

SIOS Protection Suite for Linux in the AWS Cloud (SAP)

アプリケーション設定ガイド

2020年1月

SIOS Technology Corp.

本書およびその内容は SIOS Technology Corp. (旧称 SteelEye® Technology, Inc.) の所有物であり、許可なき使用および複製は禁止されています。SIOS Technology Corp. は本書の内容に関していかなる保証も行いません。また、事前の通知なく本書を改訂し、本書に記載された製品に変更を加える権利を保有しています。SIOS Technology Corp. は、新しい技術、コンポーネント、およびソフトウェアが利用可能になるのに合わせて製品を改善することを方針としています。そのため、SIOS Technology Corp. は事前の通知なく仕様を変更する権利を保有します。

LifeKeeper、SteelEye、および SteelEye DataKeeper は SIOS Technology Corp. の登録商標です。

本書で使用されるその他のブランド名および製品名は、識別のみを目的として使用されており、各社の商標が含まれています。

出版物の品質を維持するために、弊社は本書の正確性、明瞭性、構成、および価値に関するお客様のご意見を歓迎いたします。

以下の宛先に電子メールを送信してください。

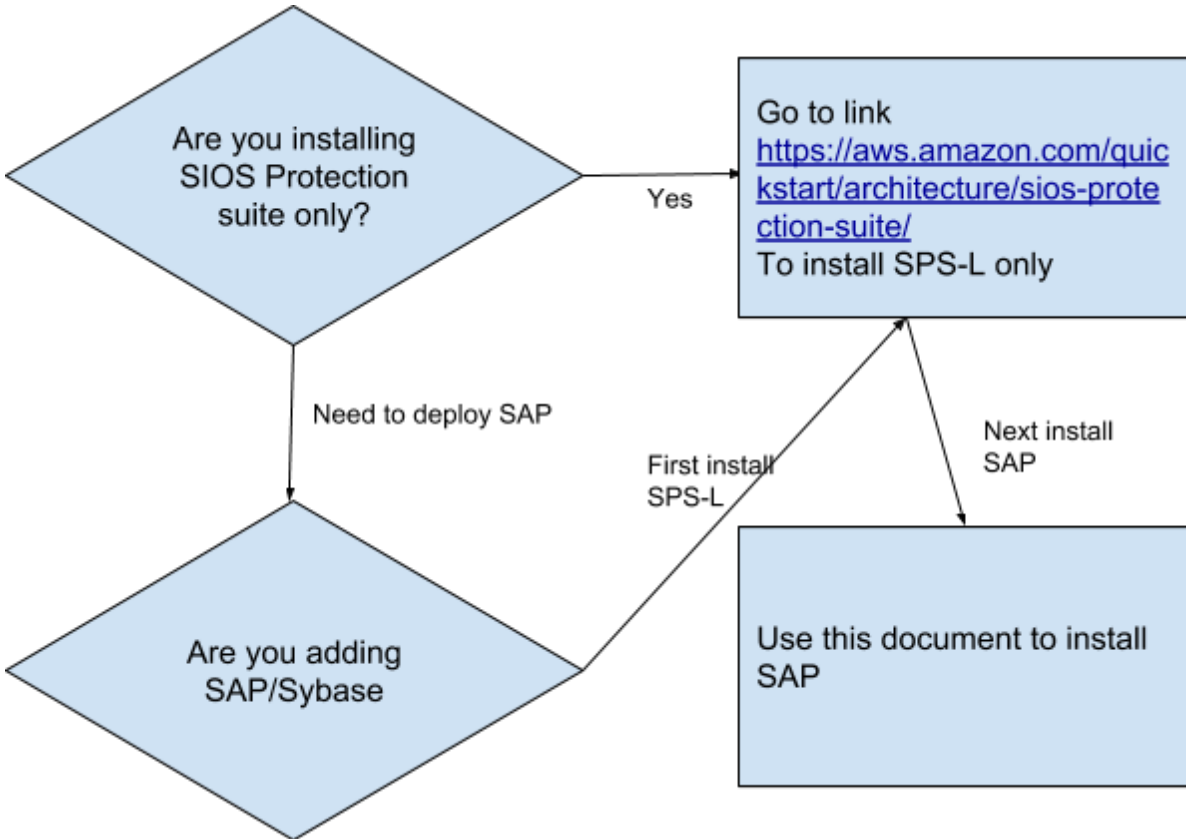
ip@us.sios.com
Copyright © 2020
By SIOS Technology Corp.
San Mateo, CA U.S.A.
All rights reserved

目次

概要.....	4
SPS 上で SAP を設定する際の追加手順.....	5
ステップ 1. デプロイメントのテスト.....	5
VNC サーバーをインストールするためのコードスニペット.....	5
ステップ 2. 仮想 IP の設定.....	6
Amazon AWS Elastic Compute Cloud (EC2) のセットアップ.....	6
仮想 IP リソースの作成.....	7
ステップ 3. SAP のセットアップ.....	10
NFS なしの ASCS.....	11
一般的なセットアップのステップ概要.....	11
SAP のインストール.....	12
SAP リソース階層の作成.....	13
NFS を使用している ASCS + ERS.....	19
一般的なセットアップのステップ概要.....	19
SAP のインストール.....	20
NFS のセットアップ.....	22
NFS リソース階層の作成.....	26
SAP リソース階層の作成.....	30
ERS リソースの作成.....	35
ENSAv2/ERSv2 使用時に ASCS/ERS に回避動作を強制する.....	37
スイッチオーバーおよびフェールオーバーのテスト.....	47
その他のリソース.....	49
AWS サービス.....	49
SIOS Protection Suite for Linux.....	49
クイックスタート.....	50
フィードバック.....	50
ドキュメントの更新.....	50

概要

本書は、SIOS Protection Suite for Linux のインストール（SPS）のユーザーガイドです。
SPS for SAP 環境のインストール方法を理解するには、以下のマトリックスに従ってください。



注記： 図中のリンクを以下にも記載します。

<https://aws.amazon.com/quickstart/architecture/sios-protection-suite/>

SPS 上で SAP を設定する際の追加手順

SPS 上で SAP を設定するには、以下の手順に従ってください。

ステップ 1. デプロイメントのテスト

SPS-L ノードに接続するには、Windows JumpBox に接続する必要があります。Windows マシンに接続するには、リモートデスクトップ端末に接続してください。

AWS コンソールで、作成された Windows JumpBox ノードを選択し、**[Action]** の次に **[Connect]** をクリックします。これでリモートデスクトッププログラムをダウンロードして接続できます。また、マシンにログインするために必要なパスワードを複合化する必要があります。

Windows マシンに接続したら、PuTTY と VNC Viewer をダウンロードすることを推奨します。以下のサイトからダウンロードしてください。

- **PuTTY** - www.putty.org
- **VNC Viewer** - <https://www.realvnc.com/en/connect/download/viewer/>

これで、PuTTY を使用して各ノードのプライベート IP アドレスに接続することができます。また、同じプライベート IP アドレスを使用しているノードに接続するために VNC Viewer を使用することもできます。ノードへは Windows JumpBox の外ではアクセスできませんが、ノードは NAT ゲートウェイを使用してインターネットにアクセスできます。(注記 : NAT ゲートウェイに問題がある場合は、セキュリティグループのルール/メインルートを確認してください。)

いずれかのノードに接続したら、先ほどのテンプレートで作成したパスワードを使用して root で su を実行し、プログラム vncserver を実行します。これにより、VNC Viewer を使用してグラフィカルインターフェイス内のそのノードに接続できます。

VNC サーバーをインストールするためのコードスニペット

次のオプションを指定して、vncserver コマンドを root として実行します。

パスワードを入力し、確認のために繰り返します
読み取り専用パスワードを **No** に設定します

オプションで `/root/.vnc/config` を編集し、以下の記述を追加します。

```
securitytypes=none  
vncserver -kill:1
```

VNC へのアクセスは `ipv4:5901` です。5901 は指定されたポート番号です。

デスクトップを右クリックして **Open Terminal** をクリックし、LifeKeeper GUI に接続するコマンド `/opt/LifeKeeper/bin/lkGUIapp` を入力します。事前に設定したパスワードで root としてログインしてください。2つのノードが接続されていることがわかります。

この時点で、基本的な LifeKeeper 2 ノードがセットアップされました。LifeKeeper を使用して、SAP のインストールと SAP サービスの保護を続行します。

ステップ 2. 仮想 IP の設定

ノード上で SAP が設定されたので、引き続き LifeKeeper を設定して SAP サービスとファイルシステムを保護します。

Amazon AWS Elastic Compute Cloud (EC2) のセットアップ

AWS Command Line Interface (CLI) を各ノードにインストールする必要があります。詳細は、[AWS Command Line Interface のインストール](#) を参照してください。すべての EC2 インスタンスは、HTTP および HTTPS を使用して Amazon EC2 サービスエンドポイントにアクセスする必要があります。

Amazon EC2 インスタンスのメタデータを取得するには、HTTP プロトコルを使用して IP アドレス 169.254.169.254 にアクセスしてください。

LifeKeeper が AWS を操作するためには、以下のアクセス権を持つ IAM ユーザーまたは IAM ロールが必要です。EC2 インスタンスの root ユーザーからアクセスできるように、[EC2 IAM ロールを設定するか、AWS CLI を適切に設定してください](#)。

- `ec2:DisassociateAddress`
- `ec2:DescribeAddresses`
- `ec2:AssociateAddress`
- `ec2:DescribeRouteTables`
- `ec2:ReplaceRoute`

仮想 IP リソースの作成

IP アドレスを決定します。IP アドレスは、ノードの現在の IP の CIDR ブロック範囲外の IP アドレスである必要があります。IP アドレスは、ノードの VPC ルートテーブルに配置してください。

以下の図では eni- ネットワークアダプタを使用して IP アドレスを 10.1.0.10/32 に設定し、ノードの 1 つに関連付けました。

<input type="checkbox"/>	Name	Route Table ID	Explicitly Associat	Main	VPC
<input type="checkbox"/>	Private subnet 1A	rtb-0f25600d73f059...	0 Subnets	No	vpc-003113c78
<input checked="" type="checkbox"/>		rtb-0259a7dda97e3...	2 Subnets	Yes	vpc-003113c78
<input type="checkbox"/>	Private subnet 2A	rtb-0d48d96bfa35f3...	0 Subnets	No	vpc-003113c78
<input type="checkbox"/>	Public Subnets	rtb-09d48b85b3b18...	2 Subnets	No	vpc-003113c78

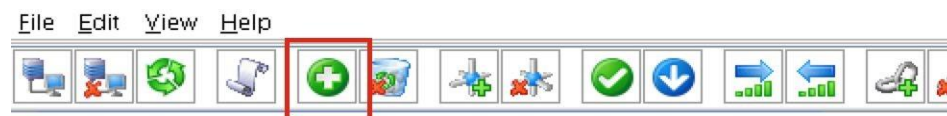
rtb-0259a7dda97e3fbd8

View: All rules

Destination	Target	Status	Propagated	Remo
10.0.0.0/16	local	Active	No	
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="nat-082059721765b9ac9"/>	Active	No	<input type="button" value="X"/>
<input type="text" value="10.1.0.10/32"/>	<input type="text" value="eni-012388209b2453af5"/>	Active	No	<input type="button" value="X"/>

