Red Hat Virtualization 4.3

管理ガイド

Red Hat Virtualization の管理タスク

Last Updated: 2023-02-24

Red Hat Virtualization 4.3 管理ガイド

Red Hat Virtualization の管理タスク

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

法律上の通知

概要

本書では、Red Hat Virtualization の管理者に関連する情報および手順について説明します。

目次

パート I. RED HAT VIRTUALIZATION 環境の管理と保守

Red Hat Virtualization 環境の稼働を維持するには管理者が必要です。管理者のタスクには以下が含まれます。

- ホストや仮想マシンなどの物理リソースおよび仮想リソースの管理。これには、ホストのアップグレードおよび追加、ドメインのインポート、外部ハイパーバイザーで作成された仮想マシンの変換、および仮想マシンプールの管理が含まれます。
- ホストのいずれかに対する極端な負荷、メモリーやディスク容量不足、必要なアクション (仮想マシンをシャットダウンして仮想マシンの別ホストへの移行して負荷を軽減したりリソースを解放するなど) などの潜在的な問題について、全体的なシステムリソースのモニターリングを行います。
- 仮想マシンの新しい要件に対応します (たとえば、オペレーティングシステムのアップグレードまたはより多くのメモリーの割り当てなど)。
- タグを使用したカスタムオブジェクトプロパティーの管理。
- パブリックブックマーク として保存された検索の管理。
- ユーザー設定の管理とパーミッションレベルの設定。
- システム機能全体の特定ユーザーまたは仮想マシンのトラブルシューティング。
- 一般および特定レポートの生成。

第1章 グローバル設定

Administration → Configure をクリックしてアクセスします。**Configure** ウィンドウでは、ユーザー、ロール、システムパーミッション、スケジューリングポリシー、インスタンスタイプ、MAC アドレスプールなどの Red Hat Virtualization 環境のグローバルリソースを複数設定できます。このウィンドウでは、ユーザーが環境のリソースと対話する方法をカスタマイズし、複数のクラスターに適用できるオプションを設定する一元的な場所を提供します。

1.1. ロール

ロールは、Red Hat Virtualization Manager から設定できる事前定義された権限のセットです。ロールは、データセンター内の異なるレベルのリソースや、特定の物理リソースおよび仮想リソースに対するアクセスおよび管理のパーミッションを提供します。

マルチレベル管理では、コンテナーオブジェクトに適用されるパーミッションは、そのコンテナー内のすべての個別オブジェクトにも適用されます。たとえば、特定のホスト上のユーザーにホスト管理者ロールが割り当てられた場合、そのユーザーは利用可能なホスト操作のいずれかを実行する権限を得ますが、割り当てられたホスト上でのみ実行できます。ただし、ホスト管理者ロールがデータセンターのユーザーに割り当てられている場合、ユーザーはデータセンターのクラスター内の全ホストでホスト操作を実行するパーミッションを取得します。

1.1.1. 新しいロールの作成

必要なロールが Red Hat Virtualization のデフォルトロールリストにない場合は、新しいロールを作成して、目的に合わせてカスタマイズできます。

新しいロールの作成

1. **Administration → Configure** をクリックして **Configure** ウィンドウを開きます。**Roles** タブはデフォルトで選択され、デフォルトのユーザーおよび管理者ロールおよびカスタムロールのリストが表示されます。
2. **New** をクリックします。
3. 新規ロールの **Name** および **Description** を入力します。
4. **Account Type** に **Admin** または **User** のいずれかを選択します。
5. **Expand All** または **Collapse All** ボタンを使用して、**Check Boxes to Allow Action** リストに記載されているオブジェクトのパーミッションの表示を拡大または縮小します。また、各オブジェクトのオプションを展開したり、折りたたんだりすることもできます。
6. それぞれのオブジェクトについて、設定しているロールを許可または拒否するアクションを選択または消去します。
7. **OK** をクリックして変更を適用します。ロールの一覧に新しいロールが表示されます。

1.1.2. ロールの編集またはコピー

作成したロールの設定を変更できますが、デフォルトのロールを変更することはできません。デフォルトのロールを変更するには、そのロールのクローンを作成して、要件に合わせて変更します。

ロールの編集またはコピー

1. **Administration → Configure** をクリックして **Configure** ウィンドウを開きます。ウィンドウには、デフォルトの User および Administrator ロールのリストとカスタムロールが表示されます。

2. 変更するロールを選択します。**Edit** をクリックして **Edit Role** ウィンドウを開くか、**Copy** をクリックして **Copy Role** ウィンドウを開きます。

3. 必要に応じて、ロールの **Name** および **Description** を編集します。

4. **Expand All** または **Collapse All** ボタンを使用して、リストされているオブジェクトのパーミッションの表示を拡大または縮小します。また、各オブジェクトのオプションを展開したり、折りたたんだりすることもできます。

5. それぞれのオブジェクトについて、編集するロールを許可または拒否するアクションを選択または消去します。

6. **OK** をクリックして、加えた変更を適用します。

1.1.3. ユーザーロールと承認の例

以下の例は、本章で説明する承認システムの異なる機能を使用して、さまざまなシナリオに対して承認制御を適用する方法を示しています。

例1.1 クラスターパーミッション

Sarah は、ある企業の経理部門のシステム管理者です。彼女の部署のすべての仮想リソースは、**Accounts** という名前の Red Hat Virtualization **クラスター**の下に編成されています。彼女には Accounts クラスターの **ClusterAdmin** ロールが割り当てられています。これにより、仮想マシンはクラスターの子オブジェクトであるため、彼女はクラスター内のすべての仮想マシンを管理できます。仮想マシンの管理には、ディスクなどの仮想リソースの編集、追加、削除、およびスナップショットの作成などが含まれます。このクラスターの外部にあるリソースを管理することはできません。**ClusterAdmin** は管理者ロールであるため、管理ポータルまたは VM ポータルを使用してこれらのリソースを管理できます。

例1.2 VM PowerUser パーミッション

John は、経理部のソフトウェア開発者です。彼は仮想マシンを使用してソフトウェアを構築し、テストします。Sarah は、John に **johndesktop** という仮想デスクトップを作成しました。John は、**johndesktop** 仮想マシンで **UserVmManager** ロールが割り当てられています。これにより、VM ポータルを使用してこの単一仮想マシンにアクセスすることができます。彼は **UserVmManager** のパーミッションを持っているため、仮想マシンを変更できます。**UserVmManager** はユーザーロールであるため、管理ポータルを使用できません。

例1.3 データセンターパワーユーザーロールパーミッション

Penelope はオフィスマネージャーです。自分の仕事に加えて、面接の日程調整やリファレンスチェックのフォローアップなど、人事マネージャーの採用業務を手伝うこともあります。会社の方針により、Penelope は採用業務に特定のアプリケーションを使用する必要があります。

Penelope はオフィス管理用に自分のマシンを持っていますが、採用アプリケーションを実行するために別の仮想マシンを作成したいと考えています。彼女には、新しい仮想マシンが設置されるデータセンターの **PowerUserRole** パーミッションが割り当てられています。これは、新しい仮想マシンを作成することにより、ストレージドメインでの仮想ディスクの作成など、データセンター内の複数のコンポーネントに変更を加える必要があるためです。

これは、Penelope に **DataCenterAdmin** 権限を割り当てることとは違うことに注意してください。データセンターの PowerUser として、Penelope は VM ポータルにログインして、データセンター内の仮想マシン固有のアクションを実行できます。彼女は、ホストまたはストレージをデータセンターに割り当てるなど、データセンターレベルの操作を実行できません。

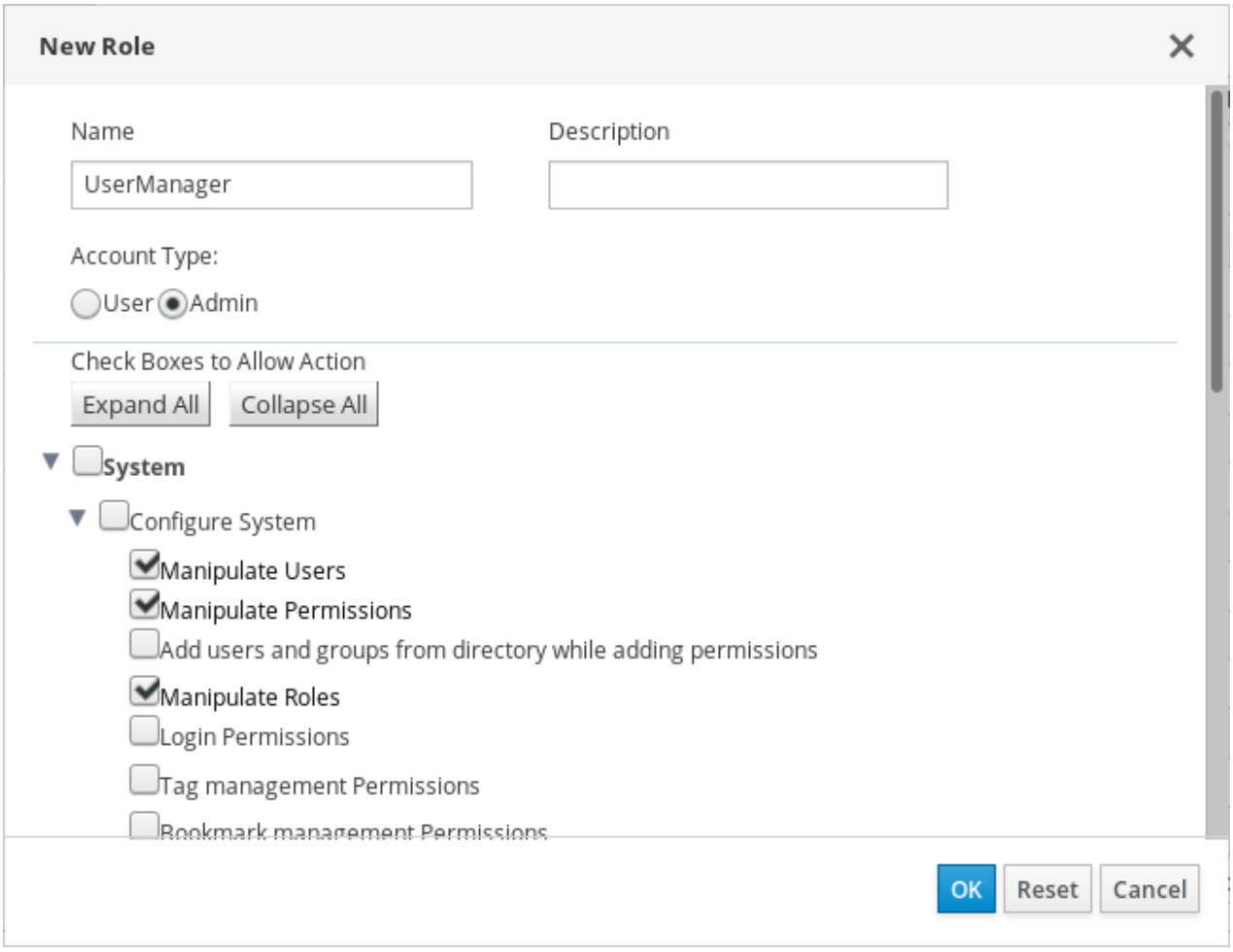
例1.4 ネットワーク管理者のパーミッション

Chris は、IT 部門のネットワーク管理者です。彼女の日常的な仕事は、部内の Red Hat Virtulization 環境におけるネットワークの作成、操作、削除などです。彼女には、リソースおよび各リソースのネットワークにおける管理者権限が必要です。たとえば、Chris が IT 部門のデータセンターの**NetworkAdmin** 権限を持っている場合、データセンター内のネットワークを追加および削除したり、データセンターに属するすべての仮想マシンのネットワークをアタッチおよびデタッチすることができます。

例1.5 カスタムロールパーミッション

レイチェルは IT 部門に所属し、Red Hat Virtualization のユーザーアカウント管理を担当しています。彼女には、ユーザーアカウントを追加し、適切なロールおよびパーミッションを割り当てる権限が必要です。彼女自身は仮想マシンを使用せず、ホスト、仮想マシン、クラスター、データセンターの管理にアクセスしてはいけません。彼女にこのような特定のパーミッションを与える組み込みのロールはありません。レイチェルの役職に適したパーミッションセットを定義するために、カスタムロールを作成する必要があります。

図1.1 UserManager カスタムロール



上記の **UserManager** のカスタムロールは、ユーザー、パーミッション、およびロールの操作を可能

にします。これらのアクションは、図1.3「Red Hat Virtualization オブジェクトの階層」 に示す階層の上位オブジェクトである **System** の下に整理されています。これは、システム内のすべてのオブジェクトに適用されることを意味します。ロールの **Account Type** は **Admin** に設定されます。つまり、このロールが割り当てられると、Rachel は Administration Portal と VM Portal の両方を使用できるようになります。

1.2. システムパーミッション

パーミッションにより、ユーザーはオブジェクトに対してアクションを実行できます。オブジェクトは個別のオブジェクトまたはコンテナーオブジェクトのいずれかになります。コンテナーオブジェクトに適用されるパーミッションは、そのコンテナーのすべてのメンバーにも適用されます。

図1.2 パーミッションおよびロール

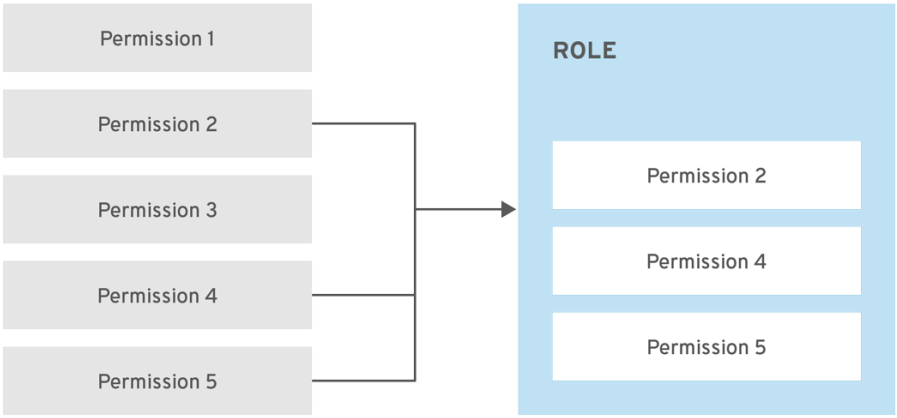
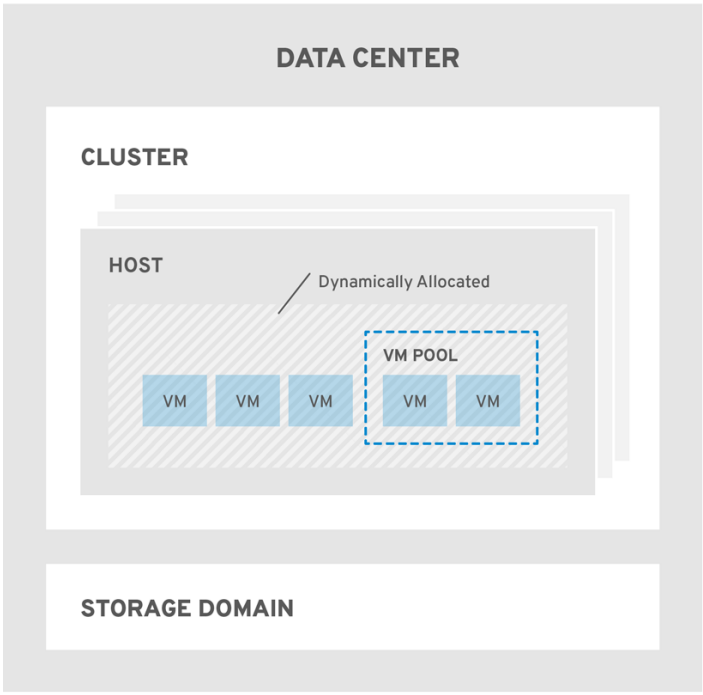


図1.3 Red Hat Virtualization オブジェクトの階層



1.2.1. ユーザープロパティー

ロールおよびパーミッションはユーザーのプロパティーです。ロールとは、異なるレベルの物理および仮想リソースへのアクセスを許可する、事前定義された一連の権限のことです。マルチレベル管理では、パーミッションを細かく階層化できます。たとえば、データセンター管理者はデータセンター内の全オブジェクトを管理するパーミッションを持ち、ホスト管理者は1つの物理ホストのシステム管理者パーミッションを持ちます。あるユーザーは、単一の仮想マシンを使用するパーミッションを持っていても、仮想マシンの設定を変更できません。一方、別のユーザーは、仮想マシンのシステムパーミッションを割り当てることができます。

1.2.2. ユーザーおよび管理者ロール

Red Hat Virtualization は、システム全体のパーミッションを持つ管理者から、1台の仮想マシンにアクセスできるエンドユーザーまで、事前設定されたさまざまなロールを提供します。デフォルトのロールを変更または削除することはできませんが、そのロールのクローンを作成してカスタマイズしたり、要件に合わせて新しいロールを作成したりできます。以下の 2 つのタイプがあります。

- 管理者ロール: 物理リソースおよび仮想リソースを管理するための **管理ポータル** へのアクセスを許可します。管理者ロールは、VM ポータルで実行するアクションのパーミッションを付与しますが、ユーザーが VM ポータルで見ることができる内容には影響しません。

- ユーザーロール: **VM ポータル** にアクセスして、仮想マシンおよびテンプレートを管理し、アクセスできるようにします。ユーザーロールは、ユーザーが VM ポータルで表示できる内容を決定します。管理者ロールを持つユーザーに付与されるパーミッションは、そのユーザーが VM ポータルで利用できるアクションに反映されます。

1.2.3. ユーザーロールの概要

以下の表は、VM ポータルで仮想マシンにアクセスし、設定するパーミッションを付与する基本的なユーザーロールを説明しています。

表1.1 Red Hat Virtualization ユーザーロール: 基本

ロール	権限	注記
UserRole	仮想マシンおよびプールにアクセスし、使用できる。	VM ポータルへのログイン、割り当てられた仮想マシンやプールの使用、仮想マシンの状態や詳細の表示が可能。
PowerUserRole	仮想マシンおよびテンプレートを作成および管理できる。	このロールを **Configure** ウィンドウで環境全体のユーザーまたは特定のデータセンターやクラスターのユーザーに適用します。たとえば、PowerUserRole がデータセンターレベルに適用されると、PowerUser はデータセンターで仮想マシンおよびテンプレートを作成できます。
UserVmManager	仮想マシンのシステム管理者。	仮想マシンの管理、スナップショットの作成と使用が可能。VM ポータルで仮想マシンを作成したユーザーには、そのマシンの UserVmManager ロールが自動的に割り当てられます。

以下の表は、VM ポータルのリソースに対するパーミッションの細かな調整を可能にする高度なユーザーロールについて説明しています。

表1.2 Red Hat Virtualization のユーザーロール - 高度

ロール	権限	注記
UserTemplateBasedVm	テンプレートのみを使用できる限定的な権限。	テンプレートを使用して仮想マシンを作成できます。
DiskOperator	仮想ディスクユーザー。	仮想ディスクの使用、表示、編集が可能です。仮想ディスクが接続されている仮想マシンを使用するパーミッションを継承します。
VmCreator	VM ポータルで仮想マシンを作成できる。	このロールは特定の仮想マシンには適用されません。**Configure** ウィンドウで環境全体のユーザーにこのロールを適用します。または、特定のデータセンターまたはクラスターにこのロールを適用することもできます。このロールをクラスターに適用する場合、データセンター全体または特定のストレージドメインに DiskCreator ロールを適用する必要もあります。
TemplateCreator	割り当てられたリソース内で仮想マシンテンプレートを作成、編集、管理、および削除できる。	このロールは特定のテンプレートには適用されません。**Configure** ウィンドウで環境全体のユーザーにこのロールを適用します。また、特定のデータセンター、クラスター、またはストレージドメインにこのロールを適用することもできます。
DiskCreator	割り当てられたクラスターまたはデータセンター内の仮想ディスクを作成、編集、管理、および削除できる。	このロールは特定の仮想ディスクには適用されません。**Configure** ウィンドウで環境全体のユーザーにこのロールを適用します。また、特定のデータセンターまたはストレージドメインにこのロールを適用することもできます。
TemplateOwner	テンプレートの編集および削除、テンプレートのユーザーパーミッションの割り当ておよび管理が可能。	このロールは、テンプレートを作成したユーザーに自動的に割り当てられます。テンプレートに **TemplateOwner** パーミッションを持たない他のユーザーは、そのテンプレートを表示または使用することができません。

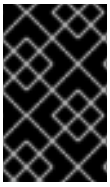
ロール	権限	注記
VnicProfileUser	仮想マシンおよびテンプレートの論理ネットワークおよびネットワークインターフェイスユーザー。	特定の論理ネットワークからネットワークインターフェイスを接続または切断できます。

1.2.4. 管理者ロールの概要

以下の表は、管理ポータルのリソースにアクセスおよび設定するパーミッションを付与する基本的な管理者ロールについて説明しています。

表1.3 Red Hat Virtualization システム管理者ロール - 基本

ロール	権限	注記
SuperUser	Red Hat Virtualization 環境のシステム管理者	すべてのオブジェクトおよびレベルでの完全なパーミッションを持ち、全データセンターで全オブジェクトを管理できます。
ClusterAdmin	クラスター管理者。	特定のクラスター下にある全オブジェクトの管理パーミッションを持ちます。
DataCenterAdmin	データセンター管理者。	特定のデータセンターの下にある、ストレージを除くすべてのオブジェクトの管理権限を保有しています。



重要

ディレクトリーサーバーの管理ユーザーを、Red Hat Virtualization の管理ユーザーとして使用しないでください。Red Hat Virtualization の管理ユーザーとして使用するように、ディレクトリーサーバーにユーザーを作成します。

以下の表は、管理者ポータルのリソースに対するパーミッションの細かな調整を可能にする高度な管理者ロールについて説明しています。

表1.4 Red Hat Virtualization システム管理者ロール - 高度

ロール	権限	注記
TemplateAdmin	仮想マシンテンプレートの管理者。	テンプレートのストレージドメインやネットワークの詳細を作成、削除、設定したり、ドメイン間でテンプレートを移動できます。

