



## Red Hat Virtualization 4.4

# スタンドアロンの Manager からセルフホストエンジンへの移行

Red Hat Virtualization Manager をスタンドアロンサーバーから自己管理の仮想マシンに移行する方法



# Red Hat Virtualization 4.4 スタンドアロンの Manager からセルフホストエンジンへの移行

---

Red Hat Virtualization Manager をスタンドアロンサーバーから自己管理の仮想マシンに移行する方法

Red Hat Virtualization Documentation Team

Red Hat Customer Content Services

rhev-docs@redhat.com

## 法律上の通知

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 概要

本書では、物理サーバー（または別の環境で実行されている仮想マシン）にインストールされた Manager から、同じ管理ホストで実行している Manager に、Red Hat Virtualization 環境を移行する方法を説明します。

---

## 目次

はじめに .....	3
スタンドアロンの MANAGER のアーキテクチャー .....	3
セルフホストエンジンのアーキテクチャー .....	3
第1章 移行の概要 .....	5
第2章 セルフホストエンジン用デプロイメントホストのインストール .....	6
2.1. RED HAT VIRTUALIZATION HOST のインストール .....	6
2.2. RED HAT ENTERPRISE LINUX ホストのインストール .....	8
第3章 RED HAT VIRTUALIZATION 用ストレージの準備 .....	12
3.1. NFS ストレージの準備 .....	13
3.2. ISCSI ストレージの準備 .....	14
3.3. FCP ストレージの準備 .....	15
3.4. RED HAT GLUSTER STORAGE の準備 .....	16
3.5. SAN ベンダーのマルチパス設定のカスタマイズ .....	16
3.6. 推奨される MULTIPATH.CONF 設定 .....	17
第4章 RED HAT VIRTUALIZATION MANAGER の更新 .....	19
第5章 元の MANAGER のバックアップ .....	21
第6章 新しいセルフホストエンジンでのバックアップの復元 .....	22
第7章 RED HAT VIRTUALIZATION MANAGER リポジトリの有効化 .....	26
第8章 既存のホストのセルフホストエンジンノードとしての再インストール .....	28
付録A カーネルモジュールが自動的に読み込まれないようにする手順 .....	29
A.1. モジュールの一時削除 .....	30
付録B 法的通知 .....	32



## はじめに

スタンドアロンの Red Hat Virtualization Manager をセルフホストエンジンに変換するには、スタンドアロンの Manager をバックアップし、新しいセルフホスト環境で復元します。

2つの環境タイプの違いは次のとおりです。

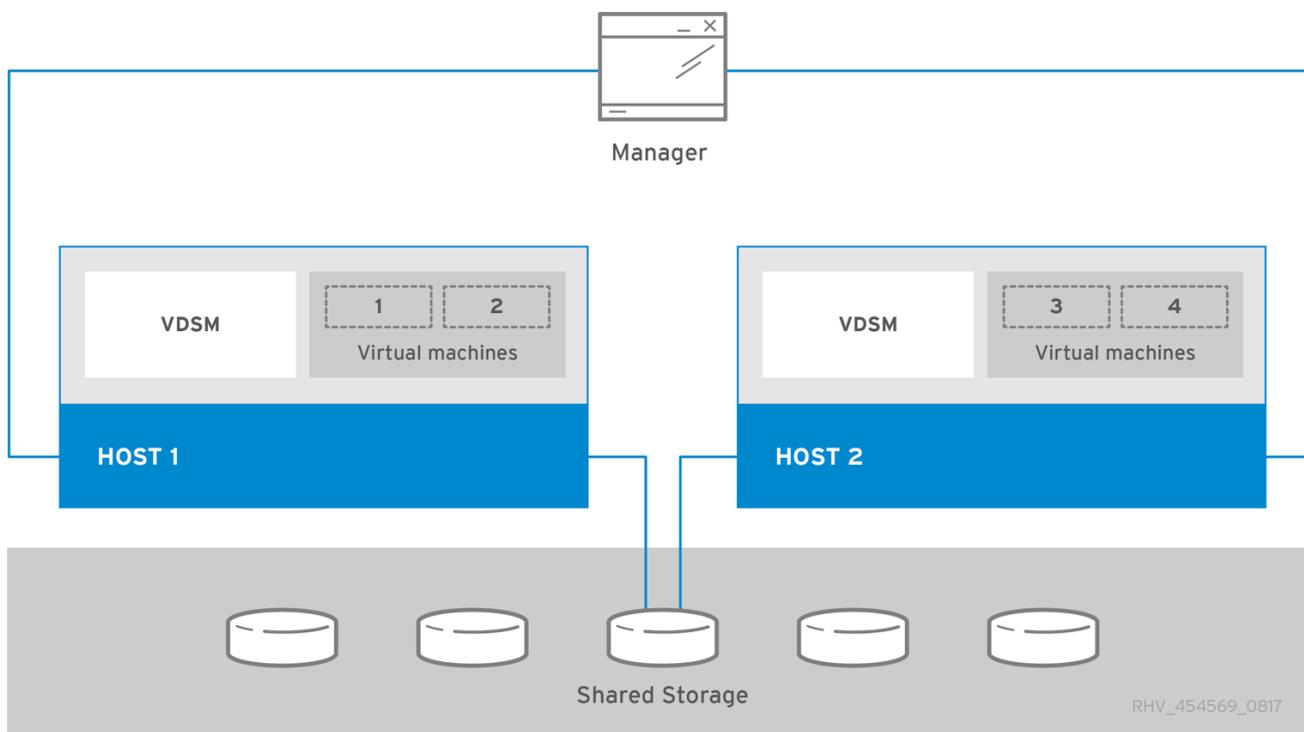
### スタンドアロンの MANAGER のアーキテクチャー

Red Hat Virtualization Manager は物理サーバーか、別の仮想環境でホストされている仮想マシン上で実行されます。スタンドアロンの Manager は、デプロイと管理が簡単ですが、追加の物理サーバーが1台必要となります。Manager は、Red Hat の High Availability Add-On などの別製品を使用して外部から管理した場合にのみ高可用性になります。

スタンドアロンの Manager 環境の最小セットアップには、以下が含まれます。

- Red Hat Virtualization Manager マシン1台。Manager は通常物理サーバーにデプロイされます。仮想マシン上にデプロイすることも可能ですが、その仮想マシンは別の環境でホストされていなければなりません。Manager は Red Hat Enterprise Linux 8 上で実行する必要があります。
- 仮想マシンの高可用性には、最小でホストが2台。Red Hat Enterprise Linux ホストまたは Red Hat Virtualization Host (RHVH) を使用することができます。VDSM (ホストエージェント) は全ホストで実行され、Red Hat Virtualization Manager との通信を円滑に行います。
- ストレージサービスを1つ。使用するストレージタイプに応じて、ローカルまたはリモートサーバーでホストすることができます。ストレージサービスは全ホストからアクセスできる必要があります。

