



LifeKeeper[®] for Linux v7.5

Apache Web Server Recovery Kit v7.4
管理ガイド

2011 年11 月

SteelEye and LifeKeeper are registered trademarks.

Adobe Acrobat is a registered trademark of Adobe Systems Incorporation. Apache is a trademark of The Apache Software Foundation. HP and Compaq are registered trademarks of Hewlett-Packard Company. IBM, POWER, DB2, Informix, ServeRAID, Rational and ClearCase are registered trademarks or trademarks of International Business Machines Corporation. Intel, Itanium, Pentium and Xeon are registered trademarks of Intel Corporation. Java is a registered trademark of Sun Microsystems, Inc. Linux is a registered trademark of Linus Torvalds. Microsoft Internet Explorer and Windows are registered trademarks of Microsoft Corporation. MySQL and MaxDB are registered trademarks or trademarks of MySQL AB. Netscape and Netscape Navigator are registered trademarks of Netscape Communications Corporation. NFS is a registered trademark of Sun Microsystems, Inc. Opteron is a trademark of Advanced Micro Devices, Inc. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. PostgreSQL is a trademark of PostgreSQL Global Development Group. Red Flag is a registered trademark of Red Flag Software Co.,Ltd. Red Hat is a registered trademark of Red Hat Software, Inc. SAP is a registered trademark of SAP AG. Sendmail is a registered trademark of Sendmail, Inc. Sun and Solaris are registered trademarks of Sun Microsystems, Inc. SUSE is a registered trademark of SUSE LINUX AG, a Novell business. Sybase is a registered trademark of Sybase, Inc. Other brand and product names used herein are for identification purposes only and may be trademarks of their respective companies.

It is the policy of SIOS Technology Corp. (previously known as SteelEye Technology, Inc.) to improve products as new technology, components, software, and firm-ware become available. SIOS Technology Corp., therefore, reserves the right to change specifications without prior notice.

To maintain the quality of our publications, we need your comments on the accuracy, clarity, organization, and value of this book.

Address correspondence to:

ip@us.sios.com

Copyright © 2011

By SIOS Technology Corp.

San Mateo, CA U.S.A.

All Rights Reserved

目次

はじめに	3
本書の内容.....	3
LifeKeeper のドキュメント.....	3
要件.....	4
キットのハードウェア／ソフトウェア要件.....	4
LifeKeeper での Apache Web Server の設定	5
設定の定義と例	5
ローカル設定	5
設定メモ：	6
共有設定.....	7
設定メモ：	8
Server1 での Apache Web Server リソース階層の作成.....	8
アクティブ／スタンバイ設定とアクティブ／アクティブ設定	9
Apache Web Server の設定上の考慮事項	9
LifeKeeper 設定作業	14
Apache Web Server リソース階層の作成.....	15
階層の拡張.....	17
階層の拡張解除	19
リソース階層の削除	20
リソース階層のテスト.....	21
GUI からの手動切り替えの実行.....	21
リカバリ操作	21
トラブルシューティング	22
階層作成時のエラー	22
Apache Web Server 実行形式ファイルの場所の検証中に発生するエラー	22
サーバ・ルート・ディレクトリの検証中に発生するエラー	23
Apache リソース階層の作成中に発生するエラー	24
階層拡張時のエラー	25
Apache Web Server 実行形式ファイルの場所の検証中に発生するエラー	25
ターゲット・システムの Apache 設定ファイルの検証中に発生するエラー.....	25
ターゲット・サーバで Apache リソース階層の作成中に発生するエラー	27
階層リソースの復元、削除、リカバリ時のメッセージとエラー	27
Apache リソースをサービス中にするとき（復元）	27
Apache リソースをサービス休止にするとき（削除）	29
Apache リソースをサービス中に戻すとき（リカバリ）	30

Apache Web Server Recovery Kit 管理ガイド

はじめに

The LifeKeeper® for Linux Apache Web Server Recovery Kitは、LifeKeeper環境でApache Web Serverソフトウェアの障害回復能力を高めます。

本書の内容

このガイドは以下のトピックについて説明しています。

- [LifeKeeperのドキュメント](#) : LifeKeeper for Linuxの全ドキュメントの一覧と入手方法。
- [要件](#) : リカバリ・ソフトウェアをインストールおよびセットアップするには、サーバが一定のハードウェア/ソフトウェア要件を満たしている必要があります。LifeKeeper Apache Recovery Kitのインストール/アンインストール方法の詳細については、『LifeKeeper for Linuxプランニングおよびインストールの手引き』を参照してください。
- [Recovery Kitの設定](#) : 必要とする保護と柔軟性が得られるようにLifeKeeperを設定するには、設定ルールを把握している必要があります。設定を適切にプランニングするには、ネットワーク設定、インタフェースの選択、ユーザ・システムのセットアップ、階層オプション、Apache設定作業についての理解が必要です。このセクションでは、設定のプランニングに加え、Recovery Kitを設定するために必要な特定の作業についても、設定例を示して説明します。
- [トラブルシューティング](#) : このセクションでは、情報メッセージとエラー・メッセージ、および推奨される対処方法の一覧を示します。

LifeKeeper のドキュメント

以下は、SIOS Technology Corp が発行している LifeKeeper 関連ドキュメントの一覧です。

- 『LifeKeeper for Linux リリース・ノート』
- 『LifeKeeper for Linux Online Product Manual』 (LifeKeeper GUI の[Help]メニューから参照できます)
- 『LifeKeeper for Linux プランニングおよびインストールの手引き』

これらのドキュメントは、その他の LifeKeeper Recovery Kit 関連のドキュメントとともに、ソフトウェアに付属する LifeKeeper ドキュメント CD に収録されています。

LifeKeeper for Linux のドキュメントは、次の Web サイトでもオンラインで入手できます。

<http://us.sios.com/support>

参考資料

以下は、Apache Web Server アプリケーションおよび LifeKeeper Apache Recovery Kit に関する参考資料の一覧です。

- Apache のオンライン・ドキュメント
- 『Apache: The Definitive Guide, 2nd Edition』、Ben Laurie and Peter Laurie、O'Reilly & Associates, Inc. 1999 年

要件

Apache Recovery Kit のインストール／アンインストールを試みる前に、ハードウェア／ソフトウェア要件とインストール／アンインストールの手順を理解しておく必要があります。

キットのハードウェア／ソフトウェア要件

LifeKeeper Apache Recovery Kit をインストールおよび設定する前に、使用している設定が以下の要件を満たしていることを確認してください。

- **サーバ : Recovery Kit** : LifeKeeper の要件に従って設定された、サポートされているコンピュータが 2 台以上必要です。LifeKeeper の要件は、製品のメディアに付属している『LifeKeeper Online Product Manual』と『LifeKeeper リリース・ノート』に記載されています。
- **LifeKeeper ソフトウェア** : 各サーバに同じバージョンの LifeKeeper ソフトウェアとパッチをインストールする必要があります。詳細な LifeKeeper 要件については、『LifeKeeper リリース・ノート』と『Online Product Manual』を参照してください。
- **LifeKeeper IP Recovery Kit** : 同じバージョンの Recovery Kit を各サーバにインストールする必要があります。
- **IP ネットワーク・インタフェース** : 各サーバには、イーサネット TCP/IP をサポートするネットワーク・インタフェースが 1 つ以上必要です。IP 切り替えが正しく機能するためには、ローカル・ネットワークに接続したユーザ・システムが標準 TCP/IP 仕様に準拠している必要があります。

