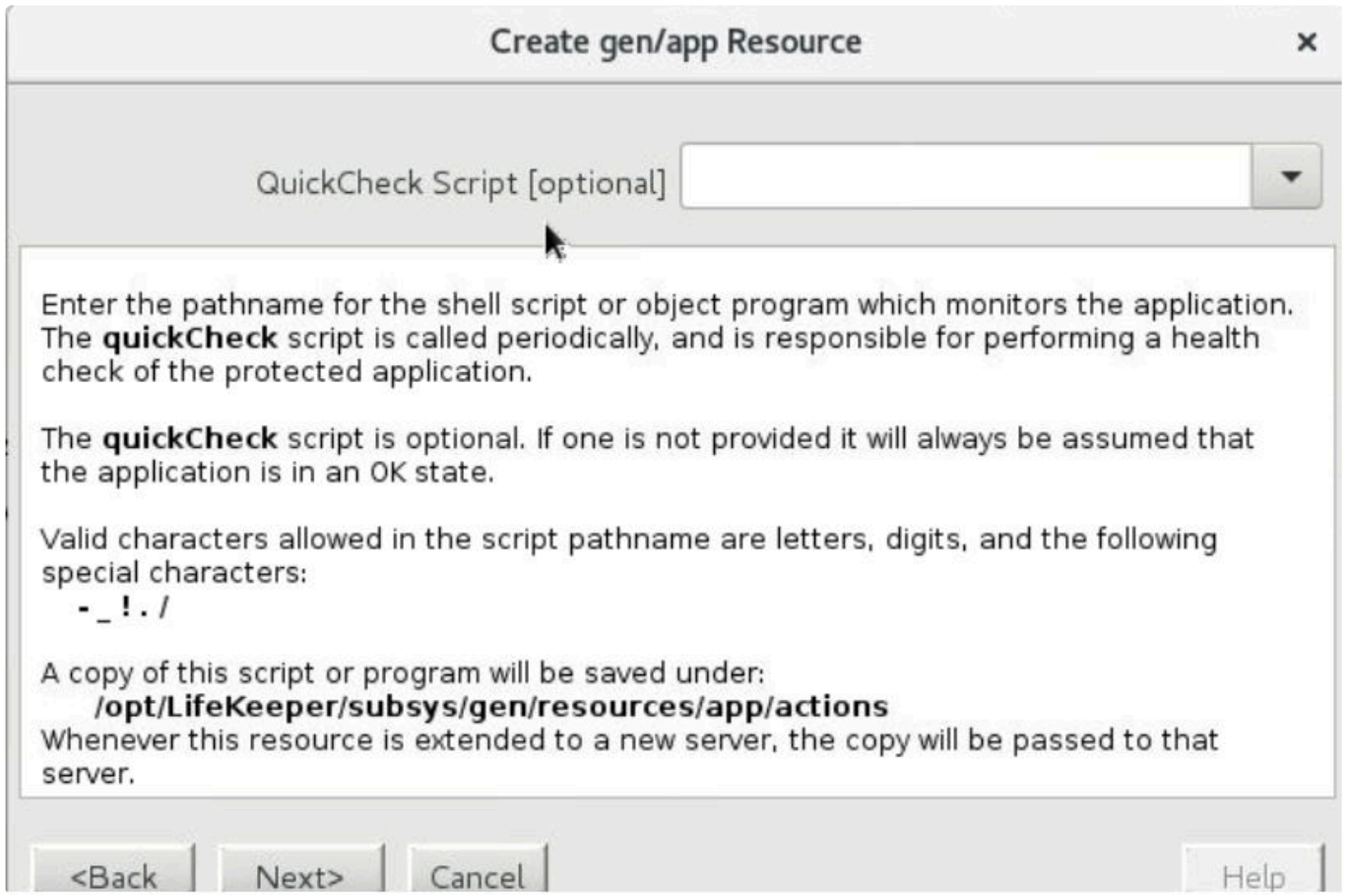
回避ターミナルリーフノードは、GUIを使用して作成できるgen/appリソースです。回避ターミナルリーフの復元スクリプトは「/opt/LifeKeeper/lkadm/bin/avoid_restore」で、削除スクリプトは「/bin/true」です。quickCheckスクリプトはありません。情報フィールドは、回避するタグの名前にする必要があります。たとえば、app1とapp2の2つのリソースがあり、可能であれば異なるノードに配置したい場合、「avoid_app1」と「avoid_app2」の2つのgen/appリソースを作成できます。avoid_app1の「情報」フィールドには「avoid_app2」が、avoid_app2の「情報」フィールドには「avoid_app1」が入ります。「avoid_app2」は「app1」の従属子リソースに、「avoid_app1」は「app2」の子リソースになります。
注記 ：タグ名は「avoid_<tag>」である必要はありませんが、これによりリソースが何をしているのかが明確になります。

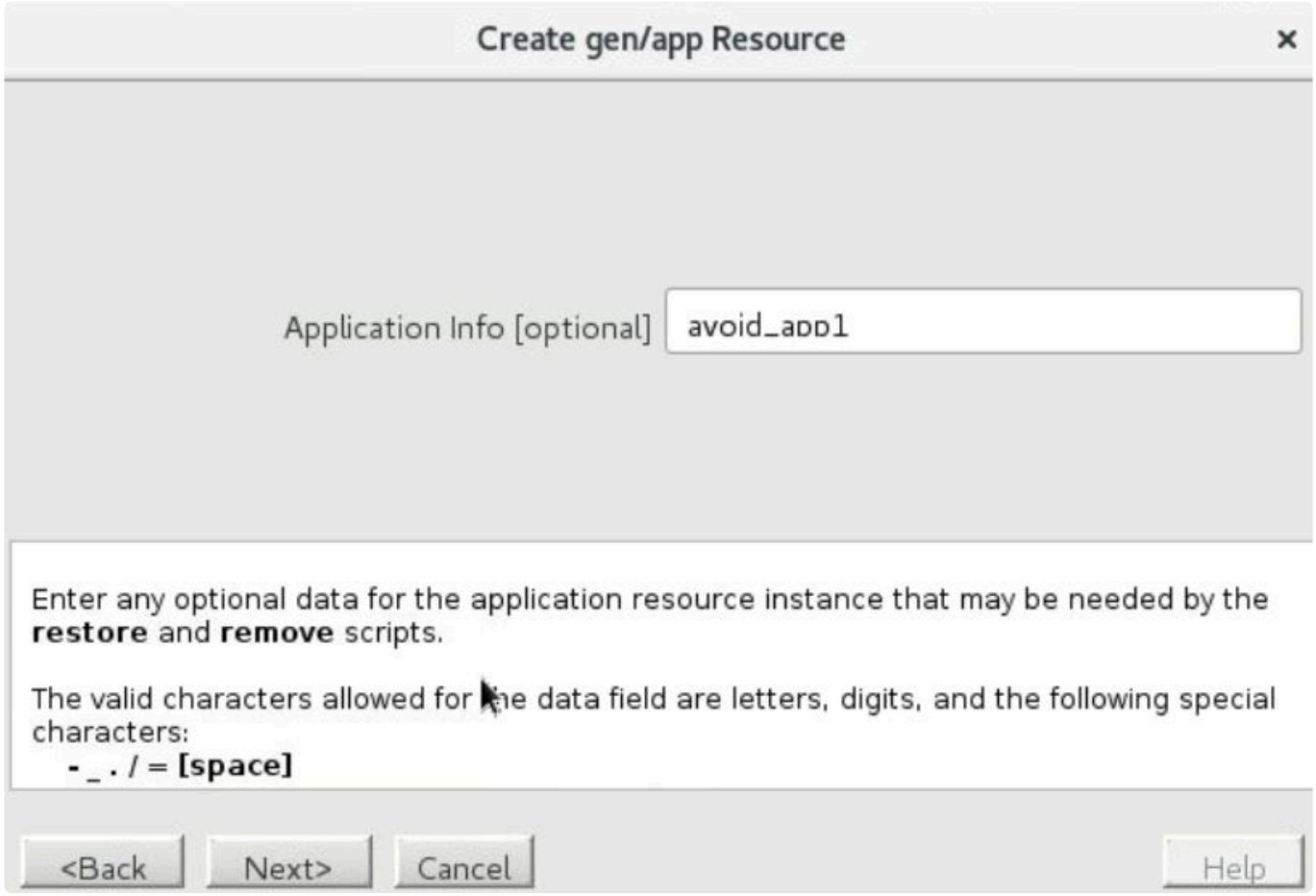



Create Resource Wizard
Please Select Recovery Kit
Generic Application
<Back
Next>
Cancel
Help


Create gen/app Resource
Restore Script
/opt/LifeKeeper/lkadm/bin/avoid_restore
Enter the pathname for the shell script or object program which starts the application. The restore script is responsible for bringing a protected application resource in-service. The restore script should not impact an active resource application when invoked.
Valid characters allowed in the script pathname are letters, digits, and the following special characters:
- _ ! . /
A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
Whenever this resource is extended to a new server, the copy will be passed to that server.
<Back
Next>
Cancel
Help

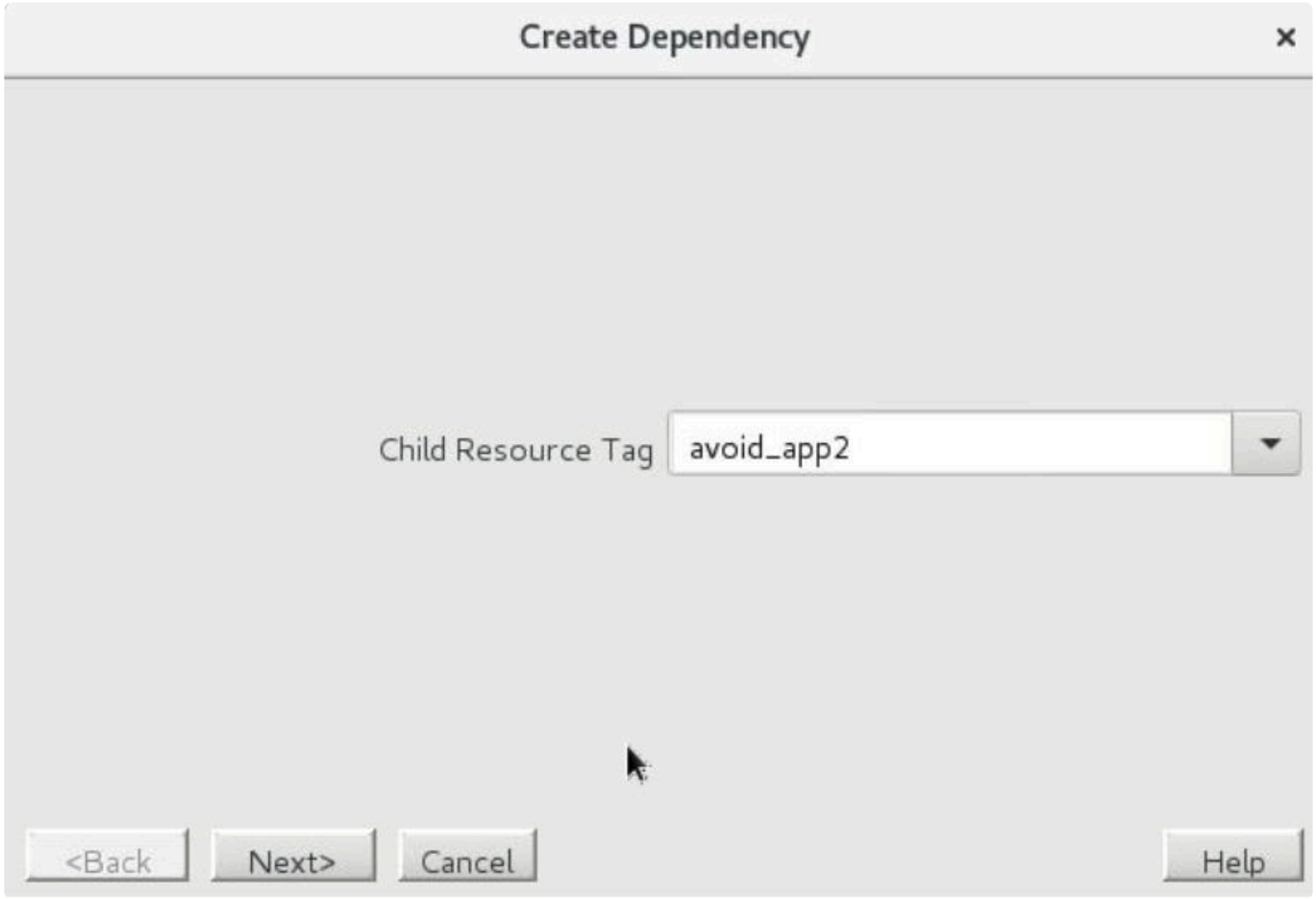

Create gen/app Resource
Remove Script
/bin/truel
Enter the pathname for the shell script or object program which stops the application. The **remove** script is responsible for stopping a protected application resource and putting it in the out-of-service state.
Valid characters allowed in the script pathname are letters, digits, and the following special characters:
- _ ! . /
A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
Whenever this resource is extended to a new server, the copy will be passed to that server.
<Back
Next>
Cancel
Help


Create gen/app Resource
QuickCheck Script [optional]
Enter the pathname for the shell script or object program which monitors the application. The **quickCheck** script is called periodically, and is responsible for performing a health check of the protected application.
The **quickCheck** script is optional. If one is not provided it will always be assumed that the application is in an OK state.
Valid characters allowed in the script pathname are letters, digits, and the following special characters:
- _ ! . /
A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
Whenever this resource is extended to a new server, the copy will be passed to that server.
<Back
Next>
Cancel
Help

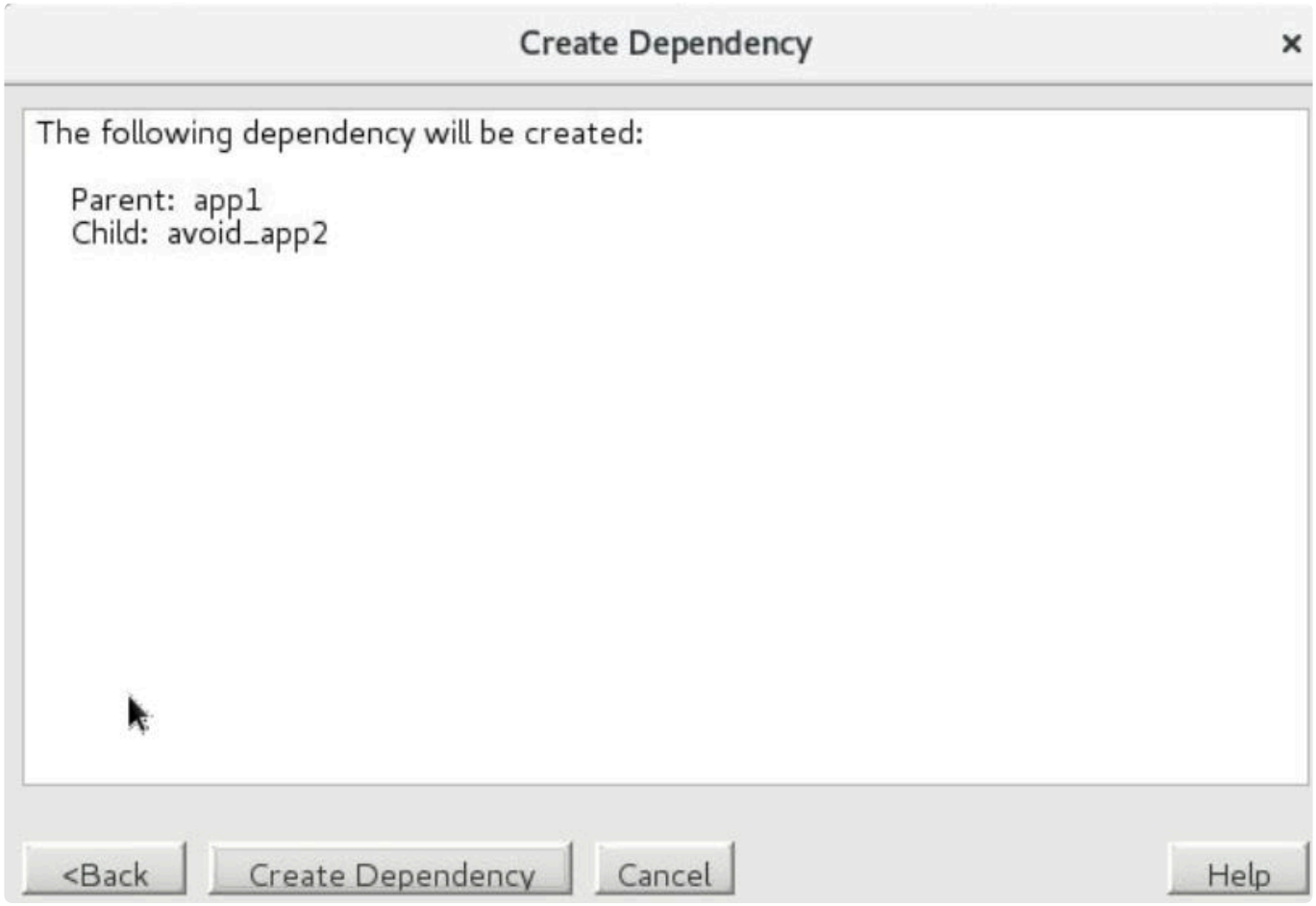


「avoid_app2」を作成したら、app1と同じ優先順位を持つすべてのノードに拡張します。

次に、「app1」リソースを選択し、「avoid_app2」との子依存関係を作成します。



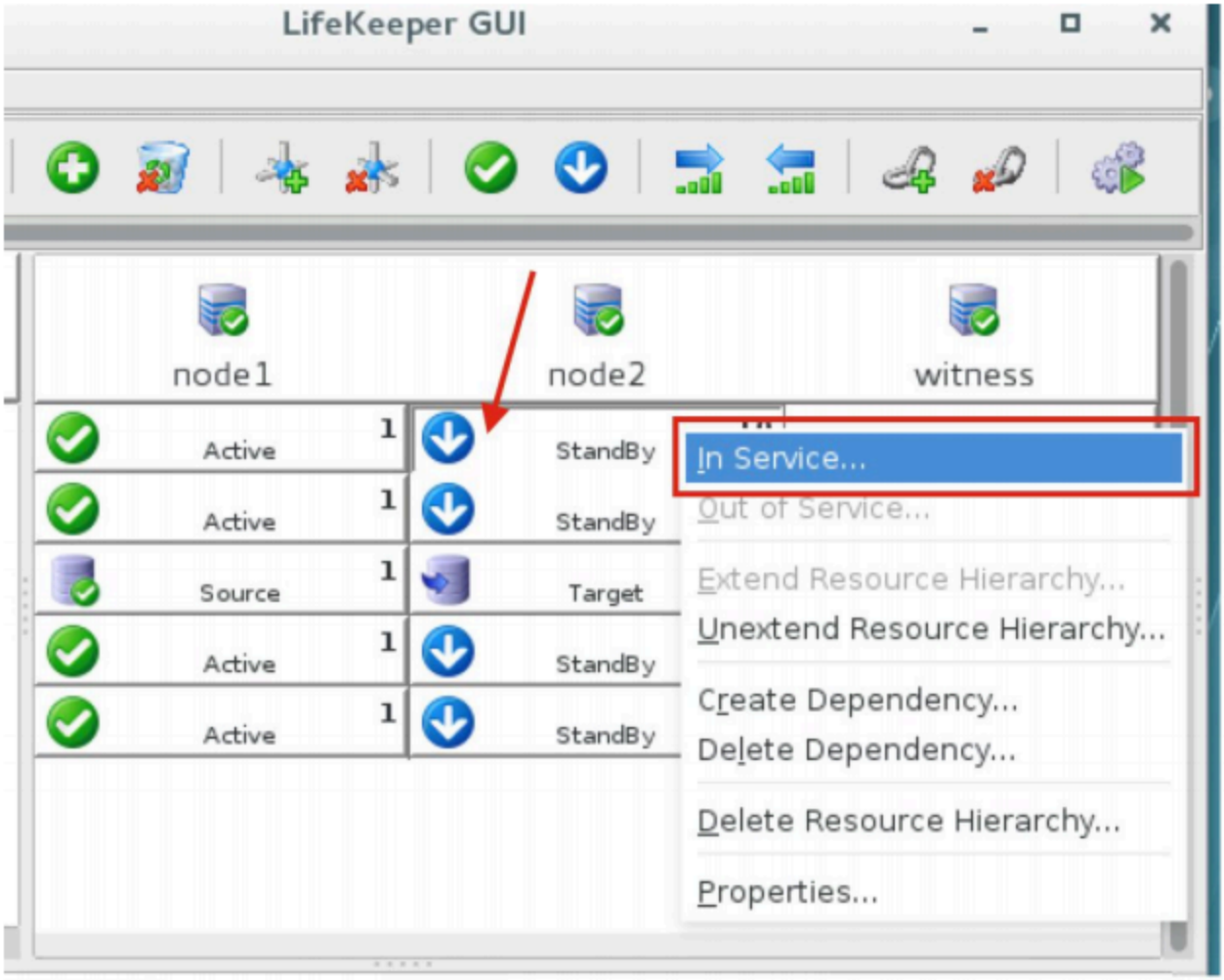

avoid_app1リソースを同様に作成すると、階層は次のようになります。


Create Dependency
The following dependency will be created:
Parent: app1
Child: avoid_app2
<Back
Create Dependency
Cancel
Help

# 12.3.4. スイッチオーバーおよびフェールオーバーのテスト

次の手順は、SAP の SIOS クラスターのスイッチオーバーとフェールオーバーをテストするための手順です。SAP 提供の Windows クライアントである「SAP Logon」または「SAP GUI for Windows」を開きます。プログラムは SAP ダウンロードサイトからダウンロードできます。仮想 IP アドレスは、**[Properties]** ページで「アプリケーションサーバー」として使用できます。これにより、仮想 IP が存在するプライマリマシンへの接続がアクティブになります。

1. LifeKeeper GUI を使用し、ノード 1 → ノード 2 にフェールオーバーします。ノード 2 の下にあるクラスターの一番上のリソースを右クリックし、[In Service ...] を選択します。これは、ノード 2 が障害発生時にノード 1 から引き継ぐことができることを示しています。




スイッチオーバーが完了したら、SAP GUI をチェックするか、必要に応じて再接続し、SAP がまだ正常に動作しているかどうかを確認します。

SAP プロセスが OS で実行されているかどうかを確認することもできます。

2. LifeKeeper GUI を使用し、ノード 2 → ノード 1 にフェールオーバーします。ノード 2 の下にあるクラスターの一番上のリソースを右クリックし、[In Service ...] を選択します。これは、ノード 1 が障

害発生時にノード 2 から引き継ぐことができることを示しています。

スイッチオーバーが完了したら、SAP GUI をチェックするか、必要に応じて再接続し、SAP がまだ正常に動作しているかどうかを確認します。

SAP プロセスが OS で実行されているかどうかを確認することもできます。

3. ノード 1（アクティブノード）のコマンドラインインターフェースで、次のコマンドを実行して OS の「ハードクラッシュ」を実行します。

   # halt -fni

   フェールオーバーが完了したら、ノード 2 の LifeKeeper GUI を使用して、サービスが正常にフェールオーバーされたことを目視で確認します。

   SAP GUI をチェックするか、必要に応じて再接続し、SAP がまだ正常に動作しているかどうかを確認します。

   SAP プロセスが OS で実行されているかどうかを確認することもできます。

   ノード 1 を再びオンにし、ノード 2 の LifeKeeper GUI を使用して、ノード 1 のサービスがスタンバイになり、レプリケーションが開始されたことを目視で確認します。

   **注記：** スイッチオーバーまたはフェールオーバーのテストをさらに行う前に、データレプリケーションリソースがすでに同期を完了し、同期中であることを確認してください。

4. 必要に応じて手順 2 または手順 1 を繰り返し、ノード 1 にスイッチオーバーするか、ノード 2 でさらにクラッシュテストを実行します。

# 12.3.4.1. その他のリソース

## AWS サービス

- Amazon EC2
  https://aws.amazon.com/documentation/ec2/

- AWS CloudFormation
  https://aws.amazon.com/documentation/cloudformation/

- Amazon VPC
  https://aws.amazon.com/documentation/vpc/

## SIOS Protection Suite for Linux

- ステップバイステップ：共有ストレージを使用せずに Amazon EC2 で Linux フェールオーバークラスターを構成する方法
  http://www.linuxclustering.net/2016/03/21/step-by-step-how-to-configure-a-linux-failover-cluster-in-amazon-ec2-without-shared-storage-amazon-aws-sanless-cluster/

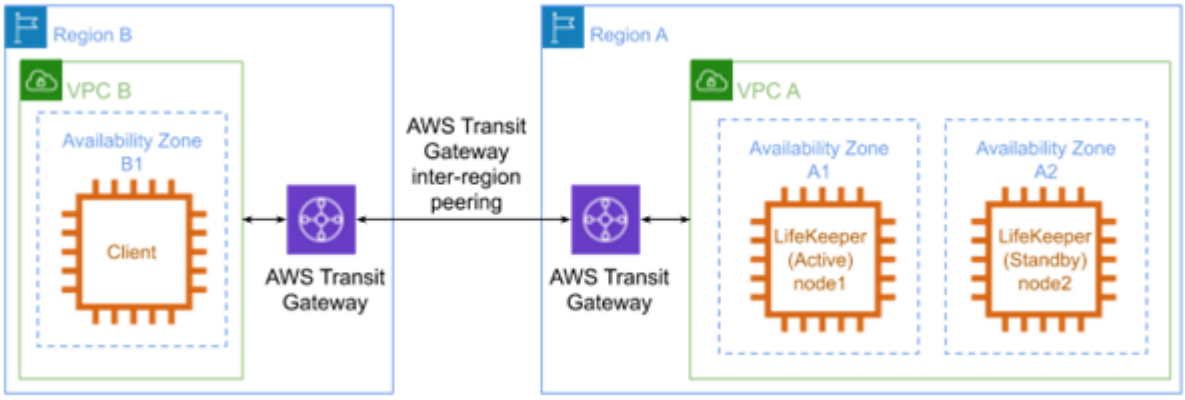
## クイックスタート

- AWS クイックスタート ホームページ
  https://aws.amazon.com/quickstart/

# 12.4. Transit Gatewayを使用したLifeKeeperクラスターとクライアント間接続クイックスタートガイド

## 目的

AWS Transit Gateway、および AWS Transit Gateway inter-region peering のリリースにより、VPC（下図の VPC A）に配置された HA クラスターノードに、別リージョンの VPC（下図の VPC B）のクライアントから接続する構成で、Recovery Kit for EC2 のルートテーブルシナリオが利用可能になりました。




本資料は、LifeKeeper for Linux で、HAクラスターノードに、別リージョンの VPC のクライアントから接続する構成を構築するための要件や基本操作を解説するものです。

なお、本資料は LifeKeeper や Amazon Web Service (以下 AWS )の基本的な設定や操作、技術的な詳細情報を解説するものではありません。本構成の前提となる LifeKeeper や AWS に関する用語･操作･技術情報等につきましては、関連のマニュアルやユーザーサイト等であらかじめご確認ください。

注記: AWS Transit Gateway inter-region peeringは、本ドキュメント執筆時点（2020年2月）で米国東部(バージニア北部)、米国東部(オハイオ)、米国西部(オレゴン)、欧州(アイルランド)、欧州(フランクフルト)の各リージョンで利用可能です。その他のリージョンにサーバーまたはクライアントを配置する場合には本ドキュメントの構成は利用できません。AWS Transit Gateway inter-region peering が利用できないリージョンにサーバーまたはクライアントを配置する場合は、Route53 Recovery Kit を使用して、AWS の DNS サービスの「Route53」に登録されている DNS の A レコード（ホスト名に対応するIPアドレス）を更新する構成を検討してください。

注記: 本ドキュメントは、単一の VPC 内にクラスターノードを配置する構成を対象としています。クラスターノードを複数のリージョンや複数の VPC にまたがって配置する構成では、ルートテーブルシナリオは利用できません。

注記:「Amazon Web Services」、「Powered by Amazon Web Services」のロゴ、「AWS」、

「Amazon EC2」、「EC2」、「Amazon Elastic Compute Cloud」、「Amazon Virtual Private Cloud」、「AWS Direct Connect」、「AWS Identity and Access Management」、「AWS Transit Gateway」、「AWS Transit Gateway inter-region peering」および「Amazon VPC」は、米国その他の国における Amazon.com, Inc. またはその関連会社の商標です。

# 12.4.1. 利用のための要件

本構成を利用するためには、環境を準備する段階で満たすべきいくつかの要件があります。以下にAWS環境とその上に作成するインスタンスに関する要件をまとめます。

## AWS環境の要件

サービスを提供するための基盤となる環境をAWS上に作成します。本構成を利用するための要件は以下の通りです。

### Amazon Virtual Private Cloud (VPC)

- クラスターノードを配置する VPC を AWS 内に設定する必要があります。
- クライアントを配置する VPC は、クラスターノードを配置する VPC とは別リージョンに設定する必要があります。
- クラスターノードを配置する VPC に、プライマリ用インスタンスを配置するサブネットとスタンバイ用インスタンスを配置するサブネットを、それぞれ異なる Availability Zone(AZ) に作成する必要があります。
- クラスターノードを配置したVPC内のサブネットのセキュリティグループは、クライアントを配置したVPC内のサブネットからのインバウンドアクセスを許可するように設定する必要があります。

### Amazon Elastic Compute Cloud (EC2)

- クラスターノード用のインスタンスが2つ以上必要です。
- プライマリインスタンスとスタンバイインスタンスは、それぞれ異なるAZに構成する必要があります。
- クラスターノード用インスタンスは、Elastic Network Interface (ENI) に接続されます。
- クラスターノード用インスタンスは、LifeKeeper のインストール要件を満たす必要があります。
- AWS Command Line Interface (AWS CLI) を全てのクラスターノード用インスタンスにインストールする必要があります。インストール方法は、「AWS Command LineInterface のインストール 」を参照してください。AWS CLIの実行可能ファイルへのパスがLifeKeeperのデフォルトファイルである/etc/default/LifeKeeperのPATHパラメータに設定されていない場合、追加する必要があります。
- https を使用して Amazon EC2 Web サービスエンドポイントの URL ( https://docs.aws.amazon.com/ja_jp/general/latest/gr/rande.html#ec2_region ) に、および http を使用して Amazon EC2 メタデータの URL (http://169.254.169.254/) にアクセス可能でなければなりません。

### AWS Identity and Access Management (IAM)

LifeKeeper が AWS を操作するために、以下のアクセス権を持った IAM ユーザーもしくは IAM ロールが必要です。EC2 インスタンスの root ユーザーからアクセスできるように EC2 の IAM ロール を設定する

か、AWS CLI の設定を適切に行ってください。

- ec2:DisassociateAddress
- ec2:DescribeAddresses
- ec2:AssociateAddress
- ec2:DescribeRouteTables
- ec2:ReplaceRoute

### AWS Transit Gateway

- クラスターノードを配置した VPC と、クライアントを配置した VPC とは、VPC 同士を AWS Inter-Region VPC Peering で直接接続するのではなく、それぞれのリージョンで AWS Transit Gateway を作成し、AWS Transit Gateway 同士を AWS Transit Gateway inter-region peering で接続する必要があります。

- 各AWS Transit Gatewayを作成する際に、Default route table associationとDefault route table propagationを有効にします。

- 各リージョンにTransit Gateway Attachmentを作成して、各AWS Transit Gatewayを対応するVPCに接続します。

- Transit Gateway Attachmentを作成して、AWS Transit Gateway間のAWS Transit Gateway inter-region peering を有効にする必要があります。このステップでは、AWSによってTransit Gateway Attachmentが実際に作成される前に、対象リージョンで手動で確認する必要があることに注意してください。

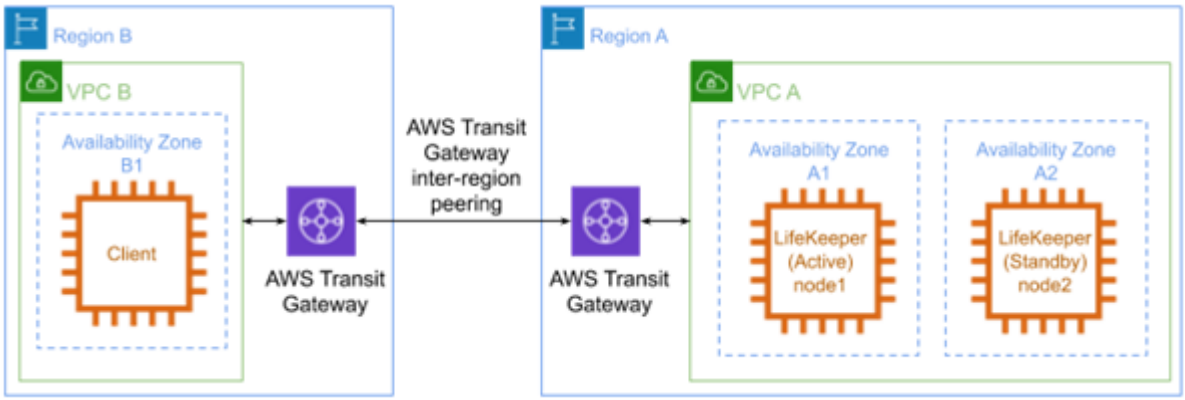
# 12.4.1.1. LifeKeeper ソフトウェアの要件

高可用性クラスターの各サーバーに、同じバージョンのLifeKeeperソフトウェアとパッチをインストールします。本構成で必要な Application Recovery Kit (ARK) は以下の通りです。具体的な LifeKeeper の要件については、 SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート を参照してください。

- LifeKeeper IP Recovery Kit
- LifeKeeper EC2 Recovery Kit

# 12.4.2. AWS VPC ピア接続構築手順

以下の構成図の様な環境を構築するための一般的な手順を解説します。




## 準備

「利用のための要件」を満たす環境を構築してください。それぞれのインスタンスに LifeKeeper をインストールして、 node1/node2 間にコミュニケーションパスを作成してください。

## IPリソース作成

ノード1で仮想IPリソースを作成し、ノード2に拡張します。IPリソースアドレスは、VPC Aによって管理されるCIDRブロック外にある必要があります。

## EC2 リソース作成

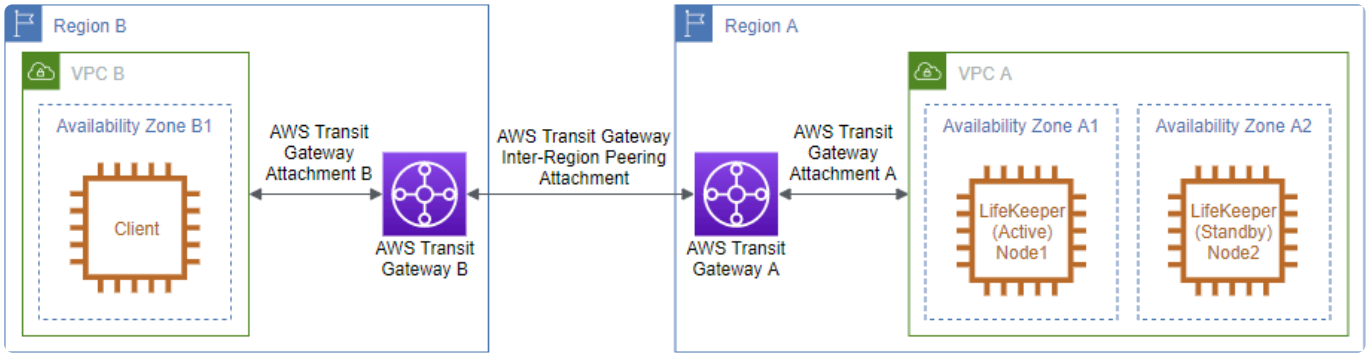
ノード1でEC2リソースを作成し、ノード2に拡張します。リソース作成時に要求される IP リソースは、上述の「IP リソース作成」で作成したリソースを指定してください。リソース作成時に要求される EC2 リソースタイプは Route Table (Backend Cluster) を指定してください。

## 保護するサービスのリソース作成

保護するサービスのためのリソースを作成してください。リソース作成に IP リソースが要求される場合は、上述の「IPリソース作成」で作成したリソースを指定してください。必要に応じて、サービス/アプリケーションが親リソース、EC2リソースが子リソースになるようにリソースの依存関係を設定して下さい。

# 12.4.3. ルートテーブルの設定

AWS環境は、以下の図に示すように構成する必要があります。




VPC Bまたはクライアントインスタンスをサブネットのルートテーブルにルートを追加します。

| 送信先 | ターゲット |
|---|---|
| VPC A CIDRブロック | AWS Transit Gateway B |
| 仮想IPアドレス | AWS Transit Gateway B |

AWS Transit Gateway Bのルートテーブルに次のルートを追加します。

| CIDR | Attachment |
|---|---|
| VPC A CIDRブロック | AWS Transit Gateway Inter-Region Peering Attachment |
| 仮想 IP アドレス | AWS Transit Gateway Inter-Region Peering Attachment |

AWS Transit Gateway Aのルートテーブルに次のルートを追加します。

| CIDR | Attachment |
|---|---|
| VPC B CIDRブロック | AWS Transit Gateway Inter-Region Peering Attachment |
| 仮想 IP アドレス | AWS Transit Gateway Attachment A |

VPC AまたはLifeKeeperインスタンスをサブネットのルートテーブルに次のルートを追加します。

| 送信先 | ターゲット |
|---|---|
| VPC B CIDRブロック | AWS Transit Gateway A |

設定が完了したら、クライアントが高可用性クラスター内の各サーバーのプライベートアドレスと仮想IPアドレスにアクセスできることを確認してください。

# 12.4.4. 本構成における設定および運用上の留意点

LifeKeeper I/O フェンシングの利用を検討してください

# 12.4.4.1. LifeKeeper I/O フェンシングの利用を検討してください

AWS 環境では共有ディスク環境が構成できないため、スプリットブレイン問題を防ぐために SCSI リザーブ方式を採ることができません。
これを踏まえて、本構成ではより安全に運用できるように LifeKeeper の I/O フェンシング機能の Quorum/Witness Server もしくは STONITH の利用をご検討ください。
特に、Quorum モードの TCP_REMOTE 設定を利用すれば、別途 Quorum サーバーを立てずに I/O フェンシング機能を実装できるため、クラウド環境においては利用しやすいと考えられます。

利用方法については以下のURLをご確認ください。

- Quorum/Witness
- STONITH

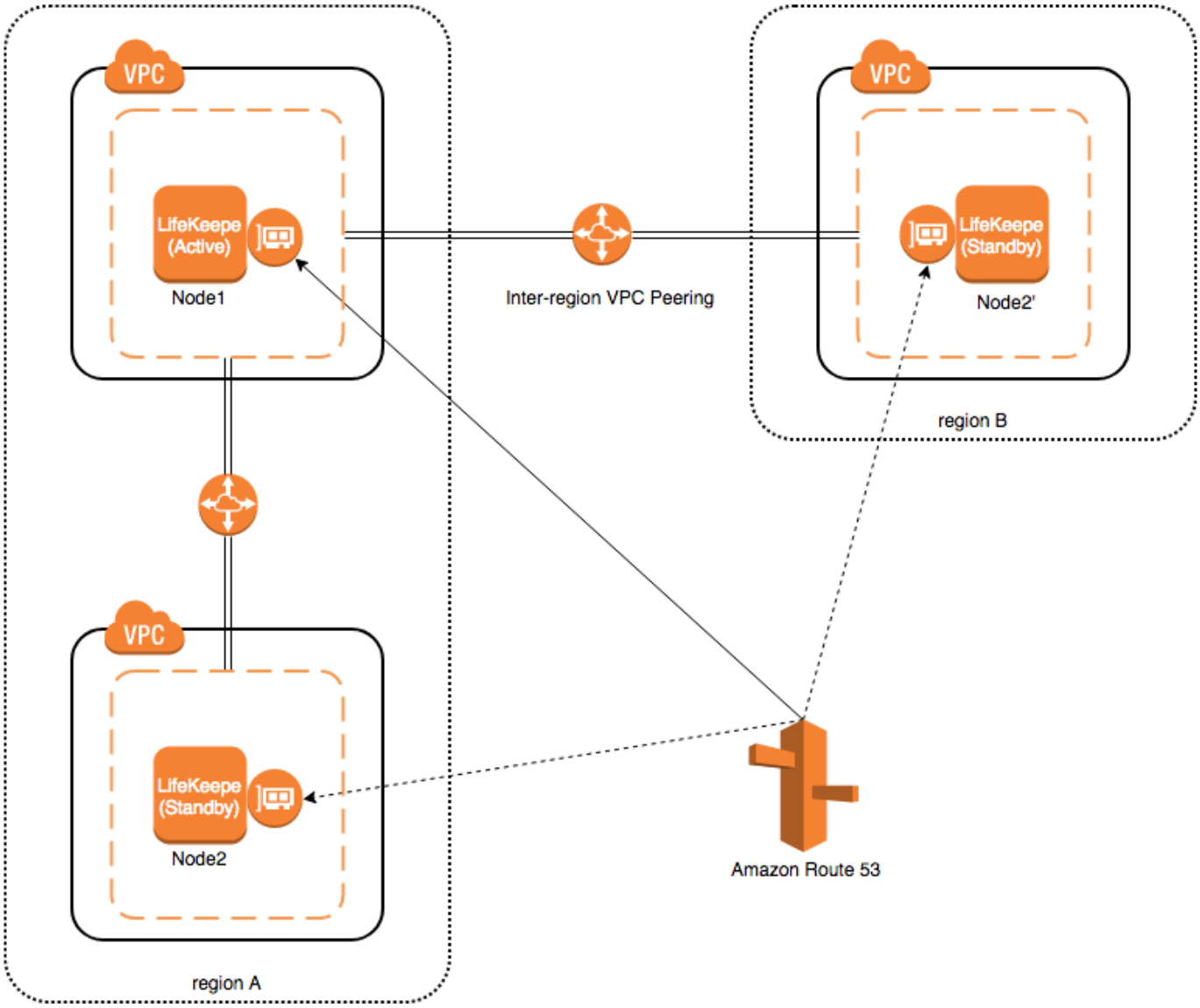
# 12.4.5. 既知の問題とトラブルシューティング

現時点での情報はありません。

# 12.5. AWS VPC ピアリングによる LifeKeeper クラスター間接続クイックスタートガイド

## 目的

LifeKeeper for Linux v9.2.2 より、リージョン間 VPC ピア接続を利用した別 VPC からの接続構成がサポートされました。




本資料は、LifeKeeper for Linux v9.5.0 で VPC 間の接続構成を構築するための要件や基本操作を解説するものです。

既存の Recovery Kit for EC2 を利用しても、AWS 環境における HA クラスターを構築することができます。しかし、以下の問題があり、AWS Direct Connect を利用したオンプレミス環境からの接続はできません。

Recovery Kit for EC2 は「ルートテーブルシナリオ」と「Elastic IP シナリオ」の 2 つの機能をサポートします。

「 **ルートテーブルシナリオ** 」では、アクティブな IP リソースにルーティングされるように VPC のルートテーブルを制御しています。この時、IP リソースのアドレスは、VPC で管理している CIDR ブロック外である必要があります。しかし、異なる VPC から VPC ピア接続 経由で接続するためには、VPC CIDR ブロック内のアドレスである必要があります。従って、ルートテーブルシナリオではオンプレミス環境からの接続はできません。

「 **Elastic IP シナリオ** 」では、Elastic IP アドレスがパブリックなアドレスであるため、インターネットからアクセスが可能であるケースで利用します。オンプレミス環境からのアクセスはインターネットを経由して行うことが可能です。この場合は VPC ピア接続を利用しなくとも、VPC 内の HA クラスタノードにアクセスすることができます。

上記の理由により、Recovery Kit for EC2 では VPC ピア接続を利用した異なる VPC からのアクセスには対応できません。VPC ピア接続を経由して VPC 内の HA クラスタノードにアクセスする必要がある場合は、本資料にてご紹介する構成を適用してください。

✻ リージョン間ピアリングに AWS Transit Gateway を使用できるようになりました。 詳細については、 Transit Gatewayを使用したLifeKeeperクラスターとクライアント間接続クイックスタートガイド を参照してください。

なお、本資料は LifeKeeper や Amazon Web Service（以下 AWS）の基本的な設定や操作、技術的な詳細情報を解説するものではありません。本構成の前提となる LifeKeeper や AWS に関する用語･操作･技術情報等につきましては、関連のマニュアルやユーザーサイト等であらかじめご確認ください。

✻ 注記：「Amazon Web Services」、「Powered by Amazon Web Services」のロゴ、「AWS」、「Amazon EC2」、「EC2」、「Amazon Elastic Compute Cloud」、「Amazon Virtual Private Cloud」、「Amazon Route 53」および「Amazon VPC」は、米国その他の国における Amazon.com, Inc. またはその関連会社の商標です。

# 12.5.1. 利用のための必要要件

本構成を利用するためには、環境を準備する段階で満たすべきいくつかの要件があります。以下に AWS 環境とその上に作成するインスタンスに関する要件をまとめます。

## AWS 環境の要件

サービスを提供するための基盤となる環境を AWS 上に作成します。本構成を利用するための要件は以下の通りです。

### Amazon Virtual Private Cloud (VPC)

- VPC を AWS 内に設定する必要があります。
- 異なる Availability Zone (AZ)、もしくは異なる VPC に 2 つ以上のサブネットを作成する必要があります。

### Amazon Elastic Compute Cloud (EC2)

- インスタンスが 2 つ以上必要です。
- プライマリ用インスタンスとスタンバイ用インスタンスがそれぞれ異なるAZ、あるいは異なる VPC で起動するように構成する必要があります。
- インスタンスは、Elastic Network Interface (ENI)に接続されます。
- インスタンスは、LifeKeeper のインストール要件を満たす必要があります。
- AWS Command Line Interface (AWS CLI) を全ての EC2 インスタンスにインストールする必要があります。インストール方法は、「 AWS Command Line Interface のインストール 」を参照してください。
- Amazon Route 53 サービスのエンドポイント route53.amazonaws.com にプロトコル HTTPS を使用してアクセスできる必要があります。EC2および OS の設定を適切に行ってください。

### AWS Identity and Access Management (IAM)

LifeKeeper が AWS を操作するために、以下のアクセス権を持った IAM ユーザーもしくは IAM ロールが必要です。EC2 インスタンスの root ユーザーからアクセスできるように EC2 の IAM ロール を設定するか、 AWS CLI の設定 を適切に行ってください。

- route53:GetChange
- route53:ListHostedZones
- route53:ChangeResourceRecordSets

- route53:ListResourceRecordSets

## Amazon Route 53

- Amazon Route 53 にドメイン名を登録しサービスを利用できるようにする必要があります。これは Route53 リソース作成時に必要となります。

# 12.5.1.1. LifeKeeper ソフトウェアの要件

各サーバに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。本構成で必要な Application Recovery Kit (ARK)は以下の通りです。具体的な LifeKeeper の要件については、SPS for Linux テクニカルドキュメンテーション および SPS for Linux リリースノート を参照してください。

- LifeKeeper IP Recovery Kit
- LifeKeeper Route53 Recovery Kit

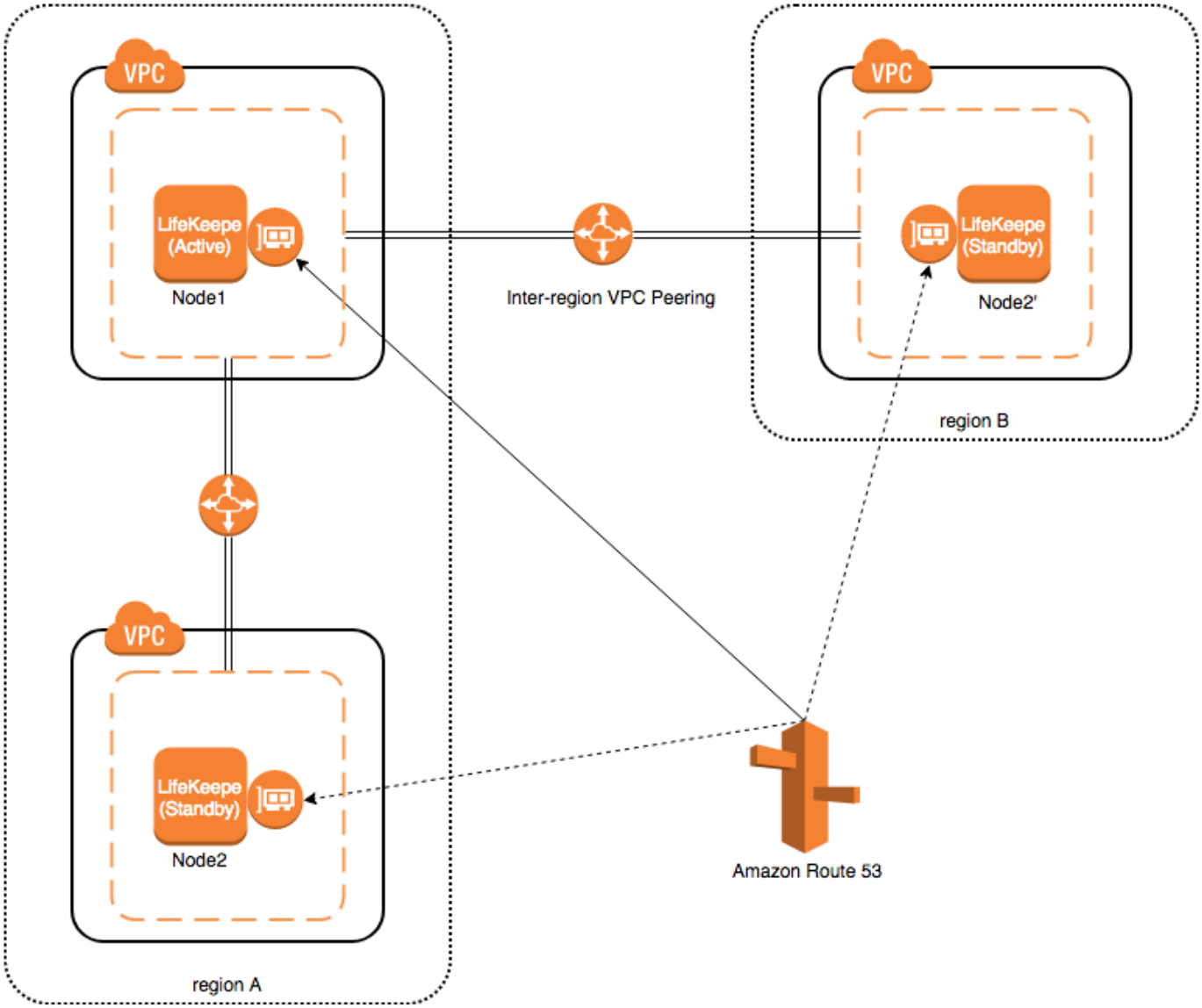
# 12.5.1.2. その他

本構成で別 VPC からサービスを利用するための要件は以下の通りです。

- サービスを利用するクライアントは、Route53 リソースで保護されるホスト名を名前解決できること。
- サービスを利用するクライアントは、Route53 リソースで保護されるホスト名でアクセスすること。

# 12.5.2. 構築手順

以下の構成図の様な環境を構築するための一般的な手順を解説します。




## 準備

前述の「 利用のための必要要件 」を満たす環境を構築します。それぞれのインスタンスに LifeKeeper をインストールして、Node1 / Node2 (あるいは Node2') 間にコミュニケーションパスを作成してください。別 VPC 環境から Node1 / Node2 (あるいは Node2') に接続された ENI のプライベートアドレスにアクセスできることを確認してください。

## IP リソース作成

IP リソースを作成します。ここでは、仮想 IP リソースではなく、実 IP リソース（注記：NIC 用に構成されたプライマリ IP アドレスのためのリソース）を作成します。リソース作成時に、ENI のプライベート IP アドレスを指定してください。また、拡張時も拡張先ノードの ENI プライベート IP アドレスを指定してください。

# Route53 リソース作成

Route53 リソースを作成します。リソース作成時に要求される IP リソースは、前述の「IP リソース作成」で作成したリソースを指定してください。

## 保護するサービスのリソース作成

保護するサービスのためのリソースを作成してください。リソース作成に IP リソースが要求される場合は、前述の「IP リソース作成」で作成したリソースを指定してください。
親リソースが保護するサービスのリソース、子リソースが Route53 リソースとなるようにリソースの依存関係を設定してください。

# 12.5.3. 関連する LifeKeeper リソースについて

## Route53 リソース

### 動作概要

スイッチオーバーが発生すると、サービスへの接続を継続して確保するために Amazon Route 53 DNS 情報の更新が必要となります。この機能は、Route53 リソースで提供されています。

## IP リソース

### 動作概要

IP リソースとは、LifeKeeper Core 製品に含まれる IP リカバリキットを使用して生成したリソースです。本構成をサポートするために、実 IP アドレスで IP リソース（実 IP リソース）が生成できるようになりました。これにより、実 IP アドレスをLifeKeeper のリソースとして扱うことができます。
なお、実 IP リソースは、本構成以外では使用しないでください。

さらに詳細な情報につきましては、IP Recovery Kit テクニカルドキュメンテーション をご覧ください。

# 12.5.4. 本構成における設定および運用上の留意点

LifeKeeper I-O フェンシングの利用を検討してください

# 12.5.4.1. LifeKeeper I-O フェンシングの利用を検討してください

AWS 環境では共有ディスク環境が構成できないため、スプリットブレイン問題を防ぐために SCSI リザーブ方式を採ることができません。また、IP リソースも各ノードで異なったアドレスを持った実 IP リソースを利用しますので、スプレットブレインが発生し得ます。

これを踏まえて、本構成ではより安全に運用できるように、LifeKeeper の I/O フェンシング機能の Quorum / Witness Server もしくは STONITH の利用をご検討ください。

特に、Quorum モードの TCP_REMOTE 設定を利用すれば、別途 Quorum サーバを立てずに I/O フェンシング機能を実装できるため、クラウド環境においては利用しやすいと考えられます。

# 12.6. MySQL Cluster with Data Replication (“Shared Nothing” Cluster)

## Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product to make MySQL highly available. If MySQL is not already installed, please allocate some time to install it on your servers. Once this task has been completed, you may install and configure SIOS Protection Suite for Linux.

There are five phases in this process:

- Prepare to Install
- Configure Storage
- Install and Configure MySQL
- Install SIOS Protection Suite for Linux
- Configure your LifeKeeper Cluster
- Test Your Environment

# 12.6.1. Terms to Know

The following terms are used throughout this document and, while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

## Network Communication Terms

**Crossover cable** – A cable used to directly connect computing devices together, instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

## Types of LifeKeeper Servers

**Server** – A computer system dedicated to running software application programs.

**Active Server** – This is the server where the resource hierarchy is currently running (IN SERVICE).

**Standby Server** – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

**Primary Server** – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

**Secondary Server** – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

**Source Server** – In a LifeKeeper cluster, using data replication, this is the Active Server. It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

**Target Server** – In a LifeKeeper cluster, using data replication, this is the Standby Server. The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

## SIOS Data Replication Terms

**Replication** – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

**Synchronous** – A replication scheme in which the data is confirmed written and valid on the target before the write operation occurs on the source disk through a series of information exchanges. Synchronous mirrors should only be implemented on high speed (100Mbps+) networks due to the network overhead involved.

**Asynchronous** – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there and can be written on them.

**Rate of Change** – A measure of the amount of data which is changing over a set period of time.

**Compression** – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

**Throttling** – An optionally implemented mechanism to limit the bandwidth used for replication.

## LifeKeeper Product Terms

**Communications Path** – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

**Heartbeat** – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

**Split Brain** – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

**Failover** – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

**Switchover** – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

**Switchback** – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

**Resource** – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

**Extend a Resource** – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously.

**Resource Hierarchy** – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

**Shared Storage** – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally

called I/O fencing.

**Data Replication (Disk Mirroring)** – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

**Source** – The partition on the source server used for replication. The “gold” copy of the data.

**Target** – The partition on the target server used for replication.

**Switchable IP Address** – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

# 12.6.2. The Evaluation Process – MySQL Cluster

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux in a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

**Important:** Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

# 12.6.3. Prepare to Install

## Hardware Requirements

### Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system ( / ) and boot (/boot) partitions are not eligible for replication.

**Note:** You may use more than one partition for replicated data, allowing for separation of multiple database files and/or log directories for performance reasons.

### Client

This system is not required but is recommended for testing the cluster environment.

- A standard linux terminal running the MySQL client can be used to test the configuration.

## Software Requirements

### Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
  - RedHat Enterprise Linux 5 (5.4+ recommended) or 6.x
  - CentOS Linux 5 (5.4+ recommended) or 6.x
  - Oracle Enterprise Linux 5 (5.4+ recommended), 6.3, 6.4 (**RedHat Compatibility Kernel**

**Only**)

- SuSE Linux Enterprise Server 10 or 11 (11 recommended)
- See Linux Release Notes for a full list of supported Operating Systems

- Current patches / security updates are recommended
- Satisfied dependencies; especially if the Linux installation package selection was base/minimal you will need to refer to the dependencies documentation at Linux Dependencies
- It is recommended that IPtables is disabled
  - # /etc/init.d/iptables off
  - # chkconfig iptables off
  - See Running LifeKeeper With a Firewall for information regarding the ports SIOS Protection Suite for Linux uses.