図1 スタンドアロンの Manager の Red Hat Virtualization アーキテクチャー



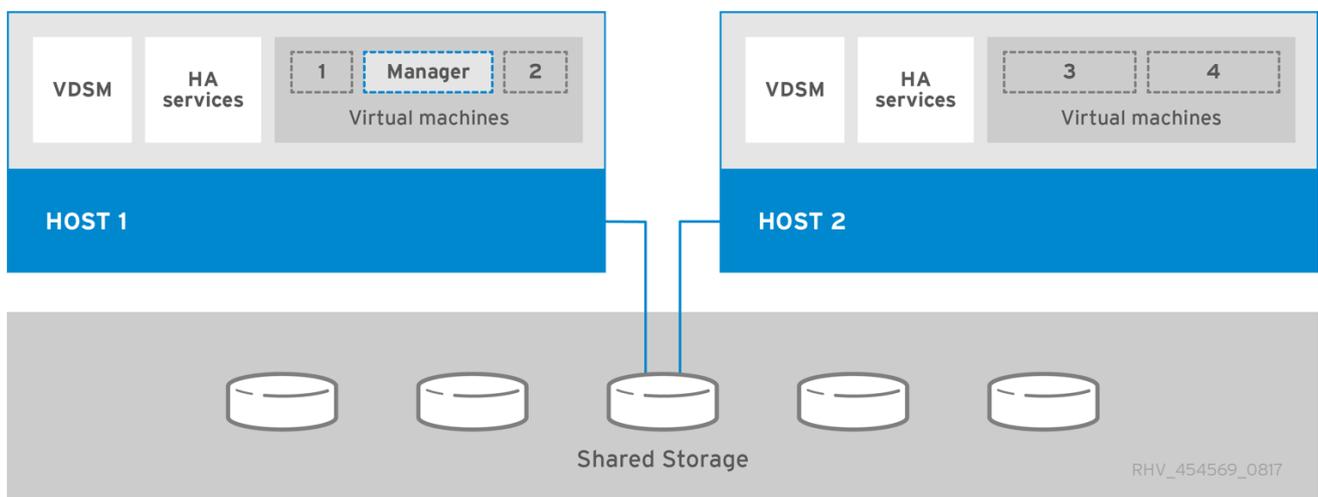
### セルフホストエンジンのアーキテクチャー

Red Hat Virtualization Manager は、管理している環境と同じ環境内のセルフホストエンジンノード (特化したホスト) で仮想マシンとして実行されます。セルフホストエンジン環境に必要な物理サーバーは1台少なくなります。デプロイと管理を行うための管理オーバーヘッドがより高くなります。Manager は、外部の HA 管理を使用せずに高可用性になります。

セルフホストエンジン環境の最小限のセットアップには、以下が含まれます。

- セルフホストエンジンノードでホストされている Red Hat Virtualization Manager 用仮想マシン 1 台。Red Hat Enterprise Linux 8 仮想マシンのインストールおよびその仮想マシンへの Manager のインストールを自動化するために、RHV-M Appliance が使用されます。
- 仮想マシンの高可用性には、最小でセルフホストエンジンノード 2 台。Red Hat Enterprise Linux ホストまたは Red Hat Virtualization Host (RHVH) を使用することができます。VDSM (ホストエージェント) は全ホストで実行され、Red Hat Virtualization Manager との通信を円滑に行います。HA サービスは、すべてのセルフホストエンジンノードで実行され、Manager 用仮想マシンの高可用性を管理します。
- ストレージサービスを 1 つ。使用するストレージタイプに応じて、ローカルまたはリモートサーバーでホストすることができます。ストレージサービスは全ホストからアクセス可能である必要があります。

図2 セルフホストエンジンの Red Hat Virtualization アーキテクチャー



## 第1章 移行の概要

セルフホストエンジンのデプロイメント時にバックアップファイルを指定すると、Manager のバックアップは専用のセルフホストエンジン用ストレージドメインを使用して、新しい仮想マシンに復元されます。新規ホストにデプロイすることを強く推奨します。デプロイメントに使用されたホストがバックアップ環境に存在している場合、新しい環境で競合を回避するために、復元されたデータベースから削除されます。新しいホストにデプロイする場合は、ホストに一意の名前を割り当てる必要があります。バックアップに含まれる既存ホストの名前を再利用すると、新しい環境で競合が発生する可能性があります。

Manager 用仮想マシンを高可用性として設定するには、少なくとも 2 台のセルフホストエンジンノードが必要です。新規ノードを追加するか、既存のホストを変換してください。

移行には主に、以下の手順が含まれます。

1. **セルフホストエンジンをデプロイする新しいホストをインストールします。** 以下のホストタイプのいずれかを使用することができます。
  - Red Hat Virtualization Host
  - Red Hat Enterprise Linux
2. **セルフホストエンジンのストレージドメイン用のストレージを準備します。** 以下のストレージタイプのいずれかを使用することができます。
  - NFS
  - iSCSI
  - ファイバーチャネル (FCP)
  - Red Hat Gluster Storage
3. 元の Manager をバックアップする前に最新のマイナーバージョンに更新します。
4. **engine-backup** ツールを使用して元の Manager をバックアップします。
5. 新しいセルフホストエンジンをデプロイし、バックアップを復元します。
6. 新しい Manager 用仮想マシンで Manager のリポジトリを有効にします。
7. 新しい Manager をホストできるセルフホストエンジンノードに通常のホストを変換します。

この手順の前提として、移行元の Manager に対するアクセス権があり、変更を加えることできる必要があります。

### 前提条件

- Manager およびデプロイメントホスト用の完全修飾ドメイン名を準備しておく。正引き (フォワードルックアップ) と逆引き (リバースルックアップ) の記録は、両方とも DNS で設定する必要があります。新しい Manager には移行元の Manager と同じ FQDN を指定する必要があります。
- Manager 用仮想マシンを管理できるように、管理ネットワーク (デフォルトは `ovirtmgmt`) を **仮想マシンネットワーク** として設定する必要があります。

## 第2章 セルフホストエンジン用デプロイメントホストのインストール

セルフホストエンジンは、[Red Hat Virtualization Host](#) または [Red Hat Enterprise Linux ホスト](#) からデプロイすることができます。



### 重要

高可用性のためにボンドインターフェイスを使用する、またはトラフィックをタイプごとに分離するために VLAN を使用する場合は (例: ストレージ用の接続と管理用の接続)、セルフホストエンジンのデプロイメント開始前にホストに設定する必要があります。詳細は、[プランニングおよび前提条件ガイド](#) の [ネットワークの推奨事項](#) を参照してください。

### 2.1. RED HAT VIRTUALIZATION HOST のインストール

Red Hat Virtualization Host (RHVH) は、Red Hat Virtualization 環境でハイパーバイザーとして機能する物理マシンの簡単な設定方法を提供するために設計された、Red Hat Enterprise Linux をベースとする最小設定のオペレーティングシステムです。この最小設定のオペレーティングシステムには、マシンがハイパーバイザーとして機能するのに必要なパッケージのみが含まれており、ホストの監視や管理タスクの実行用に Cockpit Web インターフェイスが備えられています。ブラウザーの最小要件については、[Cockpit の実行](#) を参照してください。

RHVH は NIST SP 800-53 パーティショニングの要件をサポートし、より強固なセキュリティーを提供します。RHVH は、デフォルトで NIST 800-53 パーティションレイアウトを使用します。

ホストは最低限の [ホスト要件](#) を満たしている必要があります。



### 警告

ホストのオペレーティングシステムのインストールまたは再インストールを行う場合、Red Hat では、ホストにアタッチされている既存の OS 以外のストレージを最初にデタッチすることを強く推奨しています。これは、ディスクを誤って初期化してデータが失われる可能性を避けるためです。

