



Red Hat Virtualization 4.4

コマンドラインを使用したセルフホストエンジンの Red Hat Virtualization のインストール

コマンドラインを使用して、Red Hat Virtualization Manager の管理するホスト上で動作する仮想マシンとして Manager をインストールする方法

Red Hat Virtualization 4.4 コマンドラインを使用したセルフホストエンジンの Red Hat Virtualization のインストール

コマンドラインを使用して、Red Hat Virtualization Manager の管理するホスト上で動作する仮想マシンとして Manager をインストールする方法

Red Hat Virtualization Documentation Team

Red Hat Customer Content Services

rhev-docs@redhat.com

法律上の通知

Copyright © 2024 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[®] 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 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.

概要

本書では、自動インストールを設定および実行するためにコマンドラインを使用して、セルフホストエンジン環境をインストールする方法について説明します。なお、セルフホストエンジン環境では、Red Hat Virtualization Manager (または engine) は、自分が管理する同じ環境内の特殊なホスト上で動作する仮想マシンにインストールされます。これが希望する設定ではない場合には、製品ガイドでその他のインストールオプションを参照してください。

目次

| | |
|--|-----------|
| 前書き | 4 |
| RED HAT VIRTUALIZATION の主要コンポーネント | 4 |
| セルフホストエンジンのアーキテクチャー | 4 |
| 第1章 インストールの概要 | 6 |
| 第2章 要件 | 8 |
| 2.1. RED HAT VIRTUALIZATION MANAGER の要件 | 8 |
| 2.2. ホストの要件 | 10 |
| 2.3. ネットワークの要件 | 13 |
| 第3章 RED HAT VIRTUALIZATION 用ストレージの準備 | 26 |
| 3.1. NFS ストレージの準備 | 27 |
| 3.2. ISCSI ストレージの準備 | 28 |
| 3.3. FCP ストレージの準備 | 29 |
| 3.4. RED HAT GLUSTER STORAGE の準備 | 30 |
| 3.5. SAN ベンダーのマルチパス設定のカスタマイズ | 30 |
| 3.6. 推奨される MULTIPATH.CONF 設定 | 31 |
| 第4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 | 33 |
| 4.1. RED HAT VIRTUALIZATION HOST のインストール | 33 |
| 4.2. RED HAT ENTERPRISE LINUX ホストのインストール | 35 |
| 第5章 RED HAT VIRTUALIZATION MANAGER のインストール | 39 |
| 5.1. RHV-M APPLIANCE の手動インストール | 39 |
| 5.2. ファイアウォールの有効化および設定 | 40 |
| 5.3. コマンドラインを使用したセルフホストエンジンのデプロイ | 40 |
| 5.4. RED HAT VIRTUALIZATION MANAGER リポジトリの有効化 | 48 |
| 5.5. 管理ポータルへの接続 | 50 |
| 第6章 RED HAT VIRTUALIZATION 用ホストのインストール | 52 |
| 6.1. RED HAT VIRTUALIZATION HOSTS | 52 |
| 6.2. RED HAT ENTERPRISE LINUX ホスト | 64 |
| 6.3. ホストネットワーク設定の推奨プラクティス | 66 |
| 6.4. RED HAT VIRTUALIZATION MANAGER へのセルフホストエンジンノードの追加 | 68 |
| 6.5. RED HAT VIRTUALIZATION MANAGER への通常のホストの追加 | 69 |
| 第7章 RED HAT VIRTUALIZATION 用ストレージの追加 | 70 |
| 7.1. NFS ストレージの追加 | 70 |
| 7.2. ISCSI ストレージの追加 | 71 |
| 7.3. FCP ストレージの追加 | 73 |
| 7.4. RED HAT GLUSTER STORAGE の追加 | 74 |
| 付録A セルフホストエンジンのデプロイメントのトラブルシューティング | 75 |
| A.1. MANAGER 用仮想マシンのトラブルシューティング | 75 |
| A.2. 失敗したセルフホストエンジンのデプロイメントのクリーンアップ | 78 |
| 付録B デプロイメント中に自動化を使用して MANAGER 仮想マシンをカスタマイズする | 79 |
| 付録C リモートサーバーへのデータベースおよびサービスの移行 | 80 |
| C.1. データウェアハウスを別のマシンに移行する | 80 |
| 付録D PCI パススルーを有効にするためのホストの設定 | 88 |
| 付録E カーネルモジュールの自動読み込みの回避 | 90 |

| | |
|--|-----------|
| E.1. モジュールの一時削除 | 91 |
| 付録F RED HAT VIRTUALIZATION のセキュリティー保護 | 93 |
| F.1. RHEL ベースのホストおよびスタンドアロン MANAGER での DISA STIG プロファイルの適用 | 93 |
| F.2. RHV ホストおよびスタンドアロン MANAGER での PCI-DSS プロファイルの適用 | 94 |
| 付録G セルフホステッドエンジンのデプロイで許可される CPU タイプの定義 | 96 |
| 付録H 法的通知 | 99 |

前書き

セルフホストエンジンのインストールは、Ansible により自動化されています。最初のデプロイメントホスト上でインストールスクリプト (**hosted-engine --deploy**) を実行し、デプロイメントホスト上に作成される仮想マシンに Red Hat Virtualization Manager (または engine) をインストールおよび設定します。Manager および Data Warehouse データベースは Manager 用仮想マシンにインストールされますが、必要であればインストール後に別のサーバーに移行することができます。

Manager 用仮想マシンを実行することのできるホストは、セルフホストエンジンノードと呼ばれます。高可用性機能に対応するためには、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。

Manager 用仮想マシン専用のストレージドメインは、セルフホストエンジン用ストレージドメインと呼ばれます。このストレージドメインはインストールのスクリプトにより作成されるので、インストールの開始前にベースとなるストレージを準備する必要があります。

環境オプションおよび推奨される設定については、[プランニングおよび前提条件ガイド](#) を参照してください。セルフホストエンジン環境に特化した設定については、[Self-Hosted Blue Engine Recommendations](#) を参照してください。

RED HAT VIRTUALIZATION の主要コンポーネント

| コンポーネント名 | 説明 |
|--------------------------------|---|
| Red Hat Virtualization Manager | 環境内のリソースを管理するグラフィカルユーザーインターフェイスと REST API を提供するサービス。Manager は、Red Hat Enterprise Linux を実行する物理マシンまたは仮想マシンにインストールされます。 |
| ホスト | サポートされているホストには、Red Hat Enterprise Linux ホスト (RHEL ホスト) と Red Hat Virtualization Host (イメージベースのハイパーバイザー) の 2 つのタイプがあります。ホストは、Kernel-based Virtual Machine (KVM) テクノロジーを使用して、仮想マシンを実行するためのリソースを提供します。 |
| 共有ストレージ | 仮想マシンに関連付けられたデータの保管に使用するストレージサービス。 |
| Data Warehouse | Manager から設定情報および統計データを収集するサービス。 |

セルフホストエンジンのアーキテクチャー

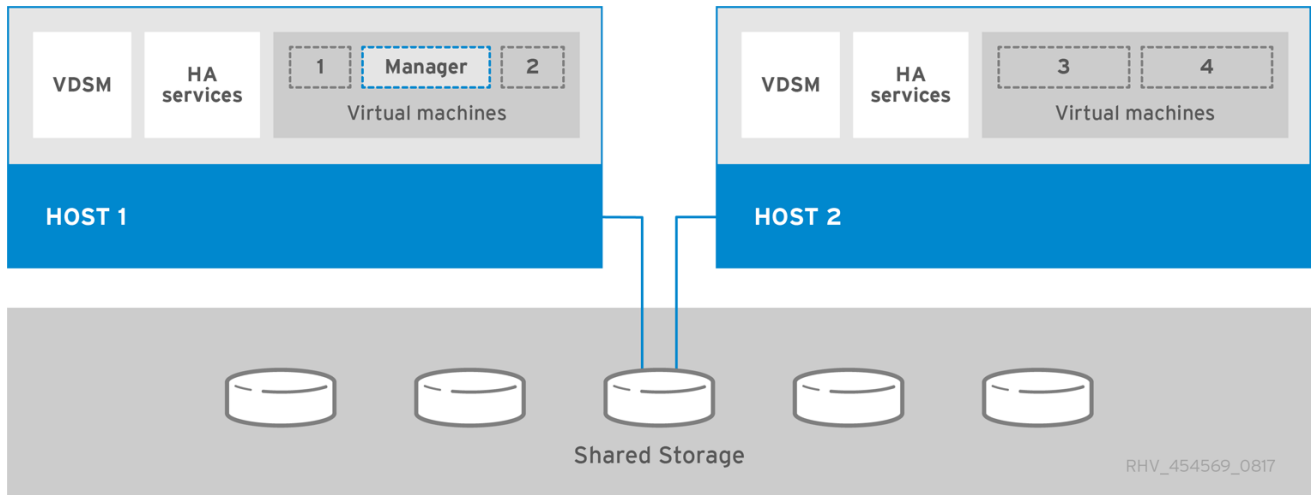
Red Hat Virtualization Manager は、管理している環境と同じ環境内のセルフホストエンジンノード (特化したホスト) で仮想マシンとして実行されます。セルフホストエンジン環境に必要な物理サーバーは 1 台少なくなります。デプロイと管理を行うための管理オーバーヘッドがより高くなります。Manager は、外部の HA 管理を使用せずに高可用性になります。

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1 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 インストールの概要

セルフホストエンジンのインストールには Ansible および RHV-M Appliance (事前設定された Manager 用仮想マシンのイメージ) が使用され、以下のタスクを自動化しています。

- 最初のセルフホストエンジンノードの設定
- そのノードへの Red Hat Enterprise Linux 仮想マシンのインストール
- その仮想マシンへの Red Hat Virtualization Manager のインストールと設定
- セルフホストエンジン用ストレージドメインの設定



注記

RHV-M Appliance が使用されるのはインストール時だけです。Manager のアップグレードには使用されません。

セルフホストエンジン環境をインストールするステップは、以下のとおりです。

1. [セルフホストエンジン用ストレージドメインおよび通常のストレージドメインに使用する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
2. [インストールを実行するデプロイメントホストをインストールします。](#) このホストが最初のセルフホストエンジンノードになります。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
3. [Red Hat Virtualization Manager をインストールおよび設定します。](#)
 - a. [ファイアウォールの有効化および設定](#)
 - b. [デプロイメントホストで `hosted-engine --deploy` コマンドを使用して、セルフホストエンジンをインストールする](#)
 - c. [Manager をコンテンツ配信ネットワークに登録し、Red Hat Virtualization Manager のリポジトリを有効化する](#)
 - d. [管理ポータルに接続し、ホストおよびストレージドメインを追加する](#)
4. [Manager にセルフホストエンジンノードおよび通常のホストをさらに追加します。](#) セルフホストエンジンノードは Manager 用仮想マシンおよびその他の仮想マシンを実行することができます。通常のホストは Manager 用仮想マシンを実行することはできませんが、その他すべての仮想マシンを実行することができます。
 - a. [以下のホストタイプのいずれか、または両方を使用します。](#)

- [Red Hat Virtualization Host](#)
 - [Red Hat Enterprise Linux](#)
- b. [Manager](#) にセルフホストエンジンノードのホストを追加します。
 - c. [Manager](#) に通常のホストを追加します。
5. [Manager](#) にストレージドメインをさらに追加します。セルフホストエンジン用ストレージドメインは、[Manager](#) 用仮想マシンだけが使用することを推奨します。
 6. データベースまたはサービスを [Manager](#) とは別のサーバーでホストする場合には、[インストール](#)の完了後にそれらに移行することができます。



重要

環境を最新の状態に維持してください。詳細は、[How do I update my Red Hat Virtualization system?](#) を参照してください。既知の問題に対するバグ修正が頻繁にリリースされるため、ホストおよび [Manager](#) の更新タスクをスケジュール化してください。

第2章 要件

2.1. RED HAT VIRTUALIZATION MANAGER の要件

2.1.1. ハードウェアの要件

以下に記載するハードウェアの最低要件および推奨要件は、一般的な中小規模のインストールをベースとしています。正確な要件は、デプロイメントの規模や負荷により異なります。

Red Hat Virtualization のハードウェア認定には、Red Hat Enterprise Linux のハードウェア認定が適用されます。詳細は、[Does Red Hat Virtualization also have hardware certification?](#) を参照してください。特定のハードウェア項目が Red Hat Enterprise Linux での使用に認定されているかどうかを確認するには、[Red Hat 認定ハードウェア](#) を参照してください。

表2.1 Red Hat Virtualization Manager ハードウェアの要件

| リソース | 最低要件 | 推奨要件 |
|----------------|---|--|
| CPU | デュアルコア x86_64 CPU | クアッドコア x86_64 CPU または複数のデュアルコア x86_64 CPU |
| メモリー | 利用可能なシステムメモリー 4 GB (Data Warehouse が未インストールで、かつ既存のプロセスによって消費されていないこと) | システムメモリー 16 GB |
| ハードディスク | ディスクの空き容量 25 GB (ローカルアクセス、書き込みが可能であること) | ディスクの空き容量 50 GB (ローカルアクセス、書き込みが可能であること) Manager 履歴データベースのサイズに適したディスク容量を算出するには、 RHV Manager History Database Size Calculator ツールを使用できます。 |
| ネットワークインターフェイス | 1 Gbps 以上の帯域幅のネットワークインターフェイスカード (NIC) 1 基 | 1 Gbps 以上の帯域幅のネットワークインターフェイスカード (NIC) 1 基 |

2.1.2. ブラウザーの要件

管理ポータルと仮想マシンポータルには、以下のブラウザーバージョンとオペレーティングシステムを使用してアクセスすることができます。

ブラウザーのサポートは下記のように階層に分かれます。

- 階層 1: 全面的に検証済みで、完全にサポートされているブラウザーおよびオペレーティングシステムの組み合わせ。Red Hat のエンジニアリングチームは、この階層のブラウザーで問題が発生した場合には、必ず修正を行います。

- 階層 2: 部分的に検証済みで、正常に機能する可能性の高いブラウザとオペレーティングシステムの組み合わせ。この階層のサポートは限定されます。Red Hat のエンジニアリングチームは、この階層のブラウザで問題が発生した場合には、修正を試みます。
- 階層 3: 未検証だが、正常に機能することが予想されるブラウザとオペレーティングシステムの組み合わせ。この階層では、最小限のサポートが提供されます。Red Hat のエンジニアリングチームは、この階層のブラウザ問題が発生した場合には、マイナーな問題に対してのみ修正を試みます。

表2.2 ブラウザーの要件

| サポート階層 | オペレーティングシステムファミリー | ブラウザ |
|--------|--------------------------|---|
| 階層 1 | Red Hat Enterprise Linux | Mozilla Firefox 延長サポート版 (ESR) のバージョン |
| | 任意 | Google Chrome、Mozilla Firefox、または Microsoft Edge の最新バージョン |
| 階層 2 | | |
| 階層 3 | 任意 | Google Chrome または Mozilla Firefox の旧バージョン |
| | 任意 | その他のブラウザ |

2.1.3. クライアントの要件

仮想マシンコンソールは、Red Hat Enterprise Linux および Windows でサポートされている Remote Viewer (**virt-viewer**) クライアントを使用した場合にのみアクセスすることができます。**virt-viewer** をインストールするには、[仮想マシン管理ガイドのクライアントマシンへのコンポーネントのインストール](#)を参照してください。**virt-viewer** のインストールには管理者権限が必要です。

仮想マシンコンソールには、SPICE、VCN、または RDP (Windows のみ) プロトコルを使用してアクセスできます。ゲストオペレーティングシステムに QXLDDOD グラフィカルドライバーをインストールして、SPICE の機能を向上させることができます。SPICE が現在サポートしている最大解像度は 2560 x 1600 ピクセルです。

クライアントオペレーティングシステムの SPICE サポート

サポートされている QXLDDOD ドライバーは、Red Hat Enterprise Linux 7.2 以降および Windows 10 で利用できます。



注記

SPICE は QXLDDOD ドライバーを使用して Windows 8 または 8.1 で動作しますが、認定もテストもされていません。

2.1.4. オペレーティングシステムの要件

Red Hat Virtualization Manager は、Red Hat Enterprise Linux 8.6 のベースインストールにインストールする必要があります。

Manager に必要なパッケージのインストールを試みる際に、依存関係の問題が発生する可能性があるため、ベースのインストール後に他のパッケージをインストールしないでください。

Manager のインストールに必要なリポジトリ以外は有効にしないでください。

2.2. ホストの要件

Red Hat Virtualization のハードウェア認定には、Red Hat Enterprise Linux のハードウェア認定が適用されます。詳細は、[Does Red Hat Virtualization also have hardware certification?](#) を参照してください。特定のハードウェア項目が Red Hat Enterprise Linux での使用に認定されているかどうかを確認するには、[認定ソリューションの検索](#) を参照してください。

ゲストに適用される要件と制限の詳細については、[Red Hat Enterprise Linux テクノロジーの機能と制限](#) および [Supported Limits for Red Hat Virtualization](#) を参照してください。