- Disable SELinux :
  - Edit /etc/selinux/config
  - Set SELINUX=disabled (note: permissive mode is also acceptable)

- Check the configuration of your /etc/hosts file
  - localhost.localdomain and localhost are the only entries that can be on 127.0.0.1
  - Create a separate entry for your hostname with a static address

- GUI Authentication with PAM
  - SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).
  - Users are identified and authenticated against the system's PAM configuration. Privilege levels are determined from group membership as provided through PAM.
  - In order to access the GUI, a user must be a member in one of the three LifeKeeper groups: lkadmin, lkoper or lkguest.
  - See the following URL for more information on this topic:

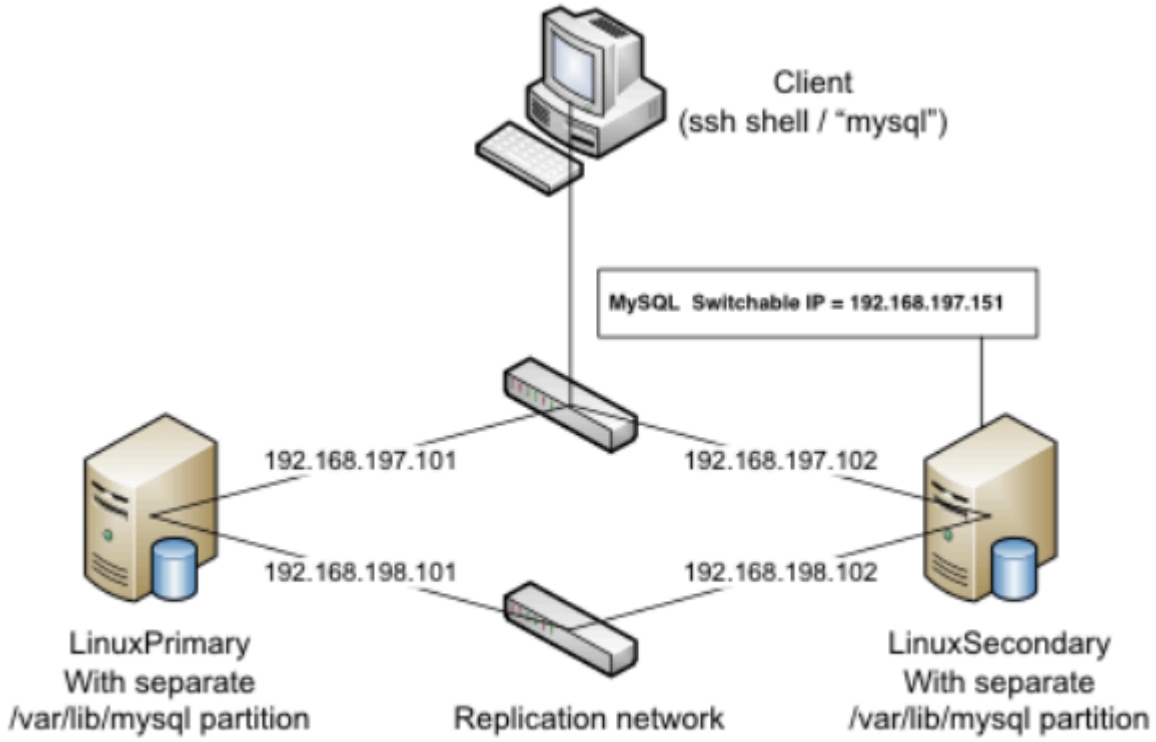
- See Configuring GUI Users for more information.

# Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi-homed, between two LAN segments (the second NIC in each server could even be connected via a cross-over cable if a second physical network is not available). The second NIC is optional in this configuration, but highly recommended in production environments to avoid a single point of failure.

In this example evaluation scenario we will be leveraging local, replicated storage.




**Network Configuration Example**

## Primary Server and Secondary Servers

- Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

  Example:
  192.168.197.101 LinuxPrimary
  192.168.197.102 LinuxSecondary

- See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.
- Public Network connection(s) configured with:

    - Static IP address
    - Correct subnet mask
    - Correct gateway address
    - Correct DNS server address(es)

- Private Network connection(s) configured with:
    - Static IP address (on a different subnet from the public network)
    - Correct network mask
    - No gateway IP address
    - No DNS server addresses

## Client

Must be able to communicate on the same subnet/network as the servers Public interface addresses. In our example, this is the 192.168.197.0/24 network.

# 12.6.4. Configure Storage

## Before You Begin

### Ensure the following:

- You have an extra disk/partition on both servers that can be used for data replication. A target volume's size must be equal to or larger than the size of its source disk/partition.

## Partition local storage for use with SIOS Data Replication

### Primary Server

On your Primary server, perform the following actions:

1. Identify an existing free, unused disk partition to use as our Apache repository. Alternatively, create a new partition. Use the "gdisk" utility to partition your disk appropriately. In this example /dev/sdb is an unused disk where we will create a single partition

   a. gdisk /dev/sdb

   b. Press "n" to create a new partition

   c.This example uses a new disk, so we will use all default values (Partition 1, entire disk and Linux filesystem partition type) Hit Enter four times to confirm these parameters.

   d. Press "w" to write the partition table

   e. Press "Y" to confirm to overwrite existing partitions

**Example**

[root@LinuxPrimary ~]# **gdisk /dev/sdb**

GPT fdisk (gdisk) version 0.8.10

Partition table scan:
MBR: protective
BSD: not present
APM: not present
GPT: present

Found valid GPT with protective MBR; using GPT.

Command (? for help): **n**
Partition number (1-128, default 1): **<enter>**

First sector (34-2047, default = 34) or {+-}size{KMGTP}: **<enter>**
Last sector (34-2047, default = 2047) or {+-}size{KMGTP}: **<enter>**
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): **<enter>**
Changed type of partition to 'Linux filesystem'

Command (? for help): **w**

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING PARTITIONS!!

Do you want to proceed? (Y/N): **Y**
OK; writing new GUID partition table (GPT) to /dev/sdb.
Warning: The kernel is still using the old partition table.
The new table will be used at the next reboot.
The operation has completed successfully.

[root@LinuxPrimary ~]#

2. Format the newly created disk partition

   # mkfs.ext3 /dev/sdb1

3. Mount the partition temporarily at /mnt

   # mount /dev/sdb1 /mnt

4. Move any existing data from /var/lib/mysql/ into this new disk partition (assumes a default MySQL configuration)

   # cd /var/lib/mysql
   # mv * /mnt

5. Remount /dev/sdb1 at /var/lib/mysql

   # cd /root
   # umount /mnt
   # mount /dev/sdb1 /var/lib/mysql

6. Note: there is no need to add this partition to /etc/fstab. Lifekeeper will take care of mounting this automatically.

**Result:**

[root@LinuxPrimary ~]# df /var/lib/mysql
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sdb1 253855 11083 229666 5% /var/lib/mysql<

## Secondary Server

7. On your Secondary server, only perform Step #1 above, where you partition the disk. The size of the Target disk/partition needs to be the same size, or greater, than our Source disk/partition.

# 12.6.5. Install, Configure, and Start MySQL

## Primary Server

On your Primary server, perform the following actions:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

   ```
   # yum install mysql mysql-server
   ```

2. Verify that your local disk partition is still mounted at /var/lib/mysql via the “df” command

3. If this is a fresh MySQL install, initialize a sample MySQL database:

   ```
   # /usr/bin/mysql_install_db – –datadir=”/var/lib/mysql” – –user=mysql
   ```

4. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

   ```
   # chown –R mysql:mysql /var/lib/mysql
   ```

   ```
   # chmod 755 /var/lib/mysql
   ```

5. Finally, manually start the MySQL daemon from the command line.

**Do Not** start it via the “service” command, or the /etc/init.d/ scripts

```
# mysqld_safe – –user=root – –socket=/var/lib/mysql/mysql.sock – –port=3306 – –datadir=/var/lib/mysql – –log &
```

6. Verify MySQL is running by connecting with the mysql client:

   ```
   [root@LinuxPrimary mysql]# mysql

   Welcome to the MySQL monitor. Commands end with ; or \g.

   Your MySQL connection id is 2

   Server version: 5.0.77-log Source distribution

   Type ‘help;’ or ‘\h’ for help. Type ‘\c’ to clear the buffer.

   mysql> exit
   ```

Bye

[root@LinuxPrimary mysql]#

7. Update the root password for your mysql configuration. In this example we set the MySQL root password to “SteelEye”

   # echo “update user set Password=PASSWORD where User=‘root’; flush privileges” | mysql mysql

8. Verify your new password:

   # mysql mysql –u root –p

   (Enter “SteelEye” as the password)

   #exit

9. Create a MySQL configuration file. We will place this in the same shared directory (/var/lib/mysql/my.cnf)

   # vi /var/lib/mysql/my.cnf

**Example**

# cat /var/lib/mysql/my.cnf

[mysqld]

datadir=/var/lib/mysql

socket=/var/lib/mysql/mysql.sock

pid-file=/var/lib/mysql/mysqld.pid

user=root

port=3306

# Default to using old password format for compatibility with mysql 3.x

# clients (those using the mysqlclient10 compatibility package).

old_passwords=1

# Disabling symbolic-links is recommended to prevent assorted security risks;

# to do so, uncomment this line:

```
# symbolic-links=0

[mysqld_safe]

log-error=/var/log/mysqld.log

pid-file=/var/run/mysqld/mysqld.pid

[client]

user=root

password=SteelEye
```

10. Delete the original MySQL configuration file, located in /etc

```
# rm /etc/my.cnf
```

## Secondary Server

On your Secondary Server:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

   ```
   # yum install mysql mysql-server
   ```

2. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

   ```
   # chown –R mysql:mysql /var/lib/mysql

   # chmod 755 /var/lib/mysql
   ```

3. There is no need to perform any of the additional steps taken on the Primary Server

# 12.6.6. Install SIOS Protection Suite for Linux – MySQL Cluster

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

| SPS for Linux Component | Install Location |
|---|---|
| LifeKeeper Software | /opt/LifeKeeper |
| LifeKeeper Config File | /etc/default/LifeKeeper |

Perform the following actions on **both** Primary and Secondary server.

## Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.

2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After you have successfully downloaded the software you will have downloaded contents similar to the follow directory listing:

   [root@LinuxPrimary ~]# ls -l <directory> total 63680

   -rw-r—r— 1 root root 23163 May 30 14:03 EULA.pdf

   -rw-r—r— 1 root root 536 May 30 14:03 readme.txt

   -rw-r—r— 1 root root 65179648 May 30 14:03 sps.img

3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

## Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

   a. # mount -o loop sps.img /mnt

b. # cd /mnt

c. # ./setup

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point.

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkSQL-<version>.noarch.rpm

   b. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

   a. # cd /root

   b. # umount /mnt

# Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

   # /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

   # /opt/LifeKeeper/bin/lklicmgr
   License File: 20101230.lic

| Product | Type | Expiry |
|---|---|---|
| LifeKeeper for Linux | Eval | 27 Mar 2013 (87 days) |

| | | |
|---|---|---|
| SIOS Data Replication ARK | Eval | 27 Mar 2013 (87 days) |
| MySQL Recovery Kit | Eval | 27 Mar 2013 (87 days) |

# Start the SIOS Protection Suite for Linux

1. Start

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

# 12.6.7. Configure the Cluster

## Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

> **Important** Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously.

### Access the LikeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 error.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

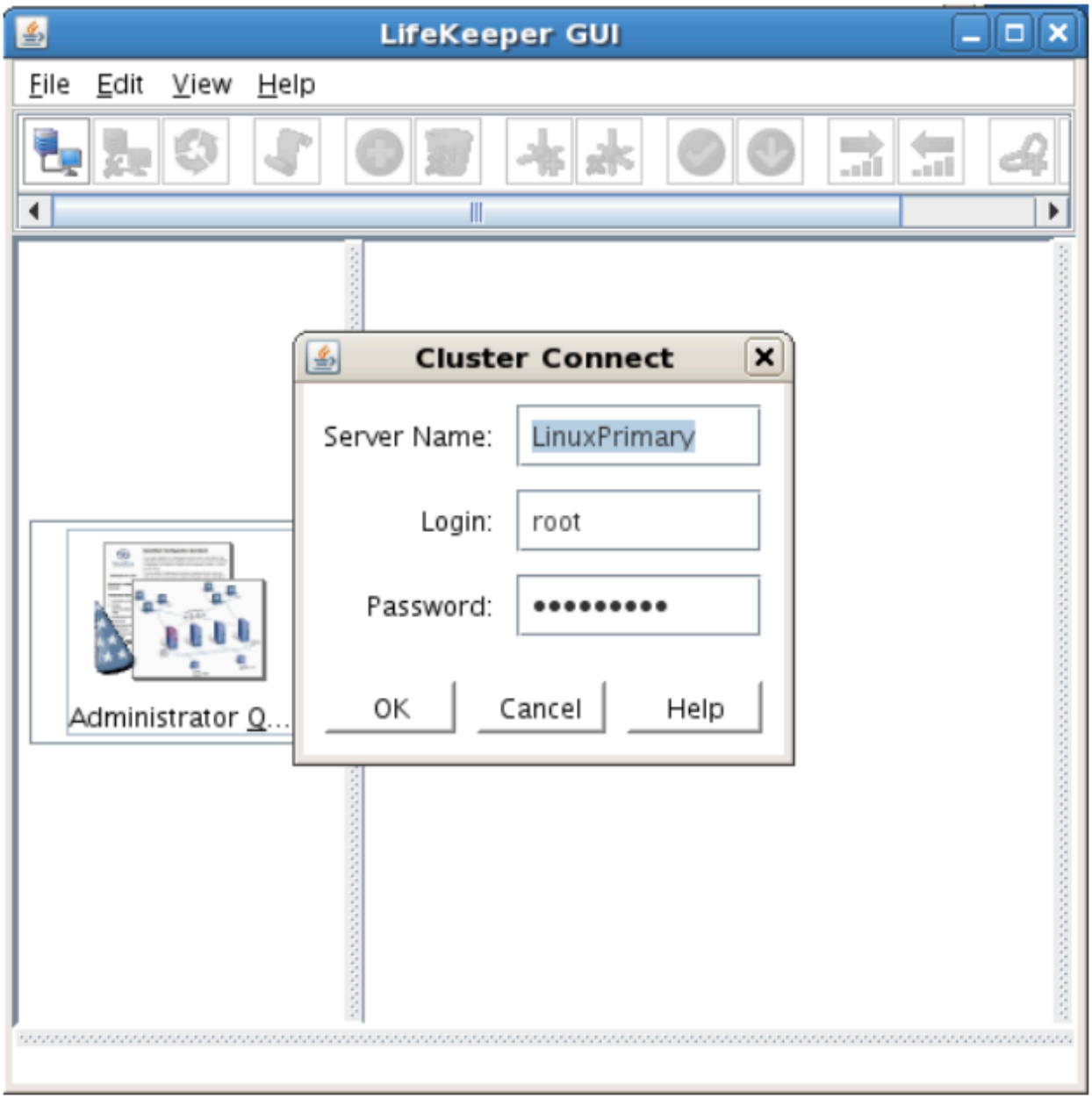
   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

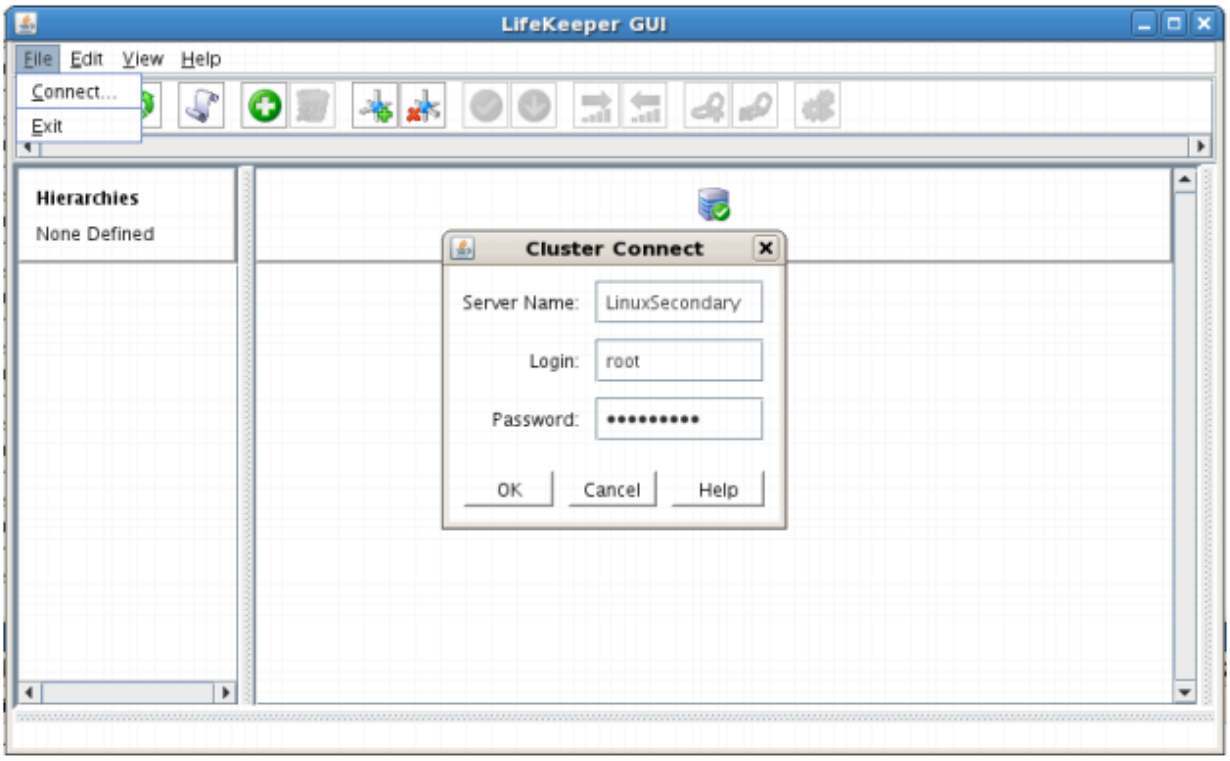
   /opt/LifeKeeper/bin/lkGUIapp &

3. Enter the name of the server you wish to connect to (this field will be populated with the name of the server you are on, if you are running the GUI from a server with LifeKeeper installed) along with your root credentials and click OK.



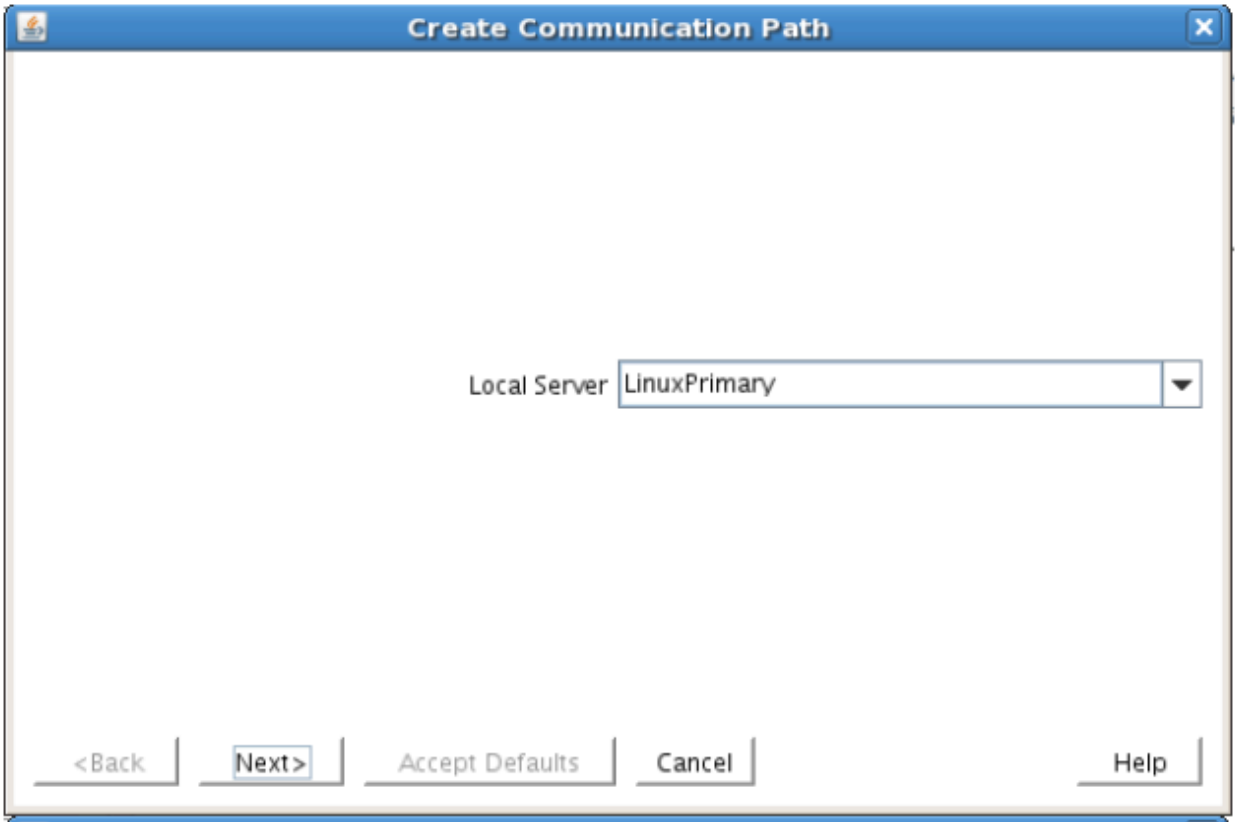

## Create Communication (Comm) Paths

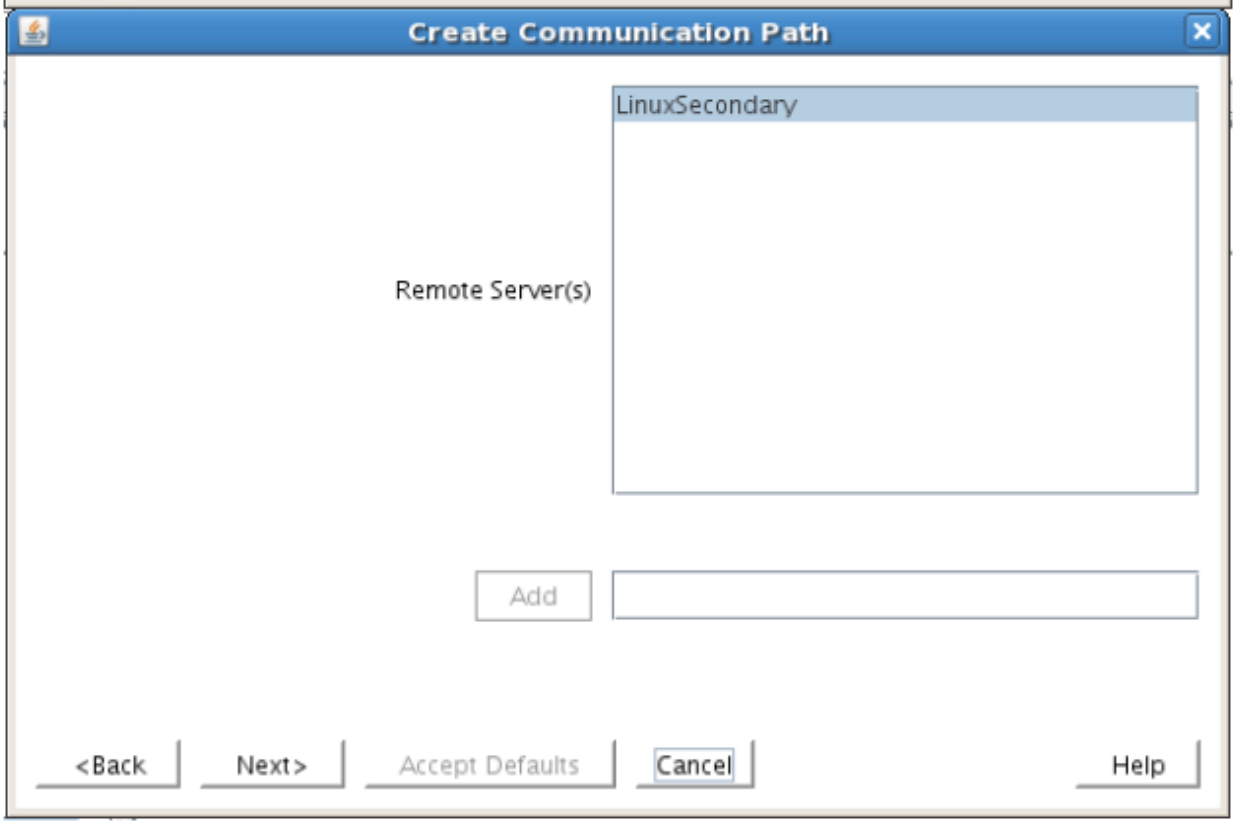
4. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



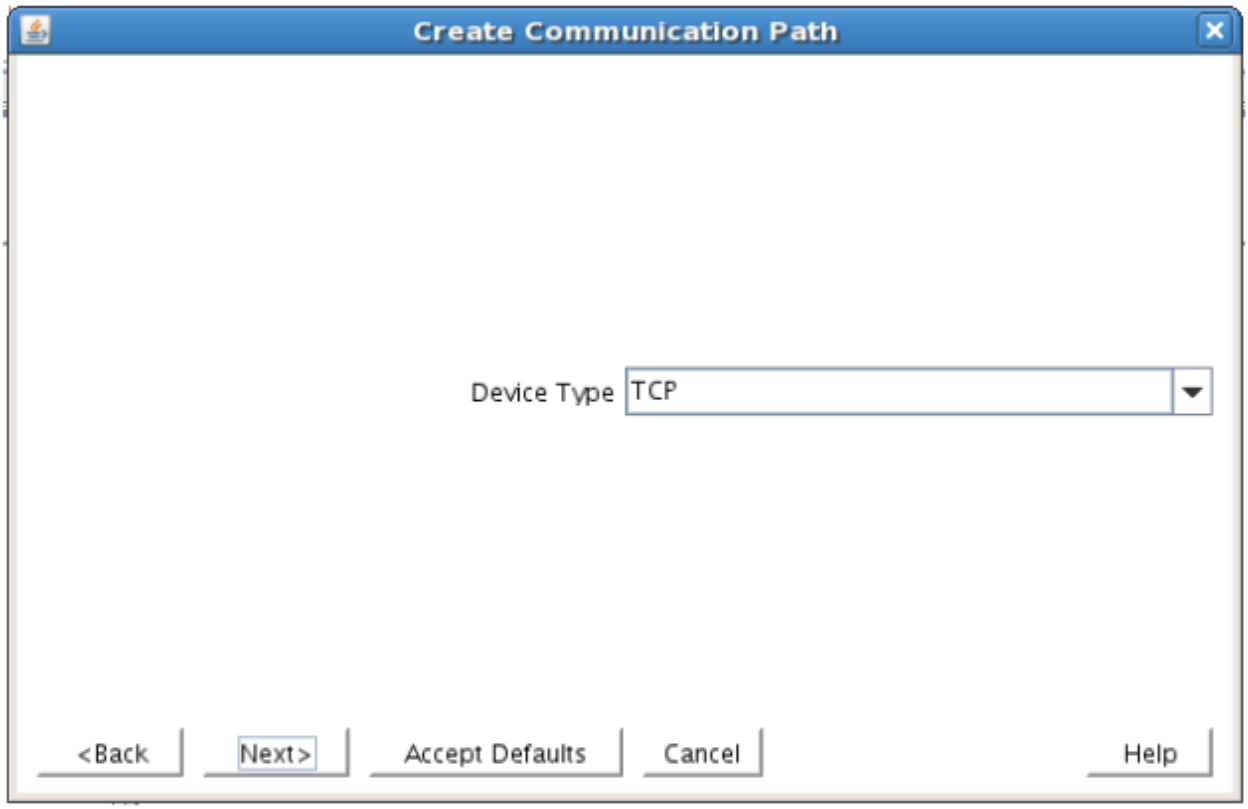

5. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

6. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

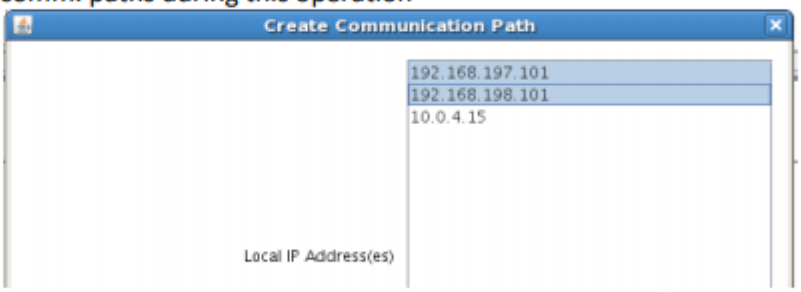


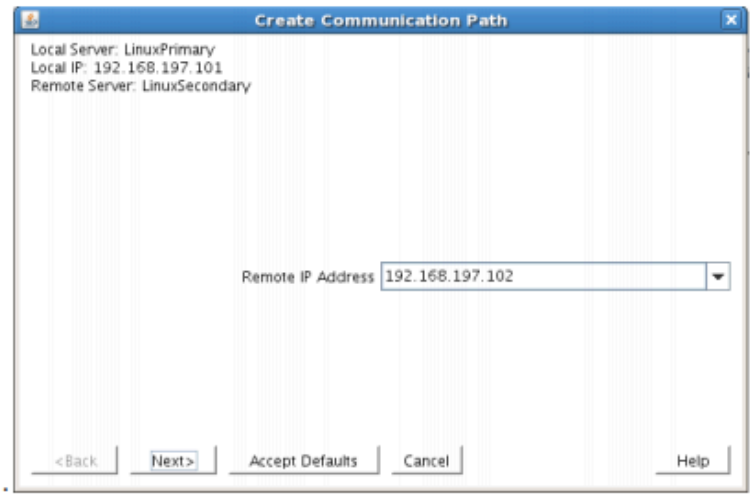




7. Select TCP for Device Type and Click Next.



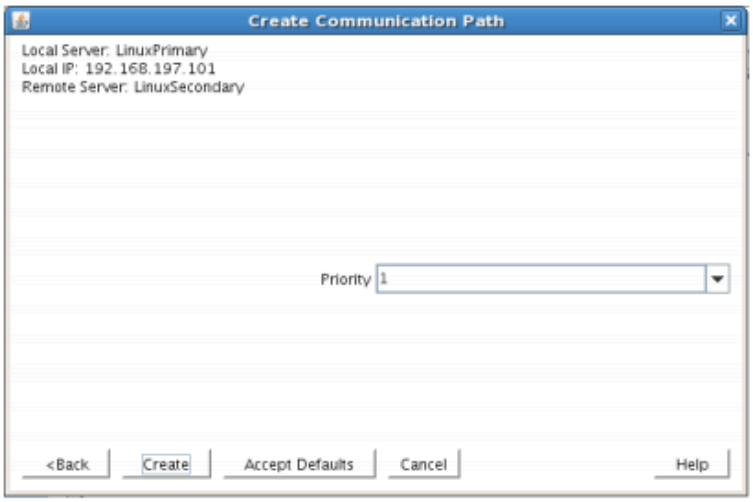

8. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

| Field | Tips |
|---|---|
| ***For TCP/IP Comm Path...*** | |
| Local IP Address | Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation   |
| Remote IP Address | Choose the IP address to be used by the remote server for this comm path |

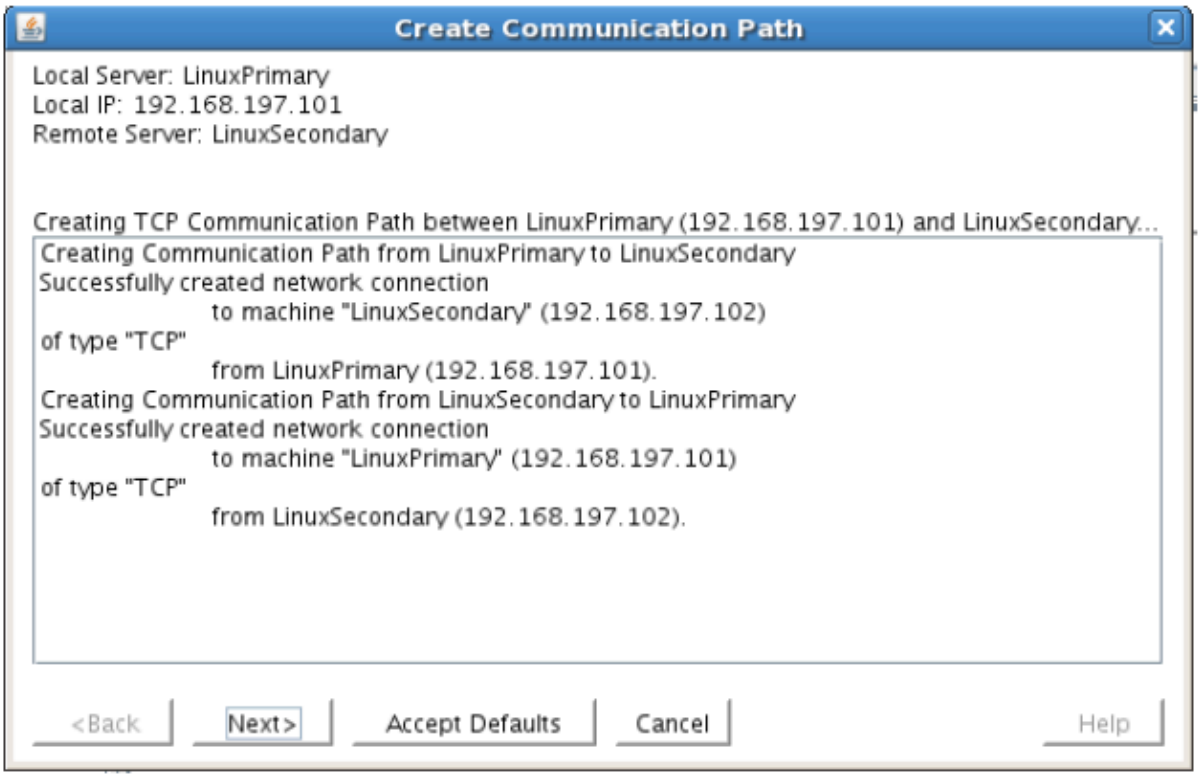



Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority




9. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



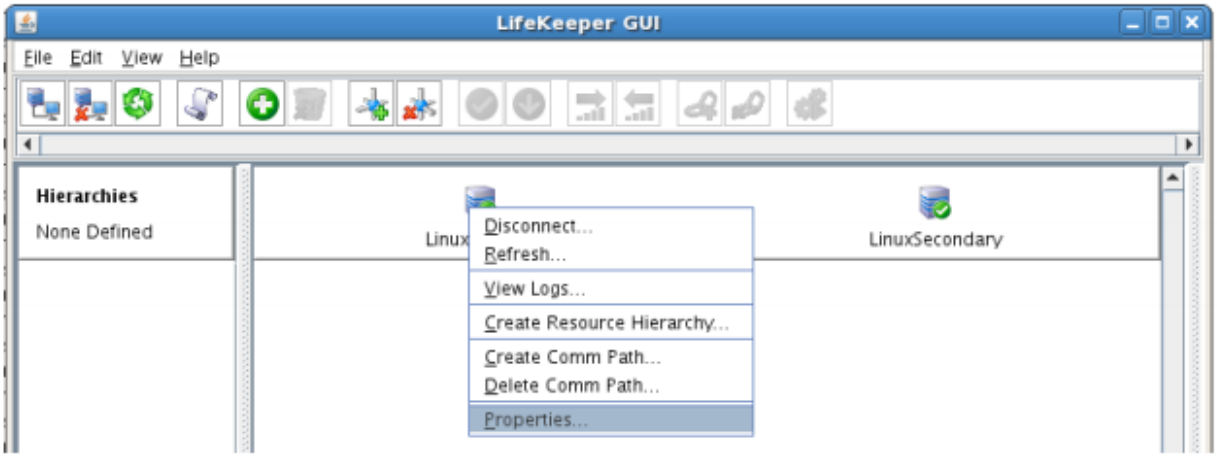

If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

10. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

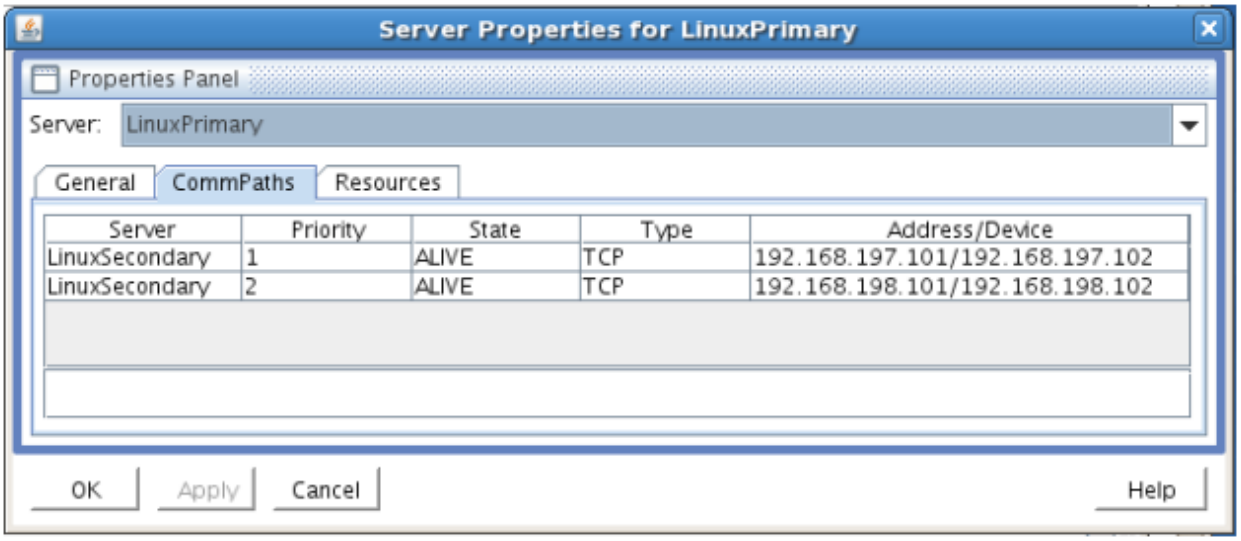
## Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.




2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.

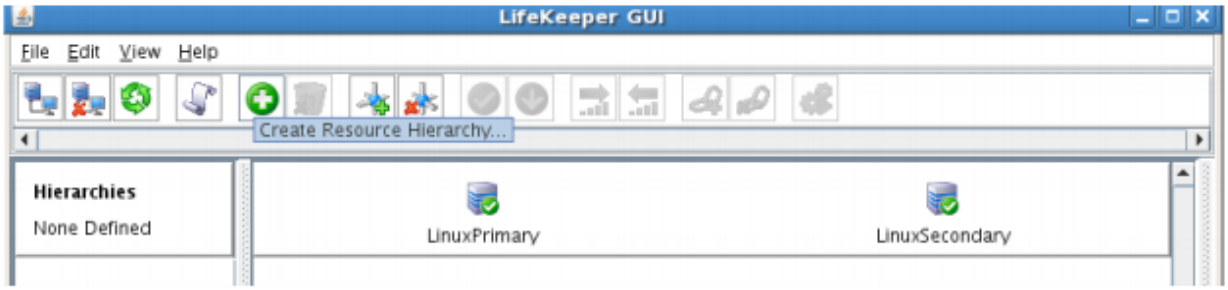



# Create the LifeKeeper Hierarchy

## Create and Extend an IP Resource

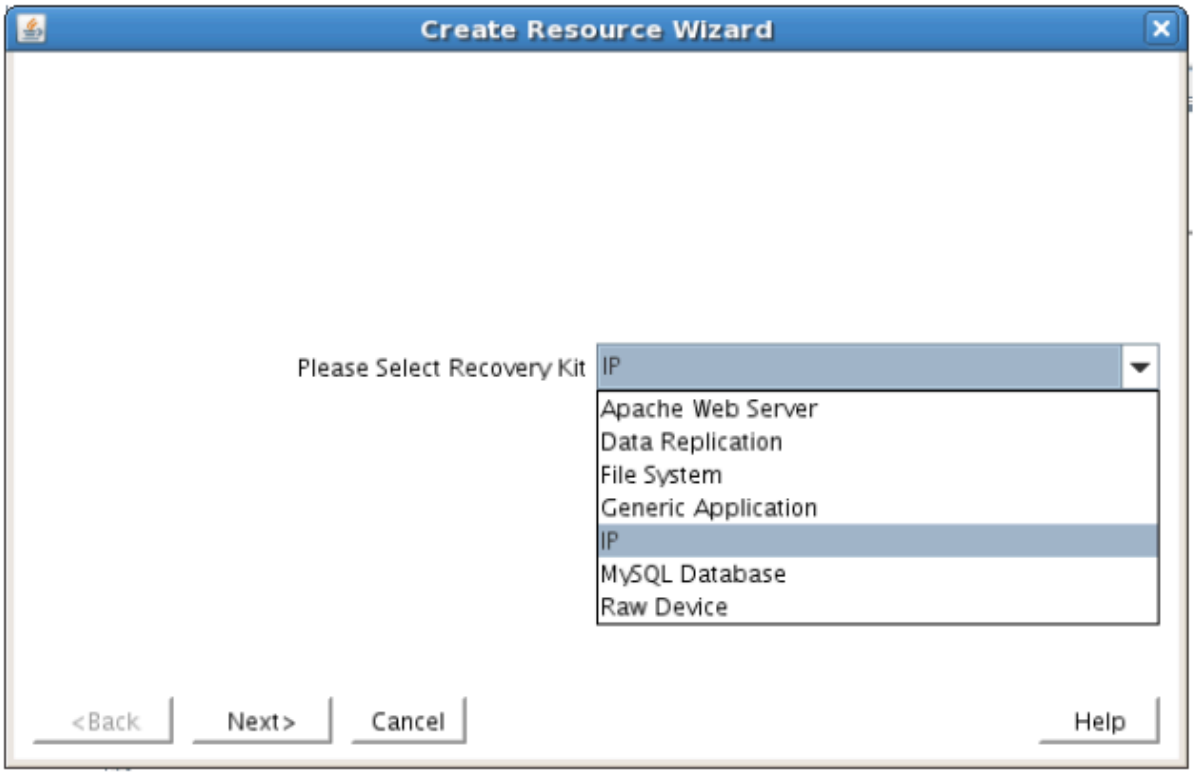
In LifeKeeper, create an IP resource and extend it to the secondary server by completing the following steps. This Virtual IP will have the ability to move between cluster nodes along the application that depends on it.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.




The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select IP Address and click Next.



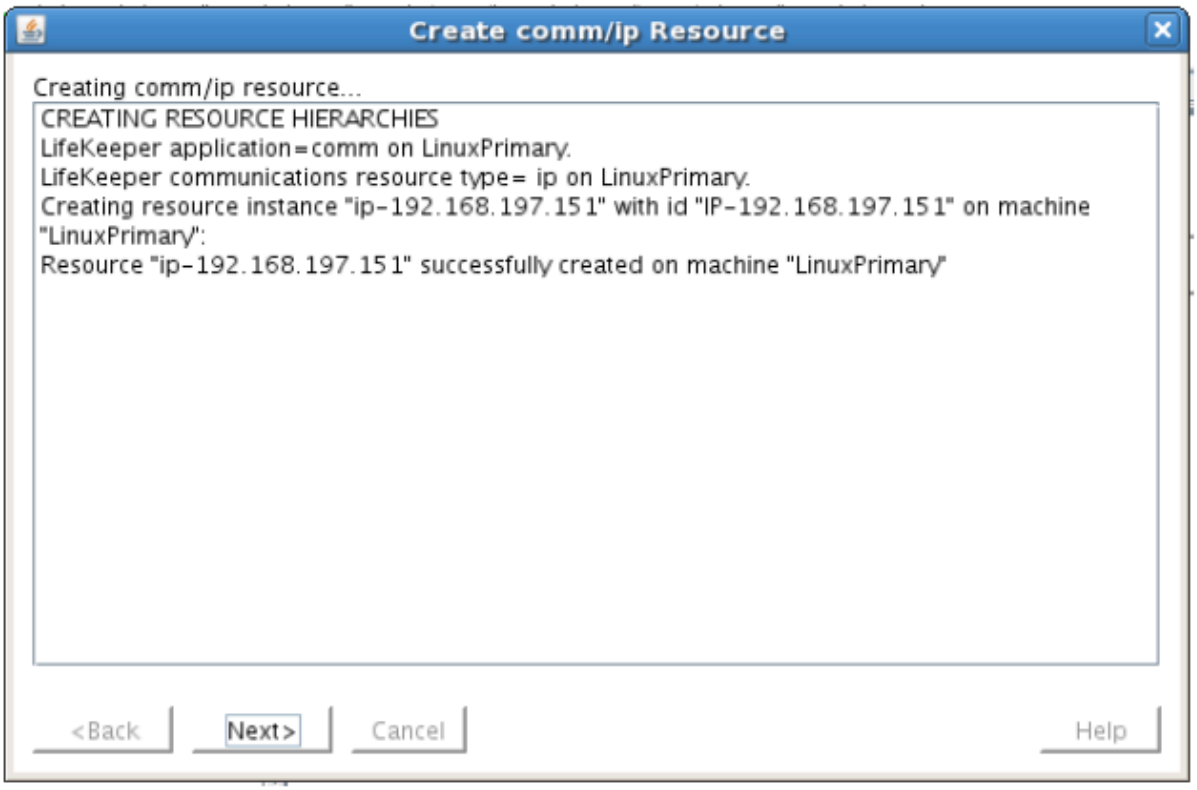

3. Enter the appropriate information for your configuration. The table below contains a list of the fields that display and additional information to assist you as you complete this procedure. Recommended values are also show below. You can also click the Help button for further information. Press Next to continue after entering the required information.

## IP Creation Field Definitions

| Field | Tips |
|---|---|
| Resource Type | Select IP Address as the resource type and click Next. |
| Switchback Type | Select Intelligent and click Next. |
| Server | Select the Server where the IP resource will be created. Select your<br><br>Primary server and click Next. |
| IP Resource | Enter the virtual IP information and click Next<br><br>Example **192.168.167.151**<br><br>**Note** This is an IP address that is not currently in use anywhere on your network. This is the address that all clients will use to connect to the protected resources. |

| | |
|---|---|
| Netmask | The IP subnet mask that your TCP/IP resource will use on the target server. Any standard netmask for the class of the specific TCP/IP resource address is valid.<br><br>In our sample configuration 255.255.255.0 is used for a subnet mask on both networks.<br><br>**Note:** The subnet mask you choose, combined with the IP address, determines the subnet that will be used by the TCP/IP resource and should be consistent with the network configuration. |
| Network Connection | This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. Select the correct NIC and click Next. |
| IP Resource Tag | Accept the default value and click Next. This value only affects how the IP is displayed in the GUI. The IP resource will be created on our Primary server. |

4. LifeKeeper will create and validate your resource. After receiving the message that the resource has been created successfully, click Next when the following dialog box appears so that you can complete the process of Extending the IP Resource to our Secondary server, below.



Extending the IP resource will start automatically after you have finished creating an IP address resource if you clicked Next in the dialog box displayed above. You can also start this from an existing IP address resource by right clicking on the active resource and selecting Extend Resource Hierarchy.

Refer to the table below to complete the Extend IP Resource procedure.

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Leave as “intelligent” and click Next |
| Template Priority | Leave as default (1) |
| Target Priority | Leave as default (10) |
| Network Interface | This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. The correct physical NIC should be selected by default. Please verify and then click Next |
| IP Resource Tag | Leave as default. |
| Target Restore Mode | Select Enable and click Next. |
| Target Local Recovery | Select Yes to enable Local Recovery for the SQL resource on the Target server. |
| Backup Priority | Accept the default value. |

5. After receiving the message Hierarchy extend operations completed, click Finish and then click Done

6. Your IP resource (192.168.197.151) is now fully protected and has the ability to “float” between cluster nodes as needed. Looking at the LifeKeeper GUI you will notice that the IP resource is Active on the Primary cluster node and Standby on the Secondary cluster node

## Create a Mirror and Begin Data Replication

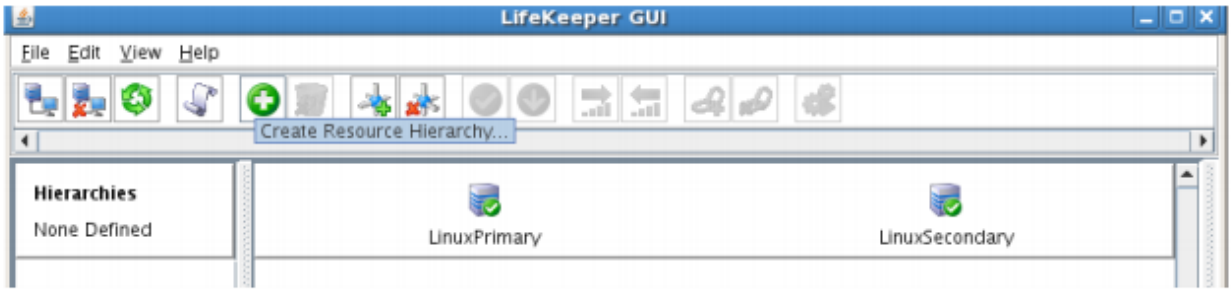
In this section we will setup and configure the Data Replication resource, which be used to synchronize

our MySQL's data between cluster nodes. The data we will replicate resides in the /var/lib/mysql partition on our Primary cluster node

Please note:

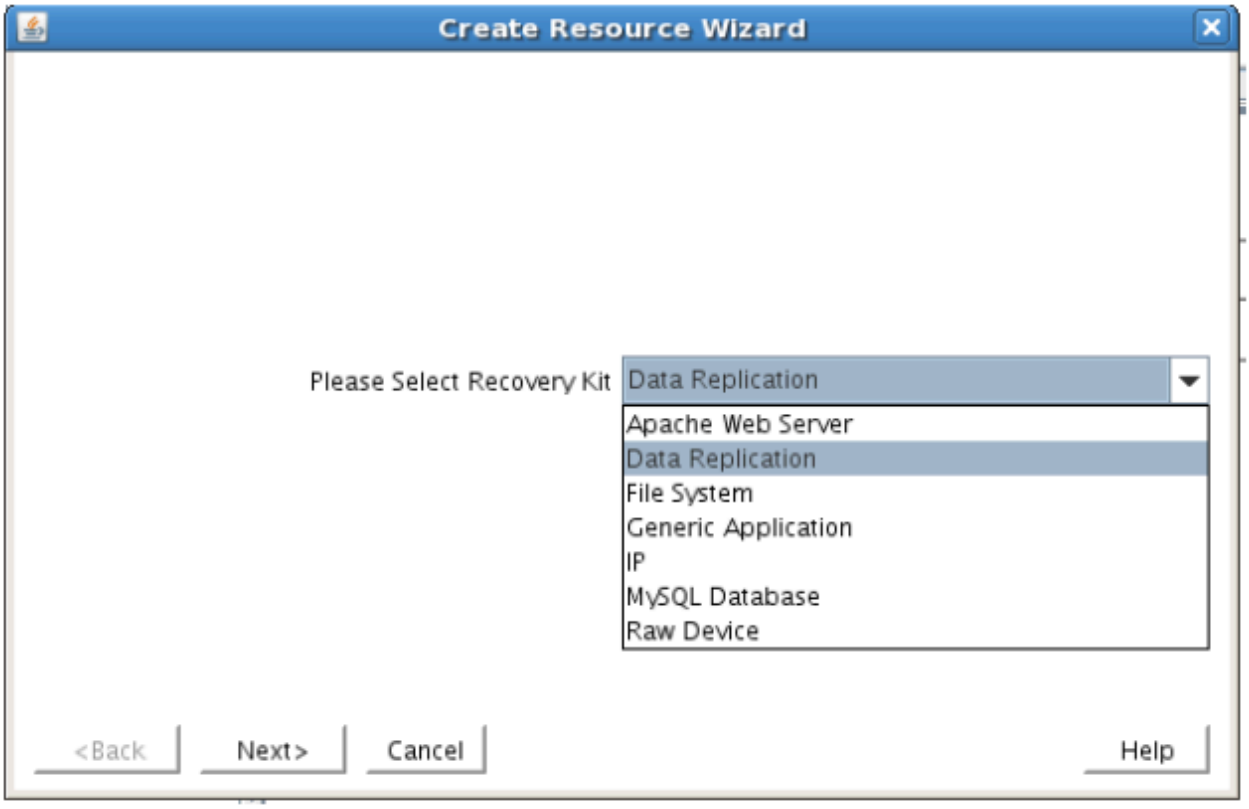
- The source volume to be replicated must be mounted on the Primary server
- The target volume, which will received replicated data, must **NOT** be mounted on the Secondary server.
- The target volume's size must equal to or larger than the size of its source volume.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.




The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

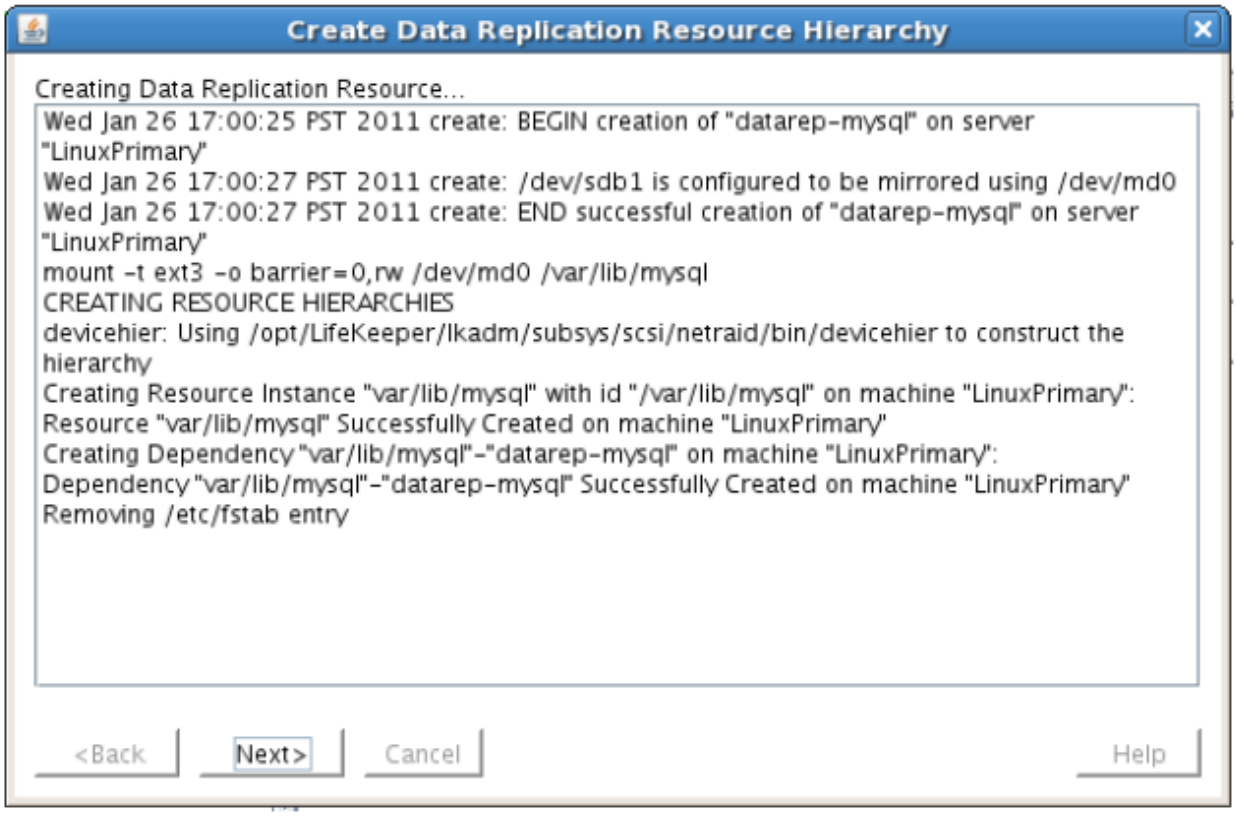
2. Select Data Replication and click Next.