/etc/default/LifeKeeper を編集して NOBCASTPING=1 を設定し、続行する前にブロードキャスト ping を無効にします。

緑色のプラスアイコンをクリックして、新規リソースを作成します。



Please Select Recovery Kit

ウィザードに従い、以下のように選択して IP リソースを作成します。

Select Recovery Kit: IP

Switchback Type: Intelligent

IP Resource: 10.1.0.10

Netmask: 255.255.255.0

Network Interface: eth0

IP Resource Tag: ip-10.1.0.10

以下のように選択して IP リソースを拡張します。

Switchback Type: Intelligent

Template Priority: 1

Target Priority: 10

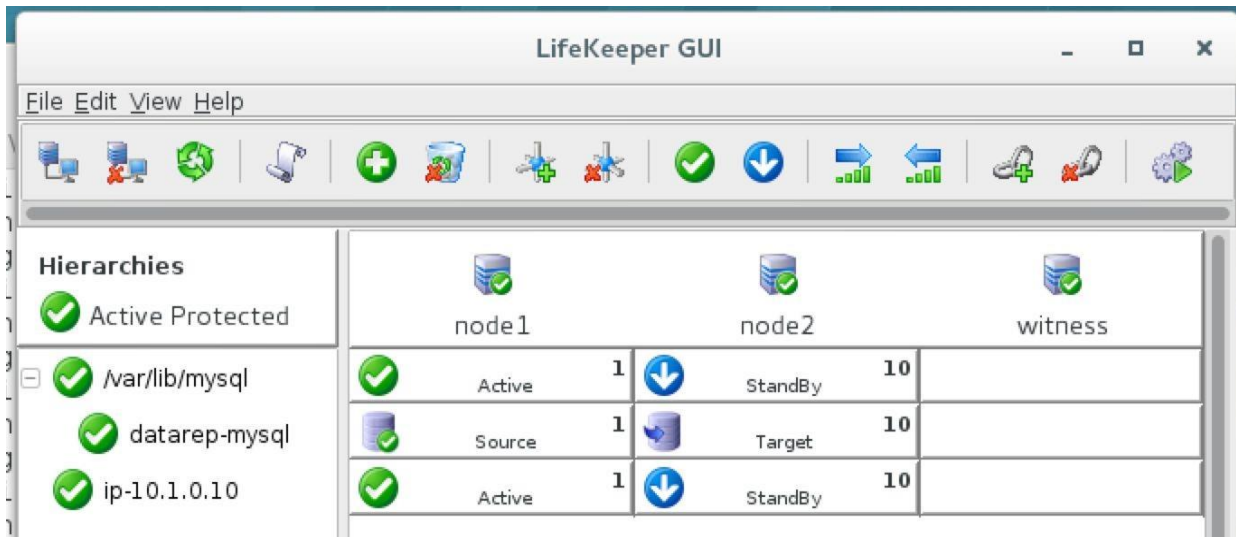
IP Resource: 10.1.0.10

Netmask: 255.255.255.0

Network Interface: eth0

IP Resource Tag: ip-10.1.0.10

クラスターは次のようになり、ミラーリソースと IP リソースの両方が作成されます。



ステップ 3. SAP のセットアップ

SAP ソフトウェアをダウンロードし、ノード上でセットアップします。SAP マーケットプレイスにアクセスして、各ノードに SAP ソフトウェアをダウンロードできます。

SAP のセットアップにはいくつかの選択肢があります。どれを実装するかは、コスト、経験、RAS（信頼性、可用性、サービスビリティ）などのさまざまな要因に依存します。

- NFS なしの ASCS
- 別々のインスタンス上の NFS を使用する ASCS + ERS
- ASCS と同じインスタンス上の NFS を使用する ASCS + ERS

各構成にはメリットとデメリットがあります。現場で SAP エキスパートと作業するか、SIOS プロフェッショナルサービスに加入して、お使いの環境に最適なものを決定することをお勧めします。

注記： 今後作成するドキュメントでは、EFS（AWS Elastic File System）と Cloudwatch を使用したオールインワンの HANA のインストールについて詳しく説明します。自動インストールのクイックスタートスクリプトやインストールを管理するための SAP Landscape Manager（LaMa）の使用も計画しています。

NFS なしの ASCS

一般的なセットアップのステップ概要

1. 仮想 IP を作成します。(先のステップで実施済み)
2. EC2 リソースを作成し、仮想 IP の依存関係として作成します。(先のステップで実施済み)
3. 「仮想 IP」に基づいた「仮想ホスト名」のノード 1 に SAP をインストールします。
4. ノード 1 で Stopsap を実行します。
5. LifeKeeper GUI を使用して仮想 IP をノード 2 に対して「In Service」にし、「仮想 IP」に基づいてノード 2 の「仮想ホスト名」に SAP をインストールします。
6. ノード 2 で Stopsap を実行し、両方のノード上でプロファイルファイルを変更します。(以下参照)
7. LifeKeeper GUI を使用して、仮想 IP をノード 1 で「In Service」にします。
8. SAP コンサルタントのアドバイスに従って、SAP に必要なマウントポイントの複製リソースを作成します。(先のステップで実施済み)
9. ノード 1 で Startsap を実行し、SAP が正常に動作していることを確認します。
10. /etc/default/LifeKeeper で、両方のノードの最後に次のエントリを追加します。

```
SAP_EXPERTMODE=1
SAP_NFS_CHECK_IGNORE=1
SAP_DB_CHECK_IGNORE=1
```

11. SPS セットアッププログラムを再実行して、SAP Recovery Kit を追加します。

```
./setup
```

Recovery Kit Selection Menu から SAP Recovery Kit をインストールしてください。矢印キーを使用して使用可能な Reco9.very Kit のメニューから SAP Recovery Kit を選択し、<スペース>で決定し、<Enter>を押してインストールを続行します。

12. SAP Recovery Kit ガイドに従って SAP リソースを作成します。

<http://docs.us.sios.com/spslinux/9.4.1/en/topic/sap-recovery-kit-administration-guide>

本書では簡素化した手順を説明します。

SAP のインストール

ASCS は、先のインストール手順でホストファイルに追加された「仮想 IP」に基づく「仮想ホスト名」に基づいてインストールする必要があります。まだ完了していない場合は、SAP のインストール前に必ず実行してください。

- SAPinst を起動する前に SAPinst プロパティ `SAPINST_USE_HOSTNAME` を設定して、必要な仮想ホスト名を指定します。(注記：SAPINST_USE_HOSTNAME 仮想 IP アドレスは後で LifeKeeper で SAP リソースを作成する際に使用するので、書き留めておいてください。)

以下を実行します。

```
./sapinst SAPINST_USE_HOSTNAME={hostname}
```

- 7 段階にわけて、**Core サービス**を作成して開始する必要があります。`jdbconnect.jar` で権限エラーが発生した場合は、`/sapmnt/STC/exe/uc/linuxx86_64` に移動し、このディレクトリでも `jdbconnect.jar` を書き込み可能にしてください (`chmod 777 ---`)。

ASCS プロファイルは、`/usr/sap`、`sapmnt` を含むローカルマウントポイント、またはご利用の環境の SAP ファイルに必要なその他のローカルマウントポイントを指す必要があります。

ASCS インスタンスのインスタンスプロファイルでは、サービスが自動的に再開されて ASCS の "sapstartsrv" を再起動しなければならなくなることを防ぐために、エントリを変更する必要があります。たとえば、

```
Autostart = 1
Restart_Program_00 = local $_ER pf=$_PFL NR=$(SCSID)
```

を次のように変更します。

```
Autostart = 0
Start_Program_00 = local $_ER pf=$_PFL NR=$(SCSID)
```

1. ノード 1 で `Sapstop` を実行して SAP を起動します。
2. LifeKeeper GUI を使用して、ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、ノード 2 を選択して IP をノード 2 に切り替えます。

3. ステップ 1 を繰り返してノード 2 に SAP をインストールし、正しく実行できることを確認します。
4. ノード 2 で Sapstop を実行します。
5. LifeKeeper GUI を使用して ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、node1 を選択して IP をノード 1 に戻します。
6. ノード 1 で Sapstart を実行し、正しく実行できることを確認します。

SAP リソース階層の作成

1. LifeKeeper GUI メニューから **[Edit]** を選択し、次に **[Server]** を選択します。ドロップダウンメニューから、**[Create Resource Hierarchy]** 選択します。

クラスター内で認識されたすべてのインストール済みのリカバリーキットがドロップダウンリストで表示されます。ドロップダウンリストから **SAP** を選択します。

Please Select Recovery Kit ▼

[Next] をクリックします。

いずれかのダイアログボックスで **[Back]** ボタンが有効になっている場合は、前のダイアログボックスに戻ることができます。これは、以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。

階層の作成中に **[Cancel]** をクリックすると、LifeKeeper は作成プロセス全体をキャンセルします。

2. **スイッチバックタイプ**を選択します。これは、バックアップサーバーへのフェールオーバー後にサービスに復帰したときに SAP インスタンスをこのサーバーにどのようにスイッチバックするかを指示します。**Intelligent** または **Automatic** のいずれかを選択できます。**Intelligent スイッチバック**では、インスタンスをプライマリ/元のサーバーに戻すために管理者の介入が必要です。**Automatic ス**

スイッチバックとは、プライマリサーバーがオンラインに戻り、LifeKeeperのコミュニケーションパスを再確立するとすぐにスイッチバックが実行されることを意味します。



Switchback Type

スイッチバックタイプは必要に応じて、**[Resource Properties]** ダイアログボックスの **[General]** タブで後から変更できます。

[Next] をクリックします。

3. SAP PAS、ASCS または SCS を配置するサーバーを選択します（通常、これをプライマリまたはテンプレートサーバーと呼びます）。クラスター内のすべてのサーバーがドロップダウンリストボックスに含まれています。



Server

4. **SAP SID** を選択します。これは、保護対象の SAP PAS、ASCS、または SCS システムのシステム識別子です。



SAP SID

[Next] をクリックします。

5. 保護対象の SID の SAP インスタンス名（例：ASCS <No.>）（コアインスタンスが先頭）を選択します。

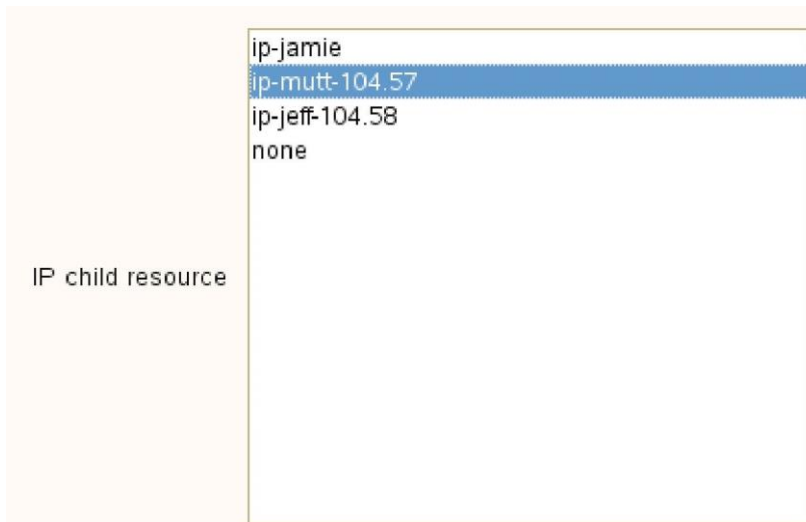


SAP Instance for PRS

[Next] をクリックします。

注記： 保護およびリカバリーレベルのカスタマイズに関連する追加の画面が表示される場合があります。

6. **IP の子リソース**を選択します。これは通常、SAP インストール時に指定された仮想ホスト IP アドレス（SAPINST_USE_HOSTNAME）またはフェールオーバーに必要な IP アドレスのいずれかです。

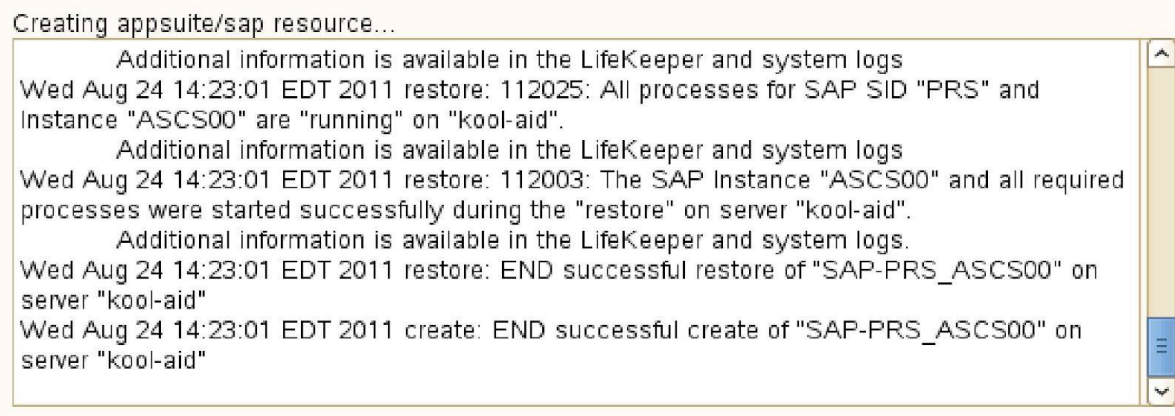


7. **SAP Tag** を選択または入力します。これは LifeKeeper が SAP の階層に付与するタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。デフォルトのタグは `SAP-<SID>_<ID>` です。