### 手順

1. Red Hat カスタマーポータル [の Red Hat Virtualization を使い始める](#) にアクセスし、ログインします。
2. **Download Latest** をクリックして、製品のダウンロードページに移動します。
3. 一覧から RHV に適した **Hypervisor Image for RHV** を選択し、**Download Now** をクリックします。
4. RHVH のインストール先となるマシンを起動し、準備したインストールメディアから起動します。
5. 起動メニューから **Install RHVH 4.4** を選択し、**Enter** を押します。

**注記**

ま**Tab** キーを押してカーネルパラメーターを編集することもできます。カーネルパラメーターはスペースで区切る必要があります。また、指定したカーネルパラメーターを使用してシステムを起動するには、**Enter** キーを押します。**Esc** キーを押してカーネルパラメーターへの変更を消去し、起動メニューに戻ります。

6. 言語を選択し、**Continue** をクリックします。
7. **Keyboard Layout** の画面からキーボードのレイアウトを選択して **Done** をクリックします。
8. **Installation Destination** の画面から RHVH のインストール先のデバイスを選択します。オプションで暗号化を有効にします。**Done** をクリックします。

**重要**

**Automatically configure partitioning** オプションを使用します。

9. **Time & Date** の画面からタイムゾーンを選択し、**Done** をクリックします。
10. **Network & Host Name** の画面からネットワークを選択し、**Configure...** をクリックして接続の詳細を設定します。

**注記**

システムを起動するたびに接続を使用する場合は、**Connect automatically with priority** のチェックボックスを選択します。詳細は、[標準的な RHEL 8 インストールの実行の ネットワークおよびホスト名のオプションの設定](#) を参照してください。

ホスト名を **Host Name** フィールドに入力し、**Done** をクリックします。

11. オプション: **Security Policy** と **Kdump** を設定します。**Installation Summary** 画面の各セクションの詳細は、Red Hat Enterprise Linux 8 [標準的な RHEL インストールの実行の GUI を使用したインストールのカスタマイズ](#) を参照してください。
12. **Begin Installation** をクリックします。
13. RHVH のインストールの際に root パスワードを設定して、オプションで追加のユーザーを作成します。

**警告**

ローカルのセキュリティー脆弱性が攻撃される可能性があるため、RHVH に信頼できないユーザーを作成しないでください。

14. **Reboot** をクリックしてインストールを完了します。



### 注記

RHVH の再起動時に、**nodectl check** はホストでヘルスチェックを実行し、コマンドラインへのログイン時に結果を表示します。**node status: OK** または **node status: DEGRADED** のメッセージはヘルスステータスを示します。**nodectl check** を実行して詳細情報を取得します。



### 注記

必要に応じて、**カーネルモジュールが自動的に読み込まれないようにする** ことができます。

## 2.1.1. Red Hat Virtualization Host のリポジトリの有効化

更新を受け取るには、システムを登録する必要があります。Red Hat Virtualization Host に必要なリポジトリは1つだけです。本セクションでは、RHVH を **コンテンツ配信ネットワーク** または **Red Hat Satellite 6** に登録する手順について説明します。

### コンテンツ配信ネットワークへの RHVH の登録

1. **Red Hat Virtualization Host 8** のリポジトリを有効にして、Red Hat Virtualization Host に対する後続の更新を可能にします。

```
# subscription-manager repos --enable=rhvh-4-for-rhel-8-x86_64-rpms
```

### Red Hat Satellite 6 への RHVH の登録

1. **https://HostFQDNorIP:9090** で Cockpit Web インターフェイスにログインします。
2. **Terminal** をクリックします。
3. RHVH を Red Hat Satellite 6 に登録します。

```
# rpm -Uvh http://satellite.example.com/pub/katello-ca-consumer-latest.noarch.rpm
# subscription-manager register --org="org_id"
# subscription-manager list --available
# subscription-manager attach --pool=pool_id
# subscription-manager repos \
--disable='*' \
--enable=rhvh-4-for-rhel-8-x86_64-rpms
```



### 注記

**virt-who** を使用して、Red Hat Satellite で仮想マシンのサブスクリプションを設定することもできます。**virt-who** を使用したホストベースのサブスクリプションの管理 を参照してください。

## 2.2. RED HAT ENTERPRISE LINUX ホストのインストール

Red Hat Enterprise Linux ホストは、**Red Hat Enterprise Linux Server** および **Red Hat Virtualization** サブスクリプションがアタッチされた、物理サーバー上の Red Hat Enterprise Linux 8 の標準的な基本インストールをベースにしています。

詳細なインストール手順は、[標準的な RHEL インストールの実行](#) を参照してください。

ホストは最低限の [ホスト要件](#) を満たしている必要があります。



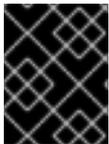
### 警告

ホストのオペレーティングシステムのインストールまたは再インストールを行う場合、Red Hat では、ホストにアタッチされている既存の OS 以外のストレージを最初にデタッチすることを強く推奨しています。これは、ディスクを誤って初期化してデータが失われる可能性を避けるためです。



### 重要

ホストの BIOS 設定で仮想化が有効になっている必要があります。ホストの BIOS 設定の変更に関する詳細は、そのホストのハードウェアのマニュアルを参照してください。



### 重要

サードパーティー製の watchdogs は、Red Hat Enterprise Linux ホストにインストールしないでください。VDSM が提供する watchdog デーモンを妨げる可能性があります。

## 2.2.1. Red Hat Enterprise Linux ホストのリポジトリーの有効化

Red Hat Enterprise Linux マシンをホストとして使用するには、システムをコンテンツ配信ネットワークに登録し、**Red Hat Enterprise Linux Server** および **Red Hat Virtualization** サブスクリプションを割り当て、ホストのリポジトリーを有効にする必要があります。

### 手順

1. コンテンツ配信ネットワークにシステムを登録します。プロンプトが表示されたら、カスタマーポータルของผู้ーザー名とパスワードを入力します。

```
# subscription-manager register
```

2. **Red Hat Enterprise Linux Server** および **Red Hat Virtualization** のサブスクリプションプールを見つけ、プール ID を記録します。

```
# subscription-manager list --available
```

3. 上記のプール ID を使用して、サブスクリプションをシステムにアタッチします。

```
# subscription-manager attach --pool=poolid
```



### 注記

現在アタッチされているサブスクリプションを表示するには、以下のコマンドを実行します。

```
# subscription-manager list --consumed
```

有効なリポジトリをすべて一覧表示するには、以下のコマンドを実行します。

```
# dnf repolist
```

#### 4. リポジトリを設定します。

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-8-for-x86_64-baseos-eus-rpms \
  --enable=rhel-8-for-x86_64-appstream-eus-rpms \
  --enable=rhv-4-mgmt-agent-for-rhel-8-x86_64-rpms \
  --enable=fast-datapath-for-rhel-8-x86_64-rpms \
  --enable=advanced-virt-for-rhel-8-x86_64-rpms \
  --enable=openstack-16.2-cinderlib-for-rhel-8-x86_64-rpms \
  --enable=rhceph-4-tools-for-rhel-8-x86_64-rpms
```

#### 5. RHEL のバージョンを 8.6 に設定します。

```
# subscription-manager release --set=8.6
```

#### 6. virt モジュールをリセットします。

```
# dnf module reset virt
```



### 注記

Advanced Virtualization ストリームでこのモジュールがすでに有効になっている場合は、この手順は必要なく、マイナス要因となることもありません。

以下を入力してストリームの値を確認できます。

