



# Dell Storage Center と Red Hat Enterprise Linux (RHEL) 7x を併用する場合のベストプラクティス

Daniel Tan (製品スペシャリスト  
Dell ストレージアプリケーションエンジニアリング)

2015 年 5 月

## リビジョン

日付	リビジョン	説明	作成者
2014 年 7 月	1.0	初期ドラフト	Daniel Tan
2015 年 5 月	1.1	Dell Storage SCv2000 に接続性を導入	Daniel Tan

このホワイトペーパーは情報提供のみを目的とし、入力ミスや技術的に不正確な記述がある可能性があります。コンテンツは現状のまま提供され、いかなる種類の明示的または黙示的保証也没有。

©2014–2015 すべての著作権は Dell Inc. にあります。

Dell、デルのロゴ、およびデルバッジは、Dell Inc. の商標です。

本書では、商標および商標名の権利を主張する会社またはその製品を参照するため、上記に記載する以外の商標および商標名が使用される場合があります。上記記載以外の商標や会社名は、デルに帰属するものではありません。



# 目次

リビジョン .....	2
エグゼクティブサマリ .....	6
1 概要 .....	7
1.1 RHEL 7x の新機能 .....	7
2 ボリューム管理 .....	8
2.1 新規ボリュームのスキャン .....	8
2.2 パーティション .....	9
2.3 論理ボリュームマネージャ .....	10
2.3.1 LVM の設定および SCSI UNMAP/TRIM .....	10
2.4 SCSI UNMAP/TRIM およびファイルシステム .....	10
2.5 持続的なデバイス管理 .....	11
2.5.1 ファイルシステムのボリュームラベル .....	11
2.5.2 汎用一意識別子 .....	12
2.5.3 /etc/fstab および /boot/grub2/grub.cfg での持続的な命名 .....	13
2.5.4 ボリュームラベルと UUID スクリプト .....	13
2.6 リプレイビューボリュームの使用 .....	14
2.7 ボリュームの拡張 .....	16
2.8 ボリュームの削除 .....	17
2.9 SAN から起動 .....	19
2.9.1 SAN リプレイボリュームからの起動のキャプチャ .....	23
2.9.2 SAN リプレイビューボリュームの起動からのリカバリ .....	26
3 サーバ構成 .....	27
3.1 Fibre Channel および modprobe .....	27
3.1.1 構成 .....	27
3.1.2 modprobe および mkinitrd の再ロード .....	27
3.2 iSCSI .....	28
3.2.1 構成 .....	28
3.2.2 iSCSI タイムアウト .....	32
3.2.3 iSCSI および /etc/fstab .....	33



3.3	シリアル接続 SCSI.....	34
3.3.1	SAS ドライバ.....	34
3.3.2	SAS /etc/multipath.conf.....	34
3.3.3	FC/iSCSI および SAS.....	35
3.3.4	Linux で SAS デバイスを識別する.....	35
3.3.5	Dell SCv2000 で SAS デバイスを識別する.....	36
3.3.6	設定されたマルチパス.....	38
3.3.7	SAS キュー深度.....	39
3.3.8	SAS から起動.....	39
3.4	キュー深度の管理.....	39
3.5	/etc/multipath.conf および Storage Center デバイスの定義.....	40
3.5.1	マルチパスエイリアス.....	42
3.6	マルチパス環境.....	42
3.6.1	マルチパス /SCSI タイムアウトとキューイング.....	43
3.6.2	PortDown タイムアウト.....	43
3.6.3	パラメーターの検証.....	44
3.7	シングルパス環境.....	44
3.7.1	PortDown タイムアウト.....	44
3.7.2	パラメーターの検証.....	44
4	パフォーマンスの考慮事項.....	45
4.1	プロファイルの調整.....	46
4.2	複数のボリューム.....	46
4.3	HBA キュー深度.....	48
4.4	SCSI UNMAP/TRIM.....	48
4.5	Linux SCSI デバイスクュー変数.....	49
4.5.1	カーネル I/O スケジューラ.....	49
4.5.2	read_ahead_kb.....	50
4.5.3	nr_requests.....	50
4.6	iSCSI.....	50
5	便利なツール.....	51
5.1	lsscsi コマンド.....	51
5.2	scsi_id コマンド.....	51



5.3	dmsetup コマンド.....	53
5.4	dmesg コマンド.....	54
5.5	/proc/scsi/scsi ファイル.....	54
6	Dell Compellent コマンドユーティリティ .....	55
6.1	Java の検証、CompCU 機能の設定とテスト .....	55
6.2	CompCU を使用して一般的なタスクを自動化する .....	57
6.2.1	CompCU での単一ボリュームの作成.....	57
6.2.2	CompCU でのリプレイおよびリプレイビューの作成 .....	58
6.2.3	CompCU による複数のボリュームの迅速な展開 .....	58
A	追加リソース .....	59
B	設定の詳細.....	60



## エグゼクティブサマリ

Red Hat Enterprise Linux® (RHEL) は、非常に強力でスケーラブルなエンタープライズクラスのオペレーティングシステムです。この文書に示されたベストプラクティスを使用して正しく設定された RHEL オペレーティングシステムは、Dell™ Storage Center を使用するための最適な環境を提供します。これらのベストプラクティスには、ボリューム検出、マルチパス、ファイルシステム、およびキュー深度を管理するための設定に関するガイドラインが含まれます。

この文書では、Storage Center OS (SCOS) バージョン 6.5.x から 6.6.x を搭載したバージョン 7x の RHEL プラットフォームの機能について説明します。多くの場合、これらのタスクを遂行するためのさまざまな方法があるため、この文書では、エンドユーザーおよびシステム Administrator のための参照開始点を提供します。

UNIX および Linux ディストリビューションで最も広く当てはまる場合が多いため、このガイドではほぼ例外なくコマンドラインインタフェース (CLI) に焦点を当てています。



# 1 概要

Storage Center では、ミッションクリティカルなデータの割り当て、管理、使用および保護の複雑性を取り除く、Linux 互換および SCSI-3 対応ディスクボリュームが利用できます。適切に設定された Storage Center では、手間がかかる物理ディスクの構成の演習や管理、および複雑な RAID 設定を行う必要がありません。また、Storage Center は、ストレージ層で RAID 10 の速度と信頼性が得られるため、ボリュームを Linux オペレーティングシステムレイヤ内でさらに RAID で管理する必要がありません。

Storage Center ボリュームでは、ミラーリング、バックアップ、複数のファイルシステム、マルチパス、SAN からの起動、災害復旧などすべての Linux ユーティリティを使用できます。

## 1.1 RHEL 7x の新機能

RHEL 7x は、信頼性、パフォーマンスとスケーラビリティで劇的な向上を実現しています。この文書では、簡単に使用でき、Dell Storage Center 製品と機能と統合できる、新しい RHEL 7x の機能を確認します。

- XFS は、起動、ルートおよびユーザーデータパーティションのための新しいデフォルトファイルシステムです。
- XFS ファイルシステムのサイズ制限は 100 TB から 500 TB に増加し、ext4 ファイルシステムのサイズ制限は 16 TB から 50 TB に増加しています。
- iSCSI および FCoE ターゲットの新しいソフトウェアの実装は、ユーザー領域ではなく、カーネルに置かれています。



## 2 ボリューム管理

Linux でのボリュームの管理方法を理解するには、/sys 擬似ファイルシステムの基本的な理解が必要です。/sys ファイルシステムは、カーネルおよびモジュールの各種要素との対話を許可するファイル構造です。読み取り専用ファイルは現在の値を保存し、読み取り / 書き込みファイルでは、正しいコマンドでイベントをトリガーします。通常は、cat および echo コマンドが、従来のテキストエディタでそれらを開く STDIN 節としてリダイレクトで使用されます。

HBA (FC、iSCSI および SAS) とやりとりするために、/sys/class/scsi\_host/ フォルダにある特殊ファイルに対してコマンドが発行されます。マルチポートカード上の各ポートは固有の HBA を表し、各 HBA ポートには、スキャンを実行し HBA パラメーターを読み取るためのファイルを含む独自の hostX フォルダがあります。このフォルダのレイアウト、ファイルおよび機能性は、HBA ベンダーまたはタイプにより異なる可能性があります (たとえば、QLogic Fibre Channel、Emulex Fibre Channel、ソフトウェア iSCSI ベース HBA または Dell 12 Gbps SAS HBA)。

### 2.1 新規ボリュームのスキャン

QLogic 24xx/25xx シリーズ HBA および Emulex HBA に必要なドライバモジュールは、ベースカーネルコードに含まれています。次の手順は、デフォルトの HBA ドライバモジュールに適用されます。ベンダー (QLogic、Emulex) 独自のドライバが使用されている場合は、手順と詳細についてベンダーのドキュメントを参照します。

Linux ホストは、オンザフライで LUN ID 0 を検出できません。LUN ID 0 は、起動中にのみ検出でき、SAN 環境からの起動ではオペレーティングシステムボリューム用に予約されます。他のすべてのボリュームは、LUN ID 1 以上に関連付ける必要があります。

次のサンプルスクリプトは、いかなる種類の表明された保証またはサポートも一切なく、現状で提供されます。このスクリプトは Linux オペレーティングシステムのメジャーリビジョン番号を特定し、echo コマンドを /sys/class/scsi\_host/ フォルダ内にある対応する hostX デバイスに適用します。このスクリプトでは、HBA ポートをスキャンし、ストレージレイからホストに提示される既存および新しいボリュームを検出し、特定します。このスクリプトは、ホストに提示される FC と iSCSI 両方のデバイスの検出に使用でき、Red Hat Linux バージョン 5x ~ 7x に適用されます。

STDOUT はこのスクリプトからは生成されません。/var/log/messages の内容または or the output from the dmesg または lsscsi コマンドの出力をチェックして、新しく検出されたボリュームがあれば識別します。

新規ボリュームのマッピングおよび検出中に HBA を再スキャンしても、ホストに悪影響はありません。

```
#!/bin/bash
```

```
OSMajor=`uname -r | awk -F.'{'{print $(NF-1)}'`
```

```
echo "INFO: OS Major rev.${OSMajor} detected!"
```

```
if [ "${OSMajor}" = "el7" -o "${OSMajor}" = "el6" ]; then
for i in /sys/class/scsi_host/*
```





```

do
    echo "- - -" >> ${i}/scan
done
elif [ "$(uname -r | awk -F.'{'{print $(NF)}')" = "el5" ]; then
    echo "INFO: OS Major rev.el5 detected instead!"
    for i in /sys/class/scsi_host/*
    do
        echo 1 >> ${i}/issue_lip
        echo "- - -" >> ${i}/scan
    done
else
    echo "WARN: OSMajor parameter of unknown value, exit 1"
    exit 1
fi

```

### このスクリプトの出力例

```

# ./scan_bus.sh
INFO: OS Major Rev.el7 detected!

```

また、sg3\_utils パッケージをインストールすると、/usr/bin フォルダにあるネイティブの Red Hat コマンド **rescan-scsi-bus.sh** が利用できるようになります。

```

# /usr/bin/rescan-scsi-bus.sh
Scanning SCSI subsystem for new devices
[snip]
0 new or changed device(s) found.
0 remapped or resized device(s) found.
0 device(s) removed.

```

## 2.2 パーティション

ブートボリューム以外のボリュームにはパーティション（およびパーティションテーブル）は必要ありません。ドライブ全体として、Storage Center にプロビジョニングされたボリュームを使用することをお勧めします。これにより、ボリュームがプロビジョニングされているすべてのディスクにまたがるボリュームのワイドストライピングという Storage Center ネイティブの長所を活用できます。RHEL 7x およびデフォルトでのインストールの場合、起動ボリュームは 2 つのパーティションに分割されます。XFS ファイルシステム（RHEL 7x でのデフォルト）は、ブートパーティションに適用され、その他のパーティションは論理ボリュームマネージャによって管理されます。

```

# parted
GNU Parted 3.1
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) select /dev/sda
Using /dev/sda
(parted) print

```



```
Model: SEAGATE ST9146803SS (scsi)
Disk /dev/sda: 147GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:
```

Number	Start	End	Size	Type	File system	Flags
1	1049kB	525MB	524MB	primary	xfs	boot
2	525MB	147GB	146GB	primary		lvm

## 2.3 論理ボリュームマネージャ

論理ボリュームマネージャ (LVM) は、Linux のボリュームに適用し、管理に使用できます。LVM メタデータ (LVM 署名など) をボリュームにインストールして、物理ボリューム (PV)、論理ボリューム (LV) およびそれに応じてボリュームグループ (VG) を一意に識別します Storage Center ボリュームをさらに含めて管理するための LVM の使用 (vgexport、vgexport およびその他の LVM 管理コマンドの使用を含む) は本書の範囲外です。

ベストプラクティスとして、ドライブ全体で Storage Center にプロビジョニングされたボリュームを使用してください。ただし、ストレージ層で提供されていない機能やメリットが得られる場合は、LVM を使用できます。

### 2.3.1 LVM の設定および SCSI UNMAP/TRIM

LVM を Storage Center ボリュームに使用する場合、LVM は SCSI UNMAP/TRIM コマンドに従い、これらのコマンドを Storage Center に戻すように設定できます。

このシナリオは、論理ボリュームがメンバーボリュームグループから削除された場合に適用されます。このときに削除される論理ボリュームからリカバリされる領域は、解放されてページプールに返される Storage Center ページと関連しています。

LVM の設定は、/etc/lvm/lvm.conf ファイルを編集することにより実行されます。

1. /etc/lvm/lvm.conf ファイルを編集します。
2. キー値ペア「issue\_discards = 0」を「issue\_discards =1」に変更します。
3. 変更を保存し、エディタセッションを終了します。

## 2.4 SCSI UNMAP/TRIM およびファイルシステム

SCSI UNMAP/TRIM の動作は、ファイルシステム (ext4、xfs または btrfs など) で動作するようにも設定できます。

このシナリオでは、ファイルとディレクトリが、ext4、xfs または btrfs ファイルシステムから削除されている場合に適用されます。このときに削除されるファイルとディレクトリからリカバリされる領域は、解放されてページプールに返される Storage Center ページと関連しています。

ファイルシステムで SCSI UNMAP/TRIM 機能を有効にするには、mount コマンドを使用します。この mount コマンドのパラメーターはファイルシステムのタイプ (ext4、xfs または btrfs) に関係なく、同様の方法で適用されます。



```
# mount -o discard /dev/mapper/<volume_name> /<mountpoint_name>
```

この mount パラメーターは、ファイルシステムの /etc/fstab ファイルに適切なフラグを追加して、再起動しても保持されるようにすることができます。

```
# cat /etc/fstab
[snip]
/dev/mapper/VolGroup00-home /home          xfs     defaults          1 2
/dev/mapper/VolGroup00-swap swap      swap     defaults          0 0
/dev/mapper/<volume_name>  /<mountpoint_name>  xfs     defaults,discard  0 0
[snip]
```

## 2.5 持続的なデバイス管理

Linux で検出されたボリュームは、サーバを SAN に接続する HBA ポートによって使用される Linux の検出方法に応じて、/dev/sdd および /dev/sdf などのデバイス指定が与えられます。

これらの /dev/sdX デバイス名の他の用途に、/etc/fstab ファイルのエントリを含む、mount コマンドのボリュームの指定があります。静的ディスク環境では、/dev/sdX デバイス名は、/etc/fstab ファイルのエントリに対して問題なく動作します。ただし、Fibre Channel または iSCSI 接続の動的な性質により、Linux は再起動後にこれらのディスクの指定を追跡できません。

これらのボリュームが、持続的な命名スキームにより割り当てられ、参照されるようにする複数の方法があります。本項では、/dev/sdX 参照ボリュームでのボリュームラベルまたは汎用一意識別子 (UUID) の使用について説明します。ボリュームラベルは、Storage Center Replay リカバリのスクリプトを作成するとき、非常に便利です。例として、サーバをバックアップするために、マウントするには、実稼働 Storage Center Replay のリプレイビューボリューム取り上げます。この場合、リプレイビューボリュームは、関連付けられている /dev/sdX デバイスを明示的に識別することなく、ボリュームラベルでの参照が可能です。ボリュームラベルは、ボリューム内に保存されたメタデータであり、Storage Center Replay から切り取られたリプレイビューボリュームに継承されます。

ボリュームラベルまたは UUID は、マルチパス環境内でも使用できます。つまり、マルチパス /dev/mapper/mpathX デバイス名（または、マルチパスエイリアス。3.5.1「[マルチパスエイリアス](#)」の項を参照）はデフォルトで持続的であり、再起動後も変更されません。ボリュームラベル、UUID またはマルチパス デバイス名は、ローカル Linux ホストの /etc/fstab ファイルにあるエントリに対して置き換えて使用できます。

リプレイビューボリュームが、ボリュームのリカバリまたは代替 Linux ホストへの再配置に使用される場合、ボリュームラベルまたは UUID の使用を推奨します。これらの値はボリュームを一意に識別し、一方で、マルチパス名は (/etc/multipath.conf ファイルの設定により) 異なるためです。

### 2.5.1 ファイルシステムのボリュームラベル

ファイルシステムのボリュームラベルは、ボリューム上でファイルシステムを作成するとき、または続けて異なるコマンドを使用することにより、適用できます。異なるファイルシステム（たとえば、ext4 または xfs）には異なるファイルシステムメタデータスキーマがあり、異なるコマンドを使用して、このデータを表示、管理および変更します。



ファイルシステムのボリュームラベルは、/dev/disk/by-label/ フォルダで作成されます。このフォルダ内のエントリは、/dev フォルダ内のそれぞれのデバイスへのシンボリックリンクとして作成されます。/dev/disk/by-label/ フォルダは動的に管理され、Linux ホスト上のボリュームのいずれにもボリュームラベルが適用されていない場合は、存在しません。