注 : 各サーバに必要なネットワーク・インタフェースが 1 つだけであっても、異種メディア要件、スループット要件、シングル・ポイント・オブ・フェイラの排除、ネットワークのセグメンテーションといった多くの理由で、複数のインタフェースを使用すべきです。

- **TCP/IP ソフトウェア** : 各サーバには TCP/IP ソフトウェアも必要です。
- **Apache ソフトウェア** : Apache Web Server ソフトウェアが各サーバにインストールおよび設定されていなければ、使用される DSO (Dynamic Shared Object) モジュールも含め、LifeKeeper と LifeKeeper Apache Web Server Recovery Kit の設定はできません。各サーバに同じバージョンの Apache Web Server ソフトウェアをインストールしてください。最新リリースの互換性および発注情報については、『LifeKeeper リリース・ノート』を参照するか、販売担当者にお問い合わせください。

LifeKeeper Apache Recovery Kit のインストール／アンインストール方法の詳細については、『LifeKeeper for Linux プランニングおよびインストールの手引き』を参照してください。

LifeKeeper での Apache Web Server の設定

このセクションでは、典型的な LifeKeeper Apache Web Server 設定の定義と例を示し、Apache Web Server の設定を開始する前に検討すべき情報について説明します。

LifeKeeper コアのリソース階層を設定する手順については、『LifeKeeper Online Product Manual』を参照してください。

設定の定義と例

Apache Web Server は、複数インスタンスの **httpd** デーモンの同時実行をサポートしています。各 LifeKeeper Apache Web Server 階層は、固有の「サーバ・ルート」ディレクトリを持つ別個の Apache インスタンスに対応しています。各インスタンスは「仮想ホスト」を使用できるように設定されていれば、複数の **Web** サイトをサポートできます。

サーバ・ルート・ディレクトリには、**Web** インスタンスの設定方法を指定する設定ファイルである `conf/httpd.conf` が格納されるため、このディレクトリが主に Apache Web Server のインスタンスを定義します。このファイルに記述されている Apache の設定ディレクティブは、そのインスタンスのログ・ファイル、**Web** ドキュメント、他の設定ファイルなどの配置場所や、使用する IP アドレスやドメイン名アドレスを決定します。

(共有ストレージを使用した) LifeKeeper のファイル・システムを使用するかどうかに応じて、Apache Web Server の LifeKeeper 用設定を特徴づけると便利です。サーバ・ルート・ディレクトリ (設定ファイル `conf/httpd.conf` が存在する) やドキュメント・ルート・ディレクトリ (および、オプションとして **httpd** 実行形式ファイル自体) には、1 つの共有ファイル・システムを使用してください。ある特定の Apache インスタンスにローカル設定または共有設定のどちらを使用するかを決めるのは、主に以下の 2 つの要素です。同一の設定ファイルや **Web** サイトのドキュメントを個々に保持することの困難さと、複数のサーバ間で共有 (またはミラーリング) できるストレージの可用性およびアクセスしやすさです。ただし、同一サーバ上で、ローカル Apache インスタンスと共有 Apache インスタンスの両方を設定することもできることに注意してください。

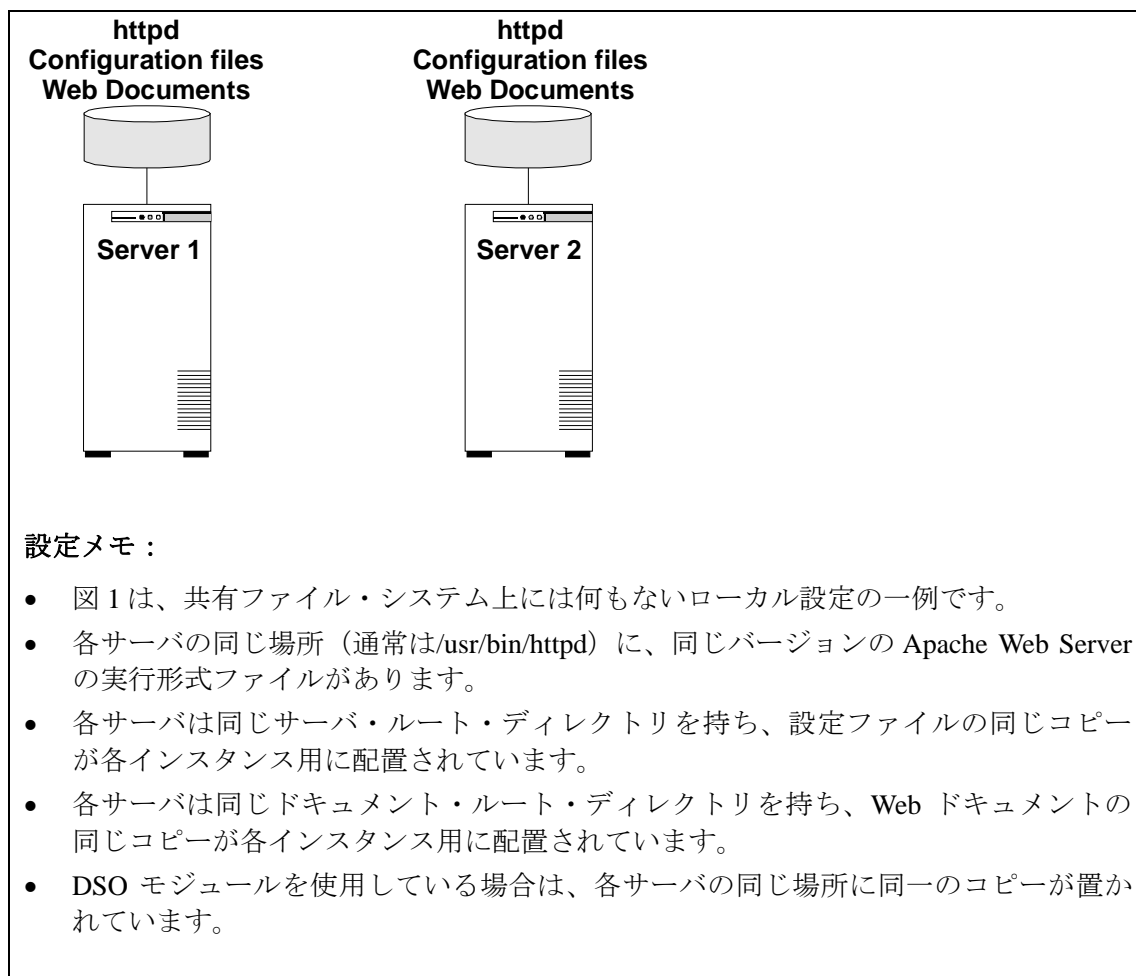
以下のセクションでは、LifeKeeper 環境における Apache Web Server の [ローカル設定](#) および [共有設定](#) の例を挙げ、それぞれの主な特徴を概説します。

ローカル設定

典型的なローカル設定では、サーバ間で何も共有されません。同一の Apache Web Server 設定ファイル、**Web** ドキュメント、DSO モジュール (および存在する場合は設定ファイル)、**httpd** 実行形式ファイルが、各サーバのまったく同じ場所に配置されます。異なるサーバで Apache コンポーネントの同じコピーを維持するのは、Apache 管理者の担当作業になります。