```
# dnf module list virt
```

#### 7. 以下のコマンドを使用して、Advanced Virtualization ストリームで virt モジュールを有効にします。

- RHV 4.4.2 の場合:

```
# dnf module enable virt:8.2
```

- RHV 4.4.3 から 4.4.5 に対応しています。

```
# dnf module enable virt:8.3
```

- RHV 4.4.6 - 4.4.10 の場合:

```
# dnf module enable virt:av
```

- RHV 4.4 以降の場合:

```
# dnf module enable virt:rhel
```



### 注記

RHEL 8.6 以降、Advanced Virtualization パッケージは標準の **virt:rhel** モジュールを使用します。RHEL 8.4 および 8.5 では、1つの Advanced Virtualization ストリーム **rhel:av** のみが使用されます。

1. 現在インストールされている全パッケージを最新の状態にします。

```
# dnf upgrade --nobest
```

2. マシンを再起動します。



### 注記

必要に応じて、[カーネルモジュールが自動的に読み込まれないようにする](#)ことができます。

既存のストレージドメインはスタンドアロンの Manager から移行されますが、Manager 用仮想マシン専用のセルフホストエンジンストレージドメインに追加のストレージを準備する必要があります。

## 第3章 RED HAT VIRTUALIZATION 用ストレージの準備

新たな環境のストレージドメインとして使用するストレージを準備する必要があります。Red Hat Virtualization 環境には少なくとも1つのデータストレージドメインが必要ですが、さらに追加することを推奨します。



### 警告

ホストのオペレーティングシステムのインストールまたは再インストールを行う場合、Red Hat では、ホストにアタッチされている既存の OS 以外のストレージを最初にデタッチすることを強く推奨しています。これは、ディスクを誤って初期化してデータが失われる可能性を避けるためです。

データドメインには、データセンター内の仮想マシンおよびテンプレートの仮想ハードディスクと OVF ファイルを格納します。このドメインは、アクティブな間は複数のデータセンター間で共有することはできません (ただし、データセンター間で移行することは可能です)。複数のストレージタイプのデータドメインを同じデータセンターに追加することは可能ですが、それらはすべてローカルドメインではなく、全ホストがアクセス可能なドメインであることが条件となります。

以下のストレージタイプのいずれかを使用することができます。

- NFS
- iSCSI
- ファイバーチャネル (FCP)
- Red Hat Gluster Storage

### 前提条件

- セルフホストエンジンには、Manager 用仮想マシン専用の 74 GiB 以上のデータドメインが追加が必要です。セルフホストエンジンのインストーラーがこのドメインを作成します。インストール前にこのドメイン用のストレージを準備します。



### 警告

セルフホストエンジンのデプロイメント後にセルフホストエンジン用ストレージドメインを拡張または変更することはサポートされていません。このような変更により、セルフホストエンジンが起動できなくなる可能性があります。

- ブロックストレージドメイン (FCP または iSCSI のいずれか) を使用する場合、セルフホストエンジンでサポートされる設定は、単一のターゲット LUN のみとなります。

- iSCSI ストレージを使用する場合には、セルフホストエンジン用ストレージドメインは専用の iSCSI ターゲットを使用する必要があります。追加のストレージドメインは、異なる iSCSI ターゲットを使用しなければなりません。
- セルフホストエンジン用ストレージドメインと同じデータセンター内に追加のデータストレージドメインを作成することを強く推奨します。セルフホストエンジンをデータセンター内にデプロイする際に、アクティブなデータストレージドメインを1つしか用意していない場合、そのストレージドメインが破損しても、新しいストレージドメインを追加したり、破損したストレージドメインを削除することはできません。セルフホストエンジンを再デプロイしなければなりません。

### 3.1. NFS ストレージの準備

ファイルストレージまたはリモートサーバーで NFS 共有を設定し、Red Hat Enterprise Virtualization Host システムのストレージドメインとして機能するようにします。リモートストレージで共有をエクスポートし、Red Hat Virtualization Manager で共有を設定すると、共有は Red Hat Virtualization Host に自動的にインポートされます。

NFS の準備、設定、マウント、およびエクスポートに関する詳細は、Red Hat Enterprise Linux 8 の [ファイルシステムの管理](#) を参照してください。

Red Hat Virtualization には、特定のシステムユーザーアカウントおよびシステムユーザーグループが必要です。これにより、Manager はストレージドメイン (エクスポートしたディレクトリー) にデータを保管することができます。以下の手順では、1つのディレクトリーのパーミッションを設定しています。Red Hat Virtualization のストレージドメインとして使用するすべてのディレクトリーについて、**chown** および **chmod** のステップを繰り返す必要があります。

#### 前提条件

1. NFS **utils** パッケージをインストールする。

```
# dnf install nfs-utils -y
```

2. 以下のコマンドを実行して、有効なバージョンを確認する。

```
# cat /proc/fs/nfsd/versions
```

3. 以下のサービスを有効にする。

```
# systemctl enable nfs-server  
# systemctl enable rpcbind
```

#### 手順

1. **kvm** グループを作成します。

```
# groupadd kvm -g 36
```

2. **kvm** グループに **vdsm** ユーザーを作成します。

```
# useradd vdsm -u 36 -g kvm
```

3. **storage** ディレクトリーを作成し、アクセス権を変更します。

```
# mkdir /storage
# chmod 0755 /storage
# chown 36:36 /storage/
```

4. **storage** ディレクトリーを、適切なパーミッションで **/etc/exports** に追加します。

```
# vi /etc/exports
# cat /etc/exports
/storage *(rw)
```

5. 以下のサービスを再起動します。

```
# systemctl restart rpcbind
# systemctl restart nfs-server
```

6. 特定の IP アドレスで利用可能なエクスポートを確認するには、以下のコマンドを実行します。

```
# exportfs
/nfs_server/srv
      10.46.11.3/24
/nfs_server <world>
```



### 注記

サービス起動後に **/etc/exports** を変更した場合は、**exportfs -ra** コマンドを使用してその変更を再読み込みできます。上記のすべての手順を実行すると、**exports** ディレクトリーが準備でき、利用可能かどうかを確認するため、別のホストでテストすることができます。

## 3.2. iSCSI ストレージの準備

Red Hat Virtualization は、LUN で設定されるボリュームグループから作成されるストレージドメインである iSCSI ストレージをサポートします。ボリュームグループおよび LUN は、いずれも同時に複数のストレージドメインにアタッチすることはできません。

iSCSI ストレージのセットアップおよび設定に関する詳細は、Red Hat Enterprise Linux 8 の [ストレージデバイスの管理](#) で、[iSCSI ターゲットの設定](#) を参照してください。



### 重要

ブロックストレージを使用する際に、仮想マシンを Raw デバイスまたは直接 LUN にデプロイして論理ボリュームマネージャー (LVM) で管理する場合は、フィルターを作成してゲストの論理ボリュームを除外する必要があります。これにより、ホストの起動時にゲストの論理ボリュームがアクティブ化されるのを防ぐことができます。アクティブ化されると、論理ボリュームの内容が古くなり、データ破損が生じる可能性があります。**vdsms-tool config-lvm-filter** コマンドを使用して、LVM のフィルターを作成します。[LVM フィルターの作成](#) を参照してください。



### 重要

現状、Red Hat Virtualization はブロックサイズ 4K のブロックストレージはサポートしていません。ブロックストレージはレガシー (512b ブロック) モードで設定する必要があります。

 **重要**

SAN ストレージから起動したホストがストレージへの接続を失うと、ストレージファイルシステムは読み取り専用になり、接続が回復した後もその状態が続きます。

この状態を回避するには、ブート LUN の SAN のルートファイルシステムにドロップインマルチパス設定ファイルを追加し、接続可能な場合にキューに置かれるようにしてください。