SAP Tag

[Create] をクリックすると、**SAP リソースの作成ウィザード**が SAP リソースを作成します。

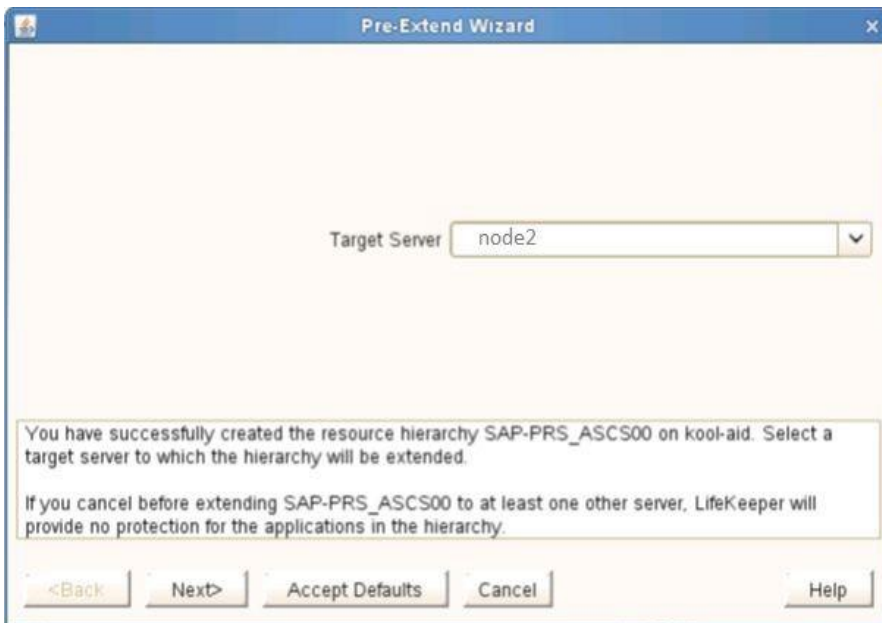
8. この時点で情報ボックスが表示され、LifeKeeper は SAP リソース階層を作成するための有効なデータが提供されているかどうかを検証します。LifeKeeper が問題を検出すると、情報ボックスに **ERROR** と表示されます。検証が成功すると、リソースが作成されます。情報ボックスに表示される SAP 起動スクリプトから出力されるエラーまたはメッセージが表示される場合もあります。



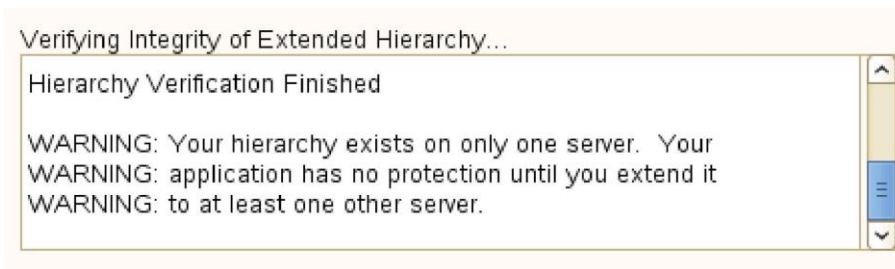
[Next] をクリックします。

9. SAP リソース階層を正常に作成したことを示す別の情報ボックスが表示されます。そして、LifeKeeper 保護下に置くためにその階層をクラスター内の別のサーバーに**拡張**する必要があります。

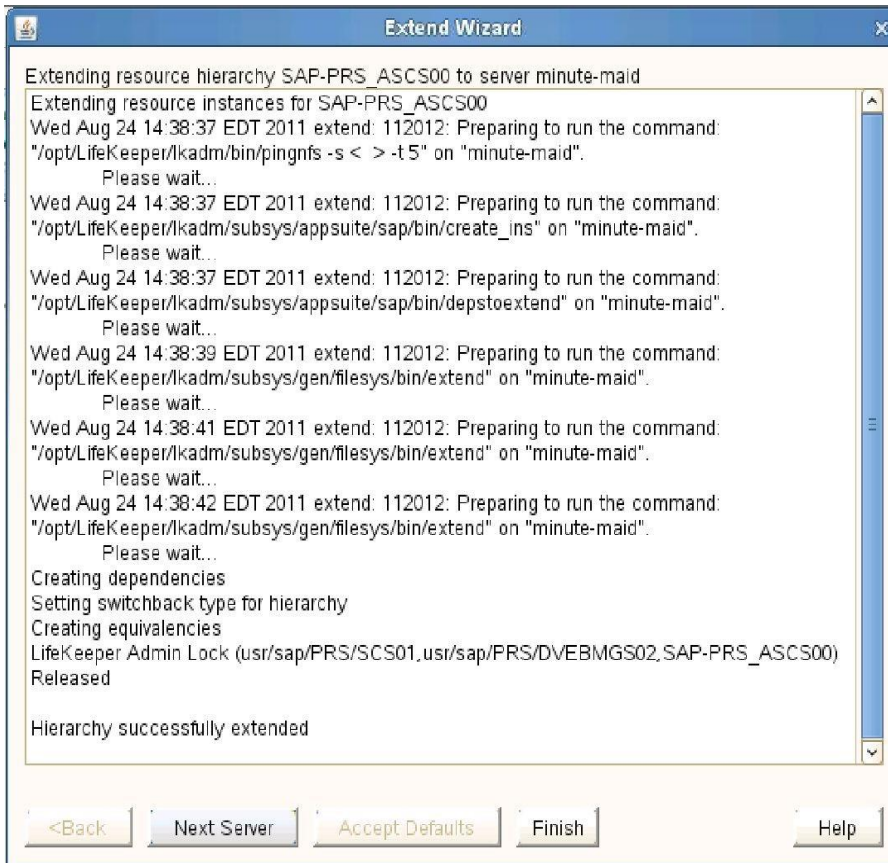
[Next] をクリックすると、LifeKeeper はこのセクションで後述する **Pre-Extend Wizard** を起動します。



ここで [Cancel] をクリックすると、後で SAP リソース階層を LifeKeeper の保護下に置くにはこのステップに戻って SAP リソース階層を別のサーバーに拡張する必要があることを警告するダイアログボックスが表示されます。



10. 階層が正常に拡張されたことを示す **[Extend Wizard]** ダイアログが表示されます。 **[Finish]** をクリックします。



[Hierarchy Integrity Verification] ダイアログが表示されます。階層の検証が終了したら、 **[Done]** をクリックして **[Create Resource Hierarchy]** メニューを終了します。

トップレベルの Core を持つ階層



SIOS Protection Suite を使用して PAS および AAS サーバーを保護することはできますが、ほとんどのお客様は、それらを追加の HA なしで独立したスタンバイサーバーとして使用しているだけです。このガイドでは保護手順については説明しないので、詳細と手順については SAP Recovery Kit を参照してください。

<http://docs.us.sios.com/spslinux/9.4.1/en/topic/sap-recovery-kit-administration-guide>

NFS を使用している ASCS + ERS

一般的なセットアップのステップ概要

1. 仮想 IP を作成し（先のステップでノード 1 で実施済み）、拡張します（先のステップで実施済み）。
2. EC2 リソースを作成し、仮想 IP の依存関係を作成します（先のステップで実施済み）。
3. 「仮想 IP」に基づく「仮想ホスト名」のノード 1 に SAP をインストールします。
4. ノード 1 で Stopsap を実行します。
5. LifeKeeper GUI を使用して仮想 IP をノード 2 で「In-Service」にし、「仮想 IP」に基づく「仮想ホスト名」のノード 2 に SAP をインストールします。
6. ノード 2 で Stopsap を実行し、両方のノードでプロファイルファイルを変更します（下記参照）。
7. Use the LifeKeeper GUI を使用して仮想 IP をノード 1 で「In-Service」に戻します。
8. SAP コンサルタントのアドバイスに従って、SAP に必要なマウントポイントのレプリケーションリソースを作成します。
9. ノード 1 で Startsap を実行し、SAP が正常に稼働していることを確認します。
10. 両方のノードの /etc/default/LifeKeeper で、次のエントリを最後に追加します。

```
SAP_EXPERTMODE=1
SAP_NFS_CHECK_IGNORE=1
SAP_DB_CHECK_IGNORE=1
```
11. SPS セットアッププログラムを再実行して、SAP Recovery Kit を追加します。

次のコマンドを使用して sps.img ファイルをマウントします（前述の手順に従ってダウンロードしてください）：

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

PATH : イメージへのパス

IMAGE_NAME : イメージ名

MOUNT_POINT : マウントする場所へのパス

マウントされた sps.img ディレクトリに移動し、次のように入力します :

```
./setup
```

Recovery Kit Selection Menu から SAP Recovery Kit をインストールしてください。SAP の Recovery Kit を選択するには、矢印キーを使用して選択して<スペースバー>を押して決定し、<Enter>を押してインストールを続行します。

12. NFS サーバーをセットアップします。

13. ファイルシステムを SAP サーバーにコピーし、冗長化とフェールオーバーのために、ファイルシステムにレプリケーションリソースを作成します。

14. NFS Recovery Kit のガイドに従って、NFS リソースを作成します。

<http://docs.us.sios.com/spslinux/9.4.1/en/topic/nfs-recovery-kit-administration-guide>

本書では簡略化した手順を説明します。

15. SAP Recovery Kit のガイドに従って、SAP リソースを作成します。

<http://docs.us.sios.com/spslinux/9.4.1/en/topic/sap-recovery-kit-administration-guide>

本書では簡略化した手順を説明します。

SAP のインストール

1. ASCS と ERS は、「仮想 IP」の「仮想ホスト名」に基づいてインストールする必要があります。これは、以前のインストール手順でホストファイルに追加されています。まだ完了していない場合は、SAP のインストール前に必ず実行してください。

- SAPinst を起動する前に SAPinst プロパティ SAPINST_USE_HOSTNAME を設定し、必要な仮想ホスト名を指定します。(注記 : SAPINST_USE_HOSTNAME の仮想 IP アドレスをメモしておいてください。LifeKeeper で SAP リソースを作成する際に後で使用します。)

```
./sapinst SAPINST_USE_HOSTNAME={hostname} を実行します。
```

- 7段階に分けて、**Core サービス**を作成して開始する必要があります。 *jdbconnect.jar* で権限エラーが発生した場合は */sapmnt/STC/exe/uc/linuxx86_64* に移動し、このディレクトリでもファイル *jdbconnect.jar* を書き込み可能にしてください (`chmod 777 ---`)。

エンキューレプリケーションは、SAP のドキュメントおよびベストプラクティスに基づいて構成し、動作を確認する必要があります。

ASCS および ERS プロファイルは、*/usr/sap*、*sapmnt* を含むローカルマウントポイント、またはご利用の環境の SAP ファイルに必要なその他のローカルマウントポイントを指す必要があります。実際のファイルは、インストールおよび構成後に NFS マウントポイントに移動されます。

ERS および ASCS インスタンスのインスタンスプロファイルでは、サービスが自動的にプログラムを再起動しないように、エントリを変更する必要があります。その後、ASCS および ERS インスタンスの「*sapstartsrv*」を再起動する必要があります。

たとえば、

```
Autostart = 1
Restart_Program_00 = local $( _ER ) pf=$( _PFL ) NR=$(SCSID)
```

を次のように変更します。

```
Autostart = 0
Start_Program_00 = local $( _ER ) pf=$( _PFL ) NR=$(SCSID)
```

2. ノード 1 で SAP の *Sapstop* を実行します。
3. LifeKeeper GUI を使用して、ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、ノード 2 を選択して IP をノード 2 に切り替えます。
4. ステップ 1 を繰り返してノード 2 に SAP をインストールし、正しく実行できることを確認します。
5. ノード 2 で SAP の *Sapstop* を実行します。
6. LifeKeeper GUI を使用して、ASCS 用に作成された IP アドレスリソースを右クリックし、「In-Service」を選択し、ノード 1 を選択して IP をノード 1 に切り替えます。
7. ノード 1 で SAP の *Sapstop* を実行し、正しく実行できることを確認します。