2.2.1. CPU の要件

すべての CPU が Intel® 64 または AMD64 CPU の拡張機能をサポートし、AMD-V™ または Intel VT® のハードウェア仮想化拡張機能が有効化されている必要があります。No eXecute flag (NX) のサポートも必要です。

以下の CPU モデルがサポートされています。

- AMD
 - Opteron G4
 - Opteron G5
 - EPYC
- Intel
 - Nehalem
 - Westmere
 - SandyBridge
 - IvyBridge
 - Haswell
 - Broadwell
 - Skylake (クライアント)
 - Skylake (サーバー)
 - Cascadelake サーバー

CPU タイプ は、セキュリティー更新のある CPU モデルごとに、基本的なタイプと安全なタイプを一覧表示します。以下に例を示します。

- Intel Cascadelake Server Family
- Secure Intel Cascadelake Server Family

Secure CPU タイプには最新の更新が含まれます。詳細は、BZ#[1731395](#) を参照してください。

2.2.1.1. プロセッサが必要なフラグをサポートしているかどうかのチェック

BIOS で仮想化を有効にする必要があります。この設定を行った後には、ホストの電源をオフにしてから再起動して、変更が適用されるようにします。

手順

1. Red Hat Enterprise Linux または Red Hat Virtualization Host の起動画面で任意のキーを押し、一覧から **Boot** か **Boot with serial console** のエントリーを選択します。
2. **Tab** を押して、選択したオプションのカーネルパラメーターを編集します。
3. 最後のカーネルパラメーターの後にスペースがあることを確認し、パラメーター **rescue** を追加します。
4. **Enter** を押して、レスキューモードで起動します。
5. プロンプトが表示されたら以下のコマンドを実行して、プロセッサに必要な拡張機能があるかどうか、またそれらが有効になっているかどうかを確認します。

```
# grep -E 'svm|vmx' /proc/cpuinfo | grep nx
```

何らかの出力が表示されれば、プロセッサはハードウェアの仮想化に対応しています。出力が何も表示されない場合でも、プロセッサがハードウェアの仮想化に対応している可能性があります。場合によっては、メーカーが BIOS で仮想化拡張機能を無効にしていることがあります。これに該当すると思われる場合には、メーカーが提供しているシステムの BIOS やマザーボードに関するマニュアルを参照してください。

2.2.2. メモリーの要件

必要最小限の RAM は 2 GB です。クラスターレベル 4.2 から 4.5 の場合、Red Hat Virtualization Host で仮想マシンごとにサポートされる最大 RAM は 6 TB です。クラスターレベル 4.6 から 4.7 の場合、Red Hat Virtualization Host で仮想マシンごとにサポートされる最大 RAM は 16 TB です。

ただし、必要な RAM 容量は、ゲストオペレーティングシステムの要件、ゲストのアプリケーションの要件、ゲストのメモリアクティビティと使用状況によって異なります。全ゲストがピークの負荷で同時に稼働しないことを前提とした場合、KVM は仮想ゲストに対して物理 RAM をオーバーコミットし、物理的に存在する RAM を超える要件でゲストをプロビジョニングすることも可能です。KVM は、ゲストが必要とする RAM だけを割り当てて、使用率の低いゲストを swap に移動することによって、オーバーコミットします。

2.2.3. ストレージの要件

ホストには、設定、ログ、カーネルダンプを格納し、swap 領域として使用するためのストレージが必要です。ストレージはローカルまたはネットワークベースとすることができます。Red Hat Virtualization Host (RHVH) は、ネットワークストレージのデフォルト割り当ての1つ、一部、またはすべてを使用して起動することができます。ネットワークストレージから起動する場合、ネットワークの接続が失われるとフリーズする場合があります。ドロップインマルチパス設定ファイルを追加すると、

ネットワーク接続の喪失に対処することができます。SAN ストレージから起動した RHVH がネットワーク接続を失うと、接続が回復するまでファイルは読み取り専用になります。ネットワークストレージを使用すると、パフォーマンスが低下する場合があります。

本セクションでは、RHVH の最低ストレージ要件について説明します。Red Hat Enterprise Linux ホストのストレージ要件は、既存の設定で使用されるディスク容量によって異なりますが、RHVH の要件よりも多くなるはずですが。

ホストのインストールの最低ストレージ要件を以下に示します。ただし、より多くのストレージ領域を利用できるデフォルトの割り当てを使用してください。

- / (root): 6 GB
- /home: 1 GB
- /tmp: 1 GB
- /boot: 1 GB
- /var: 5 GB
- /var/crash: 10 GB
- /var/log: 8 GB
- /var/log/audit: 2 GB
- /var/tmp: 10 GB
- スワップ - 1 GB 詳細は、[What is the recommended swap size for Red Hat platforms?](#) を参照してください。
- Anaconda では、将来のメタデータ拡張用に、ボリュームグループ内のシンプルサイズの 20% が確保されます。これは、通常の使用条件においてデフォルト設定でストレージを使い果たすのを防ぐためです。インストール中のシンプルのオーバープロビジョニングもサポートされていません。
- **最少の合計: 64 GiB**

セルフホストエンジンのインストールに RHV-M Appliance もインストールする場合には、**/var/tmp** は 10 GB 以上である必要があります。

メモリーのオーバーコミットを使用する場合には、すべての仮想マシンに仮想メモリーを提供するのに十分な swap 領域を追加してください。[メモリーの最適化](#) を参照してください。

2.2.4. PCI デバイスの要件

ホストには、1 Gbps 以上の帯域幅のネットワークインターフェイスが少なくとも 1 基搭載されている必要があります。各ホストに 2 つのネットワークインターフェイスを搭載し、そのうちの 1 つは仮想マシンの移行などネットワークへの負荷が高い作業専用にする必要があります。このように負荷の高い操作のパフォーマンスは、利用可能な帯域幅により制限されます。

Intel Q35 ベースの仮想マシンで PCI Express と従来の PCI デバイスを使用する方法に関する情報は、[Using PCI Express and Conventional PCI Devices with the Q35 Virtual Machine](#) を参照してください。

2.2.5. デバイス割り当ての要件

仮想マシンがホストから特定の PCIe デバイスを使用できるように、デバイス割り当ておよび PCI パススルーを実装する予定がある場合は、以下の要件を満たしていることを確認してください。

- CPU が IOMMU (例: VT-d または AMD-Vi) をサポートしていること。IBM POWER8 はデフォルトで IOMMU をサポートしています。
- ファームウェアが IOMMU をサポートしていること。
- 使用する CPU ルートポートが ACS または ACS と同等の機能をサポートしていること。
- PCIe デバイスが ACS または ACS と同等の機能をサポートしていること。
- PCIe デバイスとルートポート間の PCIe スイッチとブリッジはすべて、ACS をサポートしていること。たとえば、スイッチが ACS をサポートしていない場合には、そのスイッチの背後にあるデバイスはすべて同じ IOMMU グループを共有し、同じ仮想マシンにしか割り当てることができません。
- GPU のサポートについては、Red Hat Enterprise Linux 8 は VGA 以外のグラフィックデバイスとして PCIe ベースの NVIDIA K シリーズ Quadro (モデル 2000 シリーズ以降)、GRID、Tesla の PCI デバイス割り当てをサポートしていること。現在、標準のエミュレーションされた VGA インターフェイスの1つ以外に、仮想マシンには GPU を 2 つまでアタッチすることができます。エミュレーションされた VGA は、起動前やインストールに使用され、NVIDIA グラフィックスドライバーが読み込まれると NVIDIA GPU に引き継がれます。NVIDIA Quadro 2000 も、Quadro K420 カードもサポートされていない点にご注意ください。

ベンダーの仕様とデータシートをチェックして、お使いのハードウェアが要件を満たしていることを確認してください。 `lspci -v` コマンドを使用すると、システムにインストールされている PCI デバイスの情報を表示できます。

2.2.6. vGPU の要件

ホスト上の仮想マシンが仮想 GPU を使用するためには、ホストが以下の要件を満たす必要があります。

- GPU が vGPU に対応していること。
- ホストカーネルで GPU が有効であること。
- 適切なドライバーと共に GPU がインストールされていること。
- 仮想マシンの **Administration Portal** の **Host Devices** タブにある **Manage vGPU** ダイアログを使用して、この仮想マシンで使用する vGPU のタイプとインスタンスの数を選択します。
- クラスタ内の各ホストに vGPU に対応したドライバーがインストールされていること。
- vGPU ドライバーと共に vGPU に対応した仮想マシンのオペレーティングシステムがインストールされていること。

2.3. ネットワークの要件

2.3.1. 一般要件

Red Hat Virtualization では、Manager を実行している物理または仮想マシンで IPv6 を有効にしたままにしておく必要があります。お使いのシステムが IPv6 を使用しない場合でも、Manager マシンで IPv6 を無効にしないでください。

2.3.2. セルフホストエンジンデプロイメントのネットワーク範囲

セルフホストエンジンのデプロイメントプロセスは、**192.168** 下の /24 ネットワークアドレスを一時的に使用します。デフォルトは **192.168.222.0/24** で、このアドレスが使用されている場合は、使用されていないアドレスが見つかるまで、**192.168** 下にある他の /24 アドレスを試します。この範囲で未使用のネットワークアドレスが見つからない場合は、デプロイメントに失敗します。

コマンドラインを使用してセルフホストエンジンをインストールする場合は、オプション **--ansible-extra-vars=he_ipv4_subnet_prefix=PREFIX** を使用して、別の /24 ネットワーク範囲を使用するようにデプロイメントスクリプトを設定できます。**PREFIX** はデフォルト範囲の接頭辞に置き換えます。以下に例を示します。

```
# hosted-engine --deploy --ansible-extra-vars=he_ipv4_subnet_prefix=192.168.222
```



注記

コマンドラインで Red Hat Virtualization をセルフホストエンジンとしてインストールすることでのみ、別の範囲を設定することができます。

2.3.3. DNS、NTP、および IPMI フェンシングに対するファイアウォールの要件

以下のトピックに対するファイアウォールの要件は特殊なケースで、個別に検討する必要があります。

DNS および NTP

Red Hat Virtualization では DNS または NTP サーバーは作成されません。したがって、ファイアウォールには、受信トラフィックに対するオープンポートは必要ありません。

デフォルトでは、Red Hat Enterprise Linux は任意のアドレス上の DNS および NTP への出力トラフィックを許可します。出力トラフィックを無効にする場合には、DNS および NTP サーバーに送付されるリクエストに例外を定義します。



重要

- Red Hat Virtualization Manager およびすべてのホスト (Red Hat Virtualization Host および Red Hat Enterprise Linux ホスト) には、完全修飾ドメイン名と、全面的かつ完全な正引きおよび逆引きの名前解決が必要です。
- DNS サービスを Red Hat Virtualization 環境内の仮想マシンとして実行する方法はサポートされていません。Red Hat Virtualization 環境が使用する DNS サービスは、すべて環境の外部でホストする必要があります。
- 名前解決には、**/etc/hosts** ファイルの代わりに DNS を使用します。hosts ファイルを使用すると、より多くの作業が必要となり、誤設定の可能性がより高くなります。

IPMI およびその他のフェンシング機構 (オプション)

IPMI (Intelligent Platform Management Interface) およびその他のフェンシング機構については、ファイアウォールには、受信トラフィックに対するオープンポートは必要ありません。

デフォルトでは、Red Hat Enterprise Linux は任意のアドレス上のポートへの送信 IPMI トラフィックを許可します。発信トラフィックを無効にする場合には、IPMI またはフェンシングサーバーに送付されるリクエストに例外を設定します。

クラスター内の各 Red Hat Virtualization Host および Red Hat Enterprise Linux ホストは、クラスター内にある残りの全ホストのフェンシングデバイスに接続できる必要があります。クラスターホストにエラー（ネットワークエラー、ストレージエラーなど）が発生し、ホストとして機能できない場合は、データセンターの他のホストに接続できる必要があります。

具体的なポート番号は、使用するフェンスエージェントのタイプおよびその設定により異なります。

以降のセクションで説明するファイアウォール要件の表には、このオプションは含まれていません。

2.3.4. Red Hat Virtualization Manager ファイアウォールの要件

Red Hat Virtualization Manager では、ネットワークトラフィックがシステムのファイアウォールを通過できるように複数のポートを開放しておく必要があります。

engine-setup スクリプトは、ファイアウォールを自動的に設定できます。

本セクションに記載するファイアウォール設定は、デフォルトの設定を前提としています。



注記

これらのファイアウォール要件の模式図が、<https://access.redhat.com/articles/3932211> に記載されています。表に書かれた ID を使用して、模式図内の接続を検索できます。

表2.3 Red Hat Virtualization Manager ファイアウォールの要件

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|----|-----|-------|--|--------------------------------|----------------------------------|-----------|
| M1 | - | ICMP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Red Hat Virtualization Manager | オプション 診断に役立つ場合があります。 | × |
| M2 | 22 | TCP | バックエンドの設定やソフトウェアのアップグレードなど、Manager のメンテナンスに使うシステム | Red Hat Virtualization Manager | Secure Shell (SSH) アクセス オプション | ○ |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|----|--------|-------|--|--------------------------------|---|-----------|
| M3 | 2222 | TCP | 仮想マシンのシリアルコンソールにアクセスするクライアント | Red Hat Virtualization Manager | 仮想マシンのシリアルコンソールへの接続を可能にするための Secure Shell (SSH) アクセス。 | ○ |
| M4 | 80、443 | TCP | 管理ポータル のクライアント 仮想マシン ポータルのク ライアント Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト REST API クラ イアント | Red Hat Virtualization Manager | Manager に HTTP (ポート 80、暗号化なし) および HTTPS (ポート 443、暗号化あり) のアクセスを提供します。HTTP は接続を HTTPS にリダイレクトします。 | ○ |
| M5 | 6100 | TCP | 管理ポータル のクライアント 仮想マシン ポータルのク ライアント | Red Hat Virtualization Manager | Manager 上で WebSocket プロキシを実行している場合に、Web ベースのコンソールクライアント (noVNC) に対する Websocket プロキシアクセスを提供します。 | × |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|----|-------|-------|--|---|--|-----------------------------------|
| M6 | 7410 | UDP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Red Hat Virtualization Manager | ホストの Kdump が有効な場合には、Manager の fence_kdump リスナー用にこのポートを開きます。fence_kdump の高度な設定を参照してください。fence_kdump には、接続を暗号化する方法はありません。ただし、このポートは、適していないホストからのアクセスをブロックするように手動で設定できます。 | × |
| M7 | 54323 | TCP | 管理ポータルのクライアント | Red Hat Virtualization Manager (ovirt-imageio サービス) | ovirt-imageio サービスとの通信に必要です。 | ○ |
| M8 | 6642 | TCP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Open Virtual Network (OVN) southbound データベース | Open Virtual Network (OVN) データベースへの接続 | ○ |
| M9 | 9696 | TCP | OVN 用外部ネットワークプロバイダーのクライアント | OVN 用外部ネットワークプロバイダー | OpenStack Networking API | ○ engine-setup によって生成された設定による暗号化。 |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|-----|-------|---------|--------------------------------|---------------------|---|-----------------------------------|
| M10 | 35357 | TCP | OVN 用外部ネットワークプロバイダーのクライアント | OVN 用外部ネットワークプロバイダー | OpenStack Identity API | ○ engine-setup によって生成された設定による暗号化。 |
| M11 | 53 | TCP、UDP | Red Hat Virtualization Manager | DNS サーバー | 1023 より大きいポート番号からポート 53 への DNS ルックアップリクエストおよび応答。デフォルトで開いています。 | × |
| M12 | 123 | UDP | Red Hat Virtualization Manager | NTP サーバー | 1023 より大きいポート番号からポート 123 への NTP リクエストおよび応答。デフォルトで開いています。 | × |



注記

- デフォルトの設定では、OVN northbound データベース (6641) のクライアントは **ovirt-provider-ovn** のみなので、OVN northbound データベースのポート (6641) は記載されていません。両者は同じホスト上で動作しているので、その通信はネットワークには現れません。
- デフォルトでは、Red Hat Enterprise Linux は任意のアドレス上の DNS および NTP への出力トラフィックを許可します。出力トラフィックを無効にする場合には、Manager がリクエストを DNS および NTP サーバーに送信するように例外を設定します。他のノードでも DNS および NTP が必要な場合があります。その際には、それらのノードの要件を確認し、適切にファイアウォールを設定してください。

2.3.5. ホストファイアウォールの要件

Red Hat Enterprise Linux ホストおよび Red Hat Virtualization Host (RHVH) では、ネットワークトラフィックがシステムのファイアウォールを通過できるように複数のポートを開放しておく必要があります。新たなホストを Manager に追加する際に、ファイアウォールルールがデフォルトで自動的に設定され、既存のファイアウォール設定はすべて上書きされます。