```
# cat /etc/multipath/conf.d/host.conf
multipaths {
  multipath {
    wwid boot_LUN_wwid
    no_path_retry queue
  }
}
```

### 3.3. FCP ストレージの準備

Red Hat Virtualization は、既存の LUN で設定されるボリュームグループからストレージドメインを作成することで、SAN ストレージをサポートしています。ボリュームグループおよび LUN は、いずれも同時に複数のストレージドメインにアタッチすることはできません。

Red Hat Virtualization システムの管理者には Storage Area Networks (SAN) に関する作業知識が必要になります。SAN は通常、ホストと外部の共有ストレージ間のトラフィックにファイバーチャネルプロトコル (FCP) を使用します。このため、SAN は FCP ストレージとも呼ばれています。

Red Hat Enterprise Linux での FCP またはマルチパスの準備および設定に関する情報は、[ストレージ管理ガイド](#) および [DM Multipath ガイド](#) を参照してください。

 **重要**

ブロックストレージを使用する際に、仮想マシンを Raw デバイスまたは直接 LUN にデプロイして論理ボリュームマネージャー (LVM) で管理する場合は、フィルターを作成してゲストの論理ボリュームを除外する必要があります。これにより、ホストの起動時にゲストの論理ボリュームがアクティブ化されるのを防ぐことができます。アクティブ化されると、論理ボリュームの内容が古くなり、データ破損が生じる可能性があります。**vdsm-tool config-lvm-filter** コマンドを使用して、LVM のフィルターを作成します。[LVM フィルターの作成](#) を参照してください。

 **重要**

現状、Red Hat Virtualization はブロックサイズ 4K のブロックストレージはサポートしていません。ブロックストレージはレガシー (512b ブロック) モードで設定する必要があります。

## 重要

SAN ストレージから起動したホストがストレージへの接続を失うと、ストレージファイルシステムは読み取り専用になり、接続が回復した後もその状態が続きます。

この状態を回避するには、ブート LUN の SAN のルートファイルシステムにドロップインマルチパス設定ファイルを追加し、接続可能な場合にキューに置かれるようにしてください。

```
# cat /etc/multipath/conf.d/host.conf
multipaths {
  multipath {
    wwid boot_LUN_wwid
    no_path_retry queue
  }
}
```

## 3.4. RED HAT GLUSTER STORAGE の準備

Red Hat Gluster Storage の準備および設定に関する情報は、[Red Hat Gluster Storage インストールガイド](#) を参照してください。

Red Hat Virtualization でサポートされている Red Hat Gluster Storage のバージョンについては、[Red Hat Gluster Storage バージョンの互換性とサポート](#) を参照してください。

## 3.5. SAN ベンダーのマルチパス設定のカスタマイズ

RHV 環境が SAN とのマルチパス接続を使用するように設定されている場合には、ストレージベンダーが指定する要件を満たすようにマルチパス設定をカスタマイズできます。このカスタマイズは、`/etc/multipath.conf` で指定した設定と、デフォルトの設定の両方を上書きできます。

マルチパス設定を上書きする場合は、`/etc/multipath.conf` をカスタマイズしないでください。VDSM は `/etc/multipath.conf` を所有しているため、VDSM または Red Hat Virtualization をインストールまたはアップグレードすると、カスタマイズを含むこのファイルが上書きされます。この上書きにより、重大なストレージ障害が発生する可能性があります。

代わりに、カスタマイズまたは上書きする設定が含まれる `/etc/multipath/conf.d` ディレクトリーにファイルを作成します。

VDSM は、`/etc/multipath/conf.d` のファイルをアルファベット順に実行します。実行の順番を制御するには、ファイル名を番号で開始し、アルファベット順後の最後に来るようにします。たとえば、`/etc/multipath/conf.d/90-myfile.conf` です。

重大なストレージ障害を引き起こさないように、以下のガイドラインに従ってください。

- `/etc/multipath.conf` は変更しないでください。ファイルにユーザー変更が含まれる場合にこのファイルが上書きされると、想定外のストレージ障害が発生する可能性があります。



### 警告

これらのガイドラインに従わないと、非常に深刻なストレージ障害が発生する可能性があります。

### 前提条件

- VDSM がマルチパスモジュールを使用するように設定されている。これを確認するには、以下を入力します。

```
# vdsmd --is-configured --module multipath
```

### 手順

1. `/etc/multipath/conf.d` ディレクトリーに新しい設定ファイルを作成します。
2. 上書きする個々の設定を、`/etc/multipath.conf` から `/etc/multipath/conf.d/<my_device>.conf` 内の新しい設定ファイルにコピーします。コメントマークを削除して設定値を編集し、変更を保存します。
3. 以下を入力して、新しい設定を適用します。

```
# systemctl reload multipathd
```



### 注記

`multipathd` サービスを再起動しないでください。これにより、VDSM ログにエラーが生成されます。

### 検証手順

1. さまざまな障害シナリオで実稼働クラスター以外のクラスターを使用して、新しい設定が想定どおりに実行されことをテストします。たとえば、ストレージの接続をすべて無効にします。
2. 一度に1つの接続を有効にし、これによりストレージドメインに到達可能であることを確認します。

### 関連情報

- [推奨される Multipath.conf 設定](#)
- [Red Hat Enterprise Linux DM Multipath](#)
- [iSCSI マルチパス化の設定](#)
- [How do I customize /etc/multipath.conf on my RHV-H hypervisors? What values must not change and why?](#)

## 3.6. 推奨される MULTIPATH.CONF 設定

以下の設定は上書きしないでください。

#### user\_friendly\_names no

デバイス名は、すべてのハイパーバイザーで一貫性を保つ必要があります。たとえば、`/dev/mapper/{WWID}` です。この設定のデフォルト値 **no** は、さまざまなハイパーバイザー上の `/dev/mapper/mpath{N}` など、任意で一貫性のないデバイス名の割り当てを阻止するため、システムの動作が予測できない可能性があります。



#### 警告

この設定を **user\_friendly\_names yes** に変更しないでください。ユーザーフレンドリーな名前を使用すると、システムの想定外の動作や障害が発生する可能性が高く、サポートされていません。

#### find\_multipaths no

複数のパスを利用できる場合に RHVH がマルチパスを介してデバイスへのアクセスを試みるかどうかを制御します。現在の値 **no** を使用すると、利用できるパスが1つしかない場合でも、RHV がマルチパスを介してデバイスにアクセスできるようになります。



#### 警告

この設定は上書きしないでください。

ストレージシステムベンダーが必要な場合を除き、以下の設定は上書きしないでください。

#### no\_path\_retry 4

利用可能なパスがない場合にポーリングを再試行する回数を制御します。RHV バージョン 4.2 より前は、パスが利用できない場合に QEMU の I/O キューに問題が生じていたため、**no\_path\_retry** の値は **fail** でした。**fail** 値により、仮想マシンはすぐに失敗し、一時停止していました。RHV バージョン 4.2 ではこの値が **4** に変更されました。これにより、multipathd は最後のパスが失敗したことを検知すると、すべてのパスをさらに 4 回確認します。ポーリングがデフォルトの 5 秒間隔で行われると仮定すると、パスの確認には 20 秒かかります。パスが起動しない場合、multipathd は、パスが復元されるまでキューを停止するようにカーネルに指示し、未処理および将来の I/O をすべて失敗させます。パスが復元されると、次にすべてのパスが失敗したときのために、20 秒間のパスの確認時間がリセットされます。詳細は、[この設定を変更したコミット](#) を参照してください。

#### polling\_interval 5

パスが開いているか、または失敗したかを検出するポーリングの試行間隔の秒数を決定します。ベンダーが値を増やす理由を明示しない限り、VDSM が生成するデフォルト値を維持します。これにより、システムはパスの失敗に早めの対応することができます。

Manager をバックアップする前に、最新のマイナーバージョンに更新されていることを確認します。バックアップファイルの Manager バージョンは、新しい Manager のバージョンと一致する必要があります。

## 第4章 RED HAT VIRTUALIZATION MANAGER の更新

### 前提条件

- 更新されたストレージバージョンとの互換性を確保するために、データセンターの互換性レベルを最新バージョンに設定する必要があります。

### 手順

1. Manager マシンで、更新されたパッケージが利用可能かどうかを確認します。