次の例は、これらのコンセプトのいくつかを示しています。2.5.4“[ボリュームラベルと UUID スクリプト](#)”の項にあるサンプルのスクリプトは、すべての Linux マルチパスデバイス名を解析し、そのマルチパス UUID、既知のファイルシステムの種類、ファイルシステムのボリュームラベルおよびファイルシステム UUID 値を STDOUT に提示します。このスクリプトは、いかなる種類の表明された保証またはサポートも一切なく、現状で提供されます。

- ファイルシステムとラベルを同時に適用するには、以下の手順を実行します。

```
# mkfs.ext4 -L My_ext4_vol /dev/sdX
# mkfs.xfs -L My_xfs_vol /dev/mapper/mpathX
```

- ボリュームラベルを既存のファイルシステムに適用するには、以下の手順を実行します（ファイルシステムのタイプが異なると、異なるコマンドが使用されることに注意してください）。

```
# tune2fs -L My_ext4_vol /dev/sdX
# xfs_admin -L My_xfs_vol /dev/mapper/mpathX
```

- ボリュームラベルを既存のファイルシステムから削除するには、以下の手順を実行します（ファイルシステムのタイプが異なると、異なるコマンドが使用されることに注意してください）。

```
# tune2fs -L "" /dev/sdX
# xfs_admin -L "--" /dev/mapper/mpathX
```

## 2.5.2 汎用一意識別子

UUID 値はボリュームメタデータの複数のレイヤに存在し、変化しません。UUID は Linux で各マルチパスデバイスを一意に識別および管理するため、デバイスマッピングドライバによって作成されて使用されます。UUID は、LVM メタデータ内で作成されます（ボリュームが LVM で管理されている場合）。UUID もファイルシステムレイヤ内で作成、管理され、各ファイルシステムのボリュームを一意に識別します。UUID は、マルチパスの定義、LVM またはファイルシステムの作成時に動的に作成されます。UUID は、これらのレイヤ内でそれぞれ他のレイヤとは無関係に変更できますが、目的が確実に明確でない限り、お勧めしません。

UUID エントリは、/dev/disk/by-uuid/ フォルダで作成されます。このフォルダ内のエントリは、/dev フォルダ内のそれぞれのデバイスへのシンボリックリンクとして作成されます。

2.5.4“[ボリュームラベルと UUID スクリプト](#)”の項にあるサンプルのスクリプトは、すべての Linux マルチパスデバイス名を解析し、そのマルチパス UUID、既知のファイルシステムの種類、ファイルシステムのボリュームラベルおよびファイルシステム UUID 値を STDOUT に提示します。このスクリプトは、いかなる種類の表明された保証またはサポートも一切なく、現状で提供されます。



### 2.5.3 /etc/fstab および /boot/grub2/grub.cfg での持続的な命名

LABEL= または UUID= 構文は、mount コマンド、/etc/fstab ファイルおよび /boot/grub2/grub.cfg ファイルのエントリまたはスワップパーティションを含むさまざまな場所でボリュームを参照するために使用できます。これにより、検出デバイス名の指定、マルチパス設定または定義済みのマルチパスエイリアスとは無関係に、ボリュームを一意に識別することができます。

下のサンプル /etc/fstab 出力は、これらの概念のいくつかの使用方法を示しています。

```
# cat /etc/fstab
[snip]
/dev/mapper/VolGroup00-home /home          xfs     defaults          1 2
UUID=8284393c-18aa-46ff-9dc4-0357a5ef742d  swap          swap defaults 0 0
LABEL=TestVol /vol_001          xfs     defaults,discard  0 0
[snip]
```

下のサンプル /etc/grub2.conf 出力は、これらの概念のいくつかの使用方法を示しています。

```
# cat /boot/grub2/grub.cfg
[snip]
linux16 /vmlinuz-3.10.0-123.el7.x86_64 root=UUID=35bealc0-ce32-42a4-8e36-
72fd5e77471d ...
[snip]
```

### 2.5.4 ボリュームラベルと UUID スクリプト

このサンプルスクリプトでは、vendor="COMPELNT" として識別されたすべてのデバイスを解析して、ボリュームに適用されたファイルシステムがあるかどうかを判断し、検出されたディスクラベルの値、mpath-layer UUID および disk-layer UUID 値を抽出して表示します。このスクリプトは、いかなる種類の表明された保証または一切のサポートなく、現状で提供されます。

```
#!/bin/bash

MP=/usr/sbin/multipath
BLKID=/usr/sbin/blkid

for i in `${MP} -ll | grep COMPELNT | cut -d" " -f1 | sort -k1`
do
    echo "INFO: /dev/mapper/${i}"
    FsTyp=`${BLKID} /dev/mapper/${i} | awk -F" " '{print $(NF)}' | cut -d= -
f2 | cut -d\" -f2`
    if [ "${FsTyp}" = "" ]; then
        echo "WARN: No filesystem detected"
    else
        echo "Multipath Info:"
        /usr/sbin/dmsetup info /dev/mapper/${i} | grep UUID
        echo "Filesystem Details:"
        echo "Type: ${FsTyp}"
    fi
done
```



```

        case ${FsTyp} in
            xfs)
                Cmd="/usr/sbin/xfs_admin -lu"
                ${Cmd} /dev/mapper/${i}
                ;;
            ext4|ext3)
                Cmd="/usr/sbin/tune2fs -l"
                ${Cmd} /dev/mapper/${i} | egrep 'volume|UUID'
                ;;
            *)
                echo "WARN: Filesystem unknown"
                ;;
        esac
    fi
    echo
done

```

このスクリプトの出力例を下に示します。

```

# ./get_UUID.sh
[snip]
INFO: /dev/mapper/vol_001
Multipath Info:
UUID: mpath-36000d31000006500000000000000017f2
Filesystem Details:
Type: xfs
label = "TestVol"
UUID = fbe5bfb-94b9-4827-a103-516045b9b608

INFO: /dev/mapper/vol_002
Multipath Info:
UUID: mpath-36000d31000006500000000000000017f3
Filesystem Details:
Type: LVM2_member
WARN: Filesystem unknown
[snip]

```

## 2.6 リプレイビューボリュームの使用

XFS ファイルシステムを使用すると、Storage Center Replay とリプレイビューボリュームを簡単かつ効果的に統合できます。ボリューム上に常駐している適用済み XFS ファイルシステムは、ボリューム UUID を含むボリューム上で XFS メタデータを作成、保守および管理します。このボリューム UUID は、接続されたホストに関係なく、Linux がこのボリュームを一意に識別できるように支援します。

これらの方法は、どの Linux ホストでも XFS ファイルシステム管理に適用できます。リプレイビューボリュームが代替 Linux ホスト（UUID あり）に提示されている場合、そのボリュームは、次のいずれの方法も必要とせずに、正常にマウントされます。



既知の XFS ファイルシステムのボリュームラベルおよび UUID は、下記のコマンドで表示できます。

```
# xfs_admin -ul /dev/mapper/vol_001
UUID = fbef5bfb-94b9-4827-a103-516045b9b608
label = "TestVol"
```

ファイルシステム上のこの UUID は、元の Storage Center ボリュームから作成されたいずれのリブレイまたはリブレイビューボリュームに継承されます。したがって、（マルチパスデバイスには独自の固有 ID がありますが）どのリブレイビューボリュームも自動的に元のホストに提示し、マウントすることはできません。下の例では、リブレイビューボリュームは、XFS ファイルシステムベースの /dev/mapper/vol\_001 デバイスから作成され、/vol\_001 としてマウントされます。このリブレイビューボリュームは、/etc/multipath.conf ファイルを使用して /dev/mapper/vol\_004 として検出され、識別され、エイリアスが作成されます。

**メモ：**UUID が固有ではなく、Linux ホストですでに使用されているため、/dev/mapper/vol\_004 を /vol\_004 としてマウントしようとする試行は失敗します。

```
# df -k
Filesystem                1K-blocks      Used Available Use% Mounted on
[snip]
rhevm:/nfsroot            103213056 54833152   43137024   56% /Tools
/dev/mapper/vol_001        52418560  1939052   50479508    4% /vol_001
[snip]
```

/dev/mapper/vol\_004 to /vol\_004 をマウントしようすると、下記のエラーが返されます。

```
# mount -o discard, sync /dev/mapper/vol_004 /vol_004
mount: wrong fs type, bad option, bad superblock on /dev/mapper/vol_004,
       missing codepage or helper program, or other error
```

In some cases useful info is found in syslog - try  
dmesg | tail or so.

/dev/mapper/vol\_004 を nouuid パラメーターとともに /vol\_004 にマウントする試行では、ボリュームは正常に /vol\_004 にマウントされます。

```
# mount -o discard, sync, nouuid /dev/mapper/vol_004 /vol_004; df -k

# df -k
Filesystem                1K-blocks      Used Available Use% Mounted on
[snip]
rhevm:/nfsroot            103213056 54833152   43137024   56% /Tools
/dev/mapper/vol_001        52418560  1939052   50479508    4% /vol_001
/dev/mapper/vol_004        52418560  1939052   50479508    4% /vol_004
[snip]
```

また、/dev/mapper/vol\_004 に関連付けられた UUID 値を恒久的に変更し、mount コマンドで nouuid パラメーターを指定する必要があるようにすることができます。これは、以下に示すように実行できます（アンマウントされたファイルシステムに適用されます）。

```
# xfs_admin -U generate /dev/mapper/vol_004
Clearing log and setting UUID
writing all SBs
new UUID = 26b4a32d-4a3a-405f-899c-7bb63087cc7b
```





## 2.7 ボリュームの拡張

RHEL 7x では、Linux でのボリュームとファイルシステムの拡張は単純で効果的です。基盤となる Storage Center ボリュームのサイズを拡張した後、ボリュームの上に常駐するファイルシステムを基盤となるボリュームの限界値までサイズを変更するか、またはシステム Administrator によって指定される事前設定されたサイズに変更できます。ボリュームおよび XFS ファイルシステム拡張のプロセスを以下のステップで説明します。

1. Storage Center ボリュームを拡張します。
  - a. Storage Center ボリュームを右クリックします。
  - b. **Expand Volume** (ボリュームの拡張) を選択します。
  - c. ボリュームの新しいサイズを入力します。
  - d. **OK** をクリックして、拡張を実行します。
2. `# echo 1 >> /sys/block/sdX/device/rescan` を実行して、マルチパスデバイスの各パスのドライブジオメトリを再スキャンします。
3. `# multipathd -k "resize map <devicename>"` を実行して、マルチパスマップのサイズを変更します。  
メモ：
  - a. ここで、マルチパスデバイスは `/dev/mapper/<devicename>` です
  - b. `-k` パラメーターとコマンド文字列間にスペースはありません
4. XFS ファイルシステムを拡張します (この例では、基盤となるボリュームの境界制限サイズまで)
  - a. `# xfs_growfs -d /<mountpoint_name>`

オンザフライでのボリューム (またはそれに付随するファイルシステム) のサイズの縮小はどの環境でも推奨されません。リスクとデータ整合性に対する潜在的な影響を大幅に縮小し、この結果を達成する代替手段には、バックアップおよびリカバリ、ボリュームの内容のサイズを小さくした別の新しい Storage Center ボリュームへの再配置、またはボリュームのミラーリングと分割などがあります。

このサンプルスクリプトは、コマンドライン (`/dev/mapper/<devicename>` または `<devicename>` のみ) から入力を受信し、マルチパスデバイスをサポートするさまざまな `/dev/sdX` デバイス名を特定します。次に、それに応じて、指定されたマルチパスデバイスの各パスのボリューム形状を再スキャンしてアップデートするために、`echo 1` の `/sys/block/sdX/device/rescan` ファイルに進みます。このスクリプトは、いかなる種類の表明された保証または一切のサポートなく、現状で提供されます。

```
#!/bin/bash

DevMapperDevice=${1}

if [ "${DevMapperDevice}" = "" ]; then
    echo "ERROR: No /dev/mapper/<devicename> defined, Exiting"
    exit 1
fi

for i in `multipath -l ${DevMapperDevice} | egrep -v "${basename
${DevMapperDevice}}|^size|policy" | awk '{print $3}'`
do
```





```

        echo "INFO: Rescanning /dev/${i} device"
        echo 1 >> /sys/block/${i}/device/rescan
done

echo "INFO: Multipath device /dev/mapper/${(basename ${DevMapperDevice})}
rescanned successfully"

```

このスクリプトの出力例を下に示します。

```

# ./rescan_volume.sh /dev/mapper/vol_001
INFO: Rescanning /dev/sdf device
INFO: Rescanning /dev/sdk device
INFO: Rescanning /dev/sdg device
INFO: Rescanning /dev/sdj device
INFO: Multipath device /dev/mapper/vol_001 rescanned successfully

```

## 2.8 ボリュームの削除

Linux は、提示された各ボリュームについての情報を保存します。たとえボリュームが Storage Center でマップされていない状態でも、Linux では、次の再起動まで、そのボリュームに関する情報を保持します。Linux が再起動する前に、同じ LUN ID を使用して同じターゲットからボリュームを提示された場合、そのボリュームに関する古いデータを再利用します。これは、ボリュームの誤情報と管理ミスにつながり、ビジネス環境におけるデータの整合性に影響を与える可能性があります。

ボリュームが使用されなくなつたと見なされた後は、常に Linux 上のすべてのボリューム情報をアンマウントし、取り外し、削除することをお勧めします。Linux ボリュームメタデータのこの管理は、ボリューム自体に保存されているすべての実際のデータに対しても非破壊的です。

マルチボリュームを削除するプロセスは、以下のステップで説明します。

1. マウントされたボリュームに対する I/O を停止します。
2. ボリュームをアンマウントします。
3. /etc/multipath.conf を編集して、このボリュームを参照するすべての構文を削除します。
4. multipathd デーモンを再ロードします (systemctl が multipathd サービスを再スタートします)。
5. 以下のこのスクリプトを実行して、マルチパスバックアップデバイスファイルを削除します。
6. Storage Center からのボリュームへのマッピングがあればすべて削除します。

このサンプルスクリプトは、コマンドライン (/dev/mapper/<devicename> または <devicename> のみ) から入力を受信し、マルチパスデバイスをサポートするさまざまな /dev/sdX デバイス名を特定します。次に、それに応じて、echo 1 の /sys/block/sdX/device/delete ファイルに進み、Linux からボリュームメタデータをクリアします。このスクリプトは、いかなる種類の表明された保証または一切のサポートなく、現状で提供されます。

```

#!/bin/bash

DevMapperDevice=${1}

if [ "${DevMapperDevice}" = "" ]; then

```



```

        echo "ERRO: No /dev/mapper/<devicename> defined, Exiting"
        exit 1
    fi

    for i in `multipath -l ${DevMapperDevice} | egrep -v "$(basename
    ${DevMapperDevice})|^size|policy" | awk '{print $3}'`
    do
        echo "INFO: Removing /dev/${i} device"
        echo 1 > /sys/block/${i}/device/delete
    done

    echo "INFO: Multipath device /dev/mapper/${(basename ${DevMapperDevice})} removed
    successfully"

```

このスクリプトの出力例を下に示します。

```

# ./rm_volume.sh /dev/mapper/vol_001
INFO: Removing /dev/sdf device
INFO: Removing /dev/sdk device
INFO: Removing /dev/sdg device
INFO: Removing /dev/sdj device
INFO: Multipath device /dev/mapper/vol_001 removed successfully

```

単一のボリューム（マルチパスによって管理されていない）を削除するには、次の手順に従ってください。

1. マウントされたボリュームに対する I/O を停止します。
2. ボリュームをアンマウントします。
3. コマンドプロンプトから、`# echo 1 > /sys/block/sdX/device/delete` を実行します
4. Storage Center からのボリュームへのマッピングがあればすべて削除します。