新規ホストの追加時のファイアウォール自動設定を無効にするには、**Advanced Parameters** の下の **Automatically configure host firewall** のチェックボックスからチェックを外します。

ホストのファイアウォールルールをカスタマイズするには、[RHV: How to customize the Host's firewall rules?](#) を参照してください。



注記

これらのファイアウォール要件の図は、[Red Hat Virtualization: Firewall Requirements Diagram](#) で入手できます。表に書かれた ID を使用して、模式図内の接続を検索できます。

表2.4 仮想化ホストファイアウォールの要件

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|----|------|-------|--|--|---|-----------|
| H1 | 22 | TCP | Red Hat Virtualization Manager | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Secure Shell (SSH) アクセス オプション | ○ |
| H2 | 2223 | TCP | Red Hat Virtualization Manager | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | 仮想マシンのシリアルコンソールへの接続を可能にするための Secure Shell (SSH) アクセス。 | ○ |
| H3 | 161 | UDP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Red Hat Virtualization Manager | Simple Network Management Protocol (SNMP)。ホストから1つまたは複数の外部 SNMP マネージャーに Simple Network Management Protocol のトラップを送信する場合にのみ必要です。 オプション | × |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|----|-------------|-------|---------------------------------------|--|---|-----------|
| H4 | 111 | TCP | NFS ストレージサーバー | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | NFS 接続オプション | × |
| H5 | 5900 - 6923 | TCP | 管理ポータルのクライアント 仮想マシンポータルのクライアント | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | VNC および SPICE を介したリモートゲストのコンソールアクセス。クライアントが仮想マシンに容易にアクセスできるように、これらのポートは開放しておく必要があります。 | ○(オプション) |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|----|-------|---------|--|--|---|-----------|
| H6 | 5989 | TCP、UDP | Common Information Model Object Manager (CIMOM) | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Common Information Model Object Managers (CIMOM) がホスト上で実行中の仮想マシンをモニタリングするために使用します。このポートは、仮想化環境内の仮想マシンのモニタリングに CIMOM を使用する場合にのみ開放する必要があります。 オプション | × |
| H7 | 9090 | TCP | Red Hat Virtualization Manager クライアントマシン | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Cockpit がインストールされている場合には、Cockpit Web インターフェイスにアクセスするために必要です。 | ○ |
| H8 | 16514 | TCP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | libvirt を使った仮想マシンの移行 | ○ |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|-----|---------------|-------|--|--|--|---------------------------------------|
| H9 | 49152 - 49215 | TCP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | VDSM を使用した仮想マシンの移行とフェンシング。仮想マシンの自動および手動での移行を容易に実行できるように、これらのポートを開放しておく必要があります。 | ○フェンスエージェントに応じて、libvirt を介して移行が行われます。 |
| H10 | 54321 | TCP | Red Hat Virtualization Manager Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | VDSM による Manager およびその他の仮想化ホストとの通信 | ○ |
| H11 | 54322 | TCP | Red Hat Virtualization Manager ovirt-imageio サービス | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | ovirt-imageio サービスとの通信に必要です。 | ○ |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|-----|------|---------|--|--|--|-----------|
| H12 | 6081 | UDP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | Open Virtual Network (OVN) をネットワークプロバイダーとして使用している場合に、OVN がホスト間にトンネルを作成するために必要です。 | × |
| H13 | 53 | TCP、UDP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | DNS サーバー | 1023 より大きいポート番号からポート 53 への DNS ルックアップリクエストおよび応答。このポートは必須で、デフォルトで開いています。 | × |
| H14 | 123 | UDP | Red Hat Virtualization Hosts Red Hat Enterprise Linux ホスト | NTP サーバー | 1023 より大きいポート番号からポート 123 への NTP リクエストおよび応答。このポートは必須で、デフォルトで開いています。 | |
| H15 | 4500 | TCP、UDP | Red Hat Virtualization Hosts | Red Hat Virtualization Hosts | インターネットセキュリティプロトコル (IPSec) | ○ |

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|-----|-----|--------|------------------------------|------------------------------|----------------------------|-----------|
| H16 | 500 | UDP | Red Hat Virtualization Hosts | Red Hat Virtualization Hosts | インターネットセキュリティプロトコル (IPSec) | ○ |
| H17 | - | AH、ESP | Red Hat Virtualization Hosts | Red Hat Virtualization Hosts | インターネットセキュリティプロトコル (IPSec) | ○ |



注記

デフォルトでは、Red Hat Enterprise Linux は任意のアドレス上の DNS および NTP への出力トラフィックを許可します。出力トラフィックを無効にする場合には、Red Hat Virtualization Host に例外を設定します。

Red Hat Enterprise Linux ホストは DNS および NTP サーバーにリクエストを送信します。他のノードでも DNS および NTP が必要な場合があります。その際には、それらのノードの要件を確認し、適切にファイアウォールを設定してください。

2.3.6. データベースサーバーファイアウォールの要件

Red Hat Virtualization では、Manager データベース (**engine**) および Data Warehouse データベース (**ovirt-engine-history**) でのリモートデータベースサーバーの使用をサポートしています。リモートデータベースサーバーを使用する予定がある場合は、Manager および Data Warehouse サービス (Manager と分離することが可能) からの接続を許可する必要があります。

同様に、外部システムからローカルまたはリモートの Data Warehouse データベースにアクセスする予定がある場合は、そのシステムからのアクセスをデータベースで許可する必要があります。



重要

外部システムからの Manager データベースへのアクセスはサポートされていません。



注記

これらのファイアウォール要件の模式図が、<https://access.redhat.com/articles/3932211> に記載されています。表に書かれた ID を使用して、模式図内の接続を検索できます。

表2.5 データベースサーバーファイアウォールの要件

| ID | ポート | プロトコル | 送信元 | 送信先 | 目的 | デフォルトで暗号化 |
|----|------|---------|---|---|------------------------------|----------------------------|
| D1 | 5432 | TCP、UDP | Red Hat Virtualization Manager Data Warehouse サービス | Manager (engine) データベースサーバー Data Warehouse (ovirt-engine-history) データベースサーバー | PostgreSQL データベース接続のデフォルトポート | 無効、ただし有効にできます。 |
| D2 | 5432 | TCP、UDP | 外部のシステム | Data Warehouse (ovirt-engine-history) データベースサーバー | PostgreSQL データベース接続のデフォルトポート | デフォルトでは無効です。無効、ただし有効にできます。 |

2.3.7. 最大伝送単位の要件

デプロイメント中のホストで推奨される最大伝送単位 (MTU) の設定は 1500 です。環境が別の MTU に設定された後で、この設定を更新することができます。MTU 設定の変更に関する詳細は、[How to change the Hosted Engine VM network MTU](#) を参照してください。

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

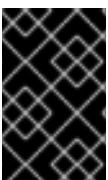
Red Hat Virtualization は、LUN で設定されるボリュームグループから作成されるストレージドメインである iSCSI ストレージをサポートします。ボリュームグループおよび LUN は、いずれも同時に複数のストレージドメインにアタッチすることはできません。

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。
- `user_friendly_names` と `find_multipaths` の設定を上書きしないでください。詳細については、[Recommended Settings for Multipath.conf](#) を参照してください。
- ストレージベンダーから明示的に求められない限り、`no_path_retry` と `polling_interval` の設定を上書きしないでください。詳細については、[Recommended Settings for Multipath.conf](#) を参照してください。



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明確な理由を提供しない限り、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。

第4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。

4.1. 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

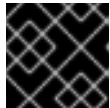
1. Red Hat カスタマーポータル [の Red Hat Virtualization を使い始める](#) にアクセスし、ログインします。
2. **Download Latest** をクリックして、製品のダウンロードページに移動します。
3. 一覧から RHV に適した **Hypervisor Image for RHV** を選択し、**Download Now** をクリックします。
4. RHVH のインストール先となるマシンを起動し、準備したインストールメディアから起動します。
5. 起動メニューから **Install RHVH 4.4** を選択し、**Enter**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

4.1.1. 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. 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 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。

4.2. 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。

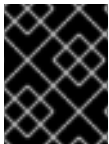
詳細なインストール手順は、[標準的な RHEL インストールの実行](#) を参照してください。

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。

4.2.1. Red Hat Enterprise Linux ホストのリポジトリの有効化

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 \
  --enable=rhel-8-for-x86_64-appstream-tus-rpms \
  --enable=rhel-8-for-x86_64-baseos-tus-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。

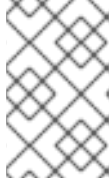
```
# dnf module enable virt:8.3
```

-
- 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](#)することが可能です。

第5章 RED HAT VIRTUALIZATION MANAGER のインストール

5.1. RHV-M APPLIANCE の手動インストール

セルフホストエンジンをデプロイすると、以下の一連のイベントが発生します。

1. インストーラーは RHV-M Appliance をデプロイメントホストにインストールします。
2. Appliance は Manager 用仮想マシンをインストールします。
3. Appliance は Manager 用仮想マシンに Manager をインストールします。

ただし、必要な場合は、事前にデプロイメントホストに Appliance を手動でインストールできます。Appliance は大規模なため、ネットワーク接続の問題により Appliance のインストールに長時間を要するか、または失敗する可能性があります。

手順

1. Red Hat Enterprise Linux ホスト上
 - a. **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
```

- b. 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. インストール済みの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
```

2. RHV-M Appliance を手動でホストにインストールします。

```
# dnf install rhvm-appliance
```

これで、セルフホストエンジンをデプロイすると、インストーラーは Appliance がすでにインストールされていることを検出します。

5.2. ファイアウォールの有効化および設定

セルフホストデプロイメントスクリプトを実行する前に、**firewalld** をインストールして実行する必要があります。インターフェイスが設定されたアクティブなゾーンも必要です。

前提条件

- **firewalld** がインストールされている。**hosted-engine-setup** には **firewalld** パッケージが必要のため、追加の手順を行う必要はありません。

手順

1. **firewalld** を起動します。

```
# systemctl unmask firewalld
# systemctl start firewalld
```

システムの起動時に **firewalld** を確実に自動起動させるには、**root** で以下のコマンドを実行します。

```
# systemctl enable firewalld
```

2. **firewalld** が以下を実行していることを確認します。

```
# systemctl status firewalld
```

3. 以下を実行して、管理インターフェイスがファイアウォールゾーンにあることを確認します。

```
# firewall-cmd --get-active-zones
```

これで、セルフホストエンジンをデプロイする準備が整いました。

5.3. コマンドラインを使用したセルフホストエンジンのデプロイ

コマンドラインからセルフホストエンジンをデプロイすることができます。セットアップパッケージをインストールしてコマンド **hosted-engine --deploy** を実行すると、スクリプトがご自分の環境の情報を収集し、その情報を使用してホストおよび Manager を設定します。

デプロイメント中に、手動で、デプロイメントを一時停止するか、自動化を使用して、Manager 仮想マシンをカスタマイズできます。

- 変数 **he_pause_host** を **true** に設定すると、Manager をインストールし、デプロイメントホストを Manager に追加した後にデプロイメントを一時停止します。
- 変数 **he_pause_before_engine_setup** を **true** に設定すると、Manager のインストール前および **he_restore_from_file** を使用して Manager を復元する前にデプロイメントを一時停止します。



注記

変数 **he_pause_host** または **he_pause_before_engine_setup** を **true** に設定すると、デプロイメントホストの接尾辞が **_he_setup_lock** のロックファイルが **/tmp** に作成されます。その後、必要に応じて仮想マシンを手動でカスタマイズできます。ロックファイルを削除した後、または 24 時間後のいずれか早い方で、デプロイメントが続行されます。

- デプロイメントホスト上の次のディレクトリーのいずれかに Ansible Playbook を追加すると、Playbook が自動的に実行されます。 **/usr/share/ansible/collections/ansible_collections/redhat/rhv/roles/hosted_engine_setup/hooks/** の配下に以下のディレクトリーに Playbook を追加します。
 - **enginevm_before_engine_setup**
 - **enginevm_after_engine_setup**
 - **after_add_host**
 - **after_setup**

前提条件

- **engine-setup** を実行する前に、アプライアンスのコンテンツを最新の製品バージョンにアップグレードしておく。
 - 手動で行うには、**he_pause_before_engine_setup** を使用してデプロイを一時停止し、**dnf update** を実行します。
 - 自動的に行うには、**enginevm_before_engine_setup** フックを適用します。
- Manager およびホスト用の完全修飾ドメイン名 (FQDN)。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。
- 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。
- オプション: 自動化を使用してデプロイ中に Manager 仮想マシンをカスタマイズする場合は、Ansible Playbook を追加する必要があります。[デプロイメント時の自動化を使用したエンジン仮想マシンのカスタマイズ](#) を参照してください。

- セルフホストエンジンセットアップスクリプトでは、エンジン仮想マシンからベアメタルホストの root アカウントへの 2048 ビット RSA キーを使用した ssh 公開キーアクセスが必要です。/etc/ssh/sshd_config で、これらの値を次のように設定する必要があります。
 - **PubkeyAcceptedKeyTypes** は、2048 ビット以上の RSA キーを許可する必要があります。デフォルトでは、この設定はシステム全体の暗号化ポリシーを使用します。詳細については、マニュアルページ **crypto-policies (7)** を参照してください。



注記

4.4.5.5 より前のバージョンで Manager に登録されている RHVH ホストでは、すべてのキーが移行されるまで、下位互換性のために RSA 2048 が必要です。

4.4.5.5 以降に登録された RHVH ホストは、Manager と RHVH の両方でサポートされている最強のアルゴリズムを使用します。**PubkeyAcceptedKeyTypes** 設定は、使用されるアルゴリズムを決定するのに役立ちます。

- **PermitRootLogin** が **without-password** または **yes** に設定されている
- **PubkeyAuthentication** が **yes** に設定されている

手順

1. デプロイメントツールをインストールします。

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

2. 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
```

3. デプロイメントスクリプトを開始します。

```
# hosted-engine --deploy
```

または、デプロイメントホストを Manager に追加した後にデプロイメントを一時停止するには、コマンドラインオプション **--ansible-extra-vars=he_pause_host=true** を使用します。

```
# hosted-engine --deploy --ansible-extra-vars=he_pause_host=true
```



注記

Ctrl+D のキーの組み合わせを使用してデプロイメントを中断すると、スクリプトをいつでも終了することができます。セッションがタイムアウトした場合や接続が中断された場合には、**tmux attach** を実行してデプロイメントセッションを復元します。

4. 確認を要求されたら、**Yes** と入力してデプロイメントを開始します。

```
Continuing will configure this host for serving as hypervisor and will create a local VM with a
running engine.
The locally running engine will be used to configure a new storage domain and create a VM
there.
At the end the disk of the local VM will be moved to the shared storage.
Are you sure you want to continue? (Yes, No)[Yes]:
```

5. ネットワークを設定します。表示されたゲートウェイが正しいことを確認し、**Enter** キーを押します。スクリプトがホストの接続性を確認することができるように、同じサブネット上の ping 送信可能なアドレスを入力します。

```
Please indicate a pingable gateway IP address [X.X.X.X]:
```

6. 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。候補のいずれかを入力するか、**Enter** キーを押してデフォルトをそのまま使用します。

```
Please indicate a nic to set ovirtmgmt bridge on: (ens1, ens0) [ens1]:
```

7. ネットワーク接続の確認方法を指定します。デフォルトは **dns** です。

```
Please specify which way the network connectivity should be checked (ping, dns, tcp, none)
[dns]:
```

ping

ゲートウェイの ping を試行します。

dns

DNS サーバーへの接続を確認します。

TCP

ホストとポートの組み合わせへの TCP 接続を作成します。宛先 IP アドレスとポートを指定する必要があります。接続が正常に作成されると、ネットワークは動作しているとみなされます。指定したホストが、指定したポートで着信 TCP 接続を許可できることを確認します。

none

ネットワークは常に接続されているとみなされます。

8. セルフホストエンジンのホストをデプロイするデータセンターの名前を入力します。デフォルトの名前は **Default** です。

```
Please enter the name of the data center where you want to deploy this hosted-engine host.
Data center [Default]:
```

9. アップグレードガイドのセルフホストエンジンの更新セルフホストエンジンのホストをデプロイするクラスターの名前を入力します。デフォルトの名前は **Default** です。