ロール	権限	注記
StorageAdmin	ストレージ管理者。	割り当てられたストレージドメインを作成、削除、設定、および管理できます。
HostAdmin	ホスト管理者。	特定のホストをアタッチ、削除、設定、および管理できます。
NetworkAdmin	ネットワーク管理者。	特定のデータセンターまたはクラスターのネットワークを設定および管理できます。データセンターまたはクラスターのネットワーク管理者は、クラスター内の仮想プールのネットワークパーミッションを継承します。
VmPoolAdmin	仮想プールのシステム管理者。	仮想プールを作成、削除、および設定できます。仮想プールユーザーを割り当ておよび削除し、プールの仮想マシンに基本操作を実行できます。
GlusterAdmin	Gluster Storage 管理者。	Gluster ストレージボリュームを作成、削除、設定、および管理できます。
VmImporterExporter	仮想マシンの管理者をインポートおよびエクスポートします。	仮想マシンをインポートおよびエクスポートできます。他のユーザーがエクスポートした仮想マシンおよびテンプレートをすべて表示できます。

1.2.5. 管理者またはユーザーロールのリソースへの割り当て

管理者またはユーザーロールをリソースに割り当て、ユーザーがそのリソースにアクセスしたり、管理したりできるようにします。

リソースへのロールの割り当て

1. リソースの名前を見つけ、クリックして詳細ビューを開きます。
2. **Permissions** タブをクリックして、割り当てられたユーザー、ユーザーのロール、および選択したリソースの継承されたパーミッションを一覧表示します。
3. **Add** をクリックします。
4. **Search** テキストボックスに既存のユーザーの名前またはユーザー名を入力し、**Go** をクリックします。表示された候補の中からユーザーを選択します。
5. **Role to Assign** ドロップダウンリストからロールを選択します。
6. **OK** をクリックします。

ユーザーは、そのリソースに対して有効になっているロールの継承されたパーミッションを持つようになります。

1.2.6. リソースからの管理者またはユーザーロールの削除

管理者またはユーザーのロールをリソースから削除すると、ユーザーはそのリソースのロールに関連付けられ継承されたパーミッションを失います。

リソースからのロールの削除

1. リソースの名前を見つけ、クリックして詳細ビューを開きます。
2. **Permissions** タブをクリックして、割り当てられたユーザー、ユーザーのロール、および選択したリソースの継承されたパーミッションを一覧表示します。
3. リソースから削除するユーザーを選択します。
4. **Remove** をクリックします。
5. **OK** をクリックします。

1.2.7. データセンターのシステムパーミッションの管理

システム管理者は **SuperUser** として、管理ポータルのすべての側面を管理します。他のユーザーに特定の管理ロールを割り当てることができます。このような制限された管理者ロールは、特定のリソースに制限される管理者権限をユーザーに付与する際に役立ちます。たとえば、**DataCenterAdmin** ロールは、割り当てられたデータセンターのストレージを除いて、そのデータセンターのみの管理者権限を持ち、**ClusterAdmin** は割り当てられたクラスターのみの管理者権限を持ちます。

データセンターの管理者は、特定のデータセンターのみのシステム管理ロールです。これは、各データセンターが管理者を必要とする複数のデータセンターを持つ仮想化環境で有用です。**DataCenterAdmin** ロールは階層モデルです。あるデータセンターのデータセンター管理者ロールを割り当てられたユーザーは、そのデータセンターのストレージを除くすべてのオブジェクトを管理することができます。ヘッダーバーの **Configure** ボタンを使用して、環境内のすべてのデータセンターにデータセンター管理者を割り当てます。

データセンターの管理者ロールでは、以下のアクションが許可されます。

- データセンターに関連付けられたクラスターの作成と削除。
- データセンターに関連付けられたホスト、仮想マシン、およびプールを追加および削除。
- データセンターに関連付けられた仮想マシンのユーザーパーミッションの編集。

注記

ロールやパーミッションは、既存のユーザーにのみ割り当てることができます。

既存のシステム管理者を削除し、新しいシステム管理者を追加すると、データセンターのシステム管理者を変更できます。

1.2.8. データセンター管理者ロールの概要

データセンターのパーミッションロール

以下の表は、データセンターの管理に適用される管理者ロールおよび権限を示しています。

表1.5 Red Hat Virtualization システム管理者ロール

ロール	権限	注記
DataCenterAdmin	データセンター管理者	クラスター、ホスト、テンプレート、仮想マシンなど、特定のデータセンター内の物理リソースおよび仮想リソースすべての作成、削除、管理が可能です。
NetworkAdmin	ネットワーク管理者	特定のデータセンターのネットワークを設定および管理できます。データセンターのネットワーク管理者は、データセンター内の仮想マシンのネットワークパーミッションも継承します。

1.2.9. クラスターのシステムパーミッションの管理

システム管理者は **SuperUser** として、管理ポータルのすべての側面を管理します。他のユーザーに特定の管理ロールを割り当てることができます。このような制限された管理者ロールは、特定のリソースに制限される管理者権限をユーザーに付与する際に役立ちます。たとえば、**DataCenterAdmin** ロールは、割り当てられたデータセンターのストレージを除いて、そのデータセンターのみの管理者権限を持ち、**ClusterAdmin** は割り当てられたクラスターのみの管理者権限を持ちます。

クラスターの管理者は、特定のクラスターのみのシステム管理ロールです。これは、複数のクラスターを持つデータセンターで、各クラスターにシステム管理者が必要な場合に有効です。**ClusterAdmin** ロールは階層モデルです。あるクラスターのクラスター管理者ロールを割り当てられたユーザーは、クラスターのすべてのオブジェクトを管理できます。ヘッダーバーの **Configure** ボタンを使用して、環境のすべてのクラスターにクラスター管理者を割り当てます。

クラスター管理者ロールは以下のアクションを許可します。

- 関連付けられたクラスターの作成および削除。
- クラスターに関連付けられたホスト、仮想マシン、およびプールの追加および削除。
- クラスターに関連付けられた仮想マシンのユーザーパーミッションを編集します。

注記

ロールやパーミッションは、既存のユーザーにのみ割り当てることができます。

また、既存のシステム管理者を削除し、新しいシステム管理者を追加すると、クラスターのシステム管理者を変更できます。

1.2.10. クラスター管理者ロールの概要

クラスターパーミッションロール

以下の表は、クラスターの管理に適用される管理者ロールおよび権限について説明しています。

表1.6 Red Hat Virtualization システム管理者ロール

ロール	権限	注記
ClusterAdmin	クラスター管理者	ホスト、テンプレート、および仮想マシンなど、特定のクラスター内の物理リソースおよび仮想リソースをすべて使用、作成、削除、管理することができます。ディスプレイネットワークの指定や、ネットワークを必須とマークするなど、クラスター内でネットワークプロパティーを設定できます。 ただし、**ClusterAdmin** には、ネットワークをクラスターにアタッチまたはデタッチするパーミッションがないため、**NetworkAdmin** パーミッションが必要です。
NetworkAdmin	ネットワーク管理者	特定のクラスターのネットワークを設定および管理できます。クラスターのネットワーク管理者は、データセンター内のクラスターネットワークパーミッションも継承します。

1.2.11. ネットワークのシステムパーミッションの管理

システム管理者は **SuperUser** として、管理ポータルのすべての側面を管理します。他のユーザーに特定の管理ロールを割り当てることができます。このような制限された管理者ロールは、特定のリソースに制限される管理者権限をユーザーに付与する際に役立ちます。たとえば、**DataCenterAdmin** ロールは、割り当てられたデータセンターのストレージを除いて、そのデータセンターのみの管理者権限を持ち、**ClusterAdmin** は割り当てられたクラスターのみの管理者権限を持ちます。

ネットワーク管理者は、特定のネットワークまたはデータセンター、クラスター、ホスト、仮想マシン、またはテンプレートにあるすべてのネットワークに適用できるシステム管理ロールです。ネットワークユーザーは、特定の仮想マシンまたはテンプレート上のネットワークの表示やアタッチなど、制限された管理ロールを実行できます。ヘッダーバーの **Configure** ボタンを使用して、環境内の全ネットワークにネットワーク管理者を割り当てできます。

ネットワーク管理者ロールは以下のアクションを許可します。

- ネットワークの作成、編集、および削除。
- ポートミラーリングの設定など、ネットワークの設定を編集。
- クラスターや仮想マシンを含むリソースからのネットワークをアタッチおよびデタッチ。

ネットワークを作成するユーザーには、作成されたネットワークに **NetworkAdmin** パーミッションが自動的に割り当てられます。また、既存の管理者を削除し、新しい管理者を追加すると、ネットワークの管理者を変更できます。

1.2.12. ネットワーク管理者およびユーザーロールの概要

ネットワークパーミッションロール

以下の表は、ネットワークの管理に適用される管理者およびユーザーロールと権限について説明しています。

表1.7 Red Hat Virtualization ネットワーク管理者およびユーザーロール

ロール	権限	注記
NetworkAdmin	データセンター、クラスター、ホスト、仮想マシン、またはテンプレートのネットワーク管理者。ネットワークを作成するユーザーには、作成されたネットワークに **NetworkAdmin** パーミッションが自動的に割り当てられます。	特定のデータセンター、クラスター、ホスト、仮想マシン、またはテンプレートのネットワークを設定および管理できます。データセンターまたはクラスターのネットワーク管理者は、クラスター内の仮想プールのネットワークパーミッションを継承します。仮想マシンのネットワークにポートミラーリングを設定するには、ネットワークに **NetworkAdmin** ロールを、仮想マシンに **UserVmManager** ロールを適用します。
VnicProfileUser	仮想マシンおよびテンプレートの論理ネットワークおよびネットワークインターフェイスユーザー。	特定の論理ネットワークからネットワークインターフェイスを接続または切断できます。

1.2.13. ホストのシステムパーミッションの管理

システム管理者は **SuperUser** として、管理ポータルのすべての側面を管理します。他のユーザーに特定の管理ロールを割り当てることができます。このような制限された管理者ロールは、特定のリソースに制限される管理者権限をユーザーに付与する際に役立ちます。たとえば、**DataCenterAdmin** ロールは、割り当てられたデータセンターのストレージを除いて、そのデータセンターのみの管理者権限を持ち、**ClusterAdmin** は割り当てられたクラスターのみの管理者権限を持ちます。

ホスト管理者は、特定のホストのみのシステム管理ロールです。これは、複数のホストを持つクラスターで、各ホストにシステム管理者が必要な場合に有効です。ヘッダーバーの **Configure** ボタンを使用して、環境内の全ホストにホスト管理者を割り当てできます。

ホスト管理者ロールは以下のアクションを許可します。

- ホストの設定編集。
- 論理ネットワークの設定。
- ホストを削除。

また、既存のシステム管理者を削除し、新しいシステム管理者を追加すると、ホストのシステム管理者を変更できます。

1.2.14. ホスト管理者ロールの概要

ホストパーミッションロール

以下の表は、ホスト管理に適用される管理者ロールおよび権限について説明しています。

表1.8 Red Hat Virtualization システム管理者ロール

ロール	権限	注記
HostAdmin	ホスト管理者	特定のホストを設定、管理、および削除できます。特定のホストでネットワーク関連の操作も実行できます。

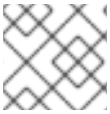
1.2.15. ストレージドメインのシステムパーミッションの管理

システム管理者は **SuperUser** として、管理ポータルのすべての側面を管理します。他のユーザーに特定の管理ロールを割り当てることができます。このような制限された管理者ロールは、特定のリソースに制限される管理者権限をユーザーに付与する際に役立ちます。たとえば、**DataCenterAdmin** ロールは、割り当てられたデータセンターのストレージを除いて、そのデータセンターのみの管理者権限を持ち、**ClusterAdmin** は割り当てられたクラスターのみの管理者権限を持ちます。

ストレージ管理者は、特定のストレージドメインのみのシステム管理ロールです。これは、複数のストレージドメインを持つデータセンターで、各ストレージドメインにシステム管理者が必要な場合に有効です。ヘッダーバーの **Configure** ボタンを使用して、環境内のすべてのストレージドメインにストレージ管理者を割り当てます。

ストレージドメイン管理者ロールは、以下のアクションを許可します。

- ストレージドメインの設定の編集。
- ストレージドメインのメンテナンスモードへの切り替え。
- ストレージドメインの削除。



注記

ロールやパーミッションは、既存のユーザーにのみ割り当てることができます。

また、既存のシステム管理者を削除し、新しいシステム管理者を追加すると、ストレージドメインのシステム管理者を変更できます。

1.2.16. ストレージ管理者ロールの概要

ストレージドメインパーミッションロール

以下の表は、ストレージドメインの管理に適用される管理者ロールおよび権限について説明しています。

表1.9 Red Hat Virtualization システム管理者ロール

ロール	権限	注記
StorageAdmin	ストレージ管理者	特定のストレージドメインを作成、削除、設定、および管理できます。
GlusterAdmin	Gluster Storage 管理者	Gluster ストレージボリュームを作成、削除、設定、および管理できます。

1.2.17. 仮想マシンプールのシステムパーミッションの管理

システム管理者は **SuperUser** として、管理ポータルのすべての側面を管理します。他のユーザーに特定の管理ロールを割り当てることができます。このような制限された管理者ロールは、特定のリソースに制限される管理者権限をユーザーに付与する際に役立ちます。たとえば、**DataCenterAdmin** ロールは、割り当てられたデータセンターのストレージを除いて、そのデータセンターのみの管理者権限を持ち、**ClusterAdmin** は割り当てられたクラスターのみの管理者権限を持ちます。

仮想マシンプールの管理者は、データセンター内の仮想マシンプールのシステム管理ロールです。このロールは、特定の仮想マシンプール、データセンター、または仮想化環境全体に適用できます。これは、異なるユーザーが特定の仮想マシンプールリソースを管理できるようにするのに役立ちます。

仮想マシンプールの管理者ロールは、以下のアクションを許可します。

- プールの作成、編集、および削除。
- プールからの仮想マシンを追加およびデタッチ。

注記

ロールやパーミッションは、既存のユーザーにのみ割り当てることができます。

1.2.18. 仮想マシンプール管理者ロールの概要

プールパーミッションロール

以下の表は、プール管理に適用される管理者ロールおよび権限について説明しています。

表1.10 Red Hat Virtualization システム管理者ロール

ロール	権限	注記
VmPoolAdmin	仮想プールのシステム管理者ロール。	仮想プールを作成、削除、および設定できます。仮想プールユーザーを割り当ておよび削除し、仮想マシンに基本操作を実行できます。
ClusterAdmin	クラスター管理者	特定のクラスター内のすべての仮想マシンプールを使用、作成、削除、および管理できます。

1.2.19. 仮想ディスクのシステムパーミッションの管理

システム管理者は **SuperUser** として、管理ポータルのすべての側面を管理します。他のユーザーに特定の管理ロールを割り当てることができます。このような制限された管理者ロールは、特定のリソースに制限される管理者権限をユーザーに付与する際に役立ちます。たとえば、**DataCenterAdmin** ロールは、割り当てられたデータセンターのストレージを除いて、そのデータセンターのみの管理者権限を持ち、**ClusterAdmin** は割り当てられたクラスターのみの管理者権限を持ちます。

Red Hat Virtualization Manager では、デフォルトの仮想ディスクユーザーロールが 2 つ提供されますが、デフォルトの仮想ディスク管理者ロールはありません。ユーザーロールの 1 つである **DiskCreator** ロールは、VM ポータルから仮想ディスクの管理を可能にします。このロールは、特定の仮想マシン、データセンター、特定のストレージドメイン、または仮想化環境全体に適用することができます。これは、異なるユーザーが異なる仮想リソースを管理できるようにするのに役立ちます。

仮想ディスク作成者ロールは、以下のアクションを許可します。

- 仮想マシンまたは他のリソースに関連付けられた仮想ディスクの作成、編集、および削除。
- 仮想ディスクのユーザーパーミッションを編集します。

注記

ロールやパーミッションは、既存のユーザーにのみ割り当てることができます。

1.2.20. 仮想ディスクユーザーロールの概要

仮想ディスクユーザーパーミッションロール

以下の表は、VM ポータルでの仮想ディスクの使用および管理に適用されるユーザーロールおよび権限について説明しています。

表1.11 Red Hat Virtualization システム管理者ロール

ロール	権限	注記
DiskOperator	仮想ディスクユーザー。	仮想ディスクの使用、表示、編集が可能です。仮想ディスクが接続されている仮想マシンを使用するパーミッションを継承します。
DiskCreator	割り当てられたクラスターまたはデータセンター内の仮想ディスクを作成、編集、管理、および削除できる。	このロールは特定の仮想ディスクには適用されません。**Configure** ウィンドウで環境全体のユーザーにこのロールを適用します。また、特定のデータセンター、クラスター、またはストレージドメインにこのロールを適用することもできます。

1.2.21. レガシー SPICE 暗号の設定

SPICE コンソールでは、デフォルトで FIPS 準拠の暗号化を行い、暗号文字列を使用します。デフォルトの SPICE 暗号文字列は **kECDHE+FIPS:kDHE+FIPS:kRSA+FIPS:!eNULL:!aNULL** です。

通常、この文字列で十分です。ただし、古いオペレーティングシステムまたは SPICE クライアントの仮想マシンがあり、そのうちのいずれかが FIPS 準拠の暗号化に対応していない場合は、弱い暗号文字列を使用する必要があります。そうしないと、新規クラスターまたは新規ホストを既存のクラスターにインストールし、その仮想マシンへの接続を試みると、接続のセキュリティーエラーが発生します。

Ansible Playbook を使用して暗号文字列を変更できます。

暗号文字列の変更

1. Manager マシンで、**/usr/share/ovirt-engine/playbooks** ディレクトリーにファイルを作成します。以下に例を示します。

```
# vim /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

2. ファイルに以下を入力し、保存します。

```
name: oVirt - setup weaker SPICE encryption for old clients
hosts: hostname
vars:
  host_deploy_spice_cipher_string: 'DEFAULT:-RC4:-3DES:-DES'
roles:
  - ovirt-host-deploy-spice-encryption
```

3. 作成したファイルを実行します。

```
# ansible-playbook -l hostname /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

または、以下のように変数 **host_deploy_spice_cipher_string** に **--extra-vars** オプションを指定して、Ansible Playbook **ovirt-host-deploy** でホストを再設定できます。

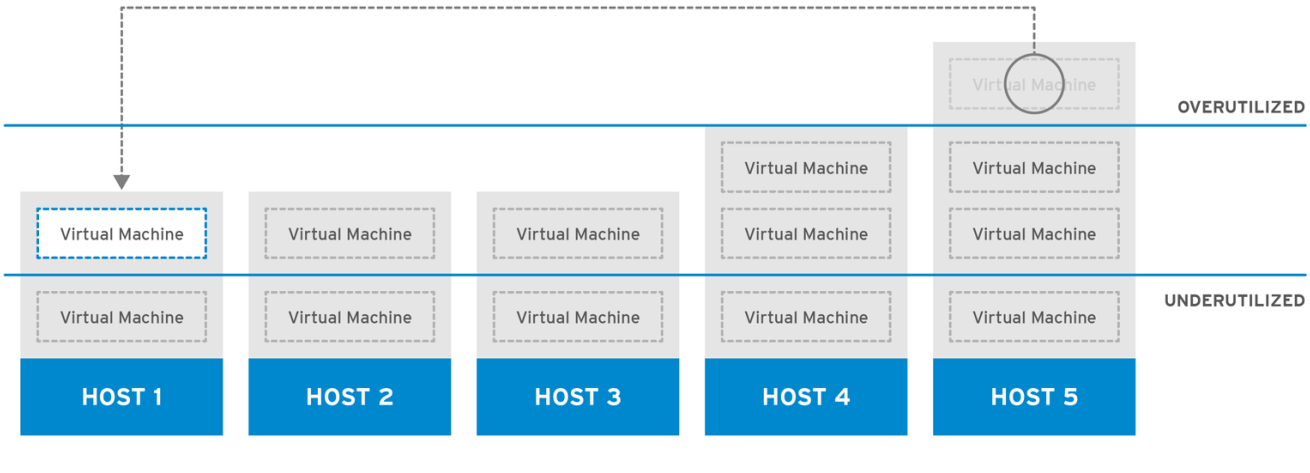
```
# ansible-playbook -l hostname \
  --extra-vars host_deploy_spice_cipher_string="DEFAULT:-RC4:-3DES:-DES" \
  /usr/share/ovirt-engine/playbooks/ovirt-host-deploy.yml
```

1.3. スケジューリングポリシー

スケジューリングポリシーは、スケジューリングポリシーが適用されるクラスター内のホスト間で仮想マシンが分散されるロジックを定義するルールのセットです。スケジューリングポリシーは、フィルター、重み付け、および負荷分散ポリシーの組み合わせにより、このロジックを決定します。フィルターモジュールはハード強制を適用し、そのフィルターで指定された条件を満たさないホストを除外します。加重モジュールはソフト強制を適用し、仮想マシンが実行できるクラスター内のホストを決定する際に考慮される要因の相対優先度を制御するために使用されます。

Red Hat Virtualization Manager には 5 つのデフォルトスケジューリングポリシー **Evenly_Distributed**、**Cluster_Maintenance**、**None**、**Power_Saving**、および **VM_Evenly_Distributed** があります。また、新しいスケジューリングポリシーを定義することで、仮想マシンの配布をきめ細かく制御することができます。スケジューリングポリシーに関わらず、CPU が過負荷状態のホストでは仮想マシンが起動しません。デフォルトでは、ホストの CPU が 5 分間 80% 以上の負荷がかかった場合に過負荷と判断されますが、この値はスケジューリングポリシーを使って変更できます。各スケジューリングポリシーのプロパティーに関する詳細は、「スケジューリングポリシー設定に関する説明」 を参照してください。