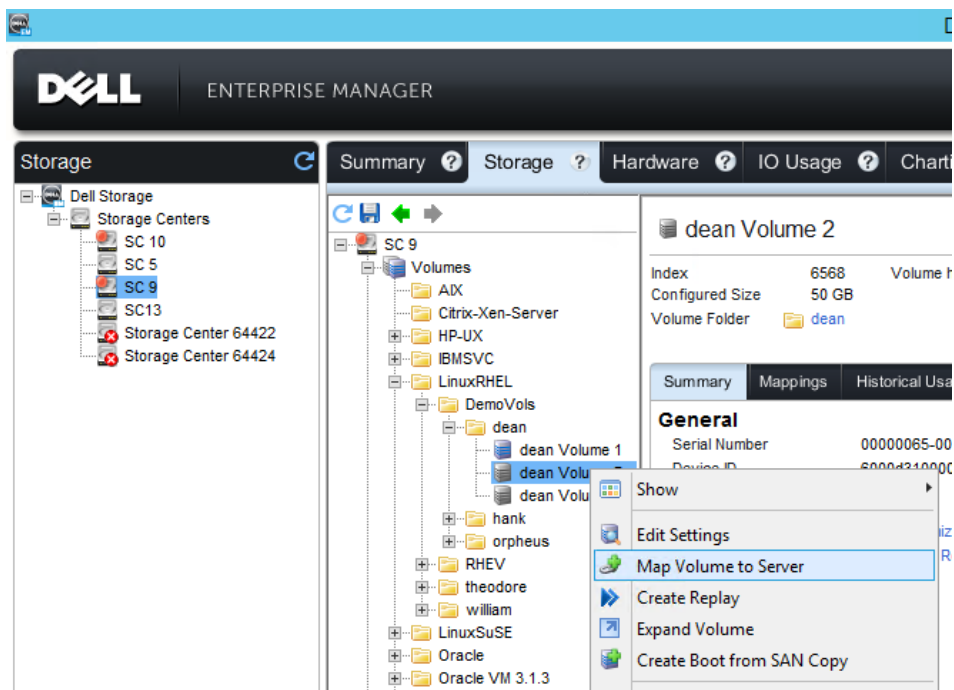


## 2.9 SAN から起動

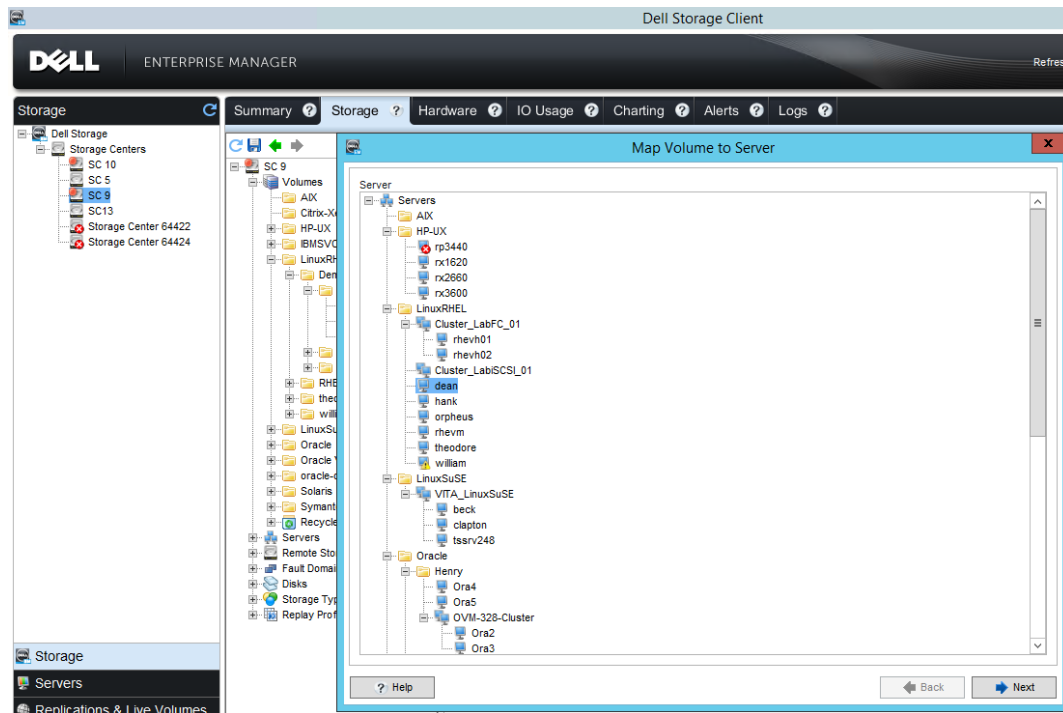
Linux で Storage Center ボリュームをブータブルボリュームとして使用する機能により、システムとストレージの Administrator は、Storage Center Replay およびリプレイビューボリュームテクノロジのメリットをさらに活用できます。Linux の起動ボリュームのリプレイの 2 つの用途は、バックアップ / リカバリメカニズム、およびアップグレードの前にオペレーティングシステムの状態を特定の時点で保持することです。

Storage Center のボリュームをブータブルボリュームとして使用するには、ターゲットボリュームを LUN ID 0 として、ターゲット Linux ホストに提示する必要があります。Storage Center 経由のボリューム LUN ID 0 のマッピングは以下のように実行されます。

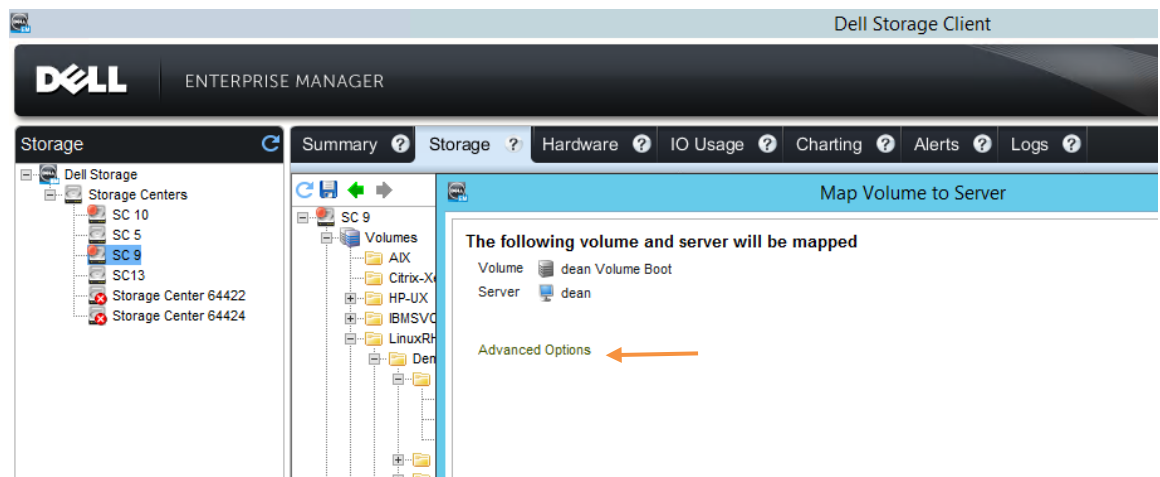
1. ボリュームを右クリックし、**Map Volume to Server**（ボリュームをサーバにマッピング）を選択します。



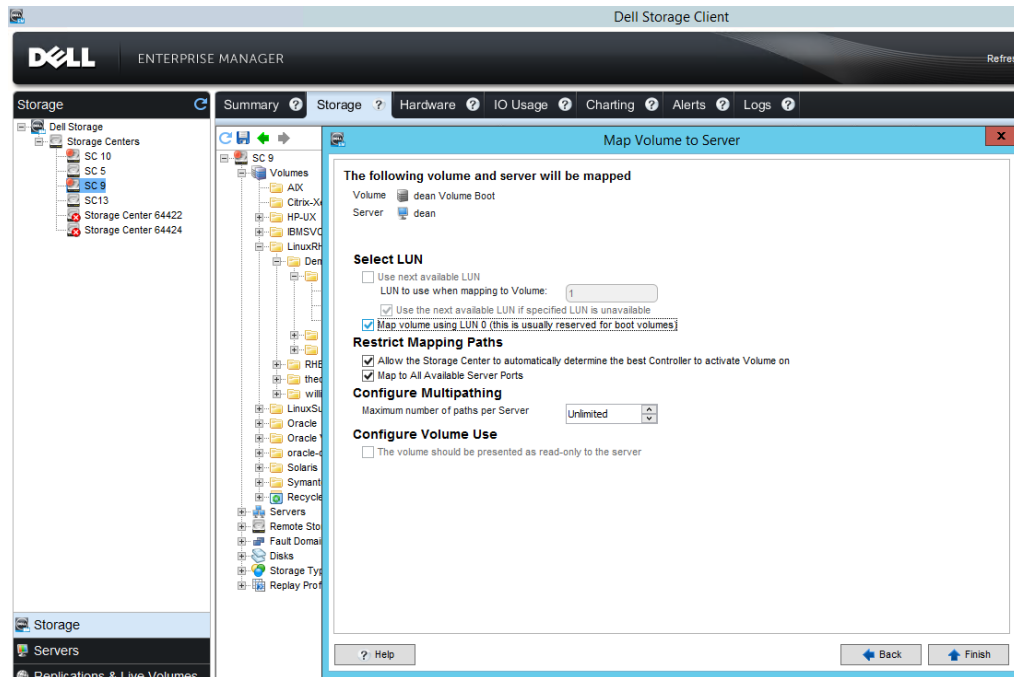
2. ボリュームをマップする必要があるホストを識別し、**Next**（次へ） をクリックします。



3. **Advanced Options**（詳細オプション） をクリックします。



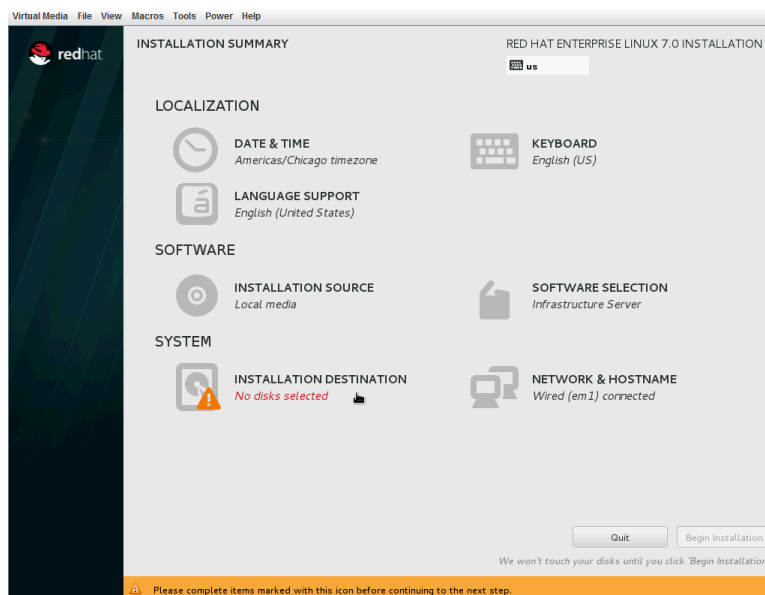
4. **Map volume using LUN 0**（LUN 0 を使用してボリュームをマッピング） チェックボックスを選択し、**Finish**（完了） をクリックして、このボリュームをマップします。



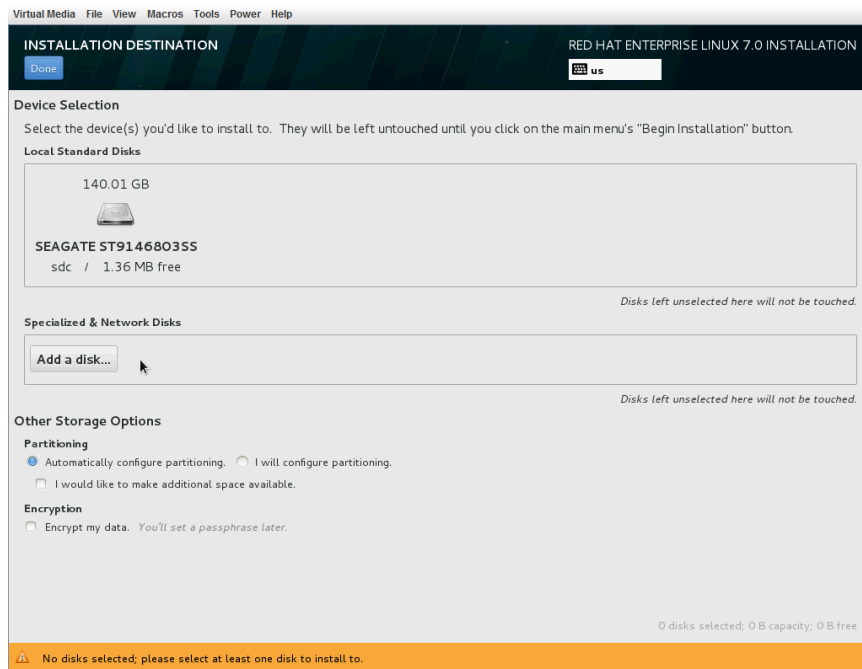
Linux ホストの起動中、HBA BIOS の起動プロセスは中断され、起動デバイスの順序でこの Storage Center ボリュームを優先起動デバイスとして識別するように指示されます。

Linux のインストールプロセス中、下記のように、この Storage Center ボリュームが識別され、インストールターゲットとして選択されます。

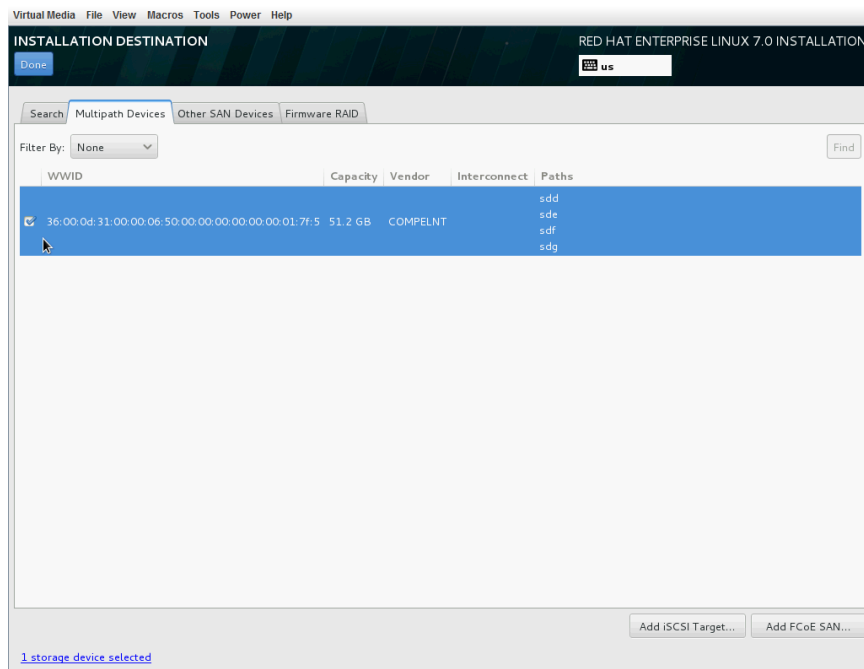
1. **Installation**（インストール）メインメニューで、**Installation Destination**（インストール先）をクリックします。



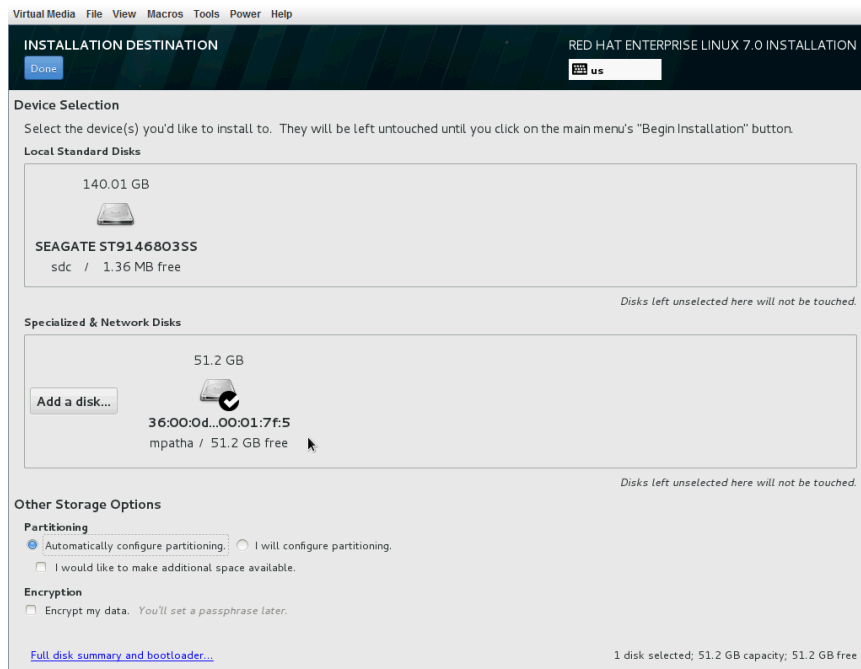
## 2. Add a disk...（ディスクの追加...）をクリックします。



## 3. Multipath Devices（マルチパスデバイス）タブで Storage Center ボリュームを識別し、それに応じてチェックボックスをオンにします。



#### 4. 完了 をクリックします。



上記の Storage Center ボリュームが識別され、Linux オペレーティングシステムの単一のインストールのターゲットとして選択されます。

#### 5. Done (完了) をクリックして、インストールメインメニューに戻ります。

### 2.9.1 SAN リプレイボリュームからの起動のキャプチャ

Storage Center では、SAN ボリュームから起動のリプレイボリュームをキャプチャするのは、簡単に行えます。ただし、リプレイボリュームをキャプチャする前にすべての I/O を停止するために、Linux ホストで LVM 以外でバインドされた XFS ボリュームをフリーズすることを推奨します。

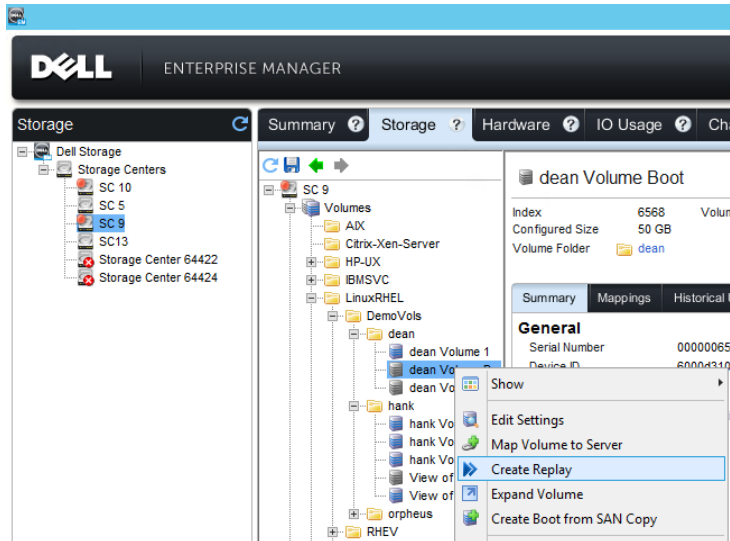
このシナリオ（デフォルトの Linux インストールを使用）では、/ および /boot ファイルシステムは XFS ベースであり、以下のコマンドでフリーズします。

```
# xfs_freeze -f /  
# xfs_freeze -f /boot
```