設定ファイルによって、各 **Web** サイトは 1 つの IP アドレス (または特定の IP アドレスにマッピングされているドメイン・アドレス) を割り当てられています。また、各 **Web** サイト用に LifeKeeper IP アドレスが作成され、Apache リソース階層に追加されます。Apache 階層があるサーバから別のサーバに切り替えられると、最初のサーバ上の **httpd** インスタンスが停止され、IP アドレスは非アクティブ化されます。次に、もう一方のサーバで IP アドレスが再アクティブ化され、インスタンスが起動されます。クライアントは TCP/IP 経由で、もう一方のサーバにある同一の **Web** サイトに自動的に接続されます。

図1. ローカル設定



Server1 での Apache Web Server リソース階層の作成

サーバ	Server1
Web Server の実行形式ファイルの場所	/usr/sbin/httpd
Web Server のルート・ディレクトリ	/home/www/examples/instance1/
ルート・タグ	apache-www.examples.instance1

Server2 への Apache Web Server リソース階層の拡張

テンプレート・サーバ	Server1
拡張するタグ	apache-www.examples.instance1
ターゲット・サーバ	Server2
ターゲットの優先順位	10

Apacheリソース階層を 1 台または複数台のサーバに拡張するときは、ローカル設定の場合でも共有設定の場合でも、Web Serverの実行形式ファイルの場所とルート・ディレクトリはすべてのサーバ上で同じでなければなりません。詳細については、上記の説明と「[Apache Web Serverの設定上の検討事項](#)」に関するセクションを参照してください。階

層の拡張中、LifeKeeperはApacheリソース階層の一部であるすべての従属リソースも拡張します。

共有設定

典型的な共有設定では、サーバ・ルート・ディレクトリとドキュメント・ルート・ディレクトリはすべて同じ共有ファイル・システム上にあります。サーバ間で同じ設定ファイルと Web ドキュメントが共有されるため、それぞれのサーバで同じコピーを保持する必要はありません。DSO モジュールを使用している場合、必要な設定ファイルやリソースとともに、同じ共有ファイル・システムに DSO モジュールも配置できます。

Web ドキュメントだけを共有ファイル・システムに置くことも可能であることに注意してください。その場合はサーバ・ルート・ディレクトリがローカルにあるため、一般的なローカル設定とよく似ていますが、階層には共有ファイル・システムも含まれます。

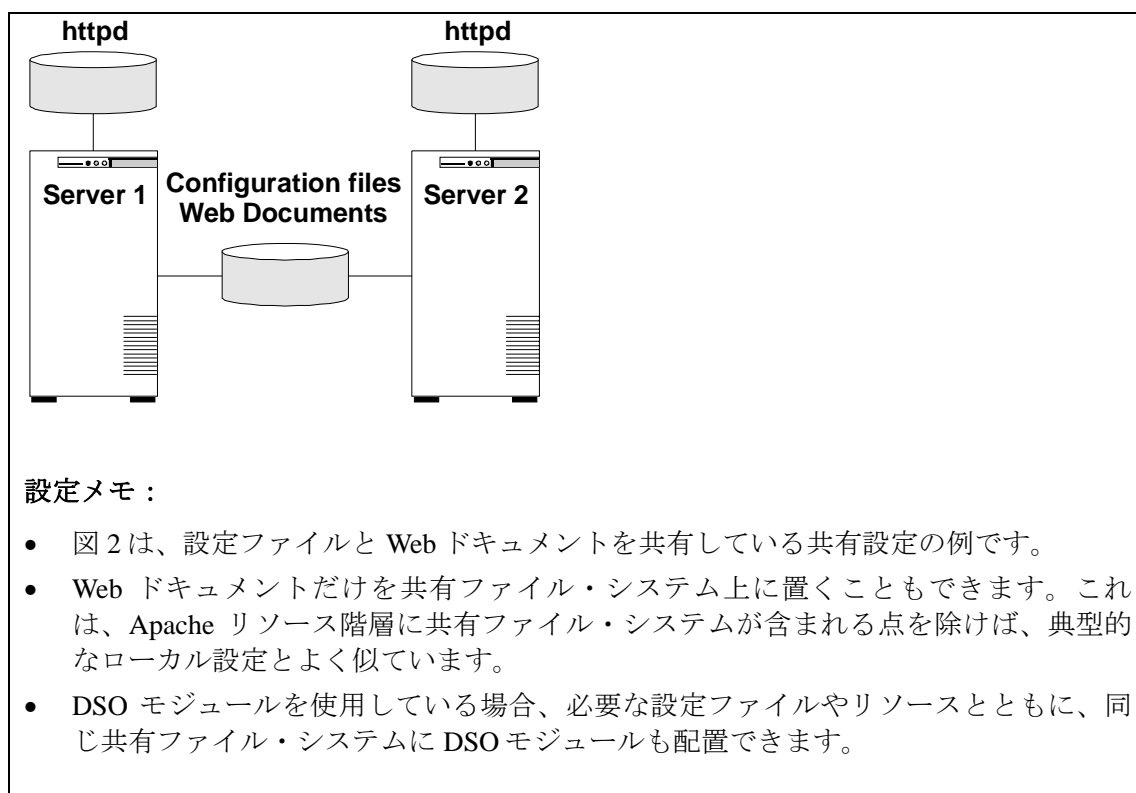
この Apache リソース階層に対して特定のバージョンまたは別個の Apache 実行形式ファイルを使用したい場合、この実行形式ファイルも共有ファイル・システム上に置けば、その実行形式ファイルを使用できるのはそのインスタンスだけになります。そのようにするには、Web Server 実行形式ファイルの場所の入力を求められたときに、共有ファイル・システム上の **httpd** 実行形式ファイルのフルパスを入力するだけです。

使用する共有ファイル・システムは 1 つだけでよいことに注意してください。これは、共有ストレージに置かれた必要なコンポーネントがすべて同時に使用可能になるためです。共有ファイル・システム上に Web Server の実行形式ファイルを置く場合、Web Server のルート・ディレクトリも同じ共有ファイル・システムに置かなければなりません。また、このサーバ・ルート・ディレクトリ用に設定するすべてのドキュメント・ルート・ディレクトリを同じ共有ファイル・システム上に置く必要があります。同様に、サーバ・ルート・ディレクトリを共有ファイル・システム上に置く場合、すべてのドキュメント・ルート・ディレクトリは同じ共有ファイル・システム上に置かなければなりません。実行形式ファイルとルート・ディレクトリのどちらも共有ファイル・システムに配置せず、いずれかのドキュメント・ルート・ディレクトリを共有する場合は、すべてのドキュメント・ルート・ディレクトリを同じファイル・システムで共有する必要があります。

これらのルールを要約すると、以下のようになります。

- Apache 実行形式ファイルを共有する場合は、サーバ・ルート・ディレクトリを共有しなければなりません。
- サーバ・ルート・ディレクトリを共有する場合は、すべてのドキュメント・ルート・ディレクトリを共有しなければなりません。
- いずれかのドキュメント・ルート・ディレクトリを共有する場合は、すべてのドキュメント・ルート・ディレクトリを共有しなければなりません。
- それぞれの Apache リソース階層で使用できる共有ファイル・システムは 1 つだけです。