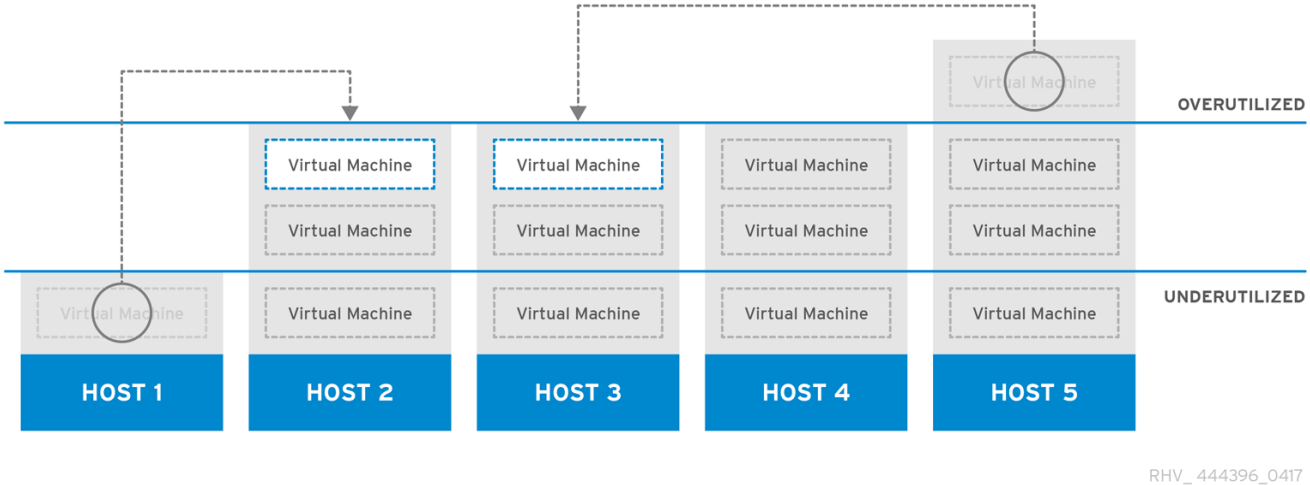
図1.4 Evenly Distributed スケジューリングポリシー



Evenly_Distributed スケジューリングポリシーは、クラスター内のすべてのノードでメモリーおよび CPU 処理の負荷を均等に分散します。ホストが定義された **CpuOverCommitDurationMinutes**、**HighUtilization**、または **MaxFreeMemoryForOverUtilized** に達した場合、ホストにアタッチされた追加の仮想マシンは起動しません。

VM_Evenly_Distributed スケジューリングポリシー仮想マシンは、仮想マシンの数に基づいてホスト間で均等に分散されます。**HighVmCount** よりも多くの仮想マシンを実行しているホストがあり、仮想マシン数が **MigrationThreshold** の範囲外であるホストが少なくとも 1 つ存在する場合、クラスターはアンバランスであると判断されます。

図1.5 Power Saving スケジューリングポリシー



Power_Saving スケジューリングポリシーは、利用可能なホストのサブセットにメモリーおよび CPU 処理の負荷を分散し、使用率の低いホストの消費電力を減らします。CPU 負荷が低稼働率の値を下回っている状態が定義された時間以上続いたホストは、すべての仮想マシンを他のホストに移行させ、電源を切れるようにします。ホストにアタッチされた追加の仮想マシンは、そのホストが定義された高使用率値に達した場合には起動しません。

仮想マシンの実行でホスト間で負荷やパワーを共有しないように、**None** ポリシーを設定します。これはデフォルトのモードです。仮想マシンが起動すると、メモリーと CPU 処理の負荷がクラスター内の全ホストに均等に分散されます。ホストが定義された **CpuOverCommitDurationMinutes**、**HighUtilization**、または **MaxFreeMemoryForOverUtilized** に達した場合、ホストにアタッチされた追加の仮想マシンは起動しません。

Cluster_Maintenance スケジューリングポリシーは、メンテナンスタスク時にクラスター内のアクティビティーを制限します。**Cluster_Maintenance** ポリシーが設定されている場合、高可用性仮想マシンを

除き、新しい仮想マシンを起動できません。ホストの障害が発生した場合、高可用性仮想マシンが正しく再起動し、どの仮想マシンも移行できます。

1.3.1. スケジューリングポリシーの作成

新規のスケジューリングポリシーを作成して、仮想マシンを Red Hat Virtualization 環境の特定のクラスターに分散するロジックを制御できます。

スケジューリングポリシーの作成

1. **Administration → Configure** をクリックします。
2. **Scheduling Policies** タブをクリックします。
3. **New** をクリックします。
4. スケジュールポリシーの **Name** と **Description** を入力します。
5. フィルターモジュールを設定します。
    a. **Filter Modules** セクションで、**Disabled Filters** セクションから **Enabled Filters** セクションに、優先するフィルターモジュールをドラッグアンドドロップしてスケジューリングポリシーに適用します。
    b. また、特定のフィルターモジュールを **First** として設定して優先度を最も高くしたり、**Last** として設定して優先度を最も低くすることもできます。優先度を設定するには、フィルターモジュールを右クリックし、**Position** にカーソルを合わせ、**First** または **Last** を選択します。
6. 加重モジュールを設定します。
    a. **Weights Modules** セクションで、**Disabled Weights** セクションから **Enabled Weights** セクションに、優先する加重モジュールをドラッグアンドドロップしてスケジューリングポリシーに適用します。
    b. 有効な加重モジュールの左側にある **+** および **-** ボタンを使用して、これらのモジュールの重みを増減します。
7. ロードバランシングポリシーを指定します。
    a. **Load Balancer** セクションのドロップダウンメニューから、スケジューリングポリシーに適用する負荷分散ポリシーを選択します。
    b. **Properties** セクションのドロップダウンメニューから、スケジューリングポリシーに適用する負荷分散プロパティーを選択し、そのプロパティーの右側にある text フィールドを使用して値を指定します。
    c. **+** ボタンおよび **-** ボタンを使用して、プロパティーを追加または削除します。
8. **OK** をクリックします。

1.3.2. New Scheduling Policy および Edit Scheduling Policy ウィンドウの設定の説明

以下の表は、**New Scheduling Policy** および **Edit Scheduling Policy** ウィンドウで利用可能なオプションの詳細を示しています。

表1.12 New Scheduling Policy および Edit Scheduling Policy の設定

フィールド名	説明
Name	スケジューリングポリシーの名前。これは、Red Hat Virtualization Manager のスケジューリングポリシーを参照するために使用される名前です。
説明	スケジューリングポリシーの説明。このフィールドは推奨されますが、必須ではありません。
Filter Modules	クラスター内の仮想マシンが実行できるホストを制御するフィルターセット。フィルターを有効にすると、以下のように、フィルターで指定された条件を満たさないホストが除外されます。 ● **CpuPinning**: CPU ピニングの定義を満たさないホスト。 ● **Migration**: 同じホストへの移行を防ぎます。 ● **PinToHost**: 仮想マシンが固定されているホスト以外のホスト。 ● **CPU-Level**: 仮想マシンの CPU トポロジーを満たさないホスト。 ● **CPU**: 仮想マシンに割り当てられた数よりも少ない CPU を持つホスト。 ● **Memory**: 仮想マシンを実行するために十分なメモリーを持たないホスト。 ● **VmAffinityGroups**: アフィニティーグループのメンバーである仮想マシンに指定した条件を満たさないホスト。たとえば、アフィニティーグループの仮想マシンは、同じホストまたは別のホストで実行される必要があることなど。 ● **VmToHostsAffinityGroups**: アフィニティーグループのメンバーである仮想マシンに指定した条件を満たさないホストのグループ。たとえば、アフィニティーグループの仮想マシンは、グループ内のいずれかのホスト上で動作するか、グループから除外された別のホスト上で動作する必要があることなど。 ● **InClusterUpgrade**: 仮想マシンが現在実行しているホストよりも以前のオペレーティングシステムを実行しているホスト。 ● **hostdevice**: 仮想マシンに必要なホストデバイスに対応していないホスト。 ● **HA**: セルフホスト型エンジン環境内の Manager 用仮想マシンを強制し、正の高可用性スコアのあるホストでのみ実行するようにします。

フィールド名	説明
	• **Emulated-Machine**: 適切なエミュレートされたマシンをサポートしていないホスト。 • **Network**: 仮想マシンのネットワークインターフェイスコントローラーで必要なネットワークがインストールされていないホスト、またはクラスターのディスプレイネットワークがインストールされていないホスト。 • **HostedEnginesSpares**: 指定した数のセルフホストエンジンノードに Manager 仮想マシンの領域を確保します。 • **Label**: 必要なアフィニティーラベルを持たないホスト。 • **Compatibility-Version**: 正しい互換性バージョンがサポートされているホスト上でのみ仮想マシンを実行します。 • **CPUOverloaded**: CPU がオーバーロードされたホスト。
Weights Modules	仮想マシンを実行できるクラスター内のホストを決定する際に考慮される要因の相対優先度を制御する重みのセット。 • **InClusterUpgrade**: オペレーティングシステムのバージョンに応じてホストを重み付けします。この重みは、仮想マシンが現在実行されているホストと同じオペレーティングシステムを持つホストよりも、以前のオペレーティングシステムを持つホストにペナルティーを与えます。これにより、より新しいホストが常に優先されるようになります。 • **OptimalForHaReservation**: 高可用性スコアに従ってホストを重み付けします。 • **None**: 均等割り付けモジュールに基づいてホストの重み付けを行います。 • **OptimalForEvenGuestDistribution**: ホスト上で稼働している仮想マシンの数に応じて、ホストを重み付けします。 • **VmAffinityGroups**: 仮想マシンに定義されたアフィニティーグループに応じて、ホストを重み付けします。この加重モジュールは、あるアフィニティーグループの仮想マシンが、そのアフィニティーグループのパラメーターに応じて、同じホスト上で実行される可能性や、別々のホスト上で実行される可能性を決定します。 • **VmToHostsAffinityGroups**: 仮想マシンに定義されたアフィニティーグループに応じて、ホストを重み付けします。この重みモジュールは、アフィニティーグループの仮想マシンが、グループ内のホストの1つ、またはグループから除外された別のホスト上で実行される可能性を決定します。

フィールド名	説明
	● **OptimalForCPUPowerSaving**: CPU 使用率に従ってホストを重み付けし、CPU 使用率が高いホストを優先します。 ● **OptimalForEvenCpuDistribution :** CPU 使用率に従ってホストを重み付けし、CPU 使用率が低いホストを優先します。 ● **HA**: 高可用性スコアに応じてホストを重み付けします。 ● **PreferredHosts**: 仮想マシンのセットアップ時に優先的に使用するホストを指定します。 ● **OptimalForMemoryPowerSaving**: メモリーの使用量に応じてホストを重み付けし、利用可能なメモリーが少ないホストを優先します。 ● **OptimalForMemoryEvenDistribution :** メモリーの使用量に応じてホストを重み付けし、利用可能なメモリーが多いホストを優先します。
ロードバランサー	このドロップダウンメニューでは、適用する負荷分散モジュールを選択できます。負荷分散モジュールは、使用率が高いホストから、使用率が低いホストに仮想マシンを移行するために使用されるロジックを決定します。
プロパティー	このドロップダウンメニューでは、負荷分散モジュールのプロパティーを追加または削除でき、スケジューリングポリシーに負荷分散モジュールを選択している場合にのみ利用できます。デフォルトではプロパティーは定義されておらず、利用可能なプロパティーは、選択された負荷分散モジュールに固有です。+ および - ボタンを使用して、負荷分散モジュールにプロパティーを追加または削除します。

1.4. インスタンスタイプ

インスタンスタイプを使用して、仮想マシンのハードウェア設定を定義できます。仮想マシンの作成時または編集時にインスタンスタイプを選択すると、ハードウェア設定フィールドが自動的に入力されます。これにより、すべてのフィールドを手動で入力しなくても、同じハードウェア設定で複数の仮想マシンを作成できます。

以下の表で説明されているように、事前定義されたインスタンスタイプのセットはデフォルトで利用できます。

表1.13 事前定義されたインスタンスタイプ

Name	メモリー	vCPU
Tiny	512 MB	1
Small	2 GB	1
中	4 GB	2
Large	8 GB	2
XLarge	16 GB	4

管理者は **Configure** ウィンドウの **Instance Types** タブから、インスタンスタイプを作成、編集、および削除できます。

インスタンスタイプにバインドされる **New Virtual Machine** および **Edit Virtual Machine** ウィンドウのフィールドの横にチェーンリンクイメージ () があります。これらのフィールドの値の 1 つが変更されると、仮想マシンはインスタンスタイプから切り離され、**Custom** に変更され、チェーンが切れたよう見えます ()。しかし、値が元に戻されると、チェーンは再度リンクし、インスタンスタイプは選択されたものに戻ります。

1.4.1. インスタンスタイプの作成

管理者は、仮想マシンの作成時または編集時にユーザーが選択する新しいインスタンスタイプを作成できます。

インスタンスタイプの作成

1. **Administration → Configure** をクリックします。
2. **Instance Types** タブをクリックします。
3. **New** をクリックします。
4. インスタンスタイプの **Name** および **Description** を入力します。
5. **Show Advanced Options** をクリックし、必要に応じてインスタンスタイプを設定します。**New Instance Type** ウィンドウに表示される設定は、**New Virtual Machine** ウィンドウの設定と同じですが、関連するフィールドのみが表示されます。**Virtual Machine Management Guide**の Explanation of Settings in the New Virtual Machine and Edit Virtual Machine Windows を参照してください。
6. **OK** をクリックします。

新規インスタンスタイプは **Configure** ウィンドウの **Instance Types** タブに表示され、仮想マシンの作成時または編集時に **Instance Type** ドロップダウンリストから選択できます。

1.4.2. インスタンスタイプの編集

管理者は、**Congifure** ウインドウから既存のインスタンスタイプを編集できます。

インスタンスタイププロパティーの編集

1. **Administration** → **Configure** をクリックします。
2. **Instance Types** タブをクリックします。
3. 編集するインスタンスタイプを選択します。
4. **Edit** をクリックします。
5. 必要に応じて設定を変更します。
6. **OK** をクリックします。

インスタンスタイプの設定が更新されます。このインスタンスタイプに基づく新しい仮想マシンが作成されるか、このインスタンスタイプに基づく既存の仮想マシンが更新されると、新しい設定が適用されます。

このインスタンスタイプに基づく既存の仮想マシンには、更新されるチェーンアイコンが付いたフィールドが表示されます。インスタンスタイプの変更時に既存の仮想マシンが稼働していた場合は、その横にオレンジ色の Pending Changes アイコンが表示され、次回の再起動時にチェーンのアイコンが付いたフィールドが更新されます。

1.4.3. インスタンスタイプの削除

インスタンスタイプの削除

1. **Administration** → **Configure** をクリックします。
2. **Instance Types** タブをクリックします。
3. 削除するインスタンスタイプを選択します。
4. **Remove** をクリックします。
5. 削除するインスタンスタイプに基づいた仮想マシンがある場合は、アタッチされた仮想マシンをリストする警告ウィンドウが表示されます。インスタンスタイプの削除を続行するには、**Approve Operation** チェックボックスを選択します。それ以外の場合は、**Cancel** をクリックします。
6. **OK** をクリックします。

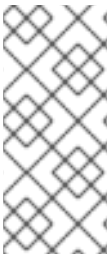
インスタンスタイプが **Instance Types** リストから削除され、新規仮想マシンの作成時に使用できなくなります。削除されたインスタンスタイプにアタッチされた仮想マシンは **Custom** (インスタンスタイプなし) にアタッチされるようになります。

1.5. MAC アドレスプール

MAC アドレスプールは、各クラスターに割り当てられる MAC アドレスの範囲を定義します。各クラスターに MAC アドレスプールが指定されます。MAC アドレスプールを使用すると、Red Hat Virtualization は MAC アドレスを自動的に新しい仮想ネットワークデバイスに生成して割り当てることができます。これは、MAC アドレスの重複を防ぐのに役立ちます。MAC アドレスプールは、クラスターに関連するすべての MAC アドレスが、割り当てられた MAC アドレスプールの範囲内にあると、メモリー効率が高くなります。

同じ MAC アドレスプールを複数のクラスターで共有できますが、各クラスターには MAC アドレスプールが 1 つ割り当てられます。デフォルトの MAC アドレスプールは Red Hat Virtualization によって作成され、別の MAC アドレスプールが割り当てられない場合に使用されます。MAC アドレスプールを

クラスターに割り当てる方法は、「新規クラスターの作成」 を参照してください。



注記

複数の Red Hat Virtualization クラスターがネットワークを共有する場合は、デフォルトの MAC アドレスプールのみに依存しないでください。これは、各クラスターの仮想マシンが同じ範囲の MAC アドレスを使用しようとするため、競合が発生するためです。MAC アドレスの競合を回避するには、MAC アドレスプールの範囲をチェックして、各クラスターに一意の MAC アドレス範囲が割り当てられていることを確認します。

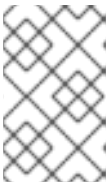
MAC アドレスプールでは、最後にプールに戻されたアドレスの次に利用可能な MAC アドレスが割り当てられます。範囲内に残されたアドレスがない場合には、範囲の先頭から検索を再開します。1 つの MAC アドレスプールに、利用可能な MAC アドレスがある複数の MAC アドレスの範囲が定義されている場合、利用可能な MAC アドレスが選択されるのと同じように、範囲が順番に受信リクエストに対応します。

1.5.1. MAC アドレスプールの作成

新しい MAC アドレスプールを作成できます。

MAC アドレスプールの作成

1. **Administration → Configure** をクリックします。
2. **MAC Address Pool**タブをクリックします。
3. **Add** をクリックします。
4. 新しい MAC アドレスプールの **Name** および **Description** を入力します。
5. **Allow Duplicates** チェックボックスを選択し、MAC アドレスをプールで複数回使用できるようにします。MAC アドレスプールでは、重複した MAC アドレスを自動的に使用することはありませんが、duplicates オプションを有効にすると、ユーザーが重複した MAC アドレスを手動で使用できます。



注記

ある MAC アドレスプールで重複を無効にし、別の MAC アドレスプールで重複を有効にした場合、重複を無効にしたプールでは各 MAC アドレスは 1 回しか使用できませんが、重複を有効にしたプールでは複数回使用できます。

6. 必要な **MAC Address Ranges**を入力します。複数の範囲を入力するには、**From** フィールドおよび **To** フィールドの横にあるプラスボタンをクリックします。
7. **OK** をクリックします。

1.5.2. MAC アドレスプールの編集

MAC アドレスプールを編集して、プールで利用可能な MAC アドレスの範囲や重複が許可されるかどうかなどの詳細を変更できます。

MAC アドレスプールプロパティーの編集

1. **Administration → Configure** をクリックします。

2. **MAC Address Pool**タブをクリックします。
3. 編集する MAC アドレスプールを選択します。
4. **Edit** をクリックします。
5. 必要に応じて **Name**、**Description**、**Allow Duplicates**、および **MAC Address Ranges** フィールドを変更します。

注記

MAC アドレス範囲を更新すると、既存の NIC の MAC アドレスは再割り当てされません。すでに割り当てられている MAC アドレスで、新しい MAC アドレスの範囲外のものは、ユーザー指定の MAC アドレスとして追加され、その MAC アドレスプールで追跡されます。

6. **OK** をクリックします。

1.5.3. MAC アドレスプールのパーミッションの編集

MAC アドレスプールの作成後に、そのユーザー権限を編集できます。ユーザーパーミッションにより、どのデータセンターが MAC アドレスプールを使用できるかが制御されます。新規ユーザーパーミッションの追加に関する詳細は、「ロール」 を参照してください。

MAC アドレスプールのパーミッションの編集

1. **Administration → Configure** をクリックします。
2. **MAC Address Pool**タブをクリックします。
3. 必要な MAC アドレスプールを選択します。
4. MAC アドレスプールのユーザーパーミッションを編集します。
   - ユーザーパーミッションを MAC アドレスプールに追加するには、以下を実行します。
     a. **Configure** ウィンドウの下にあるユーザー権限ペインで **Add** をクリックします。
     b. 必要なユーザーを検索して選択します。
     c. **Role to Assign** ドロップダウンリストから必要なロールを選択します。
     d. **OK** をクリックしてユーザーパーミッションを追加します。
   - ユーザーパーミッションを MAC アドレスプールから削除するには、以下を実行します。
     a. **Configure** ウィンドウの下にあるユーザー権限ペインで、削除するユーザー権限を選択します。
     b. ユーザーの権限を削除するには、**Remove** をクリックします。

1.5.4. MAC アドレスプールの削除

作成した MAC アドレスプールがクラスターに関連付けられていない場合は削除できますが、デフォルトの MAC アドレスプールは削除できません。

MAC アドレスプールの削除

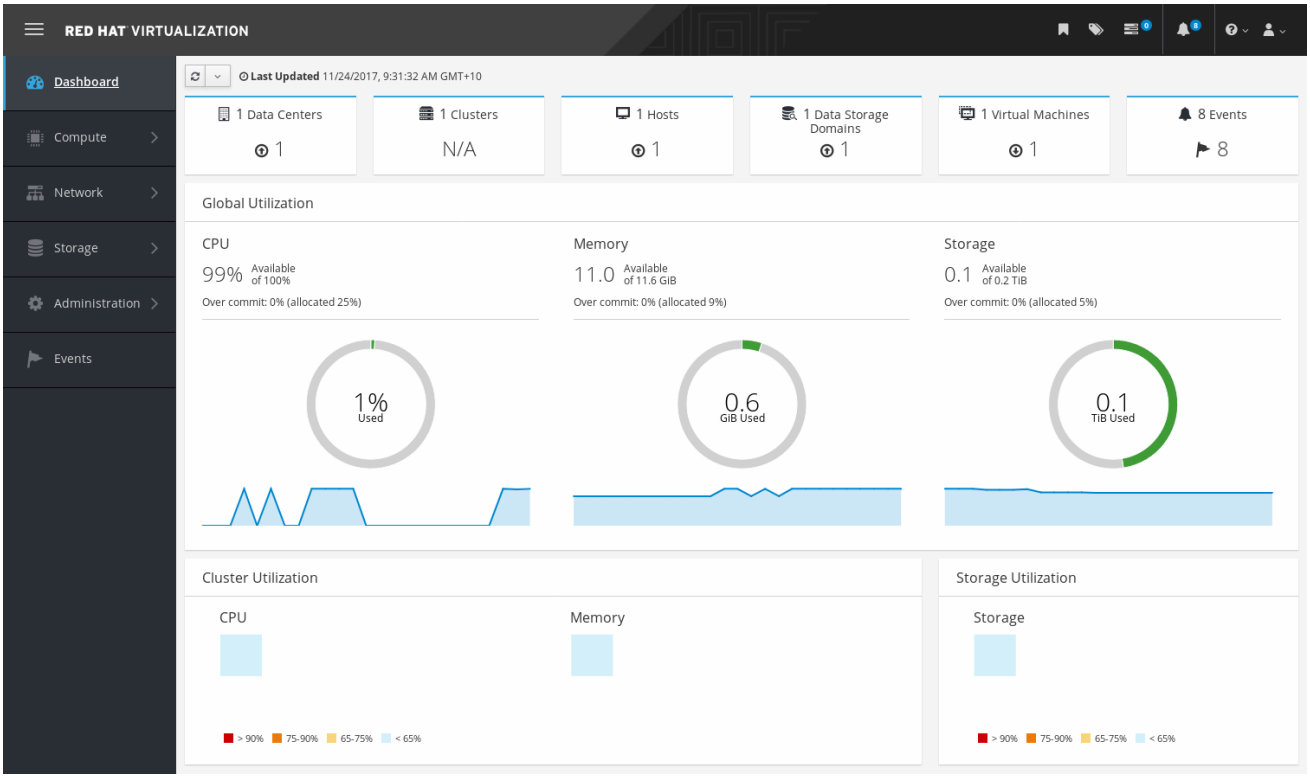
1. **Administration → Configure** をクリックします。
2. **MAC Address Pool**タブをクリックします。
3. 削除する MAC アドレスプールを選択します。
4. **Remove** をクリックします。
5. **OK** をクリックします。