```
Please enter the name of the cluster where you want to deploy this hosted-engine host.
Cluster [Default]:
```

10. 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。
11. カスタム RHV-M アプライアンスアプライアンスイメージを使用してデプロイするには、OVA アーカイブへのパスを指定します。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。

If you want to deploy with a custom engine appliance image, please specify the path to the OVA archive you would like to use.

Entering no value will use the image from the rhvm-appliance rpm, installing it if needed.
Appliance image path []:

12. Manager 用仮想マシンの CPU およびメモリー設定を入力します。

Please specify the number of virtual CPUs for the VM. The default is the appliance OVF value [4]:

Please specify the memory size of the VM in MB. The default is the maximum available [6824]:

13. **manager.example.com** などの Manager 用仮想マシンの FQDN を指定します。

Please provide the FQDN you would like to use for the engine.

Note: This will be the FQDN of the engine VM you are now going to launch, it should not point to the base host or to any other existing machine.

Engine VM FQDN []:

14. Manager 用仮想マシンのドメインを指定します。たとえば、FQDN が **manager.example.com** の場合、**example.com** を入力します。

Please provide the domain name you would like to use for the engine appliance.

Engine VM domain: [example.com]

15. Manager の root パスワードを作成し、確認のために同じパスワードを再入力します。

Enter root password that will be used for the engine appliance:

Confirm appliance root password:

16. オプションとして、パスワードを入力せずに Manager 用仮想マシンに root ユーザーとしてログインできるように 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。

You may provide an SSH public key, that will be added by the deployment script to the authorized_keys file of the root user in the engine appliance.

This should allow you passwordless login to the engine machine after deployment.

If you provide no key, authorized_keys will not be touched.

SSH public key []:

Do you want to enable ssh access for the root user (yes, no, without-password) [yes]:

17. オプション: Manager 仮想マシンに DISA STIG セキュリティープロファイルを適用できます。DISA STIG プロファイルは、デフォルトの OpenSCAP プロファイルです。

Do you want to apply a default OpenSCAP security profile? (Yes, No) [No]:

-
18. 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。

You may specify a unicast MAC address for the VM or accept a randomly generated default [00:16:3e:3d:34:47]:

19. Manager 用仮想マシンのネットワークの詳細を入力します。

How should the engine VM network be configured (DHCP, Static)[DHCP]?

Static を指定した場合には、Manager 用仮想マシンの 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。
- IPv6 の場合、Red Hat Virtualization でサポートされるのは静的アドレスのみです。

Please enter the IP address to be used for the engine VM [x.x.x.x]:

Please provide a comma-separated list (max 3) of IP addresses of domain name servers for the engine VM

Engine VM DNS (leave it empty to skip):

20. 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。

Add lines for the appliance itself and for this host to `/etc/hosts` on the engine VM?

Note: ensuring that this host could resolve the engine VM hostname is still up to you.

Add lines to `/etc/hosts`? (Yes, No)[Yes]:

21. 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。あるいは、**Enter** キーを押してデフォルトをそのまま使用します。

Please provide the name of the SMTP server through which we will send notifications [localhost]:

Please provide the TCP port number of the SMTP server [25]:

Please provide the email address from which notifications will be sent [root@localhost]:

Please provide a comma-separated list of email addresses which will get notifications [root@localhost]:

22. 管理ポータルにアクセスするための `admin@internal` ユーザーのパスワードを作成し、確認のために再度入力します。

Enter engine admin password:

Confirm engine admin password:

23. デプロイメントホストのホスト名を指定します。

```
Please provide the hostname of this host on the management network
[hostname.example.com]:
```

スクリプトにより仮想マシンが作成されます。デフォルトでは、スクリプトは最初に RHV-M アプライアンスをダウンロードしてインストールします。これにより、インストール時間が長くなります。

24. (オプション): 変数 `he_pause_host: true` を設定すると、デプロイメントホストを Manager に追加した後にデプロイメントを一時停止します。これで、デプロイメントホストから Manager 仮想マシンにログインしてカスタマイズできます。Manager の FQDN または IP アドレスのいずれかを使用してログインできます。たとえば、Manager の FQDN が **manager.example.com** の場合には以下のコマンドを実行します。

```
$ ssh root@manager.example.com
```

ヒント

インストールログでは、IP アドレスは `local_vm_ip` にあります。インストールログは、`/var/log/ovirt-hosted-engine-setup/ovirt-hosted-engine-setup-ansible-bootstrap_local_vm*` の最新のインスタンスです。

- a. 必要に応じて、Manager 仮想マシンをカスタマイズします。
 - b. 完了したら、Manager FQDN を使用してブラウザを使用して管理ポータルにログインし、ホストの状態が **Up** であることを確認します。
 - c. ロックファイルを削除すると、展開スクリプトが自動的に続行され、Manager 仮想マシンが設定されます。
25. 使用するストレージのタイプを選択します。

```
Please specify the storage you would like to use (glusterfs, iscsi, fc, nfs)[nfs]:
```

- 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。

```
Please specify the nfs version you would like to use (auto, v3, v4, v4_1)[auto]:
Please specify the full shared storage connection path to use (example: host:/path):
storage.example.com:/hosted_engine/nfs
If needed, specify additional mount options for the connection to the hosted-engine
storage domain []:
```

- 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。

Please specify the iSCSI portal IP address:
 Please specify the iSCSI portal port [3260]:
 Please specify the iSCSI discover user:
 Please specify the iSCSI discover password:
 Please specify the iSCSI portal login user:
 Please specify the iSCSI portal login password:

The following targets have been found:
 [1] iqn.2017-10.com.redhat.example:he
 TPGT: 1, portals:
 192.168.1.xxx:3260
 192.168.2.xxx:3260
 192.168.3.xxx:3260

Please select a target (1) [1]: 1

The following luns have been found on the requested target:
 [1] 360003ff44dc75adcb5046390a16b4beb 199GiB MSFT Virtual HD
 status: free, paths: 1 active

Please select the destination LUN (1) [1]:

- 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
```

Please specify the full shared storage connection path to use (example: host:/path):
storage.example.com:/hosted_engine/gluster_volume
 If needed, specify additional mount options for the connection to the hosted-engine storage domain []:

- 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

[Reusing LUNs](#) を参照してください。

The following luns have been found on the requested target:

```
[1] 3514f0c5447600351 30GiB XtremIO XtremApp
    status: used, paths: 2 active
```

```
[2] 3514f0c5447600352 30GiB XtremIO XtremApp
    status: used, paths: 2 active
```

Please select the destination LUN (1, 2) [1]:

26. Manager 用仮想マシンのディスクサイズを入力します。

Please specify the size of the VM disk in GB: [50]:

デプロイメントが正常に完了すると、1つのデータセンター、クラスター、ホスト、ストレージドメイン、および Manager 用仮想マシンがすでに稼働しているはずです。管理ポータルにログインして、その他のリソースを追加することができます。

27. オプション: 追加のユーザーを環境に追加できるように Red Hat Single Sign On をインストールして設定します。詳細は、[管理ガイドの Red Hat Single Sign-On のインストールおよび設定](#) を参照してください。
28. オプション: Grafana をデプロイして、RHV 環境からのレポートをモニターおよび表示できるようにします。詳細は、[管理ガイドの Configuring Grafana](#) を参照してください。

管理ポータルで、Manager 用仮想マシン、仮想マシンを実行しているホスト、およびセルフホストエンジン用ストレージドメインに金色の王冠のフラグが付けられます。



注記

Manager の I/O スケジューラーと、Manager の並べ替え I/O 要求をホストするハイパーバイザーの両方。この二重の並べ替えは、ストレージレイヤーへの I/O 要求を遅らせ、パフォーマンスに影響を与える可能性があります。

データセンターによっては、I/O スケジューラーを **none** に変更すると、パフォーマンスを向上させることができます。詳細については、RHEL の [Monitoring and managing system status and performance](#) の [Available disk schedulers](#) を参照してください。

次のステップは、Red Hat Virtualization Manager のリポジトリを有効にすることです。

5.4. 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のユ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 \
  --enable=rhel-8-for-x86_64-appstream-tus-rpms \
  --enable=rhel-8-for-x86_64-baseos-tus-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. **アップグレードガイド** の **セルフホストエンジンの更新** の手順を使用して、セルフホストエンジンを更新します。

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](#)

次に管理ポータルにログインします。ここで、環境にホストおよびストレージを追加することができます。

5.5. 管理ポータルへの接続

Web ブラウザーを使って管理ポータルへアクセスします。

1. Web ブラウザーで、**https://manager-fqdn/ovirt-engine** に移動します。**manager-fqdn** は、インストール時に指定した FQDN に置き換えます。



注記

別のホスト名または IP アドレスを使用して、管理ポータルにアクセスすることができます。これには、**/etc/ovirt-engine/engine.conf.d/** に設定ファイルを追加する必要があります。以下に例を示します。

```
# vi /etc/ovirt-engine/engine.conf.d/99-custom-ss0-setup.conf
SSO_ALTERNATE_ENGINE_FQDNS="alias1.example.com
alias2.example.com"
```

代替ホスト名の一覧は、スペースで区切る必要があります。また、Manager の IP アドレスを一覧に追加することもできますが、DNS で解決可能なホスト名の代わりに IP アドレスを使用することは推奨していません。

2. **Administration Portal** をクリックすると、SSO ログインページが表示されます。SSO ログインにより、管理ポータルと仮想マシンポータルに同時にログインすることができます。
3. **ユーザー名** と **パスワード** を入力します。初回ログインの場合は、ユーザー名 **admin** とインストール時に指定したパスワードを使用してください。
4. 認証する **Domain** を選択します。内部の **admin** ユーザー名を使用してログインする場合は、プロファイルに **internal** を選択します。
5. **Log In** をクリックします。

6. 管理ポータルは複数の言語で表示することができます。デフォルトの選択は、お使いの Web ブラウザーのロケール設定をベースに決定されます。デフォルト以外の言語で管理ポータルを表示する場合は、ウェルカムページのドロップダウンリストから任意の言語を選択してください。

Red Hat Virtualization 管理ポータルからログアウトするには、ヘッダーバーでユーザー名をクリックして、**Sign Out** をクリックします。すべてのポータルからログアウトされ、Manager のウェルカム画面が表示されます。

第6章 RED HAT VIRTUALIZATION 用ホストのインストール

Red Hat Virtualization は、[Red Hat Virtualization Hosts \(RHVH\)](#) および [Red Hat Enterprise Linux ホスト](#) の2つのタイプのホストをサポートしています。環境に応じて、1タイプのみまたは両方のタイプを使用することができます。移行や高可用性などの機能を利用するには、少なくとも2台のホストが必要です。

ネットワーク情報については、[ホストネットワーク設定の推奨プラクティス](#) を参照してください。



重要

SELinux はインストール時に enforcing モードに設定されます。確認するには、**getenforce** を実行します。Red Hat Virtualization 環境をサポートするには、すべてのホストと Manager で SELinux を enforcing モードに設定する必要があります。

表6.1 ホストタイプ

| ホストタイプ | 別名 | 説明 |
|------------------------------|-----------------|--|
| Red Hat Virtualization Host | RHVH、シンホスト | Red Hat Enterprise Linux をベースとする最小限のオペレーティングシステム。カスタマーポータルから ISO ファイルとして配布され、マシンがホストとして機能するためのパッケージのみが含まれています。 |
| Red Hat Enterprise Linux ホスト | RHEL ホスト、シックホスト | 適切なサブスクリプションがアタッチされた Red Hat Enterprise Linux システムは、ホストとして使用することができます。 |

ホストの互換性

新規データセンターの作成時に、互換バージョンを設定することができます。データセンター内の全ホストに適した互換バージョンを選択します。一旦設定されると、それよりも古いバージョンに変更することはできません。Red Hat Virtualization を新規インストールした場合には、最新の互換バージョンがデフォルトのデータセンターとデフォルトのクラスターに設定されるので、それ以前の互換バージョンを使用するには、追加でデータセンターおよびクラスターを作成する必要があります。互換バージョンに関する詳細は、[Red Hat Virtualization のライフサイクル](#) の [Red Hat Virtualization Manager の互換性](#) を参照してください。

6.1. RED HAT VIRTUALIZATION HOSTS

6.1.1. 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。

6.1.2. 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. 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ユーザー名とパスワードを入力します。

```
# subscription-manager register
```

2. **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。

6.1.3. 高度なインストール

6.1.3.1. カスタムパーティション設定

Red Hat Virtualization Host (RHVH) でのカスタムパーティション設定は推奨されません。**Installation Destination** ウィンドウの **Automatically configure partitioning** オプションを使用してください。

インストールでカスタムのパーティション設定が必要な場合は、インストール時に **I will configure partitioning** オプションを選択します。ただし、以下の制限が適用される点に注意してください。

- **Manual Partitioning** ウィンドウで、デフォルトの **LVM Thin Provisioning** オプションを選択する必要があります。
- 以下のディレクトリが必要で、シンプロビジョニングされた論理ボリューム上になければなりません。
 - root (/)
 - /home
 - /tmp
 - /var
 - /var/crash/
 - /var/log

◦ /var/log/audit



重要

/usr 用に別のパーティションを作成しないでください。別のパーティションを作成すると、インストールに失敗します。

/usr は、RHVH と共にバージョンを変更できる論理ボリューム上になければなりません。したがって、root (/) 上に残す必要があります。

各パーティションに必要なストレージサイズについては、[Storage Requirements](#) を参照してください。

- /boot ディレクトリーは、標準のパーティションとして定義する必要があります。
- /var ディレクトリーは、別のボリュームまたはディスク上になければなりません。
- XFS または Ext4 ファイルシステムのみがサポートされます。

キックスタートファイルでの手動パーティション設定の定義

以下の例は、キックスタートファイルでパーティションを手動設定する方法を示しています。

```
clearpart --all
part /boot --fstype xfs --size=1000 --ondisk=sda
part pv.01 --size=42000 --grow
volgroup HostVG pv.01 --reserved-percent=20
logvol swap --vgname=HostVG --name=swap --fstype=swap --recommended
logvol none --vgname=HostVG --name=HostPool --thinpool --size=40000 --grow
logvol / --vgname=HostVG --name=root --thin --fstype=ext4 --poolname=HostPool --
fsoptions="defaults,discard" --size=6000 --grow
logvol /var --vgname=HostVG --name=var --thin --fstype=ext4 --poolname=HostPool
--fsoptions="defaults,discard" --size=15000
logvol /var/crash --vgname=HostVG --name=var_crash --thin --fstype=ext4 --poolname=HostPool --
fsoptions="defaults,discard" --size=10000
logvol /var/log --vgname=HostVG --name=var_log --thin --fstype=ext4 --poolname=HostPool --
fsoptions="defaults,discard" --size=8000
logvol /var/log/audit --vgname=HostVG --name=var_audit --thin --fstype=ext4 --poolname=HostPool --
-fsoptions="defaults,discard" --size=2000
logvol /home --vgname=HostVG --name=home --thin --fstype=ext4 --poolname=HostPool --
fsoptions="defaults,discard" --size=1000
logvol /tmp --vgname=HostVG --name=tmp --thin --fstype=ext4 --poolname=HostPool --
fsoptions="defaults,discard" --size=1000
```



注記

logvol --thinpool --grow を使用する場合は、シンプールの拡張のために、**volgroup --reserved-space** または **volgroup --reserved-percent** のボリュームグループに領域を確保する必要があります。

6.1.3.2. インストーラーのサポートなしでホストに DUD ドライバーをインストールする

RHVH のデフォルト設定でサポートされていないハードウェア RAID デバイスを使用する場合など、Red Hat Virtualization Host (RHVH) のインストールにドライバー更新ディスク (DUD) が必要な場合が

あります。Red Hat Enterprise Linux ホストとは対照的に、RHVH は DUD の使用を完全にはサポートしていません。その後、ホストは RAID を認識しないため、インストール後に正常に起動できません。代わりに、緊急モードで起動します。

出力例:

```
Warning: /dev/test/rhvh-4.4-20210202.0+1 does not exist
Warning: /dev/test/swap does not exist
Entering emergency mode. Exit the shell to continue.
```

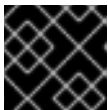
このような場合、インストールを完了する前に手動でドライバーを追加できます。

前提条件

- RHVH をインストールするマシン
- DUD
- DUD と RHVH に USB ドライブを使用している場合は 2 つ以上の使用可能な USB ポート

手順

1. DUD をホストマシンにロードします。
2. RHVH をインストールします。コマンドラインを使用したセルフホストエンジンの [Red Hat Virtualization のインストール](#) の [Red Hat Virtualization Hosts のインストール](#) を参照してください。



重要

インストールが完了したら、システムを再起動しないでください。

ヒント

SSH を使用して DUD にアクセスする場合は、次の手順を実行します。

- 文字列 **inst.sshd** をカーネルコマンドラインに追加します。

```
<kernel_command_line> inst.sshd
```

- インストール中にネットワークを有効にします。
3. **Ctrl + Alt + F3** を押して、コンソールモードに入ります。または、SSH を使用して接続することもできます。
 4. DUD をマウントします。

```
# mkdir /mnt/dud
# mount -r /dev/<dud_device> /mnt/dud
```

5. DUD 内の RPM ファイルをターゲットマシンのディスクにコピーします。

```
# cp /mnt/dud/rpms/<path>/<rpm_file>.rpm /mnt/sysroot/root/
```

以下に例を示します。

```
# cp /mnt/dud/rpms/x86_64/kmod-3w-9xxx-2.26.02.014-5.el8_3.elrepo.x86_64.rpm
/mnt/sysroot/root/
```