図2. 共有設定



Server1 での Apache Web Server リソース階層の作成

サーバ	Server1
Web Server の実行形式ファイルの場所	/usr/sbin/httpd または /shared/example/instance2/bin/httpd
Web Server のルート・ディレクトリ	/shared/example/instance2
ルート・タグ	apache-shared.example.instance2

Server2 への Apache Web Server リソース階層の拡張

テンプレート・サーバ	Server1
拡張するタグ	apache-shared.example.instance2
ターゲット・サーバ	Server2
ターゲットの優先順位	10

アクティブ/スタンバイ設定とアクティブ/アクティブ設定

Apache Web Server は、LifeKeeper ではアクティブ/アクティブ・アプリケーションと呼ばれます。これは、1 台のサーバ上でいつでも複数の Apache インスタンスを実行できるという意味です。たとえば、2 台のサーバが 1 つの Apache インスタンスを実行していて、1 台のサーバに障害が発生した場合、障害が発生したサーバの Apache インスタンスはもう一方のサーバにフェイルオーバーし、固有のインスタンスを継続して実行できます。一部のアプリケーションではこの機能をサポートしていないので、そのようなアプリケーションの各インスタンスには 1 つのサーバを提供し続ける必要があります。このようなアプリケーションはアクティブ/スタンバイ・アプリケーションと呼ばれます。どちらのモードにも設定できるアプリケーションもあります。

複数のサーバのうちの 1 台を、主に Apache を実行するために使用する場合は特に、Apache を「アクティブ/スタンバイ」モードで運用したい状況があるかもしれません。このような場合は、バックアップ・サーバで動作するインスタンスがないように、デフォルトでインストールされた標準的な Apache の自動起動を無効にしてください。

1 台または複数台の特定のサーバで、Apache インスタンスを手動でサービス中にすることによって、必要に応じて負荷を分散することができます。また、各インスタンスのサーバの優先順位を調整すれば、障害発生時に、最後の手段としてのみ Apache インスタンスを特定のサーバにフェイルオーバーさせたり、他のサーバにフェイルオーバーさせて負荷を分散することができます。

クラスタ内の全サーバ上で Apache の自動起動を無効にすると、LifeKeeper IP アドレスを使用するようにこのインスタンスを設定するだけで、単一の LifeKeeper Apache リソース階層のためにデフォルトのサーバ・ルート・ディレクトリ/etc/http を使用できます。また、場合によっては 1 つの共有ファイル・システムを複数のドキュメント・ルート・ディレクトリのために使用できます。ただし（前述のように）、これはアクティブ/スタンバイ設定なので、通常の方法ではデフォルトのインスタンスを起動できなくなる点に注意してください。もちろん、サーバ・ルートは一意であることが必要なので、デフォルトのサーバ・ルート・ディレクトリを複数の階層で使用することはできません。

Apache Web Server の設定上の考慮事項

Apache リソース階層を作成する前に、以下の Apache Web Server アプリケーションの設定作業が完了していることを確認する必要があります。

1. 標準のデフォルト Apache インスタンスは、修正を加えないと LifeKeeper が保護しているインスタンスと競合します。Apache パッケージをインストールすると、/etc/rc.d ディレクトリの runlevel (rc) スクリプトによるシステム・セットアップ中に、デフォルトの Apache インスタンスが起動するように自動的に設定されます。デフォルトの Apache インスタンスと関連がある httpd.conf ファイルには、すべての IP アドレスのリッスン (BindAddress) に相当する BindAddress ディレクティブや Listen ディレクティブがありません。*

LifeKeeper が保護している Apache インスタンスの他に、標準のデフォルト Apache インスタンスも使用したい場合、デフォルトの設定ファイル (/etc/httpd/conf/httpd.conf) *を修正し、BindAddress ディレクティブや Listen ディレクティブを使用して、LifeKeeper IP アドレスではない 1 つまたは複数の特定の IP アドレスをリッスンする必要があります。この変更を認識させるには、httpd デーモンを停止してから再起動する必要があります。httpd デーモンは、以下のコマンドで停

止や再起動が可能です (SuSE の場合、スクリプト名は/etc/init.d/apache です)。

/etc/rc.d/init.d/httpd stop

/etc/rc.d/init.d/httpd start

この設定を修正しない場合は、Apache のデフォルト・インスタンスの自動起動を無効にする必要があります。無効にしないと、LifeKeeper Apache のインスタンスの動作と干渉します。デフォルト・インスタンスの自動起動を無効にする具体的な手順については、使用している Apache Web Server のドキュメントを参照してください。

*** SuSE で Apache を使用する場合 :** SuSE に Apache をデフォルトでインストールすると、設定ファイル httpd.conf はサーバ・ルート・ディレクトリのサブディレクトリである conf には置かれませんが、SuSE に Apache をデフォルトでインストールしている場合は、設定ファイルを/etc/httpd/conf ディレクトリに再配置する必要があります。

- LifeKeeper Apache Web Server 階層ごとに、個別のルート・ディレクトリを作成する必要があります。この「サーバ・ルート」ディレクトリは、Apache の設定パラメータおよびコマンド・ライン・パラメータである `ServerRoot` に対応します。それぞれの LifeKeeper Apache リソース階層は、関連したサーバ・ルート・ディレクトリを持つ Apache インスタンスに一意に対応します。サーバ・ルート・ディレクトリは、その Apache 階層用に設定されているすべてのサーバ上で同一でなければならないことに注意してください。LifeKeeper ソフトウェアが見つけてアクセスできるように、Web サイトの設定ファイル情報はすべて、サーバ・ルート (`conf/httpd.conf`) を基準にした標準の場所に配置する必要があります。
- Apache の推奨事項に従って、すべての設定情報を 1 つの `httpd.conf` ファイルに記述する必要があります。Apache Web Server に付属する標準の `srn.conf` ファイルと `access.conf` ファイルにはディレクティブが記述されており、使用する設定と干渉しないように、これらのファイルは無効にする必要があります (デフォルトでは、使用中のサーバ・ルート・ディレクトリとは無関係に、Apache は/etc/httpd/conf でこれらのファイルを検索します)。これら 2 つの設定ファイルの使用を無効にするには、`httpd.conf` ファイルの本文に以下のディレクティブを追加します。

```
AccessConfig /dev/null
ResourceConfig /dev/null
```

- `BindAddress` ディレクティブまたは `Listen` ディレクティブを使用して特定の LifeKeeper IP アドレスをリスンするように、すべての Web サイト (仮想ホスト) を設定する必要があります。LifeKeeper で保護するこれらの IP アドレスはあらかじめ作成しておき、Apache 階層が作成される場所でサービス中にできるようにしておく必要があります。これらの IP アドレスは、Apache リソース階層に自動的に追加されます。

LifeKeeper 共有ファイル・システムを使用する場合、Apache 階層を作成する前に、LifeKeeper 共有ファイル・システムの作成に必要な準備をすべて整えておく必要があります。特に、Apache 階層を作成するサーバ上に LifeKeeper 共有ファイル・システムをマウントする必要があります。LifeKeeper のファイル・システム階層をまだ作成していない場合、Apache 階層を作成すると自動的に作成され、Apache リソース階層に結合されます。