NFS のセットアップ

SIOS 製品をインストールする前に、両方のクラスターノードに NFS サーバーをインストールしておく必要があります。

お使いの SAP 設計での SAP 要件に基づいて NFS エクスポートを作成します。以下は参考例であり、お使いの SAP 環境を表すものではありません。

LifeKeeper は inode を使用して NFS 共有情報を管理します。したがって、すべての NFS 共有には固有の inode が必要です。すべてのファイルシステムの root ディレクトリは同じ inode を持つので、LifeKeeper で保護するためには、NFS 共有は root から少なくとも 1 つ下のディレクトリレベルでなければなりません。

たとえば、上記の情報を参照すると、`/usr/sap/trans` ディレクトリが SAP サーバー上で NFS 共有である場合、共有ストレージデバイス上に `/trans` ディレクトリが作成され、共有ストレージデバイスを `/usr/sap` としてマウントすることが求められます。

しかし、`/usr/sap` 配下にあるすべてのファイルをこの配列で必要な共有ストレージに置くことは必ずしも望ましいとは限りません。

この問題を回避するには、NFS 共有のディレクトリを含むすべての共有ファイルシステムをマウントするための `/exports` ディレクトリツリーを作成するか、SAP ディレクトリと `/exports` ディレクトリの間でソフトリンクを作成することを推奨します。または、NFS 共有ディレクトリをローカルで NFS マウントすることを推奨します。（注記：ここで `/exports` と記載しているディレクトリの名前は、ユーザーの設定によって異なる場合があります。ただし、わかりやすくするために、本書では全体を通じてこのディレクトリを `/exports` と記載します。）たとえば、SAP プライマリサーバーでのこの例のディレクトリとリンク/マウントは、次のようになります。

For the <code>/usr/sap/trans</code> share	
Directory	Notes
<code>/trans</code>	created on shared file system and shared through NFS
<code>/exports/usr/sap</code>	mounted to / (on shared file system)
<code>/usr/sap/trans</code>	soft linked to <code>/exports/usr/sap/trans</code>

`<sapmnt>`/`<SAPSID>`共有のディレクトリとリンクは次のようになります。

For the <sapmnt>/<SAPSID> share	
Directory	Notes
/<SAPSID>	created on shared file system and shared through NFS
/exports/sapmnt	mounted to / (on shared file system)
<sapmnt>/<SAPSID>	NFS mounted to <virtual SAP server>:/exports/sapmnt/<SAPSID>

ローカル NFS マウント

LifeKeeper 環境における SAP の推奨ディレクトリ構造では、1 つ以上の SAP システムディレクトリに対してローカルにマウントされた NFS 共有が必要です。

ローカルにマウントされた NFS 共有のいずれかの NFS エクスポートポイントが使用できなくなった場合、エクスポートポイントが再び利用可能になるのを待っている間にシステムがハングすることがあり、システムの再起動を含め、多くのシステム操作が正しく機能しなくなります。SAP クラスター用の NFS サーバーは LifeKeeper で保護されている必要があり、ローカルマウントポイントが存在する間は手動でサービスを停止しないでください。

誤って NFS サーバーを停止してクラスターがハングすることを防ぐためには、「NFS の考慮事項」のトピックに記載されている推奨事項に従ってください。また、「intr」マウントオプションを使用してすべての NFS 共有をマウントすると、アクセスできない NFS 共有に起因してハングしたプロセスを強制終了できます。

<INST>ディレクトリの場所

/usr/sap/<SAPSID>パスは NFS 共有ではないため、ファイルシステムのルートディレクトリにマウントできます。/usr/sap/<SAPSID>パスには、SYS サブディレクトリと、サーバー上で実行できる各 SAP インスタンスの<INST>サブディレクトリが含まれています。構成によっては<INST>ディレクトリは 1 つしかないので、共有ファイルシステムの/usr/sap/<SAPSID>の下に置いても問題ありません。

ただし他の構成では、バックアップサーバーにローカル AS インスタンスが含まれている可能性があります。このローカル AS インスタンスの<INST>ディレクトリは常時利用できるわけではなく、共有ファイルシステム上に置くべきではありません。

この問題を解決するには、ある構成については、ローカルサーバーに配置する必要のある AS の `/usr/sap/<SAPSID>`、the `/usr/sap/<SAPSID>/SYS` および `/usr/sap/<SAPSID>/<AS-INST>` の代わりに、PAS、ASCS または SCS の `/usr/sap/<SAPSID>/<INST>`、`/usr/sap/<SAPSID>/<ASCS-INST>` または `/usr/sap/<SAPSID>/<SCS-INST>` ディレクトリを共有ファイルシステムにマウントすることを推奨します。

たとえば、次のディレクトリとマウントポイントを ABAP + Java 構成用に作成する必要があります。

Directory	Notes
<code>/usr/sap/<SAPSID>/DVEBMGS<No.></code>	mounted to / (Replicated non-NFS file system)
<code>/usr/sap/<SAPSID>/SCS<No.></code>	mounted to / (Replicated non-NFS file system)
<code>/usr/sap/<SAPSID>/ERS<No.></code> (for SCS instance)	should be locally mounted on all cluster nodes or mounted from a NAS share (should not be mounted on shared storage)
<code>/usr/sap/<SAPSID>/ASCS<Instance No.></code>	mounted to / (Replicated non-NFS file system)
<code>/usr/sap/<SAPSID>/ERS<No.></code> (for ASCS instance)	should be locally mounted on all cluster nodes or mounted from a NAS share (should not be mounted on shared storage)
<code>/usr/sap/<SAPSID>/AS<Instance No.></code>	created for AS on backup server

NFS のマウントとファイルシステムの移動

メイン SAP ファイルシステムのマウントポイントが作成されたら、必要に応じてマウントします。この時点で、これらの手順に進む前にすべての SAP サービスを停止します。

```
mount /dev/sap/sapmnt /exports/sapmnt
mount /dev/sap/saptrans /exports/saptrans
```

データの NFS への移動

1. `/etc/exports` を編集し、SAP のメインディレクトリのマウントポイントを挿入します。

```
/exports/sapmnt *(rw, sync, no_root_squash)
/exports/saptrans *(rw, sync, no_root_squash)
```

NFS エクスポートの例

各<nfsvip>を、その NFS 共有で使用される適切な仮想 IP に置き換えます。SAP システムの設計に応じて、異なる仮想 IP を使用して異なるファイルシステムを共有できます。

```
# more /etc/exports
/exports/sapmnt 10.2.0.69(rw,sync,all_squash,anonuid=0,anongid=1001)
/exports/sapmnt 10.2.0.11(rw,sync,all_squash,anonuid=0,anongid=1001)
/exports/usr/sap/<instance name>/ASCS01
10.2.0.69(rw,sync,all_squash,anonuid=0,anongid=1001)
/exports/sap/<instance name>/ASCS01
10.2.0.11(rw,sync,all_squash,anonuid=0,anongid=1001)

# more /etc/fstab
#
# /etc/fstab
# Created by anaconda on Mon Nov 9 20:20:10 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info #
UUID=367df610-4210-4a5a-8c8d-51ddf499fc17 / xfs defaults 0 0
/dev/xvdb swap swap defaults 0 0
/dev/xvdc /tmp xfs nodev,nosuid,noexec,relatime 0 0
/dev/xvdp1 /var xfs defaults 0 0
/dev/xvdp2 /var/log xfs defaults 0 0
/dev/xvdp3 /var/log/audit xfs defaults 0 0
/dev/xvdp4 /home xfs defaults,nodev 0 0
/tmp /var/tmp none bind,nodev,nosuid 0 0
/dev/xvdj /usr/sap xfs defaults 0 0
/dev/xvdg /exports/usr/sap/P4G/ASCS01 xfs defaults 0 0
/dev/xvdh /usr/sap/P4G/D00 xfs defaults 0 0
/dev/xvdi /sapcd xfs defaults 0 0
/dev/xvdk /exports/sapmnt xfs defaults 0 0
<nfsvip>:/exports/usr/sap/P4G/ASCS01 /usr/sap/<instance name>/ASCS01 nfs
nfsvers=3,proto=udp,rw,sync,intr,bg 0 0
<nfsvip>:/exports/sapmnt /sapmnt nfs nfsvers=3,proto=udp,rw,sync,intr,bg 0 0
<nfsvip>:/exports/usr/sap/P4G/ASCS01 /usr/sap/PG4/ASCS01 nfs nfsvers=3,proto=udp,rw,sync,intr,bg 0
0 (Note: This ERS entry will only be present if using an ERSv2 configuration with a shared ERS
filesystem.)
```

2. `rcnfsserver start` コマンドを使用して NFS サーバーを起動します (SLES の場合。Red Hat の場合は `service nfs start` を実行します)。NFS サーバーがすでにアクティブな場合は、"`exportfs -va`" を実行してこれらのマウントポイントをエクスポートします。

3. ノード 1 および 2 の両方で、次のマウントコマンドを実行します (**udp の使用方法に注意してください。これはフェールオーバーとリカバリーで重要です**)。これにより、NFS 共有をマウントできるようになります。

```
mount {virtual ip}:/exports/sapmnt/<PG4> /sapmnt/<PG4> -o rw,sync,bg,intr,udp
mount {virtual ip}:/exports/saptrans /usr/sap/trans -o rw,sync,bg,intr,udp
```

4. ノード 1 から、必要なファイルシステムを、/usr/sap および/sapmnt またはその他の必要なファイルから、NFS サーバーからノード 1 にマウントされた NFS マウントポイントにコピーします。
5. SAP にログインし、SAP を起動します (su の後に stcadm を実行します)。
startsap sap{No.}

6. すべてのプロセスが開始されていることを確認してください。

```
ps -ef | grep en.sap (2 processes)
ps -ef | grep ms.sap (2 processes)
ps -ef | grep dw.sap (17 processes)
```

「SAP Logon」または「SAP GUI for Windows」は、SAP が提供する Windows クライアントです。プログラムは SAP のダウンロードサイトからダウンロードできます。仮想 IP アドレスは **[Properties]** ページで「アプリケーションサーバー」として使用できます。これにより、仮想 IP が存在するプライマリマシンへの接続がアクティブになります。

7. まだ実行していない場合は、NFS 共有マウントポイントにデータレプリケーションクラスターリソースを作成して、ノード 1 からノード 2 にデータを複製します。

NFS リソース階層の作成

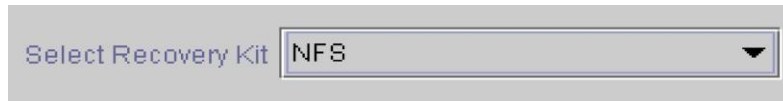
複数の NFS 共有の複数のリソース階層を作成するには、以下の手順を繰り返す必要があります。

プライマリサーバーからリソースインスタンスを作成するには、次の手順を実行してください。

1. LifeKeeper GUI メニューから **[Edit]** → **[Server]** を選択し、ドロップダウンメニューから **[Create Resource Hierarchy]** を選択します。

すでに選択されている内容を変更する場合や、NFS リソース階層の作成中にエラーメッセージが表示された場合は、**[Back]** ボタンを使用して選択内容を変更するか、または修正します (**[Back]** ボタンが有効な場合)。

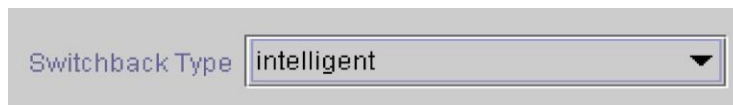
ダイアログボックスに、クラスター内にインストールされ認識されているすべての Recovery Kit の一覧がドロップダウンメニューで表示されます。ドロップダウンメニューから **NFS** を選択します。



[Next] をクリックして次のダイアログボックスに進みます。

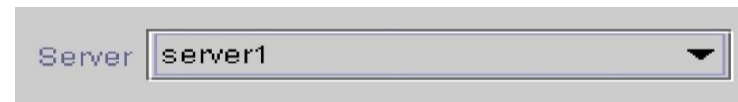
注記： 階層の作成中に [Cancel] ボタンをクリックすると、LifeKeeper は作成プロセス全体をキャンセルします。

2. **スイッチバックタイプ**を選択します。これは、バックアップサーバーへのフェールオーバー後にサービスに戻るときに、NFS インスタンスがプライマリサーバーに戻される方法を指示します。**Intelligent** または **Automatic** のいずれかを選択してください。**Intelligent スイッチバック**では、インスタンスをプライマリ/元のサーバーに戻す際に管理者の介入が必要です。**Automatic スイッチバック**では、プライマリサーバーがオンラインに戻って LifeKeeper 保護下に入るとすぐにスイッチバックが行われます。