第2章 ダッシュボード

ダッシュボードは、Red Hat Virtualization のリソースと使用率の概要を表示することで、Red Hat Virtualization のシステム状態の概要を提供します。この概要により、問題を警告することができ、問題領域を分析できます。

ダッシュボードの情報は、Data Warehouse からはデフォルトで 15 分ごと、Manager API からはデフォルトで 15 秒 ごと、またはダッシュボードが更新されるたびに更新されます。ダッシュボードは、ユーザーが他のページから戻ったときや、手動でリフレッシュしたときに更新されます。ダッシュボードは自動的に更新されません。インベントリーカードの情報は Manager API から提供され、利用状況の情報は Data Warehouse から提供されます。ダッシュボードは、UI プラグインコンポーネントとして実装されており、Manager と一緒に自動的にインストールおよびアップグレードされます。

図2.1 ダッシュボード



2.1. 前提条件

ダッシュボードを使用するには、Data Warehouse がインストールされ、設定されている必要があります。**Data Warehouse Guide**の Installing and Configuring Data Warehouse を参照してください。

2.2. グローバルインベントリー

ダッシュボードの上部には、Red Hat Virtualization リソースのグローバルインベントリーが表示され、データセンター、クラスター、ホスト、ストレージドメイン、仮想マシン、イベントなどの項目が含まれます。アイコンは各リソースの状態、数字はその状態にある各リソースの数量を表しています。

図2.2 グローバルインベントリー

タイトルにはリソースの種類別の数が表示され、その下にはステータスが表示されます。リソースのタイトルをクリックすると、Red Hat Virtualization Manager の関連ページに移動します。**Clusters** のステータスは常に N/A と表示されます。

表2.1 リソースの状況

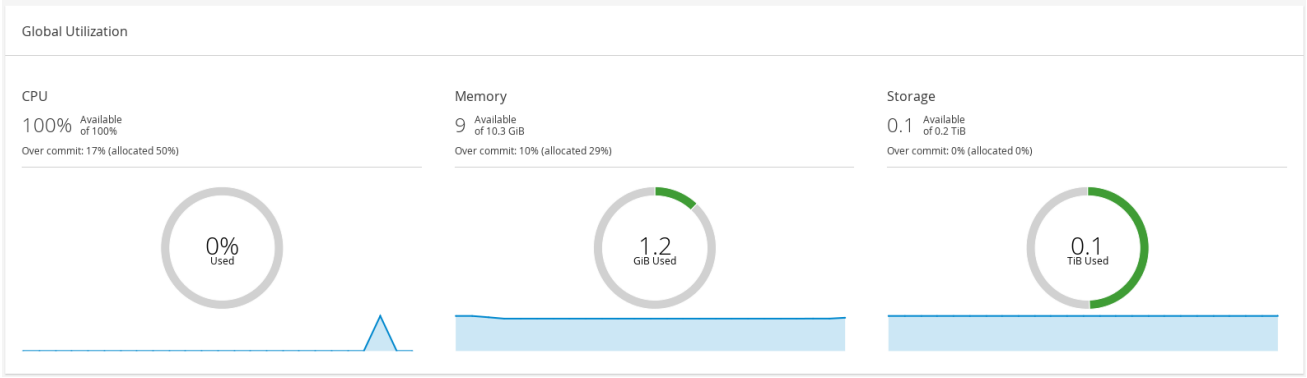
アイコン	状態
	Red Hat Virtualization にはそのようなリソースは一切追加されていません。
	警告ステータスを持つリソースの番号を表示します。アイコンをクリックすると、該当ページに移動し、検索対象は警告状態のリソースに限定されます。検索の制限は、リソースごとに異なります。 ● **Data Centers**: 検索対象は、稼働していないデータセンターまたは応答していないデータセンターに限られます。 ● **Gluster Volumes**: 検索対象は、開始中、一時停止中、移行中、待機中、中断中、または停止中の Gluster ボリュームに限定されます。 ● **Hosts**: 検索対象は、未割り当て、メンテナンスモード、インストール中、再起動中、メンテナンスの準備中、承認待ち、または接続中のホストに限定されます。 ● **Storage Domains**: 検索対象となるのは、初期化されていない、アタッチされていない、非クティブでない、メンテナンスモード、メンテナンスの準備中、デタッチ中、またはアクティベート中のストレージドメインに限られます。 ● **Virtual Machines**:: 検索対象は、開始中、一時停止中、移行中、待機中、中断中、または停止中の仮想マシンに限定されます。 ● **Events**: 重大度が警告のイベントに限定して検索を行います。
	up ステータスを持つリソースの番号を表示します。アイコンをクリックすると、該当ページに移動し、検索対象は up 状態のリソースに限定されます。

アイコン	状態
	down 状態のリソースの番号を表示します。アイコンをクリックすると、該当ページに移動し、検索対象は down 状態のリソースに限定されます。検索の制限は、リソースごとに異なります。 ● **Data Centers**: 検索対象は、初期化されていない、メンテナンスモード、または down ステータスのデータセンターに限られます。 ● **Gluster Volumes**: 検索対象は、デタッチされているまたは非アクティブな Gluster ボリュームに限られます。 ● **Hosts**: 検索対象は、反応しない、エラーが発生している、インストールエラーが発生している、動作していない、初期化中、または down 状態のホストに限られます。 ● **Storage Domains**: 検索対象は、デタッチされているまたは非アクティブなストレージドメインに限られます。 ● **Virtual Machines**: 検索対象は、Down 状態、応答していない、または再起動中の仮想マシンに限られます。
	アラートステータスを持つイベントの数を表示します。アイコンをクリックすると **Events** に移動しますが、検索対象は深刻度が警告のイベントに限定されます。
	エラーステータスを持つイベントの数を表示します。アイコンをクリックすると **Events** に移動しますが、検索対象は深刻度がエラーのイベントに限定されます。

2.3. グローバルでの活用

Global Utilization セクションでは、CPU、Memory、Storage のシステム使用状態が表示されます。

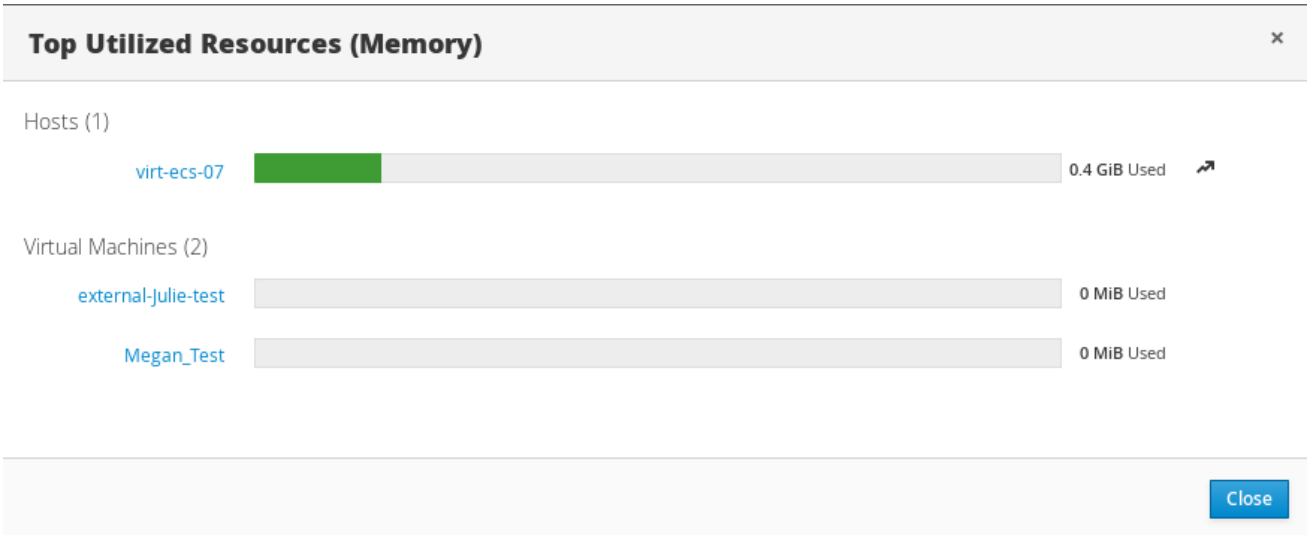
図2.3 グローバルでの活用



- 上段には、利用可能な CPU、メモリー、ストレージ、およびオーバーコミット率の割合が表示されます。たとえば、CPU のオーバーコミット率は、Data Warehouse の最新データに基づいて、仮想コアの数を実行中の仮想マシンで利用可能な物理コアの数で割って算出します。
- ドーナツは、CPU、メモリー、またはストレージの使用率をパーセンテージで表示し、過去 5 分間の平均使用率に基づいて、すべてのホストの平均使用率を表示します。ドーナツの断面にカーソルを合わせると、選択したセクションの値が表示されます。
- 下部の折れ線グラフは、過去 24 時間の傾向を表示しています。各データポイントは、特定の時間の平均使用量を示しています。グラフ上のポイントにカーソルを合わせると、CPU のグラフでは時間と使用率が、メモリーとストレージのグラフでは使用量が表示されます。

2.3.1. 最も使用されているリソース

図2.4 最も使用されているリソース (メモリー)

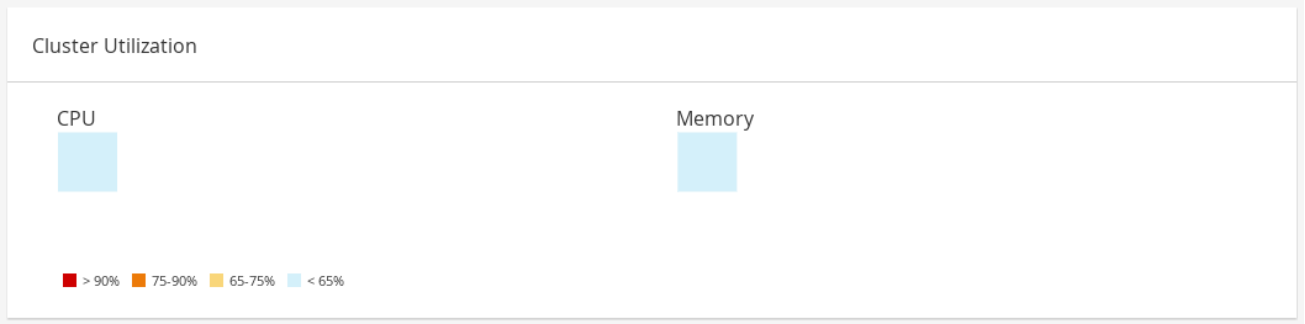


ダッシュボードのグローバル使用率の項目にあるドーナツをクリックすると、CPU、メモリー、ストレージのうち、使用率の高いリソースのリストが表示されます。CPU とメモリーについては、最も使用率の高い 10 台のホストと仮想マシンのリストがポップアップで表示されます。ストレージについては、利用されているストレージドメインと仮想マシンのトップ 10 のリストがポップアップで表示されます。使用量バーの右にある矢印は、そのリソースの直近 1 分間の使用量の傾向を示しています。

2.4. クラスターの活用

Cluster Utilization セクションは、CPU とメモリーのクラスター使用率をヒートマップで表示します。

図2.5 クラスターの活用



2.4.1. CPU

過去 24 時間の CPU の平均使用率を示す特定クラスターの CPU 使用率のヒートマップ。ヒートマップにカーソルを合わせると、クラスター名が表示されます。ヒートマップをクリックすると、**Compute → Hosts** に移動し、特定のクラスターの検索を CPU 使用率でソートした結果が表示されます。クラスターによる CPU の使用率を計算するために使用される式は、クラスターのホスト CPU 使用率の平均です。これは、クラスターによる CPU の合計平均使用率を出すために、過去 24 時間の各ホストの CPU 使用率の平均値を用いて算出されます。

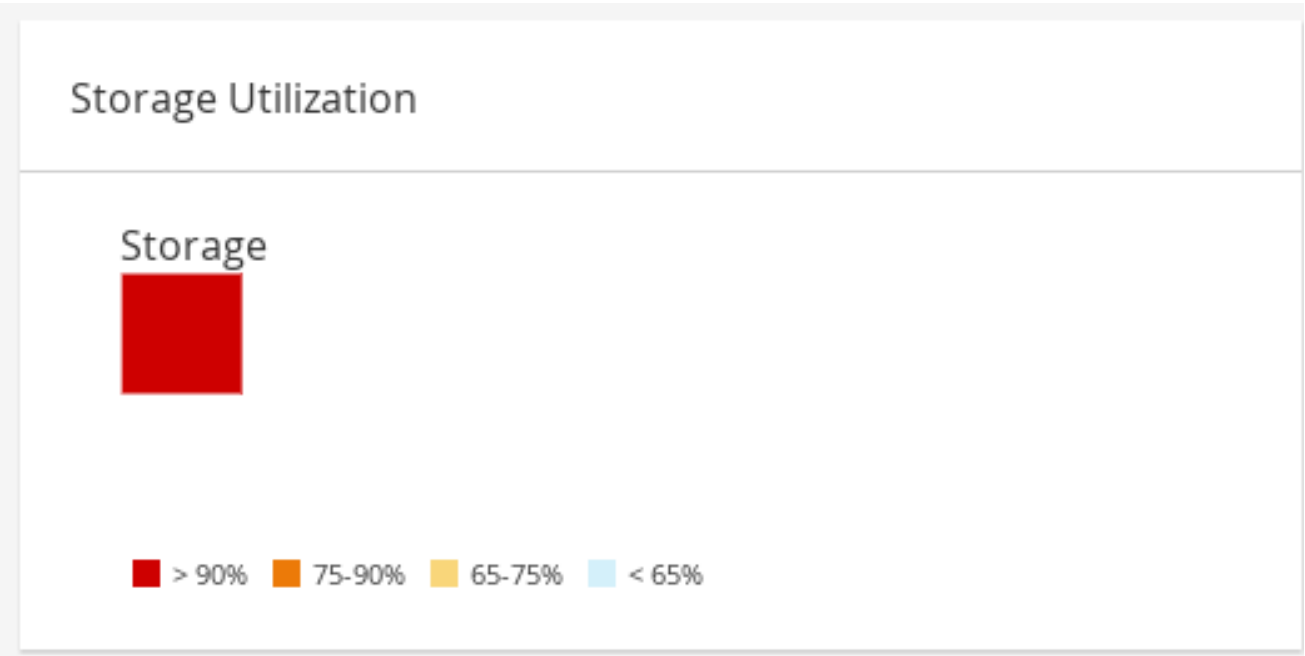
2.4.2. メモリー

過去 24 時間のメモリーの平均使用率を示す特定クラスターのメモリー使用率のヒートマップ。ヒートマップにカーソルを合わせると、クラスター名が表示されます。ヒートマップをクリックすると、**Compute → Hosts** に移動し、特定のクラスターの検索をメモリー使用率でソートした結果が表示されます。クラスターによるメモリー使用率を計算するために使用される式は、クラスターのメモリー使用率の合計 (GB 単位) です。これは、クラスターによるメモリー合計平均使用率を出すために、過去 24 時間の各ホストの平均メモリー使用率を用いて算出されます。

2.5. ストレージの活用

Storage Utilization セクションは、ヒートマップでストレージ使用率を表示します。

図2.6 ストレージの活用



ヒートマップは、過去 24 時間のストレージの平均利用率を表します。クラスターによるストレージ使用率を計算するために使用される式は、クラスターのストレージ使用率の合計です。これは、クラスターによる ストレージ の合計平均使用率を出すために、過去 24 時間の各ホストの平均ストレージ使用率を用いて算出されます。ヒートマップにカーソルを合わせると、ストレージドメイン名が表示されます。ヒートマップをクリックすると **Storage → Domains** に移動し、ストレージドメインが使用率でソートされます。

第3章 検索

3.1. RED HAT VIRTUALIZATION での検索

管理ポータルでは、仮想マシン、ホスト、ユーザーなど、何千ものリソースを管理することができます。検索を行うには、各リソースのメインページにある検索バーに、検索クエリー (フリーテキストまたは構文ベース) を入力します。検索条件をブックマークとして保存しておけば、検索結果を必要とするたびに検索条件を再入力する必要はありません。検索では大文字小文字の区別はありません。

3.2. 検索構文と例

Red Hat Virtualization リソースの検索クエリーの構文は以下のとおりです。

result type: {criteria} [sortby sort_spec]

構文の例

以下の例は、検索クエリーの使用方法と、Red Hat Virtualization が検索クエリーの構築を支援する方法を理解するのに役立ちます。

表3.1 検索クエリーの例

例	結果
Hosts: Vms.status = up page 2	稼働中の仮想マシンを実行しているすべてのホストのリストの 2 ページ目を表示します。
Vms: domain = qa.company.com	指定されたドメインで稼働しているすべての仮想マシンの一覧を表示します。
Vms: users.name = Mary	ユーザー名が Mary のユーザーに属する全仮想マシンの一覧を表示します。
Events: severity > normal sortby time	重大度が Normal よりも高いすべての Events の一覧を表示します。

3.3. 自動完了の検索

管理ポータルは、有効で強力な検索クエリーの作成に役立つ自動補完を提供します。検索クエリーの各部分を入力すると、検索の次の部分を選択するドロップダウンリストが、Search Bar の下に開きます。一覧から選択して、検索の次の部分の入力/選択を続けたり、オプションを無視したりして、手動でクエリーを入力を続けたりできます。

以下の表は、管理ポータルの自動補完がクエリーの構築を助けるする方法の例を示しています。

Hosts: Vms.status = down

表3.2 自動補完を使用した検索クエリーの例

入力	表示されているリスト項目	アクション
h	**Hosts** (1 つのオプションのみ)	**Hosts** を選択または Hosts を入力
Hosts:	すべてのホストプロパティー	v を入力
Hosts: v	**v** で始まるホストプロパティー	**Vms** を選択または Vms を入力
Hosts: Vms	すべての仮想マシンプロパティー	s を入力
Hosts: Vms.s	**s** で始まるすべての仮想マシンプロパティー	**status** を選択または status を入力
Hosts: Vms.status	**=** **!=**	= を選択または入力
Hosts: Vms.status =	すべてのステータス値	down を選択または入力

3.4. 検索結果タイプオプション

結果のタイプを使用すると、以下のタイプのリソースを検索できます。

- Vms、仮想マシンのリスト。
- Host、ホストのリスト。
- Pools、プールのリスト。
- Template、テンプレートのリスト。
- Events、イベントのリスト。
- Users、ユーザーのリスト。
- Cluster、クラスターのリスト。
- DataCenter、データセンターのリスト。
- Storage、ストレージドメインのリスト。

各タイプのリソースには、一意のプロパティーセットと、関連付けられたその他のリソースタイプのセットがあるため、各検索タイプには、有効な構文の組み合わせがあります。自動補完機能を使用して、有効なクエリーも簡単に作成できます。

3.5. 検索基準

クエリーのコロンの後に検索条件を指定できます。**{criteria}** の構文は以下のようになります。

<prop><operator><value>

または

<obj-type><prop><operator><value>

例

以下の表は、構文の部分を示しています。

表3.3 検索基準の例

部分	説明	値	例	注記
prop	検索対象リソースのプロパティー。リソースタイプのプロパティー (**obj-type** を参照) または **tag** (カスタムタグ) にすることもできます。	検索対象を特定のプロパティーを持つオブジェクトに制限します。たとえば、**status** プロパティーでオブジェクトを検索します。	状態	該当なし
obj-type	検索対象のリソースに関連付けることができるリソースタイプ。	これは、データセンターや仮想マシンなどのシステムオブジェクトです。	Users	該当なし
operator	比較演算子。	= != (等しくない) > < >= <=	該当なし	値オプションはプロパティーによって異なります。

部分	説明	値	例	注記
値	その式が何と比較されるか。	String Integer ランキング 日付 (Regional Settings に応じた書式設定)	Jones 256 normal	• ワイルドカードは文字列内で使用できます。 • "" (間にスペースが入っていない 2 つの引用符のセット) は、初期化されていない (空の) 文字列を表すために使用できます。 • スペースが含まれる文字列または日付を二重引用符で囲む必要があります

3.6. 検索: 複数の基準およびワイルドカード

ワイルドカードは文字列の構文の **<value>** 部分で使用できます。たとえば、**m** で始まる全ユーザーを検索するには、**m*** を入力します。

ブール演算子の **AND** および **OR** を使用して、2 つの基準を持つ検索を実行できます。以下に例を示します。

Vms: users.name = m* AND status = Up

このクエリーは、名前が m で始まるユーザーに対して実行中の仮想マシンをすべて返します。

Vms: users.name = m* AND tag = "paris-loc"

このクエリーは、名前が m で始まるユーザーに対して paris-loc でタグ付けされたすべての仮想マシンを返します。

AND または **OR** を使用せずに 2 つの基準を指定した場合、 **AND** が暗黙的に指定されます。**AND** は **OR** よりも優先され、**OR** は暗黙の **AND** よりも優先されます。

3.7. 検索: 検索順序の決定

返される情報の並び替え順序は、**sortby** を使用して決定できます。並べ替え方向 (昇順は **asc**、降順は **desc**) を含めることができます。

以下に例を示します。

events: severity > normal sortby time desc

このクエリーは、重大度が Normal よりも大きいすべての Events を時刻でソートして返します (降順)。

3.8. データセンターの検索

以下の表は、データセンターのすべての検索オプションを示しています。

表3.4 データセンターの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Clusters.clusters-prop	プロパティータイプによります。	データセンターに関連付けられたクラスターのプロパティー。
name	String	データセンターの名前。
description	String	データセンターの説明
type	String	データセンターのタイプ。
status	リスト	データセンターの可用性
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

datacenter: type = nfs and status != up

この例では、ストレージタイプが NFS で up 以外状態のデータセンターの一覧を返します。

3.9. クラスターの検索

以下の表は、クラスターのすべての検索オプションについて説明しています。

表3.5 クラスターの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Datacenter.datacenter-prop	プロパティータイプによります。	クラスターに関連付けられたデータセンターのプロパティー。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Datacenter	String	クラスターが属するデータセンター。
name	String	ネットワーク上のクラスターを識別する一意の名前。
description	String	クラスターの説明。
initialized	String	クラスターのステータスを示す true または False。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