```
# engine-upgrade-check
```

2. setup パッケージを更新します。

```
# yum update ovirt-*setup* rh-*vm-setup-plugins
```

3. **engine-setup** スクリプトで Red Hat Virtualization Manager を更新します。**engine-setup** スクリプトにより、設定に関する質問への回答が求められます。その後、**ovirt-engine** サービスの停止、更新パッケージのダウンロード/インストール、データベースのバックアップ/更新、インストール後設定の実施を経てから、**ovirt-engine** サービスが起動します。

```
# engine-setup
```

スクリプトが正常に完了すると、以下のメッセージが表示されます。

```
Execution of setup completed successfully
```



### 注記

**engine-setup** スクリプトは、Red Hat Virtualization Manager のインストールプロセス中にも使用され、指定した設定値が保存されます。更新時に、設定をプレビューすると保存された値が表示されますが、インストール後に **engine-config** を使用して設定を更新した場合、この値は最新ではない可能性があります。たとえば、インストール後に **engine-config** を使用して **SANWipeAfterDelete** を **true** に更新した場合、**engine-setup** は設定プレビューに "Default SAN wipe after delete: False" を出力します。ただし、更新された値が **engine-setup** によって上書きされることはありません。



### 重要

更新プロセスに時間がかかる場合があります。完了するまでプロセスを停止しないでください。

4. Manager にインストールされているベースオペレーティングシステムと、オプションパッケージを更新します。

```
# yum update --nobest
```



### 重要

更新中に必要な Ansible パッケージの競合が発生した場合は、[RHV Manager](#) で `yum update` を実行できない ([ansible の競合](#)) を参照してください。



### 重要

いずれかのカーネルパッケージが更新された場合には、マシンを再起動して更新を完了してください。

## 第5章 元の MANAGER のバックアップ

**engine-backup** コマンドを使用して元の Manager をバックアップし、バックアップファイルを別の場所にコピーして、処理中にいつでもアクセスできるようにします。

**engine-backup --mode=backup** オプションの詳細は、[管理ガイド](#) の [Red Hat Virtualization Manager のバックアップと復元](#) を参照してください。

### 手順

1. 元の Manager にログインし、**ovirt-engine** サービスを停止します。

```
# systemctl stop ovirt-engine
# systemctl disable ovirt-engine
```



### 注記

元の Manager の実行を停止することは必須ではありませんが、バックアップの作成後に環境を変更しないように推奨しています。さらに、元の Manager と新しい Manager が既存リソースを同時に管理しないようにします。

2. 作成するバックアップファイルの名前と、バックアップログを保存するログファイルの名前を指定して、**engine-backup** コマンドを実行します。

```
# engine-backup --mode=backup --file=file_name --log=log_file_name
```

3. ファイルを外部サーバーにコピーします。以下の例の **storage.example.com** は、必要になるまでバックアップを保存するネットワークストレージサーバーの完全修飾ドメイン名です。**/backup/** は指定のフォルダーまたはパスです。

```
# scp -p file_name log_file_name storage.example.com:/backup/
```

4. Manager マシンを他の目的で必要としない場合は、Red Hat Subscription Manager から登録を解除します。

```
# subscription-manager unregister
```

Manager のバックアップ後に、新しいセルフホストエンジンをデプロイし、新しい仮想マシンにバックアップを復元します。

## 第6章 新しいセルフホストエンジンでのバックアップの復元

**hosted-engine** スクリプトを新規ホストで実行し、デプロイメント中に **--restore-from-file=path/to/file\_name** オプションを使用して Manager バックアップを復元します。

### 重要

iSCSI ストレージを使用し、イニシエーターの ACL に従い iSCSI ターゲットフィルターを使用して接続をフィルタリングすると、デプロイメントは **STORAGE\_DOMAIN\_UNREACHABLE** エラーで失敗する可能性があります。これを回避するには、セルフホストエンジンのデプロイメントを開始する前に iSCSI 設定を更新する必要があります。

- 既存のホストに再デプロイする場合は、**/etc/iscsi/initiatorname.iscsi** でホストの iSCSI イニシエーター設定を更新する必要があります。イニシエーター IQN は、iSCSI ターゲットで以前にマッピングされていたものと同じか、必要に応じて新しい IQN に更新する必要があります。
- 新規ホストにデプロイする場合は、iSCSI ターゲット設定を更新して、ホストからの接続を受け入れる必要があります。

IQN はホスト側 (iSCSI イニシエーター) またはストレージ側 (iSCSI ターゲット) で更新できることに注意してください。

### 手順

1. バックアップファイルを新規ホストにコピーします。以下の例では、**host.example.com** はホストの FQDN、**/backup/** は指定されたフォルダーまたはパスです。

```
# scp -p file_name host.example.com:/backup/
```

2. 新しいホストにログインします。
3. Red Hat Virtualization Host にデプロイする場合は、**ovirt-hosted-engine-setup** はすでにインストールされているため、この手順を省略します。Red Hat Enterprise Linux にデプロイする場合は、**ovirt-hosted-engine-setup** パッケージをインストールします。

```
# dnf install ovirt-hosted-engine-setup
```

4. ネットワークやターミナルが切断された場合などにセッションが失われないように、**tmux** ウィンドウマネージャーを使用してスクリプトを実行します。**tmux** をインストールし、実行します。

```
# dnf -y install tmux
# tmux
```

5. バックアップファイルへのパスを指定して **hosted-engine** スクリプトを実行します。