スイッチバックタイプは、必要に応じて、[Resource Properties] ダイアログボックスの [General] タブから後で変更できます。

3. NFS リソースを作成する**サーバー**を選択します（通常、これはプライマリまたはテンプレートサーバーと呼ばれます）。クラスター内のすべてのサーバーがドロップダウンメニューに含まれています。



[Next] をクリックして次のダイアログボックスに進みます。

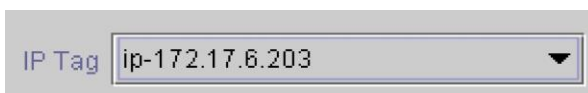
4. **Export Point** ダイアログには、次の基準を満たしている NFS ファイルシステムのエクスポートポイントのドロップダウンリストが表示されます。
- エクスポートポイントは NFS によってエクスポートされている。
 - エクスポートポイントは共有ドライブ上にある。
 - 基礎となるファイルシステムが LifeKeeper で保護されている場合、そのファイルシステムは In-Service 状態で、**サーバー**ダイアログで選択したサーバーで優先度の最も高いファイルシステムである必要があります。
 - NFSv4 の基準：
 - バインドマウントによる NFS v4 ルートエクスポートの場合、バインドマウントはエクスポートと同様に共有ドライブ上になければならず、ファイルシステムが LifeKeeper で保護されている場合は、In-Service 状態で、サーバーダイアログで選択したサーバー上で最も高い優先度を持つ必要があります。
 - NFS v4 のルートエクスポートがすでに保護されている場合は、選択肢はありません (v4 が 1 つのみで存在する必要があります。v4 と v2/v3 が混在すると保護できません)。
 - NFS v2/v3 がすでに保護されている場合は、選択肢に NFS v4 はリストされません。
 - 何も保護されていない場合、リストには v2/v3 と v4 の両方が含まれます。

ドロップダウンリストから、保護する NFS エクスポートポイントを選択します。



[Next] をクリックして次のダイアログボックスに進みます。

5. **[IP Tag]** ダイアログには、LifeKeeper 保護下にあり、NFS リソースが作成されているサーバー上で In-Service 状態の仮想 IP アドレスに対応するタグのドロップダウンリストが表示されます。クライアントが保護された NFS ファイルシステムにアクセスするために使用する仮想 IP アドレスの**タグ**を選択します。



注記：この時点で、LifeKeeper は利用可能な保護された IP リソースがあることを確認します。また、NFS 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ているかどうか検証します。LifeKeeper がこれらの検証のいずれかで問題を検出すると、ERROR ボックスが画面に表示されます。ディレクトリパスは有効であるのに NFS の設定自体にエラーがある場合、一時停止してこれらのエラーを修正し、階層作成を続行することができます。必要な LifeKeeper の IP リソースを作成するために一時停止することもできます。

注記：仮想 IP アドレスの依存関係を持つ他の LifeKeeper Recovery Kit を使用している場合は、NFS リソース用に別の仮想 IP アドレスを作成する必要があります。そうしないと、仮想 IP リソースがバックアップサーバーにフェールオーバーした場合、その IP リソースに依存するすべてのリソースが同時にフェールオーバーされてしまいます。

[Next] をクリックして次のダイアログボックスに進みます。

6. **NFS タグ**を選択または入力します。これは NFS 階層に与えられたタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。

NFS Tag

[Create] ボタンをクリックすると、**リソース作成ウィザード**が NFS リソースを作成します。

```
Creating gen/nfs resource...
Creating Resource Instance "filesys10644" with id "/export/public" on machine "server1"
Resource "filesys10644" Successfully Created on machine "server1"
Creating Dependency "filesys10644"-device10618" on machine "server1":
Dependency "filesys10644"-device10618" Successfully Created on machine "server1"
Removing /etc/fstab entry
12/11/2000 11:58:02 create: END successful creation of "hanfs-/export/public" on server "server1"
12/11/2000 11:58:03 restore: BEGIN restore of "hanfs-/export/public" on server "server1"
12/11/2000 11:58:03 restore: END successful restore of "hanfs-/export/public" on server "server1"
12/11/2000 11:58:05 create: END successful creation of "nfs-/export/public" on server "server1"
12/11/2000 11:58:07 restore: BEGIN restore of "nfs-/export/public" on server "server1"
12/11/2000 11:58:07 restore: END successful restore of "nfs-/export/public" on server "server1"
```

[Next] をクリックすると、LifeKeeper は[階層の拡張](#)で説明している **Pre-Extend Wizard** を起動します。

注記：この時点で NFS リソース階層が正常に作成されている必要があります。ただし、新しい NFS インスタンスが正しく起動できなかったことを示すエラーメッセージが表示されることがあります。新しい NFS 階層は、別のシステムに拡張する前に起動する（In-Service 状態にする）必要があります。

起動に失敗すると階層が削除されることがありますが、そうでない場合は、この時点で一時停止し、表示されるエラーメッセージに基づいて問題を修正することができます。エラーを修正できない場合はキャンセル以外に選択肢はなく、リソース作成がキャンセルされます。

[階層の拡張](#)に進む前に、新しい階層を In-Service にしてください。

***各 NFS 共有の追加のリソース階層を作成するには、上記の手順を繰り返してしてください。

注記：RHEL 7.1 以降および SLES12 SP1 以降で NFS リソースを作成した後、nfs-server.service の自動起動を無効にしてください。NFS リソースの起動時に rpcbind.service を実行する必要があるため、rpcbind.service を自動的に起動するように設定します。

SAP リソース階層の作成

1. LifeKeeper GUI メニューから、**[Edit]** → **[Server]** を選択します。ドロップダウンメニューから、**[Create Resource Hierarchy]** を選択します。

クラスター内にインストールされ認識されているすべての Recovery Kit 一覧のドロップダウンリストのあるダイアログボックスが表示されます。ドロップダウンリストから **SAP** を選択します。



Please Select Recovery Kit

[Next] をクリックします。

いずれかのダイアログボックスで **[Back]** ボタンが有効になっている場合は、前のダイアログボックスに戻ることができます。これは、以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特に役立ちます。

階層の作成中に **[Cancel]** をクリックすると、LifeKeeper は作成プロセス全体をキャンセルします。

2. **スイッチバックタイプ**を選択します。これは、バックアップサーバーへのフェールオーバー後にサービスに復帰したときに SAP インスタンスをこのサーバーに戻す方法を指示します。**Intelligent** または **Automatic** のいずれかを選択できます。**Intelligent スイッチバック**では、インスタンスをプライマリサーバー/元のサーバーに戻すために管理者の介入が必要です。**Automatic スイッチバック**では、プライマリサーバーがオンラインに戻り、LifeKeeper のコミュニケーションパスが再確立されるとすぐにスイッチバックが行われます。



Switchback Type

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。

[Next] をクリックします。

3. SAP PAS、ASCS または SCS を配置するサーバーを選択します（通常、これをプライマリまたはテンプレートサーバーと呼びます）。クラスター内のすべてのサーバーがドロップダウンリストボックスに含まれています。



Server

4. **SAP SID** を選択します。これは、保護対象の SAP PAS、ASCS、または SCS システムのシステム識別子です。



SAP SID

[Next] をクリックします。

5. 保護対象の SID の SAP インスタンス名 (例 : ASCS< No.>) (コアインスタンスが先頭) を選択します。

SAP Instance for PRS

[Next] をクリックします。

注記 : 保護およびリカバリーレベルのカスタマイズに関連して、追加の画面が表示されることがあります。

6. **IP の子リソース**を選択します。これは通常、SAP インストール時に指定された仮想ホスト IP アドレス (SAPINST_USE_HOSTNAME) またはフェールオーバーに必要な IP アドレスのいずれかです。

IP child resource

ip-jamie
ip-mutt-104.57
ip-jeff-104.58
none

7. **SAP Tag** を選択または入力します。これは LifeKeeper が SAP 階層に付与するタグ名です。デフォルトを選択するか、独自のタグ名を入力することができます。デフォルトのタグは `SAP-<SID>_<ID>` です。

SAP Tag

[Create] をクリックすると、**SAP リソース作成ウィザード**が SAP リソースを作成します。

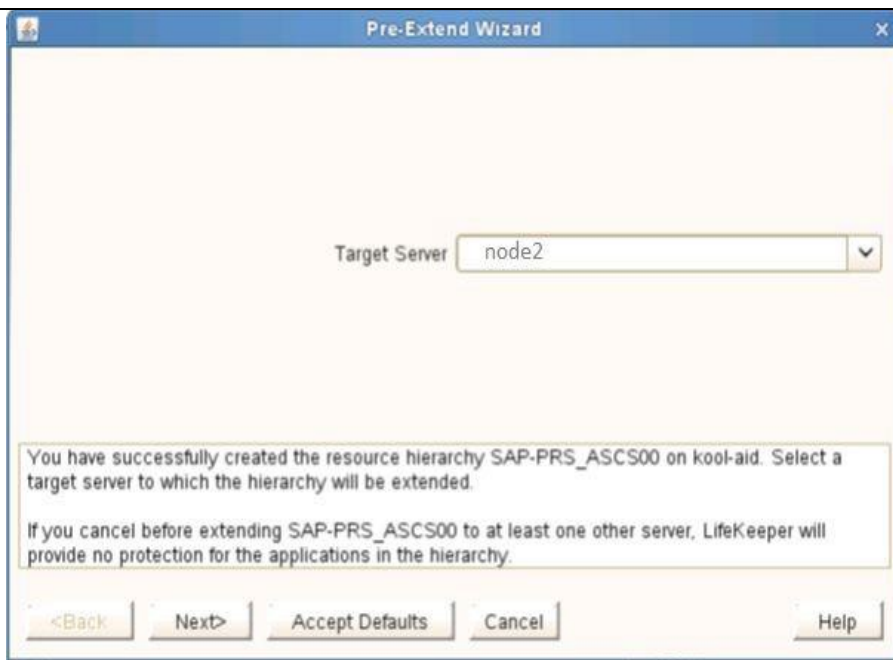
- ここで情報ボックスが表示され、LifeKeeper は利用可能な保護された IP リソースがあることを検証します。また、SAP リソース階層を作成するために有効なデータが提供されているかどうか検証します。LifeKeeper が問題を検出すると、ERROR が情報ボックスに表示されます。検証が成功すると、リソースが作成されます。情報ボックスに表示される SAP 起動スクリプトからエラーまたはメッセージが出力される場合もあります。

```
Creating appsuite/sap resource...
Additional information is available in the LifeKeeper and system logs
Wed Aug 24 14:23:01 EDT 2011 restore: 112025: All processes for SAP SID "PRS" and
Instance "ASCS00" are "running" on "kool-aid".
Additional information is available in the LifeKeeper and system logs
Wed Aug 24 14:23:01 EDT 2011 restore: 112003: The SAP Instance "ASCS00" and all required
processes were started successfully during the "restore" on server "kool-aid".
Additional information is available in the LifeKeeper and system logs.
Wed Aug 24 14:23:01 EDT 2011 restore: END successful restore of "SAP-PRS_ASCS00" on
server "kool-aid"
Wed Aug 24 14:23:01 EDT 2011 create: END successful create of "SAP-PRS_ASCS00" on
server "kool-aid"
```

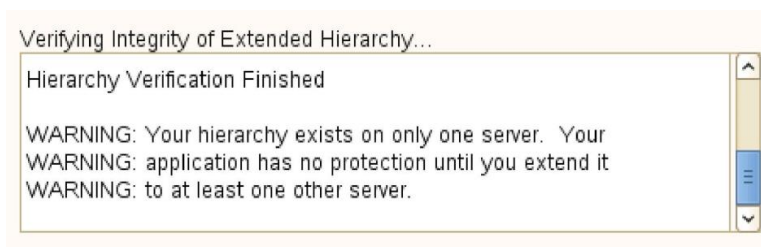
[Next] をクリックします。

- SAP リソース階層が正常に作成されたことを示す別の情報ボックスが表示されます。その階層を LifeKeeper 保護下に置くためには、クラスター内の別のサーバーに**拡張**する必要があります。