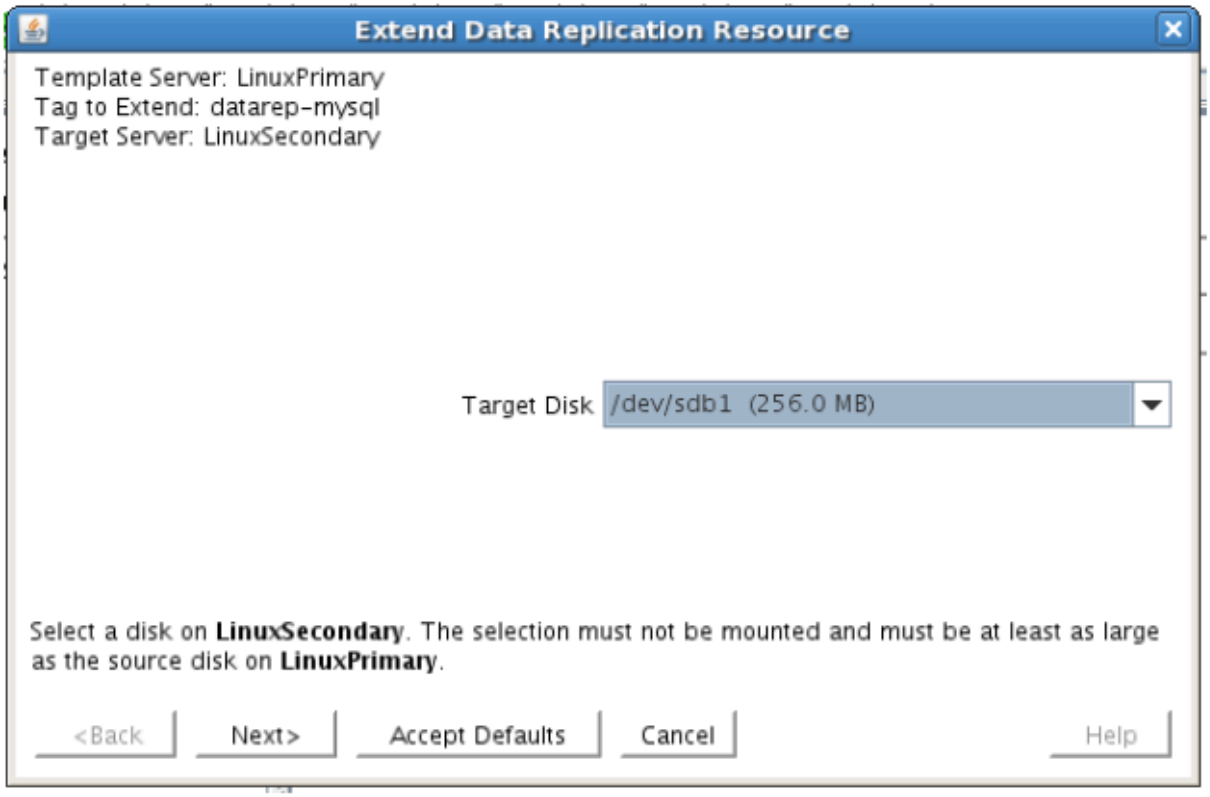
3. Follow the Data Replication wizard, and enter the following values:

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node, i.e. Mirror Source) |
| Hierarchy Type | Select: “**Replicate Existing Filesystem**” |
| Existing Mount Point | At this step you will select the mounted partition to replicate. In our example, select “/var/lib/mysql” |
| Data Replication Resource Tag | Leave as default |
| File System Resource Tag | Leave as default |
| Bitmap File | Leave as default (Note: if using high speed SSD storage you will want to create a small partition and use it for bitmap placement, i.e. /bitmaps) |
| Enable Asynchronous Replication | Leave as default (Yes) |

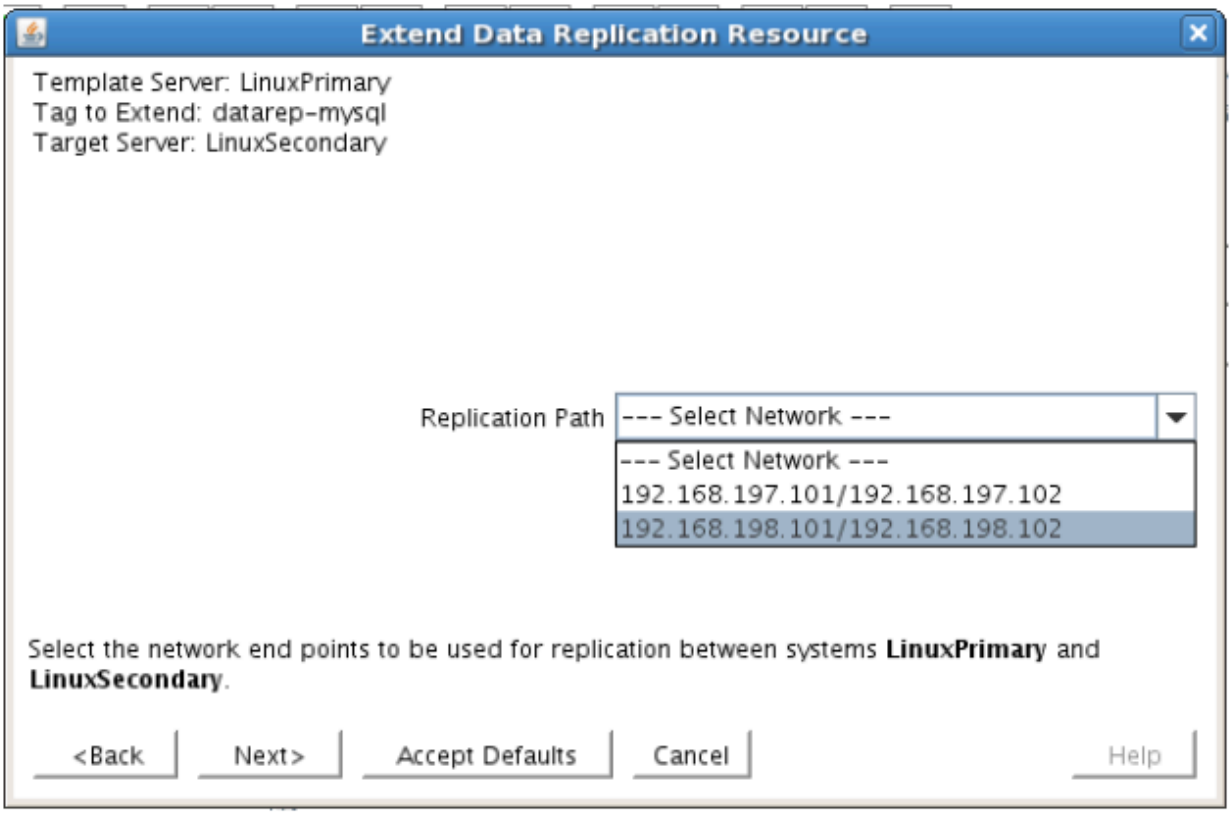
4. Click Next to begin creation of the Data Replication resource hierarchy. Status will be displayed in the GUI as follows:



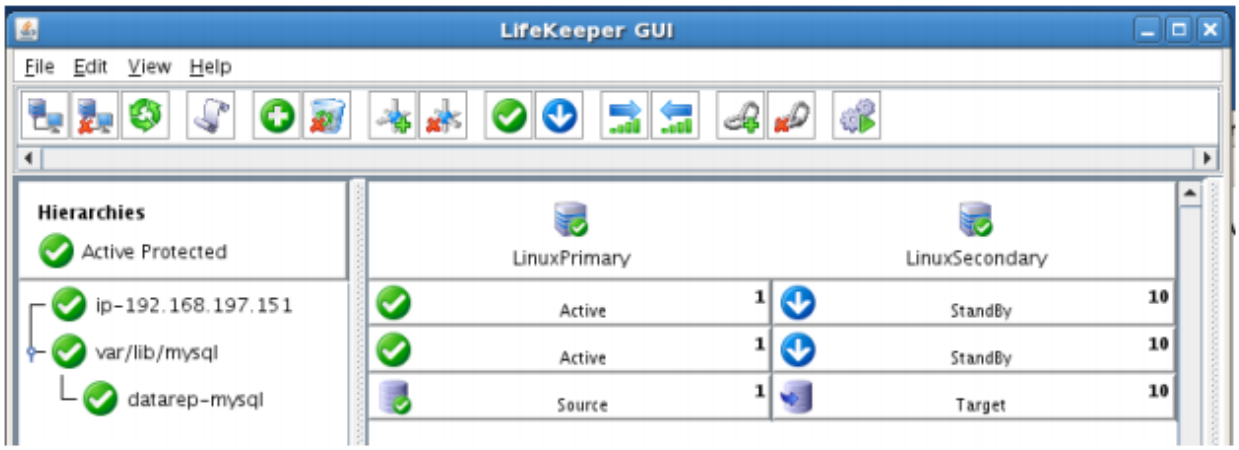
5. Click Next to begin the process to Extend the Data Replication Resource. Select all default settings. When it asks for the target disk, select a free partition on your Target server which is the same size (or greater) than the Source Volume we are replicating. This partition should NOT be mounted on the Target system.




6. Continue through the wizard, and you will be prompted to select the network you would like replication to take place over. In general, it’s a best practice to separate your user/application and your replication traffic. In our example setup we will replicate over our backend network, 192.168.198.X




7. Click Next and continue through the wizard. Once completed, your resource hierarchy will look as follows




## Create the MySQL Resource Hierarchy

Create a MySQL resource to protect the MySQL database and make it high available between cluster nodes.

**Important** At this point, MySQL needs to be already running on the Primary Server. It should NOT be running on the Secondary Server. See section titled “Install, Configure, and Start MySQL” above to review the process to configure and start MySQL as needed.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select **MySQL Database** and click **Next**.
3. Proceed Through the Resource Creation wizard, providing the following values

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node) |
| Location of my.cnf | Enter “/var/lib/mysql”. Note that earlier in the MySQL |
| executables | configuration process we created a my.cnf file in this directory |
| Location of MySQL | Leave as default (/usr/bin) since we are using a standard MySQL install/configuration in this example |
| Database tag | Leave as default |

4. Select Create to define the MySQL resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the MySQL resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Note: LifeKeeper will automatically identify that the MySQL resource has a dependency on the FileSystem (Data Replication) resource (/var/lib/mysql). The Filesystem Resource will appear underneath the MySQL resource in the GUI
9. Your resource hierarchy should look as follows:

## Create the MySQL IP Address Dependency

In this step will define an additional dependency: that MySQL depends on a Virtual IP (192.168.197.151) so that the IP address follows the MySQL database as it moves.

1. From the LifeKeeper GUI toolbar, right-click on the “mysql” resource
2. Select “Create Dependency” from the right-click context menu
3. In the Child Resource Tag dropdown menu, select “ip-192.168.197.151”
4. Click Next
5. Click Create Dependency
6. Click Done
7. The Virtual IP address resource (192.168.197.151) will now appear underneath the MySQL resource in the LifeKeeper user interface. This ensures that resources move together, and are started/stopped in the proper order.
8. Your resource hierarchy should look as follows

At this point in the Evaluation, we have fully protected MySQL, and its dependent resources: IP addresses, and replicated storage.

# 12.6.8. Test Your Environment

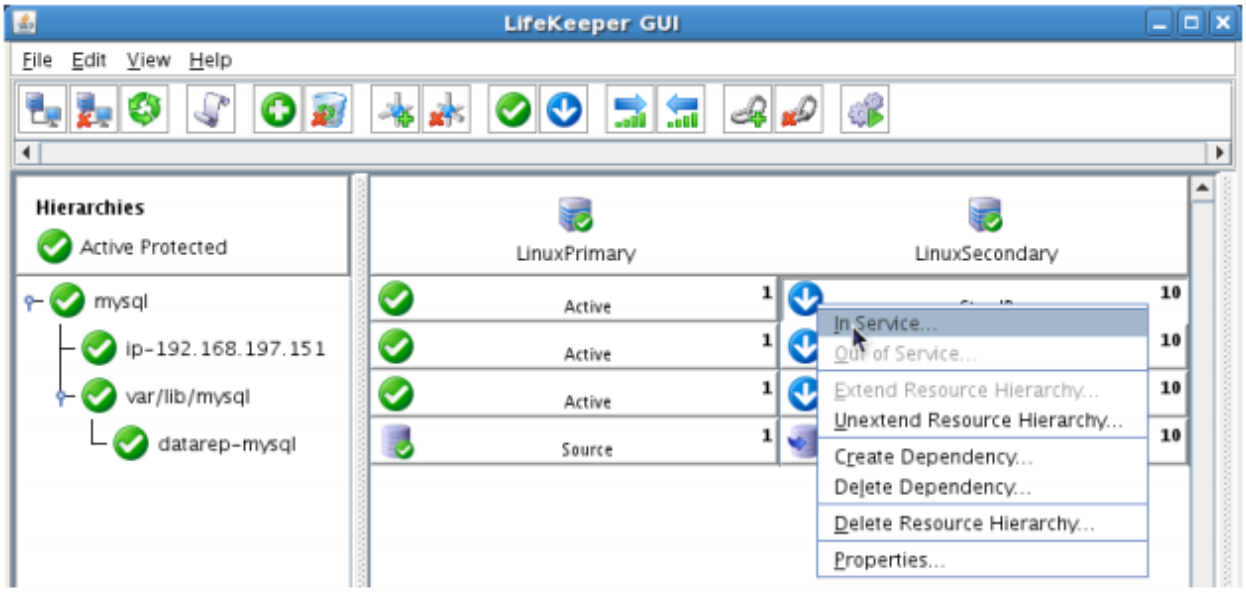
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

> **Note:** For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

## Manual Switchover of the MySQL Hierarchy to Secondary Server

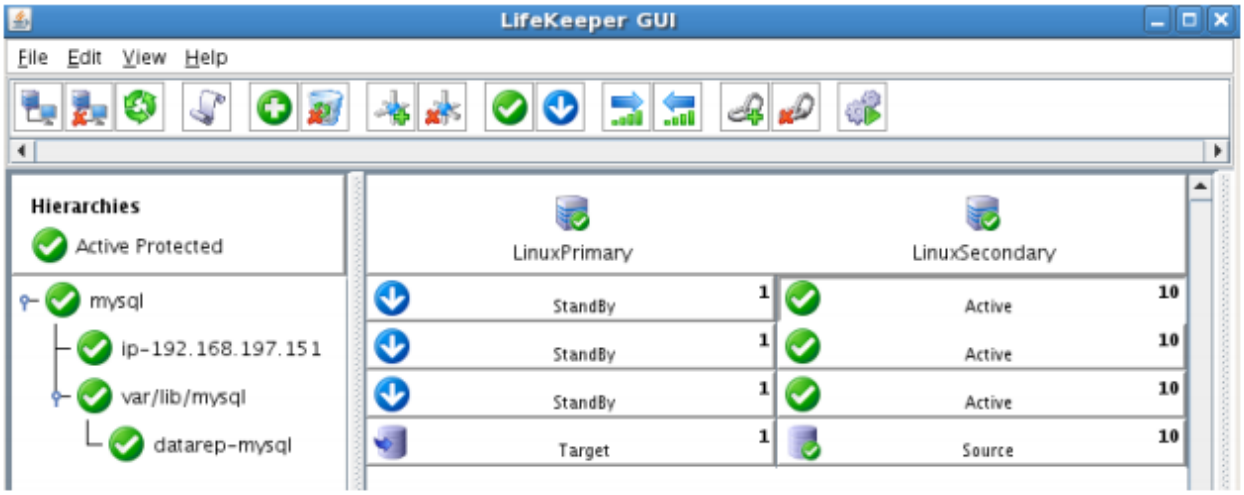
**Procedure:**

- From the LifeKeeper GUI, right click on the MySQL resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



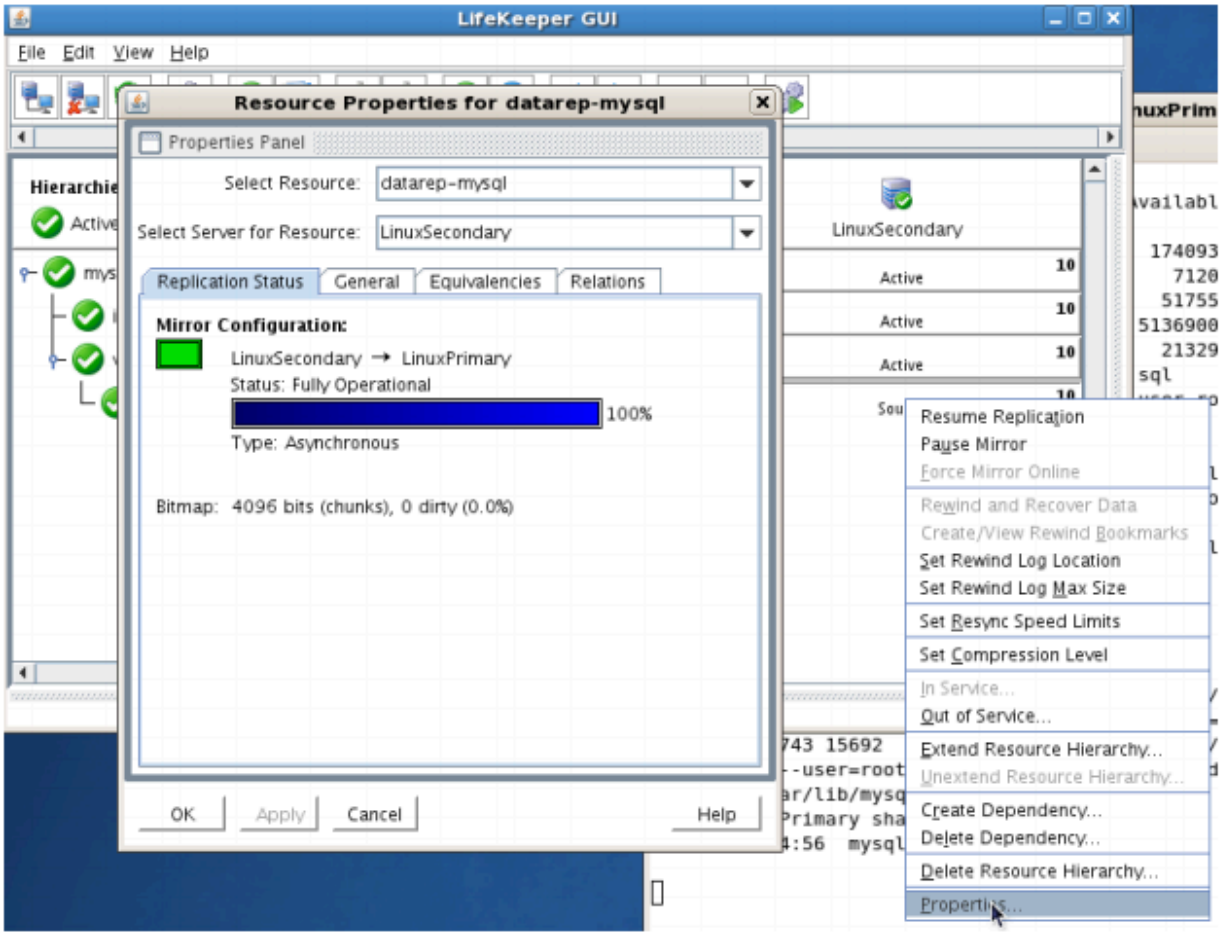

**Expected Result:**

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXSECONDARY.
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXSECONDARY -> LINUXPRIMARY
- At this point, all resources are now active on LINUXSECONDARY.




**Tests/Verification:**

- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXSECONDARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties




- Run “ifconfig –a” on LINUXSECONDARY to validate that the IP Address 192.168.197.151 is active on LINUXSECONDARY
- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device

(example: /dev/md0”) on LINUXSECONDARY
- Verify the MySQL services are running on LINUXSECONDARY by running “ps –ef | grep –i mysql”
- On LINUXSECONDARY run the following command to verify client connectivity to the MySQL database:

    - # mysql –S /var/lib/mysql/mysql.sock –u root –p
    - (enter password “SteelEye”)

## Manual Switchover of the MySQL Hierarchy back to Primary Server

**Procedure:**

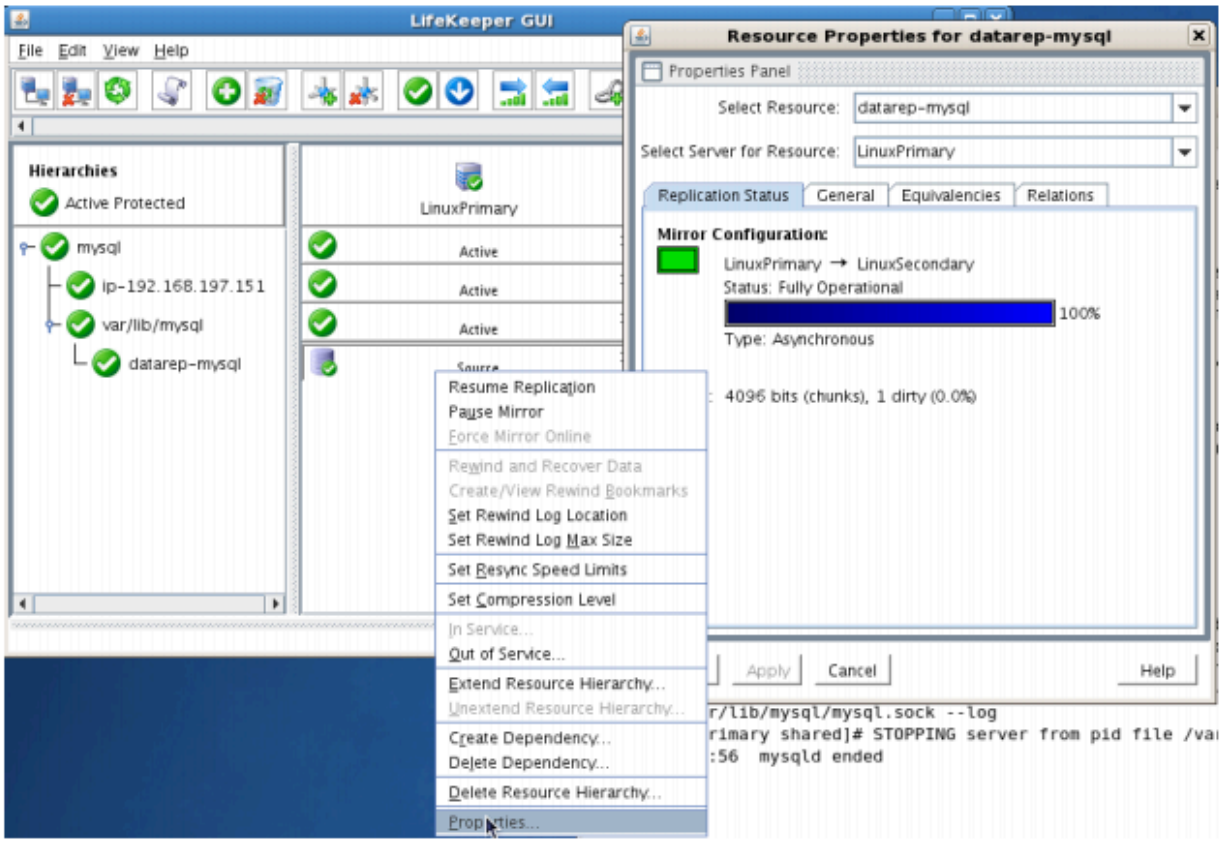
- From the LifeKeeper GUI, right click on the MySQL resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

**Expected Result:**

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXPRIMARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXPRIMARY -> LINUXSECONDARY

**Tests/Verification:**

- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXPRIMARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-mysql” resource and select Properties




- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.151 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/lib/mysql replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY
- Verify the MySQL services are running on LINUXPRIMARY by running “ps –ef | grep –i mysql”
- On LINUXPRIMARY run the following command to verify client connectivity to the MySQL database:

    - # mysql –S /var/lib/mysql/mysql.sock –u root –p
    - (enter password “SteelEye”)

## Simulate a network failure on the Primary Server by failing the IP resource

> **IMPORTANT:** Only perform this test if you have more than one communications path configured.

If you perform this test and have only one communications path configured, your system will enter a split-brain scenario as described in the LifeKeeper Administration Guide found here. Refer to this document for more information or contact SIOS presales technical support for assistance in resolving this condition.

**Procedure:**

- On LINUXPRIMARY, pull the network cable attached to the NIC that the virtual IP address is configured on

**Expected Result:**

- The IP Resource should fail first.
- The entire hierarchy should failover to LINUXSECONDARY

**Tests/Verification:**

- Check the LifeKeeper Log to verify the IP resource failed – “/opt/LifeKeeper/bin/lk_log log”
- Using the LifeKeeper GUI, verify the MySQL and Apache resource hierarchies fail over successfully to LINUXSECONDARY

## Hard failover of the resource from the Secondary Server back to the Primary Server

**Procedure:**

- Pull the power cord on LINUXSECONDARY, as this is the server with all resources currently In Service.

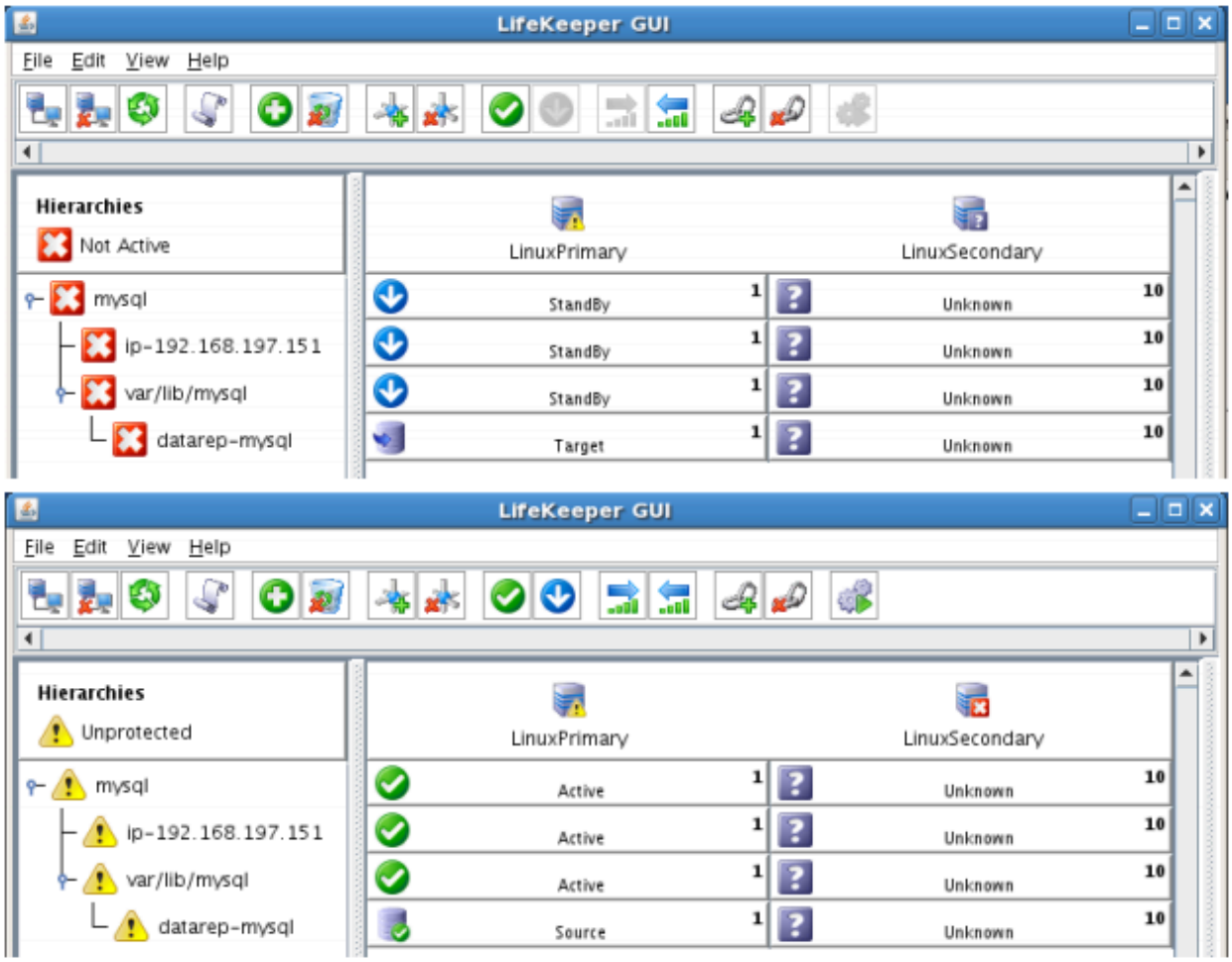
**Expected Result:**

- After failure has been detected, beginning with the dependent resources (IP and Volume), all resources will be brought in service on LINUXPRIMARY.

**Tests/Verification:**

- Using the LifeKeeper GUI, verify the mirror has reversed and is in a Resync Pending state waiting for LINUXSECONDARY to come back on line.

- Verify the Apache and MySQL Server services are running on LINUXPRIMARY.
- Verify that the client can still connect to the Webserver and database running on LINUXPRIMARY.
- Verify you can write data to the replicated volume, /var/lib/mysql on LINUXPRIMARY.




# Bring Failed Server back on line

**Procedure:**

- Plug the power cord back into LINUXSECONDARY and boot it up.

**Expected Result:**

- Using the LifeKeeper GUI, verify that LINUXSECONDARY is coming back up and has become the Standby Server.

**Tests/Verification:**

- Verify the mirror performs a quick partial resync and moves to the Mirroring state
- Verify the Apache and MySQL Hierarchy are in service on LINUXPRIMARY and standby on LINUXSECONDARY.

# Verify Local Recovery of MySQL Server

**Procedure:**

- Kill the MySQL processes via the command line:
    - # ps –ef | grep sql
    - # killall mysqld mysqld_safe
    - run “ps –ef | grep sql once again to verify that the processes no longer exist

**Expected Result: (Assumes Local Recovery for MySQL resource is set to YES)**

- The MySQL Server service should stop.
- The MySQL quickcheck process will automatically restart the MySQL Server Service when it runs periodically.
- No failure of MySQL should occur.

**Tests/Verification:**

- Execute “ps –ef | grep sql” once again to verify that the mysql processes have been restored locally on LINUXPRIMARY.

- Verify connectivity to the MySQL database by running:

    - mysql –S /var/lib/mysql/mysql.sock –u root –p
    - (Enter password “SteelEye”)

- If you inspect the LifeKeeper logs, you will see information indicating that LifeKeeper dectected the failure of the MySQL service and recovered it locally. Run /opt/LifeKeeper/bin/lk_log log for more information.

# 12.7. PostgreSQL Cluster with Shared Storage (ISCSI)

## Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product, to make PostgreSQL highly available. If PostgreSQL is not already installed, please allocate some time to install it on your servers. Once this task has been completed, you may install and configure SIOS Protection Suite for Linux.

There are five phases in this process:

- Phase 1 – Prepare to Install
- Phase 2 – Configure Storage
- Phase 3 – Install and Configure PostgreSQL
- Phase 4 – Install SIOS Protection Suite for Linux
- Phase 5 – Configure your LifeKeeper Cluster
- Phase 6 – Test Your Environment

# 12.7.1. Terms to Know – PostgreSQL

The following terms are used throughout this document and, while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

## Network Communication Terms

**Crossover cable** – A cable used to directly connect computing devices together, instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

## Types of LifeKeeper Servers

**Server** – A computer system dedicated to running software application programs.

**Active Server** – This is the server where the resource hierarchy is currently running (IN SERVICE).

**Standby Server** – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

**Primary Server** – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

**Secondary Server** – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

**Source Server** – In a LifeKeeper cluster, using data replication, this is the Active Server. It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

**Target Server** – In a LifeKeeper cluster, using data replication, this is the Standby Server. The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

## SIOS Data Replication Terms

**Replication** – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

**Synchronous** – A replication scheme in which the data is confirmed written and valid on the target before the write operation occurs on the source disk through a series of information exchanges. Synchronous mirrors should only be implemented on high speed (100Mbps+) networks due to the network overhead involved.

**Asynchronous** – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there and can be written on them.

**Rate of Change** – A measure of the amount of data which is changing over a set period of time.

**Compression** – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

**Throttling** – An optionally implemented mechanism to limit the bandwidth used for replication.

## LifeKeeper Product Terms

**Communications Path** – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

**Heartbeat** – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

**Split Brain** – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

**Failover** – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

**Switchover** – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

**Switchback** – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

**Resource** – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

**Extend a Resource** – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously.

**Resource Hierarchy** – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

**Shared Storage** – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally

called I/O fencing.

**Data Replication (Disk Mirroring)** – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

**Source** – The partition on the source server used for replication. The “gold” copy of the data.

**Target** – The partition on the target server used for replication.

**Switchable IP Address** – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

# 12.7.2. The Evaluation Process – PostgreSQL

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux in a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

**Important** Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

# 12.7.3. Prepare to Install – PostgreSQL

## Hardware Requirements

### Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional disk partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system ( / ) and boot (/boot) partitions are not eligible for replication.

**Note:** You may use more than one partition for replicated data, allowing for separation of multiple database files and/or log directories for performance reasons.

### Client

This system is not required but is recommended for testing the cluster environment.

- A standard linux terminal running the MySQL client can be used to test the configuration.

## Software Requirements

### Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
  - RedHat Enterprise Linux 5 (5.4+ recommended) or 6.x
  - CentOS Linux 5 (5.4+ recommended) or 6.x
  - Oracle Enterprise Linux 5 (5.4+ recommended), 6.3, 6.4
    - RedHat Compatibility Kernel Only
  - SuSE Linux Enterprise Server 10 or 11 (11 recommended)
  - See Linux Release Notes for a full list of supported Operating Systems
- Current patches / security updates are recommended.
- Satisfied dependencies; especially if the Linux installation package selection was base/minimal you will need to refer to the dependencies documentation at Linux Dependencies

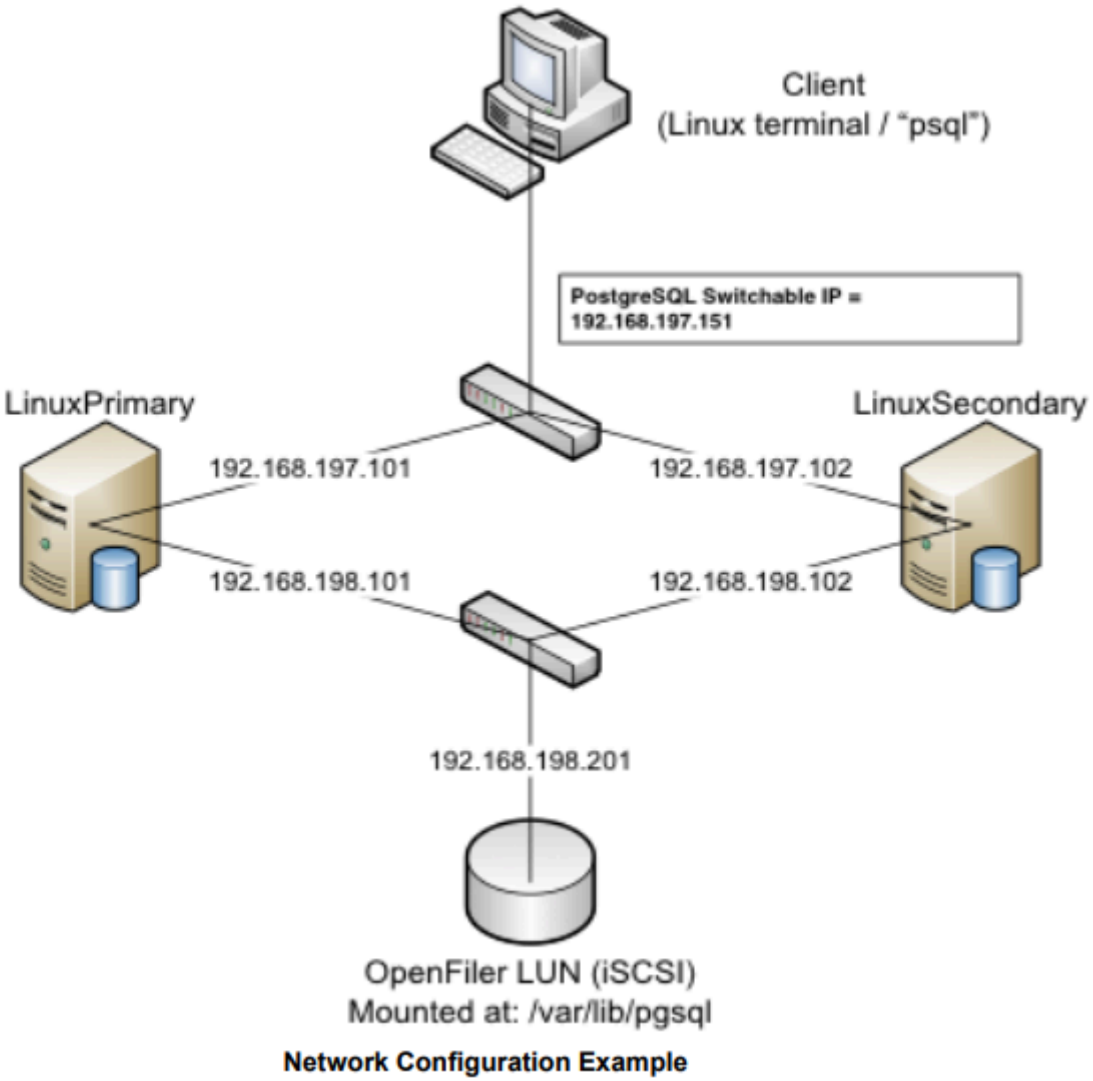
- Its recommended that IPtables is disabled
    - # /etc/init.d/iptables off
    - # chkconfig iptables off
    - See here for information regarding the ports SIOS Protection Suite for Linux uses.
- Disable SELinux :
    - Edit /etc/selinux/config
    - Set SELINUX=disabled (note: permissive mode is also acceptable)
- Check the configuration of your /etc/hosts file
    - localhost.localdomain and localhost are the only entries that can be on 127.0.0.1
    - Create a separate entry for your hostname with a static address
- GUI Authentication with PAM
    - SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).
    - Users are identified and authenticated against the system's PAM configuration. Privilege levels are determined from group membership as provided through PAM.
    - In order to access the GUI, a user must be a member in one of the three LifeKeeper groups: **lkadmin**, **lkoper** or **lkguest**.
    - See Configuring GUI Users for more information.

# Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi- in production environments to avoid a single point of failure.

In this example evaluation scenario we will be leveraging Shared (iSCSI) Storage with our PostgreSQL database. OpenFiler is a storage appliance server that will serve an iSCSI target to LinuxPrimary and LinuxSecondary.




**Network Configuration Example**

## Primary Server and Secondary Servers

- Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

  Example:

  192.168.197.101 LinuxPrimary

  192.168.197.102 LinuxSecondary

- See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.
- Public Network connection(s) configured with:
  - Static IP address
  - Correct subnet mask
  - Correct gateway address

    - Correct DNS server address(es)

- Private Network connection(s) configured with:
    - Static IP address (on a different subnet from the public network)
    - Correct network mask
    - No gateway IP address
    - No DNS server addresses

## Client

Must be able to communicate on the same subnet/network as the servers Public interface addresses. In our example, this is the 192.168.197.0/24 network.

# 12.7.4. Configure Storage – PostgreSQL

## Before you Begin

Ensure the following:

- If planning to use replicated storage, have an extra volume/partition on both servers that can be used for data replication. A target volume's size must equal to or larger than the size of its source volume.
- If planning to use shared storage, as in this example, ensure the Shared storage is configured and accessible to your cluster nodes. This can either be Fiber Channel SAN, iSCSI, NAS, etc. In this example we will review configuration of an iSCSI target for use as our PostgreSQL database storage repository.

## Configure iSCSI initiator, discover and login to iSCSI target

This Evaluation guide will not cover how to setup an iSCSI Target Server. It is assumed that the shared storage already exists in your environment. If you don't have shared storage and wish to configure it, a simple solution is to use OpenFiler (http://www.openfiler.com/), an Open Source storage management appliance, which can be run on physical hardware or as a virtual machine.

On both Primary and Secondary servers, perform the following functions:

1. If not already installed, ensure that the **iscsi-initiator-utils** rpm package is installed:

   ```
   # yum install iscsi-initiator-utils
   ```

2. Start the iscsid service and enable it to automatically start when the system boots

   ```
   # service iscsid start
   ```

   ```
   # chkconfig iscsid on
   ```

3. Configure the iscsi service to automatically start, which logs into iSCSI targets needed at system start up.

   ```
   # chkconfig iscsi on
   ```

4. Use the iscsiadm command to discover all available targets on the network storage server (OpenFiler)

   ```
   # iscsiadm -m discovery -t sendtargets -p <name or IP of iSCSI server>
   ```

**Example**
[root@LinuxPrimary init.d]# iscsiadm -m discovery -t sendtargets -p 192.168.198.201
**iqn.2006-01.com.openfiler:tsn.postgres**

5. Manually Login to the iSCSI Target

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.postgres –p 192.168.198.201 – login

6. Configure Automatic Login

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.postgres –p 192.168.198.201 —op update -n node.startup -v automatic

7. Use the “gdisk” command to format your iSCSI LUN, if needed

   # gdisk /dev/sdc

8. Create a filesystem on your new iSCSI LUN Partition, sdc1

   # mkfs.ext3 /dev/sdc1

9. Mount your iSCSI LUN at /var/lib/pgsql (assuming a default postgres configuration). If data already exists in this directory, make sure to move it into the shared iSCSI LUN

   # mount mount /dev/sdc1 /var/lib/pgsql

10. At this point you now have an iSCSI shared LUN, /dev/sdc1, mounted at /var/lib/pgsql. Our disk layout now look as follows (example):

**Example**
[root@LinuxPrimary pgsql]# df
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sda2 25967432 3683016 1976400 66% /
/dev/sda1 101086 24659 71208 26% /boot
tmpfs 517552 0 517552 0% /dev/shm
**/dev/sdc1 966644 38944 878596 5% /var/lib/pgsql**

# 12.7.5. Install, Configure, and Start PostgreSQL

## Primary Server

On your Primary server, perform the following actions:

1. Install both the “postgresql-server” and “postgresql” rpm packages if they do not exist on your system. Apply any required dependencies as well

   # yum install postgresql postgresql-server

2. Verify that your Shared iSCSI LUN is still mounted at /var/lib/pgsql via the “df” command

3. If this is a fresh PostgreSQL install, initialize a sample PostgreSQL database:

   # su – postgres

   # initdb —pgdata=/var/lib/pgsql/data

4. Ensure that all files in your PostgreSQL data directory (/var/lib/pgsql) have correct permissions and ownership

   # chown –R postgres:postgres /var/lib/pgsql

   # chmod 755 /var/lib/pgsql

5. Finally, manually start the PostgreSQL daemon from the command line. Note: **Do Not** start it via the “service” command, or the /etc/init.d/ scripts

   # su – postgres

   # pg_ctl start -D /var/lib/pgsql/data -l /var/lib/pgsql/pgstartup.log -o “-p 5432” -w

6. Verify PostgreSQL is running by connecting with the psql client (ensure you are still running as the “postgres” linux user):

   -bash-3.2$ psql

   Welcome to psql 8.1.22, the PostgreSQL interactive terminal.

   Type: \copyright for distribution terms

   \h for help with SQL commands

   \? for help with psql commands

```
\g or terminate with semicolon to execute query

\q to quit

postgres=# \q

-bash-3.2$
```

## Secondary Server

On your Secondary Server:

1. install both the “postgresql” and “postgresql-server” rpm packages if they do not exist on your system. Apply any required dependencies as well

   ```
   # yum install postgresql postgresql-server
   ```

2. Ensure that the PostgreSQL data directory (/var/lib/pgsql) has correct permissions and ownership

   ```
   # chown –R postgres:postgres /var/lib/pgsql

   # chmod 755 /var/lib/pgsql
   ```

3. There is no need to perform any of the additional steps taken on the Primary Server

# 12.7.6. Install SIOS Protection Suite for Linux – PostgreSQL

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

| SPS for Linux Component | Install Location |
|---|---|
| LifeKeeper Software | /opt/LifeKeeper |
| LifeKeeper Config File | /etc/default/LifeKeeper |

Perform the following actions on **both** Primary and Secondary server.

## Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.

2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After successful download you will have downloaded contents similar to the follow directory listing:

      [root@LinuxPrimary ~]# ls -l <directory> total 63680

      -rw-r—r— 1 root root 23163 May 30 14:03 EULA.pdf

      -rw-r—r— 1 root root 536 May 30 14:03 readme.txt

      -rw-r—r— 1 root root 65179648 May 30 14:03 sps.img

3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

## Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

   # mount -o loop sps.img /mnt

```
# cd /mnt

# ./setup
```

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

```
# cd /root

# umount /mnt
```

# Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

```
# /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic
```

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

```
# /opt/LifeKeeper/bin/lklicmgr

License File: 20101230.lic
```

| Product | Type | Expiry |
|---|---|---|
| LifeKeeper for Linux | Eval | 27 Mar 2013 (87 days) |

| | | |
|---|---|---|
| SIOS Data Replication ARK | Eval | 27 Mar 2013 (87 days) |
| PostgreSQL Recovery Kit | Eval | 27 Mar 2013 (87 days) |

# Start the SIOS Protection Suite for Linux

1. Start:

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

# 12.7.7. Configure the Cluster – PostgreSQL

## Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

> **Important** Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously.

### Access the LikeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 error.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

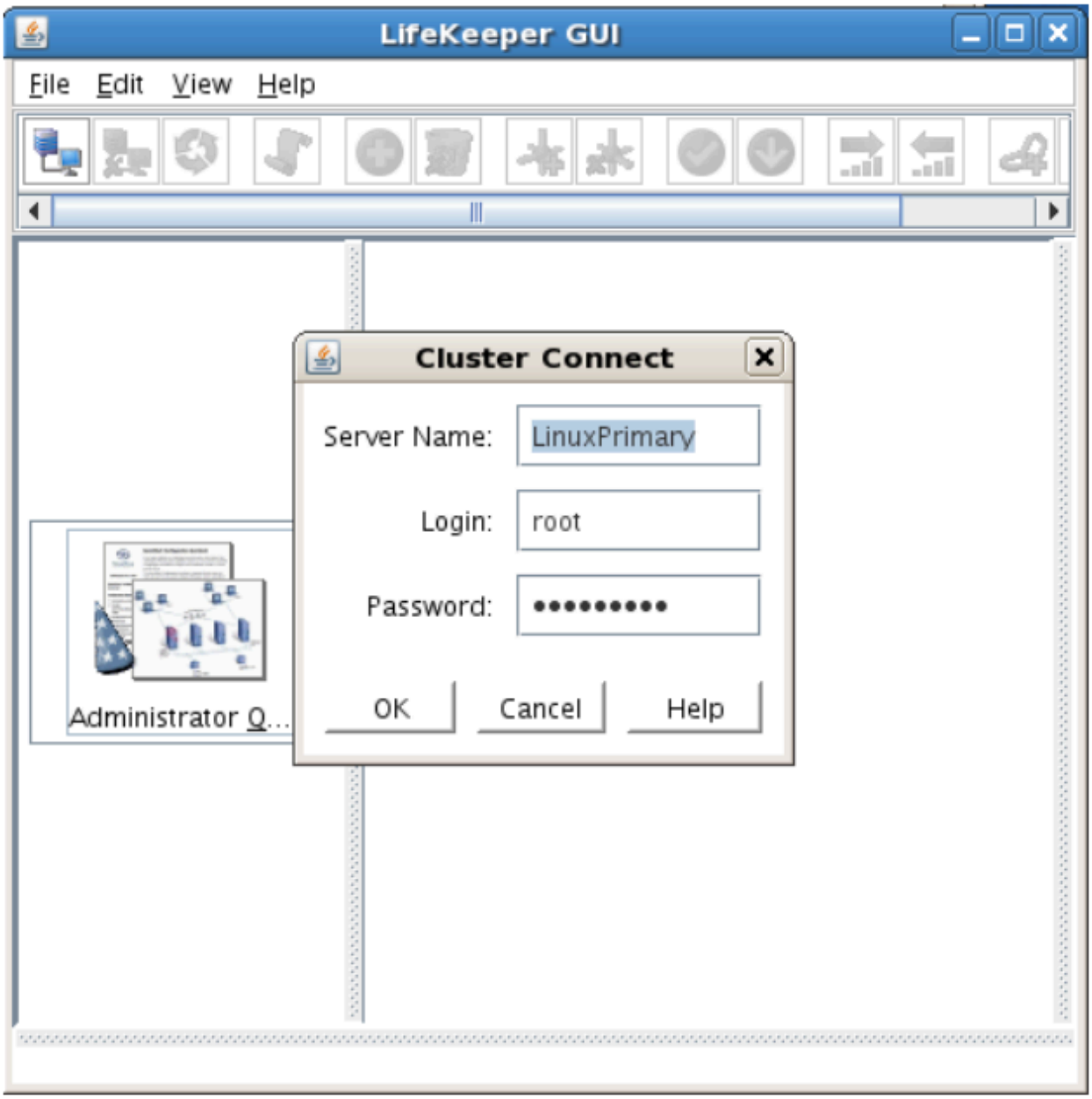
   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

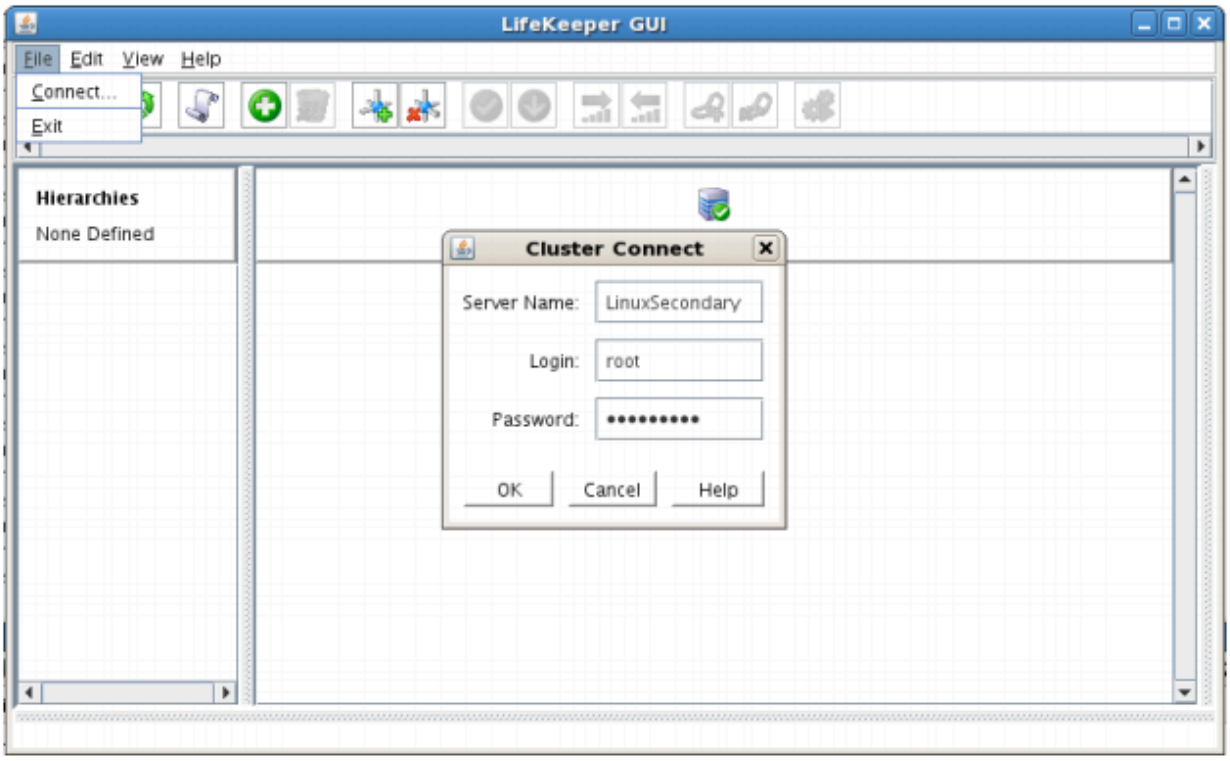
   a. /opt/LifeKeeper/bin/lkGUIapp &

3. Enter the name of the server you wish to connect to (this field will be populated with the name of the server you are on, if you are running the GUI from a server with LifeKeeper installed) along with your root credentials and click OK.