仮想ホストの設定に関する詳細な情報については、Apache Web Server のドキュメントを参照してください。前述したように、LifeKeeper の保護下にある特定のアドレスをリスンするように、すべての Apache インスタンスを設定する必要があります。たとえば、IP を使用した仮想ホストと名前を使用した仮想ホストを併せ持つインスタンスの設定ファイルには、以下のようなディレクティブを記述します。

```

User webuser
Group webgroup
ServerName localhost

AccessConfig /dev/null
ResourceConfig /dev/null

Listen 172.17.100.55:8000
NameVirtualHost 172.17.100.55:8000
Listen 172.17.100.56:80

<Virtualhost site.name_one:8000>
ServerName site.name_one
DocumentRoot /shared/site/name_one
</VirtualHost>

<VirtualHost site.name_two:8000>
ServerName site.name_two
DocumentRoot /shared/site/name_two
</VirtualHost>

<VirtualHost 172.17.100.56:80>
ServerName site.ip
DocumentRoot /shared/site/ip
</VirtualHost>

```

- Apache インスタンスに対して SSL のサポートを有効にする場合は、LifeKeeper に保護された適切な IP アドレスを使用するように SSL Listen ディレクティブを設定する必要があります。このディレクティブは通常、別の ssl.conf ファイルに記述されています。この設定を行わないと、IP アドレス 0.0.0.0 が LifeKeeper に保護されていないことを示すエラーが発生し、Apache 階層の作成に失敗します。一部の Linux ディストリビューションの Apache 設定では、デフォルトで SSL のサポートが有効にされることに注意してください。たとえば、/etc/httpd/conf.d/ssl.conf にあるデフォルトの SSL 構成ファイルには、次のエントリがあります。

```
Listen 0.0.0.0:443
```

このエントリを次のように変更します。

```
Listen 172.17.100.55:443
```

- 「ローカル」設定の場合、プライマリ・サーバ上とすべてのバックアップ・サーバ上の同じ場所に Apache をインストールして設定し、すべてのサーバで、同じサーバ・ルート・ディレクトリに同一（または同等）の設定ファイルをセットアップする必要があります。また、すべてのサーバに、すべてのドキュメント・ルート・ディレクトリが存在する必要があります。ディレクトリには同一のファイルが格納されている必要があります（「設定の定義と例」の「[ローカル設定](#)」セクションを参照してください）。
- 「共有」設定の場合、通常は LifeKeeper 共有ファイル・システム上にサーバ・ルート・ディレクトリを置くよう設定します。使用する共有ファイル・システムは 1 つだけでよいことに注意してください。これは、共有ストレージに置かれた必要なコンポーネントがすべて同時に使用可能になるためです。したがって、すべてのドク

メント・ルート・ディレクトリは同じ共有ファイル・システムのサブディレクトリでなければなりません、サーバ・ルート・ディレクトリ自体のサブディレクトリである必要はありません。Apache 実行形式ファイルを同じ共有ファイル・システム上に置くこともできますが、その実行形式ファイルは、この特定の Apache 階層しか使用できなくなります。

注：共有ストレージを使用するために、サーバ・ルート・ディレクトリを共有ファイル・システム上に置く必要はありません。ローカルのサーバ・ルート・ディレクトリに設定ファイルなどを置き、共有ファイル・システムにはドキュメント・ルート・ディレクトリだけを置くこともできます。しかし、(前述したローカル設定の場合のように)すべてのサーバで同一のサーバ・ルート・ディレクトリを設定し、同一(または同等)の設定ファイルを置く必要があります。また、すべてのドキュメント・ルート・ディレクトリは同じ共有ファイル・システム上になければなりません(「設定の定義と例」の「[共有設定](#)」セクションを参照してください)。

8. 一部の Web サイトの実装では、DSO (Dynamic Shared Object) モジュールを使用して、特定の機能に対する Apache のサポートを拡張しています。たとえば、PHP や Perl の機能を実装するモジュールが提供されています。これらのモジュールは実行時に、Apache コアによってロードとアクセスが行われます。モジュールを使用する場合は、クラスタ内の全サーバでまったく同じ設定にする必要があります。モジュール・パッケージのドキュメントと、使用中の Linux プラットフォームで Apache を設定してモジュールを使用するためにベンダが提供しているドキュメントを参照してください。使用するモジュールとリソースによっては、適切なフェイルオーバーを容易に行うために、一部のオブジェクトを共有ストレージに置く必要が生じることがあります。場合によっては、Generic Application Recovery Kit またはカスタム Recovery Kit を使用して、モジュールを個別に保護しなくてはならないこともあります。
9. Apache で SSL (Secure Sockets Layer) モジュールを使用する場合は、Apache サーバをパスワードで保護しないでください。Web サーバをパスワードで保護すると、デーモンが起動するたびに管理者が手作業でパスワードを入力しなければなりません。手作業での手順は、リカバリ時間が非常に重要な高可用性環境と調和しないため、LifeKeeper ではパスワードで保護されたインスタンスはサポートされていません。次のコマンドを使用してパスワードを削除してください。

```
openssl rsa -in server.key -out unprotected_server.key
```

入力を要求されたら、サーバ・キーのパスワードを入力します。サイトのセキュリティを守るため、サーバ・キー・ファイルは root だけが読めるようにしてください。

```
chmod 400 unprotected_server.key
```

Apache インスタンスの階層作成時、LifeKeeper Apache Recovery Kit はリソースがパスワードで保護されていないかチェックします。リソースがパスワードで保護されている場合、エラー・メッセージとともに階層の作成は失敗します。しかし、インスタンスが別のサーバに拡張される際には、LifeKeeper Apache Recovery Kit はバックアップ・サーバ上のパスワード保護をチェックしません。拡張しようとしている階層がパスワードで保護されていないことを確認する必要があります。

サーバ・キー・ファイル (Apache 設定ファイルの `SSLCertificateKeyFile` ディレクトティブで指定されています) の名前と場所は、クラスタ内の全サーバで同じにする必要があります。

注意: LifeKeeper が使用する httpd プロセスの PID ファイル名は以下のフォーマットになります。:

```
"/var/run/httpd.<TAG name>.pid"
```

この PID ファイル名は OS が使用するデフォルトの PID ファイル名とは異なります。PID (例. log rotate)ファイルの参照が必要な場合は LifeKeeper PID ファイル名およびフォーマットに気をつけてください。

LifeKeeper 設定作業

LifeKeeper GUI からは、以下の設定作業を実行できます。以下の 4 つの作業は、Apache リソース・インスタンス特有のもので、Recovery Kit ごとに異なるため、本書で説明しています。

- [リソース階層の作成](#)：アプリケーション・リソース階層をLifeKeeperクラスタに作成します。
- [リソース階層の拡張](#)：リソース階層をプライマリ・サーバからバックアップ・サーバへ拡張します。
- [リソース階層の拡張解除](#)：LifeKeeperクラスタ内の 1 つのサーバから、リソース階層を拡張解除（削除）します。
- [リソース階層の削除](#)：リソース階層をLifeKeeperクラスタ内のすべてのサーバから削除します。

以下の作業については、すべての Recovery Kit で同じ手順を使用する共通の作業であるため、『LifeKeeper Online Product Manual』の「GUI Administration」セクションで説明しています。