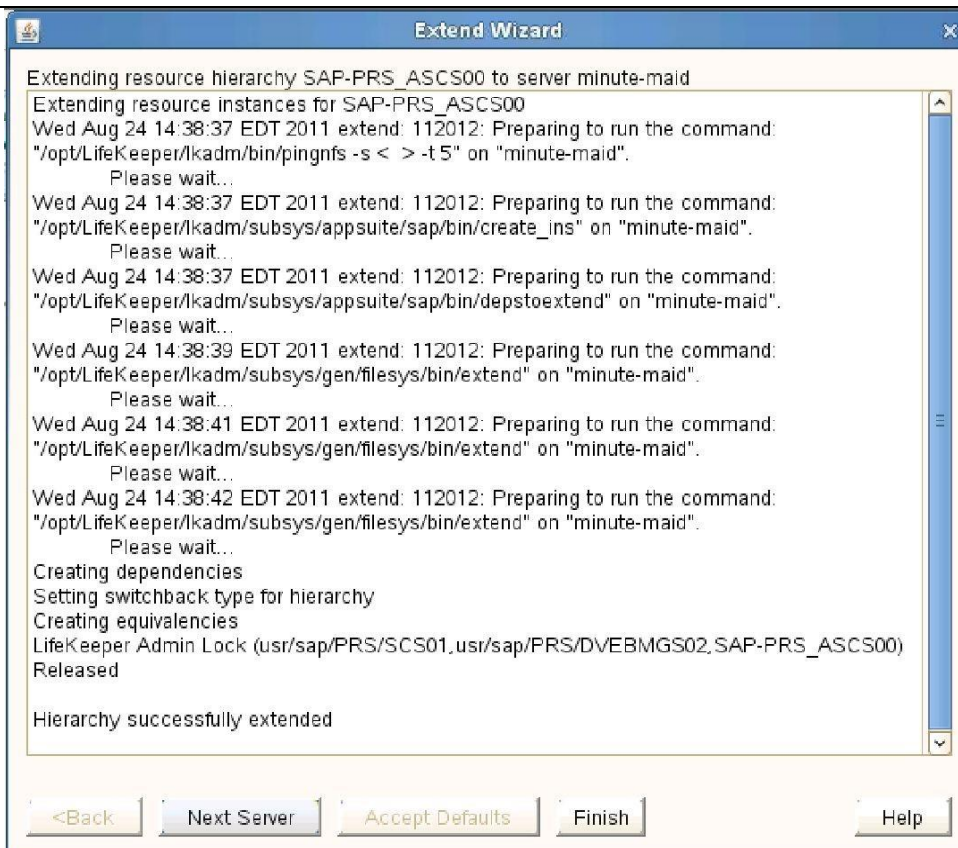
[Next] をクリックすると、LifeKeeper は **[Pre-Extend Wizard]** を起動します。このウィザードについては、本セクションで後述します。



ここで **[Cancel]** をクリックすると別のダイアログボックスが表示され、いずれかの時点でここに戻り、SAP リソース階層を別のサーバーに拡張して LifeKeeper の保護下に置く必要があることが警告されます。



10. 階層が正常に拡張されたことを示す **Extend Wizard** ダイアログが表示されます。 **[Finish]** をクリックしてください。



[Hierarchy Integrity Verification] ダイアログが表示されます。階層の検証が終了したら、**[Done]** をクリックして、**[Create Resource Hierarchy]** メニューを終了します。

コアが最上位レベルにある階層



ERS リソースの作成

ERS リソースは、コアインスタンス（セントラルサービスインスタンス）またはエンキューサーバプロセスの単一障害点に対する追加の保護を提供します。コアインスタンス（セントラルサービスインスタンス）で障害が発生して再起動されると、ERS リソースはロックテーブルとトランザクションの現在のステータスを取得します。その結果、エンキューサーバに障害が発生した場合でもトランザクションや更新は失われず、SAP システムのサービスは継続されます。

この ERS リソースを作成するには、以下の手順を実行してください。

1. この同じ SAP SID について上記の手順を繰り返し、**[ERS instance]** を選択して、ERS リソースを作成します。
2. その後、**[Dependent Instances]** を選択するよう要求されます。上記で作成した**コアリソース**を選択し、**[Next]** をクリックします。
3. 画面の指示に従って**リソース階層を拡張**してください。
4. 「**Hierarchy Successfully Extended**」と表示されたら、**[Finish]** をクリックします。
5. **[Done]** を選択してください。

注記：エンキューレプリケーションサーバ (ERS) はクラスターの各ノードで In-Service (ISP) になります。ただし、ERS のアーキテクチャと機能では、インスタンスの実際のプロセスはバックアップノードで実行する必要があります。これにより、スタンバイサーバは、プライマリサーバおよびプライマリエンキューサーバインスタンスのロックテーブル情報の完全なコピーを保持できるようになります。エンキューサーバを実行しているプライマリサーバで障害が発生すると、ERS プロセスが現在実行されているバックアップサーバ上の SIOS Protection Suite によって再起動されます。ERS に格納されているロックテーブル（レプリケーションテーブル）は復旧中のエンキューサーバプロセスに転送され、そこから新しいロックテーブルが作成されます。このプロセスが完了すると、アクティブなレプリケーションサーバは非アクティブになります（エンキューサーバへの接続が切断され、レプリケーションテーブルが削除されます）。SIOS Protection Suite は、新しいバックアップノード（以前はプライマリノード）上の、今まで非アクティブだった ERS プロセスを再起動します。ERS プロセスがアクティブになると、エンキューサーバに接続してレプリケーションテーブルを作成します。ERS プロセスがアクティブになると、エンキューサーバに接続してレプリケーションテーブルを作成します。

ERS プロセスおよび SAP アーキテクチャの機能の詳細については、<http://help.sap.com> にアクセスし、**Enqueue Replication Service** を検索してください。

ERS が最上位レベルにある階層



SIOS Protection Suite を使用して PAS および AAS サーバーを保護することはできますが、ほとんどのお客様は、それらを追加の HA なしで独立したスタンバイサーバーとして使用しているだけです。このガイドでは保護手順については説明しないので、詳細と手順については SAP Recovery Kit を参照してください。

<http://docs.us.sios.com/spslinux/9.4.1/en/topic/sap-recovery-kit-administration-guide>

ENSAv2/ERSv2 使用時に ASCS/ERS に回避動作を強制する

ERSv2 は、ASCS リソースがアクティブでないクラスター内のノードでアクティブ (In-service) になります。ERS quickCheck は、ERS と ASCS が同じノードでアクティブであり、別のノードが利用可能な場合、ERS 階層を自動的に転送します。スイッチオーバーまたはフェイルオーバー後に同じノードで ASCS と ERS の両方がアクティブ (In-service) になる状況を回避するために、gen/app ターミナルリーフリソースを作成して、対応するリソース階層がアクティブ (In-service) ではないノードに In-service 状態を自動的にルーティングできます。

このターミナルリーフノードの作成を容易にするために、新しいユーティリティ

/opt/LifeKeeper/bin/create_terminal_leaf (1M) が提供されています。回避ターミナルリーフを作成するために、ユーティリティは 2 つのパラメーター、ASCS および ERS 階層ルートリソースタグを使用します。2 つの階層は、クラスター内のすべてのノードに完全に拡張し、クラスター内のノードで In-service

にする必要があります。ユーティリティがクラスター内のノードで実行されている限り、ユーティリティが実行されているノードで階層を In-service にする必要はありません。tag が回避する適切なルートノードである場合、ターミナルリーフノードには「avoid_<tag>」という名前が付けられます。ターミナルリーフノードは、階層の各ブランチの子依存関係としてアタッチされます。

たとえば、SAP-EXM_ASCS02 および SAP-EXM_ERS12 をルートノードとして使用する構成では、次のようになります。

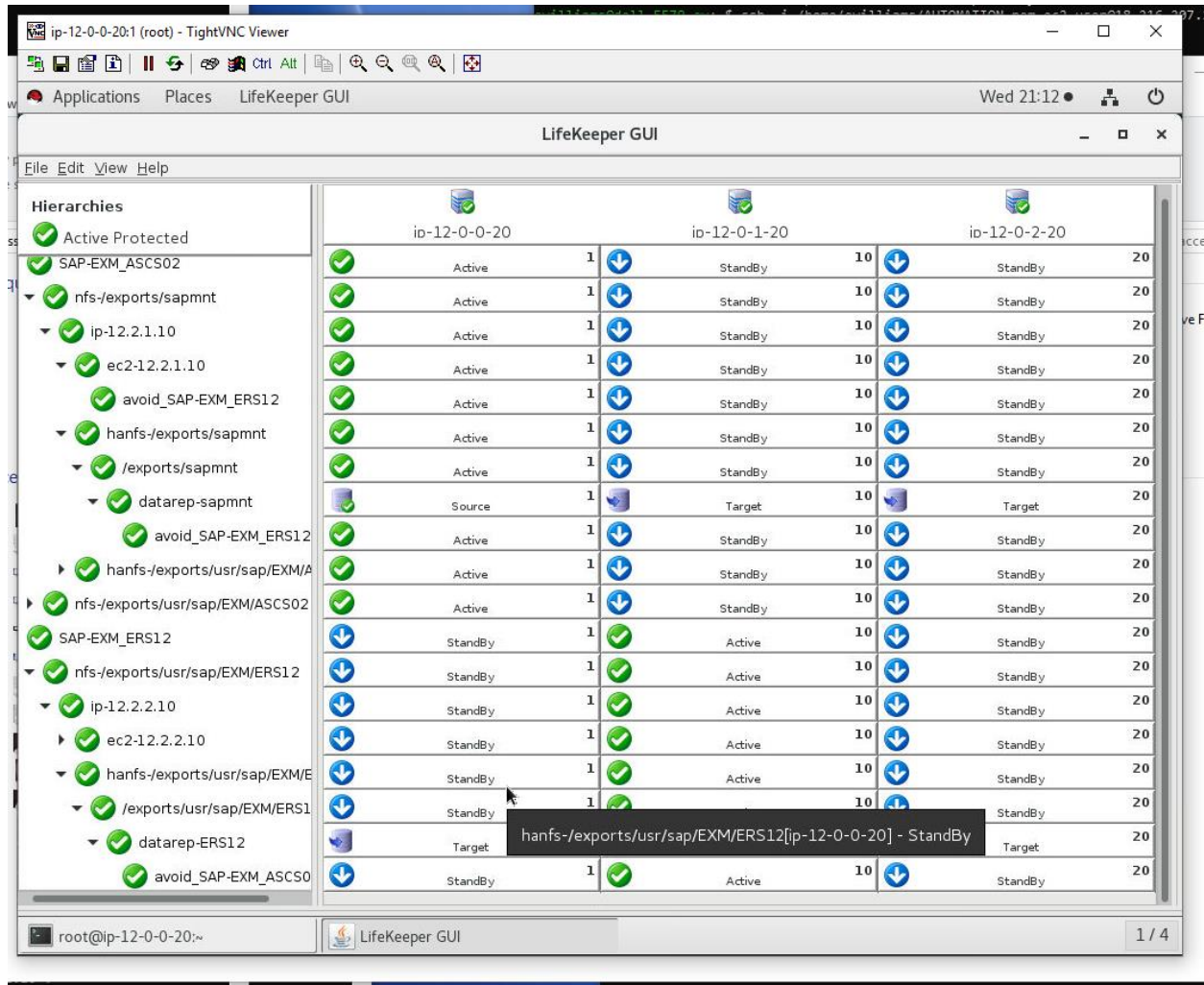
The screenshot shows the LifeKeeper GUI interface. On the left, a 'Hierarchies' tree is visible, showing the configuration for SAP-EXM_ASCS02 and SAP-EXM_ERS12. The main area displays a table of nodes and their status for each hierarchy.