6. root ディレクトリーを **/mnt/sysroot/** に変更します。

```
# chroot /mnt/sysroot
```

7. 現在の initrd イメージをバックアップします。以下に例を示します。

```
# cp -p /boot/initramfs-4.18.0-240.15.1.el8_3.x86_64.img /boot/initramfs-4.18.0-
240.15.1.el8_3.x86_64.img.bck1
# cp -p /boot/rhvh-4.4.5.1-0.20210323.0+1/initramfs-4.18.0-240.15.1.el8_3.x86_64.img
/boot/rhvh-4.4.5.1-0.20210323.0+1/initramfs-4.18.0-240.15.1.el8_3.x86_64.img.bck1
```

8. 以前に作成したコピーからドライバーの RPM ファイルをインストールします。
以下に例を示します。

```
# dnf install /root/kmod-3w-9xxx-2.26.02.014-5.el8_3.elrepo.x86_64.rpm
```



注記

インストールされた環境で再起動すると、このパッケージはシステムに表示されなくなるため、たとえば **initramfs** を再構築するために必要な場合は、そのパッケージをもう一度インストールする必要があります。インストールすると、パッケージは残ります。

dnf を使用してホストを更新する場合、ドライバーの更新は保持されるため、このプロセスを繰り返す必要はありません。

ヒント

インターネットに接続していない場合は、**dnf** の代わりに **rpm** コマンドを使用してください。

```
# rpm -ivh /root/kmod-3w-9xxx-2.26.02.014-5.el8_3.elrepo.x86_64.rpm
```

9. 新しいイメージを作成し、ドライバーを強制的に追加します。

```
# dracut --force --add-drivers <module_name> --kver <kernel_version>
```

以下に例を示します。

```
# dracut --force --add-drivers 3w-9xxx --kver 4.18.0-240.15.1.el8_3.x86_64
```

10. 結果を確認します。新しいイメージは大きくなり、ドライバーが含まれているはずですが、たとえば、元のバックアップされたイメージファイルと新しいイメージファイルのサイズを比較します。
この例では、新しいイメージファイルは 88739013 バイトであり、元の 88717417 バイトよりも大きくなっています。

```
# ls -ltr /boot/initramfs-4.18.0-240.15.1.el8_3.x86_64.img*
-rw-----. 1 root root 88717417 Jun  2 14:29 /boot/initramfs-4.18.0-
240.15.1.el8_3.x86_64.img.bck1
-rw-----. 1 root root 88739013 Jun  2 17:47 /boot/initramfs-4.18.0-
240.15.1.el8_3.x86_64.img
```

イメージファイルの一部として、新しいドライバーが含まれている必要があります。たとえば、3w-9xxx モジュールが含まれている必要があります。

```
# lsinitrd /boot/initramfs-4.18.0-240.15.1.el8_3.x86_64.img | grep 3w-9xxx
drwxr-xr-x  2 root  root      0 Feb 22 15:57 usr/lib/modules/4.18.0-
240.15.1.el8_3.x86_64/weak-updates/3w-9xxx
lrwxrwxrwx  1 root  root      55 Feb 22 15:57 usr/lib/modules/4.18.0-
240.15.1.el8_3.x86_64/weak-updates/3w-9xxx/3w-9xxx.ko-../../4.18.0-
240.el8.x86_64/extra/3w-9xxx/3w-9xxx.ko
drwxr-xr-x  2 root  root      0 Feb 22 15:57 usr/lib/modules/4.18.0-
240.el8.x86_64/extra/3w-9xxx
-rw-r--r--  1 root  root    80121 Nov 10 2020 usr/lib/modules/4.18.0-
240.el8.x86_64/extra/3w-9xxx/3w-9xxx.ko
```

11. インストールするレイヤーで使用されるカーネルを含む **/boot** の下のディレクトリーにイメージをコピーします。以下はその例です。

```
# cp -p /boot/initramfs-4.18.0-240.15.1.el8_3.x86_64.img /boot/rhvh-4.4.5.1-
0.20210323.0+1/initramfs-4.18.0-240.15.1.el8_3.x86_64.img
```

12. chroot を終了します。
13. シェルを終了します。
14. **Ctrl + Alt + F3** を使用して仮想端末にアクセスした場合は、**Ctrl + Alt + F_<n>_** (通常は **F1** または **F5**) を押してインストーラーに戻ります。
15. インストーラー画面で、再起動します。

検証

マシンは正常に再起動するはずです。

6.1.3.3. Red Hat Virtualization Host デプロイメントの自動化

物理メディアデバイスなしに Red Hat Virtualization Host (RHVH) をインストールすることができます。そのためには、インストールの質問に対する回答が含まれたキックスタートファイルを使用し、ネットワーク経由で PXE サーバーから起動します。



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。

RHVH は Red Hat Enterprise Linux とほぼ同じ方法でインストールされます。キックスタートファイルを使用して PXE サーバーからインストールする手順については、Red Hat Enterprise Linux インストールガイドの [キックスタートを使ったインストール](#) を参照してください。RHVH に固有の手順 (Red Hat Satellite を使用した RHVH のデプロイメントを例として使用) については、この後に説明します。

RHVH の自動デプロイメントは、以下の 3 つのステージで設定されます。

- [インストール環境の準備](#)
- [PXE サーバーおよびブートローダーの設定](#)
- [キックスタートファイルの作成と実行](#)

6.1.3.3.1. インストール環境の準備

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 ISO イメージをネットワーク経由で提供できるようにします。[Red Hat Enterprise Linux インストールガイドの インストールソース - ネットワーク](#) を参照してください。
5. RHVH ISO から `squashfs.img` ハイパーバイザーイメージファイルを抽出します。

```
# mount -o loop /path/to/RHVH-ISO /mnt/rvh
# cp /mnt/rvh/Packages/redhat-virtualization-host-image-update* /tmp
# cd /tmp
# rpm2cpio redhat-virtualization-host-image-update* | cpio -idmv
```



注記

`/tmp/usr/share/redhat-virtualization-host/image/` ディレクトリーにある `squashfs.img` ファイルの名前は `redhat-virtualization-host-version_number_version.squashfs.img` です。物理マシンにインストールするためのハイパーバイザーイメージが含まれます。Anaconda **inst.stage2** オプションで使用される `/LiveOS/squashfs.img` ファイルと混同しないでください。

6.1.3.3.2. PXE サーバーおよびブートローダーの設定

1. PXE サーバーを設定します。Red Hat Enterprise Linux インストールガイドの [ネットワークからのインストールの準備](#) を参照してください。
2. RHVH 起動イメージを `/tftpboot` ディレクトリーにコピーします。

```
# cp mnt/rhvh/images/pxeboot/{vmlinuz,initrd.img} /var/lib/tftpboot/pxelinux/
```

3. ブートローダー設定で RHVH 起動イメージを指定して、`rhvh` ラベルを作成します。

```
LABEL rhvh
MENU LABEL Install Red Hat Virtualization Host
KERNEL /var/lib/tftpboot/pxelinux/vmlinuz
APPEND initrd=/var/lib/tftpboot/pxelinux/initrd.img inst.stage2=URL/to/RHVH-ISO
```

Red Hat Satellite の RHVH ブートローダー設定の例

Red Hat Satellite からの情報を使用してホストをプロビジョニングする場合には、グローバルまたはホストグループレベルのパラメーターを作成し (ここでは `rhvh_image`)、ISO をマウントまたは抽出するディレクトリーの URL を定義する必要があります。

```
< %#
kind: PXELinux
name: RHVH PXELinux
%>
# Created for booting new hosts
#

DEFAULT rhvh

LABEL rhvh
KERNEL <%= @kernel %>
APPEND initrd=<%= @initrd %> inst.ks=<%= foreman_url("provision") %> inst.stage2=<%=
@host.params["rhvh_image"] %> intel_iommu=on console=tty0 console=ttyS1,115200n8
ssh_pwauth=1 local_boot_trigger=<%= foreman_url("built") %>
IPAPPEND 2
```

4. RHVH ISO の内容をローカルで利用可能な状態にし、たとえば HTTPD サーバーを使用して、ネットワークにエクスポートします。

```
# cp -a /mnt/rhvh/ /var/www/html/rhvh-install
# curl URL/to/RHVH-ISO/rhvh-install
```

6.1.3.3.3. キックスタートファイルの作成と実行

1. キックスタートファイルを作成し、ネットワーク経由で提供できるようにします。Red Hat Enterprise Linux インストールガイドの [キックスタートを使ったインストール](#) を参照してください。
2. キックスタートファイルは以下に示す RHV 固有の要件を満たす必要があります。
 - RHVH に `%packages` セクションは必要ありません。代わりに、`liveimg` オプションを使用して、RHVH ISO イメージから `redhat-virtualization-host-version_number_version.squashfs.img` ファイルを指定します。

```
liveimg --url=example.com/tmp/usr/share/redhat-virtualization-host/image/redhat-
virtualization-host-version_number_version.squashfs.img
```

- 自動パーティション設定は強く推奨されますが、使用に際して注意が必要です。最初にローカルディスクが検出されることを確認し、**ignoredisk** コマンドを追加して **sda** のようにローカルディスクが無視するように指定します。特定のドライブを使用する場合、Red Hat は、**ignoredisk --only-use=/dev/disk/<path>** または **ignoredisk --only-use=/dev/disk/<ID>** の使用を推奨します。

```
autopart --type=thinp
ignoredisk --only-use=sda
ignoredisk --only-use=/dev/disk/<path>
ignoredisk --only-use=/dev/disk/<ID>
```



注記

自動パーティション設定には、シンプロビジョニングが必要です。

/home は必須のディレクトリーであるため、RHVH では **--no-home** オプションは機能しません。

インストールで手動パーティション設定が必要な場合は、パーティション設定に適用される制限の一覧およびキックスタートファイルでの手動パーティション設定の例を [カスタムパーティション設定](#) で確認してください。

- **nodedctl init** コマンドを呼び出す **%post** セクションが必要です。

```
%post
nodedctl init
%end
```



注記

nodedctl init コマンドが **%post** セクションの最後で、再起動コードがある場合はその前であることを確認してください。

RHVH を独自にデプロイするためのキックスタート例

このキックスタートの例では、RHVH のデプロイ方法を示しています。必要に応じて、コマンドとオプションをさらに追加してください。



警告

この例では、すべてのディスクが空であり、初期化可能であることを前提としています。データを含むディスクが割り当てられている場合は、そのディスクを削除するか、**ignoredisks** プロパティーに追加します。

```

liveimg --url=http://FQDN/tmp/usr/share/redhat-virtualization-host/image/redhat-
virtualization-host-version_number_version.squashfs.img
clearpart --all
autopart --type=thinp
rootpw --plaintext ovirt
timezone --utc America/Phoenix
zerombr
text

reboot

%post --erroronfail
nodectl init
%end

```

Satellite から登録およびネットワーク設定を使用した RHVH をデプロイするためのキックスタートの例

このキックスタートの例では、Red Hat Satellite からの情報を使用してホストネットワークを設定し、ホストを Satellite サーバーに登録します。グローバルまたはホストグループレベルのパラメーターを作成し (ここでは **rhvh_image**)、**squashfs.img** ファイルを格納するディレクトリーの URL を定義する必要があります。**ntp_server1** もグローバルまたはホストグループレベルの変数です。



警告

この例では、すべてのディスクが空であり、初期化可能であることを前提としています。データを含むディスクが割り当てられている場合は、そのディスクを削除するか、**ignoredisks** プロパティーに追加します。

```

<%#
kind: provision
name: RHVH Kickstart default
oses:
- RHVH
%>
install
liveimg --url=<%= @host.params['rhvh_image'] %>squashfs.img

network --bootproto static --ip=<%= @host.ip %> --netmask=<%= @host.subnet.mask
%> --gateway=<%= @host.subnet.gateway %> --nameserver=<%=
@host.subnet.dns_primary %> --hostname <%= @host.name %>

zerombr
clearpart --all
autopart --type=thinp

rootpw --iscrypted <%= root_pass %>

```

```
# installation answers
lang en_US.UTF-8
timezone <%= @host.params['time-zone'] || 'UTC' %>
keyboard us
firewall --service=ssh
services --enabled=sshd

text
reboot

%post --log=/root/ks.post.log --erroronfail
nodectl init
<%= snippet 'subscription_manager_registration' %>
<%= snippet 'kickstart_networking_setup' %>
/usr/sbin/ntpdate -sub <%= @host.params['ntp_server1'] || '0.fedora.pool.ntp.org' %>
/usr/sbin/hwclock --systohc

/usr/bin/curl <%= foreman_url('built') %>

sync
systemctl reboot
%end
```

3. キックスタートファイルの場所を、PXE サーバーのブートローダー設定ファイルに追加します。

```
APPEND initrd=/var/tftpboot/pxelinux/initrd.img inst.stage2=URL/to/RHVH-ISO
inst.ks=URL/to/RHVH-ks.cfg
```

4. [Red Hat Enterprise Linux インストールガイドの PXE を使ったネットワークからの起動](#) に記載された手順に従って、RHVH をインストールします。

6.2. RED HAT ENTERPRISE LINUX ホスト

6.2.1. 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。

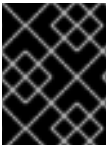


警告

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。

6.2.2. 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
```

有効なリポジトリー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. リポジトリーを設定します。

```
# 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 \  
--enable=rhel-8-for-x86_64-appstream-tus-rpms \  
--enable=rhel-8-for-x86_64-baseos-tus-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. 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
```

7. 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。

6.2.3. Red Hat Enterprise Linux ホストへの Cockpit のインストール

ホストのリソースの監視および管理タスクの実施のために、Cockpit をインストールすることができます。

手順

1. dashboard パッケージをインストールします。

```
# dnf install cockpit-ovirt-dashboard
```

2. **cockpit.socket** サービスを有効にして起動します。

```
# systemctl enable cockpit.socket  
# systemctl start cockpit.socket
```

3. ファイアウォールで Cockpit がアクティブなサービスかどうかを確認します。

```
# firewall-cmd --list-services
```

cockpit のリストが表示されるはずですが、表示されない場合には、root 権限で以下のコマンドを入力し、**cockpit** をサービスとしてファイアウォールに追加します。

```
# firewall-cmd --permanent --add-service=cockpit
```

--permanent オプションは、再起動後も **cockpit** サービスをアクティブな状態で維持します。

<https://HostFQDNorIP:9090> で Cockpit Web インターフェイスにログインできます。

6.3. ホストネットワーク設定の推奨プラクティス



重要

クラスター内のホストのネットワーク設定を変更するには、RHV Manager を常に使用します。使用しない場合は、サポート対象外の設定が作成される可能性があります。詳細は、[Network Manager のステートフル設定 \(nmstate\)](#) を参照してください。

お使いのネットワーク環境が複雑な場合には、ホストを Red Hat Virtualization Manager に追加する前に、ホストネットワークを手動で設定しなければならない場合があります。

以下に示すホストネットワーク設定のプラクティスを検討してください。

- Cockpit を使用してネットワークを設定。nmtui または nmcli を使用することも可能。
- セルフホストエンジンのデプロイメントまたは Manager へのホスト追加にネットワークが必要ない場合には、ホストを Manager に追加した後に、管理ポータルでネットワークを設定します。[データセンターまたはクラスターでの新しい論理ネットワークの作成](#) を参照。
- 以下の命名規則を使用する。
 - VLAN デバイス: **VLAN_NAME_TYPE_RAW_PLUS_VID_NO_PAD**
 - VLAN インターフェイス: **physical_device.VLAN_ID** (例: **eth0.23**, **eth1.128**, **enp3s0.50**)
 - ボンディングインターフェイス: **bondnumber** (例: **bond0**, **bond1**)
 - ボンディングインテリアの VLAN: **bondnumber.VLAN_ID** (例: **bond0.50**, **bond1.128**)
- [ネットワークボンディング](#) を使用。Red Hat Virtualization ではネットワークチーミングはサポートされておらず、セルフホストエンジンのデプロイメントにホストが使用されたりホストが Manager に追加されたりすると、エラーが発生する原因となります。
- 推奨されるボンディングモードを使用。
 - 仮想マシンが **ovirtmgmt** ネットワークを使用しない場合には、ネットワークではサポートされるいずれかのボンディングモードが使用される。
 - 仮想マシンが **ovirtmgmt** ネットワークを使用する場合には、[仮想マシンのゲストが接続するブリッジで使用される場合にどのボンディングモードが有効ですか?](#) を参照。
 - Red Hat Virtualization のデフォルトのボンディングモードは **(Mode 4) Dynamic Link Aggregation** です。お使いのスイッチがリンクアグリゲーション制御プロトコル (LACP) に対応していない場合には、**(Mode 1) Active-Backup** を使用してください。詳細は、[ボンディングモード](#) を参照。
- 以下の例に示すように、物理 NIC 上に VLAN を設定します (以下の例では nmcli を使用していますが、任意のツールを使用できます)。

```
# nmcli connection add type vlan con-name vlan50 ifname eth0.50 dev eth0 id 50
# nmcli con mod vlan50 +ipv4.dns 8.8.8.8 +ipv4.addresses 123.123.0.1/24 +ipv4.gateway 123.123.0.254
```

- 以下の例に示すように、ボンディング上に VLAN を設定します (以下の例では nmcli を使用していますが、任意のツールを使用することができます)。