Clusters: initialized = true or name = Default

この例では、初期化されたクラスターまたは Default という名前のクラスターの一覧を返します。

3.10. ホストの検索

以下の表は、ホストの全検索オプションを示しています。

表3.6 ホストの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Vms.Vms-prop	プロパティータイプによります。	ホストに関連付けられた仮想マシンのプロパティー。
Templates.templates-prop	プロパティータイプによります。	ホストに関連付けられたテンプレートのプロパティー。
Events.events-prop	プロパティータイプによります。	ホストに関連付けられたイベントのプロパティー。
Users.users-prop	プロパティータイプによります。	ホストに関連付けられたユーザーのプロパティー。
name	String	ホストの名前。
status	リスト	ホストの可用性。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
external_status	String	外部システムおよびプラグインによって報告されるホストのヘルスステータス。
cluster	String	ホストが属するクラスター。
address	String	ネットワーク上のホストを識別する一意の名前。
cpu_usage	Integer	使用される処理能力の割合。
mem_usage	Integer	使用されるメモリーの割合。
network_usage	Integer	ネットワーク使用率の割合。
load	Integer	特定のタイムスライスで、プロセッサーごとに **run-queue** で実行されるのを待っているジョブ。
version	Integer	オペレーティングシステムのバージョン番号。
cpus	Integer	ホスト上の CPU 数。
memory	Integer	使用可能なメモリーの量。
cpu_speed	Integer	CPU の処理速度。
cpu_model	String	CPU のタイプ。
active_vms	Integer	現在実行中の仮想マシンの数。
migrating_vms	Integer	現在移行中の仮想マシンの数。
committed_mem	Integer	コミットされたメモリーの割合
tag	String	ホストに割り当てられたタグ。
type	String	ホストのタイプ。
datacenter	String	ホストが属するデータセンター。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
page	Integer	表示する結果のページ番号。

例

Hosts: cluster = Default and Vms.os = rhel6

この例では、Default クラスターの一部であるホストの一覧と、Red Hat Enterprise Linux 6 オペレーティングシステムを実行するホスト仮想マシンを返します。

3.11. ネットワークの検索

以下の表は、ネットワークの全検索オプションを説明しています。

表3.7 ネットワークの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Cluster_network.clusternetwork-prop	プロパティータイプによります。	ネットワークに関連付けられたクラスターのプロパティー。
Host_Network.hostnetwork-prop	プロパティータイプによります。	ネットワークに関連付けられたホストのプロパティー。
name	String	ネットワークを識別するための人が判読可能な名前。
description	String	ネットワークを記述するキーワードまたはテキスト。オプションでネットワークの作成時に使用されます。
vlanid	Integer	ネットワークの VLAN ID。
stp	String	Spanning Tree Protocol (STP) がネットワークで有効または無効になっているかどうか。
mtu	Integer	論理ネットワークの最大伝送単位。
vmnetwork	String	ネットワークが仮想マシントラフィックのみに使用されているかどうか。
datacenter	String	ネットワークが接続されているデータセンター。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

Network: mtu > 1500 and vmnetwork = true

この例では、最大転送単位が 1500 バイトを超え、仮想マシンのみが使用するように設定されているネットワークの一覧を返します。

3.12. ストレージの検索

以下の表は、ストレージのすべての検索オプションについて説明しています。

表3.8 ストレージの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Hosts.hosts-prop	プロパティータイプによります。	ストレージに関連付けられたホストのプロパティー。
Clusters.clusters-prop	プロパティータイプによります。	ストレージに関連付けられたクラスターのプロパティー。
name	String	ネットワーク上のストレージを識別する一意の名前。
status	String	ストレージドメインのステータス。
external_status	String	外部システムおよびプラグインによって報告されるストレージドメインのヘルスステータス。
datacenter	String	ストレージが属するデータセンター。
type	String	ストレージのタイプ。
size	Integer	ストレージのサイズ。
used	Integer	使用中のストレージの量。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
committed	Integer	コミットされるストレージの量。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

ストレージサイズ > 200 または 使用済み < 50

以下の例では、合計ストレージ容量が 200 GB 以上のストレージ、または使用済みストレージ容量が 50 GB 未満のストレージ一覧を返します。

3.13. ディスクの検索

以下の表は、ディスクの全検索オプションを示しています。

注記

Disk Type および **Content Type** フィルターオプションを使用して、表示される仮想ディスクの数を減らすことができます。

表3.9 ディスクの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Datacenters.datacenters-prop	プロパティータイプによります。	ディスクに関連付けられたデータセンターのプロパティー。
Storages.storages-prop	プロパティータイプによります。	ディスクに関連付けられたストレージのプロパティー。
alias	String	ネットワーク上のストレージを識別する人が判読可能な名前。
description	String	ディスクを記述するキーワードまたはテキスト。オプションでディスクの作成時に使用されます。
provisioned_size	Integer	ディスクの仮想サイズ
size	Integer	ディスクのサイズ。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
actual_size	Integer	ディスクに割り当てられる実際のサイズ。
creation_date	Integer	ディスクが作成された日付。
bootable	String	ディスクを起動できるかどうか。有効な値は **0**、**1**、**yes**、**no** のいずれかです。
shareable	String	ディスクを一度に複数の仮想マシンにアタッチできるかどうか。有効な値は **0**、**1**、**yes**、**no** のいずれかです。
format	String	ディスクの形式。**unused**、**unassigned**、**cow**、**raw** のいずれかです。
status	String	ディスクのステータス**unassigned**、**ok**、**locked**、**invalid**、**illegal** のいずれかです。
disk_type	String	ディスクのタイプ。**image** または **lun** のいずれか。
number_of_vms	Integer	ディスクがアタッチされている仮想マシンの数。
vm_names	String	ディスクがアタッチされている仮想マシンの名前。
quota	String	仮想ディスクで強制されるクォータの名前。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

Disks: format = cow and provisioned_size > 8

以下の例では、QCOW 形式の仮想ディスクの一覧と、8 GB を超える割り当て済みのディスクサイズを返します。

3.14. ボリュームの検索

以下の表は、ボリュームのすべての検索オプションについて説明しています。

表3.10 ボリュームの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Cluster	String	ボリュームに関連付けられたクラスターの名前。
Cluster.cluster-prop	プロパティータイプ (例: name、description、comment、architecture) による	ボリュームに関連付けられたクラスターのプロパティー。
name	String	ボリュームを識別する、人が判読可能な名前。
type	String	distribute、replicate、distributed_replicate、stripe、または distributed_stripe のいずれか。
transport_type	Integer	TCP または RDMA のいずれか。
replica_count	Integer	レプリカの数。
stripe_count	Integer	ストライプの数。
status	String	ボリュームのステータスUp または Down のいずれかです。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

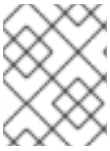
例

Volume: transport_type = rdma and stripe_count >= 2

この例では、トランスポートタイプが RDMA に設定され、ストライプが 2 つ以上あるボリュームのリストを返します。

3.15. 仮想マシンの検索

以下の表は、仮想マシンのすべての検索オプションについて説明しています。



注記

現時点で、Network Label、Custom Emulated Machine、および Custom CPU Type プロパティーはサポートされていない検索プロパティーです。

表3.11 仮想マシンの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Hosts.hosts-prop	プロパティータイプによります。	仮想マシンに関連付けられたホストのプロパティー。
Templates.templates-prop	プロパティータイプによります。	仮想マシンに関連付けられたテンプレートのプロパティー。
Events.events-prop	プロパティータイプによります。	仮想マシンに関連付けられたイベントのプロパティー。
Users.users-prop	プロパティータイプによります。	仮想マシンに関連付けられたユーザーのプロパティー。
Storage.storage-prop	プロパティータイプによります。	仮想マシンに関連付けられたストレージデバイスのプロパティー。
Vnic.vnic-prop	プロパティータイプによります。	仮想マシンに関連付けられた VNIC のプロパティー。
name	String	仮想マシンの名前。
status	リスト	仮想マシンの可用性
ip	Integer	仮想マシンの IP アドレス。
uptime	Integer	仮想マシンが実行されている期間 (分単位)。
domain	String	これらのマシンをグループ化するドメイン (通常は Active Directory ドメイン)。
os	String	仮想マシンの作成時に選択されたオペレーティングシステム。
creationdate	日付	仮想マシンが作成された日付。
address	String	ネットワーク上の仮想マシンを識別する一意の名前。
cpu_usage	Integer	使用される処理能力の割合。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
mem_usage	Integer	使用されるメモリーの割合。
network_usage	Integer	使用されるネットワークの割合。
memory	Integer	定義された最大メモリー。
apps	String	仮想マシンに現在インストールされているアプリケーション。
cluster	リスト	仮想マシンが属するクラスター。
pool	リスト	仮想マシンが属する仮想マシンプール。
loggedinuser	String	仮想マシンに現在ログインしているユーザーの名前。
tag	リスト	仮想マシンが属するタグ。
datacenter	String	仮想マシンが属するデータセンター。
type	リスト	仮想マシンタイプ (サーバーまたはデスクトップ)。
quota	String	仮想マシンに関連付けられたクォータの名前。
description	String	仮想マシンを記述するキーワードまたはテキスト。オプションとして、仮想マシンの作成時に使用されます。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。
next_run_configuration_exists	ブール値	仮想マシンに保留中の設定変更があります。

例

Vms: template.name = Win* and user.name = ""

この例では、ベーステンプレート名が **Win** で始まり、任意のユーザーに割り当てられている仮想マシンの一覧を返します。

例

Vms: cluster = Default and os = windows7

この例では、**Default** クラスターに属し、Windows 7 を実行している仮想マシンの一覧を返します。

3.16. プールの検索

以下の表は、プールの全検索オプションを示しています。

表3.12 プールの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
name	String	プールの名前。
description	String	プールの説明。
type	リスト	プールのタイプ。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

Pools: type = automatic

この例では、タイプが **automatic** のプールの一覧を返します。

3.17. テンプレートの検索

以下の表は、テンプレートの全検索オプションを示しています。

表3.13 テンプレートの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Vms.Vms-prop	String	テンプレートに関連付けられた仮想マシンのプロパティー。
Hosts.hosts-prop	String	テンプレートに関連付けられたホストのプロパティー。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Events.events-prop	String	テンプレートに関連付けられたイベントのプロパティー。
Users.users-prop	String	テンプレートに関連付けられたユーザーのプロパティー。
name	String	テンプレートの名前。
domain	String	テンプレートのドメイン。
os	String	オペレーティングシステムのタイプ。
creationdate	Integer	テンプレートが作成された日付。 日付の形式は **mm/dd/yy** です。
childcount	Integer	テンプレートから作成された仮想マシンの数。
mem	Integer	定義されたメモリー。
description	String	テンプレートの説明。
status	String	テンプレートのステータス
cluster	String	テンプレートに関連付けられたクラスター。
datacenter	String	テンプレートに関連付けられたデータセンター。
quota	String	テンプレートに関連付けられたクォータ。
sortby	リスト	返された結果をリソースプロパティーの 1 つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

Template: Events.severity >= normal and Vms.uptime > 0

この例では、テンプレートから派生した仮想マシンで重大度が Normal 以上のイベントが発生し、仮想マシンがまだ実行されているテンプレートの一覧が返されます。

3.18. ユーザーの検索

以下の表は、ユーザーの全検索オプションについて説明しています。

表3.14 ユーザーの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Vms.Vms-prop	プロパティータイプによります。	ユーザーに関連付けられた仮想マシンのプロパティー。
Hosts.hosts-prop	プロパティータイプによります。	ユーザーに関連付けられたホストのプロパティー。
Templates.templates-prop	プロパティータイプによります。	ユーザーに関連付けられたテンプレートのプロパティー。
Events.events-prop	プロパティータイプによります。	ユーザーに関連するイベントのプロパティー。
name	String	ユーザーの名前。
lastname	String	ユーザーの名字を。
usrname	String	ユーザーの一意の名前。
department	String	ユーザーが属する部。
group	String	ユーザーが属するグループ。
title	String	ユーザーのタイトル。
status	String	ユーザーの状態。
role	String	ユーザーのロール。
tag	String	ユーザーが属するタグ。
pool	String	ユーザーが属するプール。
sortby	リスト	返された結果をリソースプロパティーの1つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

Users: Events.severity > normal and Vms.status = up or Vms.status = pause

この例では、仮想マシンで重大度が Normal よりも高いイベントが発生し、かつ仮想マシンがまだ稼働している場合や、ユーザーの仮想マシンが一時停止している場合のユーザーの一覧を返します。

3.19. イベントの検索

以下の表は、イベントの検索に使用できるすべての検索オプションについて説明しています。自動補完は、必要に応じて多くのオプションに対して提供されます。

表3.15 イベントの検索

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
Vms.Vms-prop	プロパティータイプによります。	イベントに関連付けられた仮想マシンのプロパティー。
Hosts.hosts-prop	プロパティータイプによります。	イベントに関連付けられたホストのプロパティー。
Templates.templates-prop	プロパティータイプによります。	イベントに関連付けられたテンプレートのプロパティー。
Users.users-prop	プロパティータイプによります。	イベントに関連付けられたユーザーのプロパティー。
Clusters.clusters-prop	プロパティータイプによります。	イベントに関連付けられたクラスターのプロパティー。
Volumes.Volumes-prop	プロパティータイプによります。	イベントに関連付けられたボリュームのプロパティー。
type	リスト	イベントのタイプ。
severity	リスト	イベントの重大度: Warning/Error/Normal
message	String	イベントタイプの説明。
time	リスト	イベントが発生した日。
usrname	String	イベントに関連付けられたユーザー名。
event_host	String	イベントに関連付けられたホスト。
event_vm	String	イベントに関連付けられた仮想マシン。

プロパティー (リソースまたはリソースタイプの)	タイプ	説明 (参照)
event_template	String	イベントに関連付けられたテンプレート。
event_storage	String	イベントに関連付けられたストレージ。
event_datacenter	String	イベントに関連付けられたデータセンター。
event_volume	String	イベントに関連付けられたボリューム。
correlation_id	Integer	イベントの識別番号。
sortby	リスト	返された結果をリソースプロパティーの1つで並べ替えます。
page	Integer	表示する結果のページ番号。

例

Events: Vms.name = testdesktop and Hosts.name = gonzo.example.com

この例では、ホスト **gonzo.example.com** での実行中に **testdesktop** という名前の仮想マシンで発生したイベントの一覧を返します。

第4章 ブックマーク

4.1. クエリー文字列をブックマークとして保存

ブックマークは、検索クエリーを記憶し、他のユーザーとの共有するために使用できます。

クエリー文字列をブックマークとして保存

1. 検索バーに希望の検索クエリーを入力し、検索を実行します。
2. 検索バーの右側にある星の **ブックマーク** ボタンをクリックして、**新規ブックマーク** ウィンドウを開きます。
3. ブックマークの **Name** を入力します。
4. 必要に応じて **Search string** フィールドを編集します。
5. **OK** をクリックします。

ヘッダーバーの **Bookmarks** アイコン () をクリックして、ブックマークを見つけて選択します。

4.2. ブックマークの編集

ブックマークの名前および検索文字列を変更できます。

ブックマークの編集

1. ヘッダーバーの **Bookmarks** アイコン () をクリックします。
2. ブックマークを選択し、**Edit** をクリックします。
3. 必要に応じて **Name** および **Search string** フィールドを変更します。
4. **OK** をクリックします。

4.3. ブックマークの削除

ブックマークが必要なくなったら、その設定を削除します。

ブックマークの削除

1. ヘッダーバーの **Bookmarks** アイコン () をクリックします。
2. ブックマークを選択し、**Remove** をクリックします。
3. **OK** をクリックします。

第5章 タグ

5.1. タグを使用して RED HAT VIRTUALIZATION とのやり取りをカスタマイズ

Red Hat Virtualization プラットフォームをセットアップし、要件に合わせて設定したら、タグを使用して作業方法をカスタマイズできます。タグを使用すると、システムリソースをグループまたはカテゴリーに分類できます。これは、仮想化環境に多くのオブジェクトが存在し、管理者が特定のオブジェクトセットに集中したい場合に便利です。

このセクションでは、タグの作成と編集、ホストまたは仮想マシンへの割り当て、タグを基準として使用した検索などの方法について説明します。タグは、企業のニーズに合わせて、構造に一致する階層に配置できます。

管理ポータルのタグを作成、変更、および削除するには、ヘッダーバーの **Tags** アイコン () をクリックします。

5.2. タグの作成

タグを作成して、タグを使用して検索結果を絞り込みできるようにします。

タグの作成

1. ヘッダーバーの **Tags** アイコン () をクリックします。
2. **Add** をクリックして新規タグを作成するか、タグを選択して **New** をクリックし、子孫タグを作成します。
3. 新規タグの **Name** および **Description** を入力します。
4. **OK** をクリックします。

5.3. タグの変更

タグの名前と説明を編集できます。

タグの変更

1. ヘッダーバーの **Tags** アイコン () をクリックします。
2. 変更するタグを選択し、**Edit** をクリックします。
3. 必要に応じて **Name** および **Description** フィールドを変更します。
4. **OK** をクリックします。

5.4. タグの削除

タグが不要になったら、それを削除します。

タグの削除

1. ヘッダーバーの **Tags** アイコン () をクリックします。

2. 削除するタグを選択し、**Remove** をクリックします。タグを削除すると、そのタグのすべての子孫も削除されることを警告するメッセージが表示されます。

3. **OK** をクリックします。

タグとその子孫をすべて削除しました。タグは、アタッチされたすべてのオブジェクトからも削除されます。

5.5. オブジェクトに対するタグの追加および削除

ホスト、仮想マシン、およびユーザーにタグを割り当てたり、削除したりできます。

オブジェクトに対するタグの追加および削除

1. タグ付けまたはタグ付け解除するオブジェクトを選択します。

2. **More Actions** (⋮) をクリックしてから **Assign Tags** をクリックします。

3. チェックボックスを選択してタグをオブジェクトに割り当てるか、選択を解除してオブジェクトからタグの割り当てを解除します。

4. **OK** をクリックします。

指定したタグが、選択したオブジェクトのカスタムプロパティーとして追加または削除されます。

5.6. タグを使用したオブジェクトの検索

tag プロパティーとしてタグを使用し、検索条件として目的の値または値のセットを使用して、検索クエリーを入力します。

指定された基準でタグ付けされたオブジェクトは結果リストに表示されます。

注記

tag をプロパティーとして使用し、不等式演算子 (**!=**、たとえば、**Host: Vms.tag!=server1**) を使用してオブジェクトを検索する場合、結果リストにはタグなしオブジェクトは含まれません。

5.7. タグを使用したホストのカスタマイズ

タグを使用してホストに関する情報を保存できます。その後、タグに基づいてホストを検索できます。検索の詳細は、3章*検索* を参照してください。

タグを使用したホストのカスタマイズ

1. **Compute → Hosts** をクリックし、ホストを選択します。

2. **More Actions** (⋮) をクリックしてから **Assign Tags** をクリックします。

3. 該当するタグのチェックボックスを選択します。

4. **OK** をクリックします。

ホストに関する検索可能な追加情報がタグとして追加されます。

パート II. リソースの管理

第6章 QOS (QUALITY OF SERVICE)

Red Hat Virtualization では、環境のリソースがアクセスできる入出力、処理、およびネットワーク機能のレベルを詳細に制御する QoS エントリーを定義できます。QoS (Quality of Service) エントリーはデータセンターレベルで定義され、クラスターおよびストレージドメイン下で作成されるプロファイルに割り当てられます。これらのプロファイルは、プロファイルが作成されたクラスターおよびストレージドメインの個々のリソースに割り当てられます。

6.1. ストレージ QOS

ストレージ QoS はスループットの最大レベルと、ストレージドメインの仮想ディスクの入出力操作の最大レベルを定義します。ストレージ QoS を仮想ディスクに割り当てると、ストレージドメインのパフォーマンスを細かく調整でき、1 つの仮想ディスクに関連付けられたストレージ操作が、同じストレージドメインでホストされる他の仮想ディスクで利用できるストレージ機能に影響しないようにすることができます。

6.1.1. ストレージ QoS エントリーの作成

ストレージ QoS エントリーの作成

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **QoS** タブをクリックします。
4. **Storage** で、**New** をクリックします。
5. **QoS Name** と QoS エントリーの **Description** を入力します。
6. 次のいずれかのラジオボタンをクリックして、**Throughput** Quality of Service を指定します。
   - **None**
   - **Total** - **MB/s** フィールドに最大許容合計スループットを入力します。
   - **Read/Write**: 左の**MB/s** フィールドに読み取り操作の最大許容スループットを入力し、右の **MB/s** フィールドに書き込み操作の最大許容スループットを入力します。
7. 次のいずれかのラジオボタンをクリックして、入出力 (**IOps**) の QoS を指定します。
   - **None**
   - **Total** - **IOps** フィールドに 1 秒あたりの入出力操作の最大許容数を入力します。
   - **Read/Write** - 左の **IOps** フィールドに 1 秒あたりの入力操作の最大許容数を入力し、右の **IOps** フィールドに 1 秒あたりの出力操作の最大許容数を入力します。
8. **OK** をクリックします。

ストレージ QoS エントリーが作成され、データセンターに属するデータストレージドメインのそのエントリーに基づいてディスクプロファイルを作成できます。

6.1.2. ストレージ Quality of Service エントリーの削除

既存のストレージ QoS(Quality of Service) エントリーを削除します。

ストレージ Quality of Service エントリーの削除

1. **Compute** → **Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **QoS** タブをクリックします。
4. **Storage** でストレージの QoS エントリーを選択し、 **Remove** をクリックします。
5. **OK** をクリックします。

そのエントリーに基づくディスクプロファイルが存在する場合、それらのプロファイルのストレージ QOS エントリーは自動的に **[unlimited]** に設定されます。

6.2. 仮想マシンのネットワーク QOS

仮想マシンネットワーク QoS は、個々の仮想ネットワークインターフェイスコントローラーの受信および送信トラフィックの両方を制限するためのプロファイルを作成できる機能です。この機能により、複数のレイヤーで帯域幅を制限し、ネットワークリソースの使用を制御できます。