Hierarchy	Node	Status	Weight	Priority	Role	Weight	Priority	Role
SAP-EXM_ASCS02	ip-12-0-0-20	Active	1	10	StandBy	10	10	StandBy
	ip-12-0-1-20	StandBy	1	10	StandBy	10	10	StandBy
	ip-12-0-2-20	StandBy	1	10	StandBy	10	10	StandBy
	nfs-/exports/sapmnt	Active	1	10	StandBy	10	10	StandBy
	ip-12.2.1.10	Active	1	10	StandBy	10	10	StandBy
	ec2-12.2.1.10	Active	1	10	StandBy	10	10	StandBy
	hanfs-/exports/sapmnt	Active	1	10	StandBy	10	10	StandBy
	/exports/sapmnt	Active	1	10	StandBy	10	10	StandBy
	datarep-sapmnt	Source	1	10	Target	10	10	Target
	hanfs-/exports/usr/sap/EXM	Active	1	10	StandBy	10	10	StandBy
	nfs-/exports/usr/sap/EXM/ASCSC	Active	1	10	StandBy	10	10	StandBy
	SAP-EXM_ERS12	StandBy	1	10	Active	10	10	StandBy
nfs-/exports/usr/sap/EXM/ERS12	StandBy	1	10	Active	10	10	StandBy	
ip-12.2.2.10	StandBy	1	10	Active	10	10	StandBy	
ec2-12.2.2.10	StandBy	1	10	Active	10	10	StandBy	
hanfs-/exports/usr/sap/EXM	StandBy	1	10	Active	10	10	StandBy	
/exports/usr/sap/EXM/ERS	StandBy	1	10	Active	10	10	StandBy	
datarep-ERS12	Target	1	10	Source	10	10	Target	

/opt/LifeKeeper/bin/create_terminal_leaf SAP-EXM_ASCS02 SAP-EXM_ERS12 を実行することにより、適切なターミナルリーフノードが作成されます。

```
[root@ip-12-0-2-20 ~]# /opt/LifeKeeper/bin/create_terminal_leaf SAP-EXM_ASCS02 SAP-EXM_ERS12
Create avoidance terminal leaf resource for root hierarchy 'SAP-EXM_ASCS02' and 'SAP-EXM_ERS12'.
creapphier: WARNING No quickCheck script specified
creapphier: WARNING No local recovery script specified
BEGIN create of "avoid_SAP-EXM_ERS12"
creating resource "avoid_SAP-EXM_ERS12"
resource "avoid_SAP-EXM_ERS12" successfully created
restoring resource "avoid_SAP-EXM_ERS12"
BEGIN restore of "avoid_SAP-EXM_ERS12"
Attempting to avoid resource avoid_SAP-EXM_ASCS02. Since the resource was not found on ip-12-0-0-20, restore
of avoid_SAP-EXM_ERS12 is successful on ip-12-0-0-20.
END successful restore of "avoid_SAP-EXM_ERS12"
resource "avoid_SAP-EXM_ERS12" restored
END successful create of "avoid_SAP-EXM_ERS12"
Extending resource instances for avoid_SAP-EXM_ERS12
Creating dependencies
Setting switchback type for hierarchy
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ERS12) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ERS12"
END successful extend of "avoid_SAP-EXM_ERS12"
Extending resource instances for avoid_SAP-EXM_ERS12
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ERS12) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ERS12"
END successful extend of "avoid_SAP-EXM_ERS12"
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'ec2-12.2.1.10'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCs02'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCs02'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-ASCs02'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ERS12' to 'datarep-sapmnt'.
creapphier: WARNING No quickCheck script specified
creapphier: WARNING No local recovery script specified
BEGIN create of "avoid_SAP-EXM_ASCS02"
creating resource "avoid_SAP-EXM_ASCS02"
resource "avoid_SAP-EXM_ASCS02" successfully created
restoring resource "avoid_SAP-EXM_ASCS02"
BEGIN restore of "avoid_SAP-EXM_ASCS02"
Attempting to avoid resource avoid_SAP-EXM_ERS12. Since the resource is not ISP on ip-12-0-1-20, restore
of avoid_SAP-EXM_ASCS02 is successful on ip-12-0-1-20.END successful restore of "avoid_SAP-EXM_ASCS02"
resource "avoid_SAP-EXM_ASCS02" restored
END successful create of "avoid_SAP-EXM_ASCS02"
Extending resource instances for avoid_SAP-EXM_ASCS02
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ASCS02) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ASCS02"
END successful extend of "avoid_SAP-EXM_ASCS02"
Extending resource instances for avoid_SAP-EXM_ASCS02
Creating dependencies
Creating equivalencies
LifeKeeper Admin Lock (avoid_SAP-EXM_ASCS02) Released
Hierarchy successfully extended
BEGIN extend of "avoid_SAP-EXM_ASCS02"
END successful extend of "avoid_SAP-EXM_ASCS02"
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ASCS02' to 'ec2-12.2.2.10'.
Create dependency on 'ip-12-0-0-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
Create dependency on 'ip-12-0-1-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
Create dependency on 'ip-12-0-2-20' for 'avoid_SAP-EXM_ASCS02' to 'datarep-ERS12'.
[root@ip-12-0-2-20 ~]#
```

ターミナルリーフノードがアタッチされました。



avoid_SAP-EXM_ASCS02 リソース（SAP-EXM_ERS12 階層内）がノードで In-service であり、クラスター内に利用可能な別のノードがある場合、avoid_SAP-EXM_ERS12 リソースは SAP-EXM_ASCS02 階層がそのノード上で In-service になることを許可しません。

次の場合、ノードは利用できません。

1. ノードが応答していない場合。
2. LifeKeeper がそのノードで実行されていない場合。

3. ノードでローカルリカバリーに失敗した場合。失敗したかどうかは、`/opt/LifeKeeper/bin/flg_list` の出力から `!volatile!recover_fail_<tag>` フラグを確認することにより判断できます。

フラグを作成することにより、特定のシステムで回避リーフを無効にできます。

1. “`ignore_avoidance_leaf`” - リソースの回避リーフチェックを無効にします。つまり、回避リーフは常に In-service になります。
2. “`ignore_<tag>`” - 特定の `<tag>` を無効にして常に In-service になるようにしますが、他の回避リーフは In-service を回避します。

注記：回避リーフを無効にするための上記のフラグは、SAP quickCheck が ASCS で実行されていることを検出した場合、SAP quickCheck が ERS を移行することには影響しません。フラグ “`sap_no_ers_relocation_<tag>`” は、`<tag>` が ERS リソースタグの場合、quickCheck が ERS を再配置できないようにします。

GUI を使用して回避ターミナルリーフノードを作成する

回避ターミナルリーフノードは、GUI を使用して作成できる `gen/app` リソースです。回避ターミナルリーフの復元スクリプトは `/opt/LifeKeeper/lkadm/bin/avoid_restore` で、削除スクリプトは `/bin/true` です。quickCheck スクリプトはありません。情報フィールドは、回避するタグの名前にする必要があります。たとえば、`app1` と `app2` の2つのリソースがあり、可能であれば異なるノードに配置したい場合、「`avoid_app1`」と「`avoid_app2`」の2つの `gen/app` リソースを作成できます。`avoid_app1` の「情報」フィールドには「`avoid_app2`」が、`avoid_app2` の「情報」フィールドには「`avoid_app1`」が入ります。

「`avoid_app2`」は「`app1`」の従属子リソースに、「`avoid_app1`」は「`app2`」の子リソースになります。

注記：タグ名は「`avoid_<tag>`」である必要はありませんが、これによりリソースが何をしているのかが明確になります。

The screenshot shows the LifeKeeper GUI interface. On the left, there is a sidebar with a 'Hierarchies' section containing a tree view with 'Active Protected' and two sub-items, 'app1' and 'app2'. The main area displays a table with three columns representing different hierarchies: 'id-12-0-0-20', 'id-12-0-1-20', and 'id-12-0-2-20'. Each column contains two rows of application data with status icons (green checkmarks and blue arrows) and numerical values.

Hierarchy	Application	Status	Value
id-12-0-0-20	Active	Green Checkmark	1
	StandBy	Blue Arrow	10
id-12-0-1-20	StandBy	Blue Arrow	10
	Active	Green Checkmark	1
id-12-0-2-20	StandBy	Blue Arrow	20
	StandBy	Blue Arrow	20

The screenshot shows a 'Create Resource Wizard' dialog box. The main text reads 'Please Select Recovery Kit' followed by a dropdown menu currently set to 'Generic Application'. At the bottom of the dialog, there are four buttons: '<Back', 'Next>', 'Cancel', and 'Help'. A mouse cursor is visible over the 'Next>' button.

x
Create gen/app Resource

Restore Script

Enter the pathname for the shell script or object program which starts the application. The **restore** script is responsible for bringing a protected application resource in-service. The **restore** script should not impact an active resource application when invoked.

Valid characters allowed in the script pathname are letters, digits, and the following special characters:
 - _ ! . /

A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
 Whenever this resource is extended to a new server, the copy will be passed to that server.

x
Create gen/app Resource

Remove Script

Enter the pathname for the shell script or object program which stops the application. The **remove** script is responsible for stopping a protected application resource and putting it in the out-of-service state.

Valid characters allowed in the script pathname are letters, digits, and the following special characters:
 - _ ! . /

A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
 Whenever this resource is extended to a new server, the copy will be passed to that server.

Create gen/app Resource ✕

QuickCheck Script [optional]

Enter the pathname for the shell script or object program which monitors the application. The **quickCheck** script is called periodically, and is responsible for performing a health check of the protected application.

The **quickCheck** script is optional. If one is not provided it will always be assumed that the application is in an OK state.

Valid characters allowed in the script pathname are letters, digits, and the following special characters:
- _ ! . /

A copy of this script or program will be saved under:
/opt/LifeKeeper/subsys/gen/resources/app/actions
Whenever this resource is extended to a new server, the copy will be passed to that server.

<Back Next> Cancel Help

Create gen/app Resource ✕

Application Info [optional]

Enter any optional data for the application resource instance that may be needed by the **restore** and **remove** scripts.

The valid characters allowed for the data field are letters, digits, and the following special characters:
- _ . / = [space]

<Back Next> Cancel Help

Create gen/app Resource ✕

Resource Tag

Enter a unique name for the resource instance on **ip-12-0-0-20**. The valid characters allowed for the tag are letters, digits, and the following special characters:
- _ . /