```
# nmcli connection add type bond con-name bond0 ifname bond0 bond.options "mode=active-backup,miimon=100" ipv4.method disabled ipv6.method ignore
# nmcli connection add type ethernet con-name eth0 ifname eth0 master bond0 slave-type
```

```

bond
# nmcli connection add type ethernet con-name eth1 ifname eth1 master bond0 slave-type
bond
# nmcli connection add type vlan con-name vlan50 ifname bond0.50 dev bond0 id 50
# nmcli con mod vlan50 +ipv4.dns 8.8.8.8 +ipv4.addresses 123.123.0.1/24 +ipv4.gateway
123.123.0.254

```

- **firewalld** は無効にしないでください。
- ホストを Manager に追加した後に、管理ポータルでファイアウォールルールをカスタマイズします。 [ホストファイアウォールルールの設定](#) を参照してください。

6.4. RED HAT VIRTUALIZATION MANAGER へのセルフホストエンジンノードの追加

セルフホスト型エンジンノードは、通常のホストと同じ方法で追加しますが、セルフホスト型エンジンノードとしてホストをデプロイするという追加のステップが必要です。共有ストレージドメインは自動的に検出され、ノードは必要に応じて Manager 用仮想マシンをホストするフェイルオーバー用ホストとして使用できます。セルフホスト型エンジン環境に通常のホストをアタッチできますが、Manager 用仮想マシンはホストできません。Manager 用仮想マシンの高可用性を確保するためには、セルフホスト型エンジンノードを最低でも 2 つ用意します。追加のホストは、REST API を使用して追加することもできます。 [REST API ガイドのホスト](#) を参照してください。

前提条件

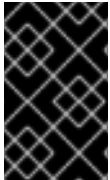
- セルフホスト型エンジンノードがすべて同じクラスター内にある。
- セルフホスト型エンジンノードを再利用する場合は、既存のセルフホスト型エンジン設定を削除する。 [セルフホスト型エンジン環境からのホストの削除](#) を参照してください。

手順

1. 管理ポータルで **Compute** → **Hosts** をクリックします。
2. **New** をクリックします。
ホストの追加設定に関する情報は、[管理ガイドの New Host および Edit Host ウィンドウの設定とコントロールの説明](#) を参照してください。
3. ドロップダウンリストを使用して、新規ホスト用の **Data Center** および **Host Cluster** を選択します。
4. 新規ホストの **Name** と **Address** を入力します。 **SSH Port** フィールドには、標準の SSH ポートであるポート 22 が自動入力されます。
5. Manager がホストにアクセスするために使用する認証メソッドを選択します。
 - パスワード認証を使用するには、root ユーザーのパスワードを入力します。
 - または、**SSH PublicKey** フィールドに表示される鍵をホスト上の `/root/.ssh/authorized_keys` にコピーして、公開鍵認証を使用します。
6. ホストにサポート対象の電源管理カードが搭載されている場合は、オプションとして電源管理を設定できます。電源管理の設定について、詳しくは [管理ガイドのホストの Power Management 設定の説明](#) を参照してください。

7. **Hosted Engine** タブをクリックします。
8. **Deploy** を選択します。
9. **OK** をクリックします。

6.5. RED HAT VIRTUALIZATION MANAGER への通常のホストの追加




重要

クラスター内のホストのネットワーク設定を変更するには、必ず RHV Manager を使用します。使用しない場合は、サポート対象外の設定が作成される可能性があります。詳細は、[Network Manager Stateful Configuration \(nmstate\)](#) を参照してください。

Red Hat Virtualization 環境にホストを追加するには、仮想化のチェック、パッケージのインストール、およびブリッジ作成の各ステップをプラットフォームで完了する必要があるため、多少時間がかかります。

手順

1. 管理ポータルから **Compute** → **Hosts** をクリックします。
2. **New** をクリックします。
3. ドロップダウンリストを使用して、新規ホスト用の **Data Center** および **Host Cluster** を選択します。
4. 新規ホストの **Name** と **Address** を入力します。**SSH Port** フィールドには、標準の SSH ポートであるポート 22 が自動入力されます。
5. Manager がホストにアクセスするために使用する認証メソッドを選択します。
 - パスワード認証を使用するには、root ユーザーのパスワードを入力します。
 - または、**SSH PublicKey** フィールドに表示される鍵をホスト上の `/root/.ssh/authorized_keys` にコピーして、公開鍵認証を使用します。
6. オプションで、**Advanced Parameters** ボタンをクリックして、以下に示すホストの詳細設定を変更します。
 - ファイアウォールの自動設定を無効にします。
 - ホストの SSH フィンガープリントを追加し、セキュリティを強化します。手動での追加または自動取得が可能です。
7. ホストにサポート対象の電源管理カードが搭載されている場合は、オプションとして電源管理を設定することができます。電源管理の設定に関する詳細は、[管理ガイドのホストパワーマネージメント設定の説明](#) を参照してください。
8. **OK** をクリックします。

新規ホストが **Installing** のステータスでホスト一覧に表示され、**通知トレイ** () のイベントセクションでインストールの進捗状況を確認できます。しばらくすると、ホストのステータスが **Up** に変わります。

第7章 RED HAT VIRTUALIZATION 用ストレージの追加

新たな環境にデータドメインとしてストレージを追加します。Red Hat Virtualization 環境には少なくとも1つのデータ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。

前の手順で準備したストレージを追加します。

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



重要

iSCSI ストレージを使用する場合には、新しいデータドメインはセルフホストエンジン用ストレージドメインと同じ iSCSI ターゲットを使用することはできません。



警告

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。

7.1. NFS ストレージの追加

ここでは、既存の NFS ストレージをデータドメインとして Red Hat Virtualization 環境にアタッチする手順を説明します。

ISO またはエクスポートドメインが必要な場合も、この手順を使用します。ただし、**Domain Function** の一覧では **ISO** または **Export** を選択します。

手順

1. 管理ポータルで **Storage** → **Domains** をクリックします。
2. **New Domain** をクリックします。
3. ストレージドメインの **Name** を入力します。
4. **Data Center**、**Domain Function**、**Storage Type**、**Format**、および **Host** のリストのデフォルト値をそのまま使用します。

5. ストレージドメインに使用する **Export Path** を入力します。エクスポートパスは、123.123.0.10:/data (IPv4 の場合)、[2001:0:0:0:0:5db1]:/data (IPv6 の場合)、または domain.example.com:/data の形式で指定する必要があります。
6. オプションで、詳細パラメーターを設定できます。
 - a. **Advanced Parameters** をクリックします。
 - b. **Warning Low Space Indicator** フィールドに、パーセンテージ値を入力します。ストレージドメインの空き容量がこの値を下回ると、ユーザーに警告メッセージが表示され、ログに記録されます。
 - c. **Critical Space Action Blocker** のフィールドに GB 単位で値を入力します。ストレージドメインの空き容量がこの値を下回ると、ユーザーにエラーメッセージが表示され、ログに記録されます。容量を消費する新規アクションは、一時的であってもすべてブロックされます。
 - d. 削除後にワイプするオプションを有効にするには、**Wipe After Delete** チェックボックスを選択します。このオプションは、ドメインの作成後に編集できますが、その場合はすでに存在している wipe after delete プロパティは変更されません。
7. **OK** をクリックします。

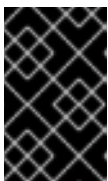
新しい NFS データドメインのステータスは、ディスクの準備ができるまで **Locked** になります。その後、データドメインはデータセンターに自動的にアタッチされます。

7.2. iSCSI ストレージの追加

ここでは、既存の iSCSI ストレージをデータドメインとして Red Hat Virtualization 環境にアタッチする手順について説明します。

手順

1. **Storage** → **Domains** をクリックします。
2. **New Domain** をクリックします。
3. 新規ストレージドメインの **Name** を入力します。
4. ドロップダウンリストから **Data Center** を選択します。
5. **Domain Function** として **Data** を、**Storage Type** として **iSCSI** を、それぞれ選択します。
6. **Host** としてアクティブなホストを選択します。



重要

ストレージドメインへの通信は、Manager から直接ではなく、選択したホストを介して行われます。したがって、ストレージドメインを設定する前には、全ホストがストレージデバイスにアクセスできる状態でなければなりません。

7. Manager は iSCSI ターゲットを LUN に、または LUN を iSCSI ターゲットにマッピングすることができます。**New Domain** ウィンドウでストレージタイプに iSCSI を選択すると、未使用の LUN が割り当てられた既知のターゲットが自動的に表示されます。ストレージの追加に使用するターゲットが表示されない場合は、ターゲットの検出機能を使用して検索することができます。表示されている場合は、次の手順に進んでください。

- a. **Discover Targets** をクリックし、ターゲットの検出オプションを有効にします。Manager がターゲットを検出してログインすると、**New Domain** ウィンドウに、その環境では未使用の LUN が割り当てられたターゲットが自動的に表示されます。



注記

環境外で使用されている LUN も表示されます。

Discover Targets のオプションを使用すると、多数のターゲットの LUN を追加したり、同じ LUN に複数のパスを追加したりすることができます。



重要

REST API メソッド **discoveriscsi** を使用して、iSCSI ターゲットを検出する場合には、FQDN または IP アドレスを使用できますが、REST API メソッド **iscsilogin** を使用してログインするには、検出された iSCSI ターゲットの詳細を使用する必要があります。詳細は、**REST API ガイド** の [discoveriscsi](#) を参照してください。

- b. **Address** フィールドに iSCSI ホストの FQDN または IP アドレスを入力します。
- c. **Port** フィールドには、ターゲットを参照する際にホストに接続するポートを入力します。デフォルトは **3260** です。
- d. ストレージのセキュリティー保護に CHAP を使用している場合は、**User Authentication** チェックボックスを選択します。**CHAP user name** と **CHAP password** を入力してください。



注記

REST API を使用して、特定ホストの iSCSI ターゲットに認証情報を定義することができます。詳細は、**REST API ガイド** の [StorageServerConnectionExtensions: add](#) を参照してください。

- e. **Discover** をクリックします。
- f. 検出結果から1つまたは複数のターゲットを選択し、1つのターゲットの場合は **Login** をクリックします。複数のターゲットの場合は **Login All** をクリックします。



重要

複数のパスのアクセスが必要な場合は、すべての必要なパスを通してターゲットを検出してログインする必要があります。ストレージドメインを変更してパスを追加する方法は、現在サポートされていません。



重要

REST API メソッド **iscsilogin** を使用してログインする場合は、**discoveriscsi** メソッドで検出された iSCSI ターゲットの詳細を使用する必要があります。詳細は、**REST API ガイド** の [iscsilogin](#) を参照してください。

8. ターゲットの横に表示されている + ボタンをクリックします。エントリーが展開され、ターゲットにアタッチされている未使用の LUN がすべて表示されます。
9. ストレージドメインの作成に使用する各 LUN のチェックボックスにチェックを入れます。
10. オプションで、詳細パラメーターを設定できます。
 - a. **Advanced Parameters** をクリックします。
 - b. **Warning Low Space Indicator** フィールドに、パーセンテージ値を入力します。ストレージドメインの空き容量がこの値を下回ると、ユーザーに警告メッセージが表示され、ログに記録されます。
 - c. **Critical Space Action Blocker** のフィールドに GB 単位で値を入力します。ストレージドメインの空き容量がこの値を下回ると、ユーザーにエラーメッセージが表示され、ログに記録されます。容量を消費する新規アクションは、一時的であってもすべてブロックされません。
 - d. 削除後にワイプするオプションを有効にするには、**Wipe After Delete** チェックボックスを選択します。このオプションは、ドメインの作成後に編集できますが、その場合はすでに存在している wipe after delete プロパティは変更されません。
 - e. **Discard After Delete** のチェックボックスを選択して、削除後に破棄のオプションを有効化します。このオプションは、ドメインの作成後に編集できます。また、このオプションを利用できるのは、ブロックストレージドメインのみです。
11. **OK** をクリックします。

同じターゲットに対して複数のストレージ接続パスを設定している場合は、[iSCSI マルチパスの設定](#) に記載されている手順に従い、iSCSI のボンディング設定を完了してください。

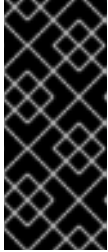
現在のストレージネットワークを iSCSI ボンディングに移行するには、[論理ネットワークから iSCSI ボンディングへの移行](#) を参照してください。

7.3. FCP ストレージの追加

ここでは、既存の FCP ストレージをデータドメインとして Red Hat Virtualization 環境にアタッチする手順について説明します。

手順

1. **Storage → Domains** をクリックします。
2. **New Domain** をクリックします。
3. ストレージドメインの **Name** を入力します。
4. ドロップダウンリストから **FCP Data Center** を選択します。
適切な FCP データセンターがない場合は **(none)** を選択します。
5. ドロップダウンリストから **Domain Function** および **Storage Type** を選択します。選択したデータセンターとの互換性がないストレージドメインタイプは選択できません。
6. **Host** フィールドでアクティブなホストを1台選択します。データセンターで初めて作成するデータドメインではない場合、そのデータセンターの SPM ホストを選択する必要があります。



重要

ストレージドメインへの通信はすべて、Red Hat Virtualization Manager から直接ではなく、選択したホストを介して行われます。システムには、アクティブなホストが少なくとも1台存在し、選択したデータセンターにアタッチされている必要があります。全ホストがストレージデバイスにアクセスできる状態でなければ、ストレージドメインは設定できません。

7. **New Domain** ウィンドウで、ストレージタイプとして **Fibre Channel** を選択した場合は、未使用の LUN が割り当てられた既知のターゲットが自動的に表示されます。LUN ID チェックボックスを選択し、使用可能な LUN をすべて選択します。
8. オプションで、詳細パラメーターを設定できます。
 - a. **Advanced Parameters** をクリックします。
 - b. **Warning Low Space Indicator** フィールドに、パーセンテージ値を入力します。ストレージドメインの空き容量がこの値を下回ると、ユーザーに警告メッセージが表示され、ログに記録されます。
 - c. **Critical Space Action Blocker** のフィールドに GB 単位で値を入力します。ストレージドメインの空き容量がこの値を下回ると、ユーザーにエラーメッセージが表示され、ログに記録されます。容量を消費する新規アクションは、一時的であってもすべてブロックされません。
 - d. 削除後にワイプするオプションを有効にするには、**Wipe After Delete** チェックボックスを選択します。このオプションは、ドメインの作成後に編集できますが、その場合はすでに存在している wipe after delete プロパティは変更されません。
 - e. **Discard After Delete** のチェックボックスを選択して、削除後に破棄のオプションを有効化します。このオプションは、ドメインの作成後に編集できます。また、このオプションを利用できるのは、ブロックストレージドメインのみです。
9. **OK** をクリックします。

使用準備中は、新規 FCP データドメインのステータスは **Locked** になります。準備が整った時点で、自動的にデータセンターにアタッチされます。

7.4. RED HAT GLUSTER STORAGE の追加

Red Hat Virtualization で Red Hat Gluster Storage を使用するには、[Red Hat Virtualization で Red Hat Gluster Storage を使用する場合の設定](#) を参照してください。

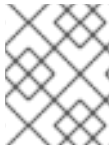
Red Hat Virtualization でサポートされている Red Hat Gluster Storage のバージョンについては、[Red Hat Gluster Storage のバージョン互換性とサポート](#) を参照してください。

付録A セルフホストエンジンのデプロイメントのトラブルシューティング

セルフホストエンジンがすでにデプロイされているかどうかを確認するには、**hosted-engine --check-deployed** を実行します。セルフホストエンジンがデプロイされていない場合にだけ、エラーが表示されます。

A.1. MANAGER 用仮想マシンのトラブルシューティング

hosted-engine --vm-status を実行して Manager 用仮想マシンのステータスを確認します。



注記

Manager 用仮想マシンに加えた変更がステータスコマンドの出力に反映されるには、20 秒ほどかかります。

出力の **Engine status** ごとに、問題を特定または解決するためのアドバイスを以下に示します。

Engine status: "health": "good", "vm": "up" "detail": "up"

1. Manager 用仮想マシンが通常通りに稼働中の場合には、以下のような出力が表示されます。

```

---= Host 1 status =---

Status up-to-date      : True
Hostname               : hypervisor.example.com
Host ID                : 1
Engine status          : {"health": "good", "vm": "up", "detail": "up"}
Score                  : 3400
stopped                : False
Local maintenance     : False
crc32                  : 99e57eba
Host timestamp         : 248542
  
```

2. 出力は正常だが Manager に接続することができない場合は、ネットワーク接続を確認してください。

Engine status: "reason": "failed liveness check", "health": "bad", "vm": "up", "detail": "up"

1. **health** が **bad** で **vm** が **up** の場合、HA サービスは Manager 用仮想マシンを再起動して Manager の復旧を試みます。数分以内に復旧しない場合は、コマンドラインからグローバルメンテナンスモードを有効にして、ホストを HA サービスの管理対象外にします。