- **リソース従属関係の作成**：既存のリソース階層と、別のリソース・インスタンスとの間に親子の従属関係を作成し、クラスタ内のすべての対象サーバに従属関係の変化を伝達します。
- **リソース従属関係の削除**：リソースの従属関係を削除して、クラスタ内のすべての対象サーバに従属関係の変化を伝達します。
- **サービスの提供**：特定のサーバでリソース階層をサービス中にします。
- **サービス休止**：特定のサーバでリソース階層のサービスを休止します。
- **プロパティの表示／編集**：特定のサーバでリソース階層のプロパティを表示または編集します。

注：このセクションの残りの部分では、LifeKeeper GUI の[Edit]メニューから作業を選択することによって、Recovery Kit を設定する方法を説明します。各設定作業は、ツールバーから選択することもできます。ステータス表示ウィンドウの[Resource Hierarchy Tree]（左側のペイン）に表示されるグローバル・リソースを右クリックし、[Edit]メニューと同じドロップダウン・メニューの選択項目を表示することもできます。もちろん、この方法を利用できるのは階層がすでに存在する場合のみです。

ステータス表示ウィンドウの[Resource Hierarchy Table]（右側のペイン）に表示されるリソース・インスタンスを右クリックしても、サーバや特定のリソースの状態に応じて、リソース階層の作成を除くすべての設定作業を実行できます。

Apache Web Server リソース階層の作成

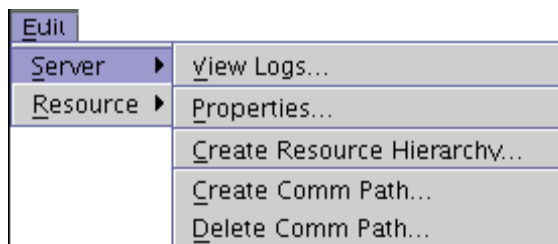
重要：

Apache Web Server リソース階層を作成する前に、LifeKeeper が保護している既存の IP リソースが Apache の設定ファイルに記述されていることを確認してください。

Web ドキュメントや設定ファイルが共有ディスク上にある共有環境では、共有ファイル・システムがマウントされていることを確認してください。リソースをバックアップ・サーバに拡張するには、機能している通信パス（つまりハートビート）が必要であるという点も重要です。

プライマリ・サーバからリソース・インスタンスを作成するには、以下の手順を実行します。

1. [LifeKeeper GUI]メニューから[編集]を選択し、次に[サーバ]を選択します。メニューから[リソース階層の作成]を選択します。



リソース作成時は Apache Web Server を動作させないでください。しかし、システム設定ファイルで Listen 変数をセットアップしている場合は、デフォルトのデーモンの実行は許可できます。

リソース階層の作成ウィザードダイアログが表示され、クラスタ内にインストールされた全ての有効なリカバリキットがドロップダウンリストボックスに表示されます。

2. [Apache Web Server]を選択し、[Next]をクリックします。
3. 次の表に示す情報を入力するように求められます。どのダイアログ・ボックスでも、[Back]がアクティブになっているときには、直前のダイアログ・ボックスに戻ることができます。これは、前に入力した情報を修正する必要があるエラーが発生した場合に便利です。[Cancel]をクリックすると、いつでも作成処理全体を取り消すことができます。

フィールド	ヒント
Switchback Type	[intelligent]または[automatic]を選択します。スイッチバック・タイプの選択は、フェイルオーバー後にサーバが復帰したときに、このサーバに Apache インスタンスを戻す方法を指定します。スイッチバック・タイプは後で、[Resource Properties]ダイアログ・ボックスの[General]タブから変更できます。
Server	Apache Web Server を配置するサーバを選択します（通常これは、プライマリ・サーバまたはテンプレート・サーバと呼ばれます）。クラスタ内のすべてのサーバがドロップダウン・リスト・ボックスに表示されます。
Web Server Binary Location	httpd Apache Web Server デーモンのフルパス名を選択するか、ファイル名を含めて入力します。デフォルトは/usr/sbin/httpd です。
Web Server Root Directory	Apache Web Server のルート・ディレクトリはフルパスで入力する必要があります。相対パスやシンボリック・リンクを使用することはできません。Apache Web Server の設定ファイルはルート・ディレクトリを基点として conf/httpd.conf に配置されます。 注： LifeKeeper はこの時点で、使用できる保護対象の IP リソースがあるかどうかをチェックします。Apache Web Server リソース階層を作成するために有効なデータが提供されたかどうかを検証します。これらのチェックのどちらかで問題が検出されると、[ERROR]ボックスが表示されます。サーバ・ルート・ディレクトリのパスは有効であっても、Apache の設定自体にエラーがある場合、作業を中断してこれらのエラーを修正し、階層の作成を続けることができます。作業を中断して、必要な LifeKeeper の IP リソースを作成することもできます。
Root Tag	Apache Web Server 階層に付けるタグ名を選択するか入力します。デフォルトの[apache<ルート・ディレクトリ>]を選択するか、独自のタグ名を入力することができます。

4. **[Create]**をクリックします。Create Resource Wizard で、Apache リソース階層が作成されます。LifeKeeper によって入力したデータが検証されます。問題が検出された場合、情報ボックスにエラー・メッセージが表示されます。
5. 情報ボックスが表示され、Apache リソース階層が正常に作成されたことが示されます。フェイルオーバーでの保護を行うには、その階層をクラスタ内の別のサーバに拡張する必要があります。**[Next]**をクリックします。

注：新しい Apache インスタンスが正しく起動されなかったことを示すエラー・メッセージが表示される場合があります。新しい Apache 階層は、別のシステムに拡張する前に起動されて（サービス中になって）いる必要があります。この時点で作業を中断し、表示されたエラー・メッセージに基づいて問題を修正してから、新しい階層をサービス中にして階層の拡張に進むことができます。

6. **[Continue]**をクリックします。Pre-Extend Wizard が起動されます。リソース階層を別のサーバに拡張する方法の詳細については、後述の「階層の拡張」の手順 2 を参照してください。

[Cancel]をクリックするとダイアログ・ボックスが表示され、Apache リソース階層を LifeKeeper の保護下に置くには、いずれかの時点でこの手順に戻り、階層を別のサーバに拡張する必要があるという警告が表示されます。

階層の拡張

この操作は**[Edit]**メニューから開始できます。または、**[Create Resource Hierarchy]**オプションが完了すると自動的に開始されます。この場合は下の手順 2 を参照してください。

1. **[Edit]**メニューの**[Resource]**から**[Extend Resource Hierarchy]**を選択します。Pre-Extend Wizard が表示されます。拡張作業に不慣れな場合は、**[Next]**をクリックします。LifeKeeper の**[Extend Resource Hierarchy]**作業のデフォルト値をよく知っていて、入力や確認を省略する場合は、**[Accept Defaults]**をクリックします。
2. Pre-Extend Wizard で、以下の情報の入力を求められます。
注：最初の 2 つのフィールドは、**[Edit]**メニューの**[Extend]**から作業を開始した場合のみ表示されます。

フィールド	ヒント
Template Server	現在 Apache リソースがサービス中のテンプレート・サーバを入力します。ここで選択する [Template Server] と次のダイアログ・ボックスで選択する [Tag to Extend] では、サービス中のリソース階層を表すことに注意してください。選択したテンプレート・サーバでサービス中でないリソース・タグを選択すると、エラー・メッセージが表示されます。このダイアログのドロップダウン・ボックスには、クラスタ内の全サーバの名前が表示されます。
Tag to Extend	テンプレート・サーバからターゲット・サーバに拡張する Apache Web Server インスタンスの名前を選択します。ドロップダウン・リスト・ボックスには、前のダイアログ・ボックスで選択したテンプレート・サーバ上に作成した全リソースの一覧が表示されます。
Target Server	Apache Web Server リソース階層の拡張先のターゲット・サーバを選択します。ドロップダウン・ボックスには、クラスタ内で、選択した階層にまだ属していないサーバの名前が表示されます。