6.2.1. 仮想マシンのネットワーク QoS エントリーの作成

仮想マシンネットワーク QoS エントリーを作成し、仮想ネットワークインターフェイスコントローラー (vNIC) プロファイル (仮想マシンネットワークインターフェイスプロファイル) に適用される際にネットワークトラフィックを規制します。

仮想マシンのネットワーク QoS エントリーの作成

1. **Compute** → **Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **QoS** タブをクリックします。
4. **VM Network** で、 **New** をクリックします。
5. 仮想マシンネットワーク QoS(Quality of Service) エントリーの **Name** を入力します。
6. **Inbound** および **Outbound** ネットワークトラフィックの制限を入力します。
7. **OK** をクリックします。

仮想ネットワークインターフェイスコントローラーで使用可能な仮想マシンネットワーク QoS エントリーが作成されました。

6.2.2. New Virtual Machine Network QoS および Edit Virtual Machine Network QoS ウインドウの設定の説明

仮想マシンのネットワーク QoS 設定により、3 つの異なるレベルで送受信トラフィックの両方に帯域幅の制限を設定できます。

表6.1 仮想マシンネットワーク QoS 設定

フィールド名	説明
Data Center	仮想マシンのネットワーク QoS ポリシーを追加するデータセンター。このフィールドは、選択したデータセンターに応じて自動的に設定されます。
Name	Manager 内の仮想マシンネットワーク QoS ポリシーを表す名前。
Inbound	受信トラフィックに適用される設定。**Inbound** チェックボックスを選択または選択解除して、これらの設定を有効または無効にします。 ● **Average**: 受信トラフィックの平均速度。 ● **Peak**: ピーク時の受信トラフィックの速度。 ● **Burst**: バースト中の受信トラフィックの速度。
Outbound	送信トラフィックに適用される設定。**Outbound** チェックボックスを選択または選択解除して、これらの設定を有効または無効にします。 ● **Average**: 送信トラフィックの平均速度。 ● **Peak**: ピーク時の送信トラフィックの速度。 ● **Burst**: バースト中の送信トラフィックの速度。

Average、Peak、または Burst フィールドによって許可される最大値を変更するには、**engine-config** コマンドを使用して **MaxAverageNetworkQoSValue**、**MaxPeakNetworkQoSValue**、または **MaxBurstNetworkQoSValue** の設定キーの値を変更します。変更を反映するには、**ovirt-engine** サービスを再起動する必要があります。以下に例を示します。

```
# engine-config -s MaxAverageNetworkQoSValue=2048
# systemctl restart ovirt-engine
```

6.2.3. 仮想マシンのネットワーク QoS(Quality of Service) エントリーの削除

既存の仮想マシンネットワーク QoS(Quality of Service) エントリーを削除します。

仮想マシンのネットワーク QoS(Quality of Service) エントリーの削除

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **QoS** タブをクリックします。

4. **VM Network** で、仮想マシンネットワーク QoS (Quality of Service) エントリーを選択して **Remove** をクリックします。

5. **OK** をクリックします。

6.3. ホストネットワーク QOS

ホストネットワーク QoS は、ホスト上のネットワークを設定し、物理インターフェイス経由のネットワークトラフィックの制御を可能にします。ホストネットワーク QoS により、同じ物理ネットワークインターフェイスコントローラー上のネットワークリソースの使用を制御することで、ネットワークのパフォーマンスをより細かく調整できます。これにより、1 つのネットワークが原因で、同じ物理ネットワークインターフェイスコントローラーにアタッチされている他のネットワークがトラフィックの輻輳により機能しなくなる状況を防ぐことができます。ホストネットワーク QoS 設定により、これらのネットワークは、輻輳問題なしに同じ物理ネットワークインターフェイスコントローラー上で機能できるようになります。

6.3.1. ホストネットワーク QoS エントリーの作成

ホストネットワーク QoS(Quality of Service) エントリーを作成します。

ホストネットワーク QoS エントリーの作成

1. **Compute → Data Centers** をクリックします。

2. データセンターの名前をクリックして、詳細ビューを開きます。

3. **QoS** タブをクリックします。

4. **Host Network** で、**New** をクリックします。

5. **QoS Name** と QoS エントリーの説明を入力します。

6. **Weighted Share**、**Rate Limit [Mbps]**、および **Committed Rate [Mbps]** に必要な値を入力します。

7. **OK** をクリックします。

6.3.2. New Host Network Quality of Service および Edit Host Network Quality of Service ウィンドウの設定の説明

ホストネットワーク QoS 設定により、送信トラフィックの帯域幅制限を設定できます。

表6.2 ホストネットワーク QoS 設定

フィールド名	説明
Data Center	ホストネットワーク QoS ポリシーを追加するデータセンター。このフィールドは、選択したデータセンターに応じて自動的に設定されます。
QoS Name	Manager 内のホストネットワーク QoS ポリシーを表す名前。
説明	ホストネットワーク QoS ポリシーの説明

フィールド名	説明
Outbound	送信トラフィックに適用される設定。 • **Weight Share**: 同じ論理リンクリンクにアタッチされた他のネットワークと比較して、特定のネットワークに割り当てる必要がある論理リンクの容量を指定します。正確な共有は、そのリンクの全ネットワークの共有の合計によって異なります。デフォルトでは、この値は 1-100 の範囲の数字になります。 • **Rate Limit [Mbps]**: ネットワークによって使用される最大帯域幅。 • **Committed Rate [Mbps]**: ネットワークに必要な最小帯域幅。要求される Committed Rate は保証されず、ネットワークインフラストラクチャーおよび同じ論理リンクの他のネットワークによって要求される Commmitted Rate によって異なります。

Rate Limit [Mbps] または **Committed Rate [Mbps]** フィールドで許可される最大値を変更するには、**engine-config** コマンドを使用して **MaxAverageNetworkQoSValue** 設定キーの値を変更します。変更を反映するには、**ovirt-engine** サービスを再起動する必要があります。以下に例を示します。

```
# engine-config -s MaxAverageNetworkQoSValue=2048
# systemctl restart ovirt-engine
```

6.3.3. ホストネットワーク QoS エントリーの削除

既存のネットワーク QoS(Quality of Service) エントリーを削除します。

ホストネットワーク QoS エントリーの削除

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **QoS** タブをクリックします。
4. **Host Network** で、ホストネットワーク QoS (Quality of Service) エントリーを選択して **Remove** をクリックします。
5. プロンプトが表示されたら **OK** をクリックします。

6.4. CPU QOS (QUALITY OF SERVICE)

CPU QoS は、仮想マシンが実行されているホスト上で仮想マシンがアクセスできる処理能力の最大量を定義します。これは、そのホストで使用可能な処理能力の合計に対する割合で表されます。CPU QoS を仮想マシンに割り当てると、クラスター内の 1 つの仮想マシンのワークロードが、そのクラスターの他の仮想マシンで利用できる処理リソースに影響を与えないようにすることができます。

6.4.1. CPU QoS エントリーの作成

CPU QoS (Quality of Service) エントリーを作成します。

CPU QoS エントリーの作成

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **QoS** タブをクリックします。
4. **CPU** で **New** をクリックします。
5. **QoS Name** と QoS エントリーの **Description** を入力します。
6. QoS (Quality of Service) エントリーで許可される最大処理能力を **Limit (%)** フィールドに入力します。**%** 記号は含めないでください。
7. **OK** をクリックします。

CPU QoS エントリーが作成され、データセンターに属するクラスターのそのエントリーに基づいて CPU プロファイルを作成できます。

6.4.2. CPU QoS エントリーの削除

既存の CPU QoS(Quality of Service) エントリーを削除します。

CPU QoS エントリーの削除

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **QoS** タブをクリックします。
4. **CPU** で CPU QoS エントリーを選択し、 **Remove** をクリックします。
5. **OK** をクリックします。

そのエントリーに基づく CPU プロファイルが存在する場合、それらのプロファイルの CPU QOS エントリーは自動的に **[unlimited]** に設定されます。

第7章 データセンター

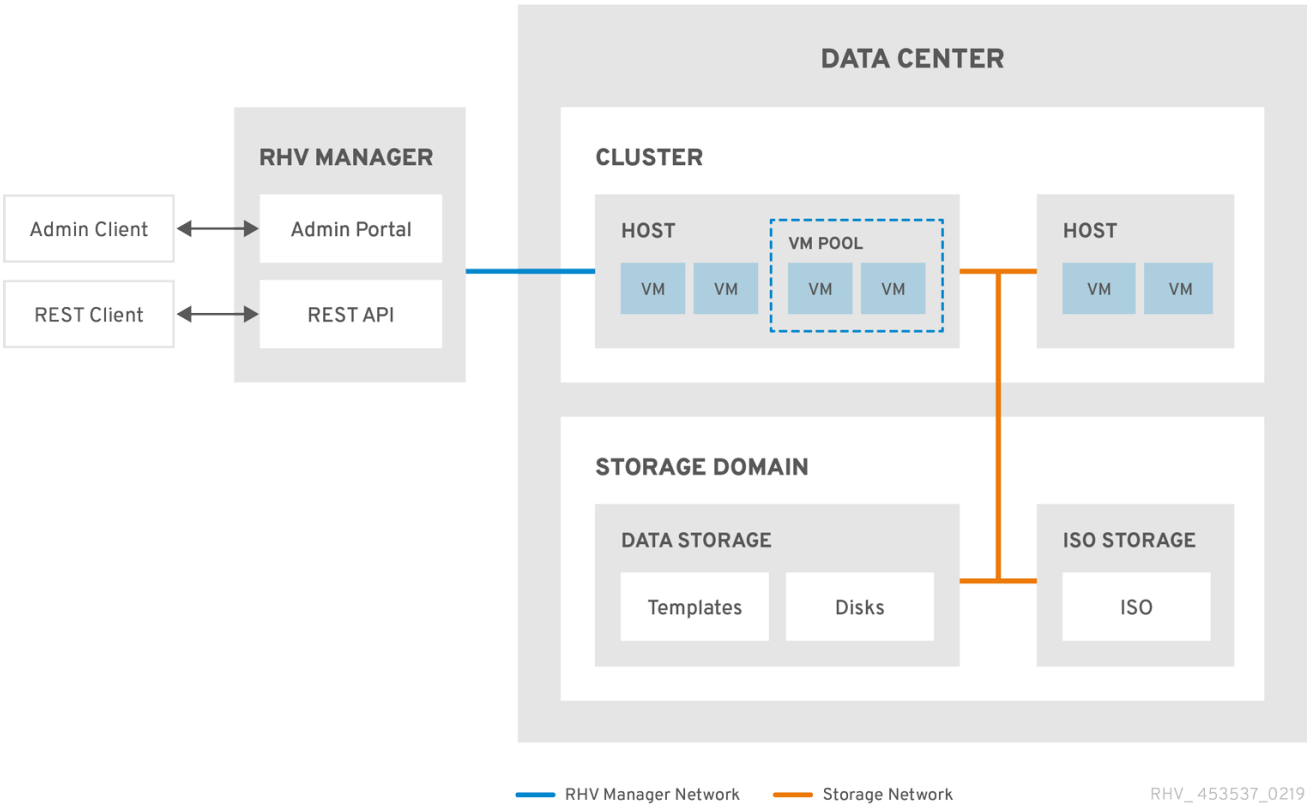
7.1. データセンターの概要

データセンターとは、特定の環境で使用するリソースのセットを定義する論理エンティティーです。データセンターは、コンテナーリソース (クラスターとホストの形式の論理リソースで設定)、ネットワークリソース (論理ネットワークと物理 NIC の形式)、およびストレージリソース (ストレージドメインの形式) と見なされます。

データセンターには、複数のホストを含む複数のクラスターを含めることができます。複数のストレージドメインが関連付けられており、各ホスト上の複数の仮想マシンをサポートすることができます。Red Hat Virtualization 環境には複数のデータセンターを含めることができます。データセンターインフラストラクチャーを使用すると、これらのセンターを分離した状態にすることができます。

すべてのデータセンターは、1 つの管理ポータルから管理されます。

図7.1 データセンター



Red Hat Virtualization は、インストール時にデフォルトのデータセンターを作成します。デフォルトのデータセンターを設定するか、または適切に名前が付けられたデータセンターを設定できます。

7.2. ストレージプールマネージャー

Storage Pool Manager(SPM) は、データセンター内のホストのいずれかに渡すロールで、データセンターのストレージドメインを管理できるようにします。SPM エンティティーはデータセンター内の任意のホストで実行できます。Red Hat Virtualization Manager はいずれかのホストにロールを付与します。SPM は標準の操作からホストを事前に設定しません。SPM として実行されているホストは依然として仮想リソースをホストできます。

SPM エンティティーは、ストレージドメイン全体でメタデータを調整することにより、ストレージへ

のアクセスを制御します。これには、仮想ディスク (イメージ)、スナップショット、およびテンプレートの作成、削除、およびテンプレート、およびスパースブロックデバイス (SAN 上) のストレージの割り当てが含まれます。これは唯一の責任です。メタデータの整合性を確保するために、データセンターの SPM となるホスト 1 つのみです。

Red Hat Virtualization Manager は、SPM が常に利用できることを確認します。SPM ホストがストレージへのアクセスに問題がある場合は、Manager は SPM ロールを別のホストに移動します。SPM が起動すると、これがロールが付与された唯一のホストであることを確認します。したがって、ストレージ中心のリースを取得します。このプロセスには時間がかかる場合があります。

7.3. SPM の優先度

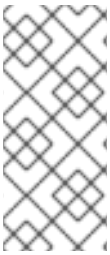
SPM ロールは、ホストの利用可能なリソースの一部を使用します。ホストの SPM 優先度の設定により、ホストが SPM ロールが割り当てられる可能性があります。SPM 優先度が高いホストには、SPM の優先度が低いホストの前に SPM ロールが割り当てられます。SPM 優先度が低いホストの重要な仮想マシンは、ホストリソースの SPM 操作と連動させる必要はありません。

Edit Host ウィンドウの SPM タブでホストの **SPM** タブの優先度を変更できます。

7.4. データセンタータスク

7.4.1. 新規データセンターの作成

以下の手順では、お使いの仮想化環境にデータセンターを作成します。データセンターが機能するには、機能しているクラスター、ホスト、およびストレージドメインが必要です。



注記

互換バージョン を設定すると、後で低くすることはできず、バージョンを下げることはできません。

データセンターに MAC プール範囲を指定するオプションが無効になり、クラスターレベルで実行されるようになりました。

新規データセンターの作成

1. **Compute → Data Centers** をクリックします。
2. **New** をクリックします。
3. データセンターの **Name** および **Description** を入力します。
4. ドロップダウンメニューから、データセンターの **Storage Type**、**Compatibility Version**、**Quota Mode** を選択します。
5. **OK** をクリックしてデータセンターを作成し、**データセンター - Guide Me** ウィンドウを開きます。
6. **Guide Me** ウィンドウには、データセンター用に設定する必要のあるエンティティーが一覧表示されます。これらのエンティティーを設定するか、**Configure Later** ボタンをクリックして設定を延期します。設定を再開するには、データセンターを選択し、**More Actions** (⋮) をクリックしてから **Guide Me** をクリックします。

新しいデータセンターは、クラスター、ホスト、およびストレージドメインが設定されるまで **Uninitialized** になります。**Guide Me** を使用してこれらのエンティティーを設定します。

7.4.2. New Data Center および Edit Data Center Windows の設定についての説明

以下の表は、**New Data Center** および **Edit Data Center** ウィンドウに表示されるデータセンターの設定について説明しています。**OK** をクリックすると、無効なエントリーがオレンジ色で囲まれ、変更が承認されません。さらに、フィールドプロンプトは予想される値または値の範囲を示します。

表7.1 データセンターのプロパティー

フィールド	説明/アクション
Name	データセンターの名前。このテキストフィールドには 40 文字の制限があり、大文字、小文字、数字、ハイフン、およびアンダースコアの組み合わせが含まれる一意の名前である必要があります。
説明	データセンターの説明このフィールドは推奨されますが、必須ではありません。
Storage Type	**Shared** または **Local** ストレージタイプを選択します。 異なるタイプのストレージドメイン (iSCSI、NFS、FC、POSIX、および Gluster) を同じデータセンターに追加できます。ただし、ローカルドメインおよび共有ドメインは混在させることはできません。 データセンターの初期化後にストレージタイプを変更できます。「データセンターストレージタイプの変更」 を参照してください。
互換バージョン	Red Hat Virtualization のバージョン。 Red Hat Virtualization Manager をアップグレードした後には、ホスト、クラスター、およびデータセンターが以前のバージョンにある可能性があります。データセンターの互換性レベルをアップグレードする前に、すべてのホストをアップグレードし、クラスターをアップグレードしていることを確認します。

フィールド	説明/アクション
クォータモード	クォータは、Red Hat Virtualization で提供されるリソース制限ツールです。以下のいずれかを選択します。 ● **Disabled**: クォータを実装しない場合に選択します。 ● **Audit**: クォータ設定を編集する場合に選択します。 ● **Enforced**: クォータを実装する場合に選択します。
Comment	オプションで、データセンターに関するプレーンテキストコメントを追加します。

7.4.3. データセンターの再初期化: リカバリー手順

この復旧手順は、データセンターのマスターデータドメインを新しいマスターデータドメインに置き換えます。データが破損している場合は、マスターデータドメインを再初期化する必要があります。データセンターを再初期化すると、クラスター、ホスト、および問題以外のストレージドメインなど、データセンターに関連付けられた他のリソースをすべて復元できます。

バックアップまたはエクスポートした仮想マシンまたはテンプレートを新しいマスターデータドメインにインポートできます。

データセンターの再初期化

1. **Compute** → **Data Centers** をクリックし、データセンターを選択します。
2. データセンターに接続されたストレージドメインがメンテナンスモードにあることを確認します。
3. **More Actions** (⋮) をクリックしてから、**Re-Initialize Data Center** をクリックします。
4. **Data Center Re-Initialize** ウィンドウには、利用可能なすべての (割り当て解除あり、メンテナンスモードの場合) ストレージドメインが一覧表示されます。データセンターに追加するストレージドメインのラジオボタンをクリックします。
5. **Approve operation** チェックボックスを選択します。
6. **OK** をクリックします。

ストレージドメインは、マスターデータドメインとしてデータセンターにアタッチされ、アクティベートされます。バックアップまたはエクスポートした仮想マシンまたはテンプレートを新しいマスターデータドメインにインポートできるようになりました。

7.4.4. データセンターの削除

データセンターを削除するには、アクティブなホストが必要です。データセンターを削除しても、関連付けられたリソースは削除されません。

データセンターの削除

1. データセンターに接続されたストレージドメインがメンテナーンスモードにあることを確認します。
2. **Compute** → **Data Centers** をクリックし、削除するデータセンターを選択します。
3. **Remove** をクリックします。
4. **OK** をクリックします。

7.4.5. データセンターの強制削除

アタッチされたストレージドメインが破損したり、ホストが **Non Responsive** になった場合、データセンターが **Non Responsive** になります。いずれの状況においても、データセンターを **削除** できません。

Force Remove では、アクティブなホストは必要ありません。また、アタッチされているストレージドメインも完全に削除します。

データセンターを **Force Remove** する前に、破損したストレージドメインを **破棄** する必要がある場合があります。

データセンターの強制削除

1. **Compute** → **Data Centers** をクリックし、削除するデータセンターを選択します。
2. **More Actions** (⋮) をクリックしてから、**Force Remove** をクリックします。
3. **Approve operation** チェックボックスを選択します。
4. **OK** をクリックします。

データセンターおよび割り当てられたストレージドメインは、Red Hat Virtualization 環境から完全に削除されます。

7.4.6. データセンターストレージタイプの変更

データセンターの初期化後に、データセンターのストレージタイプを変更できます。これは、仮想マシンまたはテンプレートの移動に使用されるデータドメインに役立ちます。

制限事項

- ローカルデータセンターの共有 - 複数のホストおよび複数のクラスターを含まないデータセンターの場合は、ローカルデータセンターがこれをサポートしないためです。
- Local to Shared: ローカルストレージドメインを含まないデータセンターの場合。

データセンターストレージタイプの変更

1. **Compute** → **Data Centers** をクリックし、変更するデータセンターを選択します。
2. **Edit** をクリックします。
3. **Storage Type** を必要な値に変更します。
4. **OK** をクリックします。

7.4.7. データセンターの互換バージョンの変更

Red Hat Virtualization データセンターには、互換バージョンがあります。互換バージョンとは、データセンターが互換性を持つ Red Hat Virtualization のバージョンを指します。データセンター内のクラスターは、すべて指定の互換性レベルをサポートする必要があります。

重要

データセンターの互換バージョンを変更するには、事前にデータセンター内のクラスターおよび仮想マシンの互換バージョンがすべて更新されている必要があります。

手順

1. 管理ポータルで **コンピュート → データセンター** をクリックします。
2. 変更を行うデータセンターを選択し、**編集** をクリックします。
3. **互換バージョン** を必要な値に変更します。
4. **OK** をクリックします。**データセンターの互換バージョンを変更** の確認ダイアログが開きます。
5. **OK** をクリックして確定します。

7.5. データセンターおよびストレージドメイン

7.5.1. 既存のデータドメインをデータセンターにアタッチ

アタッチされていない データドメインは、データセンターにアタッチすることができます。複数のタイプ (iSCSI、NFS、FC、POSIX、および Gluster) の共有ストレージドメインを同じデータセンターに追加できます。

既存のデータドメインをデータセンターにアタッチ

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **Storage** タブをクリックして、データセンターにすでにアタッチされているストレージドメインを一覧表示します。
4. **Attach Data** をクリックします。
5. データセンターにアタッチするデータドメインのチェックボックスを選択します。複数のデータドメインを割り当てる場合は、複数のチェックボックスを選択できます。
6. **OK** をクリックします。

データドメインはデータセンターにアタッチされ、自動的にアクティブになります。

7.5.2. 既存の ISO ドメインをデータセンターにアタッチ

Unattached の ISO ドメインは、データセンターにアタッチすることができます。ISO ドメインは、データセンターと同じ **Storage Type** である必要があります。

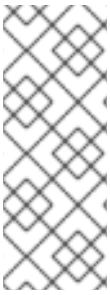
データセンターに 1 つの ISO ドメインのみをアタッチできます。

既存の ISO ドメインをデータセンターにアタッチ

1. **Compute** → **Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **Storage** タブをクリックして、データセンターにすでにアタッチされているストレージドメインを一覧表示します。
4. **Attach ISO** をクリックします。
5. 適切な ISO ドメインのラジオボタンをクリックします。
6. **OK** をクリックします。