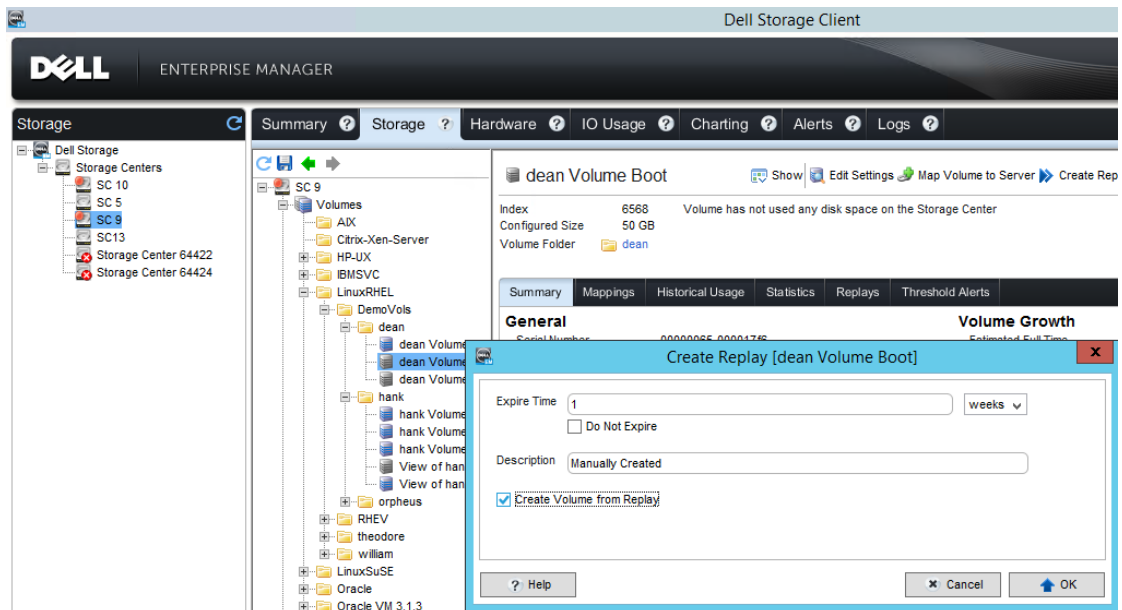


Storage Center 経由で SAN ボリュームから起動のリプレイボリュームをキャプチャするには、次に示す手順を実行します。

1. ボリュームを右クリックし、**Create Replay**（リプレイの作成）を選択します。



2. ボリュームに 1 週間の期限を設定し、OK をクリックしてリプレイを作成します。



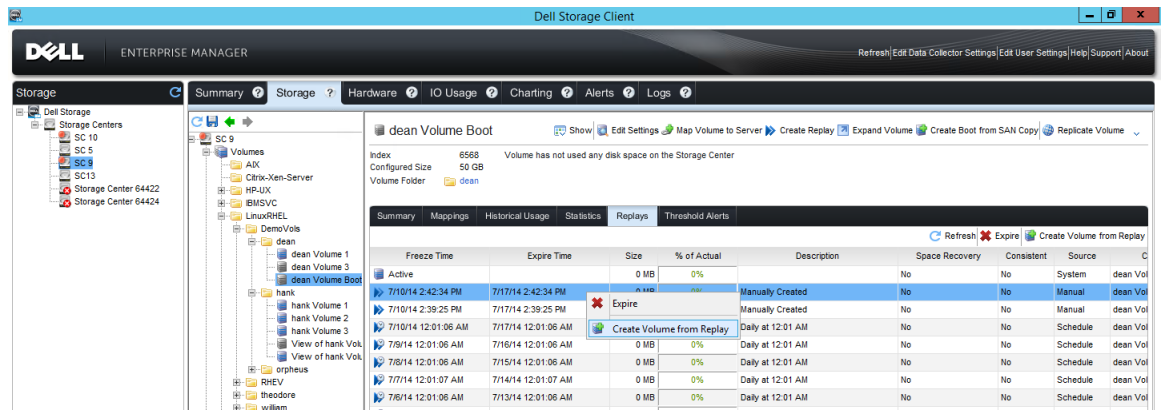
3. リプレイのキャプチャ後は、/ および /boot ファイルシステムは XFS ベースであり、以下のコマンドを使用してフリーズを解除する必要があります。

```
# xfs_freeze -u /
# xfs_freeze -u /boot
```

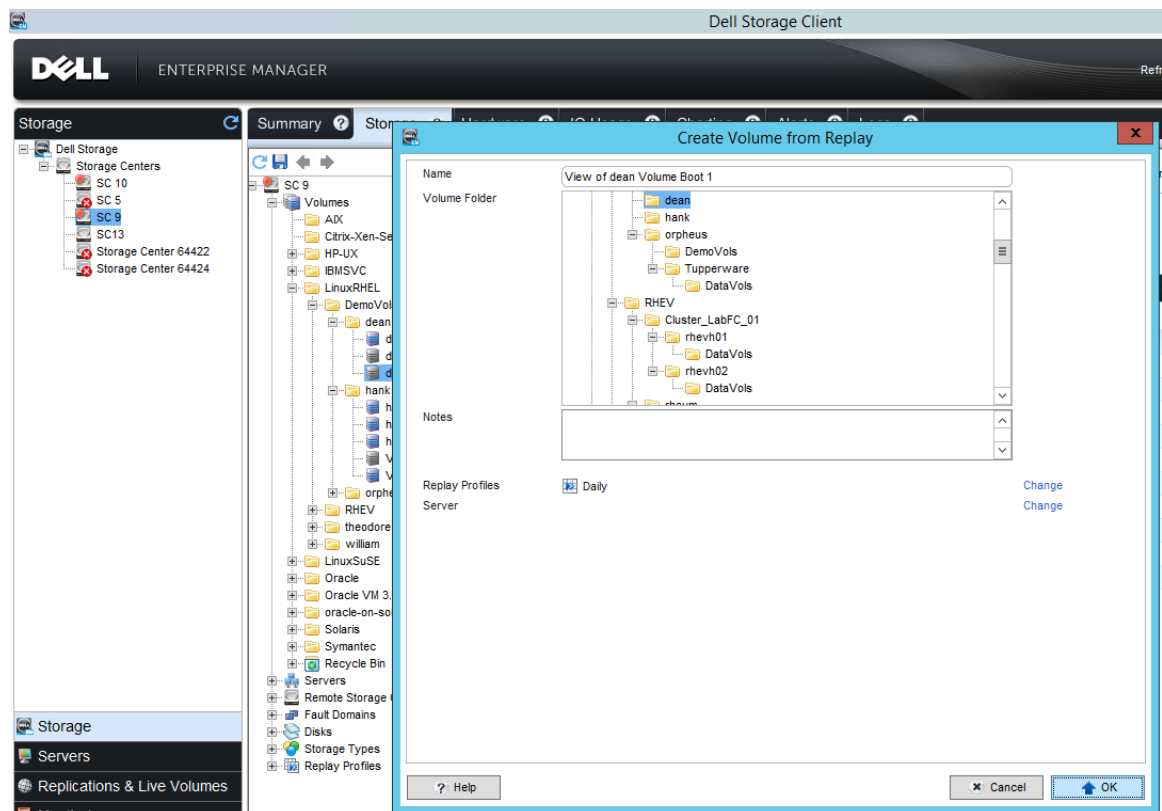
このキャプチャされたリプレイから、ステップ 4 に示すようにリプレイビューボリュームが作成されます。



4. ボリュームの **Replays** (リプレイ) タブで希望のリプレイを右クリックし、**Create Volume from Replay** (リプレイからボリュームを作成) をクリックします。

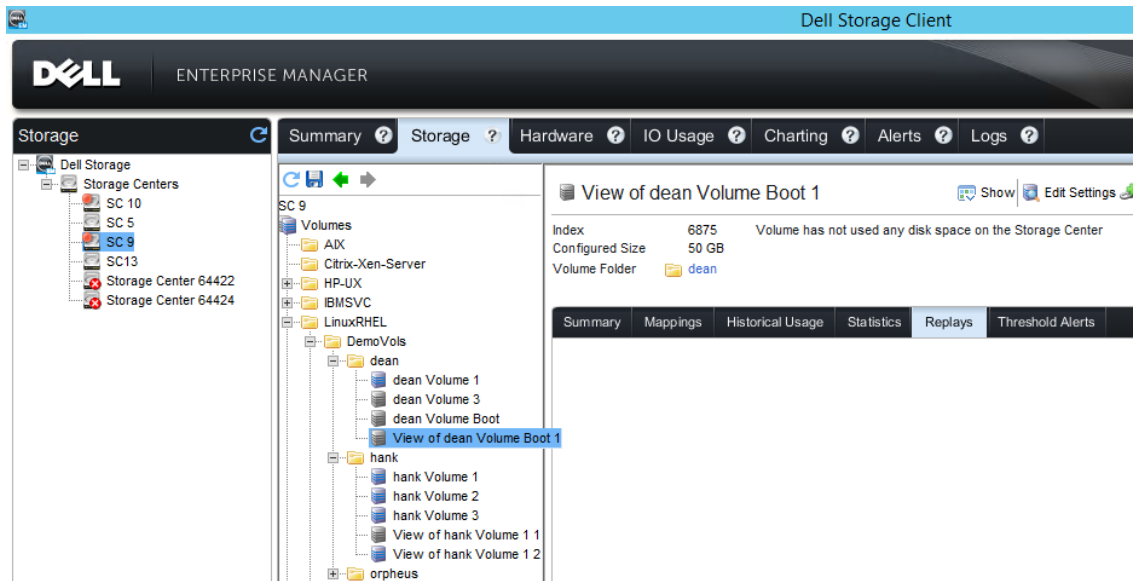


5. このリプレイビューボリュームを適切なサーバホストオブジェクトに関連付け、**OK** をクリックします。



リプレイビューボリュームは、作成されたボリュームの下にある **Volumes** (ボリューム) ツリーに表示されます。





## 2.9.2 SAN リプレイビューボリュームの起動からのリカバリ

SAN ボリュームからの起動のリプレイビューは、同じホストに再度提示されたとき、自動的にブータブルになります。これは、ボリュームがシステムデバイスファイル名ではなく、`/etc/fstab`、`/boot/grub2/grub.cfg` 内の一意で識別する UUID 値によって追跡されるようになったためです。また、`/etc/multipath/bindings` および `/etc/multipath/wwids` 内のエントリとエイリアスも自動的にアップデートされ、起動デバイスの変更を反映します。

交互にマップされるリプレイビューボリュームからの起動を試みるときは、起動時の競合を防ぐため、SAN ボリュームからの元の起動が、サーバオブジェクトにマップされていないことを確認します。リプレイビューボリュームも LUN ID 0 としてホストにマップする必要があります。

## 3 サーバ構成

本項では、Linux ホストの I/O スタックの設定に関するさまざまな面について説明します。Linux I/O スタックは、Linux のホストが動作する環境のニーズに適応するように、正確に設定できます。設定には、HBA ポートの再試行カウント、キュー深度、SCSI および iSCSI デバイスのタイムアウト値などの調整が含まれます。

### 3.1 Fibre Channel および modprobe

Linux ホストの他の面の中で、Linux modprobe 機能は、インストールされた FC HBA（QLogic および Emulex など）の動作パラメーターを管理および設定する手段を提供します。modprobe 機能は、`/etc/modprobe.d` ディレクトリ内のテキストファイルを編集（または作成）することにより設定されます。

#### 3.1.1 構成

QLogic および Emulex HBA は、`/etc/modprobe.d` ディレクトリにある個々のテキストファイル内に記述される設定構文で管理されます。これらのファイルがまだ存在していない場合は、作成する必要があります。QLogic と Emulex のハードウェアの設定構文の詳細は、それぞれ 3.6「[マルチパス環境](#)」および 3.7「[シングルパス環境](#)」で説明されています。これらのファイルが新規の RHEL 7x インストールに存在しない場合は、任意のテキストエディタを使用して手動で作成できます。

- Qlogic の場合：

```
/etc/modprobe.d/qla2xxx.conf
```

- Emulex の場合：

```
/etc/modprobe.d/lpfc.conf
```

#### 3.1.2 modprobe および mkinitrd の再ロード

設定の変更が行われた後、新規または更新された設定が有効になるように、modprobe 機能を再ロードします。

ローカル起動システムの場合、すべての SAN ボリュームをアンマウントしてからモジュールを再ロードすることをお勧めします。下に示すように、モジュールはもう一度（QLogic に）再ロードする前にメモリからアンロードする必要があります。

```
# modprobe -r qla2xxx
# modprobe qla2xxx
```

Emulex ハードウェアで作業する場合は、`qla2xxx` と `lpfc` を置き換えます。SAN ボリュームは、その後に再マウントできます。

SAN システムからの起動の場合、新しい設定が起動時に組み込まれるように、`initramfs-*.img` ファイルも再構築する必要があります。下記の方法は、`initramfs-*.img` ファイルの再構築を示しています。このプロセスでは、同じファイルが既存の場所で上書きされます。この手順を適用する前に、既存の `initramfs-*.img` ファイルをコピーしてバックアップすることをお勧めします。下に示すように、2 つのコマンドを使用できます。

```
# cp /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -r).img.bak
# dracut -f
```



## 3.2 iSCSI

RHEL 7x では、open-iSCSI スタック（RFC3720 に基づく）のアップデートされたソフトウェア実装が導入されています。この新しい iSCSI スタックは、ユーザスペースではなくカーネルメモリ領域に存在します。

iSCSI は、組織が既存のインフラストラクチャを活用しながら、エンタープライズストレージの領域にまでに拡張できる成熟したテクノロジーです。本項では、open-iSCSI スタックのみの設定、使用および実装について説明します。その他、iSCSI ソフトウェアのより高度な実装（カスタム iSCSI HBA ドライバやオフロードエンジンなど）は、この文書の対象外です。必要に応じて、関連するベンダーのマニュアルを参照してください。

### 3.2.1 構成

設定している Linux ホストには、Storage Center の iSCSI ポートと通信できるネットワークポートが必要です。この目的のためには、専用ポート / VLAN の使用をお勧めします。

iSCSI の設定で重要な考慮事項の 1 つは、ネットワークパスです。iSCSI トラフィックに必要な感度、機密性、セキュリティおよびレイテンシを慎重に決定します。iSCSI アーキテクチャのネットワークトポロジ（例：専用物理ポート、VLAN の使用状況、マルチパス、および冗長性）は、これらのニーズにより推進および定義されます。

理想的なシナリオでは、iSCSI トラフィックは専用のポート、スイッチおよびインフラストラクチャの使用によってルーチンネットワークトラフィックから分離および隔離されます。物理トポロジが拘束条件となる場合、VLAN サブネットを使用することにより、iSCSI トラフィックの分離および隔離が推奨されます。また、パスの冗長性を得るために、iSCSI を常にマルチパス設定で使用することもお勧めします。

VLAN サブネットが可能でない場合は、さらに次の 2 つのオプションを検討してください。

- 静的ルートを定義することによって、ネットワーク層でトラフィックをルーティングする
- 設定により iSCSI レベルでトラフィックをルーティングする

次のデモでは、Linux ホストの Storage Center 検出 IP からターゲットをリクエストするプロセス、これらのターゲットへのログインおよび Storage Center における iSCSI ベースのサーバオブジェクトのセットアップについて解説します。新しく検出されたボリュームのスキャンなど、他の方法については、2.1 [「新規ボリュームのスキャン」](#)の項を参照してください。

このシナリオでは、デモの目的から、下記の ifconfig および netstat 出力に示されているように、em2 インタフェースのみが iSCSI VLAN に設定されています。実稼働では、パス、冗長性および I/O キュー管理に dm-multipath ドライバを使用して、Linux ホストごとに複数の iSCSI インタフェースを設定することをお勧めします。

1. ネットワークインタフェースを設定します。

```
# ifconfig -a
[snip]
em2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.10.83.21 netmask 255.255.0.0 broadcast 10.10.255.255
    ether 14:fe:b5:c9:95:7f txqueuelen 1000 (Ethernet)
    RX packets 12670048 bytes 1031419587 (983.6 MiB)
```



```

RX errors 0   dropped 34235   overruns 0   frame 0
TX packets 59   bytes 9454 (9.2 KiB)
TX errors 0   dropped 0 overruns 0   carrier 0   collisions 0
[snip]

```

```

# netstat -rn
Kernel IP routing table
Destination      Gateway          Genmask          Flags   MSS Window  irtt Iface
0.0.0.0          172.16.16.1     0.0.0.0          UG        0 0        0 em1
10.10.0.0        0.0.0.0         255.255.0.0      U         0 0        0 em2
172.16.16.0     0.0.0.0         255.255.240.0    U         0 0        0 em1
224.0.0.0        -               255.255.255.0    !         - -        - -
255.255.255.255 -               255.255.255.255 !H        - -        - -

```

2. em2 インタフェースが設定され、Storage Center 検出 IP アドレスがそれぞれ 10.10.97.2 および 10.10.130.2 である ( ) ことがわかっている (Storage Center 管理 GUI から識別) 状態で、これらの IP アドレスが PING に応答するかどうかを確認します。

```

# ping 10.10.97.2
PING 10.10.97.2 (10.10.97.2) 56(84) bytes of data.
64 bytes from 10.10.97.2: icmp_seq=1 ttl=64 time=0.289 ms
64 bytes from 10.10.97.2: icmp_seq=2 ttl=64 time=0.353 ms
[snip]

```

```

# ping 10.10.130.2
PING 10.10.130.2 (10.10.130.2) 56(84) bytes of data.
64 bytes from 10.10.130.2: icmp_seq=1 ttl=64 time=0.394 ms
64 bytes from 10.10.130.2: icmp_seq=2 ttl=64 time=0.318 ms
[snip]