Switchback Type	[intelligent]または[automatic]を選択します。スイッチバック・タイプの選択は、バックアップ・サーバへのフェイルオーバー後にサーバが復帰したときに、このサーバに Apache Web Server インスタンスを戻す方法を指定します。インテリジェント・スイッチバックでは、インスタンスをプライマリ/オリジナル・サーバに戻すために管理者の操作が必要です。自動スイッチバックは、プライマリ・サーバがオンラインに戻り、LifeKeeper との通信パスが再確立されるとすぐにスイッチバックが行われることを意味します。
Template Priority	テンプレート階層の優先順位を選択または入力します。1~999 の範囲で、まだ優先順位として使用されていない値が有効です。小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当します)。他のシステムですでに使用されている優先順位は、この階層に指定しても拒否されず。デフォルト値を使用することをお勧めします。 注: この選択画面は階層を初めて拡張するときだけ表示されます。
Target Priority	拡張する Apache Web Server リソースの、ターゲットの優先順位を選択または入力します。優先順位は、リソースの連鎖フェイルオーバー・シーケンスにおけるサーバの優先順位を示す 1~999 の数字です。階層の優先順位は数字の順にソートされ、小さい数字ほど優先順位が高くなります (数字 1 が最高の優先順位に相当します)。LifeKeeper は、階層が作成されたサーバに自動的に「1」を割り当てることに注意してください。優先順位は連続している必要はありませんが、特定のリソースについて 2 つのサーバに同じ優先順位を割り当てることはできません。
	拡張前のチェックが成功したことを示すメッセージが表示されたら、[Next]をクリックします。 拡張する階層に応じて、拡張されるリソース・タグ (編集不可) を示す一連の情報ボックスが表示されます。[Extend]をクリックします。
Network Interface	ネットワーク・インタフェースを選択または入力します。これは、IP リソースがターゲット・サーバで使用するネットワーク・インタフェース (つまりイーサネット・カード) の名前です。
Backup Interface	IP リソースの拡張先サーバで IP ローカル・リカバリ機能を使用する場合に、バックアップ・インタフェースを選択します。デフォルト値は[none]ですが、このサーバに別のネットワーク・インタフェース・カードが設定されている場合、ドロップダ

	ウン・リストに表示されます。
IP Resource Tag	IP リソース・タグを選択または入力します。これは、ターゲット・サーバに拡張される IP リソースが使用するリソース・タグ名です。
Root Tag	ルート・タグを選択または入力します。これは、Apache Web Server 階層に与えられるタグ名です。デフォルトでは、ルート・タグ名はテンプレート・サーバでもターゲット・サーバでも同じはずです。
Mount Point	これは、サーバ・ルート・ディレクトリが共有ファイル・システム上にある場合にのみ表示されます。 Web サーバ・ルート・ディレクトリが置かれている共有ファイル・システムのマウント・ポイントを選択または入力します。共有 Web サーバ・ルート・ディレクトリのマウント・ポイントは、テンプレート・サーバでもターゲット・サーバでも同じはずです。ほとんどの場合、ダイアログ・ボックスに表示されるデフォルトのマウント・ポイントを選択する必要があります。
Root Tag	これは、サーバ・ルート・ディレクトリが共有ファイル・システム上にある場合にのみ表示されます。 ルート・タグを選択または入力します。これは、共有ファイル・システムのタグ名です。

3. 拡張を実行中であることを確認する情報ボックスが表示されます。同じ Apache リソース・インスタンスをクラスタ内の別のサーバに拡張する場合は、**[Next Server]**をクリックします。その場合は、Extend Resource Hierarchy の操作が繰り返されます。

[Finish]をクリックすると、Apache Web Server リソースの拡張が正常に完了したことを示すメッセージが表示されます。

4. 最後のダイアログ・ボックスで**[Done]**をクリックして、**[Extend Resource Hierarchy]**メニューの選択を終了します。

注：新しいインスタンスの機能は、必ず「両方の」サーバでテストしてください。

階層の拡張解除

1. **[Edit]**メニューの**[Resource]**から**[Unextend Resource Hierarchy]**を選択します。
2. Apache Web Server リソースを拡張解除するターゲット・サーバを選択します。Apache Web Server が現在サービス中のサーバは選択できません（右側ペインのリソース・インスタンスを右クリックして**[Unextend]**作業を選択した場合、このダイアログ・ボックスは表示されません）。

[Next]をクリックします。

3. 拡張解除する Web サーバ階層を選択して、**[Next]**をクリックします（左右いずれか

のペインのリソース・インスタンスを右クリックして[Unextend]作業を選択した場合、このダイアログ・ボックスは表示されません。

4. 拡張解除のために選択したターゲット・サーバと Apache Web Server リソース階層を確認する情報ボックスが表示されます。[Unextend]をクリックします。
5. Apache Web Server リソースが正常に拡張解除されたことを確認する別の情報ボックスが表示されます。
6. [Done]をクリックして終了します。

リソース階層の削除

留意する必要があるのは、リソース階層をサービス休止にする前に削除すると、そのリソース階層は LifeKeeper の保護から外されますが、Apache インスタンスは、手動で停止するかシステムを再起動しないと現在アクティブなサーバで動作し続けるということです。異なる IP アドレスを使って同じ Apache 階層を再作成しようとしたり、以前使用した IP アドレスを使って（ただし異なるサーバ・ルート・ディレクトリを使用して）新しい Apache 階層を作成しようとする、同じアドレスで動作し続けている Apache インスタントとの競合が生じます。

LifeKeeper 環境内のすべてのサーバからリソース階層を削除するには、以下の手順を完了します。

1. [LifeKeeper GUI]メニューから[Edit]を選択し、次に[Resource]を選択します。ドロップダウン・メニューから、[Delete Resource Hierarchy]を選択します。
2. Apache Web Server リソース階層を削除するターゲット サーバの名前を選択します。[Next]をクリックして、次のダイアログ・ボックスに進みます。

注：右側のペインで個々のリソース・インスタンスを右クリックするか、リソースが 1 つのサーバ上だけにある場合に左側のペインでグローバル・リソースを右クリックして[Delete Resource]作業を選択した場合、[Target Sever]ダイアログ・ボックスは表示されません。

3. [Hierarchy to Delete]を選択します（左右いずれかのペインにあるリソース・インスタンスを右クリックして[Delete Resource]作業を選択した場合、このダイアログ・ボックスは表示されません）。リスト・ボックスには、ターゲット・サーバ上のすべての階層、つまりサービス中の階層とサービス休止中の階層の「両方」が表示されることに注意してください。Apache インスタンスを停止し、リソース階層を LifeKeeper の保護から外す場合、階層を削除する前に、選択した階層がサービス休止になっていることを確認する必要があります。