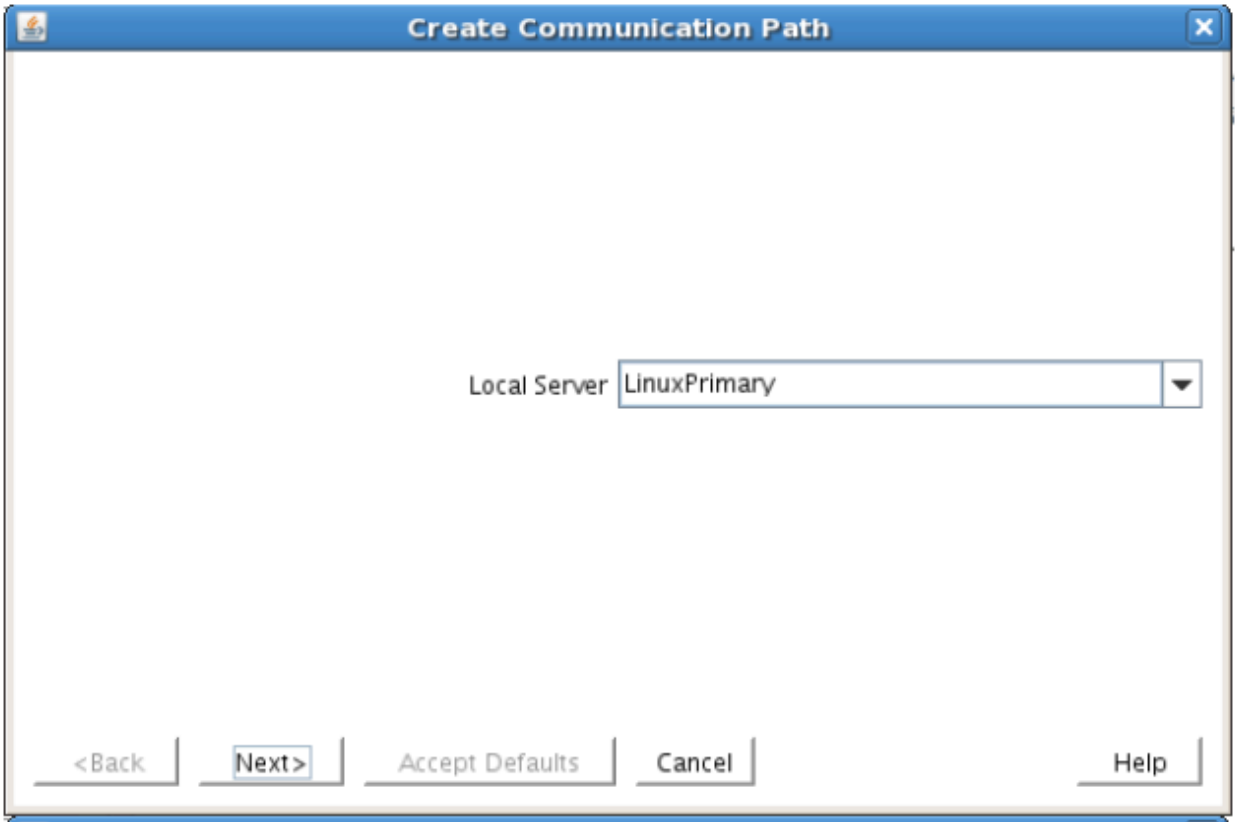

## Create Communication (Comm) Paths

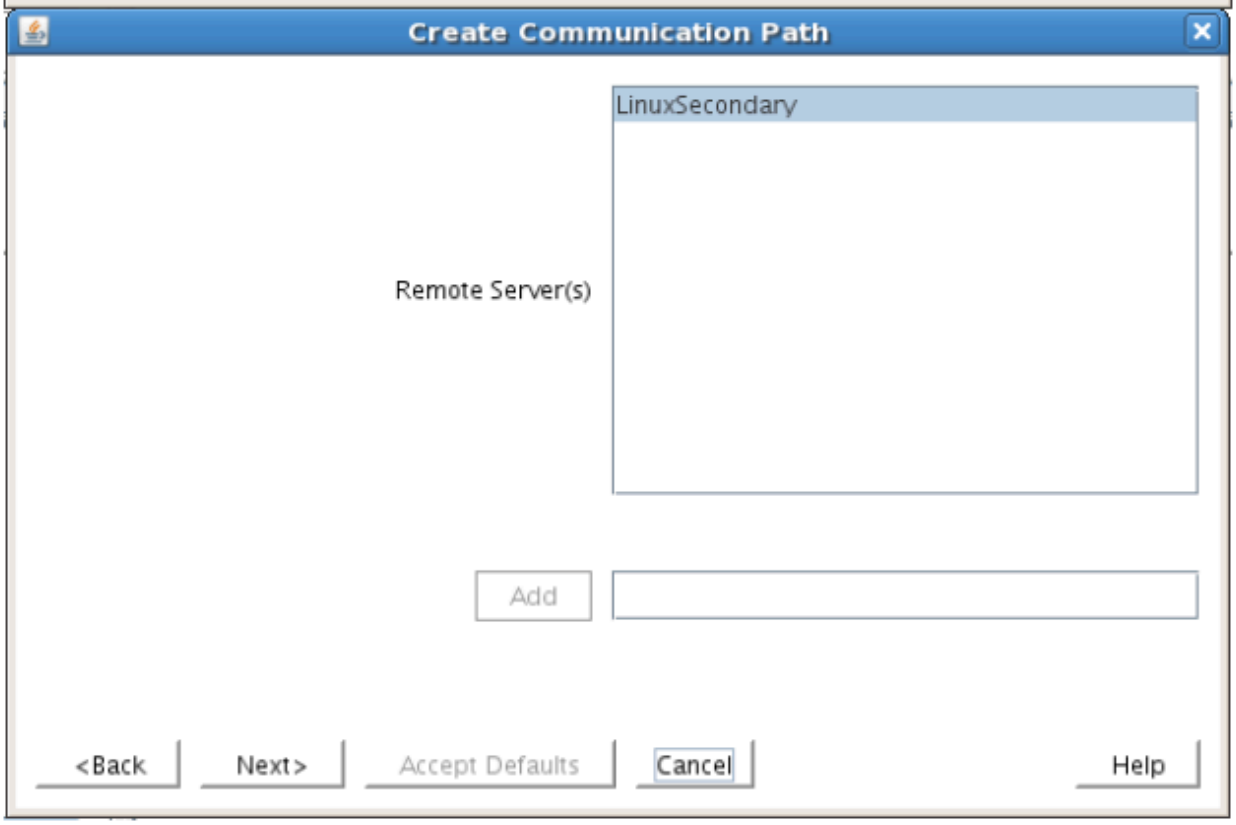
4. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



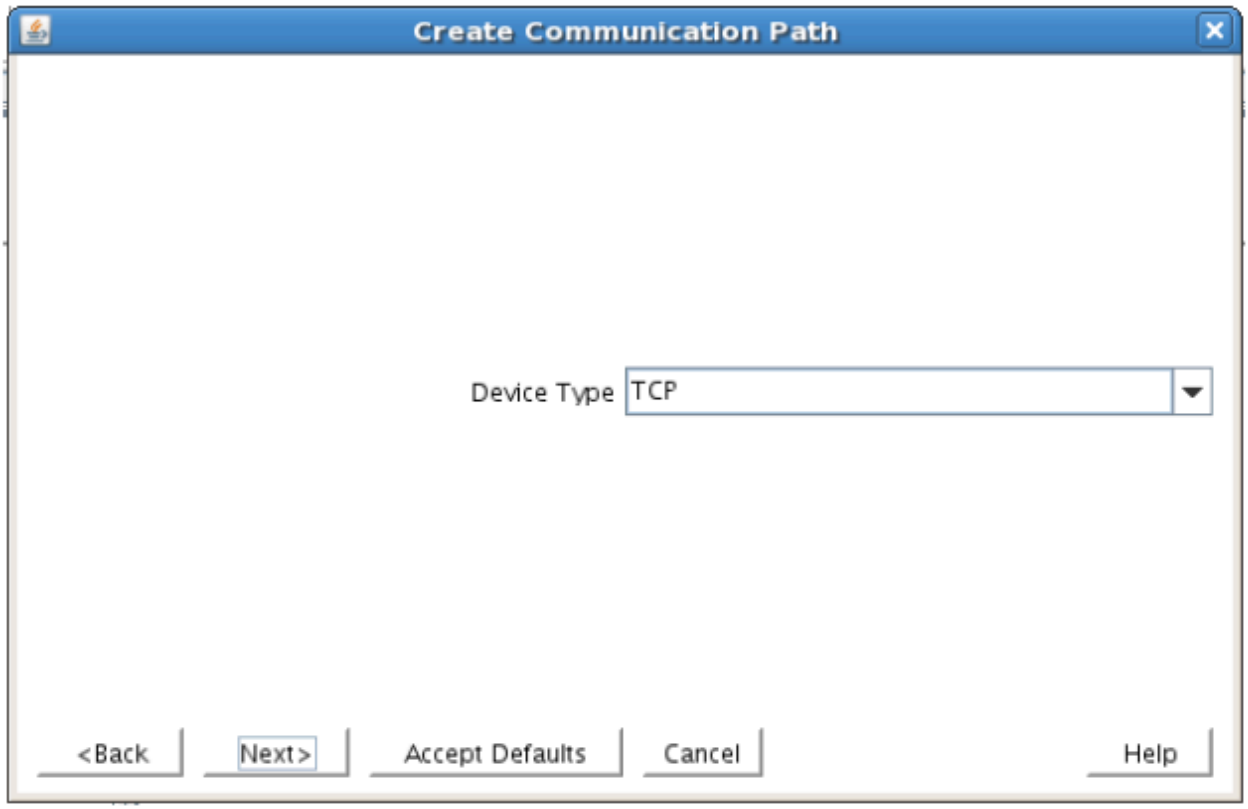

5. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

6. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

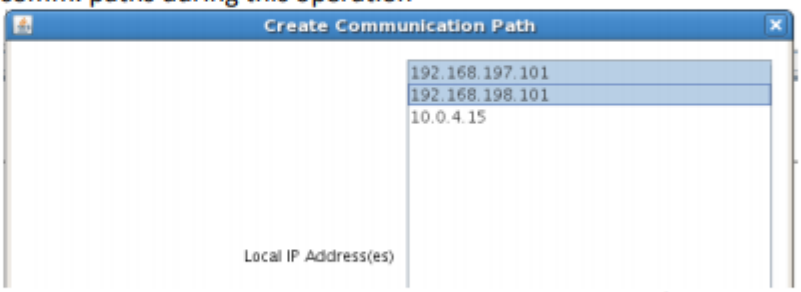


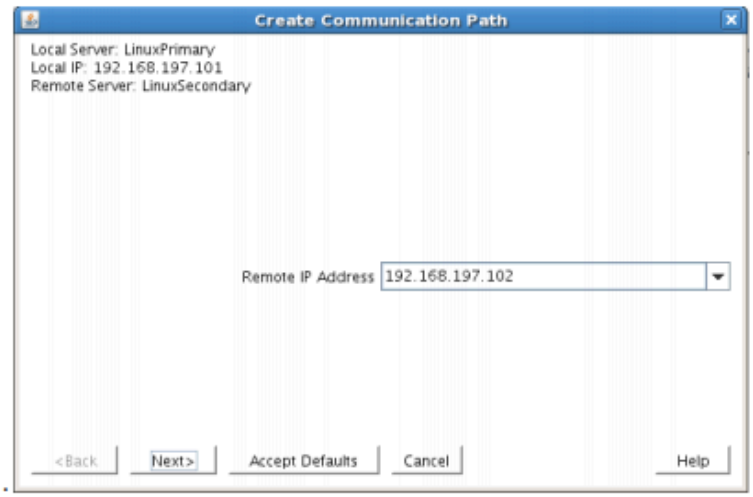




7. Select TCP for Device Type and Click Next.



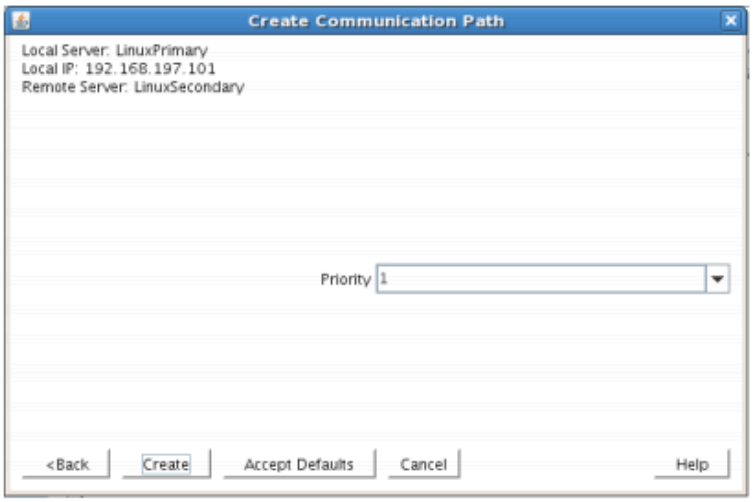

8. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

| Field | Tips |
|---|---|
| ***For TCP/IP Comm Path...*** | |
| Local IP Address | Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation   |
| Remote IP Address | Choose the IP address to be used by the remote server for this comm path |

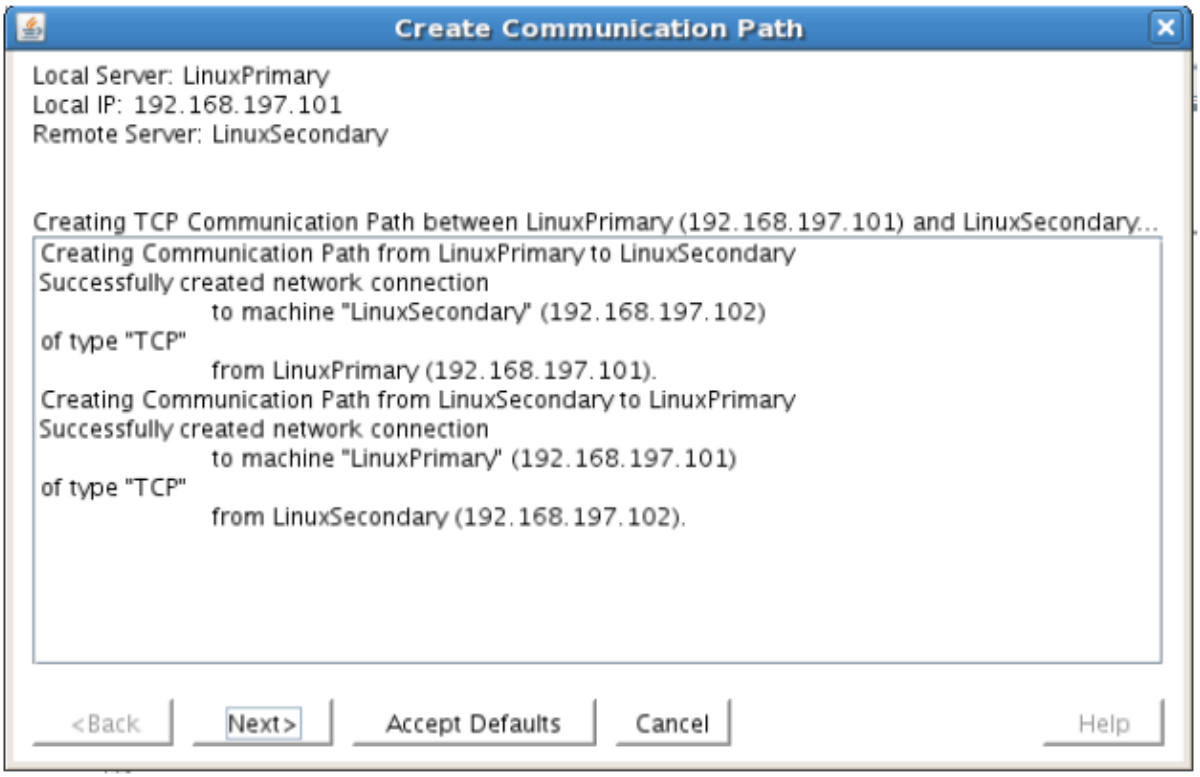



Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority




9. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



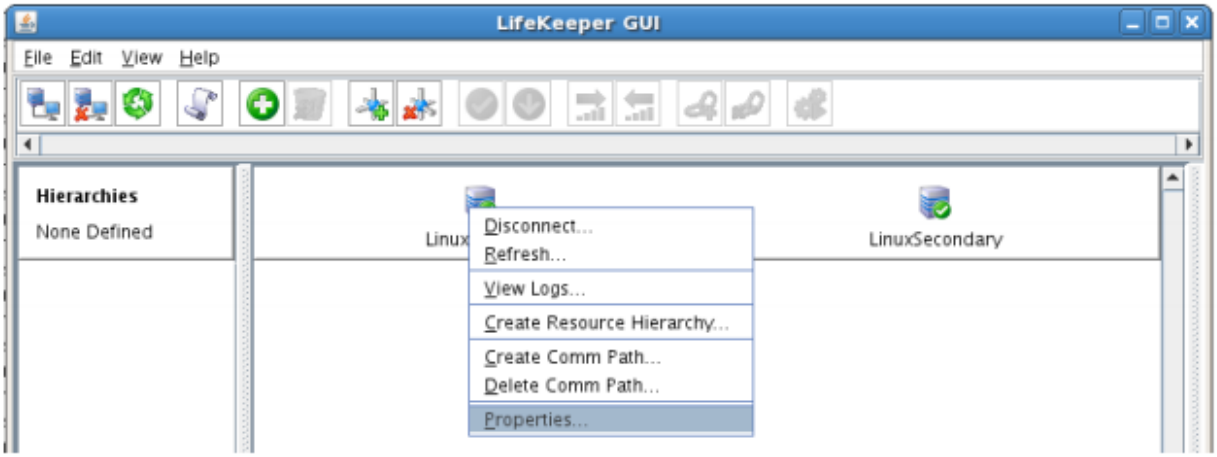

If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

10. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

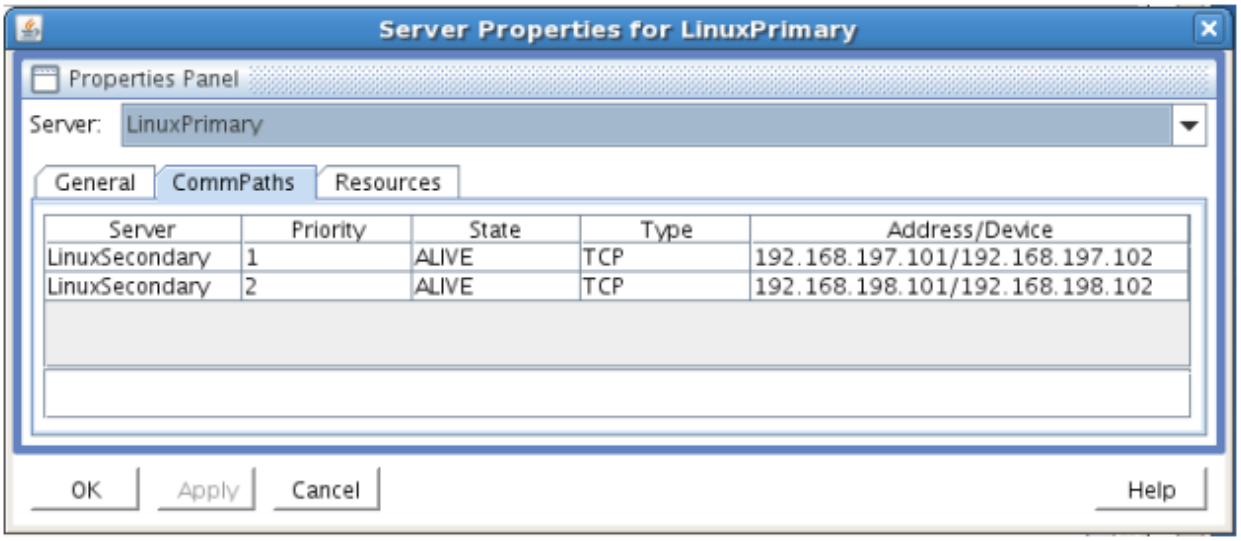
## Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.




2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.

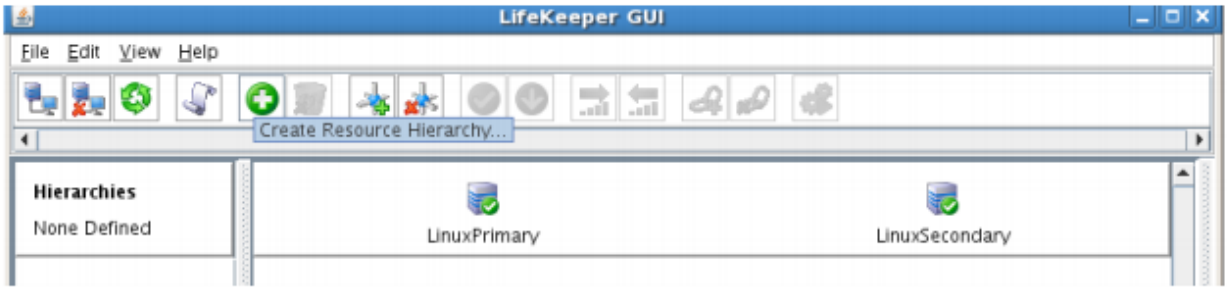



# Create the LifeKeeper Hierarchy

## Create and Extend an IP Resource

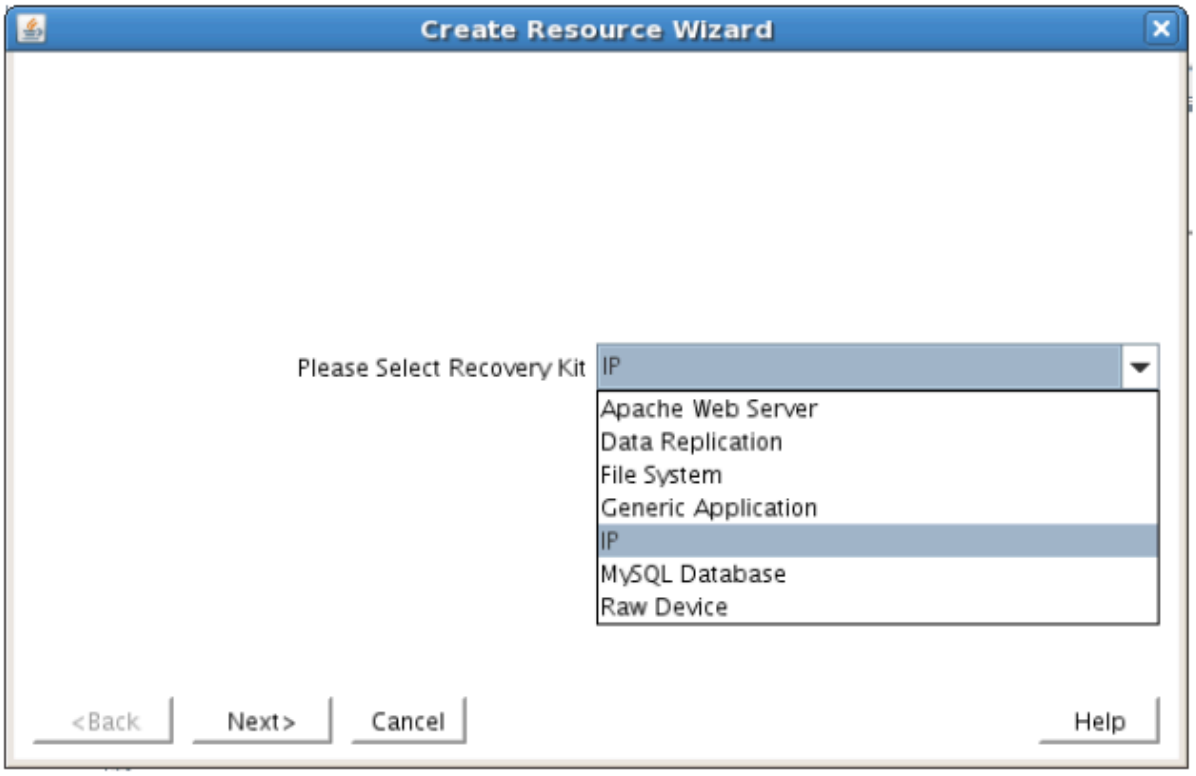
In LifeKeeper, create an IP resource and extend it to the secondary server by completing the following steps. This Virtual IP will have the ability to move between cluster nodes along the application that depends on it.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.




The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select IP Address and click Next.



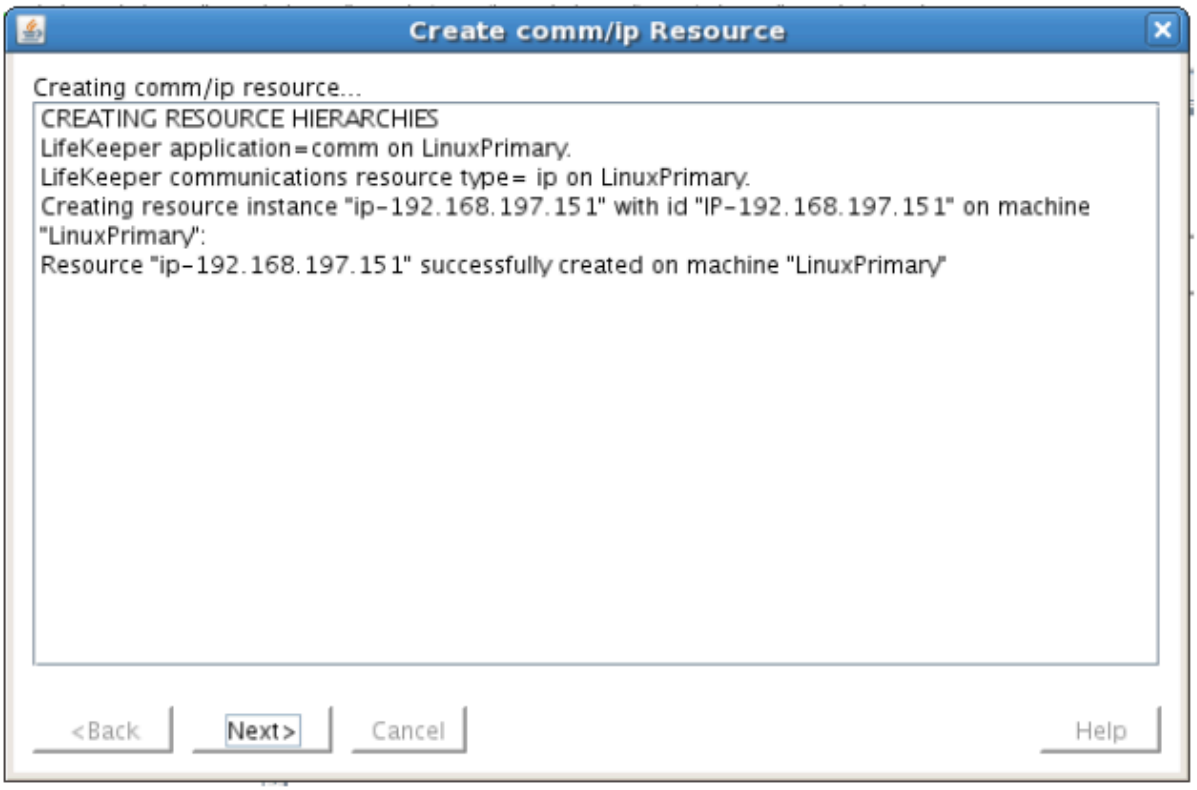

3. Enter the appropriate information for your configuration. The table below contains a list of the fields that display and additional information to assist you as you complete this procedure. Recommended values are also show below. You can also click the Help button for further information. Press Next to continue after entering the required information.

## IP Creation Field Definitions

| Field | Tips |
|---|---|
| Resource Type | Select IP Address as the resource type and click Next. |
| Switchback Type | Select Intelligent and click Next. |
| Server | Select the Server where the IP resource will be created. Select your<br><br>Primary server and click Next. |
| IP Resource | Enter the virtual IP information and click Next<br><br>Example **192.168.167.151**<br><br>**Note** This is an IP address that is not currently in use anywhere on your network. This is the address that all clients will use to connect to the protected resources. |

| | |
|---|---|
| Netmask | The IP subnet mask that your TCP/IP resource will use on the target server. Any standard netmask for the class of the specific TCP/IP resource address is valid.<br>In our sample configuration 255.255.255.0 is used for a subnet mask on both networks.<br><br>**Note:** The subnet mask you choose, combined with the IP address, determines the subnet that will be used by the TCP/IP resource and should be consistent with the network configuration. |
| Network Connection | This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. Select the correct NIC and click Next. |
| IP Resource Tag | Accept the default value and click Next. This value only affects how the IP is displayed in the GUI. The IP resource will be created on our Primary server. |

4. LifeKeeper will create and validate your resource. After receiving the message that the resource has been created successfully, click Next when the following dialog box appears so that you can complete the process of Extending the IP Resource to our Secondary server, below.



Extending the IP resource will start automatically after you have finished creating an IP address resource if you clicked Next in the dialog box displayed above. You can also start this from an existing IP address resource by right clicking on the active resource and selecting Extend Resource Hierarchy.

Refer to the table below to complete the Extend IP Resource procedure.

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Leave as “intelligent” and click Next |
| Template Priority | Leave as default (1) |
| Target Priority | Leave as default (10) |
| Network Interface | This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. The correct physical NIC should be selected by default. Please verify and then click Next |
| IP Resource Tag | Leave as default. |
| Target Restore Mode | Select Enable and click Next. |
| Target Local Recovery | Select Yes to enable Local Recovery for the SQL resource on the Target server. |
| Backup Priority | Accept the default value. |

5. After receiving the message Hierarchy extend operations completed, click Finish and then click Done

6. Your IP resource (192.168.197.151) is now fully protected and has the ability to “float” between cluster nodes as needed. Looking at the LifeKeeper GUI you will notice that the IP resource is Active on the Primary cluster node and Standby on the Secondary cluster node

## Create the Shared Filesystem Resource Hierarchy

Create a Filesystem resource to protect the shared iSCSI filesystem and make it high available between

cluster nodes. LifeKeeper leverages SCSI Persistent Group Reservations (PGR) to lock the LUN, ensuring that only the active cluster node for the storage resource can access it.

> **Important** At this point, the shared iSCSI LUN needs to already be mounted on the Primary Server. It should NOT be mounted on the Secondary Server. See section titled “Configure iSCSI initiator, discover and login to iSCSI target” above to review the steps involved.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select File System and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node) |
| Mount Point | Select /var/lib/pgsql . Note that LifeKeeper scans the system for LUNS that are sharable between cluster nodes. The list of possible shared LUNS is presented automatically in this step of the wizard. |

4. Select Create Instance to define this resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the File System resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Your resource hierarchy should look as follows:

## Create the PostgreSQLSQL Resource Hierarchy

Create a PostgreSQL resource to protect the PostgreSQL database and make it high available between cluster nodes.

> **Important** At this point, PostgreSQL needs to be already running on the Primary Server. It should NOT be running on the Secondary Server. See section titled “Install, Configure, and Start PostgreSQL” above to review the process to configure and start PostgreSQL as needed.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select PostSQL Database and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node) |
| PostgreSQL Executable | Leave as default (/usr/bin) since we are using a standard PostgreSQL |
| Location | install/configuration in this example. This field is used to specify the directory path containing the PostgreSQL executables. |
| PostgreSQL Client | Leave as default (/usr/bin/psql) . This field is used to specify the directory |
| Executable Location | path containing the PostgreSQL executable psql. |
| PostgreSQL Administration | Leave as default (/usr/bin/pg_ctl). This field is used to specify the |
| Executable Location | directory path containing the PostgreSQL executable pg_ctl. |
| PostgreSQL Data Directory | /var/lib/pgsql/data. This field is used to specify the location of the PostgreSQL data directory (datadir) that will be placed under LifeKeeper protection. The specified directory must exist and reside on a shared or replicated file system. |
| PostgreSQL Port 5432. | This field is used to specify the TCP/IP port number on which the postmaster daemon is listening for connections from client applications |
| PostgreSQL Socket Path | Leave as default (/tmp/.s.PGSQL.5432 ) . This field is used to specify the full path to the Unix- domain socket on which the postmaster daemon is listening for connections from client applications. |
| Enter Database Administrator User | Enter “postgres” . This field is used to specify a PostgreSQL Database Administrator User name for the specified database instance with connection and administrator privileges for the instance |
| PostgreSQL Logfile | Leave as default (/tmp/pgsql-5432.lk.log) . This field is used to specify the log file path that will be used for the PostgreSQL log file. |
| Database tag | Leave as default |

4. Select Create to define the PostgreSQL resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the PostgreSQL resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Note: LifeKeeper will automatically identify that the PostgreSQL resource has a dependency on the FileSystem resource (/var/lib/pgsql). The Filesystem Resource will appear underneath the PostgreSQL resource in the GUI

9. Your resource hierarchy should look as follows:

## Create the PostgreSQL IP Address Dependency

In this step will define an additional dependency: that PostgreSQL depends on a Virtual IP (192.168.197.151) so that the IP address follows the PostgresSQL database should it move.

1. From the LifeKeeper GUI toolbar, right-click on the “pgsql-5432” resource
2. Select “Create Dependency” from the right-click context menu
3. In the Child Resource Tag dropdown menu, select “ip-192.168.197.151”
4. Click Next
5. Click Create Dependency
6. Click Done
7. The Virtual IP address resource (192.168.197.151) will now appear underneath the PostgreSQL resource in the LifeKeeper user interface. This ensures that resources move together, and are started/stopped in the proper order.
8. Your resource hierarchy should look as follows

At this point in the Evaluation, we have fully protected PostgreSQL, and its dependent resources: IP addresses, and Shared Storage.

# 12.7.8. Test Your Environment – PostgreSQL

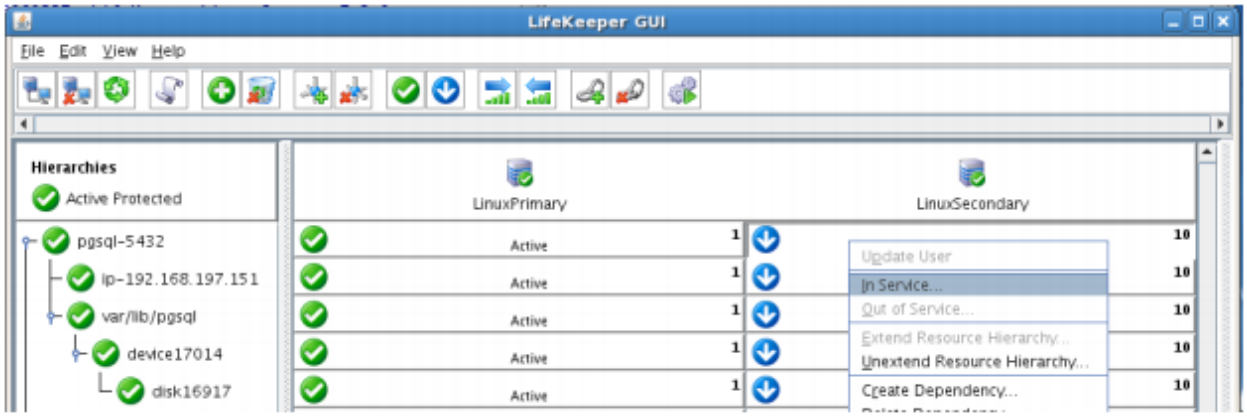
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

Note: For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

## Manual Switchover of the PostgreSQL Hierarchy to Secondary Server

**Procedure:**

- From the LifeKeeper GUI, right click on the PostgreSQL resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up




**Expected Result:**

- Beginning with the PostgreSQL resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXSECONDARY

**Tests/Verification:**

- Using the LifeKeeper GUI, verify that the PostgreSQL and dependent resources are active on LINUXSECONDARY.
- Run “ifconfig –a” on LINUXSECONDARY to validate that the IP Address 192.168.197.151 is active on LINUXSECONDARY
- Run “df –h” to verify that the /var/lib/pgsql shared iSCSI filesystem is mounted on LINUXSECONDARY
- Verify the PostgresSQL services are running on LINUXSECONDARY by running “ps –ef | grep –i

postgres”
- On LINUXSECONDARY run the following command to verify client connectivity to the PostgreSQL database:
    - # su – postgres
    - # psql
- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXPRIMARY, run “mount /dev/sdc1 /var/lib/pgsql”. This should FAIL because LINUXPRIMARY does not own the SCSI reservation on this LUN.

# Manual Switchover of the PostgreSQL Hierarchy back to Primary Server

**Procedure:**

- From the LifeKeeper GUI, right click on the PostgreSQL resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

**Expected Result**

- Beginning with the PostgreSQL resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXPRIMARY

**Tests/Verification:**

- Using the LifeKeeper GUI, verify that the PostgreSQL and dependent resources are active on LINUXPRIMARY.
- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.151 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/lib/pgsql shared iSCSI filesystem is mounted on LINUXPRIMARY
- Verify the PostgreSQL services are running on LINUXPRIMARY by running “ps –ef | grep –i postgres”
- On LINUXPRIMARY run the following command to verify client connectivity to the PostgreSQL database:
    - # su – postgres

    - #35; psql

- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXSECONDARY, run “mount /dev/sdc1 /var/lib/pgsql”. This should FAIL because LINUXSECONDARY does not own the SCSI reservation on this LUN.

## Simulate a network failure on the Primary Server by failing the IP resource

> **IMPORTANT NOTE:** Only perform this test if you have more than one communications path configured.

If you perform this test and have only one communications path configured, your system will enter a split-brain scenario as described in the LifeKeeper Administration Guide found here. Refer to this document for more information or contact SIOS presales technical support for assistance in resolving this condition.

**Procedure**

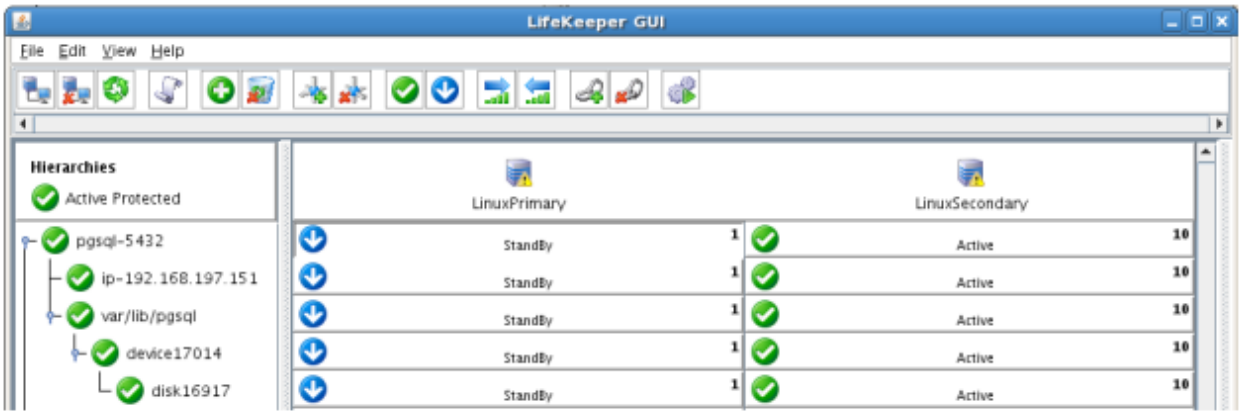
- On LINUXPRIMARY, pull the network cable attached to the NIC that the virtual IP address is configured on

**Expected Result:**

- The IP Resource should fail first.
- The entire hierarchy should failover to LINUXSECONDARY

**Tests/Verification:**

- Check the LifeKeeper Log to verify the IP resource failed – “/opt/LifeKeeper/bin/lk_log log”
- Using the LifeKeeper GUI, verify the PostgreSQL resource hierarchy fails over successfully to LINUXSECONDARY
- After this test has been completed, re-connect the network cable on LINUXPRIMARY



# Hard failover of the resource from the Secondary Server back to the Primary Server

**Procedure:**

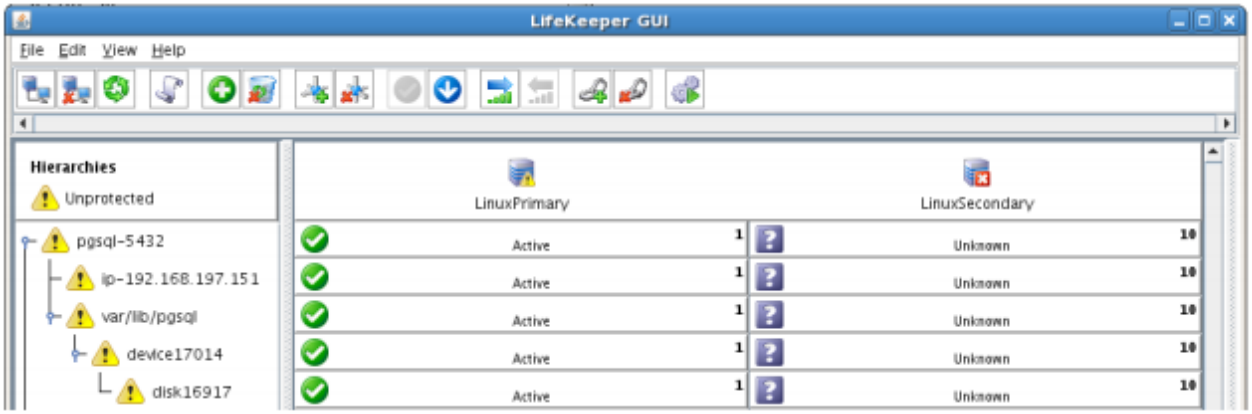
- Pull the power cord on LINUXSECONDARY, as this is the server with all resources currently In Service.

**Expected Result:**

- After failure has been detected, beginning with the dependent resources (IP and Volume), all resources will be brought in service on LINUXPRIMARY.

**Tests/Verification:**

- Using the LifeKeeper GUI, verify the mirror has reversed and is in a Resync Pending state waiting for LINUXSECONDARY to come back on line.
- Verify the PostgreSQL Server services are running on LINUXPRIMARY.




# Bring Failed Server back on line

**Procedure:**

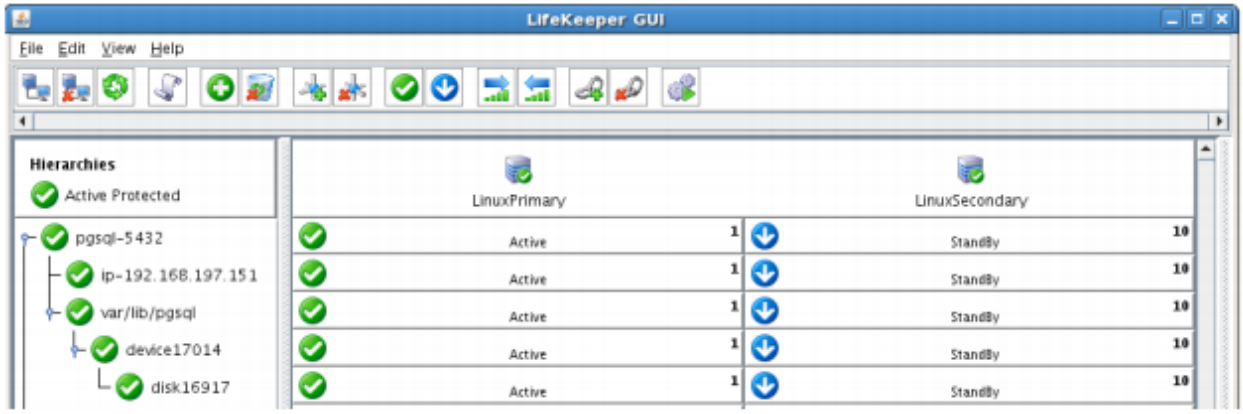
- Plug the power cord back into LINUXSECONDARY and boot it up.

**Expected Result:**

- Using the LifeKeeper GUI, verify that LINUXSECONDARY is coming back up and has become the Standby Server.

**Tests/Verification:**

- Verify the PostgreSQL Hierarchy is in service on LINUXPRIMARY and standby on LINUXSECONDARY.




# Verify Local Recovery of PostgreSQL Server

**Procedure:**

- Kill the PostgreSQL processes via the command line:
- # ps –ef | grep postgres
- # (kill -9 the PIDs returned)
- run “ps –ef | grep postgres once again to verify that the processes no longer exist

**Expected Result: (Assumes Local Recovery for SQL resource is set to YES)**

- The PostgreSQL Server service should stop.
- The PostgreSQL quickcheck process will automatically restart the PostgreSQL Server Service when it runs periodically.
- No failure of PostgreSQL should occur.

**Tests/Verification:**

- Execute “ps –ef | grep postgres” once again to verify that the postgresql processes have been restored locally on LINUXPRIMARY.
- Verify connectivity to the postgreSQL database by running:

  ↑
  - # su – postgres
  - # psql

- If you inspect the LifeKeeper logs, you will see information indicating that LifeKeeper dectected the failure of the PostgreSQL service and recovered it locally. Run /opt/LifeKeeper/bin/lk_log log for more information.

# 12.8. Apache/MySQL Cluster Using Both Shared and Replicated Storage

## Objective

This document is intended to aid you in installing, configuring and using the SIOS Protection Suite for Linux evaluation product to make Apache and MySQL highly available. If Apache and MySQL are not already installed, please allocate some time to install it on your servers. Once this task has been completed, you may install and configure SIOS Protection Suite for Linux.

There are five phases in this process:

- Phase 1 – Prepare to Install
- Phase 2 – Configure Storage
- Phase 3 – Install and Configure Apache/PHP
- Phase 4 – Install and Configure MySQL
- Phase 5 – Install SIOS Protection Suite for Linux
- Phase 6 – Configure your LifeKeeper Cluster
- Phase 7 – Test Your Environment

# 12.8.1. Terms to Know – Apache

The following terms are used throughout this document and while some may be familiar to you, it may be helpful to review how SIOS defines and uses these terms.

## Network Communication Terms

**Crossover cable** – A cable used to directly connect computing devices together instead of being connected to a network switch, hub or router. This cable creates an isolated, private network to allow cluster-related and data replication traffic to flow between systems.

## Types of LifeKeeper Servers

**Server** – A computer system dedicated to running software application programs.

**Active Server** – This is the server where the resource hierarchy is currently running (IN SERVICE).

**Standby Server** – This is the server where the resource hierarchy is defined, but is not currently running. This server is available to bring the resource hierarchy into service should something happen to the resource hierarchy on the Active Server.

**Primary Server** – This is the server in a LifeKeeper configuration with the highest priority for a given resource hierarchy. It is the server that provides services for the resource hierarchy under normal circumstances.

**Secondary Server** – This is the server in a LifeKeeper configuration with the 2nd highest priority for a given resource hierarchy.

**Source Server** – This is the server in a LifeKeeper cluster that is using data replication (Active Server). It is where the resource hierarchy is currently running and the replicated partition (Source Partition) is accessible for writes.

**Target Server** – This is the server in a LifeKeeper cluster using data replication (Standby Server). The replicated partition (Target Partition) is updated with writes from the Source Partition by the SIOS Data Replication system. This partition should not be accessed/modified manually.

## SIOS Data Replication Terms

**Replication** – Transferring data from one partition to another via a sector-by-sector copy. During replication, the target partition should not be accessed or modified assuring your data integrity.

**Synchronous** – A replication scheme in which the data is confirmed written and valid on the target before the write operation occurs on the source disk through a series of information exchanges. Synchronous mirrors should only be implemented on high speed (100Mbps+) networks due to the network overhead involved.

**Asynchronous** – A replication scheme in which the data is released for writing on the source immediately and is sent to the target(s) simultaneously for writing as fast as the data can get there andcan be written on them.

**Rate of Change** – A measure of the amount of data which is changing over a set period of time.

**Compression** – An algorithm which is optionally implemented to reduce the amount of traffic between source and target nodes. Nine levels of compression are offered. Compression is turned off by default.

**Throttling** – An optionally implemented mechanism to limit the bandwidth used for replication.

## LifeKeeper Product Terms

**Communications Path** – A mechanism supporting communication between nodes in a LifeKeeper cluster. SIOS highly recommends implementing multiple communication paths between all servers in the cluster to eliminate a single point of failure.

**Heartbeat** – A periodic message exchanged between nodes in a LifeKeeper cluster that provides server health monitoring. A heartbeat message is one type of inter-node cluster communication sent over a communications path.

**Split Brain** – A situation in which all communications paths between cluster members fail, but all servers remain up and running. In this situation, both systems believe the other has failed and both believe they should keep or bring resources into service.

**Failover** – The unplanned migration of a resource hierarchy to the Standby Server because of a system or resource failure on the Active Server.

**Switchover** – The planned migration of a resource hierarchy from the Active Server to the Standby Server.

**Switchback** – The setting that governs the recovery behavior of the server where the resource was in service when it failed. If the setting is “intelligent”, the server acts as a possible backup for the given resource. If the setting is “Automatic”, the server actively attempts to re-acquire the resource without further notice.

**Resource** – A system asset that can be protected by LifeKeeper. Resources can be used to represent disk partitions, virtual IP addresses, applications, etc.

**Extend a Resource** – Create or define an already configured LifeKeeper resource onto another server in the cluster and build an equivalency relationship that prevents the resource from coming in service on both systems simultaneously.

**Resource Hierarchy** – A grouping of resources, in a predetermined order, from high to low. This may also be referred to as simply a Hierarchy.

**Shared Storage** – One or more logical disk partitions that are physically attached to all nodes in a cluster. LifeKeeper ensures that the volume is only accessible by one server at a time. This is formally

called I/O fencing.

**Data Replication (Disk Mirroring)** – The replication of logical disk partitions to separate physical hard disks in real time to ensure continuous availability, currency and accuracy of data.

**Source** – The partition on the source server used for replication. The “gold” copy of the data.

**Target** – The partition on the target server used for replication.

**Switchable IP Address** – A unique IP address that may be moved between systems in the cluster. Client systems connect to this address and the system where the virtual IP resource is active will respond to requests.

# 12.8.2. The Evaluation Process – Apache

SIOS strongly recommends performing your evaluation of SIOS Protection Suite for Linux in a test lab environment. SIOS is not responsible and cannot provide support for evaluation software installed in a production environment.

All questions during the evaluation period should be directed to evalsupport@us.sios.com or your local Pre-Sales Engineering contact. Pre-sales support will contact you by the next business day to answer questions. Once you are a licensed customer on software maintenance, you will have access to 24 × 7 post-sales technical support.

**Important** Your evaluation license is valid for a limited period of time from the day you receive the SIOS product evaluation package and licenses via email from the SIOS sales team.

# 12.8.3. Prepare to Install – Apache

## Hardware Requirements

### Primary and Secondary Servers

- Systems must meet the minimum requirements for the Linux distribution to be used during the evaluation.
- 512MB RAM minimum; 1GB RAM recommended.
- 2GB of available hard disk space recommended.
- Multiple Network Interface Cards (NIC's) are recommended.
- Configure one or more additional disk partitions to be used for data replication. On the primary server, these will become the source partitions. On the secondary server(s), these will become the target partitions. In this evaluation example we will be replicating MySQL data (which will be a partition mounted at /var/lib/mysql)
- For replicated partitions, a target partition's size must equal to or larger than the size of its source partition.
- The system ( / ) and boot (/boot) partitions are not eligible for replication.

**Note:** You may use more than one partition for replicated data, allowing for separation of multiple database files and/or log directories for performance reasons.

### Client

This system is not required but is recommended for testing the cluster environment.

- A standard linux terminal running the MySQL client can be used to test the configuration.

## Software Requirements

### Primary Server and Secondary Server

- Linux Distribution x86_64, AMD 64:
    - RedHat Enterprise Linux 5 (5.4+ recommended) or 6.x
    - CentOS Linux 5 (5.4+ recommended) or 6.x
    - Oracle Enterprise Linux 5 (5.4+ recommended), 6.3, 6.4
        - RedHat Compatibility Kernel Only
    - SuSE Linux Enterprise Server 10 or 11 (11 recommended)
    - See Linux Release Notes for a full list of supported Operating Systems

- Current patches / security updates are recommended.
- Satisfied dependencies; especially if the Linux installation package selection was base/minimal you will need to refer to the dependencies documentation at Linux Dependencies
- Its recommended that IPtables is disabled
    - # /etc/init.d/iptables off
    - # chkconfig iptables off
    - See here for information regarding the ports SIOS Protection Suite for Linux uses.

- Disable SELinux :
    - Edit /etc/selinux/config
    - Set SELINUX=disabled (note: permissive mode is also acceptable)

- Check the configuration of your /etc/hosts file
    - localhost.localdomain and localhost are the only entries that can be on 127.0.0.1
    - Create a separate entry for your hostname with a static address

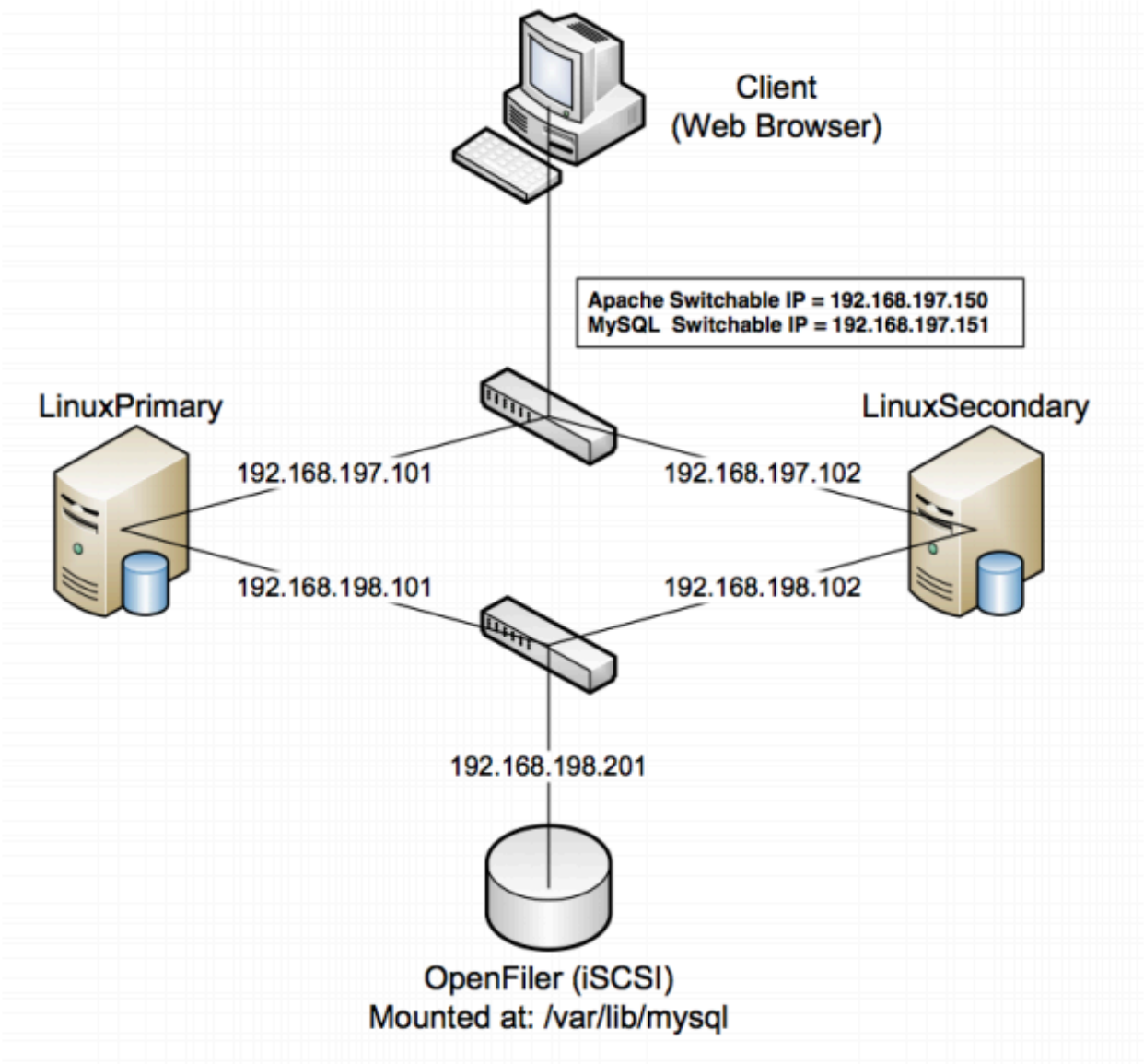
- GUI Authentication with PAM
    - SPS for Linux now leverages the Pluggable Authentication Module (PAM) provided in the Linux Standard Base (LSB).
        - Users are identified and authenticated against the system's PAM configuration. Privilege levels are determined from group membership as provided through PAM.
        - In order to access the GUI, a user must be a member in one of the three LifeKeeper groups: lkadmin, lkoper or lkguest.
        - See the following URL for more information on this topic:
            - Configuring GUI Users

# Network Requirements

For your evaluation, we recommend configuring your machines similarly to the following example.

LinuxPrimary and LinuxSecondary are multi-homed, between two LAN segments (the second NIC in each server could even be connected via a cross-over cable if a second physical network is not available). The second NIC is optional in this configuration, but highly recommended in production environments to avoid a single point of failure.

In this example evaluation scenario we will be leveraging both local, replicated storage (with the Apache configuration) as well as Shared Storage (iSCSI, for the MySQL configuration). OpenFiler is a storage appliance server that will serve an iSCSI target to LinuxPrimary and LinuxSecondary.




**Network Configuration Example**

## Primary Server and Secondary Servers

- Configure the Host file with entries for all LifeKeeper protected servers. This is typically /etc/hosts.

  Example:

  192.168.197.101 LinuxPrimary

  192.168.197.102 LinuxSecondary

- See your Network Administrator to obtain an unused IP Address to be used as the switchable IP Address. This switchable IP Address will be created later in the configuration process.
- Public Network connection(s) configured with:
    - Static IP address
    - Correct subnet mask
    - Correct gateway address
    - Correct DNS server address(es)
- Private Network connection(s) configured with:
    - Static IP address (on a different subnet from the public network)
    - Correct network mask
    - No gateway IP address
    - No DNS server addresses

## Client

Must be able to communicate on the same subnet/network as the servers Public interface addresses. In our example, this is the 192.168.197.0/24 network.

# 12.8.4. Configure Storage – Apache

## Before you Begin

Ensure the following:

- You have an extra disk/partition on both servers that can be used for data replication. A target volume's size must equal to or larger than the size of its source disk/partition.
- Shared storage is available. This can either be Fiber Channel SAN, iSCSI, NAS, etc. In this example we will review configuration of an iSCSI target for use as our MySQL database storage repository.

## Partition local storage for use with SIOS DataKeeper for Linux

### Primary Server

On your Primary server, perform the following actions:

1. Identify an existing free, unused disk partition to use as our Apache repository. Alternatively, create a new partition. Use the "gdisk" utility to partition your disk appropriately. In this example /dev/sdb is an unused disk where we will create a single partition

    a. gdisk /dev/sdb

    b. Press "n" to create a new partition

    c. Press "p" to create a primary partition

    d. This example uses a new disk, so we will use all default values (Partition 1, entire disk) Hit Enter twice to confirm these parameters

    e. Press "w" to write the partition table and exit gdisks

**Example**

[root@LinuxPrimary ~]# **gdisk /dev/sdb**

Command (m for help): **n**

Command action

e extended

p primary partition (1-4)

**p**

Partition number (1-4): **1**

First cylinder (1-256, default 1): **<enter>**

Using default value 1

Last cylinder or +size or +sizeM or +sizeK (1-256, default 256): **<enter>**

Using default value 256

Command (m for help): **w**

The partition table has been altered! Calling ioctl() to re-read partition table.

Syncing disks.

[root@LinuxPrimary ~]#

[root@LinuxPrimary ~]# df /var/www

Filesystem 1K-blocks Used Available Use% Mounted on

/dev/sdb1 253855 11083 229666 5% /var/www

[root@LinuxPrimary ~]# ls /var/www

cgi-bin error html icons lost+found manual usage

2. Format the newly created disk partition

   # mkfs.ext3 /dev/sdb1

3. Mount the partition temporarily at /mnt

   # mount /dev/sdb1 /mnt

4. Move any existing data from /var/www/ into this new disk partition (assumes a default apache configuration)

   # cd /var/www

   # mv * /mnt

5. Remount /dev/sdb1 at /var/www

   # cd /root

```
# umount /mnt

# mount /dev/sdb1 /var/www
```

6. Note: there is no need to add this partition to /etc/fstab. Lifekeeper will take care of mounting this automatically.

**Result**

```
[root@LinuxPrimary ~]# df /var/www
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sdb1 253855 11083 229666 5% /var/www
[root@LinuxPrimary ~]# ls /var/www
cgi-bin error html icons lost+found manual usage
```

### Secondary Server

7. On your Secondary server, only perform Step #1 above, where you partition the disk. The size of the Target volume needs to be the same size, or greater, than our Source volume.

## Configure iSCSI initiator, discover and login to iSCSI target

This Evaluation guide will not cover how to setup an iSCSI Target Server. It is assumed that the shared storage already exists in your environment. If you don't have shared storage and wish to configure it, a simple solution is to use OpenFiler (http://www.openfiler.com/), an Open Source storage management appliance, which can be run on physical hardware or as a virtual machine.

On both Primary and Secondary servers, perform the following functions:

1. If not already installed, ensure that the **iscsi-initiator-utils** rpm package is installed:

```
# yum install iscsi-initiator-utils
```

2. Start the iscsid service and enable it to automatically start when the system boots

```
# service iscsid start

# chkconfig iscsid on
```

3. Configure the iscsi service to automatically start, which logs into iSCSI targets needed at system start up.

```
# chkconfig iscsi on
```

4. Use the iscsiadm command to discover all available targets on the network storage server (OpenFiler)

```
# iscsiadm -m discovery -t sendtargets -p <name or IP of iSCSI server>
```

**Example**
[root@LinuxPrimary init.d]# iscsiadm -m discovery -t sendtargets -p 192.168.198.201
**iqn.2006-01.com.openfiler:tsn.mysql**

5. Manually Login to the iSCSI Target

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.mysql –p 192.168.198.201 –login

6. Configure Automatic Login

   # iscsiadm -m node –T iqn.2006-01.com.openfiler:tsn.mysql –p 192.168.198.201 —op update -n node.startup -v automatic

7. Use the “gdisk” command to format your iSCSI LUN, if needed

   # gdisk /dev/sdc

8. Create a filesystem on your new iSCSI LUN Partition, sdc1

   # mkfs.ext3 /dev/sdc1

9. Mount your iSCSI LUN at /var/lib/mysql (assuming a default mysql configuration). If data already exists in this directory, make sure to move it into the shared iSCSI LUN

   # mount /dev/sdc1 /var/lib/mysql

10. At this point you now have a local partition, /dev/sdb1 mounted at /var/www and an iSCSI shared LUN, /dev/sdc1, mounted at /var/lib/mysql. Our disk layout now look as follows (example):

**Example**
[root@LinuxPrimary mysql]# df
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sda2 25967432 3683016 1976400 66% /
/dev/sda1 101086 24659 71208 26% /boot
tmpfs 517552 0 517552 0% /dev/shm
**/dev/sdb1 253855 11132 229617 5% /var/www**
**/dev/sdc1 966644 38944 878596 5% /var/lib/mysql**

# 12.8.5. Install and Configure Apache and PHP

## Before you Begin

Ensure the following:

1. If you are not familiar with installing and configuring Apache, please refer to the Apache documentation.
2. Apache should not be running at the time you attempt to protect it with LifeKeeper.

> **Important:** If Apache and PHP are already installed, skip to step #2 in the section below. You will need to verify the Apache configuration and/or relocate the location of the website data.

## Primary Server

1. Install Apache.

   a. This example assumes that you are running a RHEL or CentOS 5.X based Linux distribution. For other Linux distros, please refer to your apache documentation for syntax differences.

   b. If the Apache and PHP packages are not already installed, from the command line, , run: “yum install httpd php”

   c. A number of dependencies will most likely be discovered. Install those as well.