```

3. 次のコマンドを発行して、Storage Center のターゲットの IQN を要求し、Linux ホストにこれらのターゲットへのログインを要求します。

```

# iscsiadm -m discovery -t sendtargets -p 10.10.97.2:3260
10.10.97.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb2f
10.10.97.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb30
10.10.97.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb32
10.10.97.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb3b

# iscsiadm -m discovery -t sendtargets -p 10.10.130.2:3260
10.10.130.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb31
10.10.130.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb39
10.10.130.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb3a
10.10.130.1:3260,0 iqn.2002-03.com.compellent:5000d3100002bb3c

# iscsiadm -m node --login
Logging in to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb2f, portal: 10.10.97.1,3260] (multiple)

```



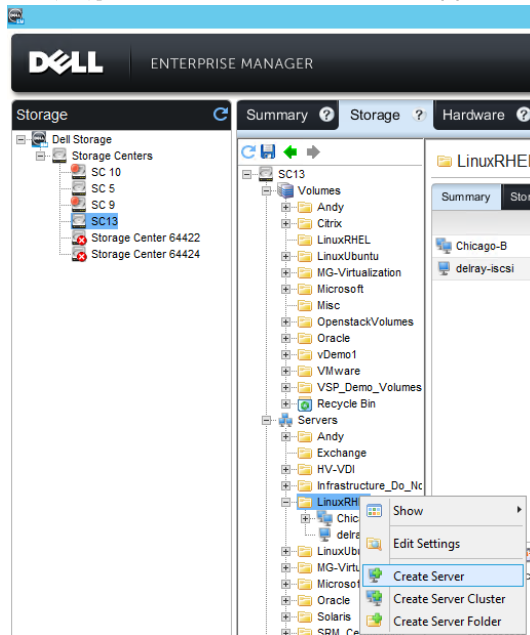
```

Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d3100002bb30, portal: 10.10.97.1,3260] (multiple)
Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d3100002bb32, portal: 10.10.97.1,3260] (multiple)
Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d3100002bb3b, portal: 10.10.97.1,3260] (multiple)
Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d3100002bb31, portal: 10.10.130.1,3260] (multiple)
Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d3100002bb39, portal: 10.10.130.1,3260] (multiple)
Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d3100002bb3a, portal: 10.10.130.1,3260] (multiple)
Logging in to [iface: default, target: iqn.2002-
03.com.compellent:5000d3100002bb3c, portal: 10.10.130.1,3260] (multiple)
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb2f,
portal: 10.10.97.1,3260] successful.
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb30,
portal: 10.10.97.1,3260] successful.
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb32,
portal: 10.10.97.1,3260] successful.
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb3b,
portal: 10.10.97.1,3260] successful.
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb31,
portal: 10.10.130.1,3260] successful.
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb39,
portal: 10.10.130.1,3260] successful.
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb3a,
portal: 10.10.130.1,3260] successful.
Login to [iface: default, target: iqn.2002-03.com.compellent:5000d3100002bb3c,
portal: 10.10.130.1,3260] successful.

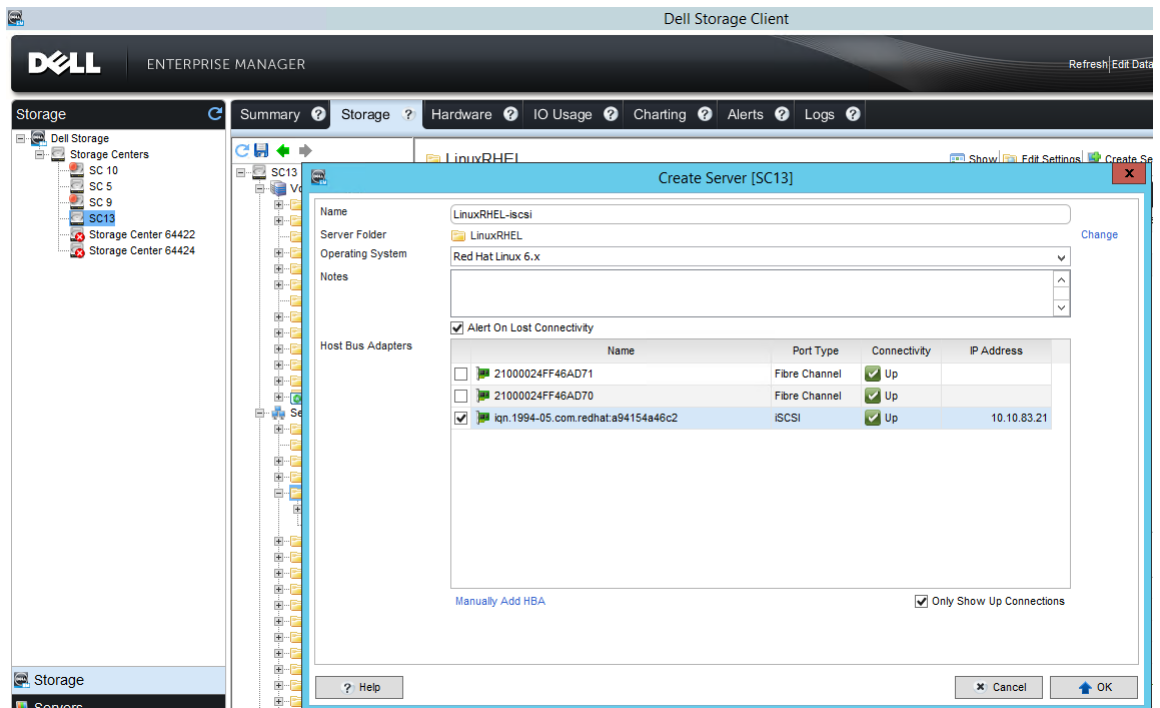
```



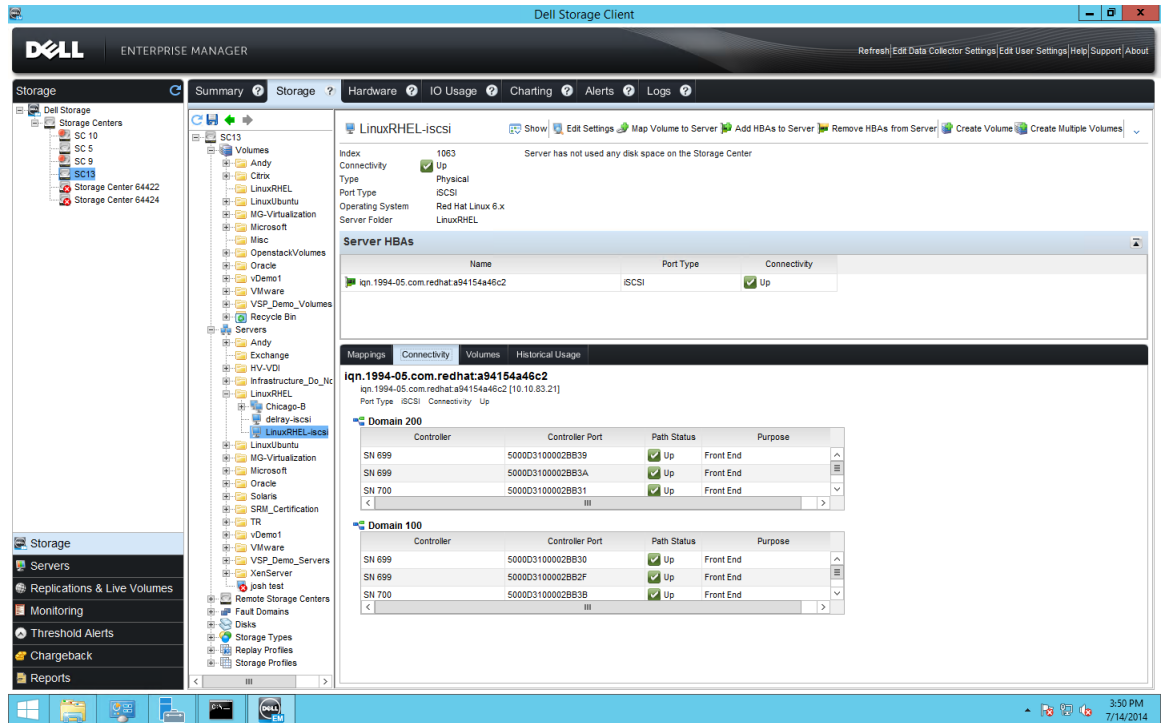
4. Storage Center で新規 iSCSI サーバオブジェクトを設定します。
  - a. 適切な サーバツリーフォルダを右クリックし、**Create Server**（サーバの作成）をクリックします。



5. サーバオブジェクトに名前を付け、オペレーティングシステムを **Red Hat Linux 6.x** に設定し、Linux ホストの IP アドレス（この場合は 10.10.83.21）に対応する IQN チェックボックスを選択して、**OK** をクリックします。



新しい LinuxRHEL-iscsi サーバオブジェクトが LinuxRHEL フォルダに表示され、ボリュームをマッピングする準備ができています。



### 3.2.2 iSCSI タイムアウト

パスが dm-multipath によって管理されるマルチパス iSCSI 環境で、5 秒ごとに SAN での問題発生を監視（NOP-Out 要求を各ターゲットに送信）するように iSCSI を設定します。また、これらの要求を 5 秒ごとにタイムアウトします。この方法で、I/O 要求を代替の引き続きアクティブな iSCSI パスに自動的に再送信することにより、単一パスの回復を待機中のレイテンシを管理し、最小限に抑えます。

ほとんどの SAN ベースのボリューム（Storage Center ボリュームを含む）のコンパイル時のデフォルトには、すべてのパスが使用できなくなった場合に自動的にすべての I/O 要求をキューに登録し再試行するように dm-multipath に指示する、features='1 queue\_if\_no\_path' オプションがあります。これは、下記のサンプル出力で確認でき、そのデフォルトの状態のままにしておく必要があります。

```
# multipath -ll
[snip]
vol_001 (36000d31000006500000000000000017f2) dm-3 COMPELNT,Compellent Vol
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='service-time 0' prio=1 status=active
    |- 1:0:3:1 sdf 8:80 active ready running
    |- 1:0:5:1 sdk 8:160 active ready running
    |- 4:0:4:1 sdg 8:96 active ready running
    `-- 4:0:5:1 sdj 8:144 active ready running
[snip]
```





iSCSI 設定パラメーター NOP-Out およびその関連付けられているタイムアウト値は、`/etc/iscsi/iscsid.conf` ファイルのこれら 2 つのエントリにより管理されます。両方のエントリで、デフォルトが RHEL 7x の優先値 5 に設定されます。

```
node.conn[0].timeo.noop_out_interval = X
node.conn[0].timeo.noop_out_timeout = X
```

マルチパス iSCSI の使用例では、`/etc/iscsi/iscsd.conf` ファイルでのこの追加パラメーターは、デフォルト値の 120 から 5 に設定する必要があります。これにより、iSCSI レイヤーは最大 5 秒間待ってからこのパスへのコマンドを失敗とし、レイテンシと I/O の待機時間を最小化します。

```
node.conn[0].timeo.replacement_timeout = 5
```

### 3.2.3 iSCSI および `/etc/fstab`

iSCSI は動作しているネットワークに依存し、`/etc/fstab` に追加されたボリュームはネットワーク依存に指定する必要があります。言いかえると、ネットワークレイヤーサービスがスタートアップを完了し、ネットワークが動作するまでは、iSCSI ボリュームをマウントしようとししないでください。下の例は、`/etc/fstab` ファイルで `_netdev` mount オプションを使用して iSCSI マウントに対するこのネットワークの依存性を作成する方法を示しています。

```
# cat /etc/fstab
[snip]
/dev/mapper/VolGroup00-home /home          xfs     defaults          1 2
/dev/mapper/VolGroup00-swap swap      swap     defaults          0 0
LABEL=iSCSI_Vol      /mnt/iSCSI_Vol  ext4    defaults,_netdev  0 0
[snip]
```



## 3.3 シリアル接続 SCSI

Dell SCv2000 シリーズ製品ラインでは、シリアル接続 SCSI (SAS) フロントエンド接続を提供し、SCOS 6.6.x の取り扱い開始と時期を合わせています。Dell SCv2000 シリーズは、ターゲットホストでの Dell 12Gbps SAS HBA の使用をサポートします。

Linux ホストへの SAS 接続では、`/etc/multipath.conf` ファイルの特定の設定スキーマが必要です。

### 3.3.1 SAS ドライバ

SAS ドライバは、特定の Linux カーネル (RHEL 6.5 以降および RHEL 7x) にプリロードされています。SAS ドライバの存在は、次のコマンドを使用して検証できます。ベストプラクティスとして、『Dell Storage Compatibility Matrix』 (Dell ストレージ互換性マトリックス) で、ドライバのバージョンを確認し、示されている最新のドライバを使用します。

```
# lsmod | grep sas
mpt3sas                188001  4
scsi_transport_sas     35588  1 mpt3sas
raid_class             4388  1 mpt3sas
megaraid_sas           96205  2

# modinfo mpt3sas | grep description
description:    LSI MPT Fusion SAS 3.0 Device Driver
```

### 3.3.2 SAS `/etc/multipath.conf`

次のデバイススキーマを `/etc/multipath.conf` ファイルに追加し、SAS 接続の Linux ホストにのみ使用します。このスキーマは、`vendor="COMPELNT"` および `product="Compellent Vol"` により識別されるすべてのデバイスの設定パラメーターを定義します。このスキーマは通常、デフォルトスキーマの後および `blacklist_exceptions` スキーマの前に追加されます。

```
devices {

    device {
        vendor COMPELNT
        product "Compellent Vol"
        path_checker tur
        prio alua
        path_selector "service-time 0"
        path_grouping_policy group_by_prio
        no_path_retry 24
        hardware_handler "1 alua"
        failback immediate
        rr_weight priorities
    }

}
```



### 3.3.3 FC/iSCSI および SAS

同じ Linux ホストでのマージされた FC/iSCSI と SAS 接続環境の使用は、検証もサポートもされません。FC/iSCSI 接続を使用する他の Linux ホストから独立している SAS 接続 Linux ホストを維持します。

SAS 固有の multipath.conf 設定スキーマは、FC/iSCSI ベースの /etc/multipath.conf ファイルとマージできます。このマージされた設定は、デルではサポートされないことに注意してください。

### 3.3.4 Linux で SAS デバイスを識別する

このサンプルスクリプトは、Linux /sys ファイルシステムを解析し、見つけ、host\_sas\_address として識別されたファイルの内容を表示します。このファイルの内容は、インストールされた SAS HBA のデバイス名を表します。このスクリプトは、いかなる種類の表明された保証または一切のサポートなく、現状で提供されます。

以下の例では、2 つの SAS HBA カードはそれぞれに応じて表示されるデバイス名で識別されます。このスクリプトは、RHEL 6x と RHEL 7x Linux 両方のホストに対して機能します。

```
# for i in `find /sys/devices -name host_sas_address`; do echo "=== $i"; cat $i;
echo; done
===
/sys/devices/pci0000:40/0000:40:01.0/0000:41:00.0/host1/scsi_host/host1/host_sas
_address
0x544a842007902800

===
/sys/devices/pci0000:40/0000:40:03.0/0000:42:00.0/host2/scsi_host/host2/host_sas
_address
0x544a84200792ce00
```

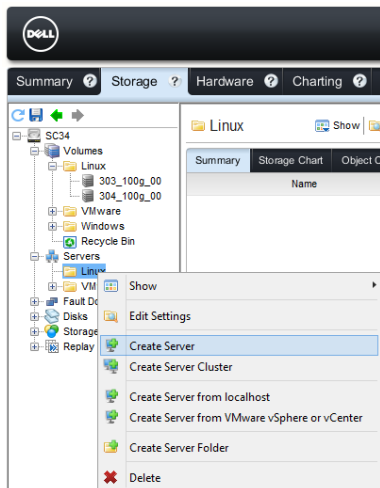


### 3.3.5 Dell SCv2000 で SAS デバイスを識別する

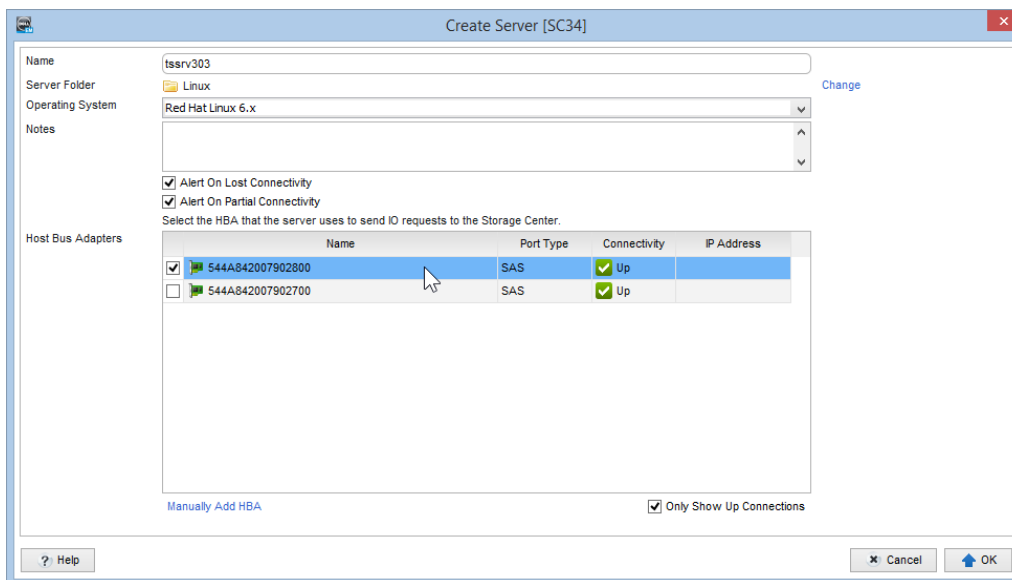
SAS デバイスは、複数の SAS ワールドワイドポート名（WWPN）ではなく、単一のデバイス名により Dell SCv2000 に公開されます。

次の一連のスクリーンショットは、Dell SCv2000 ストレージアレイで新しいサーバオブジェクトを作成するための手順の概要を示しています。

1. **Servers**（サーバ）フォルダツリーで右クリックし、**Create Server**（サーバの作成）をクリックします。



2. 新規サーバオブジェクトに名前を付け、適切なオペレーティングシステム（Red Hat Linux 6.x または 7.x）、Linux ホストに対応する識別された SAS デバイス名を選択し、**OK** をクリックします。



3. 新規サーバオブジェクト（tssrv303）は、サーバフォルダツリーで作成され、Connectivity（接続性） ペインには Linux ホストが Storage Center アレイの上部および下部コントローラ両方への接続を確立したことが表示されます。

The screenshot shows the Dell Storage Center web interface. The left sidebar displays a tree view with 'Servers' expanded, showing 'tssrv303'. The main panel shows the configuration for 'tssrv303'. The 'Index' tab is selected, showing details like 'Index: 21', 'Connectivity: Up', 'Type: Physical', 'Port Type: SAS', 'Operating System: Red Hat Linux 6.x', and 'Server Folder: Linux'. Below this, the 'Server HBAs' table shows one HBA with WWN 544A842007902800, Port Type SAS, and Connectivity Up. The 'Mappings' tab is also visible, showing the mapping for the HBA to the SAS Domain 1.