ISO ドメインはデータセンターにアタッチされ、自動的にアクティブになります。

7.5.3. 既存のエクスポートドメインをデータセンターにアタッチ



注記

エクスポートストレージドメインは非推奨になりました。ストレージデータドメインはデータセンターから接続を解除し、同じ環境または別の環境にある別のデータセンターにインポートすることができます。仮想マシン、フローティング仮想ディスク、およびテンプレートは、インポートされたストレージドメインからアタッチされたデータセンターにアップロードできます。ストレージドメインのインポートに関する詳細は、「既存のストレージドメインのインポート」 を参照してください。

Unattached のドメインは、データセンターにアタッチ することができます。データセンターには、エクスポートドメインを 1 つだけアタッチできます。

既存のエクスポートドメインをデータセンターにアタッチ

1. **Compute** → **Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **Storage** タブをクリックして、データセンターにすでにアタッチされているストレージドメインを一覧表示します。
4. **Attach Export** をクリックします。
5. 適切なエクスポートドメインのラジオボタンをクリックします。
6. **OK** をクリックします。

エクスポートドメインはデータセンターにアタッチされ、自動的にアクティブになります。

7.5.4. データセンターからストレージドメインをデタッチ

データセンターからストレージドメインをデタッチすると、データセンターがそのストレージドメインとの関連付けをできなくなります。ストレージドメインは Red Hat Virtualization 環境から削除されず、別のデータセンターにアタッチすることができます。

仮想マシンやテンプレートなどのデータは、引き続きストレージドメインにアタッチされます。

注記

マスターストレージ (これが最後の利用可能なストレージドメインである場合) は削除できません。

データセンターからストレージドメインをデタッチ

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **Storage** タブをクリックして、データセンターにアタッチされているストレージドメインを一覧表示します。
4. デタッチするストレージドメインを選択します。ストレージドメインが **Active** の場合は、**Maintenance** をクリックします。
5. **OK** をクリックしてメンテナーンスモードを開始します。
6. **デタッチ** をクリックします。
7. **OK** をクリックします。

ストレージドメインが詳細ビューから消えるまでに数分かかる場合があります。

第8章 クラスター

8.1. クラスターの概要

クラスターは、同じストレージドメインを共有し、同じタイプの CPU(Intel または AMD) を持つホストの論理グループです。ホストに異なる CPU モデルの生成がある場合は、すべてのモデルに存在する機能のみを使用します。

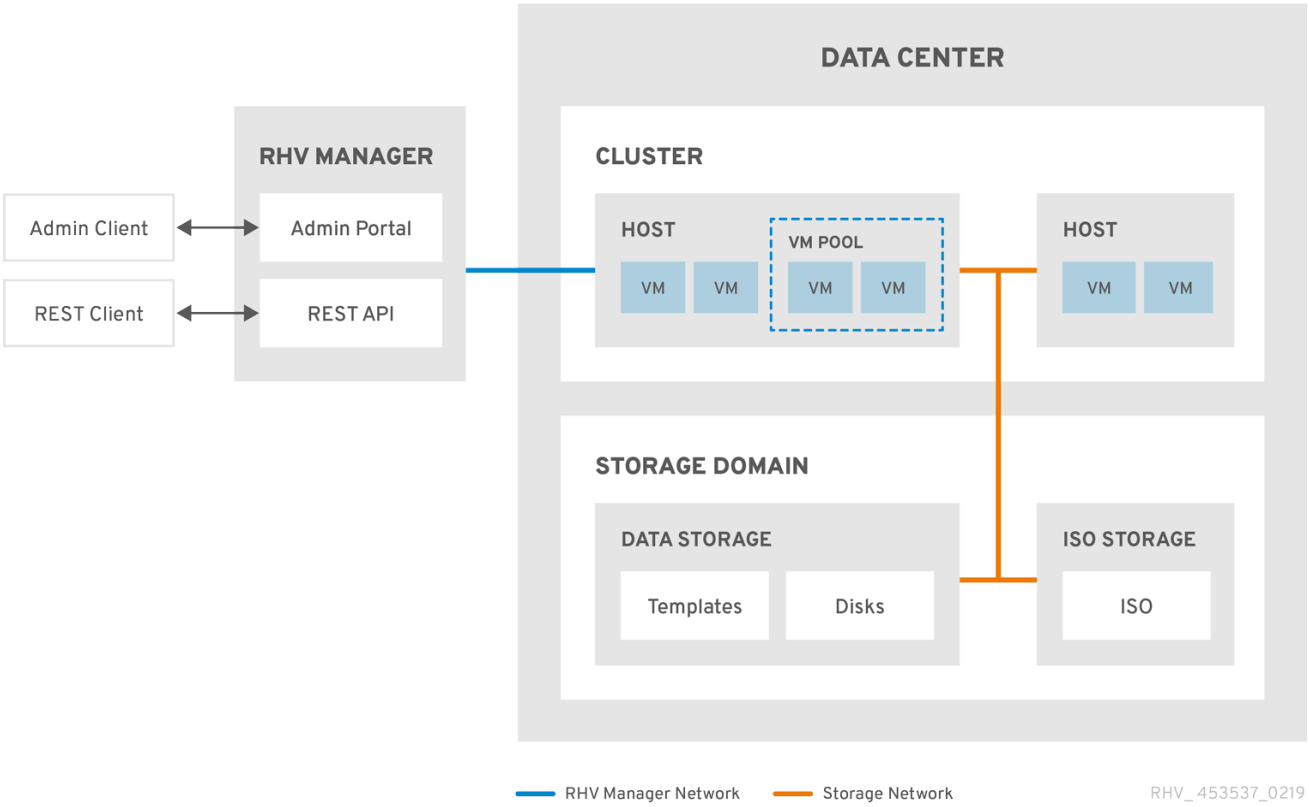
システム内の各クラスターはデータセンターに属し、システム内の各ホストはクラスターに属している必要があります。仮想マシンはクラスター内の任意のホストに動的に割り当てられ、仮想マシン上のクラスターおよび設定に合わせて、それらのホスト間で移行することができます。クラスターは、電源および負荷分散ポリシーを定義できる最上位です。

クラスターに属するホストおよび仮想マシンの数は、**Host Count** および **VM Count** のそれぞれ結果一覧にそれぞれ表示されます。

クラスターは仮想マシンまたは Red Hat Gluster Storage サーバーを実行します。これら 2 つの目的は相互排他的です。単一クラスターでは仮想化とストレージホストをまとめてサポートできません。

Red Hat Virtualization は、インストール時にデフォルトのデータセンターにデフォルトのクラスターを作成します。

図8.1 Cluster



8.2. クラスタータスク

注記

一部のクラスターオプションは Gluster クラスターには適用されません。Red Hat Virtualization で Red Hat Gluster Storage を使用する方法の詳細は、**Configuring Red Hat Virtualization with Red Hat Gluster Storage** を参照してください。

8.2.1. 新規クラスターの作成

データセンターには複数のクラスターを含めることができ、クラスターには複数のホストを含めることができます。クラスター内のホストすべては、同じ CPU タイプ (Intel または AMD) である必要があります。CPU タイプの最適化を確保するには、クラスターを作成する前にホストを作成することが推奨されます。ただし、**Guide Me** ボタンを使用してホストを後で設定できます。

新規クラスターの作成

1. **Compute → Clusters** 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からクラスターが所属する**Data Center** を選択します。
4. クラスターの **Name** および **Description** を入力します。
5. **Management Network** のドロップダウンリストからネットワークを選択して、管理ネットワークロールを割り当てます。
6. ドロップダウンリストから **CPU Architecture** と **CPU Type** を選択します。クラスターに接続するホストの最小 CPU プロセッサータイプと、CPU プロセッサーファミリーとを一致させることが重要です。そうでないと、ホストが機能しなくなります。

注記

Intel タイプおよび AMD CPU タイプの両方の場合、リストされた CPU モデルは、最も古いものから最新の順に論理的に使用されます。CPU モデルが異なるホストがクラスターに含まれる場合は、最も古い CPU モデルを選択します。各 CPU モデルの詳細は、https://access.redhat.com/solutions/634853 を参照してください。

7. ドロップダウンリストから、クラスターの**互換バージョン**を選択します。
8. ドロップダウンリストから **Switch Type** を選択します。
9. クラスターのホストに **Firewall Type** を選択します (**iptables** または **firewalld** のいずれか)。

注記

iptables は非推奨になりました。

10. **Enable Virt Service**または **Enable Gluster Service**チェックボックスを選択して、クラスターが仮想マシンホストまたは Gluster 対応ノードと共に設定されるかどうかを定義します。
11. 仮想マシンを Manager からシャットダウンする際に任意の reason フィールドを有効にするように**Enable to set VM maintenance reason**のチェックボックスを選択し、管理者がメンテナンスの説明を提供できるようにします。

12. ホストを Manager からメンテナンスモードにする時に任意の reason フィールドを有効にするように**Enable to set Host maintenance reason**のチェックボックスを選択し、管理者がメンテナンスの説明を提供できるようにします。

13. オプションで **/dev/hwrng ソース**(外部ハードウェアデバイス) のチェックボックスを選択し、クラスター内のすべてのホストが使用する乱数ジェネレーターデバイスを指定します。**/dev/urandom ソース**(Linux が提供するデバイス) はデフォルトで有効です。

14. **Optimization**タブをクリックしてクラスターのメモリーページ共有のしきい値を選択し、必要に応じてクラスター内のホストで CPU スレッド処理とメモリーバルーンを有効にします。

15. **Migration Policy** タブをクリックして、クラスターの仮想マシン移行ポリシーを定義します。

16. **Scheduling Policy**(スケジューリングポリシー) タブをクリックして、スケジューリングポリシーの設定、スケジューラー最適化の設定、クラスター内のホストの信頼できるサービスの有効化、HA Reservation の有効化を行い、カスタムのシリアル番号ポリシーを追加します。

17. **Console** タブをクリックしてオプションでグローバル SPICE プロキシーを上書きし (ある場合)、クラスター内のホストの SPICE プロキシーのアドレスを指定します。

18. **Fencing policy** タブをクリックして、クラスターでフェンシングを有効または無効にします。また、フェンスオプションを選択します。

19. **MAC Address Pool**タブをクリックして、クラスターのデフォルトのプール以外の MAC アドレスプールを指定します。MAC アドレスプールの作成、編集、または削除に関する詳細は、「MAC アドレスプール」 を参照してください。

20. **OK** をクリックしてクラスターを作成し、**Cluster - Guide Me** ウインドウを開きます。

21. **Guide Me** ウィンドウには、クラスターに設定する必要のあるエンティティーが一覧表示されます。これらのエンティティーを設定するか、**Configure Later** ボタンをクリックして設定を延期します。設定を再開するには、クラスターを選択し、**More Actions** (⋮) をクリックしてから、**Guide Me** をクリックします。

8.2.2. 一般的なクラスター設定に関する説明

以下の表は、**New Cluster** および **Edit Cluster** ウィンドウの **General** タブの設定について説明しています。**OK** をクリックすると、無効なエントリーがオレンジ色で囲まれ、変更が承認されません。さらに、フィールドプロンプトは予想される値または値の範囲を示します。

表8.1 一般的なクラスター設定

フィールド	説明/アクション
Data Center	クラスターが含まれるデータセンター。クラスターを追加する前にデータセンターを作成する必要があります。
Name	クラスターの名前。このテキストフィールドには 40 文字の制限があり、大文字、小文字、数字、ハイフン、およびアンダースコアの組み合わせが含まれる一意の名前である必要があります。

フィールド	説明/アクション
Description / Comment	クラスターまたは追加のメモの説明。これらのフィールドは推奨されますが、必須ではありません。
Management Network	管理ネットワークロールを割り当てる論理ネットワーク。デフォルトは **ovirtmgmt** です。移行ネットワークが移行元または移行先ホストに正しくアタッチされていない場合は、このネットワークは仮想マシンの移行にも使用されます。 既存のクラスターでは、詳細ビューの **Logical Networks** タブにある **Manage Networks** ボタンを使用すると管理ネットワークを変更できます。
CPU アーキテクチャー	クラスターの CPU アーキテクチャー。選択した CPU アーキテクチャーに応じて、さまざまな CPU タイプを利用できます。 ● **undefined**: すべての CPU タイプが利用できます。 ● **x86_64**: Intel および AMD CPU の全タイプが利用できます。 ● **ppc64**: IBM POWER 8 のみが利用できます。
CPU Type	クラスターの CPU タイプ。サポートされる CPU タイプの一覧は、**Planning and Prerequisites Guide**の CPU Requirements を参照してください。クラスター内のホストはすべて、Intel、AMD、または IBM POWER 8 の CPU タイプのいずれかを実行している必要があり、これは作成後には、大きな中断なしで変更することはできません。CPU タイプは、クラスター内の最も古い CPU モデルに設定する必要があります。すべてのモデルに存在する機能のみを使用できます。Intel タイプおよび AMD CPU タイプの両方の場合、リストされた CPU モデルは、最も古いものから最新の順に論理的に使用されます。
互換バージョン	Red Hat Virtualization のバージョン。データセンターに指定したバージョンよりも、以前のバージョンを選択することはできません。
Switch Type	クラスターが使用するスイッチのタイプ。**Linux Bridge** は、標準の Red Hat Virtualization スイッチです。**OVS** は Open vSwitch のネットワーク機能をサポートします。

フィールド	説明/アクション
Firewall Type	クラスター内のホストのファイアウォールタイプを指定します (**iptables** または **firewalld** のいずれか)。 **注記: iptables** は 非推奨になりました。 既存のクラスターのファイアウォールタイプを変更する場合は、クラスターで すべてのホストを再インストールして、変更を適用する必要があります。
Default Network Provider	クラスターが使用するデフォルトの外部ネットワークプロバイダーを指定します。Open Virtual Network (OVN) を選択する場合、クラスターに追加されたホストは OVN プロバイダーと通信するように自動的に設定されます。 デフォルトのネットワークプロバイダーを変更する場合には、クラスターのすべてのホストを再インストールして、変更を適用する必要があります。
ログメモリーの最大しきい値	最大メモリー消費のロギングしきい値をパーセンテージまたは絶対値 (MB 単位) で指定します。ホストのメモリー使用量がパーセンテージ値を超えている場合や、ホストで利用可能なメモリーが絶対値 (MB 単位) を下回る場合にログに記録されます。デフォルトは **95%** です。
Enable Virt Service	このラジオボタンを選択すると、このクラスター内のホストは仮想マシンを実行するために使用されます。
Enable Gluster Service	このラジオボタンが選択される場合、このクラスターのホストは Red Hat Gluster Storage Server ノードとして使用されるため、仮想マシンの実行には使用されません。

フィールド	説明/アクション
Import existing gluster configuration	このチェックボックスは、**Enable Gluster Service** ラジオボタンが選択されている場合にのみ利用できます。このオプションを使用すると、既存の Gluster 対応クラスターおよびその割り当てられたすべてのホストを Red Hat Virtualization Manager にインポートできます。 以下のオプションは、インポートされているクラスター内のホストごとに必要になります。 ● **Address**: Gluster ホストサーバーの IP または完全修飾ドメイン名を入力します。 ● **フィンガープリント**: Red Hat Virtualization Manager はホストのフィンガープリントを取得して、正しいホストに接続します。 ● **Root Password**: ホストとの通信に必要な root パスワードを入力します。
Enable to set VM maintenance reason	このチェックボックスを選択すると、クラスターの仮想マシンが Manager からシャットダウンすると、オプションの reason フィールドが表示されます。これにより、ログに表示されるメンテナーンスの説明と、仮想マシンの電源が再びオンになります。
ホストのメンテナーンス理由の設定の有効化	このチェックボックスを選択すると、クラスターのホストが Manager からメンテナーンスモードに移動すると、オプションの reason フィールドが表示されます。これにより、ログに表示されるメンテナーンスの説明と、ホストが再度アクティベートされたタイミングを指定できます。
追加の乱数ジェネレーターソース	このチェックボックスを選択した場合には、クラスター内の全ホストには、追加の乱数ジェネレーターデバイスが利用可能になります。これにより、乱数ジェネレーターデバイスから仮想マシンへのエントロピーのパススルーが可能になります。

8.2.3. 最適化設定の説明

メモリーに関する考慮事項

メモリーページの共有により、仮想マシンは、他の仮想マシンで未使用のメモリーを利用することで、割り当てられたメモリーの最大 200% を使用できます。このプロセスは、Red Hat Virtualization 環境内の仮想マシンが同時に実行されるという前提であり、未使用のメモリーを特定の仮想マシンに一時的に割り当てられるようにします。

CPU の考慮事項

- **CPU 集約型ではないワークロードの場合**、ホスト内のコア数よりも大きいプロセッサーコアの合計数を持つ仮想マシンを実行できます。これを実行することで、以下が可能になります。
  - より多くの仮想マシンを実行することができます。これにより、ハードウェアの要件が減少します。
  - 仮想コアの数がホストコア数とホストスレッドの数の間にある場合など、それ以外の CPU トポロジーで仮想マシンを設定できます。
- **最適なパフォーマンス、特に CPU 集約型のワークロードの場合**、ホストと同じトポロジーを使用する必要があります。ホストと仮想マシンは同じキャッシュの使用を期待します。ホストのハイパースレッディングが有効な場合、QEMU がホストのハイパースレッドをコアとして扱うため、仮想マシンは複数のスレッドを持つ単一のコアで実行されていることを認識しません。ホストコアのハイパースレッドに実際に対応する仮想コアは、仮想マシンのパフォーマンスに影響する可能性があります。これは、同じホストコアのハイパースレッドと単一のキャッシュを共有する可能性がありますが、仮想マシンは別のコアとして扱います。

以下の表は、**New Cluster** および **Edit Cluster** ウィンドウの **Optimization** タブの設定について説明しています。

表8.2 最適化の設定

フィールド	説明/アクション
メモリーの最適化	• **None - Disable memory overcommit** メモリーページ共有を無効にします。 • **Server Load - Allow scheduling of 150% of physical memory**: 各ホストのシステムメモリーのメモリーページ共有のしきい値を 150% に設定します。 • **For Desktop Load - Allow scheduling of 200% of physical memory**: 各ホストのシステムメモリーの 200% にメモリーページ共有のしきい値を設定します。
CPU スレッド	**Count Threads As Cores** チェックボックスを選択すると、ホストは、ホストのコア数よりも大きいプロセッサーコアの合計数で仮想マシンを実行することができます。 このチェックボックスを選択すると、公開されるホストスレッドは仮想マシンが使用できるコアとして扱われます。たとえば、コアごとに 2 つのスレッドがある 24 コアのシステム (全部で 48 スレッド) では、それぞれ最大 48 コアを持つ仮想マシンを実行できます。そして、ホストの CPU 負荷を計算するアルゴリズムは、使用量の多くのコアを 2 回比較します。

フィールド	説明/アクション
メモリーバルーン	**Enable Memory Balloon Optimization** のチェックボックスを選択し、このクラスターのホストで実行している仮想マシンでメモリーのオーバーコミットを有効にします。このチェックボックスを選択すると、Memory Overcommit Manager(MoM) は、可能な限りバルーニングを開始し、可能な場合はすべての仮想マシンのメモリーサイズが保証されます。 バルーンが実行しているには、仮想マシンに適切なドライバーを持つバルーンデバイスが必要です。各仮想マシンには、特別に削除しない限り、バルーンデバイスが含まれます。このクラスター内の各ホストは、ステータスが **Up** に変わったときにバルーンポリシーの更新を受け取ります。必要に応じて、ステータスを変更せずに、ホストのバルーンポリシーを手動で更新できます。「クラスター内のホストでの MoM ポリシーの更新」 を参照してください。 シナリオのバルーンでは、KSM と一致している可能性がある点を理解することが重要です。このような場合は、MoM は競合の可能性を最小限に抑えるためにバルーンサイズの調整を試みます。さらに、シナリオのバルーンによっては、仮想マシンに対して最大のパフォーマンスが最適化される可能性があります。管理者は、バルーンの最適化を注意して使用することが推奨されます。
KSM コントロール	**Enable KSM** チェックボックスを選択すると、MoM が必要に応じて Kernel Same-page Merging(KSM) を実行できるため、メモリーの保存で CPU コストの重み付けがメリットを得ることができます。

8.2.4. 移行ポリシー設定の説明

移行ポリシーは、ホストに障害が発生した場合に仮想マシンのライブマイグレーションの条件を定義します。これらの状態には、移行中の仮想マシンのダウンタイム、ネットワーク帯域幅、および仮想マシンの優先順位が含まれます。

表8.3 移行ポリシーの説明

Policy	説明
Legacy	3.6 バージョンのレガシー動作**vdsm.conf** のオーバーライドは引き続き適用されます。ゲストエージェントフックメカニズムが無効になっている。

Policy	説明
最小ダウンタイム	仮想マシンを一般的な状況で移行できるようにするポリシー。仮想マシンは、ダウンタイムを大幅に発生しません。移行は、長時間 (QEMU の反復により最大 500 ミリ秒) 後に仮想マシンの移行が収束されない場合に中止されます。ゲストエージェントフックメカニズムが有効になっている。
コピー後の移行	仮想マシンは、ダウンタイムを最小限に抑えるポリシーと同様の、大きなダウンタイムは発生しません。コピー後のポリシーは、最初に事前コピーして、コンバージェンスが発生したかどうかを検証します。仮想マシンの移行が長時間後に収束されない場合、移行は post-copy に切り替わります。このポリシーの欠点は、コピー後のフェーズでは、メモリーの不足している部分がホスト間で転送されるため、仮想マシンが大幅に遅くなる可能性があります。 ホスト間の移行ネットワークのネットワーク障害など、コピー後のフェーズで異常が発生した場合は、移行プロセスで仮想マシンが一貫性がなく、一時停止されて結果が失われます。そのため、コピー後のフェーズで移行を中止することはできません。 **制限** コピー後のプロセスの完了前にネットワーク接続が破損すると、Manager は一時停止し、実行中の仮想マシンを強制終了します。仮想マシンの可用性が重要である場合や、移行ネットワークが不安定な場合は、コピー後の移行を使用しないでください。
必要に応じてワークロードを一時停止	負荷が大きい仮想マシンを含む、ほとんどの状況で仮想マシンが移行できるようにするポリシー。このため、仮想マシンでは、他の設定よりも大きなダウンタイムが生じる場合があります。移行は、極端なワークロードに対して中止される場合があります。ゲストエージェントフックメカニズムが有効になっている。