2. Apache should be configured so that it will not automatically start when the server boots. LifeKeeper will control the start/stop of the webserver once its protected

   a. Check the status of the webserver: “/etc/init.d/httpd status

   b. If running, please stop of: /etc/init.d/httpd stop

   c. Disable automatic startup: chkconfig httpd off

3. Apache Configuration. In this example, we will assume default Apache settings and directory locations, specifically:

   a. Validate the following settings in /etc/httpd/conf/httpd.conf:

      i. ServerRoot “/etc/httpd”

      ii. DocumentRoot “/var/www/html”

iii. Listen 192.168.197.150:80 (note 192.168.197.150 is the Apache Switchable IP we will configure later in the LifeKeeper user interface)

b. Edit the Listen parameter in the /etc/httpd/conf.d/ssl.conf configuration file

i. Listen 192.168.197.150:443

4. Create a sample Index page:

a. vi /var/www/html/index.php

b. Insert the following single line of code into this file:

<? phpinfo(); ?>

## Secondary Server

1. Install Apache/PHP exactly as you did on the primary server, making all of the same configuration changes.
2. There is no need to perform Step #4 in which you create a sample index page. All data in /var/www will be replicated from LinuxPrimary to LinuxSecondary
3. Stop all Apache services on the secondary server

# 12.8.6. Install, Configure, and Start MySQL – Apache

## Primary Server

On your Primary server, perform the following actions:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

   ```
   # yum install mysql mysql-server
   ```

2. Verify that your Shared iSCSI LUN is still mounted at /var/lib/mysql via the “df” command

3. If this is a fresh MySQL install, initialize a sample MySQL database:

   ```
   # /usr/bin/mysql_install_db —datadir=”/var/lib/mysql” —user=mysql
   ```

4. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

   ```
   # chown –R mysql:mysql /var/lib/mysql
   ```

   ```
   # chmod 755 /var/lib/mysql
   ```

5. Finally, manually start the MySQL daemon from the command line. Note: **Do Not** start it via the “service” command, or the /etc/init.d/ scripts

   ```
   # mysqld_safe —user=root —socket=/var/lib/mysql/mysql.sock —port=3306 —datadir=/var/lib/mysql —log &
   ```

6. Verify MySQL is running by connecting with the mysql client:

   ```
   [root@LinuxPrimary mysql]# mysql

   Welcome to the MySQL monitor. Commands end with ; or \g.

   Your MySQL connection id is 2

   Server version: 5.0.77-log Source distribution

   Type ‘help;’ or ‘\h’ for help. Type ‘\c’ to clear the buffer.

   mysql> exit
   ```

Bye

[root@LinuxPrimary mysql]#

7. Update the root password for your mysql configuration. In this example we set the MySQL root password to “SteelEye”

   # echo “update user set Password=PASSWORD where User=‘root’; flush

   privileges” | mysql mysql

8. Verify your new password:

   # mysql mysql –u root –p

   (Enter “SteelEye” as the password)

   #exit

9. Create a MySQL configuration file. We will place this in the same shared directory (/var/lib/mysql/my.cnf)

   # vi /var/lib/mysql/my.cnf

**Example**

```
 # cat /var/lib/mysql/my.cnf
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
pid-file=/var/lib/mysql/mysqld.pid
user=root
port=3306
 # Default to using old password format for compatibility with mysql 3.x
 # clients (those using the mysqlclient10 compatibility package).
old_passwords=1
 # Disabling symbolic-links is recommended to prevent assorted security risks;
 # to do so, uncomment this line:
 # symbolic-links=0
[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
[client]
user=root
password=SteelEye
```

10. Delete the original MySQL configuration file, located in /etc

    # rm /etc/my.cnf

# Secondary Server

On your Secondary Server:

1. Install both the “mysql” and “mysql-server” rpm packages if they do not exist on your system.

   Apply any required dependencies as well

   ```
   # yum install mysql mysql-server
   ```

2. Ensure that all files in your MySQL data directory (/var/lib/mysql) have correct permissions and ownership

   ```
   # chown –R mysql:mysql /var/lib/mysql
   ```

   ```
   # chmod 755 /var/lib/mysql
   ```

3. There is no need to perform any of the additional steps taken on the Primary Server

# 12.8.7. Install SIOS Protection Suite for Linux – Apache

For ease of installation, SIOS has provided the SIOS Protection Suite for Linux with an installation script. Towards the end of the script, the desired Application Recovery Kits (ARKs) should be selected for installation. The software will be installed to the following locations:

| SPS for Linux Component | Install Location |
|---|---|
| LifeKeeper Software | /opt/LifeKeeper |
| LifeKeeper Config File | /etc/default/LifeKeeper |

Perform the following actions on **both** Primary and Secondary server.

## Download Software

1. Open the SIOS Protection Suite evaluation email you received from SIOS.
2. Download the SIOS Protection Suite Software from the link provided in your email. It is generally easiest to use “wget” to recursively download all files. Example:

   a. # cd /root

   b. # wget -r <URL>

   c. After successful download you will have downloaded contents similar to the follow directory listing:

   [root@LinuxPrimary ~]# ls -l <directory> total 63680

   -rw-r—r— 1 root root 23163 May 30 14:03 EULA.pdf

   -rw-r—r— 1 root root 536 May 30 14:03 readme.txt

   -rw-r—r— 1 root root 65179648 May 30 14:03 sps.img
3. Download your Evaluation license key from the link specified in your evaluation email. Save the license file to an easy to remember location on both servers.

## Run the SIOS Protection Suite Installer Script

1. Loopback mount the sps.img file previously downloaded, which is an ISO9660 image file. Run the “setup” script inside:

   # mount -o loop sps.img /mnt

```
# cd /mnt

# ./setup
```

2. During this procedure, you will hit Enter in most cases to accept default values and continue to the next screen. Note the following exceptions:

   a. On the screen titled “High AvailabilityNFS” you may select “n” as in this particular eval guide we will not be creating a highly available NFS server cluster configuration.

   b. If you have plans to create a highly available NFS service, adjust your response accordingly.

3. Towards the end of the setup script, you can choose to install a trial license key now, or later. We will install the license key in the next step, so you can safely select “n” at this point

4. In the final screen of the “setup” select the DataKeeper from the list displayed on the screen.

5. The following RPMs should be be installed:

   a. steeleye-lkAPA-<version>.noarch.rpm

   b. steeleye-lkSQL-<version>.noarch.rpm

   c. steeleye-lkDR-<version>.noarch.rpm

6. Un-mount the Distribution Enabling disk image:

```
# cd /root

# umount /mnt
```

## Install the Evaluation License Keys

The last phase of the setup process installs the licensing keys. You must install the evaluation license key file (“.lic”) that you downloaded with your evaluation software before starting the SIOS Protection Suite for Linux.

1. To install your trial license key, run the “lkkeyins” command on both Primary and Secondary Server. This command is located at /opt/LifeKeeper/bin/lkkeyins . Example:

```
# /opt/LifeKeeper/bin/lkkeyins <path_to_license/<filename>.lic
```

2. Validate your license keys were installed via the /opt/LifeKeeper/bin/lklicmgr command

```
# /opt/LifeKeeper/bin/lklicmgr

License File: 20101230.lic
```

| Product | Type | Expiry |
|---|---|---|
| LifeKeeper for Linux | Eval | 27 Mar 2013 (87 days) |
| Apache Recovery Kit | Eval | 27 Mar 2013 (87 days) |
| SIOS Data Replication ARK | Eval | 27 Mar 2013 (87 days) |
| MySQL Recovery Kit | Eval | 27 Mar 2013 (87 days) |

# Start the SIOS Protection Suite for Linux

1. Start:

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

# 12.8.8. Configure the Cluster – Apache

## Primary Server

Complete the following steps on the primary server to configure the cluster:

- Create TCP Communication (Comm) Path(s)
- Verify the Communication (Comm) Path(s)

Before you begin, SIOS recommends at least two TCP communications paths between each server within the cluster to each remote server for heartbeat redundancy.

> **Important:** Supported configurations require that you define redundant comm. paths, so that the failure of a single communication line will not cause a split brain where resource hierarchies may come in-service on multiple servers simultaneously.

### Access the LikeKeeper GUI

The LifeKeeper Graphical User Interface (GUI) is a Java based application that can be run as a native Linux application.

The LifeKeeper GUI is based on Java RMI with callbacks. Hostnames must be resolvable or you may receive a Java 115 or 116 error.

1. Verify that both short and fully qualified hostnames of all cluster nodes resolve to the proper locations

   # ping LinuxPrimary

   # ping LinuxPrimary.domain.com

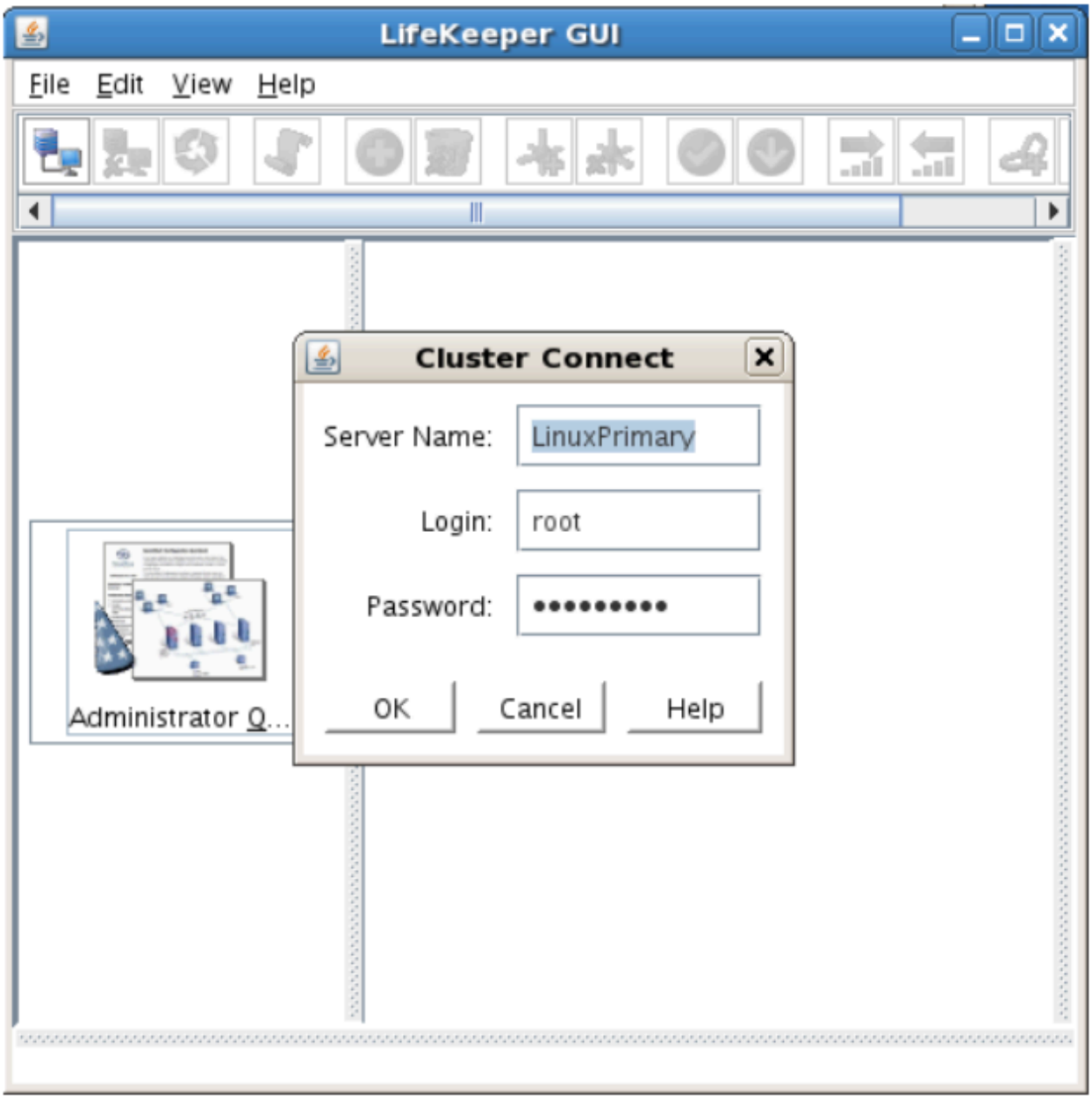
   # ping LinuxSecondary

   # ping LinuxSecondary.domain.com

2. To start the LifeKeeper Linux GUI Application:

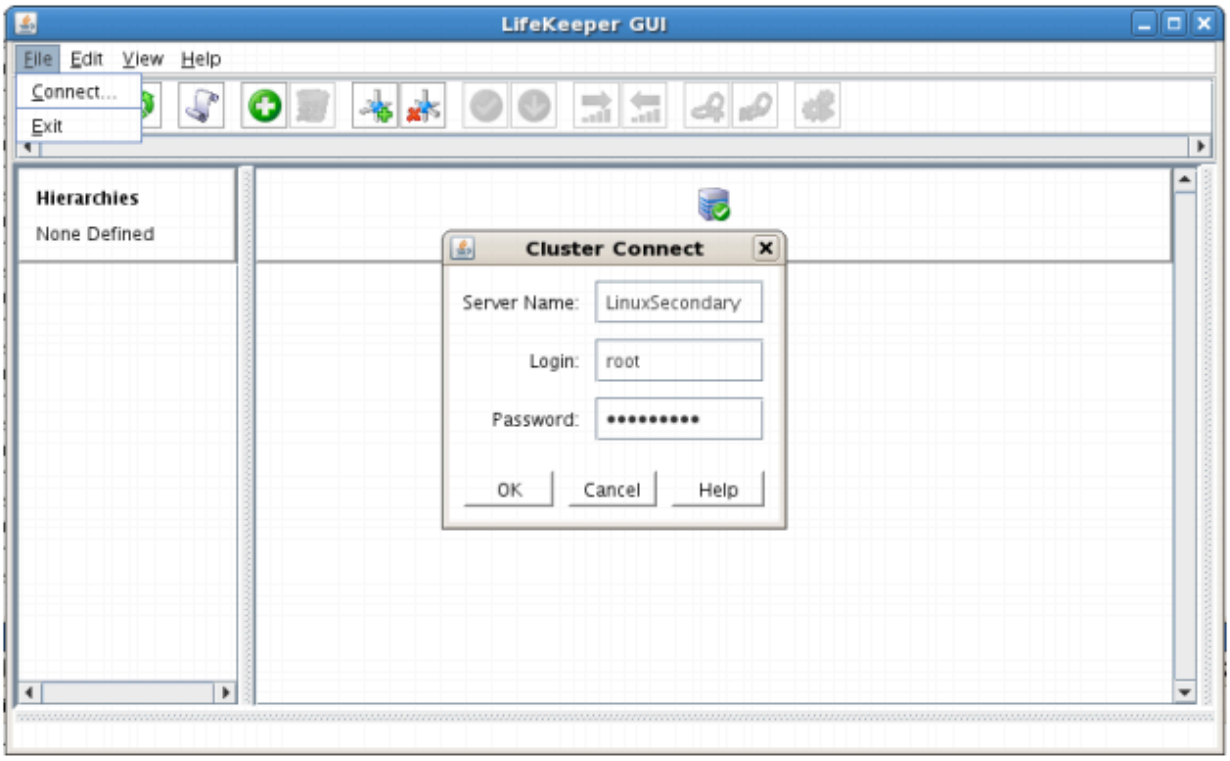
   a. /opt/LifeKeeper/bin/lkGUIapp &

3. Enter the name of the server you wish to connect to (this field will be populated with the name of the server you are on, if you are running the GUI from a server with LifeKeeper installed) along with your root credentials and click OK.



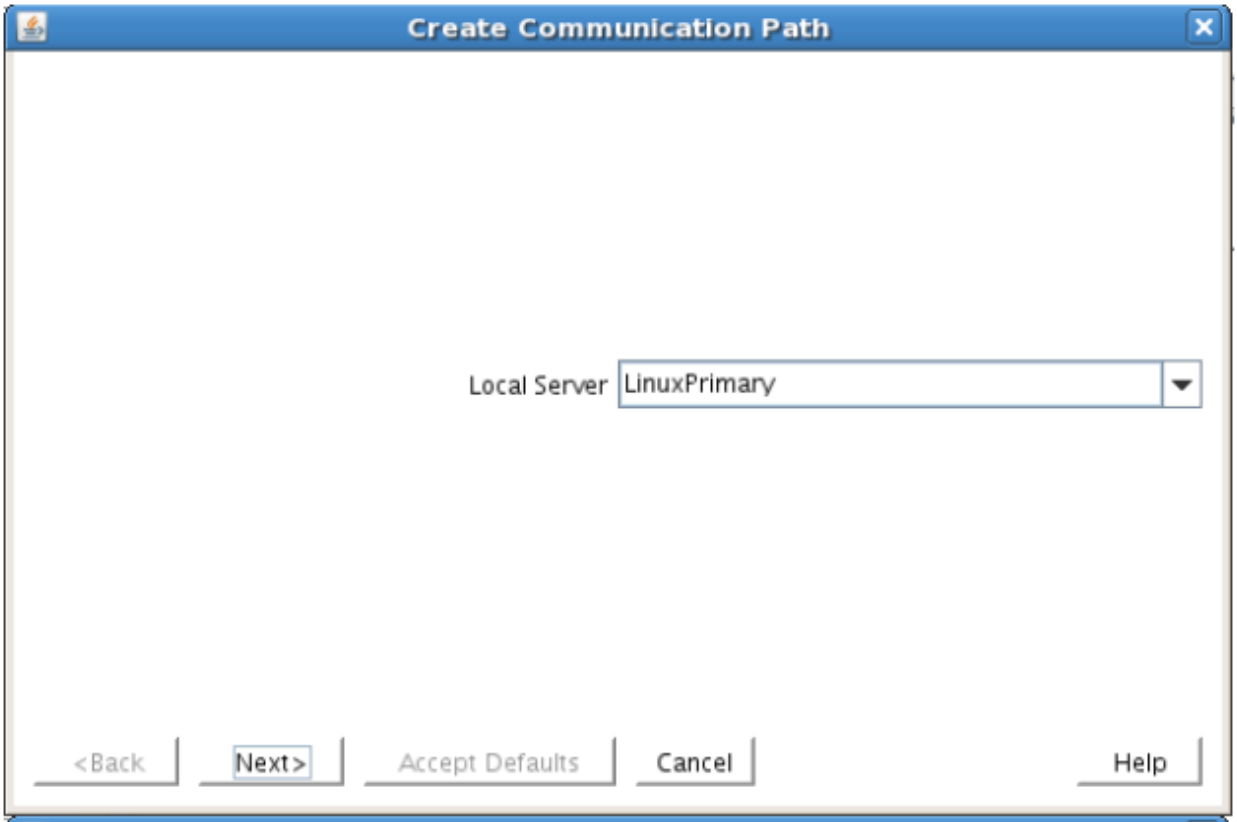

## Create Communication (Comm) Paths

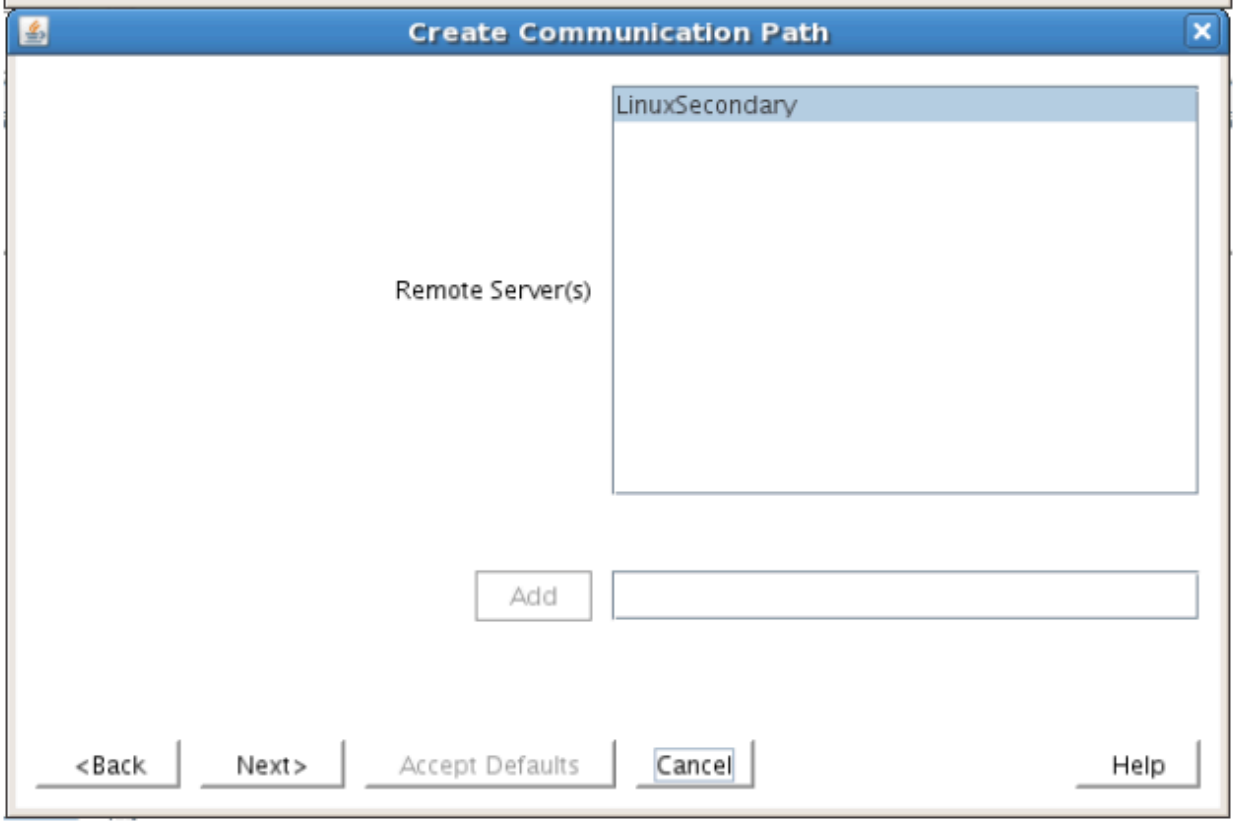
4. Within the LifeKeeper GUI, from the File menu, select Connect. Enter the name of your Secondary server, login and password when the Cluster Connect window displays.



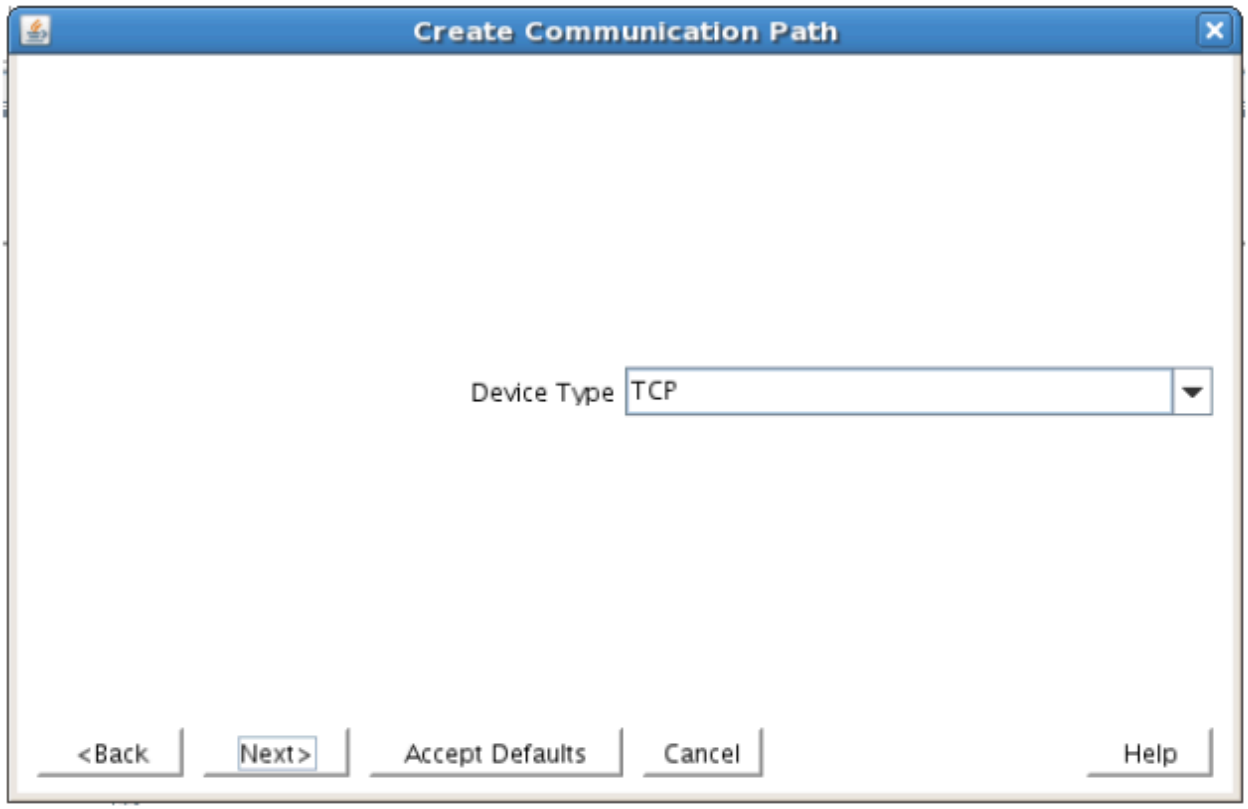

5. Within the LifeKeeper GUI, click the Create Comm Path button on the toolbar. You can also right click one of the servers and click Create Comm Path from the pop-up menu as well.

6. Select your Local and Remote Server(s) from the list box. If a server does not appear in the list box, you may enter it by typing its name and clicking the Add Server button. When using the Add Server procedure, you must make sure that the computer names for both network interfaces on the servers respond correctly when you ping them (from all of the partner server(s)) using the **ping –a IP ADDRESS** syntax. If they do not, this must be corrected prior to continuing. Click Next.

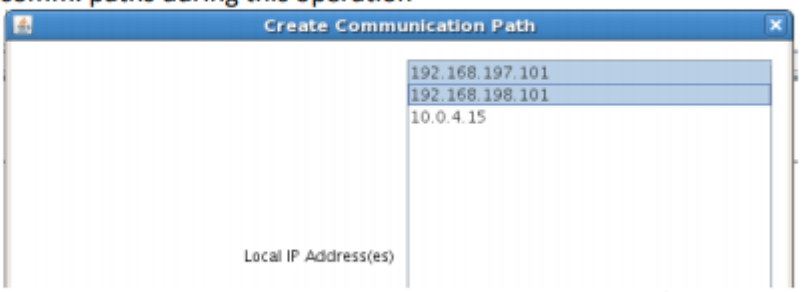


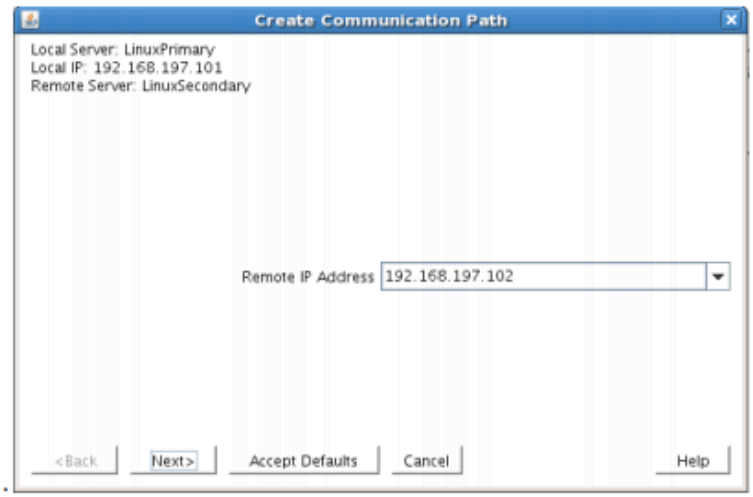




7. Select TCP for Device Type and Click Next.



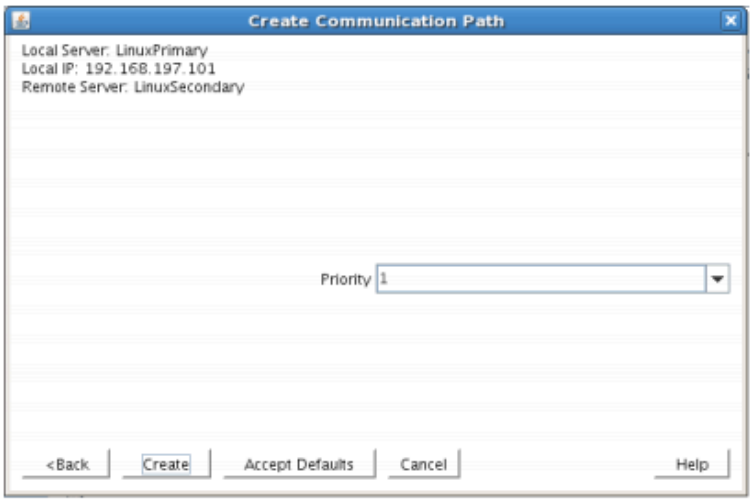

8. Provide all the required information and click Next for the following series of dialog boxes. For each field in the dialog box you can click Help for further information or refer to the table below for an explanation or recommendation

| Field | Tips |
| --- | --- |
| ***For TCP/IP Comm Path...*** | |
| Local IP Address | Choose the IP address to be used by the local server for this comm path. Select both interfaces so that the wizard creates multiple comm. paths during this operation   |
| Remote IP Address | Choose the IP address to be used by the remote server for this comm path |

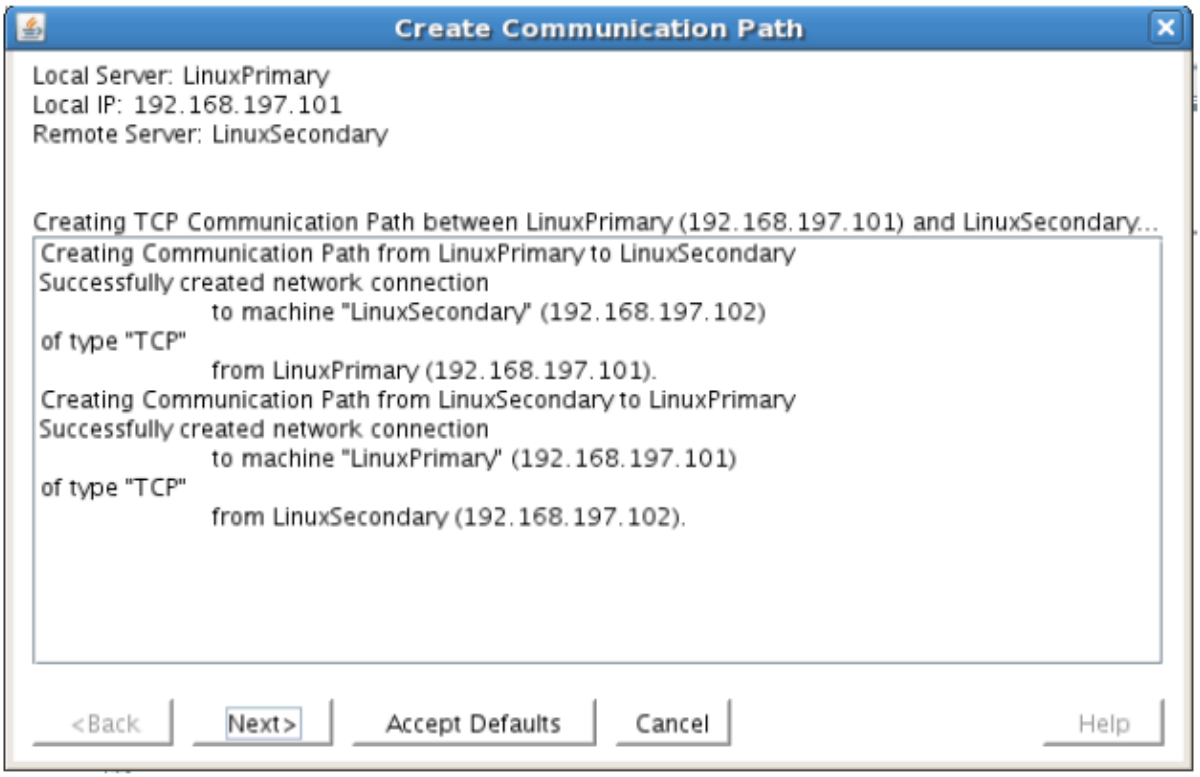



Enter the priority for the comm path on the local server. The priority will be used to determine the order that the comm paths between two servers will be used. Priority 1 is the highest; 99 is the lowest.

Priority




9. After entering data in all the required fields, select Create. A message will display indicating the network communication path is successfully created. Click Next.



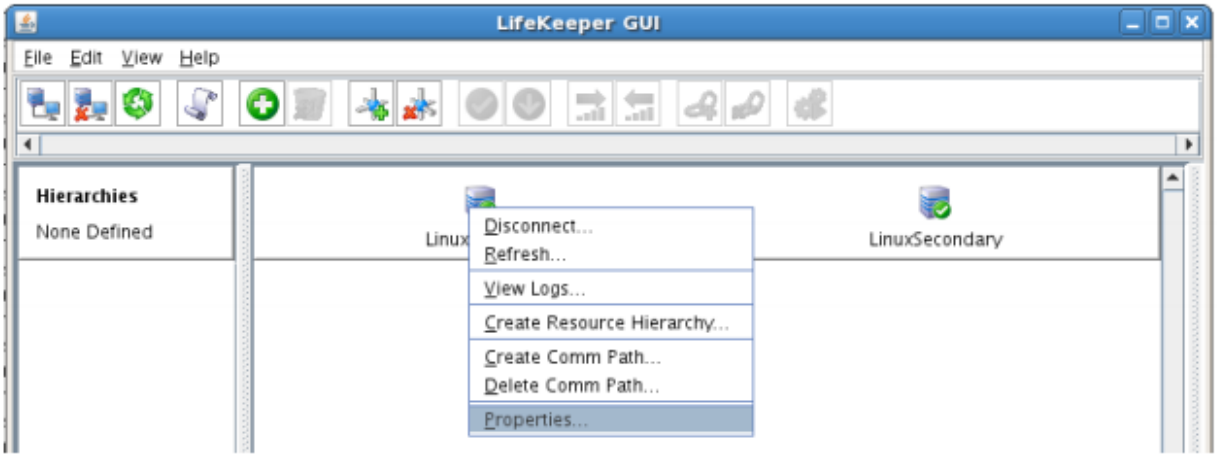

If you selected multiple Local IP Addresses or multiple Remote Servers and the Device Type was set to TCP, then the procedure will return you to the setup wizard the next Comm Path.

10. Select Done in the last dialog box.

    Repeat this process until you have defined all the communication paths you plan to use. SIOS strongly recommends that you define at least two communication paths for redundancy.

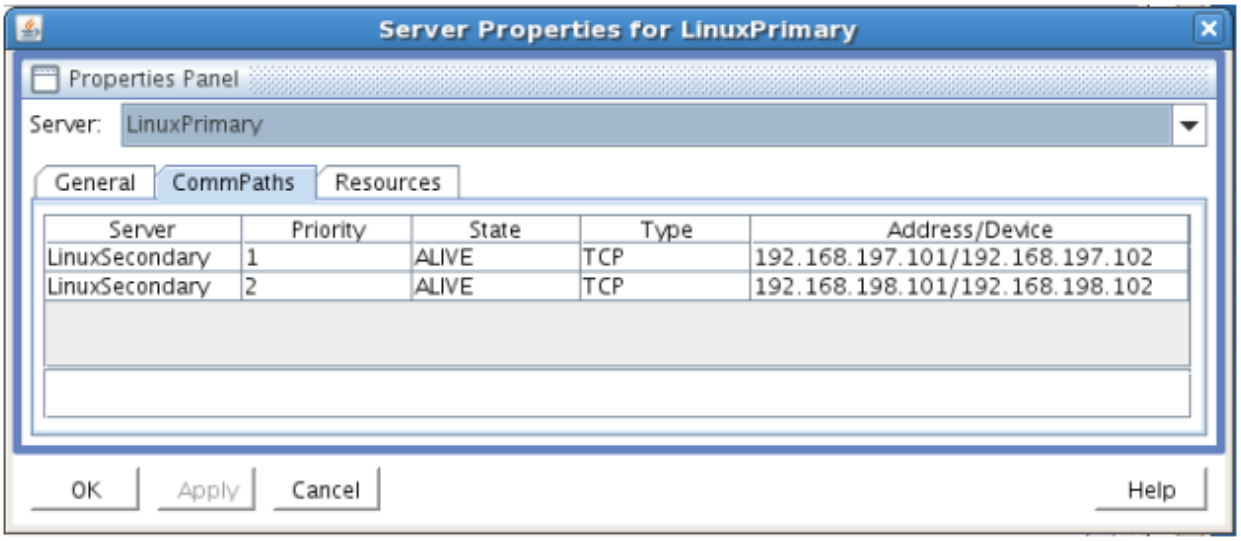
## Verify the Communications Paths

1. Verify that the communications paths are configured properly by viewing the Server Properties dialog box. From the LK GUI, select Edit, Server, Properties and then the Comm Paths tab.




2. Note the State displayed is ALIVE. You can also check the server icon in the right, main pane of

the GUI. If only one comm path has been created, the server icon shows a yellow warning icon on the server icon, indicating that one comm. path is ALIVE, but there is no redundant comm path. The server icon will display a green heartbeat checkmark when there are at least two comm paths configured and ALIVE.

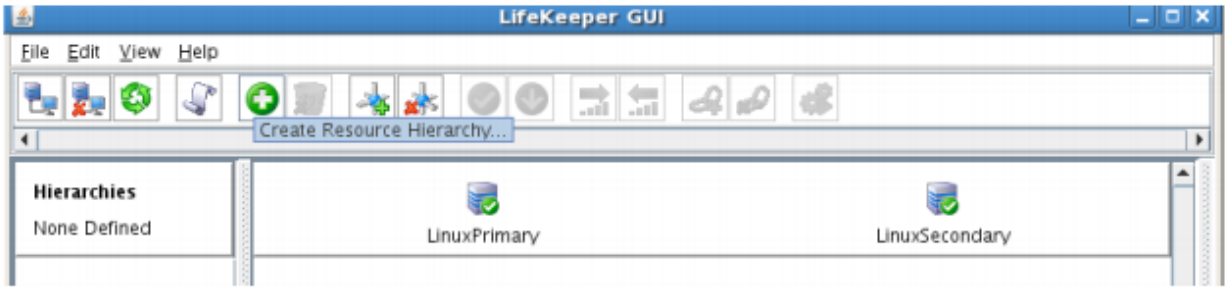



# Create the LifeKeeper Hierarchy

## Create and Extend an IP Resource

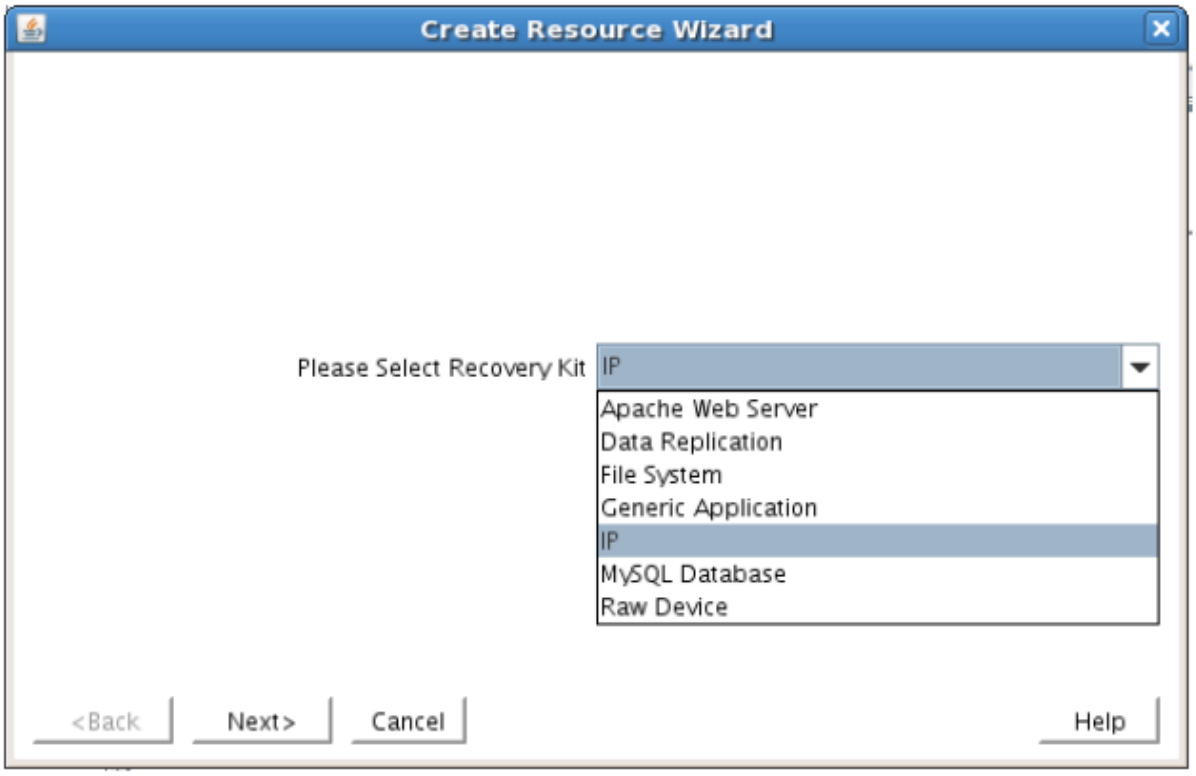
In LifeKeeper, create an IP resource and extend it to the secondary server by completing the following steps. This Virtual IP will have the ability to move between cluster nodes along the application that depends on it.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.




The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

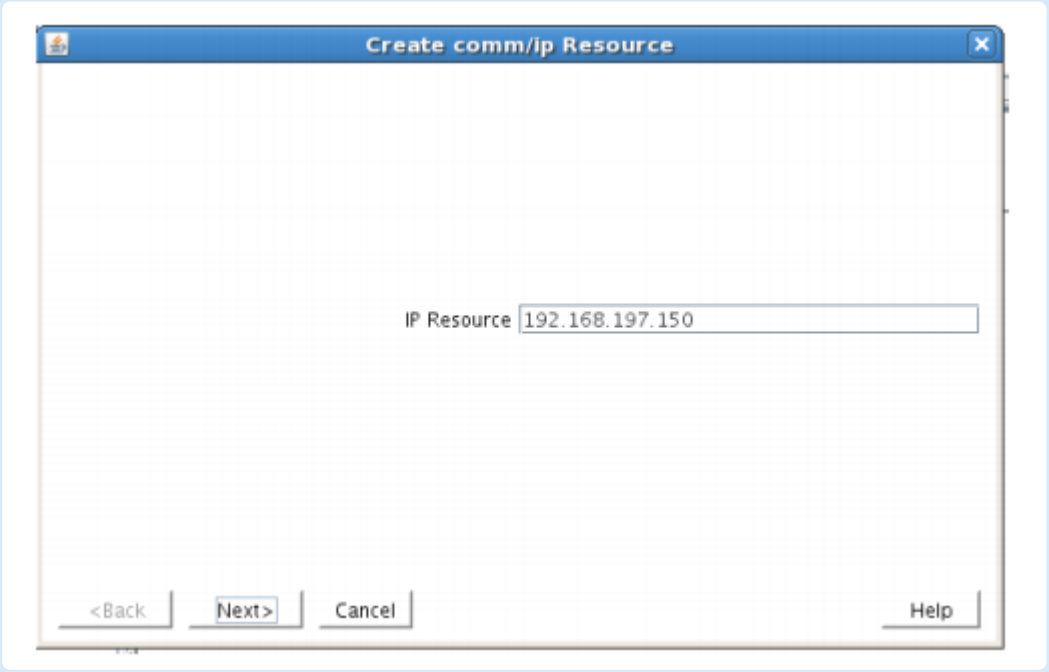
2. Select IP Address and click Next.

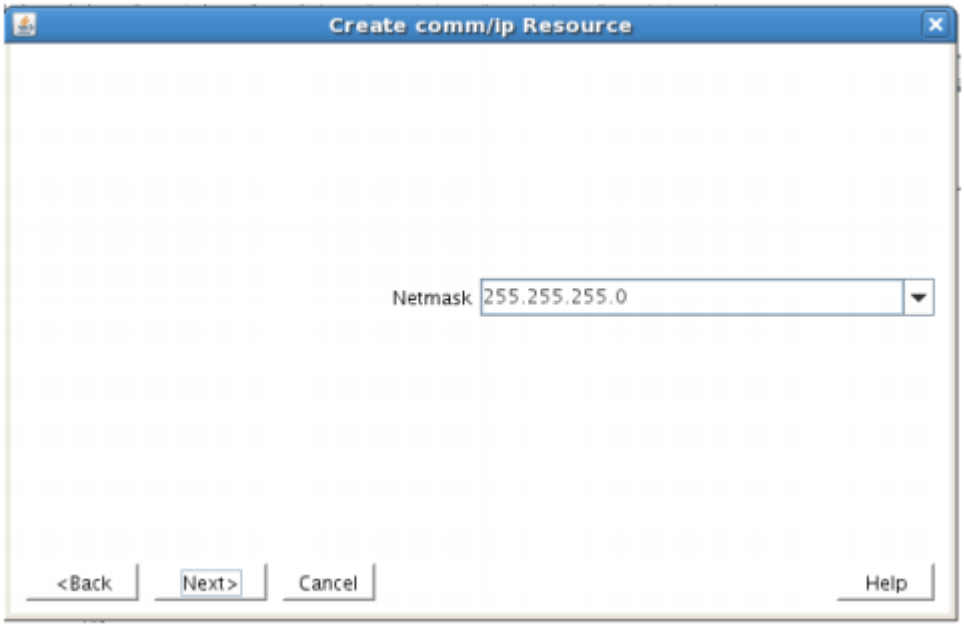



3. Enter the appropriate information for your configuration. The table below contains a list of the fields that display and additional information to assist you as you complete this procedure. Recommended values are also show below. You can also click the Help button for further information. Press Next to continue after entering the required information.

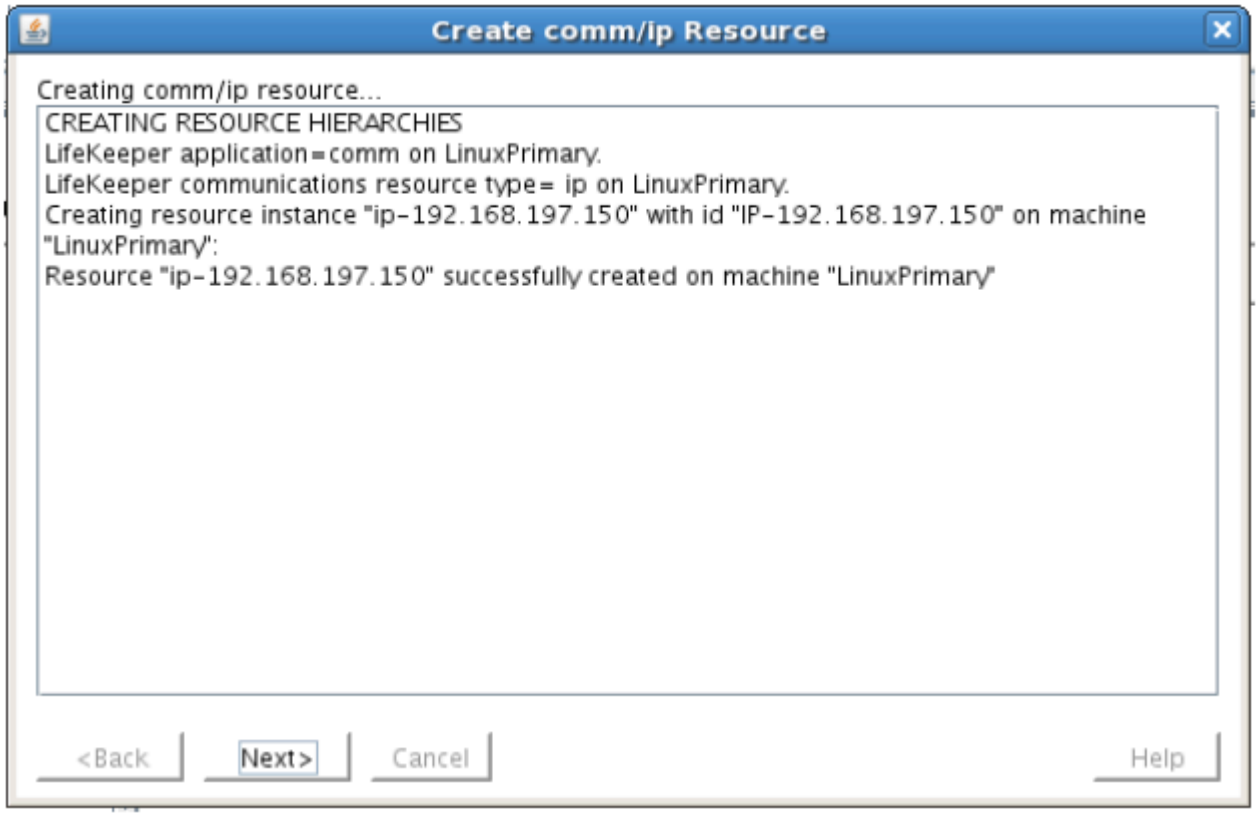
## IP Creation Field Definitions

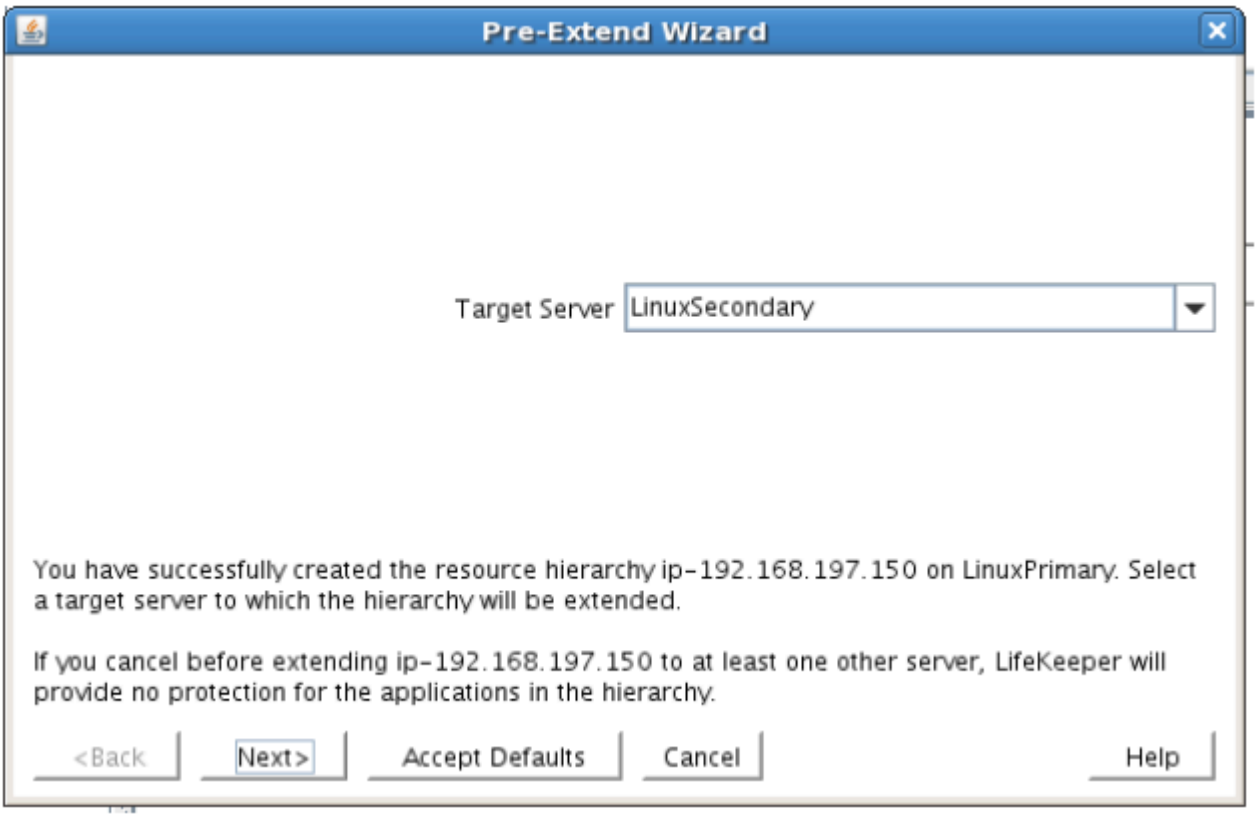
| Field | Tips |
|---|---|
| Resource Type | Select IP Address as the resource type and click Next. |
| Switchback Type | Select Intelligent and click Next. |
| Server | Select the Server where the IP resource will be created. Select your Primary server and click Next. |
| IP Resource | Enter the virtual IP information and click Next<br><br>Example **192.168.167.151**<br><br>**Note** This is an IP address that is not currently in use anywhere on your network. This is the address that all clients will use to connect to the protected resources.<br><br>In this configuration example, we will be protecting two (2) virtual IPs.<br><br>First we will protect 192.168.197.150, which our Apache webserver will |

| | |
|---|---|
| | use.<br>  <br><br>The second time through this wizard we will protect 192.168.197.151, which will be used by MySQL |
| Netmask | The IP subnet mask that your TCP/IP resource will use on the target server. Any standard netmask for the class of the specific TCP/IP resource address is valid<br><br>In our sample configuration 255.255.255.0 is used for a subnet mask on both networks.<br><br>**Note:** The subnet mask you choose, combined with the IP address, determines the subnet that will be used by the TCP/IP resource and should be consistent with the network configuration. |

| | |
|---|---|
| |  |
| Network Connection | This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. Select the correct NIC and click Next. |
| IP Resource Tag | Accept the default value and click Next. This value only affects how the IP is displayed in the GUI. The IP resource will be created on our Primary server. |

4. LifeKeeper will create and validate your resource. After receiving the message that the resource has been created successfully, click Next when the following dialog box appears so that you can complete the process of Extending the IP Resource to our Secondary server, below.




Extending the IP resource will start automatically after you have finished creating an IP address resource if you clicked Next in the dialog box displayed above. You can also start this from an existing IP address resource by right clicking on the active resource and selecting Extend Resource Hierarchy.

Refer to the table below to complete the Extend IP Resource procedure.

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Leave as “intelligent” and click Next |
| Template Priority | Leave as default (1) |
| Target Priority | Leave as default (10) |
| Network Interface | This is the physical Ethernet card that the IP address is interfacing with. Chose the network connection that will allow your virtual IP address to be routable. The correct physical NIC should be selected by default. Please verify and then click Next |
| IP Resource Tag | Leave as default. |
| Target Restore Mode | Select Enable and click Next. |
| Target Local Recovery | Select Yes to enable Local Recovery for the SQL resource on the Target server. |
| Backup Priority | Accept the default value. |

5. After receiving the message Hierarchy extend operations completed, click Finish and then click Done

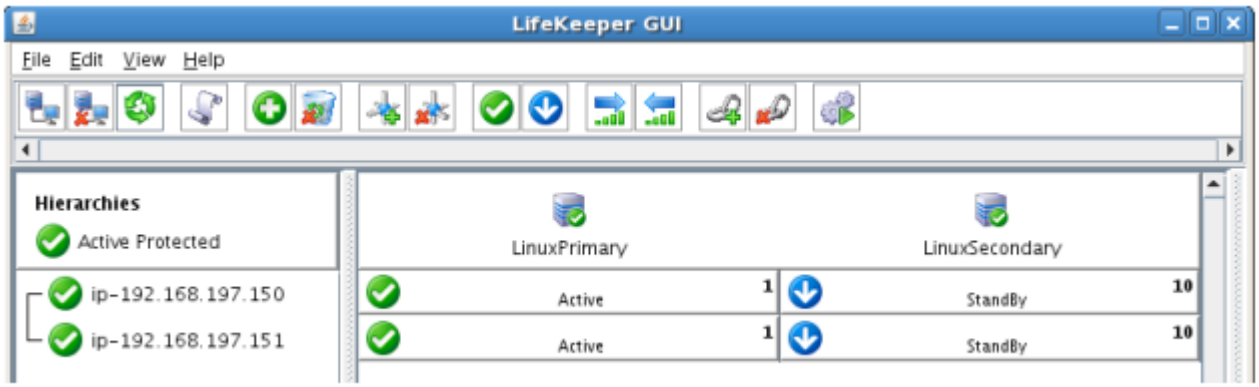
6. Your IP resource (192.168.197.151) is now fully protected and has the ability to “float” between cluster nodes as needed. Looking at the LifeKeeper GUI you will notice that the IP resource is Active on the Primary cluster node and Standby on the Secondary cluster node

## Create a Second IP Resource

Repeat the procedure above to protect a 2nd IP resource.

This second time, protect 192.168.197.151, which is the IP address our MySQL database will later use.