Name	Port Type	Connectivity
544A842007902800	SAS	Up

Controller	Controller Port	Path Status	Slot	Slot Port	Fault Domains
Bottom Controller	FE 5000D31000FEB315 (1-1)	Up	1	1	
Top Controller	FE 5000D31000FEB308 (1-1)	Up	1	1	

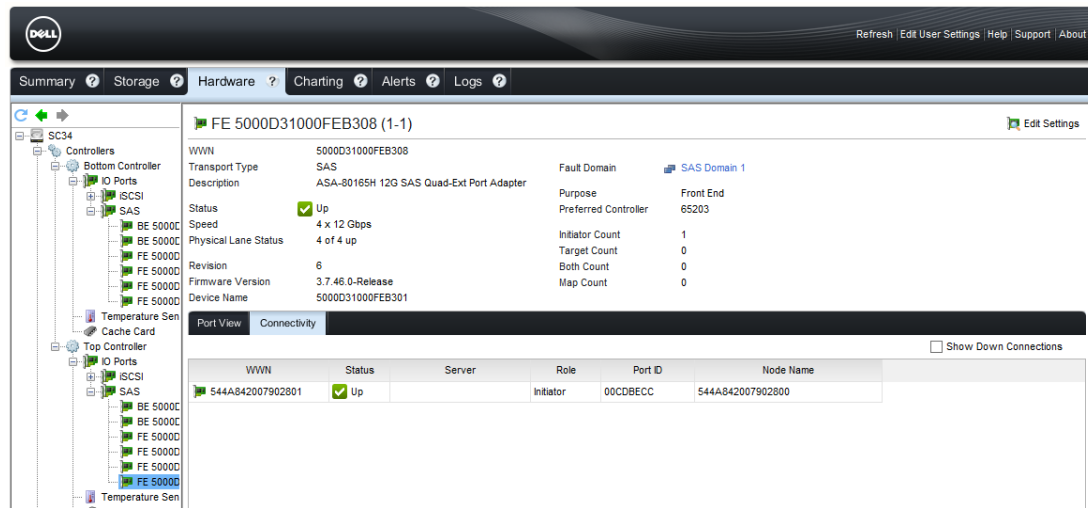
4. Hardware（ハードウェア） ペインで、\*FEB315 で終わる SAS コントローラを選択し、Connectivity（接続性） ペインに \*2800 で終わる実際の SAS WWPN を表示します。

The screenshot shows the Dell Storage Center web interface with the 'Hardware' tab selected. The left sidebar shows the 'Controllers' tree with 'FE 5000D31000FEB315 (1-1)' selected. The main panel shows the configuration for this controller. The 'Port View' tab is selected, showing a table of ports. The table shows the WWN 544A842007902800, Status Up, Server tssrv303, Role Initiator, Port ID 001699BB, and Node Name 544A842007902800.

WWN	Status	Server	Role	Port ID	Node Name
544A842007902800	Up	tssrv303	Initiator	001699BB	544A842007902800



5. Hardware（ハードウェア） ペインで、\*FEB308 で終わる SAS コントローラを選択し、Connectivity（接続性） ペインに \*2801 で終わる実際の SAS WWPN を表示します。



### 3.3.6 設定されたマルチパス

適切に設定された SAS ボリュームは、次の “multipath -ll” 出力を返します。この Storage Center ボリュームは、マルチパス ALUA 対応ボリュームとして検出され、各パスで I/O が可能です。prio=50 で表されたパス（アクティブな / 最適化されたパス）は、すべてのアクティブな I/O 要求に使用されます。prio=1 で表されたパス（スタンバイパス）は、高可用性、冗長パスでアクティブな / 最適化されたパスが使用不可になった場合に使用されます。

```
# multipath -ll
Compelnt_0016 (36000d31000feb30000000000000000016) dm-3 COMPELNT,Compellent Vol
size=100G features='1 queue_if_no_path' hwhandler='1 alua' wp=rw
|+-+ policy='service-time 0' prio=50 status=active
|  `-- 1:0:0:1 sdb 8:16 active ready running
`+-+ policy='service-time 0' prio=1 status=enabled
   `-- 1:0:1:1 sdc 8:32 active ready running
Compelnt_001a (36000d31000feb3000000000000000001a) dm-4 COMPELNT,Compellent Vol
size=100G features='1 queue_if_no_path' hwhandler='1 alua' wp=rw
|+-+ policy='service-time 0' prio=50 status=active
|  `-- 1:0:1:2 sdd 8:48 active ready running
`+-+ policy='service-time 0' prio=1 status=enabled
   `-- 1:0:2:2 sde 8:64 active ready running
```



### 3.3.7 SAS キュー深度

現在サポートされている SAS HBA のデフォルトは、ボリュームごとに 254 のキュー深度に設定されます。この値は、アプリケーション層の設定要件によって具体的に指定されない限り、出荷時のデフォルト値のままにする必要があります。

```
# lsscsi -L | egrep 'COMPELNT|depth'
[1:0:0:1]    disk      COMPELNT Compellent Vol    0606   /dev/sdb
             queue_depth=254
[1:0:1:1]    disk      COMPELNT Compellent Vol    0606   /dev/sdc
             queue_depth=254
```

### 3.3.8 SAS から起動

SAS 機能からの起動は、Linux 上の Dell 12Gbps SAS HBA での使用では、検証もサポートもされません。

## 3.4 キュー深度の管理

Fibre Channel HBA のキュー深度を設定するには 2 つの場所があります。設定することができる最初の場所は、HBA の BIOS 設定メニュー内です。この設定メニューは、通常起動時または HBA ベンダーが提供するツールを使用してアクセスします。キュー深度も、オペレーティングシステム内の Linux modprobe 機能により管理される HBA 設定ファイルを使用して設定できます。キュー深度の値は、これらの 2 つの場所で異なり、2 つの値のうち小さい方が優先されます。BIOS を高い値（8Gb QLogic HBA の場合、この値は Execution Throttle（実行スロットル）と呼ばれ、デフォルト値は 65535 です）に設定し、その後でそれぞれのオペレーティングシステム設定ファイルを使用して HBA キュー深度を管理することをお勧めします。キュー深度の値 128 が、推奨する開始点です。個々の環境のニーズを満たすために、値を適宜評価、テスト、調整します。

- Qlogic の場合：

```
options qla2xxx ql2xmaxqdepth=<value>
```

- Emulex の場合：

```
options lpfc lpfc_hba_queue_depth=<value> lpfc_lun_queue_depth=<value>
```

これらのファイル内の値は、必要に応じて他の追加のキー値ペアとスタッキングして、完全な構成設定を定義できます。



## 3.5 /etc/multipath.conf および Storage Center デバイスの定義

Storage Center デバイス定義は、カーネルに組み込まれています。以下のように、ネイティブ構文制約内の /etc/multipath.conf を使用します。この例では、他のすべてのベンダーのデバイスはブラックリストに載っており、Storage Center のデバイスは、一意の WWID 値により blacklist\_exceptions 句で識別されています。この /etc/multipath.conf 設定は FC と iSCSI の実装でのみ適用されます。SAS /etc/multipath.conf 設定は、[5.3.2 項](#)で説明されており、SAS 実装では代わりに使用する必要があります。

```
# cat /etc/multipath.conf
# multipath.conf written by anaconda

defaults {
    user_friendly_names yes
}

blacklist {
    devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
    devnode "^hd[a-z]"
    devnode "^dcssblk[0-9]*"
    device {
        vendor "DGC"
        product "LUNZ"
    }
    device {
        vendor "IBM"
        product "S/390.*"
    }
    # don't count normal SATA devices as multipaths
    device {
        vendor "ATA"
    }
    # don't count 3ware devices as multipaths
    device {
        vendor "3ware"
    }
    device {
        vendor "AMCC"
    }
    # nor highpoint devices
    device {
        vendor "HPT"
    }
    wwid "20080519"
    wwid "20080519"
    device {
        vendor iDRAC
        product Virtual_CD
    }
}
```





```

    }
    device {
        vendor PLDS
        product DVD-ROM_DS-8D3SH
    }
    #wwid "*"
}

blacklist_exceptions {
    device {
        vendor "COMPELNT"
        product "Compellent Vol"
    }
}

multipaths {
    multipath {
        alias vol_001
        wwid "36000d310000065000000000000000017f2"
        uid 0
        gid 0
        mode 0600
    }
    multipath {
        alias vol_002
        wwid "36000d310000065000000000000000017f3"
        uid 0
        gid 0
        mode 0600
    }
    multipath {
        alias vol_003
        wwid "36000d310000065000000000000000017f4"
        uid 0
        gid 0
        mode 0600
    }
}

```



### 3.5.1 マルチパスエイリアス

上記の `/etc/multipath.conf` ファイルの例では、Storage Center WWID 値はそれぞれ、マルチパス句内でエイリアスの値 (`vol_XXX`) によりさらに識別されます。Linux では、マルチパスのエイリアス / 名前は、ボリュームの機能をより正確に識別するために役立つだけでなく、これらのデバイス名が再起動および再構成の後でも持続性を維持します。つまり、マルチパスのエイリアス / 名前は一貫しており、スクリプティング、`mount` コマンド、`/etc/fstab` 構文などで使用しても安全だということです。

マルチパスのエイリアスが `/etc/multipath.conf` で定義されていない場合、ボリュームのデフォルト設定はデフォルトの命名スキームの `/dev/mapper/mpathX` になり、この場合も再起動後の持続性を維持します。エイリアスの指定が推奨され、マルチパスデバイス名を詳細説明ラベル（たとえば、ビジネス機能または用途）と関連付けるときは、非常に便利です。

エイリアスを定義するか、または `/etc/multipath.conf` 内の構成を更新した後は、下に示されているように、`multipathd` デモンを再起動して変更を反映させる必要があります。

```
# systemctl restart multipathd.service
```

RHEL 7x では、コマンド構成を新しい `systemd` サービス管理フレームワークに渡すためにまだ従来のコマンド構文が使用されている可能性があります。

```
# service restart multipathd
```

## 3.6 マルチパス環境

Storage Center のフェイルオーバーイベント中、レガシーポートモードでの動作により、故障したコントローラ上のアクティブなポートの WWPN が、他にアクティブなコントローラ上の同じフォールトドメインの予約ポートへと移動する少し前に、ファブリックに表示されなくなります。仮想ポートモードのパスフェイルオーバーでは、アクティブな NPIV ポートの WWPN は同じフォールトドメインの別なアクティブ NPIV ポートに移動します。コントローラのフェイルオーバーの場合、WWPN は他のアクティブなコントローラにある同じフォールトドメインの別のアクティブな NPIV ポートに移動します。いずれのフェイルオーバーシナリオでも、Storage Center がこれらの変更をファブリック全体に伝播するのに 5 ～ 60 秒かかる場合があります。

マルチパス接続のシナリオで I/O の中断を軽減するため、最大 5 秒待機してからポートをダウン / 故障としてマークするように HBA レベルコードに指示することを推奨します。これにより、SAN ファブリック経由で I/O 要求を代替のまだアクティブな HBA パスに迅速に移動して、I/O またはその上にあるアプリケーションスタックへの I/O レイテンシを最小限に抑えます。すべてのパスがダウンしている場合は、1 つ、または複数のパスが回復されるまで、`dm-multipath` モジュールレイヤが I/O のキューイングを開始します。これにより、Storage Center は、失敗したポートの WWPN をアクティブなポートに移動し、SAN ファブリック全体に変更を伝播するのに十分な時間が得られます。マルチパス環境への設定構文は次の項で説明します。

SAS 接続環境では、両方のコントローラからのパスが、接続されたホスト（アクティブ / 最適化およびスタンバイ）に提示されますが、どの時点においてもすべてのアクティブな I/O にアクティブな / 最適化されたパスのみが使用されます。アクティブな / 最適化されたパスが使用不可能になると、Storage Center では、残りのスタンバイパスのいずれかがアクティブな / 最適化されたパスの役割を引き受けるかを判断し、新しいアクティブな / 最適化されたパスにアクティブな I/O を引き続きストリームします。



### 3.6.1 マルチパス /SCSI タイムアウトとキューイング

ほとんどの SAN ベースのボリューム（Storage Center ボリュームを含む）のコンパイル時のデフォルトには、すべてのパスが使用できなくなった場合に自動的にすべての I/O 要求をキューに登録し再試行するように dm-multipath に指示する、`features='1 queue_if_no_path'` オプションを備えています。これは、下記のサンプル出力から確認でき、そのデフォルトの状態のままにしておく必要があります。

```
# multipath -ll
[snip]
vol_001 (36000d310000065000000000000000017f2) dm-3 COMPELNT,Compellent Vol
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='service-time 0' prio=1 status=active
    |- 1:0:3:1 sdf 8:80 active ready running
    |- 1:0:5:1 sdk 8:160 active ready running
    |- 4:0:4:1 sdg 8:96 active ready running
    `-- 4:0:5:1 sdj 8:144 active ready running
[snip]
```

このデフォルトを使用する例外は、I/O がキューに登録されず、すべてのパスが使用できなくなった場合は、すぐに失敗とされるビジネス / アプリケーションのシナリオです。これらの状況では、I/O キューイングは `/etc/multipath.conf` で、グローバルデバイスレベルでまたは特定のデバイスに対して無効にするか、または実行時に特定のデバイスに対して適用できます。

グローバルデバイスレベルでは、`/etc/multipath.conf` に以下を挿入してから `multipathd` サービスを再起動します。

```
devices {
    device {
        vendor "COMPELNT"
        product "Compellent Vol"
        features 0
        no_path_retry fail
    }
}
```

これを実行時に特定のデバイスに実装するには、下に示すように、希望するマルチパスデバイスごとに次のコマンドを発行します。

```
# dmsetup message /dev/mapper/vol_001 0 "fail_if_no_path"
```

### 3.6.2 PortDown タイムアウト

これらの設定は、Linux システムがポートとの接続を失った後、接続を破棄するまで待機する時間を指定します。`qla2xxx.conf` または `lpfc.conf` ファイルで適宜以下を設定します。

- Qlogic の場合：  
`options qla2xxx qlport_down_retry=5`
- Emulex の場合：  
`options lpfc lpfc_devloss_tmo=5`



### 3.6.3 パラメーターの検証

設定の変更内容が反映されていることを確認するには、次のコマンドを適用します。

- Qlogic の場合：

```
# cat /sys/module/qla2xxx/parameters/qlport_down_retry
5
```
- Emulex の場合：

```
# cat /sys/class/scsi_host/hostX/lpfc_devloss_tmo
5
```

## 3.7 シングルパス環境

Storage Center のフェイルオーバーイベント中、レガシーポートモードでの動作により、故障したコントローラ上の 1 つのアクティブなポートの WWPN が、他にアクティブなコントローラ上の同じフォールトドメインの予約ポートへと移動する少し前に、ファブリックに表示されなくなります。仮想ポートモードのパスフェイルオーバーでは、1 つのアクティブな NPIV ポートの WWPN は同じフォールトドメインの別なアクティブ NPIV ポートに移動します。コントローラのフェイルオーバーの場合、WWPN は他のアクティブなコントローラにある同じフォールトドメインにある 1 つのアクティブな NPIV ポートに移動します。いずれのフェイルオーバーシナリオでも、Storage Center がこれらの変更をファブリック全体に伝播するのに 5 ～ 60 秒かかる場合があります。

シングルパス接続のシナリオで I/O の中断を軽減するため、最大 60 秒待機してからポートをダウン / 故障としてマークするように HBA レベルコードに指示します。これにより、Storage Center は、失敗したポートの WWPN をアクティブなポートに移動し、SAN ファブリック全体に変更を伝播するのに十分な時間が得られます。シングルパス環境の設定構文を以下に示します。

### 3.7.1 PortDown タイムアウト

これらの設定は、Linux システムがポートとの接続を失った後、接続を破棄するまで待機する時間を指定します。**qla2xxx.conf** または **lpfc.conf** ファイルで適宜以下を設定します。