```
# hosted-engine --set-maintenance --mode=global
```

2. コンソールに接続します。プロンプトが表示されたら、オペレーティングシステムの root パスワードを入力します。その他のコンソールオプションについては、[How to access Hosted Engine VM console from RHEV-H host?](#) を参照してください。

```
# hosted-engine --console
```

3. Manager 用仮想マシンにログインして、オペレーティングシステムが動作していることを確認します。

4. **ovirt-engine** サービスのステータスを確認します。

```
# systemctl status -l ovirt-engine
# journalctl -u ovirt-engine
```

5. `/var/log/messages`、`/var/log/ovirt-engine/engine.log`、および `/var/log/ovirt-engine/server.log` のログを確認します。
6. 問題を解決したら、セルフホストエンジンノードのいずれかから、手動で Manager 用仮想マシンを再起動します。

```
# hosted-engine --vm-shutdown
# hosted-engine --vm-start
```



注記

セルフホストエンジンノードがグローバルメンテナンスモードにある場合は、Manager 用仮想マシンを手動で再起動する必要があります。コマンドラインから **reboot** コマンドを送信して Manager 用仮想マシンを再起動しようとしても、Manager 用仮想マシンは電源オフのままです。設計上、このようになります。

7. Manager 用仮想マシンで **ovirt-engine** サービスが稼働中であることを確認します。

```
# systemctl status ovirt-engine.service
```

8. Manager 用仮想マシンが稼働中であることを確認した後は、コンソールセッションを終了して、メンテナンスモードを無効にし、HA サービスを再び有効にします。

```
# hosted-engine --set-maintenance --mode=none
```

Engine status: "vm": "down", "health": "bad", "detail": "unknown", "reason": "vm not running on this host"



注記

このメッセージは、現在 Manager 用仮想マシンを稼働していないホストで表示されることが想定されます。

1. 環境内に複数のホストがある場合は、現在別のホストが Manager 用仮想マシンの再起動を試みしていないことを確認します。
2. グローバルメンテナンスモードにないことを確認します。
3. `/var/log/ovirt-hosted-engine-ha/agent.log` で、**ovirt-ha-agent** のログを確認します。
4. セルフホストエンジンノードのいずれかから、手動で Manager 用仮想マシンの再起動を試みます。

```
# hosted-engine --vm-shutdown
# hosted-engine --vm-start
```


Engine status: "vm": "unknown", "health": "unknown", "detail": "unknown", "reason": "failed to getVmStats"

このステータスは、**ovirt-ha-agent** が VDSM から仮想マシンの詳細を取得できなかったことを意味しています。

1. `/var/log/vdsm/vdsm.log` で VDSM のログを確認します。
2. `/var/log/ovirt-hosted-engine-ha/agent.log` で、**ovirt-ha-agent** のログを確認します。

Engine status: セルフホストエンジンの設定が共有ストレージから取得されていないステータスが表示されると、共有ストレージからセルフホストエンジン設定は取得されていません。**ovirt-ha-agent** が実行されており、ストレージサーバーにアクセスできることを確認してください。**ovirt-ha-agent** サービス、ストレージ、またはその両方に問題があります。

1. ホストで **ovirt-ha-agent** のステータスを確認します。

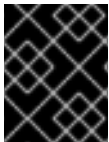
```
# systemctl status -l ovirt-ha-agent
# journalctl -u ovirt-ha-agent
```

2. **ovirt-ha-agent** がダウンしている場合は、再起動します。

```
# systemctl start ovirt-ha-agent
```

3. `/var/log/ovirt-hosted-engine-ha/agent.log` で、**ovirt-ha-agent** のログを確認します。
4. 共有ストレージに ping を送信できることを確認します。
5. 共有ストレージがマウントされているかどうかを確認します。

その他のトラブルシューティング用コマンド



重要

以下のコマンドのいずれかを実行してセルフホストエンジン環境のトラブルシューティングを行う必要がある場合には、Red Hat サポートまでご連絡ください。

- **hosted-engine --reinitialize-lockspace**: このコマンドは、sanlock ロックスペースが壊れている場合に使用します。sanlock ロックスペースを再初期化する前に、グローバルメンテナンスモードが有効で Manager 用仮想マシンが停止していることを確認してください。
- **hosted-engine --clean-metadata**: ホストのエージェントのメタデータをグローバルステータスデータベースから削除します。これにより、他のホストではすべて、このホストについての情報はなくなります。ターゲットのホストが停止状態でグローバルメンテナンスモードが有効であることを確認してください。
- **hosted-engine --check-liveliness**: このコマンドは、ovirt-engine サービスの liveliness ページを確認します。Web ブラウザーで <https://engine-fqdn/ovirt-engine/services/health/> に接続して確認することもできます。
- **hosted-engine --connect-storage**: このコマンドは、ホストと Manager 用仮想マシンに必要なストレージ接続をすべて準備するように VDSM に指示します。これは通常、セルフホストエンジンのデプロイ中にバックエンドで実行します。このコマンドを実行してストレージの問題のトラブルシューティングを行う必要がある場合には、グローバルメンテナンスモードを必ず有効にしてください。

A.2. 失敗したセルフホストエンジンのデプロイメントのクリーンアップ

セルフホストエンジンのデプロイメントが中断された場合には、その後のデプロイメントは失敗して、エラーメッセージが表示されます。このエラーはデプロイメントが失敗した段階によって異なります。

エラーメッセージが表示される場合には、デプロイメントホストでクリーンアップスクリプトを実行して、失敗したデプロイメントをクリーンアップすることができます。ただし、最良の手段は、ベースのオペレーティングシステムを再インストールして、デプロイメントを最初からやり直すことです。



注記

クリーンアップスクリプトには、以下の制約があります。

- スクリプトの実行中にネットワークの接続が中断すると、スクリプトによる管理ブリッジの削除や作業用ネットワーク設定の再作成に失敗する場合があります。
- スクリプトでは、失敗したデプロイメント中に使用された共有ストレージデバイスをクリーンアップすることができません。以降のデプロイメントで再使用するには、共有ストレージデバイスをクリーンアップする必要があります。

手順

1. `/usr/sbin/ovirt-hosted-engine-cleanup` を実行して **y** を選択し、失敗したセルフホストエンジンのデプロイメントで残されたものを削除します。

```
# /usr/sbin/ovirt-hosted-engine-cleanup
This will de-configure the host to run ovirt-hosted-engine-setup from scratch.
Caution, this operation should be used with care.
Are you sure you want to proceed? [y/n]
```

2. 同じ共有ストレージデバイスに再インストールするか、別の共有ストレージデバイスを選択するかを定義します。

- 同じストレージドメインにインストール環境をデプロイする場合は、NFS、Gluster、PosixFS またはローカルストレージドメインのサーバーの適切なディレクトリーで以下のコマンドを実行し、そのストレージドメインをクリーンアップします。

```
# rm -rf storage_location/*
```

- iSCSI またはファイバーチャネルプロトコル (FCP) ストレージの場合は、[How to Clean Up a Failed Self-hosted Engine Deployment?](#) を参照してください。
- セルフホストエンジンホストを再起動するか、別の共有ストレージデバイスを選択してください。



注記

次の試行の前にストレージへのすべての接続がクリーンアップされていることを確認するには、再起動が必要です。

3. セルフホストエンジンを再デプロイします。

付録B デプロイメント中に自動化を使用して MANAGER 仮想マシンをカスタマイズする

自動化を使用して、1つ以上の Ansible Playbook を使用して、デプロイ中に Manager 仮想マシンを調整またはカスタマイズできます。デプロイ中の次の時点で Playbook を実行できます。

- セルフホストエンジンセットアップの前
- セルフホストエンジンのセットアップ後、ストレージが設定される前
- デプロイメントホストを Manager に追加した後
- デプロイメントが完全に完了した後

手順

1. デプロイメントプロセスの特定のポイントで Manager 仮想マシン上で実行する1つ以上の Ansible Playbook を作成します。
2. Playbook を `/usr/share/ansible/collections/ansible_collections/redhat/rhv/roles/hosted_engine_setup/hooks/` の下に追加します。

enginevm_before_engine_setup

セルフホストエンジンをセットアップする前に、Playbook を実行してください。

enginevm_after_engine_setup

セルフホストエンジンのセットアップ後、ストレージを設定する前に、Playbook を実行します。

after_add_host

デプロイメントホストを Manager に追加した後、Playbook を実行します。

after_setup

展開が完了したら、Playbook を実行します。

self-hosted-engine インストーラーを実行すると、デプロイメントスクリプトは **ovirt-engine-setup** ロールを実行します。このロールは、これらのディレクトリーのいずれかで Playbook を自動的に実行します。

関連情報

- [コマンドラインを使用したセルフホストエンジンのデプロイ](#)
- [Ansible を使用した設定タスクの自動化](#)
- Ansible ドキュメントの [Playbook](#) の紹介

付録C リモートサーバーへのデータベースおよびサービスの移行

自動インストール中にリモートデータベースおよびサービスを設定することはできませんが、インストール後にこれらを別のサーバーに移行することができます。

C.1. データウェアハウスを別のマシンに移行する

このセクションでは、Data Warehouse データベースおよびサービスを Red Hat Virtualization Manager マシンから別のマシンに移行する方法を説明します。Data Warehouse サービスを別のマシンでホストすると、各個別マシンの負荷が削減され、CPU やメモリーリソースを他のプロセスと共有することで競合が生じる可能性を回避できます。



注記

Data Warehouse データベース、Data Warehouse サービス、Grafana はそれぞれ別々のマシンにインストールできますが、Red Hat はこれらの各コンポーネントをすべて同じマシンにインストールする場合のみサポートします。

以下の移行オプションがあります。

- Manager マシンから Data Warehouse サービスを移行し、既存の Data Warehouse データベース (`ovirt_engine_history`) に接続できます。
- Manager マシンから Data Warehouse データベースを移行してから、Data Warehouse サービスを移行することができます。

C.1.1. 別のマシンへの Data Warehouse データベースの移行

Data Warehouse サービスを移行する前に、Data Warehouse データベース (`ovirt_engine_history`) を移行します。`engine-backup` を使用してデータベースのバックアップを作成し、それを新規データベースマシンで復元します。`engine-backup` の詳細が必要な場合は、`engine-backup --help` を実行してください。



注記

Data Warehouse データベース、Data Warehouse サービス、Grafana はそれぞれ別々のマシンにインストールできますが、Red Hat はこれらの各コンポーネントをすべて同じマシンにインストールする場合のみサポートします。

新規データベースサーバーに Red Hat Enterprise Linux 8 がインストールされている必要があります。

新規データベースサーバーで必要なりポジトリを有効にします。

C.1.1.1. Red Hat Virtualization Manager リポジトリの有効化

ログインして、Red Hat Subscription Manager で Data Warehouse 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ユ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 \
  --enable=rhel-8-for-x86_64-appstream-tus-rpms \
  --enable=rhel-8-for-x86_64-baseos-tus-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. **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
```

7. **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
```

8. [アップグレードガイド](#) の [セルフホストエンジンの更新](#) の手順を使用して、セルフホストエンジンを更新します。

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](#)
- [後のストリームへの切り替え](#)

C.1.1.2. 別のマシンへの Data Warehouse データベースの移行

手順

1. Manager で Data Warehouse データベースおよび設定ファイルのバックアップを作成します。

```
# engine-backup --mode=backup --scope=grafanadb --scope=dwhdb --scope=files --
file=file_name --log=log_file_name
```

2. そのバックアップファイルを Manager マシンから新たなマシンにコピーします。

```
# scp /tmp/file_name root@new.dwh.server.com:/tmp
```

3. **engine-backup** を新しいマシンにインストールします。

```
# dnf install ovirt-engine-tools-backup
```

4. PostgreSQL サーバーパッケージをインストールします。

```
# dnf install postgresql-server postgresql-contrib
```

5. PostgreSQL データベースを初期化し、**postgresql** サービスを開始して、このサービスが起動時に開始されることを確認します。

```
# su - postgres -c 'initdb'
# systemctl enable postgresql
# systemctl start postgresql
```

6. 新しいマシンで Data Warehouse データベースを復元します。**file_name** は、Manager からコピーされたバックアップファイルです。

```
# engine-backup --mode=restore --scope=files --scope=grafanadb --scope=dwhdb --
file=file_name --log=log_file_name --provision-dwh-db
```

復元モードで **--provision-*** オプションを使用すると、デフォルトで **--restore-permissions** が適用されます。

これで、Manager がホストされるマシンとは別のマシンで、Data Warehouse データベースがホストされるようになりました。Data Warehouse データベースを正常に復元したら、**engine-setup** コマンドの実行を指示するプロンプトが表示されます。このコマンドを実行する前に、Data Warehouse サービスを移行します。

C.1.2. 別のマシンへの Data Warehouse サービスの移行

Red Hat Virtualization Manager にインストールおよび設定した Data Warehouse サービスは、別のマシンに移行することができます。Data Warehouse サービスを別のマシンでホストすることは、Manager マシンの負荷を削減する上で役立ちます。

この手順では、Data Warehouse サービスのみを移行することに注意してください。

Data Warehouse サービスを移行する前に Data Warehouse データベース (**ovirt_engine_history**) を移行するには、[Data Warehouse のデータセットの別のマシンへの移行](#) を参照してください。



注記

Data Warehouse データベース、Data Warehouse サービス、Grafana はそれぞれ別々のマシンにインストールできますが、Red Hat はこれらの各コンポーネントをすべて同じマシンにインストールする場合のみサポートします。

前提条件

- Manager と Data Warehouse が同じマシン上にインストールおよび設定されている。
- 新たな Data Warehouse マシンを設定する場合は以下を満たしていること。
 - Manager の `/etc/ovirt-engine/engine.conf.d/10-setup-database.conf` ファイルからのパスワード。
 - Data Warehouse マシンから Manager データベースマシンの TCP ポート 5432 へのアクセス許可。
 - Manager の `/etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/10-setup-database.conf` ファイルからの Data Warehouse データベースのユーザー名とパスワード。
[Data Warehouse データセットの別のマシンへの移行](#) で説明されている手順を使用して **ovirt_engine_history** データベースを移行した場合、バックアップには、そのマシンでのデータベースのセットアップ中に定義したこれらの認証情報が含まれます。

このシナリオのインストールでは、以下の 4 つのステップを実施する必要があります。

1. 新たな Data Warehouse マシンの準備
2. Manager マシンでの Data Warehouse サービスの停止
3. 新たな Data Warehouse マシンの設定
4. Manager マシンでの Data Warehouse サービスの無効化

C.1.2.1. 新たな Data Warehouse マシンの準備

Red Hat Virtualization のリポジトリを有効にし、Red Hat Enterprise Linux 8 マシンに Data Warehouse セットアップパッケージをインストールします。

1. 必要なリポジトリを有効にします。

- a. 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
```

- b. **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
```

- c. 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
```

- d. 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

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

2. **pki-deps** モジュールを有効にします。

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

3. 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
```

4. **ovirt-engine-dwh-setup** パッケージをインストールします。

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

C.1.2.2. Manager マシンでの Data Warehouse サービスの停止

手順

1. Data Warehouse サービスを停止します。

```
# systemctl stop ovirt-engine-dwhd.service
```


- データベースのリセットマシンでホストされる場合は、`postgres.conf` ノアイルを編集して手動でアクセス権限を付与する必要があります。`/var/lib/pgsql/data/postgresql.conf` ファイルを編集し、`listen_addresses` 行を変更して以下と一致するようにします。

```
listen_addresses = '*'
```

その行が存在しない、またはコメントアウトされている場合は、手動で追加します。

Manager マシンでデータベースがホストされていて、そのデータベースが Red Hat Virtualization Manager のクリーンセットアップ中に設定された場合は、デフォルトでアクセス権限が付与されます。

- `postgresql` サービスを再起動します。

```
# systemctl restart postgresql
```

C.1.2.3. 新たな Data Warehouse マシンの設定

このセクションで示すオプションまたは設定の順序は、お使いの環境によって異なる場合があります。

- ovirt_engine_history** データベースと Data Warehouse サービスの両方を **同じ** マシンに移行する場合は、以下のコマンドを実行します。移行しない場合は、次のステップに進みます。

```
# sed -i '/^ENGINE_DB_/d' \
    /etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/10-setup-database.conf

# sed -i \
    -e 's;^\(OVESETUP_ENGINE_CORE/enable=bool\):True;1:False;' \
    -e '/^OVESETUP_CONFIG/fqdn/d' \
    /etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf
```

- `apache/grafana` PKI ファイルを削除して、**engine-setup** によって正しい値で再生成されるようにします。

```
# rm -f \
    /etc/pki/ovirt-engine/certs/apache.cer \
    /etc/pki/ovirt-engine/certs/apache-grafana.cer \
    /etc/pki/ovirt-engine/keys/apache.key.nopass \
    /etc/pki/ovirt-engine/keys/apache-grafana.key.nopass \
    /etc/pki/ovirt-engine/apache-ca.pem \
    /etc/pki/ovirt-engine/apache-grafana-ca.pem
```

- engine-setup** コマンドを実行し、マシンでの Data Warehouse の設定を開始します。