As a result, your LifeKeeper GUI will display as follows, with both IP resources Active and protected on the Primary cluster node:



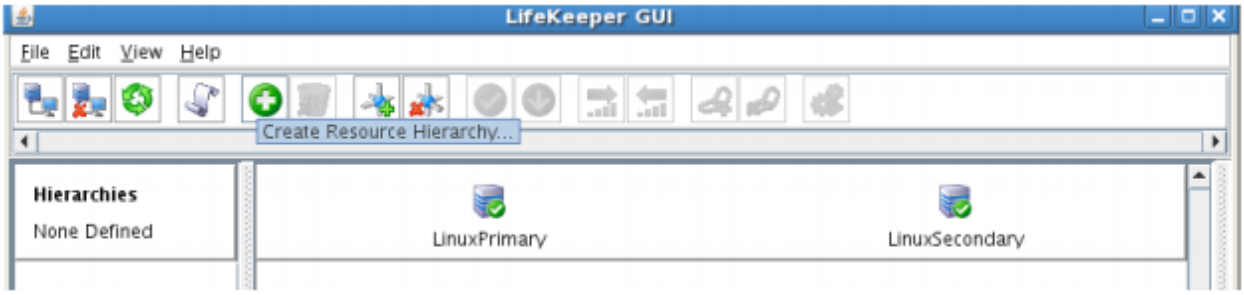

## Create a Mirror and Begin Data Replication

In this section we will setup and configure the Data Replication resource, which be used to synchronize our Apache Webserver's data between cluster nodes. The data we will replicate resides in the /var/www partition on our Primary cluster node

Please note:

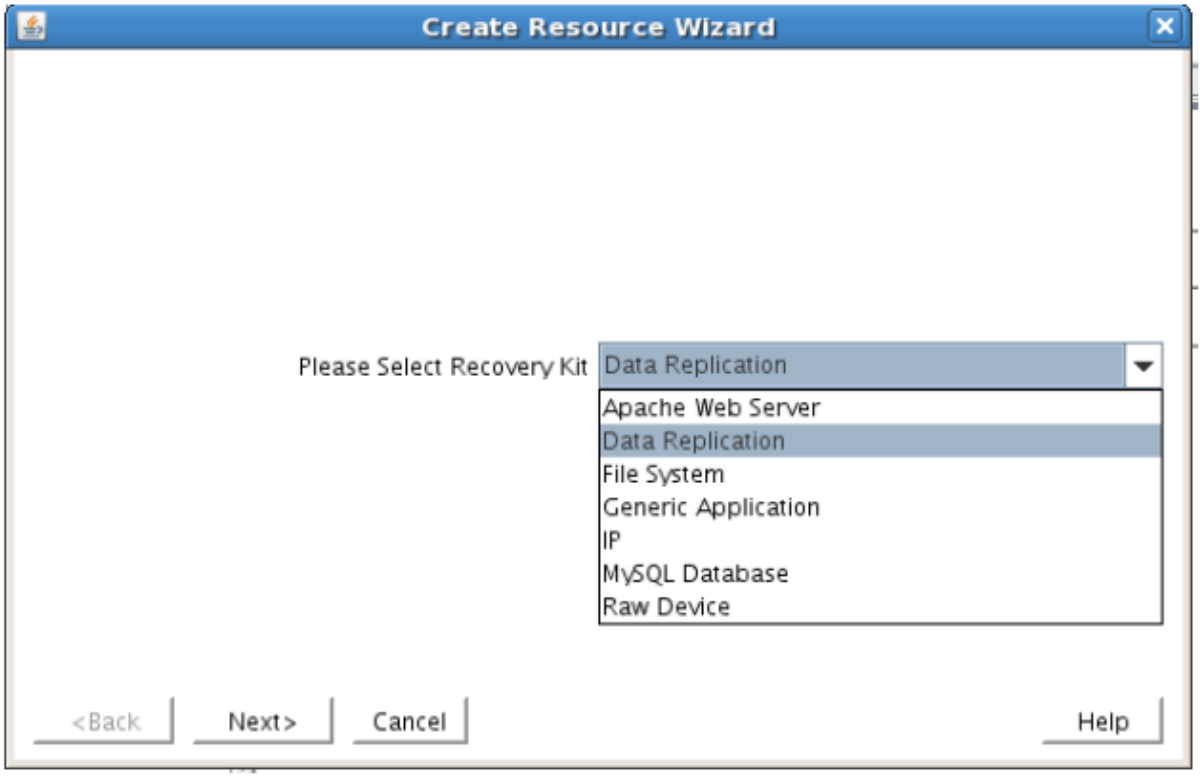
- The source volume to be replicated must be mounted on the Primary server
- The target volume, which will received replicated data, must NOT be mounted on the Secondaryserver.
- The target volume's size must equal to or larger than the size of its source volume.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.




The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

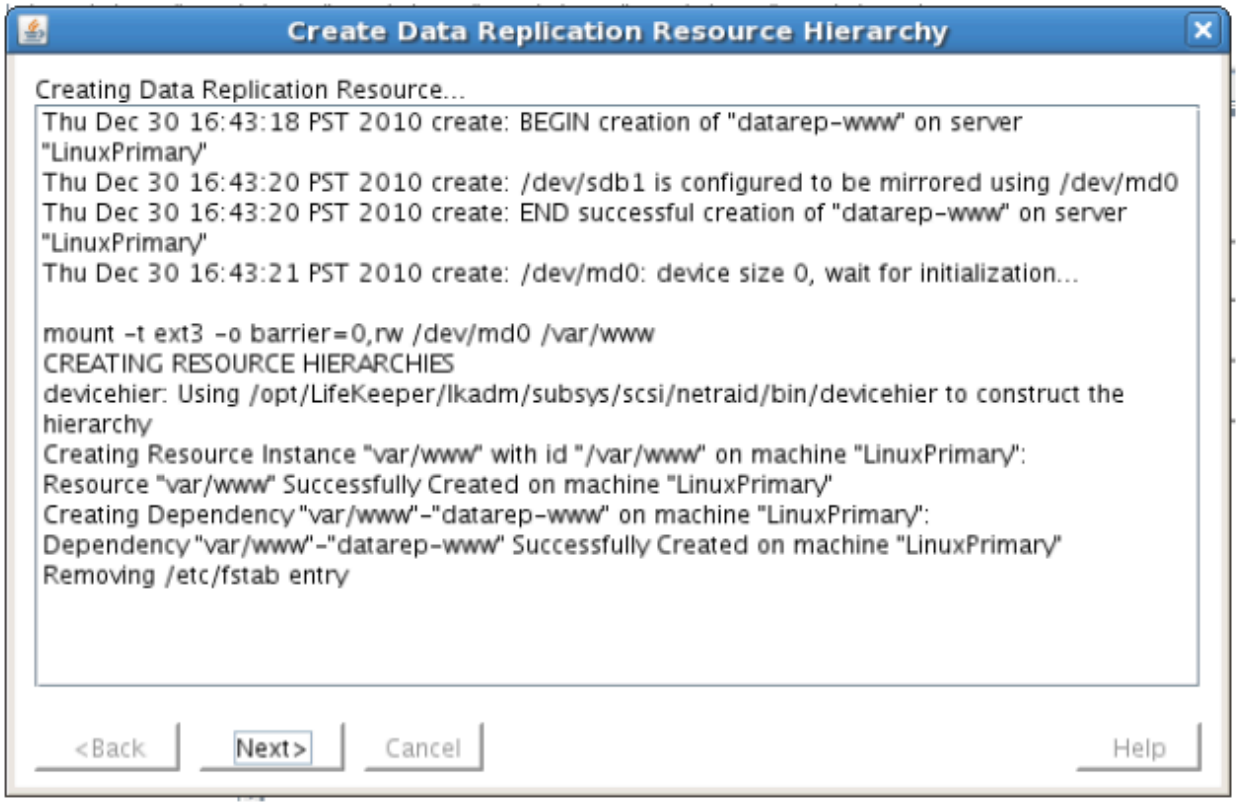
2. Select Data Replication and click Next.



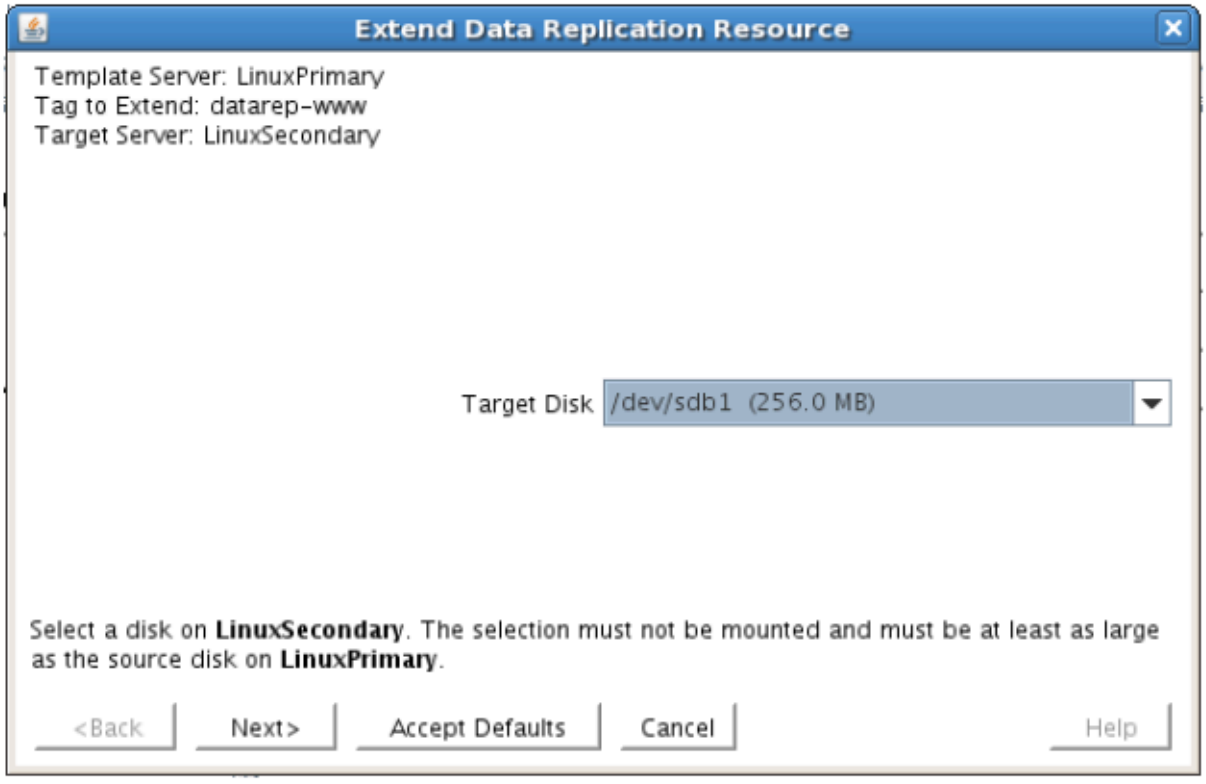

3. Follow the Data Replication wizard, and enter the following values:

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node, i.e. Mirror Source) |
| Hierarchy Type | Select: “**Replicate Existing Filesystem**” |
| Existing Mount Point | At this step you will select the mounted partition to replicate. In our example, select “/var/lib/mysql” |
| Data Replication Resource Tag | Leave as default |
| File System Resource Tag | Leave as default |
| Bitmap File | Leave as default (Note: if using high speed SSD storage you will want to create a small partition and use it for bitmap placement, i.e. /bitmaps) |
| Enable Asynchronous Replication | Leave as default (Yes) |

4. Click Next to begin creation of the Data Replication resource hierarchy. Status will be displayed in the GUI as follows:

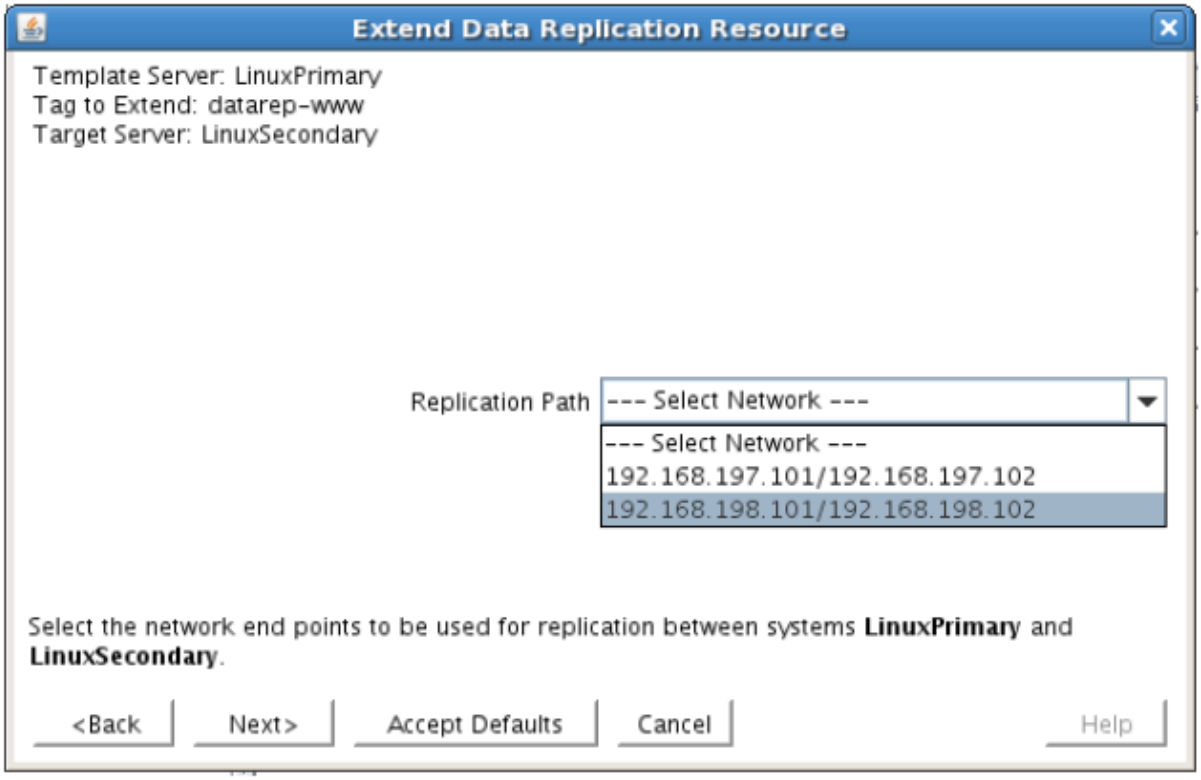



5. Click Next to begin the process to Extend the Data Replication Resource. Select all default settings. When it asks for the target disk, select a free partition on your Target server which is the same size (or greater) than the Source Volume we are replicating. This partition should NOT be mounted on the Target system.




6. Continue through the wizard, and you will be prompted to select the network you would like

replication to take place over. In general, it's a best practice to separate your user/application and your replication traffic. In our example setup we will replicate over our backend network, 192.168.198.X



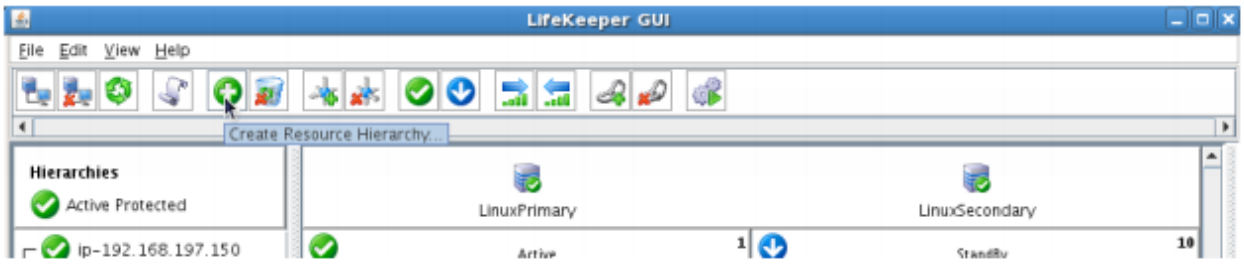

7. Click Next and continue through the wizard. Once completed, your resource hierarchy will look as follows

## Create the Apache Hierarchy

In this section we will create an Apache resource hierarchy on the primary server and extend it to the backup server. This step will create a dependency on the IP resource created in previous the step.

**Important:** The Apache web server should not be running at this time.

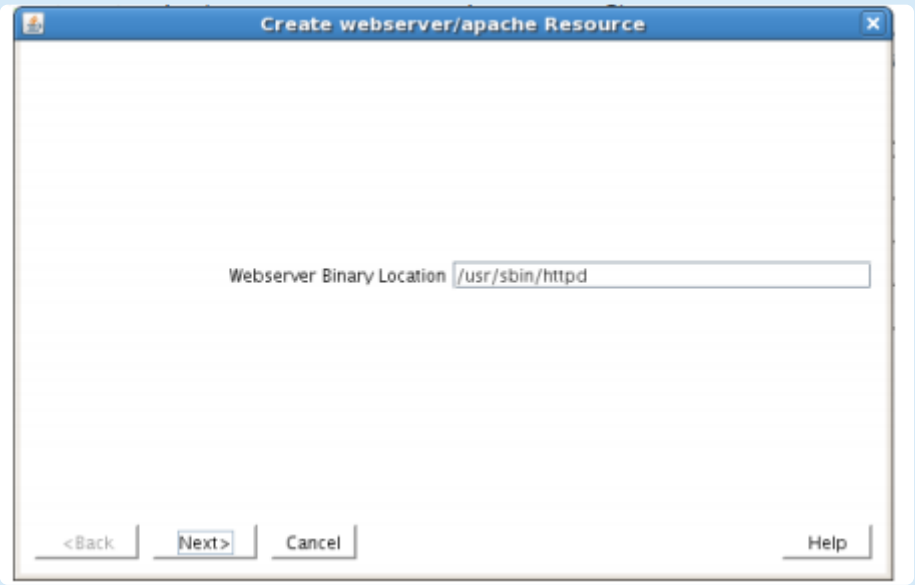
1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.

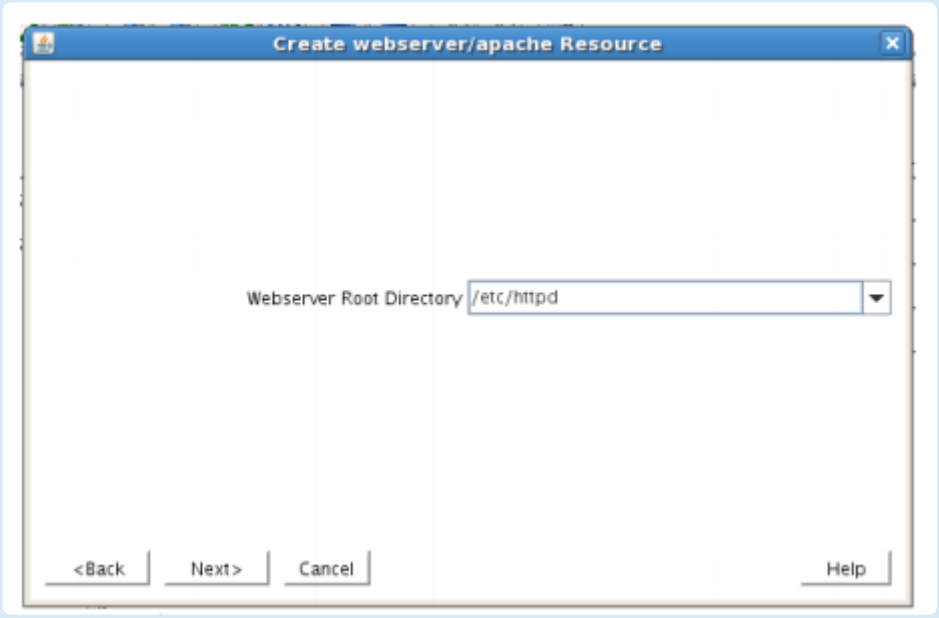



The Create Resource Wizard dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.

2. Select Apache Web Server and click Next.

3. Proceed Through the Resource Creation wizard, providing the following values

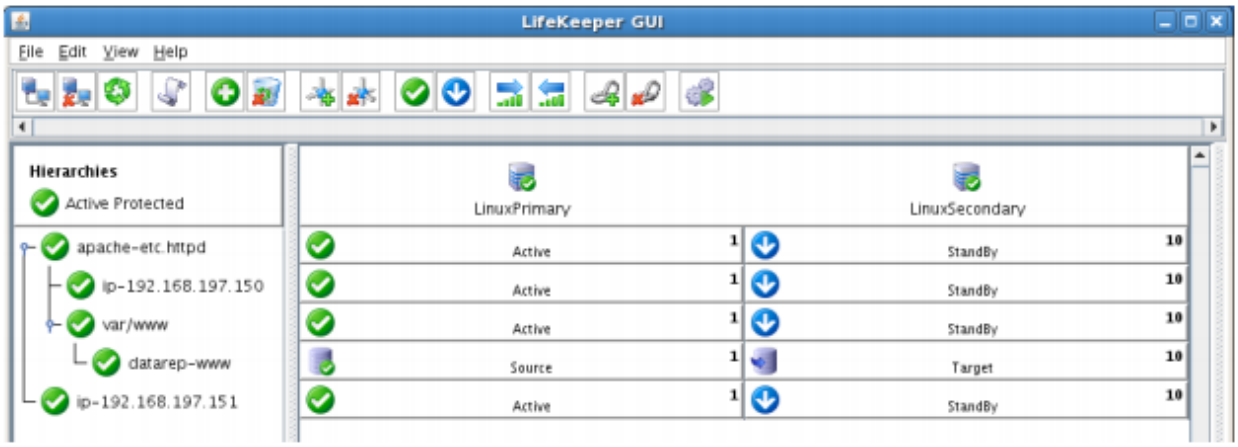
| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node, i.e. Mirror Source) |
| Webserver Binary Location | /usr/sbin/httpd (assumes a standard apache config)<br>  |
| Webserver Root Directory | /etc/httpd (assumes a standard apache config) |

| | |
|---|---|
| |  |
| Root Tag | Leave as default |

4. Click “Create” to begin resource hierarchy creation on the primary server. Once complete, click “Next” to extend this resource to the secondary server.

5. During the Extend Resource wizard, leave all settings as default.

6. Note: during the resource creation process, LifeKeeper extracted the existing configuration of the existing Apache webserver, and identified that it depends on the IP resource (192.168.197.150) and the Data Replication resource (/var/www) that were created in previous steps. These resources now appear underneath the newly created Apache resource, to indicate the dependency relationship.




## Create the Shared Filesystem Resource Hierarchy

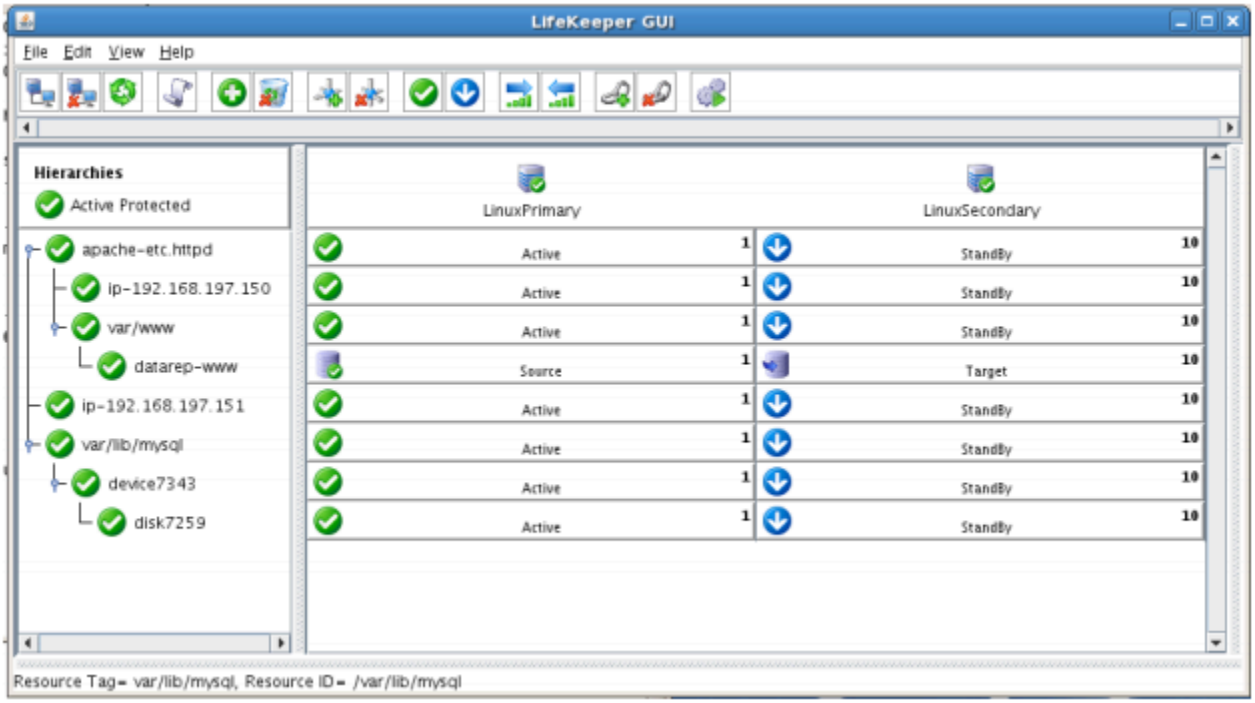
Create a Filesystem resource to protect the shared iSCSI filesystem and make it high available between cluster nodes. SPS for Linux leverages SCSI Persistent Group Reservations (PGR) to lock the LUN, ensuring that only the active cluster node for the storage resource can access it.

**Important:** At this point, the shared iSCSI LUN needs to already be mounted on the Primary Server. It should NOT be mounted on the Secondary Server. See section titled “Configure iSCSI initiator, discover and login to iSCSI target” above to review the steps involved.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select File System and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node) |
| Mount Point | Select /var/lib/mysql. Note that LifeKeeper scans the system for LUNS that are sharable between cluster nodes. The list of possible shared LUNS is presented automatically in this step of the wizard. |

4. Select Create Instance to define this resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the File System resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Your resource hierarchy should look as follows:



## Create the MySQL Resource Hierarchy

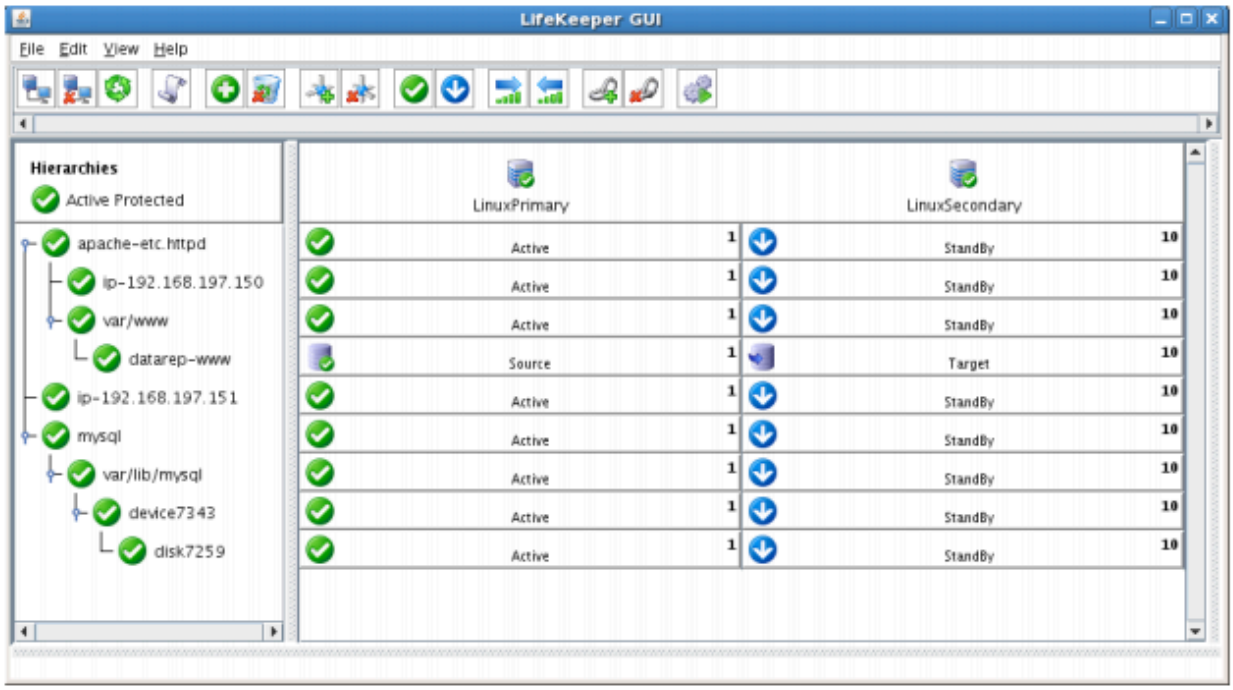
Create a MySQL resource to protect the MySQL database and make it high available between cluster nodes.

> **Important:** At this point, MySQL needs to be already running on the Primary Server. It should NOT be running on the Secondary Server. See section titled “Install, Configure, and Start MySQL” above to review the process to configure and start MySQL as needed.

1. From the LifeKeeper GUI toolbar, click Create Resource Hierarchy.
2. Select **MySQL Database** and click Next.
3. Proceed Through the Resource Creation wizard, providing the following values

| Field | Recommended Entries or Notes |
|---|---|
| Switchback Type | Intelligent |
| Server | LinuxPrimary (Primary Cluster Node) |
| Location of my.cnf | Enter “/var/lib/mysql”. Note that earlier in the MySQL configuration process we created a my.cnf file in this directory. |
| Location of MySQL executables | Leave as default (/usr/bin) since we are using a standard MySQL install/ configuration in this example |
| Database tag | Leave as default |

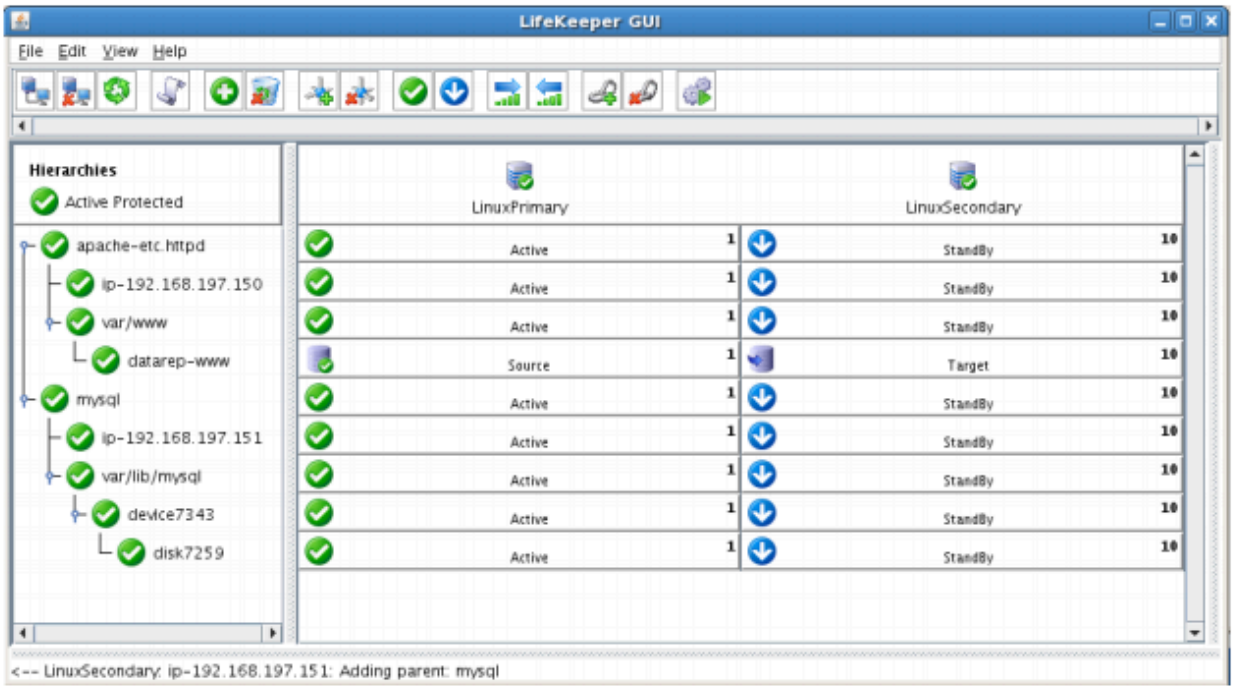
4. Select Create to define the MySQL resource hierarchy on the Primary Server
5. Click Next to Extend the File System Resource to the Secondary Server
6. In the Extend Wizard, select “Accept Defaults”
7. As a result the MySQL resource is now protected on both cluster nodes. Click Finish to exit the Extend wizard.
8. Note: LifeKeeper will automatically identify that the MySQL resource has a dependency on the FileSystem resource (/var/lib/mysql). The Filesystem Resource will appear underneath the MySQL resource in the GUI
9. Your resource hierarchy should look as follows:




## Create the MySQL IP Address Dependency

In this step will define an additional dependency: that MySQL depends on a Virtual IP (192.168.197.151) so that the IP address follows the MySQL database should it move. This IP (.151) is the IP the webserver will use to access the MySQL database.

1. From the LifeKeeper GUI toolbar, right-click on the “mysql” resource
2. Select “Create Dependency” from the right-click context menu
3. In the Child Resource Tag dropdown menu, select “ip-192.168.197.151”
4. Click Next
5. Click Create Dependency
6. Click Done
7. The Virtual IP address resource (192.168.197.151) will now appear underneath the MySQL resource in the LifeKeeper user interface. This ensures that resources move together, and are started/stopped in the proper order.
8. Your resource hierarchy should look as follows




At this point in the Evaluation, we have fully protected Apache, MySQL, and their dependent resources: IP addresses, and Storage, both shared and replicated.

# 12.8.9. Test Your Environment – Apache

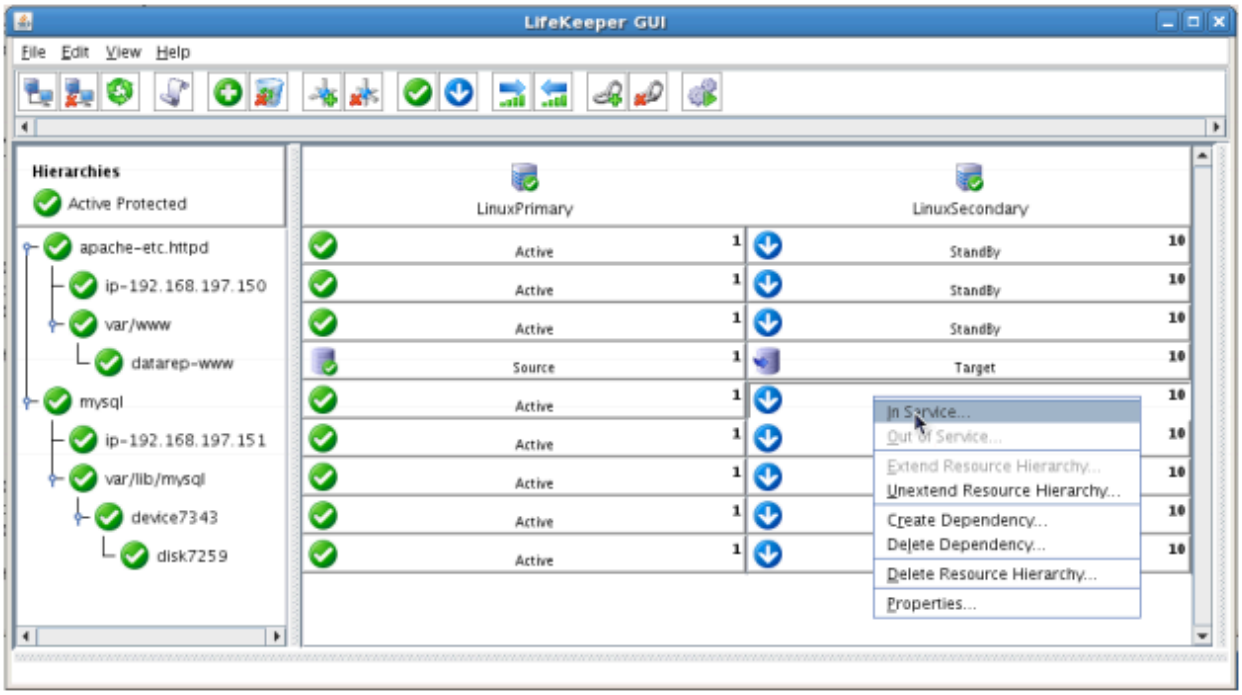
The following test scenarios have been included to guide you as you get started evaluating SIOS Protection Suite for Linux. **Before beginning these tests, make sure the data replication resources are in the mirroring state**.

> **Note:** For these test examples, the Primary Server is referred to as LINUXPRIMARY. The Backup or Secondary Server is referred to as LINUXSECONDARY.

## Manual Switchover of the MySQL Hierarchy to Secondary Server

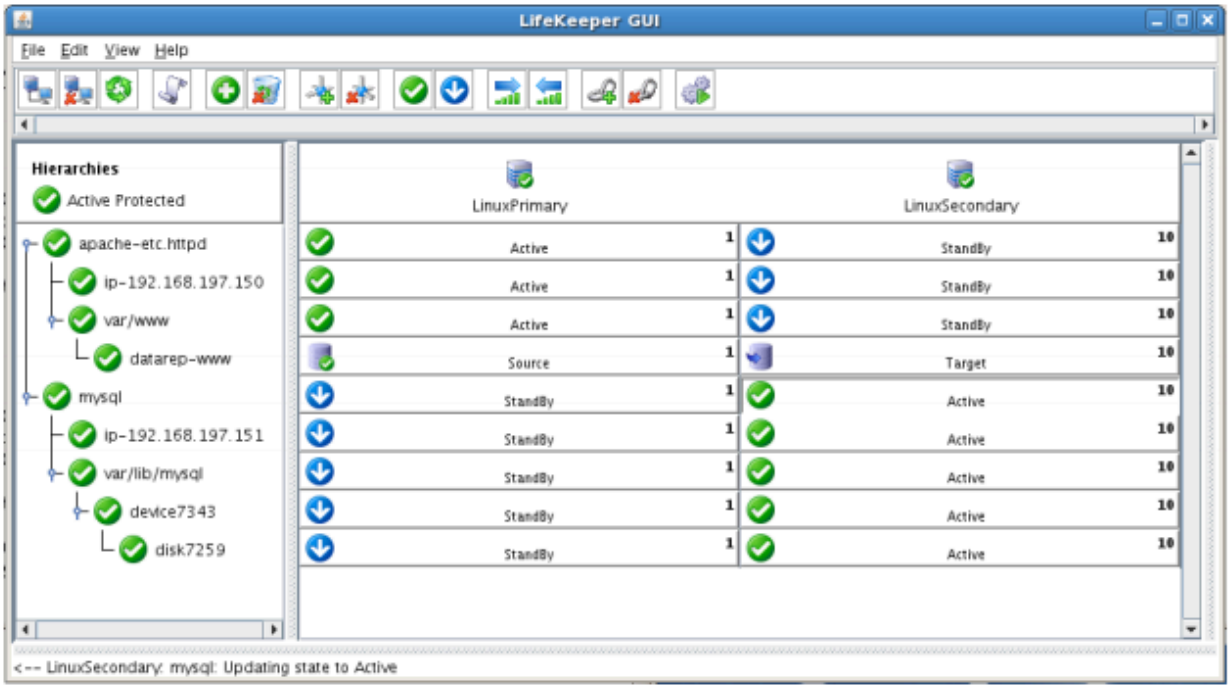
**Procedure:**

- From the LifeKeeper GUI, right click on the MySQL resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up




**Expected Result:**

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXSECONDARY
- At this point, we now have an “Active/Active” cluster because both cluster nodes are actively running resources.



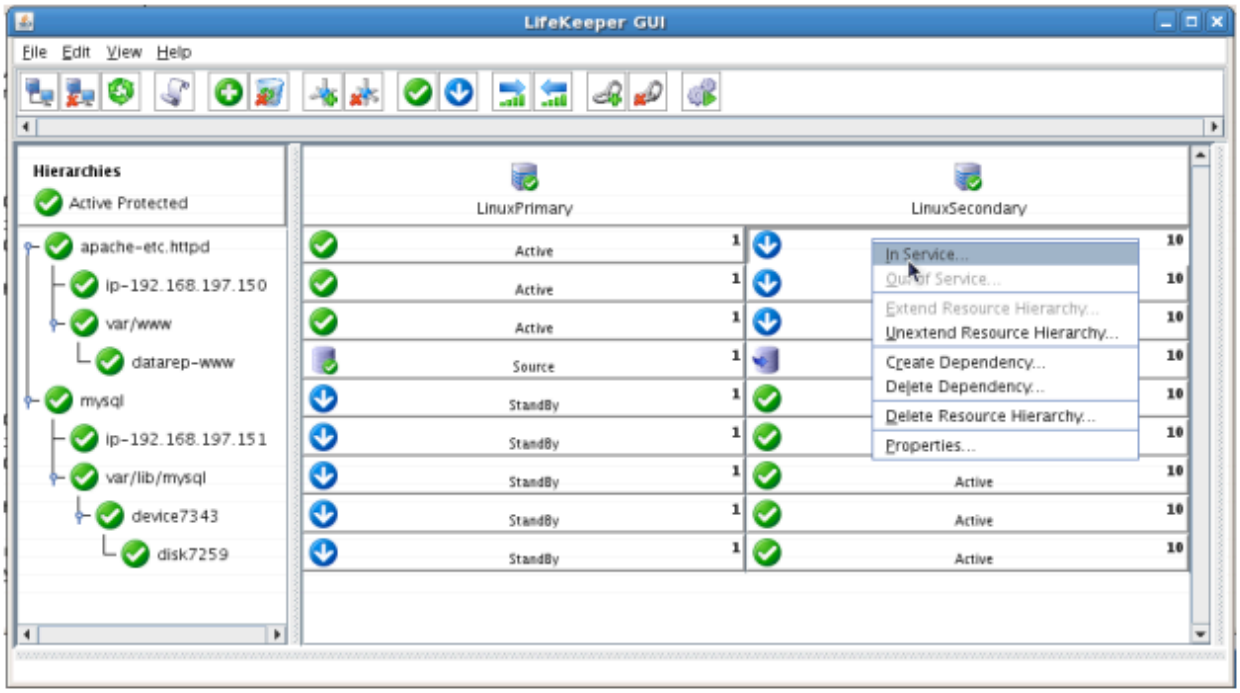

**Tests/Verification:**

- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXSECONDARY.
- Run “ifconfig –a” on LINUXSECONDARY to validate that the IP Address 192.168.197.151 is active on LINUXSECONDARY
- Run “df –h” to verify that the /var/lib/mysql shared iSCSI filesystem is mounted on LINUXSECONDARY
- Verify the MySQL services are running on LINUXSECONDARY by running “ps –ef | grep –i mysql”
- On LINUXSECONDARY run the following command to verify client connectivity to the MySQL database:
  - # mysql –S /var/lib/mysql/mysql.sock –u root –p
  - (enter password “SteelEye”)
- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXPRIMARY, run “mount /dev/sdc1 /var/lib/mysql”. This should FAIL because LINUXPRIMARY does not own the SCSI reservation on this LUN.

# Manual Switchover of the Apache Hierarchy to Secondary Server

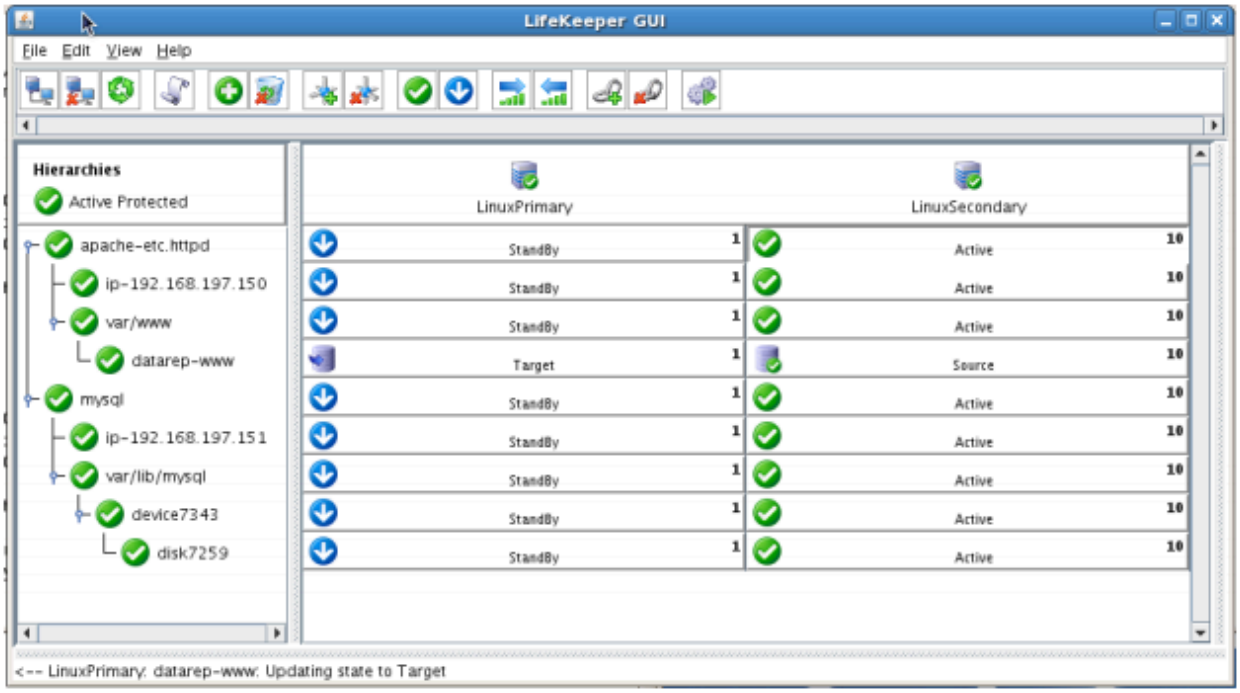
**Procedure:**

- From the LifeKeeper GUI, right click on the Apache resource on the Secondary Server (LINUXSECONDARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



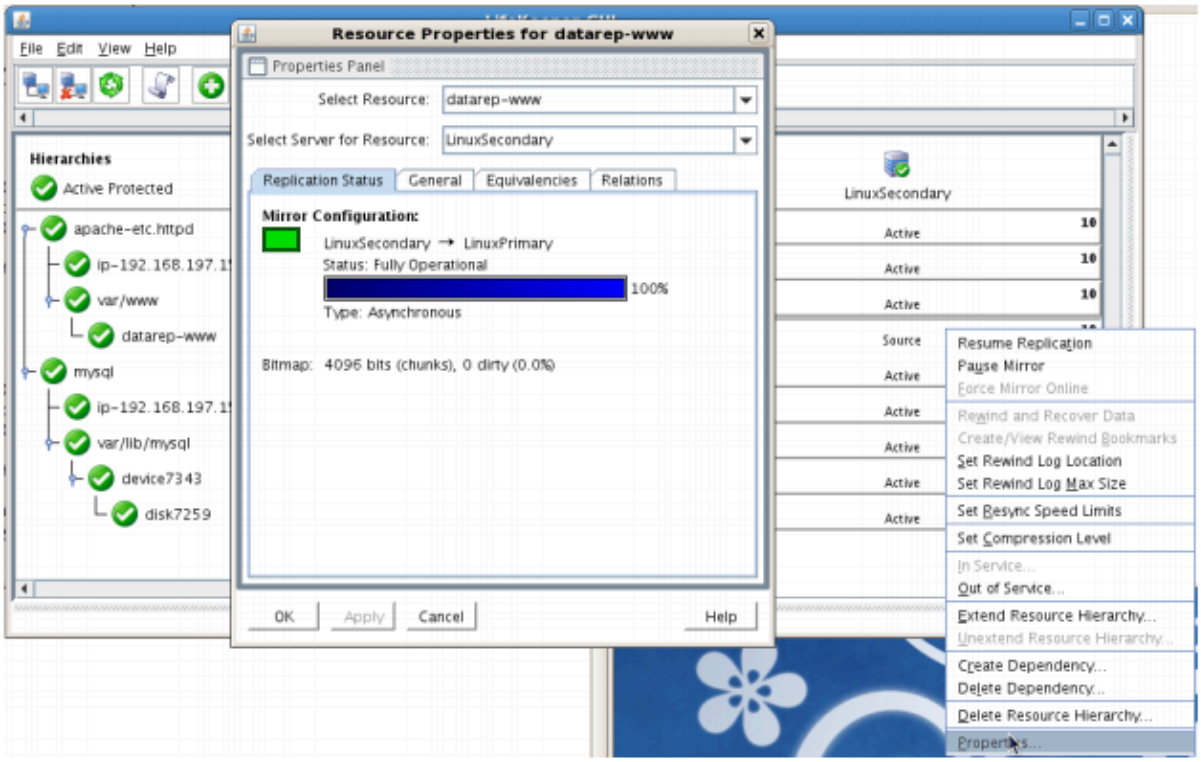

**Expected Result:**

- Beginning with the Apache resource, all resources will be removed from service on the Active Server (LINUXPRIMARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXSECONDARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXSECONDARY -> LINUXPRIMARY
- At this point, we now back to an “Active/Passive” cluster because all services are now actively running on LINUXSECONDARY

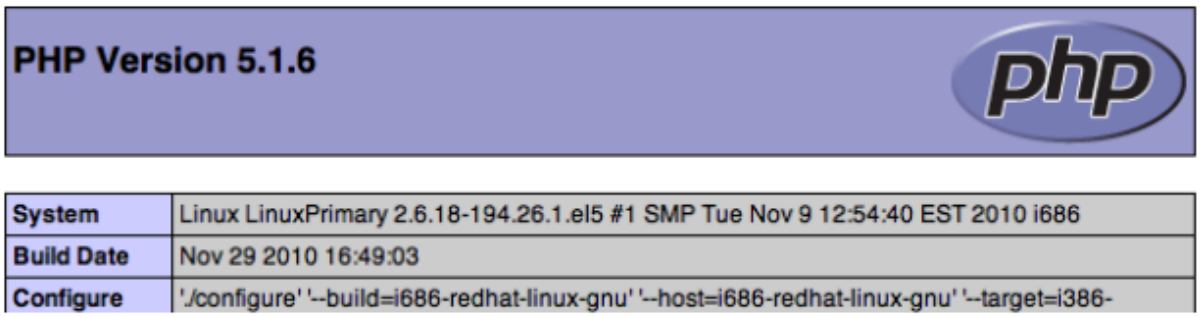


**Tests/Verification:**

- Using the LifeKeeper GUI, verify that the Apache and dependent resources are active on LINUXSECONDARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-www” resource and select Properties




- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.150 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/www replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY
- Verify the Apache services are running on LINUXPRIMARY by running “ps –ef | grep –i httpd”
- Open a Web Browser to http://192.168.197.150 and verify that it can successfully connect. The PHPInfo output should indicate that the system name is “LinuxPrimary



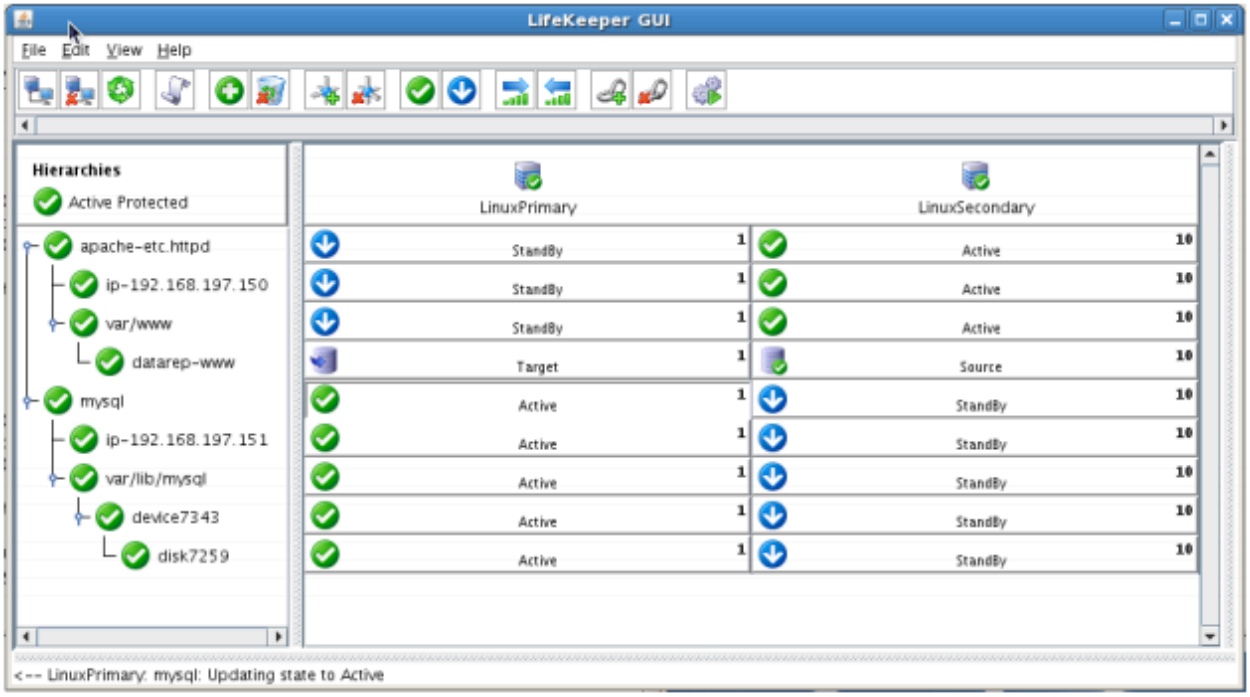
# Manual Switchover of the MySQL Hierarchy back to Primary Server

**Procedure:**

- From the LifeKeeper GUI, right click on the MySQL resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up

**Expected Result:**

- Beginning with the MySQL resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Shared Volume), all resources will be brought in service on LINUXPRIMARY




**Tests/Verification:**

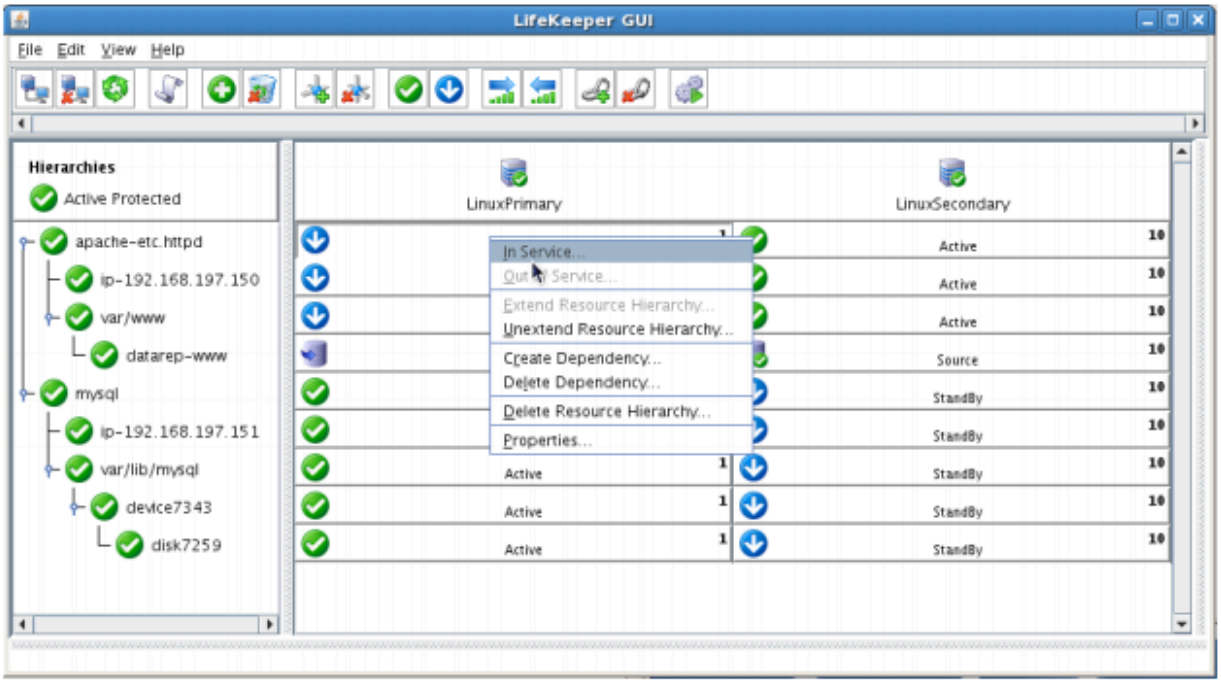
- Using the LifeKeeper GUI, verify that the MySQL and dependent resources are active on LINUXPRIMARY.
- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.151 is active on LINUXPRIMARY
- Run “df –h” to verify that the /var/lib/mysql shared iSCSI filesystem is mounted on LINUXPRIMARY
- Verify the MySQL services are running on LINUXPRIMARY by running “ps –ef | grep –i mysql”
- On LINUXPRIMARY run the following command to verify client connectivity to the MySQL database:
  - # mysql –S /var/lib/mysql/mysql.sock –u root –p

    - (enter password “SteelEye”)
- Verify that the SCSI reservation has properly locked the iSCSI LUN to the currently active cluster node. On LINUXSECONDARY, run “mount /dev/sdc1 /var/lib/mysql”. This should FAIL because LINUXSECONDARY does not own the SCSI reservation on this LUN.