- Qlogic の場合：

```
options qla2xxx qlport_down_retry=60
```
- Emulex の場合：

```
options lpfc lpfc_devloss_tmo=60
```

### 3.7.2 パラメーターの検証

設定の変更内容が反映されていることを確認するには、次のコマンドを適用します。

- Qlogic の場合：

```
# cat /sys/module/qla2xxx/parameters/qlport_down_retry
60
```
- Emulex の場合：

```
# cat /sys/class/scsi_host/hostX/lpfc_devloss_tmo
60
```



## 4 パフォーマンスの考慮事項

本項には、一般的なパフォーマンス調整オプションと Linux で利用可能な変数に関する一般的な情報とガイドランスが記載されています。この情報は、すべてを包含することを目的としておらず、使用されている値は最終なものとはみなされません。この項では Linux とストレージ Administrator が Linux インストールを微調整し、最適なパフォーマンスを達成するために使用できる開始点を示します。

次のパラメーターを変更する前に、現在の環境の作業負荷をよく理解します。これを実行するには、環境をサポートする日々の経験に基づく、システムまたはストレージ Administrator の認識の評価を含め、多数の方法があります。Dell Performance Analysis Collection Kit (DPACK) は、[DPACK\\_Support@Dell.com](mailto:DPACK_Support@Dell.com) にメールを送信して入手できる無料のツールキットです。

Linux でのパフォーマンス調整について留意すべきいくつかの一般的なガイドラインは次のとおりです。

- パフォーマンス調整は、それが科学であるのと同様にアートでもあります。パフォーマンスに影響する変数（特に I/O）が多数あるため、すべての環境に対する特定の値は推奨できません。少数の変数から開始し、システムを調整しながら変数またはレイヤを追加します。たとえば、1 つのパスから開始し、調整してからマルチパスを追加します。
- 1 つずつ変更し、テストして測定し、パフォーマンスの監視ツールを使用してパフォーマンスへの影響を評価してから以降の変更を行います。
- ベストプラクティスとして、必要に応じて変更を既知の状態に戻すことができるように必ず元の設定を記録します。
- まず、非実稼働環境でシステム調整原則（フェイルオーバーなど）を適用し（可能な場合）、実稼働環境にこれらの変更を反映させる前に、できるだけ多くの環境条件により変更を検証します。
- 現在の構成設定でパフォーマンスのニーズが満たされている場合、システムを不安定にする変更を導入することを防ぐため、その設定をそのままにしておくことがベストプラクティスです。
- パフォーマンスに最も効果的な影響を及ぼすように調整可能な設定を効率的に標的とするために、ブロックとファイルレベルデータの違いを理解する必要があります。Storage Center アレイはブロックベースのストレージデバイスですが、iSCSI 転送メカニズムのサポートにより通常ネットワークやファイルレベルの調整に関連付けられているパフォーマンスの考慮事項が導入されています。
- 変更がパフォーマンスに影響しているかどうかを検証するとき、Dell Storage Manager のグラフ作成機能を利用してパフォーマンスを追跡します。また、I/O パフォーマンスに対して最も大きい（好または悪）影響がある変数がどれかをより正確に追跡するために、繰り返しの間で行う変更は 1 つのみにします。



## 4.1 プロファイルの調整

RHEL 7x では、Red Hat Linux ホストの調整で Administrator およびストレージの専門家をサポートする新しいツールセットが導入されています。これは、新しい 2 つのコマンド (**tuned** および **tuned-adm**) で確認でき、またこれらのコマンドは、事前定義のパフォーマンス調整済みのプロファイルを管理します。次の出力は、アクティブな調整プロファイル (デフォルト)、および適用できる代替調整プロファイルのリストを示しています。現在、どの Storage Center 実装でも、デフォルトの調整プロファイル **throughput-performance** の使用が推奨されます。各調整プロファイルおよびその長所の説明はこの文書の対象外ですが、このトピックの詳細な説明については、以下のサイトを参照してください。

[https://access.redhat.com/site/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/Performance\\_Tuning\\_Guide/sect-Red\\_Hat\\_Enterprise\\_Linux-Performance\\_Tuning\\_Guide-Performance\\_Monitoring\\_Tools-tuned\\_and\\_tuned\\_adm.html](https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Performance_Tuning_Guide/sect-Red_Hat_Enterprise_Linux-Performance_Tuning_Guide-Performance_Monitoring_Tools-tuned_and_tuned_adm.html)

```
# tuned-adm active
Current active profile: throughput-performance

# tuned-adm list
Available profiles:
- balanced
- desktop
- latency-performance
- network-latency
- network-throughput
- powersave
- sap
- throughput-performance
- virtual-guest
- virtual-host
```

## 4.2 複数のボリューム


Storage Center のボリュームは、一度に 1 つのみがアクティブになり、Storage Center コントローラに提示されます。この考慮点に留意し、Linux ホストで複数のボリュームを使用する場合は、デュアル I/O 処理および負荷に対して両方のコントローラを最も効果的に活用するために、両方の Storage Center コントローラで Linux ホストに交互にボリュームを提示します。Linux ホストから、これらのボリュームを取り外された LVM ボリュームグループに取り込むことで、Storage Center アレイへのすべてのパスでパラレル I/O のスレッド数が倍増します。個々のニーズ、また多くの場合、多様なビジネス環境や使用事例に合わせてこの方法を評価、適合およびテストします。


SAS 接続環境では、両方のコントローラからのパスが、接続されたホスト (アクティブ / 最適化およびスタンバイ) に提示されますが、どの時点においてもすべてのアクティブな I/O にアクティブな / 最適化されたパスのみが使用されます。上で説明した方法を適用して、各ボリュームで異なるアクティブな / 最適化されたパスを使用することにより、複数のコントローラで I/O スレディングとスループットをさらに向上させることができます。これは、ボリュームをサーバオブジェクトにマッピングするとき、これらのボリュームを異なるコントローラに明示的に固定することによって実行されます。以下に示すように、この機能はマッピングダイアログボックスの Advanced Options (詳細オプション) を介してアクセスできます。



1. **Advanced Options**（詳細オプション）リンクをクリックします。

The following volume and server will be mapped

Volume  304\_100g\_01


Server  tssrv304


[Advanced Options](#)

2. **Restrict Mapping Paths**（マッピングパスの制限）のチェックを外し、サーバオブジェクトをマッピングするためにボリュームを固定するコントローラを選択します。

Map Volume to Server

The following volume and server will be mapped

Volume  304\_100g\_01

Server  tssrv304

**Select LUN**

☒ Use next available LUN

☐ Map volume using LUN 0 (this is usually reserved for boot volumes)

**Restrict Mapping Paths**

☐ Allow the Storage Center to automatically determine the best Controller to activate Volume on

Activate Volume on Controller:

**Configure Multipathing**

Maximum number of paths per Server

**Configure Volume Use**

☐ The volume should be presented as read-only to the server

Top Controller  
Top Controller  
Bottom Controller  
Unlimited

? Help Back Finish



## 4.3 HBA キュー深度

キュー深度は、保留中の I/O 要求の数を指します。この値を変更すると、特定の作業負荷での I/O パフォーマンスの向上につながります。一般に、キュー深度を増加するとスループットが向上しますが、この値を増やすと、I/O レイテンシが高くなるため、考慮する必要があります。I/O の大部分が小さい読み取りと書き込みである環境など、この値を増やすと、別のアプリケーションで利点がある場合があります。IOPS 要件が低いけれども、高いスループットを必要とする環境では、最適なレベルのパフォーマンスが得られるまでこのキュー深度の設定を低くすることによってこれが達成されます。

この値は、HBA ファームウェアまたは HBA 用の Linux カーネルモジュールで変更することができます。これらの 2 つの設定値が異なる場合、低い値が優先されることに注意してください。検討すべき有効な方法は、HBA ファームウェアを許容される最大の数字に設定し、その後で Linux カーネルモジュール内からこの値を下方向に調整することです。

使用している特定ベンダーの HBA に対してこの値を変更するための詳細については、項 3「[サーバ設定](#)」を参照してください。

## 4.4 SCSI UNMAP/TRIM

ファイルシステムの使用事例での SCSI UNMAP/TRIM の応用と使用は、ビジネスエンタープライズでのストレージおよびコスト管理に非常に効果的です。ただし、この機能の実装方法に関する考慮も必要です。

discard mount パラメーターを使用すると、ファイルシステムは Storage Center アレイに対して、リアルタイムでオンザフライの SCSI UNMAP コマンドを実行できます。高 I/O ベースのアプリケーションランドスケープでは、これにより、I/O 制御トラフィックのレベルが不必要に高くなり、また、CPU の負荷が増加して I/O のレイテンシが増え、結果としてビジネスクリティカルな I/O に影響を与える可能性があります。

この実装方法の代替となるのは、fstrim コマンドの使用です。fstrim コマンドは、util-linux パッケージの一部であり、マウントされたファイルシステムから使用されたブロックを破棄する 1 回限りの要求ができます。このコマンドは、スクリプト化し（下にサンプルスクリプトを表示）、スケジュールされた cron ジョブに注入し、ビジネスクリティカルな機能への影響が小さい時間帯に、マウントされた一連のファイルシステムに一括して適用できます。

```
#!/bin/bash

FSTrim=/usr/bin/fstrim

MntPoints="u01 u02 u03 u04 u05"

for i in ${MntPoints}
do
    echo "INFO: Applying ${FSTrim} to mount point ${i}"
    ${FSTrim} -v /${i}
done
```





## 4.5 Linux SCSI デバイスキュー変数

パフォーマンスを向上するために複数の Linux SCSI デバイスキュー設定を調整することができます。最も一般的なものは、4.5.1 ~ 4.5.3 の項に記載されていて、I/O に関連した各パラメーターの機能を簡単に説明しています。これらの値は、/sys/block/dm-X/queue ディレクトリ（マルチパスデバイスの場合）、および /sys/block/sdX/queue ディレクトリ（ブロックデバイスの場合）にあり、各デバイスでパフォーマンスを変更する対象のボリュームについて変更する必要があります。

### 4.5.1 カーネル I/O スケジューラ

/sys/block/dm-X/queue/scheduler パラメーターとその内容は SCSI (SD) デバイス用の Linux カーネルによって使用されている I/O スケジューラを定義します。一部のアプリケーションベンダー（Oracle など）は、アプリケーションプラットフォームでの最適なパフォーマンスを達成するために、使用する I/O スケジューラを具体的に推奨しています。RHEL 7x のデフォルトでは、この変数はファイル内の [] ブラケットで示される **deadline** に設定されています。**deadline** スケジューラは、sysfs 調整可能サブパラメーター（fifo\_batch、read\_expire、write\_expire など）を使用してさらに調整できます。詳細については、次の Red Hat サイトを参照してください。

[https://access.redhat.com/site/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/Performance\\_Tuning\\_Guide/chap-Red\\_Hat\\_Enterprise\\_Linux-Performance\\_Tuning\\_Guide-Storage\\_and\\_File\\_Systems.html#sect-Red\\_Hat\\_Enterprise\\_Linux-Performance\\_Tuning\\_Guide-Considerations-IO\\_Schedulers](https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Performance_Tuning_Guide/chap-Red_Hat_Enterprise_Linux-Performance_Tuning_Guide-Storage_and_File_Systems.html#sect-Red_Hat_Enterprise_Linux-Performance_Tuning_Guide-Considerations-IO_Schedulers)

```
# cat /sys/block/dm-X/queue/scheduler
noop [deadline] cfq
```

このパラメーターは、次のコマンドを実行して動的に変更できます。ただし、すべての Storage Center の実装で、スケジューラを **deadline** のままにしておくことを推奨します。また、前述のように、スケジューラの値も RHEL 7x 調整プロファイルの使用により動的に調整できます。

```
# echo cfq > /sys/block/dm-X/queue/scheduler
# cat /sys/block/dm-X/queue/scheduler noop deadline [cfq]
```

スケジューラが、変更された場合、オペレーティングシステムの現在実行中のインスタンスにのみ適用され、それが適用される特定の /dev/sdX デバイスにのみ適用されます。スクリプトを追加でを使用して、必要なすべての SCSI (SD) デバイスで（デバイスごとに）起動時にこの変更を持続させることができます。また、下に示されているように、elevator= キー値オプションを /boot/grub2/grub.cfg 起動設定ファイルの末尾に追加することにより、起動時にこの変更をシステム全体に適用することもできます。

```
linux16 /vmlinuz-0-rescue-c417dfc159fc4450ac28137506041 root=UUID=35bea1c0-
ce32-42a4-8e36-72fd5e77471d ro rd.lvm.lv=VolGroup00/root
vconsole.font=latarcyrheb-sun16 rd.lvm.lv=VolGroup00/swap crashkernel=auto
vconsole.keymap=us rhgb quiet elevator=deadline
```



## 4.5.2 read\_ahead\_kb

このパラメーターは、ブロックデバイスから連続的に読み取りが行われていること、および何キロバイトの I/O を Linux カーネルが読み取るかを定義することを検出する場合に使用されます。この値を修正すると、大量かつ連続的な読み取りワークロードでパフォーマンスに大きな影響がある可能性があります。このパラメーターの RHEL 7x のデフォルト値は 4096 です。このデフォルトの状態が良好な開始点です。

## 4.5.3 nr\_requests

nr\_requests 値は、Linux カーネルによって、要求キューの深度を設定するために使用され、多くの場合、HBA のキュー深度設定の変更と連動して使用されます。RHEL 7x のデフォルト値は 128 です。この値を増やすと、I/O サブシステムが大きいしきい値に設定され、そこで引き続き要求をスケジューリングします。これにより、I/O サブシステムは 1 つの方向に長く移動し続け、ディスク I/O の処理が効率化します。まずこの値を 1024 に引き上げてから、評価して、パフォーマンスの結果に基づいて調整します。

## 4.6 iSCSI

iSCSI の調整は、イーサネットネットワークとブロックレベル調整の両方で行います。iSCSI で最適なパフォーマンスゲインが得られる設定を決定するために、共通のイーサネットカーネルの調整可能パラメーターを評価します。1 Gb/10 Gb イーサネットと併用する場合、ジャンボフレームを使用すると、iSCSI のパフォーマンスが向上します。Fibre Channel 同様、iSCSI の変更は個別に段階的に行い、全体的なパフォーマンスへの影響を十分に理解するために、複数のワークロードタイプに対して評価します。つまり、iSCSI の調整は、ネットワーク（イーサネット）の調整に加えてブロックレベルのサブシステムの調整に関する考慮事項があるため、長く時間がかかる場合があります。システムを効果的に調整するには、関係するさまざまな Linux サブシステムレイヤを確実に理解することが必要です。

パフォーマンスのために調整できるカーネルパラメーターは、/proc/sys/net/core および /proc/sys/net/ipv4 カーネルパラメーターにあります。最適な値が決定したら、/etc/sysctl.conf ファイルに恒久的に設定します。その他ほとんどの現代のオペレーティングシステムのプラットフォームと同様に、Linux は効率的に TCP バッファを自動調整できます。ただし、デフォルトでは、一部の設定は控えめに低くなっています。次のカーネルパラメーターを試すとネットワークパフォーマンスの向上につながると可能性があります。その後は、iSCSI のパフォーマンスが改善します。

- TCP の最大バッファサイズ：
  - net.core.rmem\_max
  - net.core.wmem\_max
- Linux 自動調整バッファの制限：
  - net.ipv4.tcp\_rmem
  - net.ipv4.tcp\_wmem
- net.ipv4.tcp\_window\_scaling
- net.ipv4.tcp\_timestamps
- net.ipv4.tcp\_sack



## 5 便利なツール

次のネイティブ Linux ツールを使用し、それぞれの Storage Center デバイスを識別して、ボリュームを関連させることができます。

### 5.1 lsscsi コマンド

lsscsi コマンドは、/proc および /sys 擬似ファイルシステムからの情報を人間が読める出力に解析するツールです。lsscsi ツールは、次のコマンドを使用してインストールされます。

```
# yum -y install lsscsi
```

lsscsi 出力を以下に示します。grep コマンドを使用して、Storage Center デバイスのみを表示しています。最初の行には、各ボリュームの [host:channel:target:lun] の指定先が表示されます。ホスト番号は、ボリュームがマップされているローカルの HBA hostX デバイスファイルに対応します。チャンネル番号は、SCSI バスのアドレスで常に 0 です。ターゲット番号は、Storage Center フロントエンドポート（ターゲット）に相关しています。Lun 番号は、マップされている Storage Center コントローラ上のボリュームの LUN ID を表します。

```
# lsscsi | grep COMPELNT | sort -k7
[snip]
[1:0:0:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdb
[1:0:2:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdc
[1:0:3:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdd
[1:0:3:2]      disk      COMPELNT  Compellent Vol   0605   /dev/sde
[1:0:5:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdf
[1:0:5:2]      disk      COMPELNT  Compellent Vol   0605   /dev/sdg
[4:0:0:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdh
[4:0:2:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdi
[4:0:4:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdj
[4:0:4:2]      disk      COMPELNT  Compellent Vol   0605   /dev/sdk
[4:0:5:1]      disk      COMPELNT  Compellent Vol   0605   /dev/sdl
[4:0:5:2]      disk      COMPELNT  Compellent Vol   0605   /dev/sdm
[snip]
```

### 5.2 scsi\_id コマンド

scsi\_id コマンド (/usr/lib/udev フォルダにあります) を使用して、ボリュームの WWID を報告できます。この WWID を使用して、ボリュームをそれぞれの Storage Center デバイスに関連させることができます。