```
# hosted-engine --deploy --restore-from-file=backup/file_name
```

任意のタイミングでスクリプトをエスケープするには、**CTRL+D** を使用してデプロイメントを中止します。

6. **Yes** を選択してデプロイメントを開始します。

7. ネットワークを設定します。スクリプトにより、環境の管理ブリッジとして使用する NIC 候補が検出されます。
8. 仮想マシンのインストールにカスタムアプライアンスを使用する場合は、OVA アーカイブへのパスを入力します。使用しない場合は、このフィールドを空欄のままにして RHV-M Appliance を使用します。
9. Manager の root パスワードを入力します。
10. root ユーザーとして Manager にログインできる SSH 公開鍵を入力し、root ユーザーの SSH アクセスを有効にするかどうかを指定します。
11. 仮想マシンの CPU およびメモリー設定を入力します。



### 注記

仮想マシンには、Manager の移行元の物理マシンと同じ RAM が必要です。Manager の移行元の物理マシンより RAM が少ない仮想マシンに移行する必要がある場合は、[Configuring the amount of RAM in Red Hat Virtualization Hosted Engine](#) を参照してください。

12. Manager 用仮想マシンの MAC アドレスを入力するか、無作為に生成される MAC アドレスを適用します。Manager 用仮想マシンへの IP アドレス割り当てに DHCP を使用する場合は、この MAC アドレスに有効な DHCP 予約があることを確認してください。デプロイメントスクリプトは、DHCP サーバーの設定は行いません。
13. 仮想マシンのネットワーク情報を入力します。**Static** を指定する場合は、Manager の IP アドレスを入力します。



### 重要

静的 IP アドレスは、ホストと同じサブネットに属している必要があります。たとえばホストが 10.1.1.0/24 内にある場合、Manager 用仮想マシンの IP は同じサブネット範囲 (10.1.1.1-254/24) になければなりません。

14. Manager 用仮想マシンおよびベースホストのエントリーを、仮想マシンの **/etc/hosts** ファイルに追加するかどうかを指定します。ホスト名は解決可能でなければなりません。
15. SMTP サーバーの名前と TCP ポート番号、メール通知を送信するメールアドレス、メール通知を受信するメールアドレス (複数ある場合はコンマ区切りリスト) を指定します。
16. 管理ポータルにアクセスする際に使用する **admin@internal** ユーザーのパスワードを入力します。  
スクリプトにより仮想マシンが作成されます。RHV-M Appliance をインストールする必要がある場合は、時間がかかることがあります。



## 注記

必要なネットワークがないなどの理由でホストが動作しなくなると、デプロイが一時停止し、次のようなメッセージが表示されます。

```
[ INFO ] You can now connect to https://<host name>:6900/ovirt-engine/ and
check the status of this host and eventually remediate it, please continue only
when the host is listed as 'up'
[ INFO ] TASK [ovirt.ovirt.hosted_engine_setup : include_tasks]
[ INFO ] ok: [localhost]
[ INFO ] TASK [ovirt.ovirt.hosted_engine_setup : Create temporary lock file]
[ INFO ] changed: [localhost]
[ INFO ] TASK [ovirt.ovirt.hosted_engine_setup : Pause execution until
/tmp/ansible.<random>_he_setup_lock is removed, delete it once ready to
proceed]
```

プロセスを一時停止すると、以下が可能になります。

- 提供された URL を使用して管理ポータルに接続します。
- 状況を評価し、ホストが動作していない理由を調べ、必要に応じて修正します。たとえば、このデプロイメントがバックアップから復元され、ホストクラスタに **必要なネットワーク** がバックアップに含まれている場合は、ネットワークを設定し、関連するホスト NIC をこれらのネットワークに接続します。
- すべてが正常に見え、ホストのステータスが **Up** になったら、上記のメッセージに表示されているロックファイルを削除します。デプロイメントは続行されます。

## 17. 使用するストレージのタイプを選択します。

- NFS の場合は、バージョン、完全なアドレス、およびストレージへのパスならびにマウントオプションを入力します。
- iSCSI の場合は、ポータルの詳細を入力し、自動検出された一覧からターゲットおよび LUN を選択します。デプロイメント時に選択できる iSCSI ターゲットは1つだけですが、マルチパスがサポートされているので、同じポータルグループのポータルをすべて接続できます。



## 注記

複数の iSCSI ターゲットを指定するには、セルフホストエンジンをデプロイする前にマルチパスを有効にする必要があります。詳細は、[Red Hat Enterprise Linux DM マルチパス](#) を参照してください。[Multipath Helper](#) ツールを使用して、さまざまなオプションでマルチパスをインストールおよび設定するスクリプトを生成することもできます。

- Gluster ストレージの場合は、完全なアドレスおよびストレージへのパスならびにマウントオプションを入力します。



## 重要

レプリカ1およびレプリカ3 Gluster ストレージのみがサポートされます。必ず以下のようにボリュームを設定します。

```
gluster volume set VOLUME_NAME group virt
gluster volume set VOLUME_NAME performance.strict-o-direct on
gluster volume set VOLUME_NAME network.remote-dio off
gluster volume set VOLUME_NAME storage.owner-uid 36
gluster volume set VOLUME_NAME storage.owner-gid 36
gluster volume set VOLUME_NAME network.ping-timeout 30
```

- ファイバーチャネルの場合は、自動検出された一覧から LUN を選択します。ホストのバスアダプターが設定および接続されている必要があります。また、LUN には既存のデータが含まれないようにする必要があります。既存の LUN を再利用するには、[管理ガイドの LUN の再利用](#) を参照してください。

18. Manager のディスクサイズを入力します。  
スクリプトはデプロイメントが完了するまで続行されます。
19. デプロイメントプロセスでは Manager の SSH キーが変更されます。クライアントマシンが SSH エラーなしで新規の Manager にアクセスできるようにするには、元の Manager にアクセスするクライアントマシンの `.ssh/known_hosts` ファイルから元の Manager のエントリーを削除します。

デプロイメントが完了したら、新しい Manager 仮想マシンにログインし、必要なりポジトリーを有効にします。

## 第7章 RED HAT VIRTUALIZATION MANAGER リポジトリの有効化

ログインして、Red Hat Subscription Manager で Manager マシンを登録し、**Red Hat Virtualization Manager** のサブスクリプションをアタッチして Manager のリポジトリを有効にする必要があります。

### 手順

1. コンテンツ配信ネットワークにシステムを登録します。プロンプトが表示されたら、カスタマーポータルของผู้ーザー名とパスワードを入力します。

```
# subscription-manager register
```



#### 注記

IPv6 ネットワークを使用している場合は、IPv6 移行メカニズムを使用して、コンテンツ配信ネットワークおよびサブスクリプションマネージャーにアクセスします。

2. **Red Hat Virtualization Manager** のサブスクリプションプールを見つけ、プール ID を記録します。

```
# subscription-manager list --available
```

3. 上記のプール ID を使用して、サブスクリプションをシステムにアタッチします。

```
# subscription-manager attach --pool=pool_id
```



#### 注記

現在アタッチされているサブスクリプションを表示するには、以下のコマンドを実行します。

```
# subscription-manager list --consumed
```

有効なリポジトリをすべて一覧表示するには、以下のコマンドを実行します。

```
# dnf repolist
```

4. リポジトリを設定します。

```
# subscription-manager repos \
--disable='*' \
--enable=rhel-8-for-x86_64-baseos-eus-rpms \
--enable=rhel-8-for-x86_64-appstream-eus-rpms \
--enable=rhv-4.4-manager-for-rhel-8-x86_64-rpms \
--enable=fast-datapath-for-rhel-8-x86_64-rpms \
--enable=jb-eap-7.4-for-rhel-8-x86_64-rpms \
--enable=openstack-16.2-cinderlib-for-rhel-8-x86_64-rpms \
--enable=rhceph-4-tools-for-rhel-8-x86_64-rpms
```

- 
5. RHEL のバージョンを 8.6 に設定します。