# Manual Switchover of the Apache Hierarchy back to the Primary Server

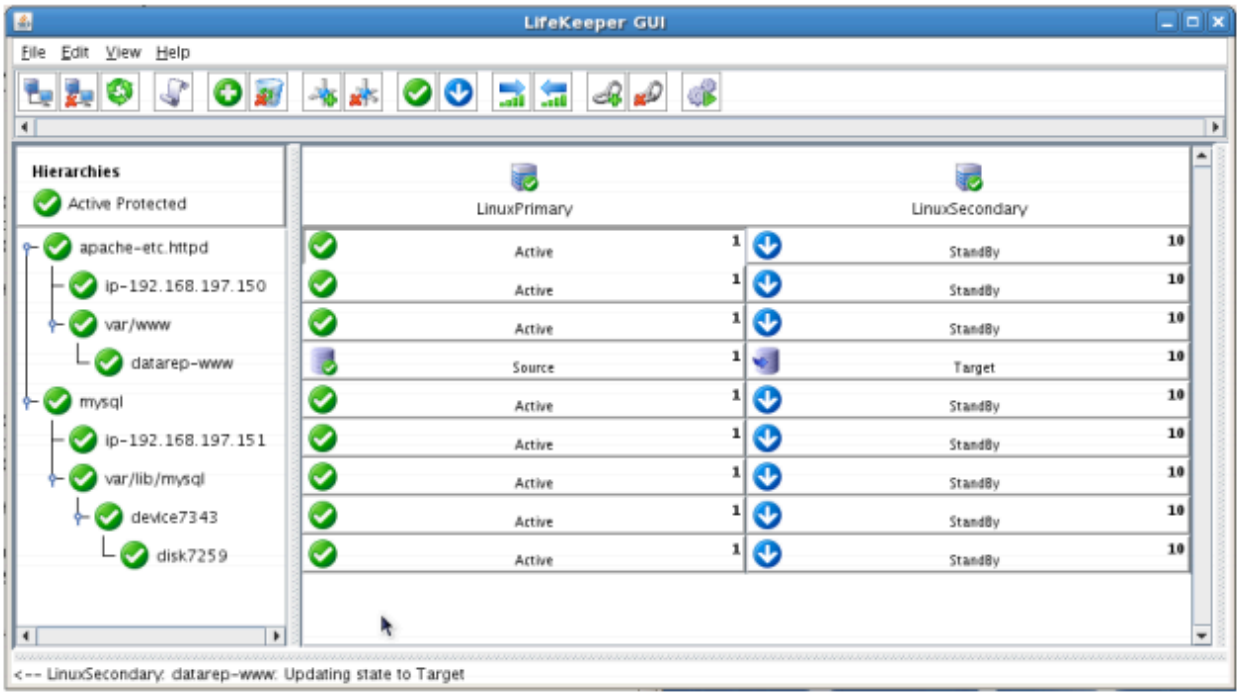
**Procedure:**

- From the LifeKeeper GUI, right click on the Apache resource on the Primary Server (LINUXPRIMARY) and choose IN SERVICE.
- Click “In Service” in the window that pops up



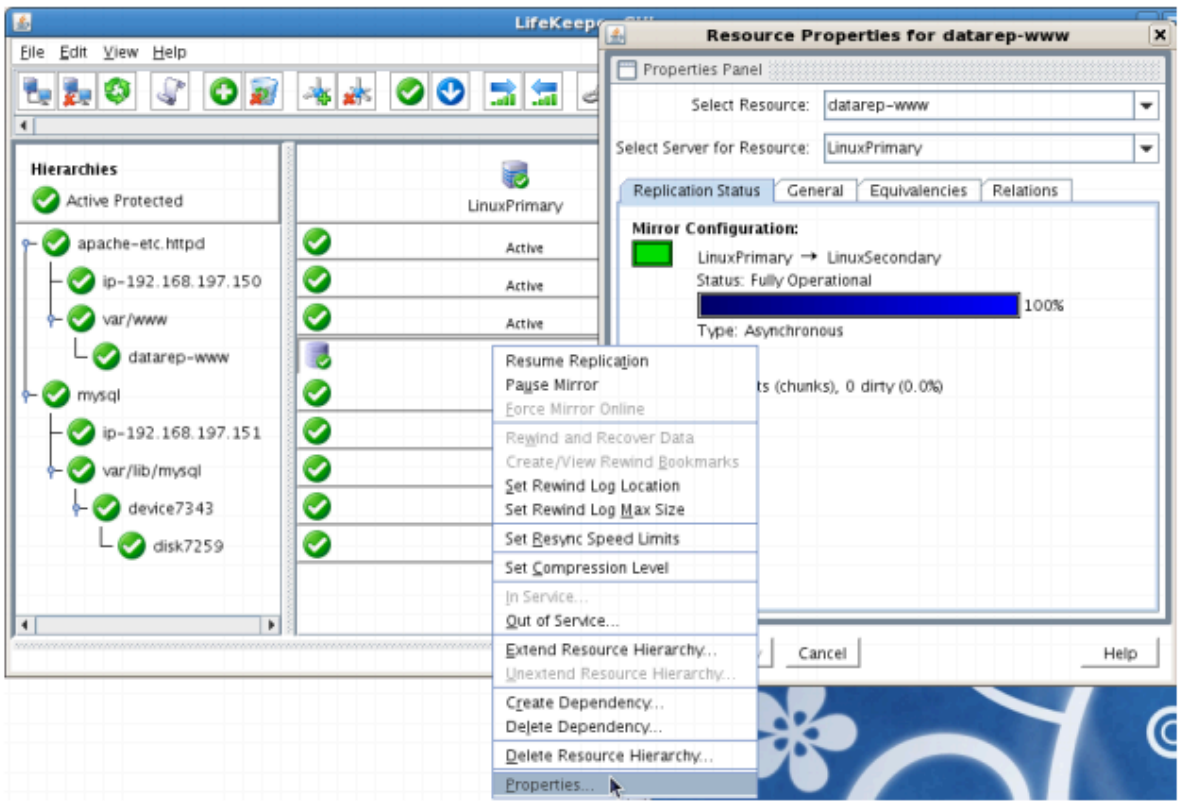

**Expected Result:**

- Beginning with the Apache resource, all resources will be removed from service on the Active Server (LINUXSECONDARY).
- Beginning with the dependent resources (IP and Replicated Volume), all resources will be brought in service on LINUXPRIMARY
- During this process, the direction of the mirror reversed. Data is now transmitting from LINUXPRIMARY -> LINUXSECONDARY




**Tests/Verification:**

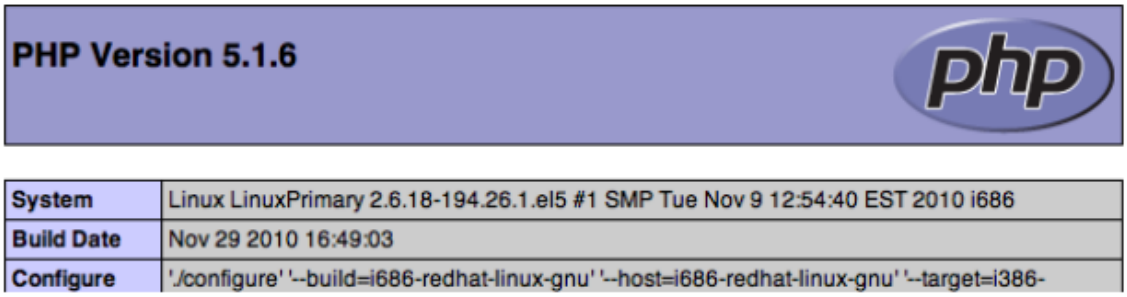
- Using the LifeKeeper GUI, verify that the Apache and dependent resources are active on LINUXSECONDARY.
- Using the LifeKeeper GUI, verify the mirror is now reversed and mirroring in the opposite direction. Right click on the “datarep-www” resource and select Properties




- Run “ifconfig –a” on LINUXPRIMARY to validate that the IP Address 192.168.197.150 is active on

LINUXPRIMARY

- Run “df –h” to verify that the /var/www replicated filesystem is mounted as an “md” device (example: /dev/md0”) on LINUXPRIMARY
- Verify the Apache services are running on LINUXPRIMARY by running “ps –ef | grep –i httpd”
- Open a Web Browser to http://192.168.197.150 and verify that it can successfully connect. The PHPInfo output should indicate that the system name is “LinuxPrimary




## Simulate a network failure on the Primary Server by failing the IP resource

> **IMPORTANT:** Only perform this test if you have more than one communications path configured.

If you perform this test and have only one communications path configured, your system will enter a split-brain scenario as described in the LifeKeeper Administration Guide found here. Refer to this document for more information or contact SIOS presales technical support for assistance in resolving this condition.

**Procedure:**

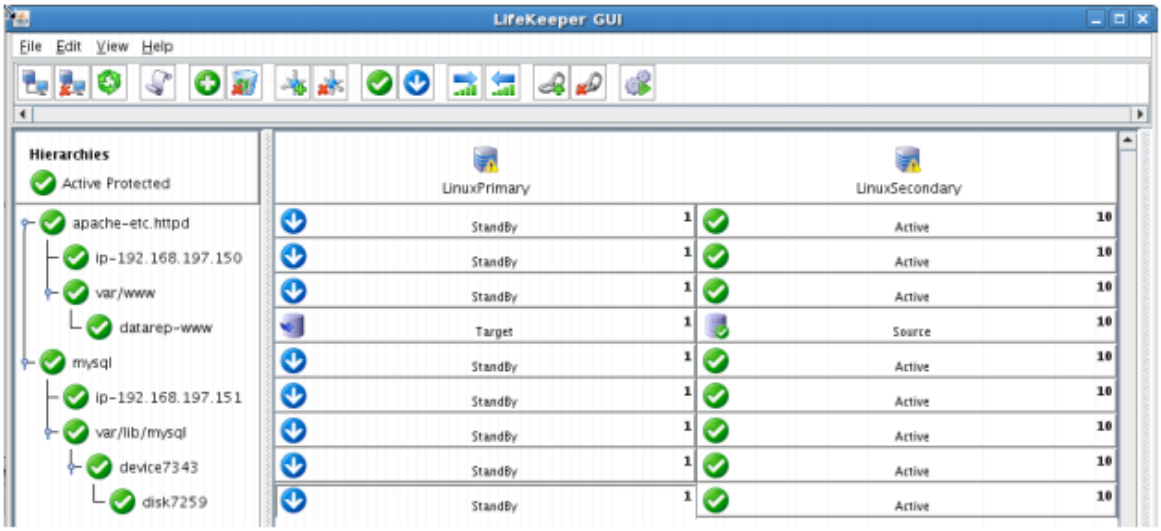
- On LINUXPRIMARY, pull the network cable attached to the NIC that the virtual IP address is configured on

**Expected Result:**

- The IP Resource should fail first.
- The entire hierarchy should failover to LINUXSECONDARY

**Tests/Verification:**

- Check the LifeKeeper Log to verify the IP resource failed – “/opt/LifeKeeper/bin/lk_log log”
- Using the LifeKeeper GUI, verify the MySQL and Apache resource hierarchies fail over successfully to LINUXSECONDARY




## Hard failover of the resource from the Secondary Server back to the Primary Server

**Procedure:**

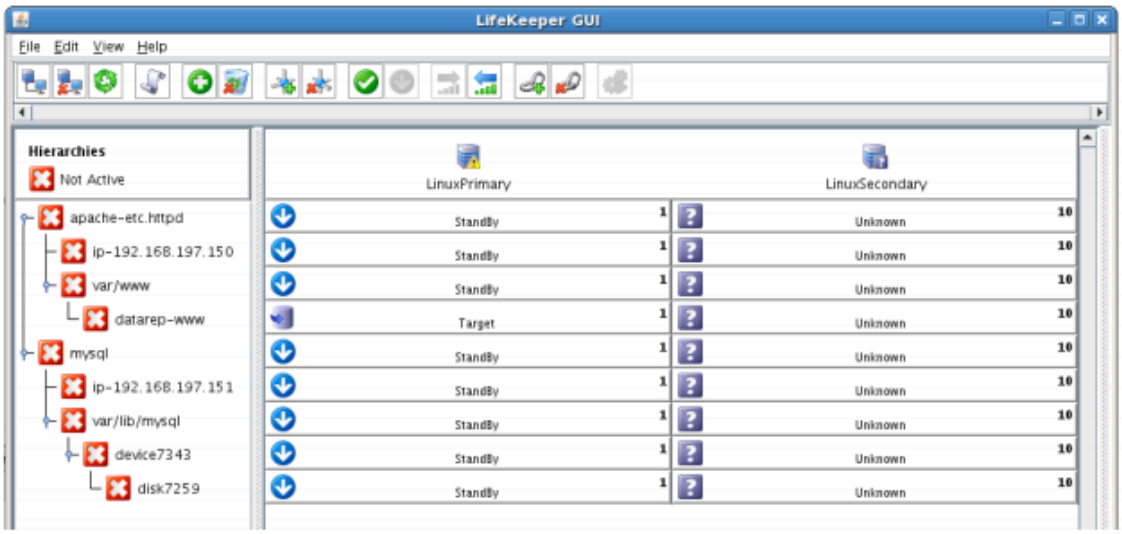
- Pull the power cord on LINUXSECONDARY, as this is the server with all resources currently In Service.

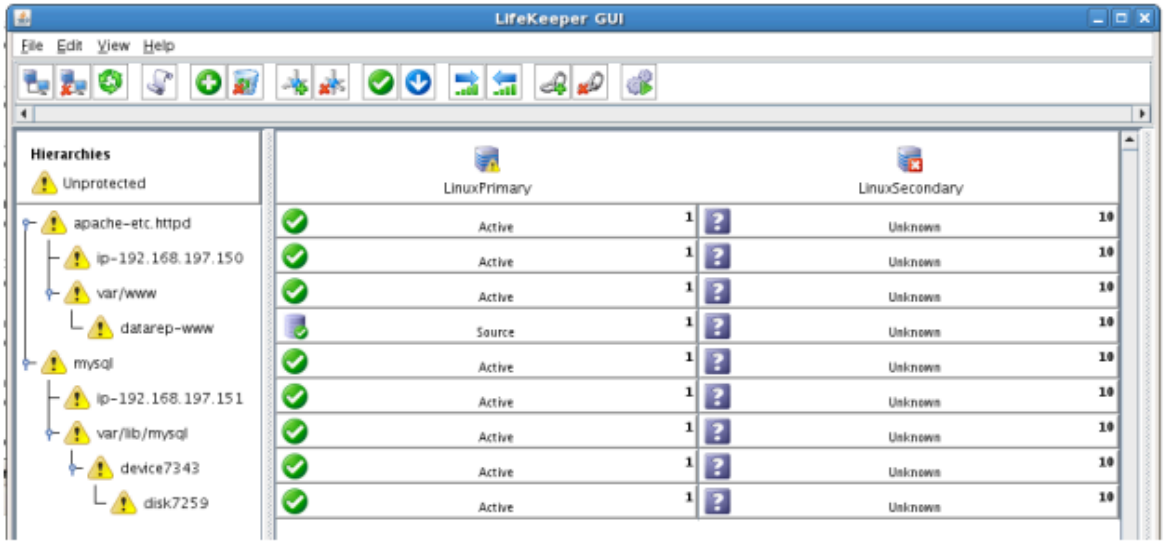
**Expected Result:**

- After failure has been detected, beginning with the dependent resources (IP and Volume), all resources will be brought in service on LINUXPRIMARY.

**Tests/Verification:**

- Using the LifeKeeper GUI, verify the mirror has reversed and is in a Resync Pending state waiting for LINUXSECONDARY to come back on line.
- Verify the PostgreSQL Server services are running on LINUXPRIMARY.
- Verify that the client can still connect to the Webserver and database running on LINUXPRIMARY.
- Verify you can write data to the replicated volume, /var/www on LINUXPRIMARY.




# Bring Failed Server back on line

**Procedure:**

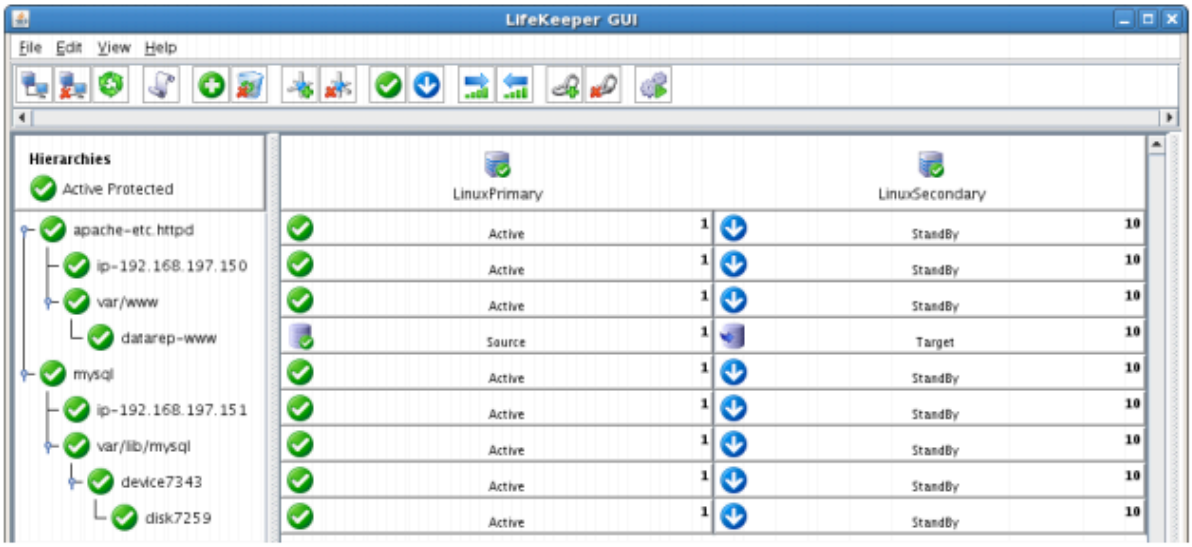
- Plug the power cord back into LINUXSECONDARY and boot it up.

**Expected Result:**

- Using the LifeKeeper GUI, verify that LINUXSECONDARY is coming back up and has become the Standby Server.

**Tests/Verification:**

- Verify the mirror performs a quick partial resync and moves to the Mirroring state
- Verify the Apache and MySQL Hierarchy are in service on LINUXPRIMARY and standby on LINUXSECONDARY.




# Verify Local Recovery of MySQL Server

**Procedure:**

- Kill the PostgreSQL processes via the command line:
  - # ps –ef | grep sql
  - # killall mysqld mysqld_safe
  - run “ps –ef | grep sql once again to verify that the processes no longer exist

**Expected Result: (Assumes Local Recovery for SQL resource is set to YES)**

- The MySQL Server service should stop.
- The MySQL quickcheck process will automatically restart the MySQL Server Service when it runs periodically.
- No failure of MySQL should occur.

**Tests/Verification:**

- Execute “ps –ef | grep sql” once again to verify that the mysql processes have been restored locally on LINUXPRIMARY.
- Verify connectivity to the MySQL database by running:
  - # mysql –S /var/lib/mysql/mysql.sock –u root –p
  - (Enter password “SteelEye”)
- If you inspect the LifeKeeper logs, you will see information indicating that LifeKeeper dectected the failure of the MySQL service and recovered it locally. Run /opt/LifeKeeper/bin/lk_log log for more information.

# 13. LifeKeeper Single Server Protection

LifeKeeper Single Server Protection リリースノート

LifeKeeper Single Server Protection インストレーションガイド

# 13.1. LifeKeeper Single Server Protection リリースノート

### バージョン 9.5.2

**Released 11 August, 2021**

**重要!!**

*本製品をインストールまたは使用する前に、必ずこのドキュメントをお読みください。*
*このドキュメントには、インストール時とその前後に留意すべき重要な項目に関する情報が記載されています。*

## はじめに

このリリースノートの対象読者は、LifeKeeper Single Server Protection for Linux 製品のインストール、設定、管理を行うユーザです。このドキュメントには、LifeKeeper Single Server Protection の正式マニュアルには詳細に記述されていない重要な情報、たとえば、システム要件、新機能、製品の制限へのリンク、トラブルシューティングのヒントなどが記載されています。LifeKeeper Single Server Protection ソフトウェアをインストールして設定する前に、必ずこのドキュメントの内容を確認してください。

## LifeKeeper Single Server Protection の製品説明

LifeKeeper Single Server Protection は、単一ノード構成におけるアプリケーション監視を可能にします (つまり、クラスタの要件または制約はありません)。単一ノード環境は、物理的なものでも仮想 (vSphere、KVM、Amazon EC2) でも構いません。LifeKeeper Single Server Protection は、実績がある安定した SIOS LifeKeeper アーキテクチャ上に構築されます。LifeKeeper Single Server Protection は優れたアプリケーション監視機能を提供し、障害が発生したアプリケーションおよびシステムインフラストラクチャ項目 (例: NFS 共有、IP アドレス、ファイルシステム) のリカバリを実行することができます。何らかの理由でアプリケーションをリカバリできない場合、LifeKeeper Single Server Protection は、システムのリブートまたは VM とアプリケーション監視を設定された VMware 仮想マシンの VMware HA 再起動によって、ノードの再起動を開始します。

## コンポーネント

バンドルされる LifeKeeper Single Server Protection ソフトウェアは、64 ビットシステム (x86_64、AMD64) で動作し、以下のコンポーネントが含まれています。

- LifeKeeper Single Server Protection ソフトウェア

# LifeKeeper Single Server Protection のオプションのリカバリソフトウェア

次のオプションソフトウェアは、記載してあるバージョンのアプリケーション用のリソース定義およびリカバリソフトウェアを提供します。各リカバリソフトウェアに必要な要件は サポートマトリックス や各リカバリソフトウェアの管理ガイドを参照してください。

| パッケージ | パッケージ名 | 保護対象のアプリケーション |
|---|---|---|
| LifeKeeper Apache Web Server Recovery Kit | steeleye-lkAPA-9.5.2-7301.noarch.rpm | Apache Web Server v2.4 |
| LifeKeeper SAP DB / MaxDB Recovery Kit | steeleye-lkSAPDB-9.5.2-7301.noarch.rpm | SAP MaxDB v7.9 |
| LifeKeeper DB2 Recovery Kit | steeleye-lkDB2-9.5.2-7301.noarch.rpm | IBM DB2 Universal Database v10.5、v11.1 および v11.5<br><br>IBM DB2 Enterprise Server Edition (ESE) v10.5、v11.1 および v11.5<br><br>IBM DB2 Workgroup Server Edition (WSE) v10.5、v11.1 および v11.5<br><br>IBM DB2 Express Edition v10.5、v11.1 および v11.5 |
| LifeKeeper Oracle Recovery Kit | steeleye-lkORA-9.5.2-7301.noarch.rpm | Oracle Database Enterprise Edition v12c, v12c R2, v18c および v19c (ASM および pluggable database 機能除く)<br><br>Oracle Database Standard Edition 2 (SE2) v12c, v12c R2, v18c および v19c (ASM および pluggable database 機能除く) |
| LifeKeeper MySQL Recovery Kit | steeleye-lkSQL-9.5.2-7301.noarch.rpm | MySQL および MySQL Enterprise v5.7、v8.0<br><br>MariaDB 10.3、v10.4 および v10.5 |

| | | |
|---|---|---|
| LifeKeeper PostgreSQL Recovery Kit | steeleye-lkPGSQL-9.5.2-7301.noarch.rpm | PostgreSQL v9.6、v10、v11、v12、v13<br><br>EnterpriseDB Postgres Plus Advanced Server/EDB Postgres Advanced Server v9.6、v10.0、v11.0、v12.0、v13 .0 |
| LifeKeeper Sybase ASE Recovery Kit | steeleye-lkSYBASE-9.5.2-7301.noarch.rpm | Sybase ASE 15.7 および 16.0 |
| LifeKeeper Postfix Recovery Kit | steeleye-lkPOSTFIX-9.5.2-7301.noarch.rpm | Postfix ソフトウェアは、それぞれのサーバにサポートされた Linux ディストリビューションをインストールし、設定します。同じバージョンの Postfix が、それぞれのサーバにインストールされる必要があります。 |
| LifeKeeper Samba Recovery Kit | steeleye-lkSMB-9.5.2-7301.noarch.rpm | サポート対象の Linux ディストリビューションに付属の標準 samba ファイルサービス |
| LifeKeeper NFS Server Recovery Kit | steeleye-lkNFS-9.5.2-7301.noarch.rpm | Linux kernel version 2.6 以降<br><br>NFS Server およびクライアントパッケージが SLES システム上にインストールされている必要があります。 |
| LifeKeeper Network Attached Storage Recovery Kit | steeleye-lkNAS-9.5.2-7301.noarch.rpm | NFS サーバまたは NAS デバイス v2、v3、v4 からマウントされた NFS ファイルシステムの NFS バージョン |
| LifeKeeper WebSphere MQ Recovery Kit | steeleye-lkMQS-9.5.2-7301.noarch.rpm | IBM MQ v8.0, v9.0, v9.1 および 9.2<br><br>既知の問題と制限 > インストールを参照してください。 |
| Quick Service Protection | steeleye-lkQSP-9.5.2-7301.noarch.rpm | SPS Quick Service Protection を使用すると、OSサービスの簡易保護機能を提供します |

# LifeKeeper Single Server Protection の機能

| 機能 | 説明 |
|---|---|
| 時間的リカバリロジック | ローカルリカバリの試行回数に制限を設定して、アプリケーションの可用性を向上できます。 |
| マルチレベルポリシー | サーバレベルとリソースレベルでリカバリオプションを指定して、クライアントがアプリケーションごとに最適なリカバリストラテジーを定義することができます。 |
| 通知のみ/メンテナンスモード | ユーザが 1 つ以上のリソースの監視を一時的に無効にして、LifeKeeper Single Server Protection がメンテナンス中のリソースを復旧させないようにすることができます。 |
| VMware vSphere の統合 | VMware の vSphere プラットフォームと統合して、組織がサーバ仮想化と自動化を最大限に活用できるようにしながら、アプリケーションの可用性を向上します(VMware 環境のみ)。 |
| vSphere Client プラグイン | vSphere Client により一元化された管理と監視(VMware 環境のみ)。 |

# LifeKeeper Single Server Protection Version 9 の新機能

| 製品 | 機能 |
|---|---|
| **バージョン 9.5.2 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 8.4 をサポートしました。 |
| | CentOS 8.3 をサポートしました。 |
| | Oracle Linux 8.3 をサポートしました。 |
| | Oracle Linux 7 UEK 6をサポートしました。<br>注意：カーネルバージョンが5.4.17-2102.202.5 以降のみサポートされます。 |
| | Oracle Linux 8 UEK 6をサポートしました。<br>注意：カーネルバージョンが5.4.17-2102.202.5 以降のみサポートされます。 |
| | 以下のARKでもLKCLIを利用できるようになりました。<br>• SAP MaxDB RK<br>• MQ RK |
| | lk group に所属したroot以外のユーザーからでも lkcliを実行できるようになりました。 |
| | lkcli stop でも lkstop 同様に「-i」オプションを使用できるようになりました。 |
| | バグの修正 |
| MySQL | MariaDB 10.5 をサポートしました。 |

| | |
|---|---|
| | バグの修正 |
| MQ | WebSphere MQ 9.2 for RHEL 7.9 をサポートしました。 |
| | WebSphere MQ 9.2 for RHEL 8.3 をサポートしました。 |
| Install、NFS、IP、Oracle、Quorum/Witness、Filesystem、Generic、lksupport | バグの修正 |
| **バージョン 9.5.1 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 7.9 をサポートしました。（2020年12月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | CentOS 7.9 をサポートしました。（2020年12月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | Oracle Linux 7.9 をサポートしました。（2020年12月認定）<br>注意: DataKeeperを使用する場合、LifeKeeperインストール時に特定の手順に従う必要があります。詳細は こちら をご参照ください。 |
| | 以下の ARK でも LKCLI を利用できるようになりました。<br>• DB2 RDBMS RK<br>• RAW RK<br>• Postfix RK<br>• VMDK as Shared Storage RK<br>• SAP ASE RK<br>• Samba RK<br>• HULFT RK / HULFT HUB RK<br>• Oracle RK の Oracle PDB 保護機能 |
| | Red Hat Enterprise Linux 8.3 をサポートしました。（2021年1月認定） |
| | Red Hat Enterprise Linux 8.2 をサポートしました。 |
| | CentOS 8.2 をサポートしました。 |
| | Oracle Linux 8.2（UEK6 は除く）をサポートしました。 |
| | SLES15 SP2 をサポートしました。 |
| | LifeKeeper 内部で使用している OpenSSL パッケージを 1.1.1g に更新しました。 |
| | LifeKeeper 内部で使用している cURL パッケージを 7.68.0 に更新しました。 |
| | **lkstop** コマンドに **-i オプション** を追加しました。 このオプションは保護するリソースを停止せずに LifeKeeper のみを停止します。 また、停止処理を続行するために「yes/no」の確認を行う必要があります。 |
| | バグの修正 |

| | |
|---|---|
| MQ | SIOS Protection Suite for Linux は IBM MQ 9.2 のサポートを開始しました。 (2021年1月) |
| QSP | Apache Tomcat を Quick Service Protection (QSP) RK で保護できるようになりました。 |
| install | LifeKeeper の 構成のみ行うセットアップのモード をサポートしました。その他、セットアップに関する改修を行っています。 |
| | バグの修正 |
| PostgreSQL | PostgreSQL 13 をサポートをサポートしました。（2020年12月認定） |
| | EDB Postgres Advanced Server 13.0 をサポートしました。（2020年12月認定） |
| | FUJITSU Software Enterprise Postgres 12 をサポートしました。（2020年12月認定） |
| SAP MaxDB、DB2、Filesystem、GenericApplication、IP | バグの修正 |
| **バージョン 9.5.0 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 7.8 をサポートしました。（2020年7月認定） |
| | CentOS 7.8 をサポートしました。（2020年7月認定） |
| | Oracle Linux 7.8 をサポートしました。（2020年7月認定） |
| | SUSE Linux Enterprise Server 12 SP5 をサポートしました。（2020年7月認定） |
| | VMware vSphere 7.0 をサポートしました。（2020年7月認定） |
| | CentOS 8.0 をサポートしました。 |
| | Oracle Linux 8.0 をサポートしました。 |
| | Red Hat Enterprise Linux 8.1 をサポートしました。 |
| | CentOS 8.1 をサポートしました。 |
| | Oracle Linux 8.1 をサポートしました。 |
| | CLI が強化され、CLI で LifeKeeper を制御できるようになりました。詳細は LKCLI を参照してください。 |
| | バグの修正 |
| PostgreSQL | EDB Postgres Advanced Server 12.0 をサポートしました。（2020年7月認定） |
| | PostgreSQL 12 をサポートしました。 |
| LifeKeeper Core, Filesystem, NFS, DB2, MaxDB, Sybase ASE | バグの修正 |

| バージョン **9.4.1** の新機能 | |
|---|---|
| LifeKeeper Core | OS同梱のOpenJDKをインストールするようになりました。詳細につきましてはテクニカルドキュメンテーションの LifeKeeper の GUI の設定 のページを参照してください。 |
| | SUSE Linux Enterprise Server 15 SP1 をサポートしました。 |
| | Oracle Linux 7.7 をサポートしました。 |
| | CentOS 7.7 をサポートしました。 |
| | AWS Nitro systemをサポートしました。 |
| | AWS Transit Gatewayをサポートしました。 |
| | バグの修正 |
| Install、IP、MaxDB | バグの修正 |
| バージョン **9.4** の新機能 | |
| LifeKeeper Core | Red Hat Enterprise Linux 8.0 をサポートしました。<br>注意：RHEL7 から RHEL8 のようにカーネルをメジャーバージョンアップすることはサポートされません。 |
| | Red Hat Enterprise Linux 7.7 をサポートしました。（2019年11月認定） |
| | Oracle Linux 7 UEK5をサポートしました。 |
| | バグの修正 |
| MySQL | MariaDB 10.3 をサポートしました。 |
| DB2 | DB2 11.5 をサポートしました。 |
| Oracle、Install | バグの修正 |
| バージョン **9.3.2** の新機能 | |
| LifeKeeper Core | Red Hat Enterprise Linux 7.6 をサポートしました。 |
| | CentOS 7.6 をサポートしました。 |
| | Oracle Linux 7.6 をサポートしました。 |
| | SUSE Linux Enterprise Server 12 SP4 をサポートしました。 |
| | SUSE Linux Enterprise Server 15 をサポートしました。 |
| Install | 現在の setup コンフィグレーションを保存するための -s オプションを setup コマンドに追加しました。 |
| PostgreSQL | PostgreSQL 11 をサポートしました。 |
| | EDB Postgres Advanced Server v11 をサポートしました。 |
| MQ | IBM MQ v9.1 をサポートしました。 |
| Oracle | Oracle 19c をサポートしました。 (2019年8月に認定) |

| | |
|---|---|
| Core, Install, Filesystem, NFS, PostgreSQL, Generic, Apache, GUI, lksupport | バグの修正 |
| **バージョン 9.3.1 の新機能** | |
| LifeKeeper Core | OpenSSLパッケージを 1.0.2p に更新しました。 |
| | Red Hat Enterprise Linux 6.10 をサポートしました。 |
| | CentOS 6.10 をサポートしました。 |
| | Oracle Linux 6 Update 10 をサポートしました。 |
| MySQL | MySQL 8.0 をサポートしました。 |
| Oracle | Oracle 18c をサポートしました。 (2019年3月に認定) |
| Install | バグの修正 |
| **バージョン 9.3 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux Version 7.5 をサポートしました。 |
| | Oracle Linux Version 7.5 をサポートしました。 |
| | CentOS7.5 をサポートしました。 |
| | VMware vSphere 6.7 をサポートしました。（2018年10月認定） |
| | バグの修正 |
| Install | インストールスクリプトが刷新されました。詳細はこちら をご参照ください。 |
| Oracle, Samba, MQ, Sybase, Filesystem, Generic Application, QSP | バグの修正 |
| **バージョン 9.2.2 の新機能** | |
| PostgreSQL | PostgreSQL 10 をサポートしました |
| | EDB Postgres Advanced Server v10.0をサポートしました（2018年4月に認定） |
| NAS | バグの修正 |
| **バージョン 9.2.1 の新機能** | |
| LifeKeeper Core | Oracle Linux 7.4 をサポートしました。 |
| | CentOS 7.4 をサポートしました |
| | SUSE Linux Enterprise Server 12 SP3 をサポートしました |

| | |
|---|---|
| | ※ SUSE Linux Enterprise Server 12 SP3 を使用する場合は、カーネルを 4.4.82-6.9.1 にアップグレードしてください |
| | バグの修正 |
| PostgreSQL | EDB Postgres Advanced Server 9.6 をサポートしました |
| MQ | IBM MQ 9.0 をサポートしました |
| **バージョン 9.2 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 7.4 をサポートしました |
| | SNMP トラップを複数のターゲットに送信することができるようになりました |
| | 仮想化環境 Nutanix Acropolis Hypervisor をサポートしました。（SPSは非サポート） |
| | バグの修正 |
| IP | 実 IP （NIC に設定されたプライマリ IP アドレス） を用いた IP リソースを作成できるようになりました |
| PostgreSQL | PostgreSQL 9.6をサポートしました |
| MQ | IBM MQ 9.0 をサポートしました (2017 年 12 月に認定) |
| SAP MaxDB, Install | バグの修正 |
| **バージョン 9.1.2 の新機能** | |
| LifeKeeper Core | SUSE Linux Enterprise Server 12 SP2 をサポートしました。 |
| | CentOS7.3をサポートしました。 |
| | Red Hat Enterprise Linux Version 6.9をサポートしました。 |
| | Oracle Linux Version 7.3の UEK カーネルをサポートしました。 |
| | バグの修正 |
| Oracle | Oracle 12c R2をサポートしました。 |
| DB2 | DB2 11.1をサポートしました。 |
| IP, QSP, MySQL, NFS | バグの修正 |
| **バージョン 9.1.1 の新機能** | |
| LifeKeeper Core | SUSE Linux Enterprise Server 12 SP1 のサポート<br>※ SP1 を適用していない SLES12 の利用はサポートされません。<br>※ Btrfs の利用はサポートされません。 |
| | Red Hat Enterprise Linux Version 7.3 のサポート |
| | Oracle Linux Version 7.3 のサポート |

| | |
|---|---|
| | ※ UEK の利用はサポートされません。 |
| | vSphere6.5 のサポート（SMC 機能は vSphere6.5 ではサポートされません） |
| | バグの修正 |
| PostgreSQL | PostgreSQL 9.5 のサポート<br>EDB Postgres Advanced Server v9.5 のサポート<br>詳細はオプションの SPS リカバリソフトウェアの要件 、 PostgreSQL Recovery Kit 管理ガイド > 管理 をご参照ください。 |
| Sybase ASE | Sybase ASE 16.0 のサポート |
| MySQL | RHEL 7.x/CentOS 7.x/OEL 7.x での MySQL 5.7 のサポート<br>※ 上記以外の環境での　MySQL 5.7 は既にサポートされています。 |
| **バージョン 9.1.0 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux 6.8のサポート(2016年9月に認定)<br>CentOS 6.8, Oracle Linux 6.8についてもサポートいたします(2016年9月に認定)。 |
| | LifeKeeper API for Monitoringのサポート<br>LifeKeeperのステータスやログの情報を提供するAPIを設けました。 |
| | Quick Service Protectionのサポート<br>OSサービスの簡易保護機能を提供します。 |
| **バージョン 9.0.2 の新機能** | |
| LifeKeeper Core | Red Hat Enterprise Linux Version 7.2のサポート<br>※MySQL RKはRHEL 7.x/CentOS 7.x/OEL 7.xをサポートしていません。<br>※アプリケーション側のRHEL 7.2対応状況はユーザー様にてご確認ください |
| | OpenSSLパッケージを1.0.1qに更新 |
| | バグの修正 |
| MQ | WebSphere MQ – マルチバージョンの WebSphere MQ のサポートが追加されました。本サポートにより、バージョン 7.1、7.5、および 8.x のキューマネージャのすべてを同クラスタノードで保護できるようになりました。 |
| | MQコマンドをmqmグループのユーザが代替で実行できる機能を追加 |
| | バグの修正 |
| IP, Filesystem, PostgreSQL, SAP DB/MaxDB, Oracle | バグの修正 |
| Licensing | FlexNetパッケージの更新 |
| **バージョン 9.0.1 の新機能** | |
| LifeKeeper Core | バグの修正 |
| DataKeeper | バグの修正 |

| バージョン 9.0 の新機能 | |
|---|---|
| LifeKeeper Core | パラメーター一覧 のドキュメントを統合し、lkchkconf コマンド を追加 |
| | vSphere 6のサポート（SMC機能はvSphere6ではサポートされません） |
| | reiserfsファイルシステムのサポートを廃止 |
| | Red Hat Enterprise Linux Version 7.0/7.1、Community ENTerprise Operating System (CentOS) Version 7.0/7.1、Oracle Linux Version 7.0/7.1でサポートされるARKは、LifeKeeper for Linux v8.4.1と同じです (対象ARK：PostgreSQL, MySQL, Oracle, DB2, Apache, Postfix, NFS, NAS, Samba) |
| | バグの修正 |
| GUI | JRE 8u51のサポート（JRE 7はサポートされません） |
| | Chromeサポート廃止 |
| | バグの修正 |
| FileSystem, PostgreSQL | バグの修正 |

# バグの修正

下記は、最新のバグの修正および拡張機能のリストです。

| Bug | Description |
|---|---|
| PL-3822 | サポートされていないカーネルバージョンにアップグレードした後にLifeKeeperのアップグレードを開始しようとした時に、アップグレードを中止するようにしました。 |
| PL-4122 | quickCheckの失敗やrecoverの際に取得するログが多くなる問題を修正しました。 |
| PL-4318 | システムの応答に時間がかかるときにログメッセージを出力するようにしました。 |
| PL-4769 | setupスクリプトでパッケージのアンインストールを行う際に使用中のリソースに対して警告を出すように修正しました。 |
| PL-5163 | lkcli resource infoコマンドの終了ステータスがエラー時にも0になることがある問題を修正しました。 |
| PL-5402 | lksupportでgrub2の情報を取得できるようにしました。 |
| PL-5415 | quickCheck実行前にrecoverスクリプトが実行されているか確認するように修正しました。 |
| PL-5594 | 一部環境であったLifeKeeperとOS同梱のOpenJDKで発生するメモリ管理不整合の問題を修正しました。 |
| PL-5596 | lkcli importコマンドでリソースの依存関係が正しく構築されないことがある問題を修正しました。 |

| | |
|---|---|
| PL-6608 | Filesystemリソースをimportする際にdiskをマウントするように修正しました。 |
| PL-6616 | システムのシャットダウン時にlifekeeper.serviceとmultipathd.serviceが競合することがある問題を修正しました。 |
| PL-6691 | Oracleのコマンドを実行時にoracle userにセットされている環境変数PERL5LIBをリセットするようにしました。 |
| PL-6780 | lkscsidがメモリ/スレッドリークしてしまう問題を修正しました。 |
| PL-7066 | リソースの拡張時に生じるpriorityの問題を修正しました。 |
| PL-7185 | IPv6でスイッチオーバー時にルートテーブルを削除してしまうことがある問題を修正しました。 |
| PL-7308 | lkcliのconfigコマンドにおいてallオプションが必須になっている問題を修正しました。 |
| PL-7309 | lkcliのgenappにあるconfigコマンドで追加したスクリプトに実行権限が付与されない問題を修正しました。 |
| PL-7310 | lkcliのcleanコマンドで一部のリソースが削除されないことがある問題を修正しました。 |
| PL-7348 | SSP上でsystemd管理のmysqlを保護するリソースが正しく起動しないバグを修正しました。 |
| PL-7363 | パラメータWAITFORLCMのデフォルト値を20から30に変更しました。 |
| PL-7858 | IPリソースのrestoreまたはrecover時に物理IPが0.0.0.2に変わることがある問題を修正しました。 |
| PL-8035 | LKセットアップ時にオプションとしてインストール対象となるHADRをファイル指定できるように改修しました。 |
| PL-8152 | NFSリソースにおいて、LKCLI importが成功しない問題を修正しました。 |
| PL-8971 | IPリソースのネットマスクが32bitで設定できるようにしました。 |
| PL-9012 | LKCLI importにおける、deviceのvalidationの問題を修正しました。 |
| PL-9033 | NFSリソースを作成する際、依存関係を作成するIPリソースがプライマリとなっているサーバー上でなければならない問題を修正しました。 |
| PL-9036 | LKCLIのdeleteコマンドでhanfsリソースが残ることがある問題を修正しました。 |
| PL-9056 | ディスクリソースのIDが不適切に設定されることがある問題を修正しました。 |
| PL-9900 | SLESとUEKでインストールの際にSecureBootが有効になっていた場合、警告を出すようにしました。 |

## SIOS Protection Suite for Linux Version 9 で廃止された機能

| 機能 | 説明 |
|---|---|
| 本リリースで廃止になった機能 | |
| Core | SteelEye 管理コンソールはサポートされなくなりました。 |

| | |
|---|---|
| | Red Hat Enterprise Linux 6 はサポートされなくなりました。 |
| | CentOS 6 はサポートされなくなりました。 |
| | Oracle Linux 6 はサポートされなくなりました。 |
| Oracle | Oracle virtual machine (OVM) 3.4.6はサポートされなくなりました。 |
| PostgreSQL | PostgreSQL 9.5はサポートされなくなりました。 |
| | EnterpriseDB Postgres Plus Advanced Server 9.5はサポートされなくなりました。 |
| VMware | VMware vSphere 5.5 , 6.0はサポートされなくなりました。 |
| **バージョン 9.5.1 で廃止になった機能** | |
| | ありません |
| **バージョン 9.5.0 で廃止になった機能** | |
| LifeKeeper Core | syslog-ng を用いた system ログ管理はサポートされなくなりました。rsyslog を使用してください。 |
| | SUSE Linux Enterprise Server (SLES) 11.0 ～ SP4 はサポートされなくなりました。 |
| Oracle | Oracle Database Enterprise Edition 11g R2 はサポートされなくなりました。 |
| | Oracle Database Standard Edition 11g R2 はサポートされなくなりました。 |
| | Oracle Database Standard Edition One 11g R2 はサポートされなくなりました。 |
| MySQL | MariaDB 5.5, 10.0 はサポートされなくなりました。 |
| PostgreSQL | PostgreSQL 9.4 はサポートされなくなりました。 |
| | EnterpriseDB Postgres Plus Advanced Server 9.4 はサポートされなくなりました。 |
| **バージョン 9.4.1 で廃止になった機能** | |
| | ありません |

# システム要件

## LifeKeeper Single Server Protection の製品要件

LifeKeeper Single Server Protection は、下表に示す最低要件を満たすすべての Linux プラットフォームでサポートされます。

| 説明 | 要件 |
|---|---|
| Linux オペレーティングシステム | 個々のオペレーティングシステム情報については、「Linux Configuration Table」(カーネルセクションのみ) を参照してください。 |

| | |
|---|---|
| 仮想環境 | 仮想マシン内で起動するゲスト OS が Linux Configuration Table に記載されているサポート対象のバージョンのうちの 1 つである必要があります。以下の仮想環境は SIOS Protection Suite for Linux が展開されている場合の例です。<br>サポート対象となる仮想化環境の詳細なバージョンについてはサポートマトリックス を参照ください。LifeKeeper Single Server Protection との組み合わせで利用可能なクラウド環境については こちら を参照してください。<br>• KVM<br>• Oracle VM Server for x86<br>• VMware vSphere v6.5、v6.7、v7.0<br>• Amazon EC2<br>• Nutanix Acropolis Hypervisor<br>VMware vSphere v6.5 以降ではvSAN構成についてもサポートします。ただし RDM はvSAN で非サポートとなるため利用できません。<br>ファイバーチャンネル SAN および共有 SCSI クラスタ設定は、KVM および Oracle VM Server for x86 仮想マシン上で動作する LifeKeeper Single Server Protection for Linux をサポートしません。<br>注記： AWSやAzure環境で動作するSLESにおいて、クラウドネットワークプラグインによって仮想 IP アドレスの動的な変更が行われることによりLifeKeeperクラスタの動作に影響を与えることがあります。詳細や回避策は トラブルシューティング > 既知の問題と制限 > LifeKeeper Core をご確認ください。 |
| メモリ | LifeKeeper Single Server Protection を実行するシステムの最小メモリ要件は 512MB です。これは LifeKeeper Single Server Protection がサポートする Linux ディストリビューションが必要とする最低限の容量です。システムのメモリは LifeKeeper Single Server Protection が保護するシステム上で動作するアプリケーションに対してサイジングする必要があります。 |
| ディスク容量 | LifeKeeper Single Server Protection Cluster に必要なディスク容量は次のとおりです。<br>• /opt – 約 100MB (インストールするキットに依存します)<br>• / – 約 110MB |
| Java Runtime Environment | • OpenJDK 1.8、10以降 |

## LifeKeeper Single Server Protection サポートソフトウェアの

# 要件

下表のサポートソフトウェアは、VM とアプリケーション監視を設定された VMware VM でのみ必要です。

| 製品 | 要件 | ディスク容量要件 |
|---|---|---|
| VMware | VMware vSphere Client (LifeKeeper Single Server Protection vSphere Client プラグイン機能用)<br><br>保護されるすべての仮想マシンに VMware Tools がインストールされ、実行されている<br><br>VMware アプリケーション HA 監視が有効で、保護されるすべての仮想マシンに対して VM とアプリケーション監視が設定されている | */opt* で約 175 KB (VMware Tools の場合) |

# クライアントのプラットフォームとブラウザ

LifeKeeper Single Server Protection web クライアントは、Java Runtime 環境　JRE 8 update 51 をサポートするすべてのプラットフォームで動作します。現在動作が確認されている環境は、JRE 8 update 51 を使用した Linux、Windows Server 2008 R2、 Windows Server 2012 R2、Windows Server 2016、Windows 7、Windows8、Windows 10 上の Firefox (Firefox 51 まで)、および Internet Explorer です。 その他の最近のプラットフォームやブラウザも、SPS web クライアントが動作する可能性がありますが、SIOS Technology Corp では、それらの環境でのテストをしていません。また、各ブラウザ固有の機能についても、テストしていません。

LifeKeeper Single Server Protection コンポーネントおよび保護される Linux ゲストの IP アドレスは、DNS またはローカルの hosts ファイル (通常、/etc/hosts または C:\windows\system32\drivers\etc\hosts) によって解決可能でなければなりません。ローカルの hosts ファイルを使用すると、クライアントの接続時間が最小になり、DNS 停止時であっても接続が可能になります。

# 既知の問題

既知の問題、回避策、およびその他のトラブルシューティング情報については、LifeKeeper Single Server Protection for Linux テクニカルドキュメンテーション の 既知の問題と回避策 セクションを参照してください。

**商標について:**
「Amazon Web Services」、「Powered by Amazon Web Services」のロゴ、「AWS」、「Amazon EC2」、「EC2」、「Amazon Elastic Compute Cloud」、「Amazon Route 53」、「Amazon Virtual Private Cloud」および「Amazon VPC」は、米国その他の国における Amazon.com, Inc. またはその関連会社の商標です。なお、本文および図表中では、「™」、「®」は明記しておりません。

# 13.2. LifeKeeper Single Server Protection インストレーションガイド

## LifeKeeper Single Server Protection for Linux について

LifeKeeper Single Server Protection (SSP) は、単一ノード構成におけるアプリケーション監視を可能にします (つまり、クラスタの要件または制約はありません)。単一ノード環境は、物理的なものでも仮想 (vSphere、KVM) でも構いません。LifeKeeper SSP は、実績がある安定した SIOS LifeKeeper アーキテクチャ上に構築されます。LifeKeeper SSP は優れたアプリケーション監視機能を提供し、障害が発生したアプリケーションおよびシステムインフラストラクチャ項目 (例: NFS 共有、IP アドレス、ファイルシステム) のリカバリを実行することができます。何らかの理由でアプリケーションをリカバリできない場合、LifeKeeper SSP は、システムのリブートまたは VM とアプリケーション監視を設定された VMware 仮想マシンの VMware HA 再起動によって、ノードの再起動を開始します。

**注記:** LifeKeeper SSP は SIOS LifeKeeper 技術を使用して構築されているため、ドキュメント全体で LifeKeeper を参照します。また、両製品に共通するトピックについては SIOS Protection Suite for Linux ドキュメンテーションの情報を参照します。これらの共通のトピックを参照する場合、LifeKeeper SSP には以下の話題は適用されません。また、サポートもされません。

- クラスタリング
- コミュニケーションパス
- 共有ストレージ (要件、構成、...)
- リソース階層の拡張/拡張解除
- Quorum/Witness
- アプリケーションリカバリーキット
  - Recovery Kit for EC2
  - LVM Recovery Kit
  - MD Recovery Kit
  - Route53 Recovery Kit
  - SAP Recovery Kit
  - SAP HANA Recovery Kit
  - VMDK as Shared Storage Recovery Kit
- ストレージキット (DR、DMMP、HDLM、LVM、MD、PPATH、NEC SPS)

  **注記:** LifeKeeper SSP にベースとなるストレージキットがない場合、保護されるファイルシステムのマウントに必要なデバイスがシステム起動時にアクティベートされるようにするための手順が必要です (例: ファイルシステムが論理ボリューム上でマウントされる場合、LifeKeeper SSP が起動する前にボリュームがアクティブな状態になっていなければなりません)。
- リソース/マシンのフェイルオーバ (LifeKeeper SSP のデフォルトでは、これによってノードが再起

動されます)

- リソースのスイッチオーバ

- 切り替え可能な IP アドレス (LifeKeeper SSP では、保護されるアプリケーションの一部には切り替え可能な IP アドレスが必要ですが、単一ノードしかないため、実際には切り替えは行われません)

LifeKeeper SSP のベースになっている SIOS LifeKeeper 製品の詳細については、共通するリリース番号の SIOS Protection Suite for Linux ドキュメンテーション を参照してください。このドキュメンテーションは、リソース階層の作成、リソースタイプ、状態と関係、グラフィカルユーザインターフェース (GUI)、および共通の作業と高度な作業に関する詳細情報を提供します。

# 13.2.1. はじめに

## LifeKeeper Single Server Protection for Linux について

LifeKeeper Single Server Protection (SSP) は、単一ノード構成におけるアプリケーション監視を可能にします (つまり、クラスタの要件または制約はありません)。単一ノード環境は、物理的なものでも仮想 (vSphere、KVM) でも構いません。LifeKeeper SSP は、実績がある安定した SIOS LifeKeeper アーキテクチャ上に構築されます。LifeKeeper SSP は優れたアプリケーション監視機能を提供し、障害が発生したアプリケーションおよびシステムインフラストラクチャ項目 (例: NFS 共有、IP アドレス、ファイルシステム) のリカバリを実行することができます。何らかの理由でアプリケーションをリカバリできない場合、LifeKeeper SSP は、システムのリブートまたは VM とアプリケーション監視を設定された VMware 仮想マシンの VMware HA 再起動によって、ノードの再起動を開始します。

> 注記: LifeKeeper SSP は SIOS LifeKeeper 技術を使用して構築されているため、ドキュメント全体で LifeKeeper を参照します。また、両製品に共通するトピックについては SIOS Protection Suite for Linux ドキュメンテーションの情報を参照します。これらの共通のトピックを参照する場合、LifeKeeper SSP には以下の話題は適用されません。

- クラスタリング
- コミュニケーションパス
- 共有ストレージ (要件、構成、...)
- リソース階層の拡張/拡張解除
- ストレージキット (DR、DMMP、HDLM、LVM、MD、PPATH、NEC SPS)

> 注記: LifeKeeper SSP にベースとなるストレージキットがない場合、保護されるファイルシステムのマウントに必要なデバイスがシステム起動時にアクティベートされるようにするための手順が必要です (例: ファイルシステムが論理ボリューム上でマウントされる場合、LifeKeeper SSP が起動する前にボリュームがアクティブな状態になっていなければなりません)。

- リソース/マシンのフェイルオーバ (LifeKeeper SSP のデフォルトでは、これによってノードが再起動されます)
- リソースのスイッチオーバ
- 切り替え可能な IP アドレス (LifeKeeper SSP では、保護されるアプリケーションの一部には切り替え可能な IP アドレスが必要ですが、単一ノードしかないため、実際には切り替えは行われません)

> 注記: Amazon EC2上で使用する場合、IP リソースを作成する前に Amazon EC2マネジメ

> ントコンソールを使用して NIC にセカンダリプライベート IP アドレスを割り当ててください。次に、LifeKeeper GUI クライアントを使用している NIC でプライベート IP アドレスとして IP リソースを作成してください。Elastic IP を IP リソースに関連付けることもできます。

LifeKeeper SSP のベースになっている SIOS LifeKeeper 製品の詳細については、共通するリリース番号の SIOS Protection Suite for Linux ドキュメンテーション を参照してください。このドキュメンテーションは、リソース階層の作成、リソースタイプ、状態と関係、グラフィカルユーザインターフェース (GUI)、および共通の作業と高度な作業に関する詳細情報を提供します。

# 13.2.2. LifeKeeper Single Server Protection ソフトウェアのインストール

本ページでは Linux OS について基本的な知識がある方にむけて LifeKeeper Single Server Protection Software (SSP) のインストールについて説明しています。その他詳細については LifeKeeper Single Server Protection テクニカルドキュメンテーション を参照してください。

## インストール前の要件

SSP for Linux をインストールする前に、下記の要件について確認してください。

- **LifeKeeper Single Server Protection リリースノート** -リリースノートにはサポート対象のプラットフォーム、オペレーティングシステム、アプリケーション、ストレージについて記載されています。最新のバグフィックスについても記載されています。
- **TCP/IP 接続と名前解決** – GUI の機能を使用するためにクラスターノードの双方で名前解決ができる必要があります。DNS サービスまたは */etc/hosts* を使用してください。 ローカルホストは 127.0.0.1 となります。
- ファイアウォール – 以下のポートを使用します。
  - コミュニケーションパス (TCP): 7365/tcp
  - GUI サーバーとの通信: 81/tcp、82/tcp
  - GUI サーバーとクライアントとの RMI 通信: 1024/tcp 以降の全ポート

ファイアウォールに関して
- GUI サーバー、クライアントの通信に使用するポートは SSP がインストールされるクラスターノードと、GUI クライアントを動作させる全てのシステムでポートが開放されている必要があります。
- GUI サーバー、クライアント間の通信では Java の RMI にて 1024 番以降のポートをランダムに使用します。設定方法等は テクニカルドキュメント をご参照ください。
- 使用しているポート番号に以下を追加します。 WebGUI サーバーのプロセスと lkpolicy コマンドによるポリシー設定で使用 : 778(SSL) /tcp

- **SELinux の設定確認** – SELinux のモードによっては、SSP for Linux はインストールできません。
  - enforcing モード – SSP for Linux をインストールできません。
  - permissive モード – SSP for Linux をインストールできます。(いくつかのテスト済み ARK 環境を除いて推奨できません。)
    - SAP 環境で必要にならない限り、SELinux を permissive モードにすることは推奨されていません。クラスターでの動作が必要となるアプリケーションが permissive モードをサポートしているか確認してください。SELinux の permissive モードは次の ARK でテスト済みです。: SAP MaxDB / Sybase / Oracle / DB2 / NFS / NAS / IP / FileSystem / MQ。必要となるパッケージについては Linux の依存関係 を参照してください。
  - disabled モード – SSP for Linux をインストールできます。
    - SELinux を無効にする方法は、OS 側のドキュメントを確認してください。
  - ご利用のディストリビューションから提供される適切なパッケージをインストールしてください。

- 既知の問題 を確認してください。– ご利用の環境に既知の問題がないか確認してください。

# SSP for Linux のインストール

LifeKeeper Single Server Protection (SSP) 構成内の各サーバに LifeKeeper Single Server Protection ソフトウェアをインストールしてください。

LifeKeeper のセットアップスクリプトがパッケージマネージャー(yum もしくは zypper)を使用してパッケージをインストールするため、LifeKeeper が依存するパッケージは自動的にインストールされるようになりました。

> **重要** : LifeKeeper を正しくインストールするには、yum または zypper の機能構成が必要です。非機能構成では、インストールが失敗する可能性があります（詳細については、インストールに関する既知の問題 を参照してください）。さらに、インストールがハングする可能性があるため、パッケージマネージャーレポジトリまたは rpm データベースはロックしないでください。依存パッケージをパッケージマネージャーから自動的にインストールできない場合は、事前にLinux の依存関係を参照して必要なパッケージをインストールしてください。

LifeKeeper Single Server Protection のイメージファイル （`lkssp.img`） は、システムに LKSSP をインストールする際に必要な、ユーザ対話型システムセットアップ作業を実行するための一連のインストールスクリプトを提供しています（詳細については、対話モード を参照）。 非対話型インストールも実行できます（詳細については、非対話モード を参照）。

サーバの Host ID または Entitlement ID を取得および表示するためのユーティリティを提供するライセンスユーティリティパッケージもインストールされます。 LKSSP を実行するための有効なライセンスを取得するために、Host ID または Entitlement ID、あるいはその両方が使用されます。