次のスクリプトは、いかなる種類の表明された保証またはサポートも一切なく、現状で提供されます。このスクリプトは scsi\_id コマンドを適用し、Linux デバイス名 (/dev/sdX) をそれぞれの WWID 値に関連させます。

```
#!/bin/bash
```

```
SCSIID=/usr/lib/udev/scsi_id
```



```

#OSMajor=`cat ${ReleaseFile} | awk '{print $7}' | cut -d.-f1`
OSMajor=`uname -r | awk -F.''{print $4}'`

echo "INFO: OS Major Rev.${OSMajor} detected!"

if [ "${OSMajor}" = "el7" -o "${OSMajor}" = "el6" ]; then
{
    for i in `cat /proc/partitions | awk '{print $4}' | grep sd`
    do
        echo "Device: $i WWID: `${SCSIID} --page=0x83 --whitelisted --
device=/dev/$i`"
    done
    } | sort -k4 $1
elif [ "${OSMajor}" = "el5" ]; then
{
    for i in `cat /proc/partitions | awk '{print $4}' | grep sd`
    do
        echo "Device: $i WWID: `${SCSIID} -g -u -s /block/$i`"
    done
    } | sort -k4 $1
else
    echo "WARN: OSMajor parameter of unknown value, Exiting"
    exit 1
fi

```

このスクリプトの出力例を下に示します。

```

# ./get_WWID.sh
INFO: OS Major Rev.el7 detected!
[snip]
Device: sdd WWID: 36000d310000065000000000000000017f2
Device: sdf WWID: 36000d310000065000000000000000017f2
Device: sdj WWID: 36000d310000065000000000000000017f2
Device: sdl WWID: 36000d310000065000000000000000017f2
[snip]

```



これらの WWID 値は、Storage Center デバイスのシリアルナンバーの値と関連します。

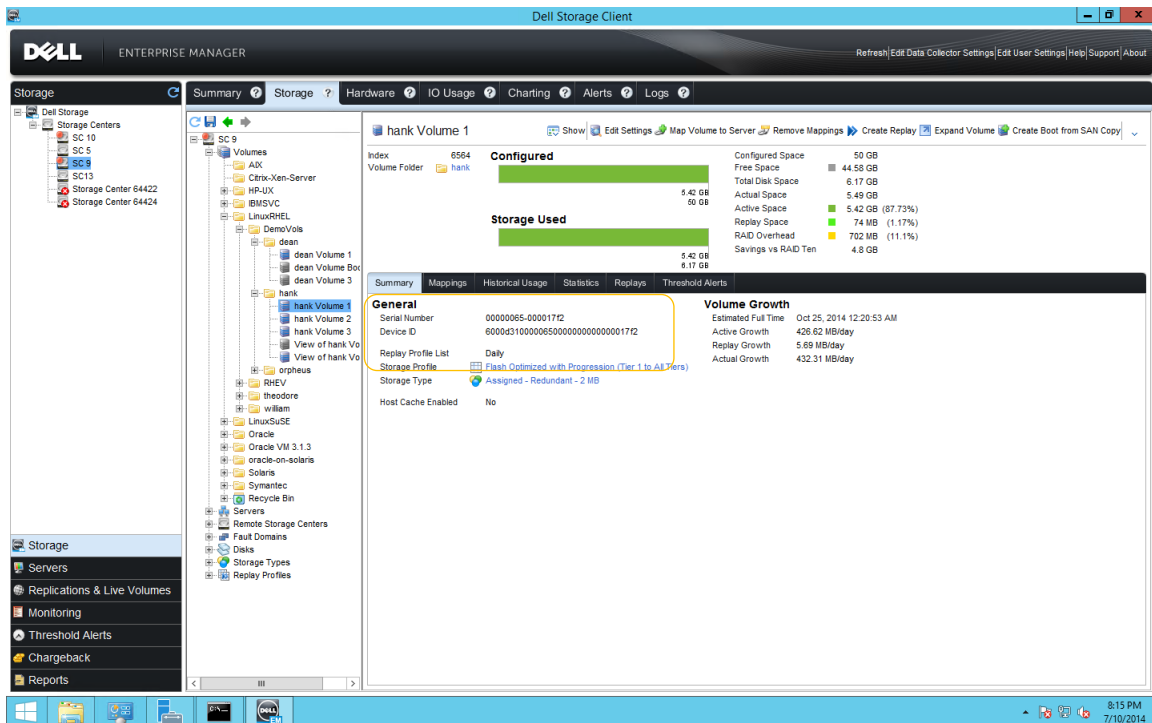


Figure 1 Serial Number (シリアルナンバー) フィールド

## 5.3 dmsetup コマンド

次の dmsetup コマンドは、デバイスマッパーデバイスが管理する論理ボリュームのメタデータを表示、管理および変更するために使用できます。下記の例では、dmsetup コマンドは、Storage Center ボリュームの値を表示するために使用されています。このボリュームの名前と UUID 値です（UUID 値は "mpath-" プレフィックスをボリュームのシリアルナンバーの値に連結したもの）。

```
# dmsetup info /dev/mapper/vol_001
Name:                vol_001
State:               ACTIVE
Read Ahead:          256
Tables present:      LIVE
Open count:           0
Event number:         0
Major, minor:        253, 4
Number of targets:    1
UUID: mpath-36000d310000065000000000000000017f2
```



## 5.4 dmesg コマンド

dmesg コマンドは、最近検出されたボリュームに割り当てられているデバイス名の検出に役立ちます。以下の出力は、新しい Storage Center ボリュームの検出と /dev/sdh デバイスファイルへの割り当てを示しています。

```
# dmesg
[snip]
[1203830.267568] scsi 4:0:0:1: Direct-Access      COMPELNT Compellent Vol   0605
PQ: 0 ANSI: 5
[1203830.267996] sd 4:0:0:1: Attached scsi generic sg8 type 0
[1203830.268089] sd 4:0:0:1: [sdh] 20971520 512-byte logical blocks: (10.7
GB/10.0 GiB)
[1203830.268093] sd 4:0:0:1: [sdh] 4096-byte physical blocks
[1203830.268847] sd 4:0:0:1: [sdh] Write Protect is off
[1203830.268858] sd 4:0:0:1: [sdh] Mode Sense: 8f 00 00 08
[1203830.269033] sd 4:0:0:1: [sdh] Write cache: disabled, read cache: enabled,
doesn't support DPO or FUA
[snip]
```

## 5.5 /proc/scsi/scsi ファイル

/proc/scsi/scsi ファイルには、Linux ホスト上のボリュームとターゲットに関するその他の詳細が記載されています。

```
# cat /proc/scsi/scsi
Attached devices:
[snip]
Host: scsi1 Channel: 00 Id: 00 Lun: 01
  Vendor: COMPELNT Model: Compellent Vol   Rev: 0605
  Type:   Direct-Access                    ANSI  SCSI revision: 05
Host: scsi1 Channel: 00 Id: 02 Lun: 01
  Vendor: COMPELNT Model: Compellent Vol   Rev: 0605
  Type:   Direct-Access                    ANSI  SCSI revision: 05
Host: scsi1 Channel: 00 Id: 03 Lun: 01
  Vendor: COMPELNT Model: Compellent Vol   Rev: 0605
  Type:   Direct-Access                    ANSI  SCSI revision: 05
Host: scsi1 Channel: 00 Id: 03 Lun: 02
  Vendor: COMPELNT Model: Compellent Vol   Rev: 0605
  Type:   Direct-Access                    ANSI  SCSI revision: 05
[snip]
```



## 6 Dell Compellent コマンドユーティリティ

Storage Center SAN は、Dell Compellent コマンドユーティリティ (CompCU) と呼ばれるリモートコマンドユーティリティを介して日常的な機能の多くを管理できます。これにより、Linux オペレーティングシステムと Storage Center 間の SAN のタスクのスクリプト作成と自動化を統合できます。CompCU は Java パッケージアプリケーションで、Linux ホスト上での Java のインストールが必要です。CompCU は、大幅に時間を節約でき、Storage Center のボリュームおよびリプレイの管理に一貫したフレームワークを提供する、一般的な管理タスクのスクリプト作成に使用できます。

CompCU では、ホストに適切な Java リリースをインストールしておく必要があります。詳細については、(付録 A にある) 『コマンドユーティリティユーザーガイド』を参照してください。CompCU.jar オブジェクトは、Dell Compellent サポートサイトからダウンロードできます。このツールを Linux ホスト上にインストールすると、新規または既存のユーザー管理スクリプトに組み込むことができるシェルスクリプトから Storage Center のタスクを実行できます。以下に、CompCU のいくつかの一般的な使用事例を示します。

- ボリュームの作成、サーバへのマッピング。
- リプレイの取得、リプレイのリカバリなど。

下の例は、CompCU の実用性を完全にカバーするものではありません。CompCU により自動化できるタスクの種類を紹介するために設計されています。

### 6.1 Java の検証、CompCU 機能の設定とテスト

最初に、Linux ホストに Java (RTE v1.6.x 以降) をインストールします。Java ランタイムは、すでにオペレーティングシステムとインストールされている場合があります。次のコマンドで確認できます。

```
# /usr/bin/java -version
java version "1.7.0_07"
Java(TM) SE Runtime Environment (build 1.7.0_07-b10)
Java HotSpot(TM) Server VM (build 23.3-b01, mixed mode)
```

CompCU パッケージを Dell Compellent サポートサイト

([http://kc.compellent.com/Published%20Documents/CU060401\\_001.zip](http://kc.compellent.com/Published%20Documents/CU060401_001.zip)) からダウンロードします。パッケージには、PDF のユーザーガイドと必要な CompCU.jar ファイルが含まれています。この CompCU.jar ファイルを論理ファイルシステムの場所に保存します。下のコマンド実行し、ヘルプおよび使用構文を表示して、CompCU が Java と動作していることを検証します。

```
# /usr/bin/java -jar ./CompCU.jar -h
Compellent Command Utility (CompCU) 6.4.1.1
```

```
usage: java -jar CompCU.jar [Options] "<Command>"
  -c <arg>                Run a single command (option must be within
                           quotes)
  -default                Saves host, user, and password to encrypted file
  -defaultname <arg>      File name to save default host, user, and password
                           encrypted file to
```



```

-file <arg>          Save output to a file
-h                  Show help message
-host <arg>         IP Address/Host Name of Storage Center Management
                    IP
-password <arg>     Password of user
-s <arg>            Run a script of commands
-user <arg>         User to log into Storage Center with
-verbose           Show debug output
-xmloutputfile <arg> File name to save the CompCU return code in xml
                    format.Default is cu_output.xml.

```

[snip]

CompCU 使用時のアクセスを容易にするためには、下に示すように、ツールを最初に -default スイッチで実行し、暗号化パスワードファイルを設定することができます。ローカルディレクトリに default.cli という名前のファイルが作成されます。このファイルは、明確化と用途に合わせて名前を変更できます。

```

# /usr/bin/java -jar ./CompCU.jar -default -host 172.16.2.109 -user Admin -
password XXX
Compellent Command Utility (CompCU) 6.4.1.1

```

```

=====
User Name:          Admin
Host/IP Address:    172.16.2.109
=====

```

```

Connecting to Storage Center: 172.16.2.109 with user: Admin
java.lang.IllegalStateException: TrustManagerFactoryImpl is not initialized
Saving CompCu Defaults to file [default.cli]...
The "default.cli" file may then be referenced in other commands to login to the
same Storage Center and perform tasks. A separate .cli file may be created for
each Storage Center under management with each containing the appropriate login
credentials for the respective Storage Center array. The example below
demonstrates a "volume show" command applied to the Storage Center located at IP
address 172.16.2.109.
# /usr/bin/java -jar ./CompCU.jar -defaultname default.cli -host 172.16.2.109 -
user Admin -password XXX -c "volume show"
Compellent Command Utility (CompCU) 6.4.1.1

```

```

=====
User Name:          Admin
Host/IP Address:    172.16.2.109
Single Command:     volume show
=====

```

```

Connecting to Storage Center: 172.16.2.109 with user: Admin
java.lang.IllegalStateException: TrustManagerFactoryImpl is not initialized
Running Command: volume show

```





Index	Name	Status	ConfigSize	ActiveSize
ReplaySize	Folder			
StorageProfile	DeviceI		D	
SerialNumber	ConfigSizeBlock	ActiveSizeBlock	ReplaySizeBlock	
MaxWriteSizeBlo	ReadCache	WriteCache		

283	Fal-asm-mirror-test-failgroup1	Up	100.00 GB	31.33 GB
0.00 KB	Oracle/11gR2/ASM-Mirror			
Recommended	6000d31		00000650000000000000000012f	
00000065-0000012f	209715200	65712128	0	0
Enabled	Enabled			
290	Fal-asm-mirror-sp-fg1	Up	100.00 GB	7.94 GB
0.00 KB	Oracle/11gR2/ASM-Mirror			
Recommended	6000d31		000006500000000000000000136	
00000065-00000136	209715200	16658432	0	0
Enabled	Enabled			
824	ibmsvc00-managed-mdisk1	Up	500.00 GB	98.02 GB
0.00 KB	IBMSVC			
Recommended	6000d31		00000650000000000000000034c	
00000065-0000034c	1048576000	205561856	0	0
Enabled	Enabled			

[snip]

Successfully finished running Compellent Command Utility (CompCU) application.

## 6.2 CompCU を使用して一般的なタスクを自動化する

本項では、Linux 上の CompCU で Storage Center タスクを管理するための使用事例を示しています。前述のように、これらの例は CompCU を使用して Linux シェルプロンプトから簡単に実行できるタスクの種類を示しています。これらは、システム Administrator がこの強力なツールセットを理解するための開始点としてのみ使用されます。

### 6.2.1 CompCU での単一ボリュームの作成

この例では、CompCU を使用して、Linux ホストから hank\_100g\_00 という名前の単一の 100 GB Storage Center ボリュームを作成し、Storage Center Linux フォルダに配置する方法を示しています。ボリュームは、Linux hank ホストにマップされます。

```
# /usr/bin/java -jar ./CompCU.jar -defaultname default.cli -host 172.16.2.109 -
user Admin -password XXX -c "volume create -name hank_100g_00 -folder Linux -
server hank -size 100g"
```



## 6.2.2 CompCU でのリプレイおよびリプレイビューの作成

この例では、単一の CompCU コマンドを使用して以下を示します。

- Storage Center 上の既存の hank\_100g\_00 ボリュームのリプレイ、hank\_100g\_00\_Replay を作成、
- 前記のリプレイからリプレイビュー、hank\_100g\_00\_View を作成、および
- リプレイビューを Linux ホスト、dean にマッピングする。

```
# /usr/bin/java -jar ./CompCU.jar -defaultname default.cli -host 172.16.2.109 -  
user Admin -password XXX -c "replay create -volume 'hank_100g_00' -name  
'hank_100g_00_Replay' -view 'hank_100g_00_RpView' -server 'dean'"
```

## 6.2.3 CompCU による複数のボリュームの迅速な展開

この最後の例では、CompCU を使用して、Storage Center から迅速にボリュームを作成、展開し、これらのボリュームを Linux ホスト、dean にマッピングする方法を示しています。

```
# for i in 0 1 2 3 4 5 6 7 8 9; do /usr/bin/java -jar ./CompCU.jar -defaultname  
default.cli -host 172.16.2.109 -user Admin -password XXX -c "volume create -name  
dean_10g_0${i} -folder Linux -server 'dean' -size 10g"; done
```



## A 追加リソース

『Dell Storage Center System Manager 6.5 Administrator's Guide』 (Storage Center System Manager 6.5 管理者ガイド)

<http://kc.compellent.com/Knowledge%20Center%20Documents/680-019-016.pdf>

『Dell Enterprise Manager 2014 R2 Administrator's Guide』 (Dell Enterprise Manager 2014 R2 管理者ガイド)

<http://kc.compellent.com/Knowledge%20Center%20Documents/680-017-021.pdf>

『SC8000 Connectivity Guide』 (SC8000 接続性ガイド)

<http://kc.compellent.com/Knowledge%20Center%20Documents/680-027-013.pdf>

『Dell Storage Center 6.0 Command Utility (CompCU) Reference Guide』

(Dell Storage Center 6.0 コマンドユーティリティ (CompCU) リファレンスガイド)

<http://kc.compellent.com/Knowledge%20Center%20Documents/680-018-007.pdf>

『Red Hat Enterprise Linux 7.0 Beta Release Notes』 (Red Hat Enterprise Linux 7.0 ベータ版リリースノート)

[https://access.redhat.com/site/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7-Beta/html/7.0\\_Release\\_Notes/index.html](https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/7-Beta/html/7.0_Release_Notes/index.html)

『Red Hat Enterprise Linux 7.0 Beta System Administrator's Guide』 (Red Hat Enterprise Linux 7.0 ベータ版システム管理者ガイド)

[https://access.redhat.com/site/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7-Beta/html/System\\_Administrators\\_Guide/index.html](https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/7-Beta/html/System_Administrators_Guide/index.html)

Red Hat Enterprise Linux Document Portal (Red Hat Enterprise Linux 文書ポータル)

[https://access.redhat.com/site/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/](https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/)

Red Hat Labs (ベータ版)

<https://access.redhat.com/labs/>



## B 設定の詳細

コンポーネント	説明
オペレーティングシステム	Red Hat Enterprise Linux 7.1 GA (Maipo)
ドライババージョン	ドライババージョン = 8.04.00.04.06.3-k-debug BIOS バージョン = 3.00
ファームウェアバージョン	ファームウェアバージョン = 5.06.05 (90d5)
アプリケーション	NA
ケーブル接続	QLE2562 8Gb デュアルポート PCIe FC HBA Dell 12Gbps SAS HBA
サーバ	Dell R630 x 2
保管時	Dell Storage Center SC 8000 SCOS 6.6.x 仮想ポートモード Dell Storage Center SC v2x00 SCOS 6.6.x 仮想ポートモード
スイッチ	Dell PowerConnect 5500 シリーズ