```
# engine-setup
```

- Enter** キーを押して自動検出されたホスト名をそのまま使用するか、別のホスト名を入力して **Enter** キーを押します。

```
Host fully qualified DNS name of this server [autodetected host name]:
```

- Enter** キーを押して、ファイアウォールを自動設定するか、**No** と入力し、**Enter** キーを押して、既存の設定を維持します。

Setup can automatically configure the firewall on this system.

Note: automatic configuration of the firewall may overwrite current settings.

Do you want Setup to configure the firewall? (Yes, No) [Yes]:

ファイアウォールの自動設定を選択した場合に、ファイアウォール管理機能がアクティブ化されていない場合は、サポートされているオプション一覧から、選択したファイアウォール管理機能を指定するように要求されます。ファイアウォール管理機能の名前を入力して、**Enter** キーを押してください。この操作は、オプションが1つしかリストされていない場合でも必要です。

6. Manager の完全修飾ドメイン名およびパスワードを入力します。その他のフィールドについては、**Enter** キーを押してそれぞれのデフォルト値をそのまま使用します。

Host fully qualified DNS name of the engine server []: **engine-fqdn**

Setup needs to do some actions on the remote engine server. Either automatically, using ssh as root to access it, or you will be prompted to manually perform each such action.

Please choose one of the following:

1 - Access remote engine server using ssh as root

2 - Perform each action manually, use files to copy content around

(1, 2) [1]:

ssh port on remote engine server [22]:

root password on remote engine server **engine-fqdn: password**

7. Manager データベースマシンの完全修飾ドメイン名 (FQDN) およびパスワードを入力します。その他のフィールドについては、**Enter** キーを押してそれぞれのデフォルト値をそのまま使用します。

Engine database host []: **manager-db-fqdn**

Engine database port [5432]:

Engine database secured connection (Yes, No) [No]:

Engine database name [engine]:

Engine database user [engine]:

Engine database password: **password**

8. インストールの設定を確認します。

Please confirm installation settings (OK, Cancel) [OK]:

これで、Data Warehouse サービスがリモートマシンに設定されました。次は、Manager マシンの Data Warehouse サービスを無効にします。

C.1.2.4. Manager マシンでの Data Warehouse サービスの無効化

前提条件

- Manager マシンの Grafana サービスが無効になっている。

```
# systemctl disable --now grafana-server.service
```

手順

1. Manager マシンで Manager を再起動します。

```
# service ovirt-engine restart
```

2. 以下のコマンドを実行して `/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf` ファイルを変更し、オプションを **False** に設定します。

```
# sed -i \  
-e 's;^\(OVESETUP_DWH_CORE/enable=bool\):True;\1:False;' \  
-e 's;^\(OVESETUP_DWH_CONFIG/remoteEngineConfigured=bool\):True;\1:False;' \  
/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf
```

```
# sed -i \  
-e 's;^\(OVESETUP_GRAFANA_CORE/enable=bool\):True;\1:False;' \  
/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf
```

3. Data Warehouse サービスを無効にします。

```
# systemctl disable ovirt-engine-dwhd.service
```

4. Data Warehouse に関するファイルを削除します。

```
# rm -f /etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/*.conf /var/lib/ovirt-engine-  
dwh/backups/*
```

これで、Data Warehouse サービスが Manager とは別のマシンでホストされるようになりました。

付録D PCI パススルーを有効にするためのホストの設定



注記

これは、Red Hat Virtualization で SR-IOV を準備およびセットアップする方法を示す一連のトピックの1つです。詳細は、[SR-IOV のセットアップと設定](#) を参照してください。

PCI パススルーを有効化すると、デバイスが仮想マシンに直接アタッチされているかのように、ホストのデバイスを仮想マシンで使用できます。PCI パススルー機能を有効化するには、仮想化拡張機能および IOMMU 機能を有効化する必要があります。以下の手順では、ホストを再起動する必要があります。すでにホストが Manager にアタッチされている場合は、最初にホストがメンテナンスモードに設定されていることを確認してください。

前提条件

- ホストハードウェアが PCI デバイスパススルーおよび割り当ての要件を満たしていることを確認する。詳細は、[PCI デバイスの要件](#) を参照してください。

PCI パススルー用ホストの設定

1. BIOS の仮想化拡張機能および IOMMU 拡張機能を有効にします。詳細は、[Red Hat Enterprise Linux 仮想化の導入および管理ガイドの BIOS での INTEL VT-X と AMD-V の仮想化ハードウェア拡張の有効化](#) を参照してください。
2. ホストを Manager に追加する際に **Hostdev Passthrough & SR-IOV** のチェックボックスを選択するか、手動で **grub** 設定ファイルを編集して、カーネルの IOMMU フラグを有効化します。
 - 管理ポータルから IOMMU フラグを有効化する方法については、[Red Hat Virtualization Manager への通常ホストの追加](#) および [カーネル設定の説明](#) を参照してください。
 - 手動で **grub** 設定ファイルを編集する方法については、[IOMMU の手動での有効化](#) を参照してください。
3. GPU パススルーを有効にするには、ホストとゲストシステムの両方で追加の設定手順を実行する必要があります。詳細は、[Red Hat Virtualization での仮想マシン用 NVIDIA GPU のセットアップの GPU デバイスパススルー: 単一の仮想マシンへのホスト GPU の割り当て](#) を参照してください。

IOMMU の手動での有効化

1. **grub** 設定ファイルを編集して IOMMU を有効化します。



注記

IBM POWER8 ハードウェアを使用している場合は、デフォルトで IOMMU が有効になっているため、この手順は省略してください。

- Intel の場合は、マシンを起動し、**grub** 設定ファイルの **GRUB_CMDLINE_LINUX** 行の末尾に **intel_iommu=on** を追加します。

```
# vi /etc/default/grub
```

```
...
```

```
GRUB_CMDLINE_LINUX="nofb splash=quiet console=tty0 ... intel_iommu=on
...
```

- AMD の場合は、マシンを起動し、`grub` 設定ファイルの **GRUB_CMDLINE_LINUX** 行の末尾に **amd_iommu=on** を追加します。

```
# vi /etc/default/grub
...
GRUB_CMDLINE_LINUX="nofb splash=quiet console=tty0 ... amd_iommu=on
...
```

注記

intel_iommu=on または AMD IOMMU が検出される場合は、**iommu=pt** を追加してみてください。**pt** オプションでは、パススルーで使用するデバイスの IOMMU のみが有効化され、ホストのパフォーマンスが向上します。ただし、このオプションはすべてのハードウェアでサポートされているわけではありません。**pt** オプションがお使いのホストで機能しない場合は、以前のオプションに戻してください。

ハードウェアが割り込みの再マッピングをサポートしていないためにパススルーが失敗する場合、仮想マシンが信頼できるのであれば

allow_unsafe_interrupts オプションを有効化することも検討してください。**allow_unsafe_interrupts** を有効化すると、ホストが仮想マシンからの MSI 攻撃にさらされる可能性があるため、このオプションはデフォルトで有効化されていません。オプションを有効化するには、以下のとおり設定してください。

```
# vi /etc/modprobe.d
options vfio_iommu_type1 allow_unsafe_interrupts=1
```

2. `grub.cfg` ファイルをリフレッシュしてからホストを再起動し、変更を有効にします。

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

```
# reboot
```

付録E カーネルモジュールの自動読み込みの回避

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"
```

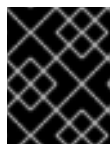


注記

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。

E.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F RED HAT VIRTUALIZATION のセキュリティー保護

これは、Red Hat Virtualization 固有の情報です。次のいずれかに関連する、基本的なセキュリティープラクティスについての説明ではありません。

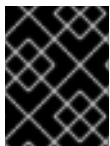
- 不要なサービスの無効化
- 認証
- 認可
- アカウンティング
- RHV 以外のサービスの侵入テストおよび耐性
- 機密アプリケーションデータの暗号化

前提条件

- 所属する組織のセキュリティー標準およびプラクティスを熟知していること。可能な場合は、所属する組織のセキュリティー担当者にお問い合わせください。
- Red Hat Enterprise Linux [Security hardening](#) を参照してから、RHEL ホストをデプロイすること。

F.1. RHEL ベースのホストおよびスタンドアロン MANAGER での DISA STIG プロファイルの適用

RHV をインストールするときに、UI インストーラーで DISA STIG プロファイルを選択できます。これは、RHEL 8 が提供するプロファイルです。



重要

DISA STIG プロファイルは、Red Hat Virtualization Host (RHVH) ではサポートされていません。

手順

1. **Installation Summary** 画面で、**Security Policy** を選択します。
2. **Security Policy** 画面で、**Apply security policy** を **On** に設定します。
3. **DISA STIG for Red Hat Enterprise Linux 8** を選択します。
4. **Select profile** をクリックします。クリックすると、プロファイルの横に緑色のチェックマークが追加され、パッケージが **Changes that were done or need to be done** の一覧に追加されます。変更するように指示された場合は、画面の指示に従ってください。
5. **Done** をクリックします。
6. **Installation Summary** 画面で、**Security Policy** のステータスが **Everything okay** であることを確認します。
7. ホストを再起動します。

F.1.1. セルフホストエンジンでの DISA STIG の有効化

コマンドラインを使用すると、デプロイメント中にセルフホスト型エンジンで DISA STIG を有効にすることができます。

手順

1. セルフホスト型エンジンのデプロイメントスクリプトを開始します。 [コマンドラインを使用して Red Hat Virtualization をセルフホストエンジンとしてインストール](#) を参照してください。
2. デプロイメントスクリプトで **Do you want to apply an OpenSCAP security profile?** のプロンプトが表示されたら、**Yes** と入力します
3. デプロイメントスクリプトで、**Please provide the security profile you would like to use?** のプロンプトが表示されたら、**stig** と入力します。

F.2. RHV ホストおよびスタンドアロン MANAGER での PCI-DSS プロファイルの適用

RHVH をインストールする際に、RHEL 8 が提供するプロファイルである PCI-DSS プロファイルを UI インストーラーで選択できます。

手順

1. **Installation Summary** 画面で、**Security Policy** を選択します。
2. **Security Policy** 画面で、**Apply security policy** を **On** に設定します。
3. **PCI-DSS v3.2.1 Control Baseline for Red Hat Enterprise Linux 8** を選択します。
4. **Select profile** をクリックします。クリックすると、プロファイルの横に緑色のチェックマークが追加され、パッケージが **Changes that were done or need to be done** の一覧に追加されず。変更するように指示された場合は、画面の指示に従ってください。
5. **Done** をクリックします。
6. **Installation Summary** 画面で、**Security Policy** のステータスが **Everything okay** であることを確認します。
7. ホストを再起動します。

F.2.1. セルフホストエンジンでの PCI-DSS の有効化

コマンドラインを使用すると、デプロイメント中にセルフホスト型エンジンで PCI-DSS を有効にできます。

手順

1. セルフホスト型エンジンのデプロイメントスクリプトを開始します。 [コマンドラインを使用して Red Hat Virtualization をセルフホストエンジンとしてインストール](#) を参照してください。
2. デプロイメントスクリプトで **Do you want to apply an OpenSCAP security profile?** のプロンプトが表示されたら、**Yes** と入力します

3. デプロイメントスクリプトで、**Please provide the security profile you would like to use?** のプロンプトが表示されたら、**pci-dss** と入力します。

付録G セルフホステッドエンジンのデプロイで許可される CPU タイプの定義

手順

1. **deploy.json** という名前のファイルを作成し、以下に示す表から **he_cluster_cpu_type** の CPU タイプを選択します。たとえば、必要な CPU タイプが **Secure Intel Nehalem Family** の場合、**deploy.json** は次のようになります。

```
[root@host ~]# cat deploy.json
{
  "he_cluster_cpu_type": "Secure Intel Nehalem Family"
}
```

2. **deploy.json** ファイルを **hosts-engine --deploy** プロセスに提供します。

```
[root@host ~]# hosted-engine --deploy --ansible-extra-vars=@/root/deploy.json
```

表G.1 許可される CPU タイプ

| CPU タイプ名 | CPU プロパティ |
|--------------------------------|---|
| Intel Nehalem ファミリー | vmx,nx,model_Nehalem:Nehalem:x86_64 |
| Secure Intel Nehalem ファミリー | vmx,spec_ctrl,ssbd,model_Nehalem:Nehalem,+spec-ctrl,+ssbd:x86_64 |
| Intel Westmere ファミリー | aes,vmx,nx,model_Westmere:Westmere:x86_64 |
| Secure Intel Westmere ファミリー | aes,vmx,spec_ctrl,ssbd,model_Westmere:Westmere,+pcid,+spec-ctrl,+ssbd:x8_64 |
| Intel SandyBridge ファミリー | vmx,nx,model_SandyBridge:SandyBridge:x86_64 |
| Secure Intel SandyBridge ファミリー | vmx,spec_ctrl,ssbd,md_clear,model_SandyBridge:SandyBridge,+pcid,+spec-ctrl,+ssbd,+md-clear:x86_64 |
| Intel IvyBridge ファミリー | vmx,nx,model_IvyBridge:IvyBridge:x86_64 |
| Secure Intel IvyBridge ファミリー | vmx,spec_ctrl,ssbd,md_clear,model_IvyBridge:IvyBridge,+pcid,+spec-ctrl,+ssbd,+md-clear:x86_64 |
| Intel Haswell ファミリー | vmx,nx,model_Haswell-noTSX:Haswell-noTSX:x86_64 |
| Secure Intel Haswell ファミリー | vmx,spec_ctrl,ssbd,md_clear,model_Haswell-noTSX:Haswell-noTSX,+spec-ctrl,+ssbd,+md-clear:x86_64 |

| CPU タイプ名 | CPU プロパティ |
|--|--|
| Intel Broadwell ファミリー | vmx,nx,model_Broadwell-noTSX:Broadwell-noTSX:x86_64 |
| Secure Intel Broadwell ファミリー | vmx,spec_ctrl,ssbd,md_clear,model_Broadwell-noTSX:Broadwell-noTSX,+spec-ctrl,+ssbd,+md-clear:x86_64 |
| Intel Skylake クライアントファミリー | vmx,nx,model_Skylake-Client:Skylake-Client,-hle,-rtm,-mpx:x86_64 |
| 安全な Intel Skylake クライアントファミリー | vmx,ssbd,md_clear,model_Skylake-Client-noTSX-IBRS:Skylake-Client-noTSX-IBRS,+ssbd,+md-clear,-mpx:x86_64 |
| Intel Skylake Server ファミリー | vmx,nx,model_Skylake-Server:Skylake-Server,-hle,-rtm,-mpx:x86_64 |
| Secure Intel Skylake Server ファミリー | vmx,ssbd,md_clear,model_Skylake-Server-noTSX-IBRS:Skylake-Server-noTSX-IBRS,+ssbd,+md-clear,-mpx:x86_64 |
| Intel Cascadelake Server Family | vmx,model_Cascadelake-Server:Cascadelake-Server,-hle,-rtm,-mpx:x86_64 |
| Secure Intel Cascadelake Server Family | vmx,model_Cascadelake-Server-noTSX:Cascadelake-Server-noTSX,-mpx:x86_64 |
| Intel Icelake Server ファミリー | vmx,model_Icelake-Server-noTSX:Icelake-Server-noTSX,-mpx:x86_64 |
| Secure Intel Icelake Server ファミリー | vmx,arch-capabilities,rdctl-no,ibrs-all,skip-lldfl-vmentry,mds-no,pschange-mc-no,taa-no,model_Icelake-Server-noTSX:Icelake-Server-noTSX,+arch-capabilities,+rdctl-no,+ibrs-all,+skip-lldfl-vmentry,+mds-no,+pschange-mc-no,+taa-no,-mpx:x86_64 |
| AMD Opteron G4 | svm,nx,model_Opteron_G4:Opteron_G4:x86_64 |
| AMD Opteron G5 | svm,nx,model_Opteron_G5:Opteron_G5:x86_64 |
| AMD EPYC | svm,nx,model_EPYC:EPYC:x86_64 |
| Secure AMD EPYC | svm,nx,ibpb,ssbd,model_EPYC:EPYC,+ibpb,+virt-ssbd:x86_64 |

| CPU タイプ名 | CPU プロパティ |
|------------------|---------------------------------------|
| IBM POWER8 | powernv,model_POWER8:POWER8:ppc64 |
| IBM POWER9 | powernv,model_POWER9:POWER9:ppc64 |
| IBM z114, z196 | sie,model_z196-base:z196-base:s390x |
| IBM zBC12, zEC12 | sie,model_zEC12-base:zEC12-base:s390x |
| IBM z13s, z13 | sie,model_z13-base:z13-base:s390x |
| IBM z14 | sie,model_z14-base:z14-base:s390x |

付録H 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.