> **重要**: LifeKeeper Single Server Protection をインストールする前に、Linux の依存関係 トピックを参照してください。

インストール作業はセットアップスクリプトの操作 を参照してください。

アップグレード方法については、SSP のアップグレード を参照してください。

# 13.2.3. セットアップスクリプトの操作

LifeKeeper をインストールするために、セットアップスクリプトを使用して以下の作業を行います。

## セットアップスクリプトの機能

セットアップスクリプトは以下の2つの使用方法があります。

1. 対話型インストール
   メニュー画面からLifeKeeperの構成を設定して、インストールまで行います。
   このとき構成情報を保存すると、後述の非対話型インストールで使用できます。
2. 非対話型インストール
   保存した構成情報を用いて、LifeKeeperをインストールします。
   ユーザーへの問い合わせは発生しないので、構築ツール（Ansible）などから実行することも可能です。

## セットアップの手順

### 対話型インストール

1. root ユーザーでログイン後、次のコマンドを使用して、lkssp.img ファイルをマウントします。

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

   ここで、PATH はイメージへのパスです
   IMAGE_NAME はイメージの名前です
   MOUNT_POINT はマウント位置へのパスです

2. lkssp.imgがマウントされたディレクトリに移動して、次のコマンドを入力します。

   ```
   ./setup [-s <response_file>]
   ```

   -s オプションを指定すると、設定した構成情報をresponse_fileに保存できます。このファイルは非対話型インストールで使用します。

3. スクリプトを実行すると、まず環境情報の収集を行います。

   インストールまたはアップグレードのシステム要件が満たされていない場合は、エラーメッセージが表示され、インストール/アップグレードはキャンセルされます。

   また、何らかの問題が発生した場合や設定の変更が必要な場合は、警告メッセージが表示され、インストールを続行するか中止するかをユーザーに決定するように求めます。

4. メニュー画面にてインストールするSSPの機能やアプリケーションリカバリーキット（ARK）を選

択します。詳しくは「対話画面の操作」を参照してください。

5. インストールするSSPの機能やアプリケーションリカバリーキット（ARK）を選択し、**<Done>** を選択すると、インストールが始まります。

   インストール完了後に何らかの通知が出力された場合は、その指示に従い必要な対応を行ってください。

## 非対話的インストール

1. root ユーザーでログイン後、次のコマンドを使用して、lkssp.img ファイルをマウントします。

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

   ここで、PATH はイメージへのパスです
   IMAGE_NAME はイメージの名前です
   MOUNT_POINT はマウント位置へのパスです

2. LifeKeeperのインストールを行うシステムへ、構成情報ファイルをコピーした後、次のコマンドを実行します。

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

   「-q y」オプションを指定すると、警告に対して了解したと渡します。

> **注記:** 非対話モードでのインストールで、-sオプションで保存した設定ファイルを使用する場合、そのファイルが使用されるシステムは、そのファイルが生成されたシステムと同じように構成されている必要があります。システムの違いが多すぎると、非対話モードでのインストールは失敗する可能性があります。
> create_response_fileスクリプトで作成した設定ファイルはそのような制限がありません。

## 構成情報の作成

非対話型インストールで使用する構成情報ファイルは、`setup -s <response_file>`でセットアップ作業中に作成するか、create_response_fileスクリプトで事前に作成することが出来ます。

1. root ユーザーでログイン後、次のコマンドを使用して、lkssp.img ファイルをマウントします。

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

   ここで、PATH はイメージへのパスです
   IMAGE_NAME はイメージの名前です
   MOUNT_POINT はマウント位置へのパスです

2. sps.img がマウントされたディレクトリに移動して、次のコマンドを入力します。

```
./create_response_file <response_file>
```

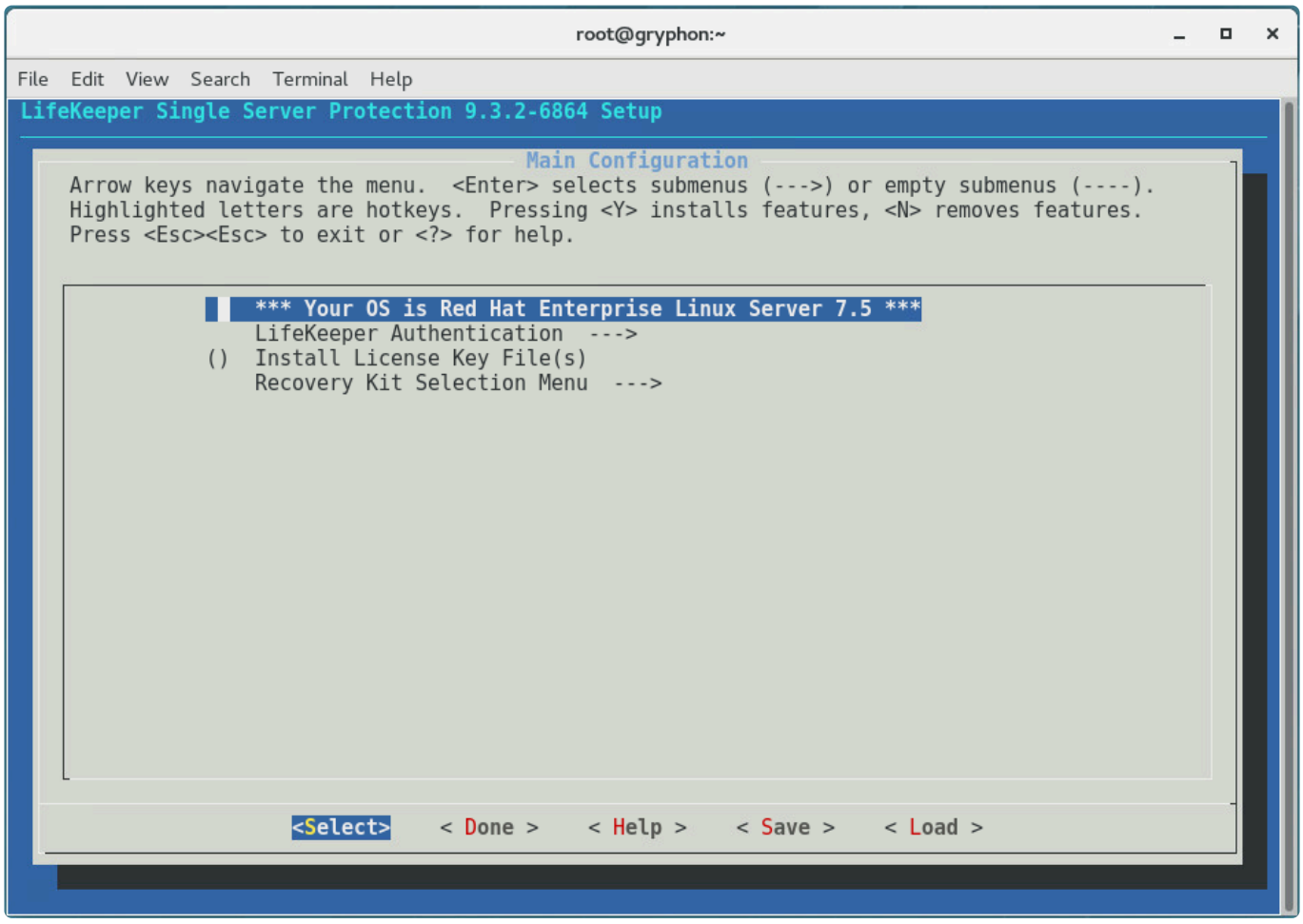
3. メニュー画面にてインストールするSPSの機能やアプリケーションリカバリーキット（ARK）を選択します。詳しくは「対話画面の操作」を参照してください。

4. インストールするSPSの機能やアプリケーションリカバリーキット（ARK）を選択し、**<Done>** を選択すると、構成をresponse_fileへ保存してスクリプトが終了します。response_fileはインストール先のシステムへコピーして使用します。

   保存された構成情報を変更する場合は、再度手順2～4を実行してください。

## 対話画面の操作

以下のようなメニュー画面が表示されます。




以下のキーで項目選択を行います。

| | |
|---|---|
| ↑↓ | 選択項目の移動 |
| ← → | 最下行ボタンの移動 |
| ENTER | 選択したサブメニューを開く |
| Y / N / Enter | 選択した項目のオン、オフ、または反転 |

最下行ボタンは以下の動作を行います。

| | |
|---|---|
| Select | 選択項目の ON / OFF ・子画面へ移動 |
| Done | 現画面を終了して親画面に戻ります<br>Main Configuration 画面の場合は、メニュー画面を終了して、インストールの確認画面へ移行します |
| Help | 選択項目のヘルプ画面を表示します |
| Save | 設定した構成情報をファイルに保存します<br>保存した構成情報ファイルは非対話モードでのインストールに使用できます<br>(create_response_fileでは無効です) |
| Load | 保存済み構成情報ファイルの読み込み<br>(create_response_fileでは無効です) |

メニューボタン <Save> および <Load> をクリックすると、現在の設定を保存するとき、または保存した設定を読み込むときに使用する設定ファイル名を尋ねるダイアログ画面が表示されます。 デフォルトのファイル名を変更したい場合は、[TAB] キーを使用してファイル名フィールドに移動し、新しいファイル名を入力してください。 **注記:** <Save> をクリックした場合、同じ名前でファイルを上書きする前に確認を求められます。

各項目では以下の機能を設定できます。

アップグレード中は、設定可能な項目のみが一覧表示されます。 ホットキー <Z> を使用すると、アップグレード時は変更されない項目が表示されます。

- **Install Java Runtime (JRE)**

  LifeKeeper GUI が使用する Java 実行環境をインストールします。

- **LifeKeeper Authentication**

  LifeKeeper SSP GUI へのログインを許可するユーザーとその権限を指定します。 複数のユーザー名を空白で区切って指定できます。 詳細は、 GUIユーザー設定 を参照してください。

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

  LifeKeeper のライセンスファイルを指定します。ファイル名を入力できるダイアログに切り替わるので、ライセンスファイルのフルパス名を入力して下さい。スペース区切りで複数のライセンスファイルを指定することが出来ます。

  ライセンスの取得方法についてはライセンスの取得とインストール を参照してください。

- **Recovery Kit Selection Menu**

  インストールするアプリケーションリカバリーキットを選択します。 アプリケーションリカバリーキットは、機能に基づいていくつかのカテゴリーに分類されています。

詳細については、アプリケーションリカバリーキットの分類 を参照してください。

- **LifeKeeper Startup After Install**

  インストールもしくはアップデート完了後に LifeKeeper を起動します。

## アプリケーションリカバリーキットの追加と削除

インストール完了後にアプリケーションリカバリーキットを追加するには、setup を実行し、[Recovery Kit] でリカバリーキットを選択し、次にアプリケーションリカバリーキットのカテゴリーを選択して、ご希望のキットを選択します。不要になったアプリケーションリカバリーキットの選択を解除すると、そのキットは削除されます。
ただし、使用中のリソースはキットを削除できないので、事前にリソースを削除してください。

## 修復インストール

LifeKeeper SSP インストールを修復するには、「`--force`」オプションを使用してセットアップを実行します。 修復インストールでは、失われたファイルや破損したファイルを置き換えてインストールを更新します。

## setup スクリプトのオプション

setup スクリプトは以下のオプションを指定して実行できます。

- `-f <response_file>`

  非対話モードでインストールを行います。<response_file>には、インストール中に使用する設定情報が含まれています。

- `-s <response_file>`

  メニューで選択した内容を含む設定ファイルを保存します。このファイルを「-f」オプションと一緒に使用して、同じ LifeKeeper 構成を別のシステムにインストールすることができます。たとえば、次のコマンドを実行します。

  ```
  ./setup -s <response_file>
  ```

  必要なパッケージとオプションを選択し、セットアップを完了します。

  その後、以下のコマンドを実行し、最初にセットアップを実行したときに選択したのと同じオプションで LifeKeeper のサイレントインストールを（別のシステムで）実行します。

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

- `--force`

LifeKeeper SSP を強制的に再インストールします。

- `-q <y/n>`

  非対話モードでのインストール中に発生する可能性のある確認への回答を指定します。

## リカバリーキットの分類

| カテゴリー | 説明 |
|---|---|
| Application Suite | IBM MQ アプリケーションを保護するリカバリーキットが所属します |
| Database | Oracle、PostgreSQL、MaxDBなどのデータベースアプリケーションを保護するリカバリーキットが所属します |
| File Sharing | NFSやSambaなどのファイル共有サービスを保護するリカバリーキットが所属します |
| Mail Server | Postfixなどのメールサービスを保護するリカバリーキットが所属します |
| Storage | Network Attached Storage（NAS）など（これらに限らない）のデータ保管方法を保護するリカバリーキットが所属します |
| Web Server | Apache などのウェブサービスを保護するリカバリーキットが所属します |

# 13.2.4. SSP のアップグレード

LifeKeeper Single Server Protection (SSP) は、既存の階層を保ちながら将来のリリースにアップグレードすることができます。

> 注記: 最新バージョンの LifeKeeper Single Server Protection にアップグレードできるのは、2 世代前までのバージョンです。それより前のバージョンからアップグレードする場合、古いバージョンをアンインストールし、LifeKeeper Single Server Protection を再インストールする必要があります。 旧バージョンをアンインストールする代わりに、旧バージョンを 1 世代前もしくは 2 世代前のバージョンにアップグレードした上で、最新バージョンにアップグレードする方法もあります。

> 注記: アップグレード中に `lkbackup` を使用する場合の詳細については、lkbackup の既知の問題 を参照してください。

1. 必要な場合、SSP をアップグレードする前に、Linux オペレーティングシステムをアップグレードしてください。

2. セットアップスクリプトの操作 を参照してLifeKeeperをアップグレードします。

# 13.2.5. ライセンスの取得とインストール

LifeKeeper Single Server Protection では、サーバーごとに別々のライセンスが必要です。ライセンスは、ランタイムライセンスです。つまり、LifeKeeper Single Server Protection のインストールはライセンスなしでも可能ですが、正常に製品を起動して実行するためには、事前にライセンスをインストールする必要があります。

インストールスクリプトによってインストールされるライセンスユーティリティーパッケージは、LifeKeeper Single Server Protection ソフトウェアの初期インストール時にサーバーの使用可能なすべての Host ID を取得して表示します。ライセンスがインストールされると、このユーティリティーは Entitlement ID (使用可能な場合) または Host ID (使用できない場合) を返します。

> 注記: Host ID が表示される場合は常に NIC の MAC アドレスに基づいています。

SIOS Technology Corp. ライセンス管理ポータルから取得した LifeKeeper Single Server Protection ライセンスには Entitlement ID が含まれ、クラスタ内の特定のノードにロックされることはありません。LifeKeeper Single Server Protection ソフトウェアと一緒に提供された Entitlement ID (認証コード) は、LifeKeeper Single Server Protection ソフトウェアを実行するために必要なパーマネントライセンスを取得するために使用されます。このプロセスを以下の図に示します。

> 注記: ソフトウェアパッケージごとに、サーバーごとのライセンスが必要になります。

各サーバーのライセンスを取得してインストールするには、次の手順を実行します。

1. **Host ID** を取得します。 SPSのインストールの最後に、 **ライセンスキーインストーラーユーティリティ** に表示される **Host ID** をメモしてください。

2. **Host ID** をノートにメモするか、ファイルに保存します。ファイルに保存した場合は、そのファイルをインターネットにアクセスできるシステムにコピーしてください。それ以外の場合は、インターネットにアクセスできるシステムまでノートを持っていってください。

3. SIOS Protection Suite **Entitlement ID** (認証コード) を所有していることを確認します。ライセンスを取得するために必要な Entitlement ID が記載された E メールをソフトウェアとともに受け取っているはずです。

4. SIOS Technology Corp. の Licensing Operations Portal からライセンスを取得します。

   a. インターネットにアクセスできるシステムから SIOS Technology Corp. の Licensing Operations Portal にアクセスし、 **ユーザー名** と **パスワード** を入力してログインします。 (アカウントをお持ちでない場合は登録してください。)
   **注記:** 新規ユーザーの場合、出荷メールに記載されているEntitlement IDを入力してください。

   b. **[Activation and Entitlements]** ドロップダウンから、 **[List Entitlements]** を選択します。

   c. ライセンスを取得したい製品の左側にあるチェックボックスにチェックを入れます。

   d. **[Action]** ドロップダウンから **[Activate]** を選択し、必要な情報 (システムのHOSTNAMEなど) を入力して **[Next]** を選択します。

   e **[Gray Plus Sign]** をクリックして定義済みのホストを選択するか、 **[Green Plus Sign]** を選択して新しいホストを作成します。

   f. [Node Locked Host] の選択肢に [ANY] が表示されている場合には **[ANY]** を選択して [OK] をクリックしてください。[ANY] が表示されていない場合には **[ETHERNET MAC ADDRESS]** を選択し、Host ID (MACアドレス) を入力して **[OK]** をクリックし、次に **[Generate]** をクリックします。

> **注記:** Host IDは12文字で、スペース、コロン、ダッシュ、区切り文字は使用できません。

   g. **[Fulfillment ID]** の左側のボックスをオンにして、 **[Complete]** を選択します。

   h. **[License Support]** ドロップダウンから **[List Licenses]** を選択します。 **[Fulfillment ID]** の左側のボックスをオンにして、 **[View]** ドロップダウンから **[Email]** を選択します。

   i. ライセンス送信先の有効なメールアドレスを入力して、 **[Send]** を選択します。

   j. 受信したメールを確認します。

   k. ファイルを該当するシステムにコピーします。

5. ライセンスをインストールします。

- 各システム上で、ライセンスファイルを /var/LifeKeeper/licenseにコピーします。/opt/LifeKeeper/bin/lkkeyins を実行し、ファイル名（フルパスを含む）を指定します。

6. その他のサーバーについて、上記の手順を繰り返します。他のSPSサーバー上では、サーバーごとに一意の Host ID を使用してライセンスをインストールする必要があります

7. SIOS Protection Suite for Linuxを再起動します。

## プライマリーネットワークインターフェースを変更すると、ライセンスの Rehost が必要になる場合がある

ライセンスキーインストーラーユーティリティー で使用されるホスト ID は、 SIOS Protection Suite for Linux サーバーのプライマリーネットワークインターフェースカード (NIC) から取得されます。 SIOS Protection Suite for Linux は、起動されるたびに有効なライセンスを確認します。将来、 SIOS Protection Suite for Linux サーバーの NIC を交換してホスト ID が変わった場合は、次に SIOS Protection Suite for Linux を停止したとき、再起動する前にライセンスの Rehost を実行する必要があります。 SIOS Technology Corp. の Licensing Operations Portal にログインして、画面上メニューを [License Support] > [List Licenses] とクリックして画面遷移させた後、目的のライセンスを選択して [Action] > [Rehost] の順に選択し、Rehost を実行してください。

> **注記:** 製品サポートへのご連絡なしにお客様ご自身で Rehost を行うことができるのは 6 か月に 4 回です。

# 13.2.6. リソースポリシー管理

## 概要

LifeKeeper Single Server Protection (SSP) のリソースポリシー管理では、リソースのローカルリカバリとフェイルオーバの動作管理機能が提供されます。リソースポリシーは、 **lkpolicy** コマンドラインツール (CLI) を使用して管理できます。

## LifeKeeper SSP のリカバリ動作

LifeKeeper SSP には、個々のアプリケーションおよび関連し合うアプリケーションのグループを監視する機能があり、定期的にローカルリカバリを実行したり、保護下のアプリケーションに障害が発生したときに通知したりすることができます。関連し合うアプリケーションの例としては、主アプリケーションが下位のストレージまたはネットワークリソースに依存する階層などがあります。アプリケーションまたはリソースに障害が発生した場合のデフォルトの動作は以下の通りです。

1. **ローカルリカバリ:** 最初に、リソースまたはアプリケーションの **ローカル** でリカバリを試みます。このときは、外部の介入なしにローカルサーバ上でリソースまたはアプリケーションをリストアしようとします。ローカルリカバリが成功した場合、LifeKeeper SSP は追加のアクションを実行しません。

2. **フェイルオーバ:** 次に、ローカルリカバリでリソースまたはアプリケーションのリストアに *失敗* した (またはリソースを監視する *リカバリキット* がローカルリカバリをサポートしていない) 場合、 **フェイルオーバ** が開始されます (下記の標準ポリシーセクションの Failover を参照)。

リカバリ動作の詳細については、LifeKeeper Single Server Protection 障害検出とリカバリのシナリオ を参照してください。

## ポリシーによるカスタム動作およびメンテナンスモード動作

LifeKeeper SSP は、デフォルトのリカバリ動作を変更する追加ポリシーを設定する機能をサポートします。リソース単位またはサーバ単位で、4 つのポリシーが設定可能です (リソース単位のポリシーに関する注意については下のセクションを参照してください)。 **サーバレベルでポリシーを変更する方法を推奨します。**

利用可能なポリシーは以下の通りです。

### 標準ポリシー

- **Failover** - LifeKeeper SSP では、このポリシー設定を使用すると、リソースフェイルオーバを有効 / 無効にできます (これによって再起動されます)。

- **LocalRecovery** - LifeKeeper SSP は、デフォルトでは、フェイルオーバ (再起動) を実行する前に、個々のリソースまたは保護対象アプリケーション全体を再起動することにより、保護対象リソースのリカバリを試みます。このポリシー設定を使用すると、ローカルリカバリを有効 / 無効にできます。

- **TemporalRecovery** - 通常、LifeKeeper SSP は、障害リソースのローカルリカバリを実行します。ローカルリカバリに失敗すると、LifeKeeper SSP は再起動を実行します。ローカルリカバリに成功した場合は、フェイルオーバ (再起動) は実行されません。

ローカルリカバリに成功した場合でも、サーバの何らかの異常によって短時間の間にローカルリカバリが再試行される場合があり、結果として何度も連続してローカルリカバリが試行されることになります。これが発生すると、問題のアプリケーションは可用性が悪化します。

この反復的なローカルリカバリ / 障害サイクルを回避するために、時間的リカバリポリシーを設定できます。時間的リカバリポリシーを使用すると、管理者は指定した時間内に試行するローカルリカバリの回数を (成功かどうかにかかわらず) 制限することができます。

> ソースが試行するローカルリカバリの回数を 30 分間で 3 回に限定するポリシー定義をユーザが設定した場合、30 分以内に 3 回目のローカルリカバリが試行されると、LifeKeeper SSP はフェイルオーバ (再起動) を実行します。

定義した時間的リカバリポリシーは有効または無効にできます。時間的リカバリポリシーが無効の場合、時間的リカバリ処理は継続して実行され、ポリシーが適用されるはずの時間に通知がログに表示されますが、実際のアクションは実行されません。

**注記:** 時間的リカバリポリシーを設定した状態で、フェイルオーバとローカルリカバリの一方または両方を無効にすることは可能です。フェイルオーバまたはローカルリカバリを無効にした場合に、時間的リカバリポリシーは実行されることがないため、この状態は非論理的です。

### メタポリシー

「メタ」ポリシーは、他の複数のポリシーに影響を与える可能性があるポリシーです。通常、これらのポリシーは、標準ポリシーであれば複数個の設定が必要になるような特定のシステム動作を実現するためのショートカットとして使用します。

- **NotificationOnly** - このモードでは、管理者は LifeKeeper SSP を「監視専用」状態にすることができます。 **1 つのリソース (または、サーバ単位のポリシーの場合はすべてのリソース) の** ローカルリカバリ **および** フェイルオーバ (再起動) **の両方が影響を受けます。** 障害が検知されると、ユーザインターフェースには **Failure** 状態が表示されます。ただし、リカバリもフェイルオーバ (再起動) も実行されません。 **注記:** 管理者は、障害の原因となった問題を手動で修正し、障害が起きたリソースを復帰させて通常の LifeKeeper SSP の運用を継続する必要があります。

## リソースレベルのポリシーに関する重要な考慮事項

リソースレベルのポリシーとは、リソース階層全体またはサーバレベルのポリシーとは異なり、特定のリソースにのみ適用されるポリシーです。

> *例:*
>
> アプリケーション
>
> - IP

- file system

上記のリソース階層では、アプリケーションは IP とファイルシステムの両方に依存しています。ポリシーは、特定のリソースのローカルリカバリまたはフェイルオーバを無効にするように設定できます。これは、例えば、IP リソースのローカルリカバリが失敗し、IP リソースのフェイルオーバが無効に設定されていた場合、IP リソースはフェイルオーバを実行せず、他のリソースのフェイルオーバも発生させないことを意味します。ただし、ファイルシステムリソースのローカルリカバリが失敗し、ファイルシステムリソースのポリシーのフェイルオーバが無効化されていない場合、階層全体が再起動を伴うフェイルオーバを実行します。

✻ **注記:** 重要事項として、リソースレベルのポリシーは設定対象の特定のリソースにのみ適用されることに注意してください。

上記は単純な例です。複雑な階層を構成することもできるため、リソースレベルのポリシーを設定するときは注意してください。

! 既知の問題を参照してください。 *(“フェイルオーバ時に誤った順序でリソースが停止されます“の項目)*

# lkpolicy ツール

`lkpolicy` ツールは、LifeKeeper SSP が稼働するサーバのポリシーを管理 (参照、設定、削除) するためのコマンドラインツールです。`lkpolicy`は、ポリシーの設定および修正、ポリシーの削除、利用可能なポリシーと現在の設定値の表示をサポートします。さらに、設定したポリシーは、有効または無効に設定できるため、リカバリ動作に影響を与えながらリソース / サーバ設定を保持できます。

全体的な使用方法は次の通りです。

```
lkpolicy [--list-policies | --get-policies | --set-policy | --remove-policy]
<name value pair data…>
```

`<name value pair data…>`は、運用方法および対象のポリシーによって異なります (特にポリシーを設定する場合)。 *たとえば、以下のようになります。* 有効 / 無効タイプのポリシーのほとんどでは、必要なのは `-on or --off` のスイッチのみですが、時間的ポリシーの場合は、しきい値を設定するための値も必要です。

## lkpolicy の使用方法の例

### ローカルおよびリモートサーバとの認証

`lkpolicy` ツールは、サーバが公開する API を通じて LifeKeeper SSP サーバと通信します。この API は、`lkpolicy`ツールなどのクライアントに対して認証を要求します。`lkpolicy` ツールで LifeKeeper SSP サーバに最初にアクセスしようとしたときに、そのサーバに対する認証情報がまだ保存されていない場合、ユーザは認証情報を求められます。認証情報はユーザ名とパスワードの形式であり、さらに以下の

条件があります。

1. クライアントには LifeKeeper SSP の管理者権限が必要です。したがって、そのユーザ名は、(PAM による) オペレーティングシステムの認証設定によって `lkadmin` グループに属する必要があります。必ずしも **root** で実行する **必要はありません** が、root ユーザはデフォルトで適切なグループに属しているため、root を使用することもできます。

2. 認証情報は *認証情報ストア* に保存されるため、ツールを使用してこのサーバにアクセスするたびに手動で認証情報を入力する必要はありません。

認証情報ストアと credstore ユーティリティによる管理の詳細については、SIOS Protection Suite の認証情報の設定 を参照してください。

`lkpolicy` によるセッションの例は以下のようになります。

```
[root@thor49 ~]# lkpolicy -l -d v6test4
Please enter your credentials for the system 'v6test4'.
Username: root
Password:
Confirm password:
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]# lkpolicy -l -d v6test4
Failover
LocalRecovery
TemporalRecovery
NotificationOnly
[root@thor49 ~]#
```

### ポリシーのリスト表示

lkpolicy --list-policy-types

### 現在のポリシーの表示

lkpolicy --get-policies

lkpolicy --get-policies tag=*

lkpolicy --get-policies --verbose tag=mysql* # all resources starting with mysql

lkpolicy --get-policies tag=mytagonly

### ポリシーの設定

lkpolicy --set-policy Failover --off

```
lkpolicy --set-policy Failover --on tag=myresource

lkpolicy --set-policy Failover --on tag=\*

lkpolicy --set-policy LocalRecovery --off tag=myresource

lkpolicy --set-policy NotificationOnly --on

lkpolicy --set-policy TemporalRecovery --on recoverylimit=5 period=15

lkpolicy --set-policy TemporalRecovery --on --force recoverylimit=5 period=10
```

### ポリシーの削除

```
lkpolicy --remove-policy Failover tag=steve
```

**注記:** *NotificationOnly* はポリシーのエイリアスです。 *NotificationOnly* を有効にすることは、対応する LocalRecovery および Failover ポリシーを無効にすることと同じです

# 13.2.7. LifeKeeper Single Server Protection のインストールの検証

LifeKeeper Single Server Protection パッケージが正常にインストールされたことを確認するには、コマンドラインで次のように入力してください。

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

**注記:** パッケージが正しくインストールされている場合、このコマンドは何も出力しません。

コマンドラインから照会を実行するには、次のように入力してください。

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

**注記:** このコマンドの予想される出力は、パッケージ情報です。

# 14. LifeKeeper Single Server Protection テクニカルドキュメンテーション

## LifeKeeper Single Server Protection for Linux について

LifeKeeper Single Server Protection (SSP) は、単一ノード構成におけるアプリケーション監視を可能にします (つまり、クラスタの要件または制約はありません)。単一ノード環境は、物理的なものでも仮想 (vSphere、KVM) でも構いません。LifeKeeper SSP は、実績がある安定した SIOS LifeKeeper アーキテクチャ上に構築されます。LifeKeeper SSP は優れたアプリケーション監視機能を提供し、障害が発生したアプリケーションおよびシステムインフラストラクチャ項目 (例: NFS 共有、IP アドレス、ファイルシステム) のリカバリを実行することができます。何らかの理由でアプリケーションをリカバリできない場合、LifeKeeper SSP は、システムのリブートまたは VM とアプリケーション監視を設定された VMware 仮想マシンの VMware HA 再起動によって、ノードの再起動を開始します。

**注記:** LifeKeeper SSP は SIOS LifeKeeper 技術を使用して構築されているため、ドキュメント全体で LifeKeeper を参照します。また、両製品に共通するトピックについては SIOS Protection Suite for Linux ドキュメンテーションの情報を参照します。これらの共通のトピックを参照する場合、LifeKeeper SSP には以下の話題は適用されません。また、サポートもされません。

- クラスタリング
- コミュニケーションパス
- 共有ストレージ (要件、構成、...)
- リソース階層の拡張/拡張解除
- Quorum/Witness
- アプリケーションリカバリーキット
  - Recovery Kit for EC2
  - LVM Recovery Kit
  - MD Recovery Kit
  - Route53 Recovery Kit
  - SAP Recovery Kit
  - SAP HANA Recovery Kit
  - VMDK as Shared Storage Recovery Kit
- ストレージキット (DR、DMMP、HDLM、LVM、MD、PPATH、NEC SPS)

  **注記:** LifeKeeper SSP にベースとなるストレージキットがない場合、保護されるファイルシステムのマウントに必要なデバイスがシステム起動時にアクティベートされるようにするための手順が必要です (例: ファイルシステムが論理ボリューム上でマウントされる場合、LifeKeeper SSP が起動する前にボリュームがアクティブな状態になっていなければなりません)。

- リソース/マシンのフェイルオーバ (LifeKeeper SSP のデフォルトでは、これによってノードが再起動されます)
- リソースのスイッチオーバ
- 切り替え可能な IP アドレス (LifeKeeper SSP では、保護されるアプリケーションの一部には切り替え可能な IP アドレスが必要ですが、単一ノードしかないため、実際には切り替えは行われません)

LifeKeeper SSP のベースになっている SIOS LifeKeeper 製品の詳細については、共通するリリース番号の SIOS Protection Suite for Linux ドキュメンテーション を参照してください。このドキュメンテーションは、リソース階層の作成、リソースタイプ、状態と関係、グラフィカルユーザインターフェース (GUI)、および共通の作業と高度な作業に関する詳細情報を提供します。

# 14.1. ドキュメンテーションとトレーニング

SIOS LifeKeeper Single Server Protection for Linux のインストール、設定、管理、およびトラブルシューティングの方法を説明する関連ドキュメントは、SIOS Technology Corp. ウェブサイト の ドキュメンテーション セクションから参照できます。SIOS LifeKeeper Single Server Protection for Linux のあらゆる側面について、以下のセクションで説明しています。

| セクション | 説明 |
|---|---|
| はじめに および インストール | LifeKeeper Single Server Protection 環境のプランニングと設定、LifeKeeper Single Server Protection のインストールとライセンス、LifeKeeper のグラフィカルユーザインターフェース (GUI) の設定に役立つ情報を提供します。 |
| 管理 | サーバのプロパティの編集やリソースの作成などのサーバレベルの作業、およびリソースの編集、拡張、削除などのリソースレベルの作業について説明します。 |
| ユーザガイド | LifeKeeper GUI で実行できる多数の作業を含めて、LifeKeeper の GUI に関する詳細情報があります。 |
| トラブルシューティング | 既知の問題と制限について説明し、SIOS LifeKeeper Single Server Protection for Linux のインストール、設定、および使用を行うときに発生する可能性がある問題に対する解決策を説明します。 |
| Recovery Kit | LifeKeeper Single Server Protection で特定のアプリケーションを管理および制御するために必要なオプションの Recovery Kit のプランニングおよびインストール手順、管理、設定、およびユーザ情報が含まれます。 |

## トレーニング

LifeKeeper Single Server Protection のトレーニングは、SIOS Technology Corp. または代理店から受講可能です。詳細については、営業担当者にお問い合わせください。

## テクニカルサポート

SIOS Technology Corp. と有効なサポート契約を結んだお客様は、新しい SIOS Technology Corp. のセルフサービスサポートポータル にアクセスできます。

SIOS Technology Corp. のセルフサービスサポートポータル では、以下のことができます。

- 弊社のソリューションナレッジベースから、問題の解決策と質問に対する回答を検索する。
- 次のメニューを選択して、年中無休の SIOS Technology Corp. のサポートチームにアクセスする。
- **Log a Case** - 新しいインシデントを報告する。
- **View Cases** - お客様の未解決と解決済みのインシデントをすべて表示する。
- **Review Top Solutions** - 弊社のお客様が表示した、最も一般的な問題の解決策の情報を表示する。

セルフサービスポータルを設定してアカウントを有効にする方法については、SIOS Technology Corp. のサポート ( support@us.sios.com ) にお問い合わせください。

また、SIOS Technology Corp. のサポートには、以下の方法でも連絡できます。

1-877-457-5113 (通話料無料)

1-803-808-4270 (米国以外のお客様)

電子メール: support@us.sios.com

# 14.2. VMware HA との連携

「はじめに」セクションで説明したように、LifeKeeper Single Server Protection は、物理環境と仮想環境の両方で使用できるように設計されています。LifeKeeper SSP を VMware VM にインストールした場合、VMware の HA 機能を LifeKeeper SSP と組み合わせて、保護対象リソースやノードの障害を監視し、復旧することができます。 これらの機能を有効にする方法については、VMware HA と LifeKeeper Single Server Protection の連係の有効化 を参照してください。

# 14.3. LifeKeeper Single Server Protection の管理の概要

LifeKeeper Single Server Protection は操作時に管理を必要としません。LifeKeeper Single Server Protection は、保護されたリソースを監視し、障害が発生した場合に指定されたリカバリアクションを実行するように、自動的に機能します。以下のケースでは LifeKeeper Single Server Protection GUI を使用します。

- **リソースおよび階層の定義** 。LifeKeeper Single Server Protection は次のインターフェースオプションを提供します。
    - º LifeKeeper Single Server Protection GUI。
    - º LifeKeeper Single Server Protection コマンドラインインターフェース。
- **リソース監視。** LifeKeeper Single Server Protection GUI は、リソースステータス情報および LifeKeeper Single Server Protection ログへのアクセスを提供します。
- **手動での処理。** メンテナンスやその他の管理アクションのために、サーバまたは特定のリソースを停止することが必要になる場合があります。LifeKeeper Single Server Protection GUI には、特定のリソースを稼動させたり停止させたりすることができるメニュー機能が用意されています。アプリケーションが LifeKeeper Single Server Protection の保護下に置かれると、これらの LifeKeeper Single Server Protection のインターフェースを介してのみアプリケーションを起動および停止させることができます。LifeKeeper Single Server Protection の起動および停止は、コマンドラインを介してのみ行われます。

リソース階層の作成など、管理、設定、およびメンテナンス操作を実行する詳細な手順については、SPS for Linux ドキュメンテーションの管理作業 、GUI の作業 、およびメンテナンス作業 を参照してください。

※LifeKeeperが提供しているコマンド（実行可能なスクリプトやプログラム）を実行するには、スーパーユーザ権限が必要です。
su コマンドや sudo でスーパーユーザ権限を付与したユーザで、LifeKeeper のコマンドを実行することは可能ですが、SIOS Technology Corp では、root ユーザ以外で LifeKeeper のコマンドのテストはしていません。

# 14.3.1. VMware HA と LifeKeeper Single Server Protection の連携を有効にする

デフォルトでは、VMware VM 上にインストールした場合、LifeKeeper Single Server Protection と VMware HA の連携は無効になっています。連携を有効にするには、以下の手順が必要です。

1. LifeKeeper Single Server Protection VM に VMware Tools をインストールしてください。

2. */etc/default/LifeKeeper* を編集して、VMware HA 連携を調整する `HA_DISABLE` の値を 1 から 0 に変更します。

3. VMware HA のパラメーター *das.iostatsinterval* を 0 に設定してください。
   0 に設定すると、 仮想マシンの監視感度 に対する I/O 統計間隔が無効になります。(チェックが行われません。)デフォルトは120秒です。

4. LifeKeeper Single Server Protection を再起動します。LifeKeeper Single Server Protection が実行中の場合、 */etc/default/LifeKeeper* の上記の変更内容が検出されるように、停止してから再起動する必要があります。

# 14.3.2. VMware HA を有効化した障害検出およびリカバリシナリオ

アプリケーション内の問題を検出して通知する機能は、最適な総合的耐障害性ソリューションを構築する上で非常に重要です。すべての個々のアプリケーションは、障害発生のメカニズムと形式によって異なるため、一般的なメカニズムを示すことはできません。ただし、一般的に、多くのアプリケーションの設定は、LifeKeeper Single Server Protection に用意されている Core システムのエラー検出機能を利用することができます。このトピックでは、LifeKeeper Single Server Protection Core の機能について説明します。

アプリケーションに障害が発生したときに LifeKeeper Single Server Protection が障害を検出しリカバリを実行する仕組みを説明したリカバリシナリオを以下に示します。

1. LifeKeeper Single Server Protection は最初に、アプリケーションを再起動することでリカバリを試みます。

2. リカバリが成功した場合、アプリケーションは正常動作を継続します。

3. リカバリに失敗した場合、以下の処理が実行されます。

    a. LifeKeeper Single Server Protection が HA を有効 (/etc/default/LifeKeeper で HA_DISABLE=0) にした VMware ゲスト OS にインストールされている場合にリカバリに失敗すると、LifeKeeper Single Server Protection がアプリケーション監視インターフェースに送信するハートビートを抑制することで VMware HA がトリガされます。VMware HA はサーバを再起動することで応答します。

    b. LifeKeeper Single Server Protection が VMware ゲスト OS にインストールされていないか、HA を無効 (/etc/default/LifeKeeper で HA_DISABLE=1) にした VMware ゲスト OS にインストールされている場合にリカバリに失敗すると、システムが強制的に再起動されます。

# 14.3.3. LifeKeeper Single Server Protection ハートビートと VMware HA

LifeKeeper Single Server Protection ハートビートは、保護対象のアプリケーションが正常であることを示すために、(VMware ゲスト OS で動作し、HA が有効である場合、10 秒ごとに) VMware HA に送信される信号です。アプリケーションで障害が発生すると、LifeKeeper Single Server Protection は最初にアプリケーションを復旧しようとします。復旧に失敗すると、LifeKeeper Single Server Protection はハートビートを抑制し、VMware HA に VM の再起動を指示します。

# 14.3.4. LifeKeeper Single Server Protection で保護するシステムのメンテナンス

LifeKeeper Single Server Protection で保護されているサーバでシステムまたはアプリケーションのメンテナンスを実行するときには、LifeKeeper Single Server Protection による監視を停止するか、保護対象のリソースをメンテナンスモードにしてください。これにより、アプリケーションのリカバリ、および VMware HA の障害イベントのトリガが無効になるので、LifeKeeper Single Server Protection がシステムやアプリケーションのメンテナンス作業に干渉しなくなります。

LifeKeeper Single Server Protection を停止して再起動するには、以下の操作を実行してください。

1. **LifeKeeper Single Server Protection を停止します。 "/opt/LifeKeeper/bin/lkstop -f"** コマンドを使用して、LifeKeeper Single Server Protection を停止してください。リソースは継続して動作しますが、LifeKeeper Single Server Protection からは監視されなくなります。障害は手動で処理する必要があります。

2. **メンテナンスを実行します** 。必要なメンテナンスを実行します。

3. **SIOS LifeKeeper Single Server Protection を起動します。"/opt/LifeKeeper/bin/lkstart"** コマンドを使用して、LifeKeeper Single Server Protection を開始してください。リソースが保護されている状態になります。

**別の方法** - リソースをメンテナンス (「通知のみ」とも呼ばれる) モードにします。以下の操作を実行してください。

1. **リソースをメンテナンスモードにします。 /opt/LifeKeeper/bin/lkpolicy -s NotificationOnly --On** コマンドを使用してください。リソースは復旧されなくなり、VMware HA の障害イベントがトリガされなくなります。

2. **メンテナンスを実行します** 。必要なメンテナンスを実行します。

3. **メンテナンスモードをオフにします。 /opt/LifeKeeper/bin/lkpolicy -s NotificationOnly --Off** コマンドを使用してください。リソースが保護されている状態になります。

# 14.3.5. Quick Service Protection (QSP) Recovery Kit

QSP Recovery Kit は、OSのサービスを簡易的に保護する仕組みを提供します。OSのserviceコマンドで起動 (start) ・停止 (stop) ができるサービスであれば、容易にリソースが作成できます。Generic Applicationを使っても同じことが実現できますが、QSPを使えばコードの開発が不要です。また、依存関係を作成することで、他のリソースで保護しているアプリケーションと連携してサービスを起動･停止させることができます。

ただし、QSPのquickCheckでは簡易なチェックしか行っていません（serviceコマンドのstatusを利用しています）。実際にそのサービスが提供できているか、プロセスが動作しているかは保証されません。複雑な起動や停止処理、堅牢なチェックが必要な場合は、Generic Applicationの利用を検討してください。

詳細は以下のURLをご参照ください。

Quick Service Protection (QSP) Recovery Kit

# 14.3.6. LifeKeeper API for Monitoring

LifeKeeper API for Monitoringは、稼働中のLifeKeeperに対して問い合わせを行うことで、LifeKeeperノードの稼働状況や保護しているリソースの状況を取得できます。

本APIはlighttpd上でCGIとして動作し、lcdstatusコマンドで取得できる情報を、API経由で取得できるようにしたものです。本APIで取得できる情報は以下のとおりです。

- LifeKeeperノードの死活
- コミュニケーションパスの死活
- リソースの保護状況

詳細は以下のURLをご参照ください。

LifeKeeper API for Monitoring

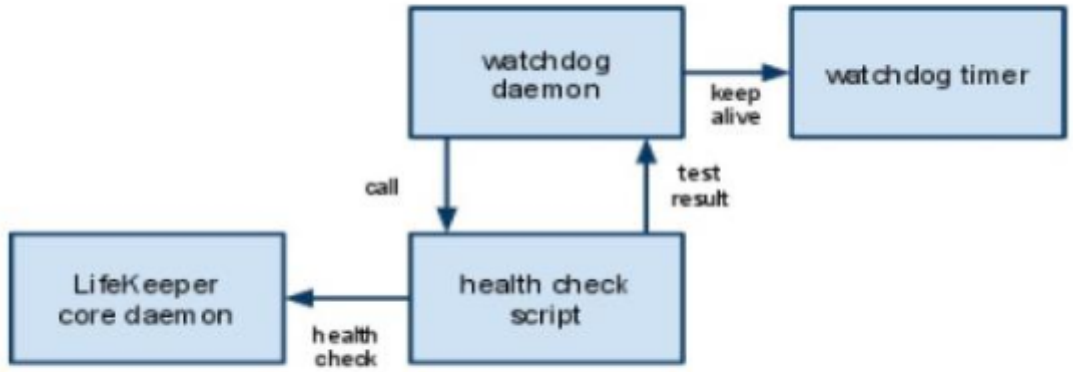
# 14.3.7. Watchdog

Watchdog は、サーバーが正常に動作しない場合に、問題の発生を予防する修正処置 (リブート) を確実に実行できるようにサーバーを監視する方法です。 Watchdog は、特別な Watchdog ハードウェアを使用して実装する場合と、ソフトウェアのみのオプションを使用して実装する場合があります。

> **注記:** この構成は、Red Hat Enterprise Linux Versions 7 でのみ検証されています。他のオペレーティングシステムでは検証されていないため、現時点ではサポートされません。

## コンポーネント

- Watchdog timer - ソフトウェアドライバまたは外部ハードウェアコンポーネント
- Watchdog daemon - 該当する Linux ディストリビューションを通じて rpm が入手可能
- LifeKeeper Core daemon - LifeKeeper のインストールに付随
- health cehck script - LifeKeeper SSP core の動作状況を確認するスクリプト




**LifeKeeper Interoperability with Watchdog**

**LifeKeeper と Watchdog の相互運用性**

次のセクションを注意深く読んでください。デーモンは、エラーからリカバリするように設計されており、注意深く設定しないとデーモンはシステムをリセットします。インストールおよび設定を行う前に慎重に計画してください。このセクションの目的は、 Watchdog についての説明や設定をすることではありません。ここでは、 Watchdog 構成での LifeKeeper SSP との相互運用についての説明や設定のみ行います。

## 設定

以下の手順は、root ユーザ権限を持つ管理者が行う必要があります。管理者は、 Watchdog のリスクおよび問題についてすでに熟知しているものとします。

ヘルスチェックスクリプト (LifeKeeper 監視スクリプト `/opt/LifeKeeper/samples/watchdog/`

`LifeKeeper-watchdog`) は、LifeKeeper の設定と Watchdog の設定を関連付けるコンポーネントです。このスクリプトは、LifeKeeper コアコンポーネントの基本部分を監視できます。

1. 以前に Watchdog を設定していた場合は、次のコマンドを入力して Watchdog を停止します。そうでない場合は、手順 2 に進みます。
   `systemctl stop watchdog`

2. Watchdog ソフトウェアのインストールで供給される Watchdog 設定ファイル (*`/etc/watchdog.conf`*) を編集します。

   a. test-binary を次のように修正します。

   ```
   test-binary = /opt/LifeKeeper/samples/watchdog/LifeKeeper-watchdog
   ```

   b. test-timeout を次のように修正します。

   ```
   test-timeout = 5
   ```

   c. interval を次のように修正します。

   ```
   interval = 7
   ```

   interval は、test-timeout 以上の値であることが必要です。また、あまりに長い間隔にすると障害検出が遅れてしまうので、5〜10程度を推奨します。

3. LifeKeeper SSP が起動していることを確認します。まだの場合は、LifeKeeper の起動 トピックを参照してください。

4. 次のコマンドを入力して Watchdog を起動します。
   `systemctl start watchdog`

5. 今後の再起動の際に Watchdog を自動的に起動させるには、次のコマンドを入力します。
   `systemctl enable watchdog`

> **注記:** Watchdog を設定すると、予想外のリブートがときどき発生する可能性があります。これは、 Watchdog の仕組みから来る一般的な性質です。正常に応答しないプロセスがあると、 Watchdog 機能は LifeKeeper (またはオペレーティングシステム) がハングしていると判断し、(警告なしに) システムをリブートします。

## アンインストール

LifeKeeper をアンインストールする場合は、慎重に行ってください。以下に列記の通り、上記の手順を逆の順で実行します。

> **警告:** LifeKeeper を構成する RPM パッケージを削除する方法で LifeKeeper をアンインストールする場合は、 **先に Watchdog を停止してください** 。上記の手順 2 では、LifeKeeper の Watchdog スクリプトを呼び出すように Watchdog 設定ファイルを修正しています。したがって、先に Watchdog を停止しておかないと、存在しないスクリプトを呼び出すことになります。リブートを実行するこのスクリプトが見つからない場合は、エラーが発生します。この状態は Watchdog を停止するまで継続します。

1. 次のコマンドを入力して Watchdog を停止します。
   `systemctl stop watchdog`

2. Watchdog ソフトウェアのインストールで供給される Watchdog 設定ファイル (*/etc/watchdog.conf*) を編集します。

   a. test-binary および interval の両エントリをコメントアウトします (各行の先頭に # を追加します)。

   ```
   #test-binary =
   #interval =
   ```

   **注記:** interval が他の機能によって以前から使用されていた場合は、そのままにしておくこともできます。

3. LifeKeeper をアンインストールします。LifeKeeper の削除 トピックを参照してください。

4. これで Watchdog を起動し直すことができます。LifeKeeper のみが Watchdog を使用していた場合は、次のコマンドを入力すると Watchdog を永続的に無効にできます。
   `systemctl disable watchdog`

# 14.3.8. LKCLI（LifeKeeper コマンドラインインターフェース）

LKCLI は、LifeKeeper の GUI で操作可能な機能をコマンドラインインターフェースで実現しています。また、GUI には無い機能として、コミュニケーションパスおよびリソース情報の export/import 機能があります。export/import 機能は、作成したシステムの量産や、テスト環境から本番環境のデプロイを容易に行える手段となります。

## サポート対象

次に示す環境での使用をサポート対象としています。

- OS：LifeKeeper 本体でサポートしている OS はすべてサポートされます。
- アプリケーションリカバリーキット： 後述する制限事項に該当するものを除き ARK 一覧 に記載されたアプリケーションリカバリーキットのみで構成された環境のみサポートされます。ノード内にサポート対象外の ARK リソースが作成されている場合、その環境はサポート対象外となります。

サポート対象外の環境でのコマンドライン操作をご検討されているお客様は、「 コマンドラインインターフェース 」の使用をご検討ください。

## 制限事項

Single Server Protection for Linux では、以下関連する機能については適用されません。またサポートもされません。

- 共通オプション
  - [ --remote <str> ]
- コミュニケーションパスの操作
  - lkcli commpath …
  - lkcli import（コミュニケーションパスの作成のみ適用されません。）
- リソースの拡張/拡張解除や切り替えに関する操作
  - lkcli resource extend
  - lkcli resource unextend
  - lkcli resource eqv
  - lkcli resource reorder-priority
  - lkcli resource switchback
  - lkcli server …
- アプリケーションリカバリーキット
  - SIOS DataKeeper for Linux
  - Recovery Kit for EC2
  - LVM Recovery Kit
  - MD Recovery Kit
  - Route53 Recovery Kit
  - SAP Recovery Kit
  - SAP HANA Recovery Kit

    - VMDK as Shared Storage Recovery Kit
    - Multipath Recovery Kit（DMMP / HDLM / PPATH / NECSPS）
- IP Recovery Kit のlkcli export または lkcli resource info コマンド
  lkcli export コマンド、 lkcli resource info コマンドで処理が停止してしまう場合は、ホスト名が正しく設定されていることを確認してください。

Single Server Protection for Linux でも、SIOS Protection Suite for Linux と同様に LKCLI を使用できます。詳細については「 LKCLI（LifeKeeper コマンドラインインターフェース） 」を参照してください。

# 14.4. トラブルシューティング

このセクションには、LifeKeeper Single Server Protection の制限または既知の問題が記載されています。

# 14.4.1. 既知の問題と回避策

下記に、LifeKeeper Single Server Protection で明らかになっている制限または既知の問題を示します。

## Core

バグ **2257**

**LifeKeeper Single Server Protection** および **SIOS Protection Suite** のノードに

**credstore** 経由でアクセスするときに、正しい **credstore** キーが必要です

**解決方法:** `credstore` を使用して、LifeKeeper Single Server Protection または SIOS Protection Suite のノードの認証情報を保存するときに、credstore の認証情報キーについてホスト名の正しい形式 ( `credstore -k <hostname>`) を使用する必要があります。

LifeKeeper Single Server Protection プラグインの場合、LifeKeeper Single Server Protection プラグイン画面の **[Hostname:]** フィールドに表示されるシステムのホスト名を使用して `credstore` を実行する必要があります。

SIOS Protection Suite の場合、認証情報の保存に使用するホスト名は、コマンドラインツール (`lkpolicy` など) の -d 引数に使用するものと同じである必要があります。例えば、**`lkpolicy -d mynode1`**, を実行する場合、**`credstore -k mynode1`** を使用して認証情報を保存する必要があります。この場合、認証情報の保存に FQDN を使用することはできません。FQDN を使用する場合は、**`lkpolicy -d FQDN`** を実行する必要があります。

**対応策:** LifeKeeper Single Server Protection または SIOS Protection Suite のすべてのノードで機能するデフォルトの認証情報セット (**`credstore -k default`**) を保存した場合、この問題の影響を受けることはありません。

---

バグ **2408**

**HA** ハートビートが不正に有効になります

2 番目のリソースに障害が発生した後、lkvmhad が HA ハートビートを不正に有効にします。

**対応策:** */etc/default/LifeKeeper* の `LKCHECKINTERVAL` に、VMware HA の [VM Monitoring Failure Interval] (VM の障害監視間隔) よりも大きい値を設定してください。 **注記:** `LKCHECKINTERVAL` のデフォルト値は 120 秒です。また、これは、VMware HA の VM 監視の監視感度「low」のデフォルト値でもあります。

# GUI

**LifeKeeper Single Server Protection GUI の更新の問題**

GUI のリソースツリーが不規則に表示されることがあります (リソースの依存関係が正しく表示されない)。

**対応策:** GUI の更新を実行してください。

**注記:** lkoper、lkguest でGUI画面を更新できない

# Apache

**Apache リソースの作成に失敗する**

エラーメッセージの例：

Error: valid_http_root: Since "/usr/sbin/httpd" is shareable on "/usr", "/etc/httpd" must be also

原因：

不具合のため、マウントポイント"/"(root)にあるファイルを正しく検索できません。

例えば、マウントポイント"/"と同じファイルシステムに/etc/httpdがある場合、リソースの作成に失敗します。

**対応策:**

次のいずれかのようにマウントすることにより、エラーを回避することができます。

(a) /etc/httpdなどを別のマウントポイント以下に移動する

(b) /etcを/dev/sdb1などにマウントする

# Oracle

バグ **2387**

**LifeKeeper Single Server Protection 環境の root ファイルシステムに、Oracle 階層を作成できません。**

**対応策:** 以下の手順に従って、Oracle を新しいファイルシステムにコピーしてください。

Oracle データを格納できる十分に大きい新しいディスク (例: /dev/sdb) を作成してください。(注記: ディスクの大きさを見積もるために、/oracle ディレクトリの大きさを参考にできます。ログを含めるために、50 % 以上増加してください)。

gdisk を使用して、そのディスクに新しいパーティションを作成してください。

```
gdisk /dev/sdb
```

ファイルシステムを作成してください。

mkfs -t ext3 /dev/sdb1

このファイルシステムをマウントしてください (この例では /mnt/oracle を使用)。

mkdir /mnt/oracle

mount /dev/sdb1 /mnt/oracle

Oracle と Listener を停止してください。

Oracle を新しいファイルシステムにコピーしてください。

cd /oracle

cp -a * /mnt/oracle

(注記: データ量により、この手順には時間がかかることがあります)

新しいファイルシステムをアンマウントしてください。

umount /mnt/oracle

新しいファイルシステムを /oracle にマウントしてください。

mount /dev/sdb1 /oracle

Listener を開始し、次に Oracle を開始してください。

**Oracle Recovery Kitは、AWS EC2上のOracle Database Standard Edition 2 (SE2)をサポートしません。**

AWS EC2上のシステムにおいてOracle Database Standard Edition 2 (SE2)のテストを行った際に、幾つかの奇妙な振る舞いが確認されました。しかしながら、EC2以外のシステムにおいては正常動作が確認できておりますので、EC2以外のシステムにおいてAWS EC2上のOracle Database Standard Edition

2 (SE2)の利用をサポートします。

## SAP

バグ **2388**

**SAP の場合、GUI を使用して階層を作成することはできません。**

**対応策**: コマンドラインオプションを使用して、階層を作成してください。ただし、以下に示すように、コマンドラインの最後に番号 76 を指定してください。

- $LKROOT/lkadm/subsys/appsuite/sap/bin/create <primary sys> <tag> <SAP SID> <SAP Instance> <switchback type> <IP Tag> <Protection Level> <Recovery Level> <Additional SAP Dependents> 76

コマンドラインの詳細については、「コマンドラインによる SAP の設定」を参照してください。

SIOS Protection Suite for Linuxにおける「既知の問題と制限」も参照ください。

# 14.5. アプリケーションリカバリーキット

LifeKeeper Single Server Protection for Linux Application Recovery Kits (ARKs) は特定のアプリケーションをSSP上で管理、制御するためのツールとユーティリティを含んでいます。今バージョンのSSPで利用できるARKは以下の通りです。

- Apache Recovery Kit 管理ガイド
- DB2 Recovery Kit 管理ガイド
- IP Recovery Kit 管理ガイド
- MySQL Recovery Kit 管理ガイド
- WebSphere MQ Recovery Kit 管理ガイド
- NAS Recovery Kit 管理ガイド
- NFS Recovery Kit 管理ガイド
- Oracle Recovery Kit 管理ガイド
- PostgreSQL Recovery Kit 管理ガイド
- Postfix Recovery Kit 管理ガイド
- Samba Recovery Kit 管理ガイド
- SAP MaxDB Recovery Kit 管理ガイド
- Sybase Recovery Kit 管理ガイド