```
# subscription-manager release --set=8.6
```

6. **pki-deps** モジュールを有効にします。

```
# dnf module -y enable pki-deps
```

7. **postgresql** モジュールのバージョン 12 を有効にします。

```
# dnf module -y enable postgresql:12
```

8. **nodejs** モジュールのバージョン 14 を有効にします。

```
# dnf module -y enable nodejs:14
```

9. インストール済みパッケージを同期して、利用可能な最新バージョンに更新します。

```
# dnf distro-sync --nobest
```

## 関連情報

モジュールおよびモジュールストリームの詳細は、[ユーザー空間コンポーネントのインストール、管理、および削除](#) の以下のセクションを参照してください。

- [モジュールストリーム](#)
- [パッケージインストールの前のストリーム選択](#)
- [モジュールストリームのリセット](#)
- [後続のストリームへの切り替え](#)

Red Hat Virtualization Manager がセルフホストエンジンの設定に移行されました。Manager が新しいセルフホストエンジンノード上の仮想マシンで動作します。

ホストは新しい環境で実行されますが、Manager 仮想マシンをホストすることはできません。これらのホストの一部またはすべてをセルフホストエンジンノードに変換できます。

## 第8章 既存のホストのセルフホストエンジンノードとしての再インストール

セルフホスト型エンジン環境内の既存の標準ホストは、Manager 仮想マシンをホストするセルフホスト型エンジンノードに変換できます。



### 警告

ホストのオペレーティングシステムをインストールまたは再インストールする場合、Red Hat では、ホストにアタッチされている既存 OS 以外のストレージを最初にデタッチすることを強く推奨しています。これは、ディスクを誤って初期化してデータが失われる可能性を避けるためです。

### 手順

1. **Compute** → **Hosts** をクリックし、ホストを選択します。
2. **Management** → **Maintenance** をクリックしてから **OK** をクリックします。
3. **Installation** → **Reinstall** をクリックします。
4. **Hosted Engine** タブをクリックし、ドロップダウンリストから **DEPLOY** を選択します。
5. **OK** をクリックします。

ホストは、セルフホストエンジン設定を使用して再インストールされ、管理ポータルで王冠アイコンのフラグを付けます。

セルフホストエンジンノードとしてホストを再インストールした後に、いずれかのノードで以下のコマンドを実行して、新しい環境のステータスを確認できます。

```
# hosted-engine --vm-status
```

新しい環境が問題なく実行されている場合は、元の Manager マシンの使用を停止できます。

## 付録A カーネルモジュールが自動的に読み込まれないようにする手順

モジュールを直接読み込むか、別のモジュールから依存関係として読み込むか、起動プロセス中に読み込むかにかかわらず、カーネルモジュールが自動的に読み込まれないようにします。

### 手順

1. モジュール名は、**modprobe** ユーティリティーの設定ファイルに追加する必要があります。このファイルは、設定ディレクトリー **/etc/modprobe.d** に配置する必要があります。この設定ディレクトリーの詳細は、**modprobe.d** の man ページを参照してください。
2. モジュールが以下のいずれかで読み込まれるように設定されていないか確認してください。

- **/etc/modprobe.conf**
- **/etc/modprobe.d/\***
- **/etc/rc.modules**
- **/etc/sysconfig/modules/\***

```
# modprobe --showconfig <_configuration_file_name_>
```

3. 出力にモジュールが表示される場合は、そのモジュールが無視され、読み込まれないことを確認します。

```
# modprobe --ignore-install <_module_name_>
```

4. 読み込まれている場合は、実行中のシステムからモジュールの読み込みを解除します。

```
# modprobe -r <_module_name_>
```

5. システム固有の設定ファイルに **blacklist** 行を追加して、モジュールを直接読み込まないようにします (例: **/etc/modprobe.d/local-dontload.conf**)。

```
# echo "blacklist <_module_name_> >> /etc/modprobe.d/local-dontload.conf"
```



### 注記

必須モジュールの場合や、別のモジュールにおいて任意の依存関係にある場合、この手順を実行してもモジュールの読み込みは回避されません。

6. オプションのモジュールがオンデマンドで読み込まれないようにします。

```
# echo "install <_module_name_>/bin/false" >> /etc/modprobe.d/local-dontload.conf
```



### 重要

除外したモジュールが他のハードウェアで必要とされている場合、除外してしまうと予期しない結果が生じる可能性があります。

7. **initramfs** のバックアップコピーを作成します。

```
# cp /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -r).img.$(date +%m-%d-%H%M%S).bak
```

8. カーネルモジュールが **initramfs** の一部である場合は、初期 **ramdisk** イメージを再構築し、そのモジュールを省略します。

```
# dracut --omit-drivers <_module_name_> -f
```

9. 現在のカーネルコマンドラインパラメーターを取得します。

```
# grub2-editenv - list | grep kernelopts
```

10. **<\_module\_name\_>.blacklist=1 rd.driver.blacklist=<\_module\_name\_>** を、生成された出力に追加します。

```
# grub2-editenv - set kernelopts="<> <_module_name_>.blacklist=1 rd.driver.blacklist=<_module_name_>"
```

以下に例を示します。

```
# grub2-editenv - set kernelopts="root=/dev/mapper/rhel_example-root ro crashkernel=auto resume=/dev/mapper/rhel_example-swap rd.lvm.lv=rhel_example/root rd.lvm.lv=rhel_example/swap <_module_name_>.blacklist=1 rd.driver.blacklist=<_module_name_>"
```

11. **kdump initramfs** のバックアップコピーを作成します。

```
# cp /boot/initramfs-$(uname -r)kdump.img /boot/initramfs-$(uname -r)kdump.img.$(date +%m-%d-%H%M%S).bak
```

12. **kdump initramfs** から省略するには、**rd.driver.blacklist=<\_module\_name\_>** を **/etc/sysconfig/kdump** の **KDUMP\_COMMANDLINE\_APPEND** 設定に追加します。

```
# sed -i '/^KDUMP_COMMANDLINE_APPEND=/s/"$/ rd.driver.blacklist=module_name"/ /etc/sysconfig/kdump
```

13. **kdump initrd** への変更を適用するには、**kdump** サービスを再起動します。

```
# kdumpctl restart
```

14. **kdump** の初期 **ramdisk** イメージを再構築します。

```
# mkdumprd -f /boot/initramfs-$(uname -r)kdump.img
```

15. システムを再起動します。

## A.1. モジュールの一時削除

モジュールを一時的に削除できます。

## 手順

1. **modprobe** を実行して、現在読み込まれているモジュールを削除します。

```
# modprobe -r <module name>
```

2. モジュールの読み込みを解除できない場合、そのモジュールはプロセスまたは別のモジュールで使用されている可能性があります。その場合はプロセスを終了し、上記で作成した **modprobe** コマンドを別のタイミングで実行してモジュール読み込みを解除します。

## 付録B 法的通知

Copyright © 2022 Red Hat, Inc.

Licensed under the ([Creative Commons Attribution–ShareAlike 4.0 International License](#)). Derived from documentation for the ([oVirt Project](#)). If you distribute this document or an adaptation of it, you must provide the URL for the original version.

Modified versions must remove all Red Hat trademarks.

Red Hat, Red Hat Enterprise Linux, the Red Hat logo, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.