[Next]をクリックします。

4. ターゲット・サーバの選択と、削除の対象として選択した階層を確認する情報ボックスが表示されます。[Next]をクリックします。
5. Apache Web Server リソースが正常に削除されたことを示す別の情報ボックスが表示されます。
6. [Done]をクリックして終了します。

リソース階層のテスト

手動切り替えを開始することによって、Apache リソース階層をテストできます。これは、プライマリ・サーバからバックアップ・サーバへのリソース・インスタンスのフェイルオーバーをシミュレートします。

GUI からの手動切り替えの実行

手動切り替えを開始するには、[LifeKeeper GUI] メニューから [Edit] を選択し、次に [Resource] を選択し、最後にドロップダウン・メニューから [In Service] を選択します。たとえば、バックアップ・サーバで [In Service] 要求を実行すると、アプリケーション階層はバックアップ・サーバでサービス中になり、プライマリ・サーバでは停止されます。この時点で、元のバックアップ・サーバがプライマリ・サーバで、元のプライマリ・サーバがバックアップ・サーバになっています。

[Out of Service] 要求を実行すると、アプリケーションは他のサーバでサービス中にならずに停止されます。

リカバリ操作

プライマリ・サーバに障害が発生すると、Apache Recovery Kit ソフトウェアは以下の作業を実行します。

- バックアップ・サーバの 1 つまたは複数の物理ネットワーク・インタフェースの IP アドレスをサービス中にすることによって、Apache をバックアップ・サーバでサービス中にします。
- ファイル・システムが使用されている場合、バックアップ・サーバの共有ディスクにファイル・システムをマウントします。
- Apache に関連するデーモン・プロセスを開始します。

リカバリ後、Apache Web Server のユーザは、ブラウザの再ロード／リフレッシュ・ボタンをクリックして再接続することができます。

トラブルシューティング

このセクションでは、LifeKeeper Apache Web Server リソース階層の作成と拡張、およびリソースの削除と復元の処理中に発生する可能性があるメッセージの一覧を示し、必要に応じて、エラーの原因とエラー状態を解決するために必要な処置について追加説明します。他の LifeKeeper スクリプトやユーティリティから、その他のメッセージが生成される可能性もあります。その場合は、実際のスクリプトまたはユーティリティのマニュアルを参照してください。このセクションのメッセージは、以下のトピックに分類されています。

- [階層の作成](#)
- [階層の拡張](#)
- [階層の削除、復元、リカバリ](#)

階層作成時のエラー

Apache 階層の作成中に表示される可能性のあるエラー・メッセージを、各メッセージの推測される説明とともに以下に示します。LifeKeeper コアおよび他の Recovery Kit によって表示されるエラー・メッセージは、このドキュメントには記載していません。作業を中断してここで説明する問題を修正したら、作業を中断した段階から、Apache の設定に必要な新しい LifeKeeper リソースの作成を含め、階層の作成を続行できることに注目してください。

Apache Web Server 実行形式ファイルの場所の検証中に発生するエラー

「**Error: valid_httpd_path: Must specify absolute path to httpd executable.**」(エラー : **valid_httpd_path** : 実行形式ファイル **httpd** への絶対フルパス名を指定しなければなりません。)

有効な Apache 実行形式ファイル **httpd** への絶対フルパス名を入力してください。

「**Error: valid_httpd_path: File does not exist at path specified.**」(エラー : **valid_httpd_path** : 指定したパスにファイルが存在しません。)

指定した場所に有効な Apache 実行形式ファイル **httpd** が存在しません。

「**Error: valid_httpd_path: Httpd failed to display Server version.**」(エラー : **valid_httpd_path** : **httpd** はサーバ・バージョンの表示に失敗しました。)

指定した場所にある実行形式ファイル **httpd** は、標準的な Apache の「サーバ・バージョン」を表示できません。

「**Error: valid_httpd_path: Incorrect version \$MAJOR.\$MINOR.\$POINT of Apache at \$HTTPD_PATH.**」(エラー : **valid_httpd_path** : **\$HTTPD_PATH** にある Apache のバージョン **\$MAJOR.\$MINOR.\$POINT** は正しくありません。)

指定した場所にある Apache 実行形式ファイル **httpd** は、正しくない「サーバ・バージョン」を表示しています。

サーバ・ルート・ディレクトリの検証中に発生するエラー

「**Error: valid_http_root: Cannot find Apache configuration file at \$CONF_FILE.**」 (エラー: **valid_http_root : \$CONF_FILE** で Apache 設定ファイルが見つかりません。)

指定したサーバ・ルート・ディレクトリを基準にした `conf/httpd.conf` に、有効な Apache 設定ファイルが存在する必要があります。SuSE にデフォルトで Apache をインストールすると、`conf` という名前のサーバ・ルートのサブディレクトリには、設定ファイル `httpd.conf` が配置されないことに注意してください。SuSE に Apache をデフォルトでインストールしている場合は、設定ファイルの場所を `/etc/httpd/conf` ディレクトリに変更する必要があります。

「**Error: valid_http_root: Must specify absolute path to Apache server root directory.**」 (エラー: **valid_http_root : Apache** サーバ・ルート・ディレクトリへの絶対フルパス名を指定しなければなりません。)

サーバ・ルート・ディレクトリへの絶対フルパス名を入力してください。

「**Error: valid_http_root: Apache instance at \$HTTP_ROOT is already under LifeKeeper protection.**」 (エラー: **valid_http_root : \$HTTP_ROOT** にある Apache インスタンスはすでに **LifeKeeper** の保護下にあります。)

各インスタンスは固有で一意的なサーバ・ルート・ディレクトリを持ち、設定ファイルが `conf/httpd.conf` に置かれている必要があります。指定したサーバ・ルート・ディレクトリは別の Apache インスタンスがすでに使用しています。

「**Syntax error on line <行番号> of <設定ファイル・パス>, etc...**」 (<設定ファイル・パス>の<行番号>にシンタックス・エラー、...)

Apache 設定ファイルでシンタックス・エラーが見つかりました。このようなエラー・メッセージは、`httpd -T` コマンドを使用して `$CONF_FILE` のシンタックスをチェックした場合には表示されます。詳細については、表示されたエラー・メッセージの内容を参照してください。

「**Error: valid_http_root: Since \$HTTPD_PATH is shareable on \$HTTPD_PATH_SHARED, \$HTTP_ROOT must be also.**」 (エラー: **valid_http_root : \$HTTPD_PATH** が **\$HTTPD_PATH_SHARED** 上で共有可能なので、**\$HTTP_ROOT** も共有可能でなければなりません。)

実行形式ファイル `httpd` が共有／共有可能ストレージ上にある場合、サーバ・ルート・ディレクトリとすべてのドキュメント・ルート・ディレクトリも同じストレージ上になければなりません。

「**Error: valid_http_root: Since \$HTTP_ROOT is shareable on \$HTTP_ROOT_SHARED, all document root directories must shareable on this same filesystem.**」 (エラー: **valid_http_root : \$HTTP_ROOT** が **\$HTTP_ROOT_SHARED** 上で共有可能なので、同じファイル・システム上ですべてのドキュメント・ルート・ディレクトリが共有可能でなければなりません。)

サーバ・ルート・ディレクトリが共有／共有可能ストレージ上にある場合、すべてのドキュメント・ルート・ディレクトリも同じストレージ上になければなりません。

「**Error: http_docs_shared: Since one/more Apache document root directories are shareable**