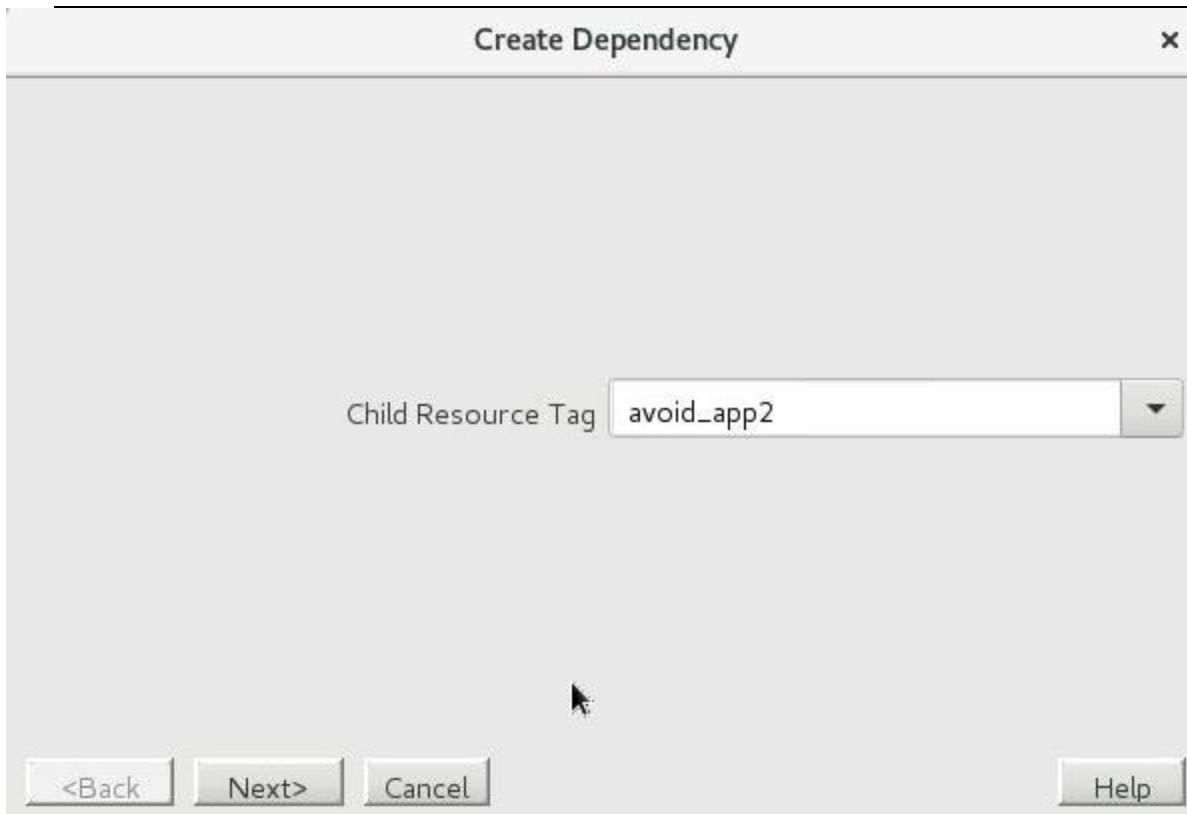
<Back Create Instance Cancel Help

「avoid_app2」を作成したら、app1 と同じ優先順位を持つすべてのノードに拡張します。

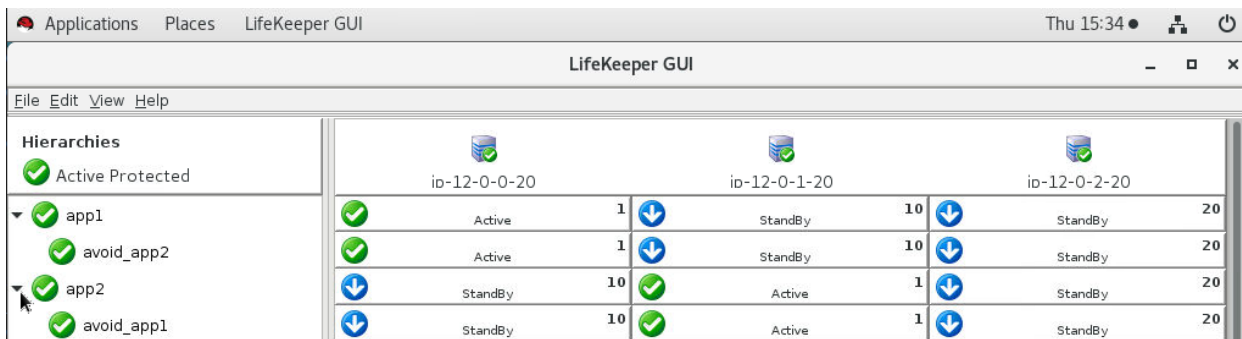
The screenshot shows the LifeKeeper GUI interface. On the left, a 'Hierarchies' sidebar lists 'Active Protected' and three resources: 'app1', 'app2', and 'avoid_app2'. The main area displays a table with three columns representing nodes: 'id-12-0-0-20', 'id-12-0-1-20', and 'id-12-0-2-20'. Each node has a status icon (green checkmark or blue arrow) and a priority value.

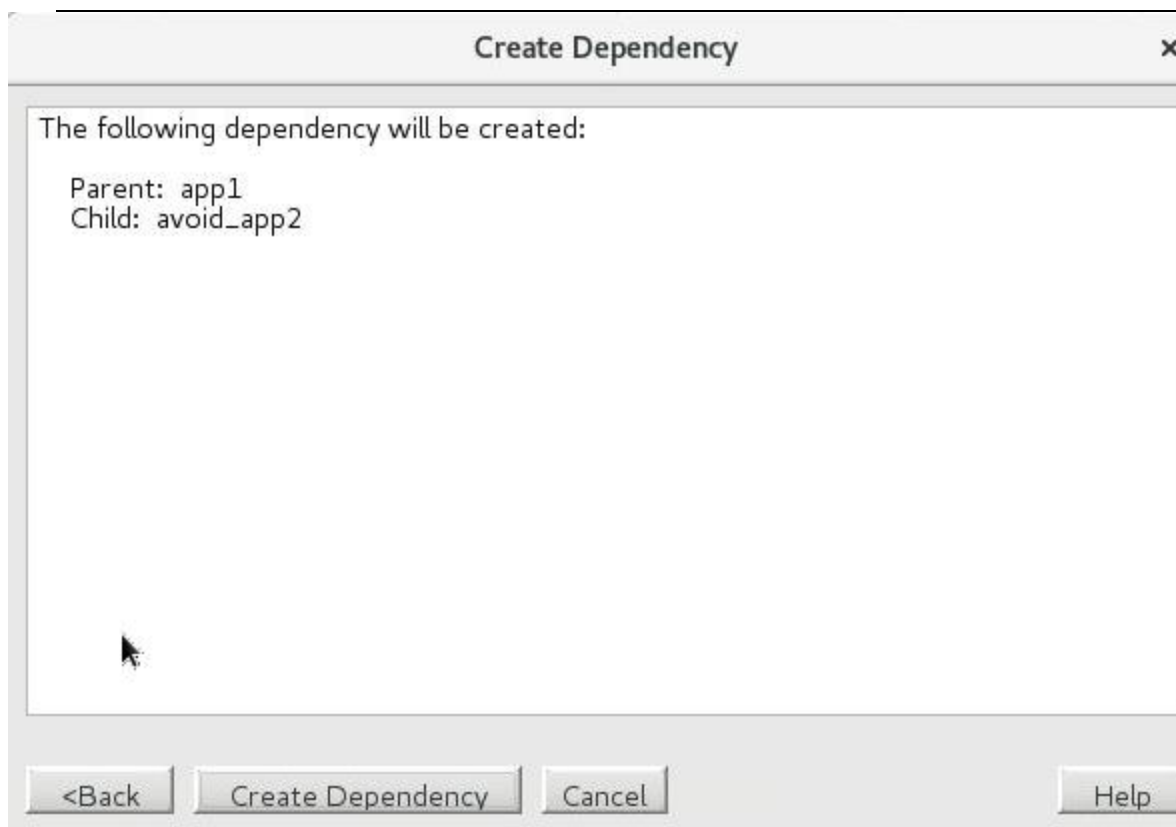
Resource	id-12-0-0-20	id-12-0-1-20	id-12-0-2-20
app1	Active 1	StandBy 10	StandBy 20
app2	StandBy 10	Active 1	StandBy 20
avoid_app2	Active 1	StandBy 10	StandBy 20

次に、「app1」リソースを選択し、「avoid_app2」との子依存関係を作成します。



avoid_app1 リソースを同様に作成すると、階層は次のようになります。

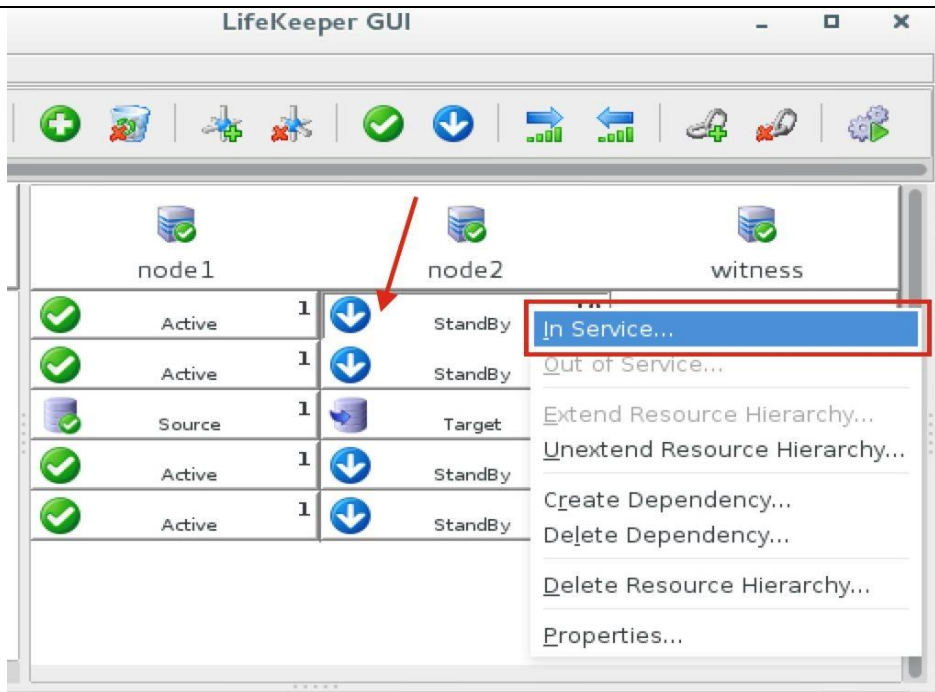




スイッチオーバーおよびフェールオーバーのテスト

次の手順は、SAP の SIOS クラスターのスイッチオーバーとフェールオーバーをテストするための手順です。SAP 提供の Windows クライアントである「SAP Logon」または「SAP GUI for Windows」を開きます。プログラムは SAP ダウンロードサイトからダウンロードできます。仮想 IP アドレスは、**[Properties]** ページでアプリケーションサーバーとして使用できます。これにより、仮想 IP が存在するプライマリマシンへの接続がアクティブになります。

1. LifeKeeper GUI を使用し、ノード 1 -> ノード 2 にフェールオーバーします。ノード 2 の下にあるクラスターの一番上のリソースを右クリックし、[In Service ...] を選択します。これは、ノード 2 が障害発生時にノード 1 から引き継ぐことができることを示しています。



スイッチオーバーが完了したら、SAP GUI をチェックするか、必要に応じて再接続し、SAP がまだ正常に動作しているかどうかを確認します。

SAP プロセスが OS で実行されているかどうかを確認することもできます。

2. LifeKeeper GUI を使用し、ノード 2 -> ノード 1 にフェールオーバーします。ノード 2 の下にあるクラスターの一番上のリソースを右クリックし、[In Service ...] を選択します。これは、ノード 1 が障害発生時にノード 2 から引き継ぐことができることを示しています。

スイッチオーバーが完了したら、SAP GUI をチェックするか、必要に応じて再接続し、SAP がまだ正常に動作しているかどうかを確認します。

SAP プロセスが OS で実行されているかどうかを確認することもできます。

3. ノード 1 (アクティブノード) のコマンドラインインターフェースで、次のコマンドを実行して OS の「ハードクラッシュ」を実行します。

```
# halt -fni
```

フェールオーバーが完了したら、ノード 2 の LifeKeeper GUI を使用して、サービスが正常にフェールオーバーされたことを目視で確認します。

SAP GUI をチェックするか、必要に応じて再接続し、SAP がまだ正常に動作しているかどうかを確認します。

SAP プロセスが OS で実行されているかどうかを確認することもできます。

ノード 1 を再びオンにし、ノード 2 の LifeKeeper GUI を使用して、ノード 1 のサービスがスタンバイになり、レプリケーションが開始されたことを目視で確認します。

注記： スイッチオーバーまたはフェールオーバーのテストをさらに行う前に、データレプリケーションリソースがすでに同期を完了し、同期中であることを確認してください。

4. 必要に応じて手順 2 または手順 1 を繰り返す、ノード 1 にスイッチオーバーするか、ノード 2 でさらにクラッシュテストを実行します。

その他のリソース

AWS サービス

- Amazon EC2
<https://aws.amazon.com/documentation/ec2/>
- AWS CloudFormation
<https://aws.amazon.com/documentation/cloudformation/>
- Amazon VPC
<https://aws.amazon.com/documentation/vpc/>

SIOS Protection Suite for Linux

- ステップバイステップ：共有ストレージを使用せずに Amazon EC2 で Linux フェールオーバークラスターを構成する方法
<http://www.linuxclustering.net/2016/03/21/step-by-step-how-to-configure-a-linux-fail-over-cluster-in-amazon-ec2-without-shared-storage-amazon-aws-sanless-cluster/>

クイックスタート

- AWS クイックスタート ホームページ
https://aws.amazon.com/jp/quickstart/?nc1=h_ls

フィードバック

ご質問やご意見がある場合は、[メール](#)でご連絡ください。

ドキュメントの更新

日付	変更
2018年11月	初版
2019年10月	<ul style="list-style-type: none">● コピーライトに関する文言を2ページ目に追記● 11ページ、20ページのコマンドを「./setup -k」から「./setup」へ変更● 古いページ内リンクを LifeKeeper for Linux v9.4 のものに修正● 「別のインスタンスで NFS を使用している ASCS + ERS」と「ASCS + ERS を NFS と一緒に ASCS と同じインスタンス上で使用する」を「NFS を使用している ASCS + ERS」に統合
2020年1月	<ul style="list-style-type: none">● 古いページ内リンクを LifeKeeper for Linux v9.4.1 のものに修正