帯域幅設定は、ホストごとの送信移行と受信両方の最大帯域幅を定義します。

表8.4 帯域幅の説明

Policy	説明
Auto	帯域幅は、データセンターの **Host Network QoS** の **Rate Limit [Mbps]** 設定からコピーされます。レート制限が定義されていない場合は、ネットワークインターフェイス送受信の最小リンク速度として計算されます。レート制限が設定されていない場合や、リンク速度が利用できない場合には、ホスト送信時にローカルの VDSM 設定により決定されます。
ハイパーバイザーのデフォルト	帯域幅は、ホスト送信時にローカルの VDSM 設定によって制御されます。
カスタム	ユーザーで定義されます (Mbps 単位)。この値は、同時移行の数 (デフォルトは 2 で、継続的な移行および送信移行のために考慮するには 2) で分割されます。したがって、ユーザー定義の帯域幅は、すべての同時移行に対応できるように十分な大きさである必要があります。 たとえば、**Custom** 帯域幅が 600 Mbps として定義されている場合、仮想マシンの移行の最大帯域幅は実際には 300 Mbps になります。

耐障害性ポリシーは、移行での仮想マシンの優先順位を定義します。

表8.5 耐障害性ポリシーの設定

フィールド	説明/アクション
仮想マシンの移行	定義された優先順位で、すべての仮想マシンを移行します。
高可用性仮想マシンのみの移行	他のホストのオーバーロードを防ぐために、高可用性の仮想マシンのみを移行します。
Do not migrate Virtual Machines(**仮想マシンを移行しない**)	仮想マシンを移行しないようにします。

Additional Properties は、Legacy 移行ポリシーにのみ適用されます。

表8.6 その他のプロパティーの説明

プロパティー	説明

プロパティー	説明
自動コンバージョン	仮想マシンのライブマイグレーション中に自動コンバージェンスが使用されるかどうかを設定できます。負荷が大きいが大きい仮想マシンでは、ライブマイグレーション中に行われる転送速度よりも速くメモリーがダーティーなり、移行が収束できなくなります。QEMU の自動調整機能を使用すると、仮想マシン移行の収束を強制的に実行できます。QEMU は、コンバージェンスの欠如を自動的に検出し、仮想マシン上の vCPU のスロットルダウンをトリガーします。オートコンバージェンスはデフォルトで無効になっています。 ● グローバルレベルで設定される auto-convergence 設定を使用するには、**Inherit from global setting** を選択します。このオプションはデフォルトで選択されます。 ● **Auto Converge** を選択してグローバル設定を上書きし、仮想マシンの自動調整を許可します。 ● **Don't Auto Converge** を選択してグローバル設定を上書きし、仮想マシンの自動調整を防ぎます。
移行圧縮の有効化	仮想マシンのライブマイグレーション中に移行圧縮を使用するかどうかを設定できます。この機能は、Xor Binary Zero Run-Length-Encoding を使用して、メモリー書き込みを必要とするワークロードまたはスパースメモリー更新パターンを使用するアプリケーションに対して、仮想マシンのダウンタイムと合計移行時間を短縮します。移行圧縮は、デフォルトでは無効になっています。 ● グローバルレベルで設定する圧縮設定を使用するには、**Inherit from global setting** を選択します。このオプションはデフォルトで選択されます。 ● **Compress** を選択してグローバル設定を上書きし、仮想マシンの圧縮を許可します。 ● **Don't compress** を選択してグローバル設定を上書きし、仮想マシンの圧縮を防ぎます。

8.2.5. スケジューリングポリシー設定に関する説明

スケジューリングポリシーにより、利用可能なホスト間での仮想マシンの使用状況および分散を指定することができます。スケジューリングポリシーを定義して、クラスター内のホスト全体で自動負荷分散を有効にします。スケジューリングポリシーに関わらず、CPU が過負荷状態のホストでは仮想マシン

が起動しません。デフォルトでは、ホストの CPU が 5 分間 80% 以上の負荷がかかった場合に過負荷と判断されますが、この値はスケジューリングポリシーを使って変更できます。スケジューリングポリシーの詳細情報は、「スケジューリングポリシー」 を参照してください。

表8.7 スケジューリングポリシータブプロパティー

フィールド	説明/アクション
ポリシーの選択	ドロップダウンリストからポリシーを選択します。 • **None**: すでに実行中の仮想マシンのホスト間で負荷分散または電源共有がされないように、ポリシーの値を **none** に設定します。これがデフォルトのモードです。仮想マシンが起動すると、メモリーと CPU 処理の負荷がクラスター内の全ホストに均等に分散されます。ホストが定義された **CpuOverCommitDurationMinutes**、**HighUtilization**、または **MaxFreeMemoryForOverUtilized** に達した場合、ホストにアタッチされた追加の仮想マシンは起動しません。 • **evenly_distributed**: メモリーおよび CPU 処理をクラスター内のすべてのホストで均等に分散します。ホストが定義された **CpuOverCommitDurationMinutes**、**HighUtilization**、または **MaxFreeMemoryForOverUtilized** に達した場合、ホストにアタッチされた追加の仮想マシンは起動しません。 • **cluster_maintenance**: メンテナンスタスク中のクラスターでのアクティビティーの制限。高可用性の仮想マシンを除き、新規の仮想マシンを起動することはできません。ホストの障害が発生した場合、高可用性仮想マシンが正しく再起動し、どの仮想マシンも移行できます。 • **power_saving**: 使用率の低いホストの電力消費を減らすために、利用可能なホストのサブセットにメモリーおよび CPU 処理負荷を分散します。CPU 負荷が低稼働率の値を下回っている状態が定義された時間以上続いたホストは、すべての仮想マシンを他のホストに移行させ、電源を切れるようにします。ホストにアタッチされた追加の仮想マシンは、そのホストが定義された高使用率値に達した場合には起動しません。 • **vm_evenly_distributed**: 仮想マシンの数に基づいて、仮想マシンをホスト間で均等に分散します。**HighVmCount** よりも多くの仮想マシンを実行しているホストがあり、仮想マシン数が **MigrationThreshold** の範囲外であるホストが少なくとも 1 つ存在する場合、クラスターはアンバランスであると判断されます。
プロパティー	以下のプロパティーは選択したポリシーをもとに表示され、必要に応じて編集できます。

フィールド	説明/アクション
	● **HighVmCount**: 負荷分散を有効にするためホストごとに実行する必要のある仮想マシンの最小数を設定します。デフォルト値は、1 台のホストで仮想マシンを実行する 10 です。負荷分散は、少なくとも HighVmCount が仮想マシンを実行するクラスターに 1 つ以上のホストがある場合にのみ有効になります。 ● **MigrationThreshold**: 仮想マシンがホストから移行される前にバッファーを定義します。これは、最も高使用率の低いホストと最も使用率の低いホスト間の仮想マシン数の包含的な差異です。クラスターのすべてのホストが移行しきい値内に留まる仮想マシン数がある場合、クラスターが分散されます。デフォルト値は **5** です。 ● **SpmVmGrace**: SPM ホストで予約される仮想マシンのスロット数を定義します。SPM ホストは他のホストよりも負荷が低くなるため、この変数は SPM ホストが他のホストと比較できる少ない仮想マシン数を定義します。デフォルト値は **5** です。 ● **CpuOverCommitDurationMinutes**: スケジューリングポリシーの実行前に、定義された使用率値以外の CPU 負荷をホストで実行できる時間 (分単位) を設定します。定義した時間間隔は、CPU 負荷のスケジューリングポリシーで一時的な急増から保護し、不要な仮想マシンの移行を軽減します。最大 2 文字デフォルト値は **2** です。 ● **HighUtilization**: パーセンテージで表されます。定義された時間間隔で CPU 使用率以上の CPU 使用率を指定してホストを実行する場合、Red Hat Virtualization Manager はホストの CPU 負荷が最大サービスしきい値を下回るまで、仮想マシンをクラスター内の他のホストに移行します。デフォルト値は **80** です。 ● **LowUtilization**: パーセンテージとして示されています。定義された時間間隔で、ホストが使用率の低い値を下回る場合に、Red Hat Virtualization Manager は仮想マシンをクラスター内の他のホストに移行します。Manager は元のホストマシンの電源をオフにし、負荷分散が必要か、またはクラスターに空きホストが十分にない場合に再び再起動します。デフォルト値は **20** です。 ● **ScaleDown**: ホストのスコアを指定した量で除算して、**HA Reservation** 機能の影響を減らします。これは、**none** を含む、任意のポリシーに追加できる任意のプロパティーです。 ● **HostsInReserve**: 実行中の仮想マシンがない場合でも、実行し続ける多数のホストを指定します。これは、**power_saving** ポリシーに追加できる任意のプロパティーです。

フィールド	説明/アクション
	● **EnableAutomaticHostPowerManagement**: クラスター内のすべてのホストの自動電源管理を有効にします。これは、**power_saving** ポリシーに追加できる任意のプロパティーです。デフォルト値は **true** です。 ● **MaxFreeMemoryForOverUtilized**: 最小サービスレベルに必要な最小空きメモリーを設定します (MB 単位)。Red Hat Virtualization Manager は、ホストの使用可能なメモリーがこの値以下になると、ホストで利用可能なメモリーが最小限のサービスしきい値よりも少ない間に、仮想マシンをクラスター内の他のホストに移行します。**MaxFreeMemoryForOverUtilized** および **MinFreeMemoryForUnderUtilized** の両方を 0 MB に設定すると、メモリーベースのバランシングが無効になります。**MaxFreeMemoryForOverUtilized** が設定されている場合は、予期しない動作を回避するために **MinFreeMemoryForUnderUtilized** も設定する必要があります。これは、**power_saving** ポリシーおよび **evenly_distributed** ポリシーに追加できる任意のプロパティーです。 ● **MinFreeMemoryForUnderUtilized**: ホストが十分に活用されていないと見なされる、最小限必要な空きメモリーを MB 単位で設定します。ホストの利用可能なメモリーがこの値を上回る場合には、Red Hat Virtualization Manager は仮想マシンをクラスター内の他のホストに移行し、ホストマシンの電源を切って、負荷分散の要件が十分であるか、またはクラスターに十分な空きホストがない場合には再起動します。**MaxFreeMemoryForOverUtilized** および **MinFreeMemoryForUnderUtilized** の両方を 0 MB に設定すると、メモリーベースのバランシングが無効になります。**MinFreeMemoryForUnderUtilized** が設定されている場合は、予期しない動作を回避するために **MaxFreeMemoryForOverUtilized** も設定する必要があります。これは、**power_saving** ポリシーおよび **evenly_distributed** ポリシーに追加できる任意のプロパティーです。 ● **HeSparesCount**: 移行するまたはシャットダウンした場合に Manager 用仮想マシンを起動するのに十分な空きメモリーを予約する必要があるセルフホスト型エンジンノードの数を設定します。セルフホスト型エンジンノードでその他の仮想マシンは起動できなくなります。そうでないと、Manager 用仮想マシンには十分な空きメモリーが残ってしまいます。これは、**power_saving**、**vm_evenly_distributed**、**evenly_distributed** ポリシーに追加できる任意のプロパティーです。デフォルト値は **0** です。

フィールド	説明/アクション
スケジューラーの最適化	ホストの重み付け/順序のスケジューリングを最適化します。 ● **Optimize for Utilization**: スケジューリングに重みモジュールを追加し、最適な選択を可能にします。 ● **Optimize for Speed**: 保留中のリクエストの数が 10 個ある場合に、ホストの重みをスキップします。
信頼できるサービスの有効化	OpenAttestation サーバーとのインテグレーションを有効にします。これを有効にする前に、**engine-config** ツールを使用して OpenAttestation サーバーの詳細を入力します。詳細は、「信頼できるコンピュートプール」 を参照してください。
HA 予約の有効化	Manager が高可用性仮想マシンのクラスター容量を監視できるようにします。Manager は、既存のホストが予期せずに失敗した場合に移行するために、高可用性として指定された仮想マシンのクラスター内に適切な容量が存在することを確認します。
Provide custom serial number policy	このチェックボックスを選択すると、クラスター内の仮想マシンのシリアル番号ポリシーを指定できます。以下のオプションのいずれかを選択します。 ● **Host ID**: ホストの UUID を仮想マシンのシリアル番号として設定します。 ● **VM ID**: 仮想マシンの UUID をそのシリアル番号として設定します。 ● **Custom serial number**: カスタムのシリアル番号を指定できます。

ホストの空きメモリーが 20% 未満になると、**mom.Controllers.Balloon - INFO Ballooning guest:half1 from 1096400 to 1991580** が **/var/log/vdsm/mom.log** に記録されます。**/var/log/vdsm/mom.log** は、Memory Overcommit Manager のログファイルです。

8.2.6. クラスターコンソール設定の説明

以下の表は、**New Cluster** および **Edit Cluster** ウィンドウの **Console** タブの設定について説明しています。

表8.8 コンソールの設定

フィールド	説明/アクション

フィールド	説明/アクション
クラスターの SPICE プロキシーの定義	このチェックボックスを選択し、グローバル設定で定義された SPICE プロキシーの上書きを有効にします。この機能は、ユーザー (たとえば、仮想マシンポータル経由で接続する) がハイパーバイザーが存在するネットワーク外にある場合に役に立ちます。
SPICE プロキシーアドレスのオーバーライド	SPICE クライアントが仮想マシンに接続するプロキシーアドレスは以下の形式でなければなりません。 `protocol://[host]:[port]`

8.2.7. フェンシングポリシー設定の説明

以下の表は、**New Cluster** および **Edit Cluster** ウィンドウの **Fencing Policy** タブの設定について説明しています。

表8.9 フェンシングポリシーの設定

フィールド	説明/アクション
フェンシングの有効化	クラスターのフェンシングを有効にします。フェンシングはデフォルトで有効になっていますが、必要に応じて無効にできます。たとえば、一時的なネットワークの問題が発生したり、予想される場合に、管理者は診断またはメンテナンスアクティビティーが完了するまでフェンシングを無効にできます。フェンシングが無効になっている場合は、応答しないホストで実行している高可用性仮想マシンは、別の場所で再起動されないことに注意してください。
ホストのストレージにライブリースがある場合はフェンシングをスキップ	このチェックボックスを選択しないと、Non Responsive のクラスター内のホストは、ストレージに接続されたホストはフェンスされません。
クラスター接続の問題のフェンシングをスキップ	このチェックボックスを選択すると、接続の問題が発生するクラスター内のホストのパーセンテージが、定義された **Threshold** 以上になると、フェンシングが一時的に無効になります。**Threshold** 値はドロップダウンリストから選択されます。利用可能な値は 25、50、75、および 100 です。

フィールド	説明/アクション
gluster ブリックが起動している場合は、フェンシングをスキップ	このオプションは、Red Hat Gluster Storage 機能が有効にされている場合にのみ利用できます。このチェックボックスを選択すると、ブリックが実行中で、他のピアから到達できる場合にフェンシングはスキップされます。2 章を参照してください。フェンシングポリシー を使用して高可用性を設定します。詳細は、**Maintaining Red Hat Hyperconverged Infrastructure** の Appendix A. Fencing Policies for Red Hat Gluster Storage を参照してください。
Gluster クォーラムが満たされない場合は、フェンシングをスキップします。	このオプションは、Red Hat Gluster Storage 機能が有効にされている場合にのみ利用できます。このチェックボックスを選択している場合、ブリックが実行されているとフェンシングがスキップされ、ホストをシャットダウンするとクォーラムが失われます。2 章を参照してください。フェンシングポリシー を使用して高可用性を設定します。詳細は、**Maintaining Red Hat Hyperconverged Infrastructure** の Appendix A. Fencing Policies for Red Hat Gluster Storage を参照してください。

8.2.8. クラスター内のホストの負荷および電源管理ポリシーの設定

evenly_distributed および **power_saving** スケジューリングポリシーを使用すると、許容可能なメモリーおよび CPU 使用率の値と、仮想マシンをホストへ/から移行する必要のあるポイントを指定することができます。**vm_evenly_distributed** スケジューリングポリシーは、仮想マシンの数に基づいて、ホスト間で仮想マシンを均等に配布します。スケジューリングポリシーを定義して、クラスター内のホスト全体で自動負荷分散を有効にします。各スケジューリングポリシーの詳細は、「スケジューリングポリシー設定に関する説明」 を参照してください。

ホストの負荷および電源管理ポリシーの設定

1. **Compute → Clusters** をクリックし、クラスターを選択します。
2. **Edit** をクリックします。
3. **Scheduling Policy** タブをクリックします。
4. 以下のポリシーのいずれかを選択します。
    - **none**
    - **vm_evenly_distributed**
        a. **HighVmCount** フィールドで負荷分散を有効にするために、少なくとも 1 台のホストで実行されている必要がある仮想マシンの最小数を設定します。
        b. 最も高使用率の低いホスト上の仮想マシン数と、**MigrationThreshold** フィールドの最も使用率の低いホストにある仮想マシン数との間の許容可能な最大差を定義します。
        c. **SpmVmGrace** フィールドの SPM ホストで予約される仮想マシンのスロット数を定義します。

d. 任意で、**HeSparesCount** フィールドに、移行またはシャットダウンした場合に Manager 用仮想マシンを起動するのに十分な空きメモリーを確保する、追加のセルフホストエンジンノードの数を入力します。詳細は、「追加ホストでセルフホスト型エンジン用に予約されたメモリースロットの設定」 を参照してください。

- **evenly_distributed**

  a. スケジューリングポリシーで **CpuOverCommitDurationMinutes** フィールドでアクションが実行される前に、定義された使用率値以外の CPU 負荷をホストで実行できる時間 (分単位) を設定します。

  b. **HighUtilization** フィールドの他のホストへの移行を開始する CPU 使用率のパーセンテージを入力します。

  c. **MinFreeMemoryForUnderUtilized** で、仮想マシンが他のホストへの移行を開始するために必要な最小空きメモリーを MB 単位で入力します。

  d. **MaxFreeMemoryForOverUtilized** で、仮想マシンが他のホストへの移行を開始するまでに必要な最大空きメモリーを MB 単位で入力します。

  e. 任意で、**HeSparesCount** フィールドに、移行またはシャットダウンした場合に Manager 用仮想マシンを起動するのに十分な空きメモリーを確保する、追加のセルフホスト型エンジンノードの数を入力します。詳細は、「追加ホストでセルフホスト型エンジン用に予約されたメモリースロットの設定」 を参照してください。

- **power_saving**

  a. スケジューリングポリシーで **CpuOverCommitDurationMinutes** フィールドでアクションが実行される前に、定義された使用率値以外の CPU 負荷をホストで実行できる時間 (分単位) を設定します。

  b. 以下の CPU 使用率の割合を入力します。このパーセンテージは、**LowUtilization** フィールドでホストの使用率が低く 考慮されます。

  c. **HighUtilization** フィールドの他のホストへの移行を開始する CPU 使用率のパーセンテージを入力します。

  d. **MinFreeMemoryForUnderUtilized** で、仮想マシンが他のホストへの移行を開始するために必要な最小空きメモリーを MB 単位で入力します。

  e. **MaxFreeMemoryForOverUtilized** で、仮想マシンが他のホストへの移行を開始するまでに必要な最大空きメモリーを MB 単位で入力します。

  f. 任意で、**HeSparesCount** フィールドに、移行またはシャットダウンした場合に Manager 用仮想マシンを起動するのに十分な空きメモリーを確保する、追加のセルフホスト型エンジンノードの数を入力します。詳細は、「追加ホストでセルフホスト型エンジン用に予約されたメモリースロットの設定」 を参照してください。

5. クラスターの **Scheduler Optimization** として、以下のいずれかを選択します。

   - **Optimize for Utilization**を選択して、スケジューリングに重みモジュールを組み込むため、最適な選択を可能にします。

   - **Optimize for Speed**を選択して、保留中のリクエストが 10 個以上ある場合にホストの重みをスキップします。

6. OpenAttestation サーバーを使用してホストを確認し、**engine-config** ツールを使用してサーバーの詳細を設定している場合は、**Enable Trusted Service** チェックボックスを選択します。

7. 必要に応じて、**Enable HA Reservation** チェックボックスを選択し、Manager が高可用性仮想マシンのクラスター容量を監視できるようにします。

8. オプションで **Provide custom serial number policy**チェックボックスを選択し、クラスター内の仮想マシンのシリアル番号ポリシーを指定し、以下のオプションのいずれかを選択します。

   - **Host ID** を選択して、ホストの UUID を仮想マシンのシリアル番号として設定します。
   - **Vm ID** を選択して、仮想マシンの UUID をシリアル番号として設定します。
   - **Custom serial number** を選択し、テキストフィールドにカスタムのシリアル番号を指定します。

9. **OK** をクリックします。

8.2.9. クラスター内のホストでの MoM ポリシーの更新

Memory Overcommit Manager は、ホストのメモリーバルーンと KSM 機能を処理します。クラスターレベルでのこれらの機能への変更は、次にホストが再起動後またはメンテナーンスモードで **Up** のステータスに移行したときにのみホストに渡されます。ただし、必要な場合は、ホストが **Up** 時に MoM ポリシーを同期することにより、重要な変更をホストをすぐに適用することができます。以下の手順は、各ホストで個別に実行する必要があります。

ホストでの MoM ポリシーの同期

1. **Compute → Clusters** をクリックします。
2. クラスターの名前をクリックして、詳細ビューに移動します。
3. **ホスト** タブをクリックして、更新された MoM ポリシーが必要なホストを選択します。
4. **Sync MoM Policy** をクリックします。

ホストの MoM ポリシーは、ホストをメンテナンスモードに移行し、バック**アップ**を行わずに更新されます。

8.2.10. CPU プロファイルの作成

CPU プロファイルは、クラスター内の仮想マシンが、実行しているホストでアクセスできる最大処理機能を定義します。これは、そのホストで利用可能な合計処理機能のパーセントで表現されます。CPU プロファイルは、データセンターで定義された CPU プロファイルに基づいて作成され、クラスター内のすべての仮想マシンに自動的に適用されません。これらのプロファイルは、プロファイルを有効にするために個々の仮想マシンに手動で割り当てる必要があります。

この手順では、クラスターが属するデータセンター配下に 1 つ以上の CPU 品質のサービスエントリーがすでに定義されていることを前提としています。

CPU プロファイルの作成

1. **Compute → Clusters** をクリックします。
2. クラスターの名前をクリックして、詳細ビューに移動します。
3. **CPU Profiles** タブをクリックします。
4. **New** をクリックします。

5. CPU プロファイルの **Name** および **Description** を入力します。
6. **QoS** 一覧から CPU プロファイルに適用するサービスの品質を選択します。
7. **OK** をクリックします。

8.2.11. CPU プロファイルの削除

Red Hat Virtualization 環境から既存の CPU プロファイルを削除します。

CPU プロファイルの削除

1. **Compute → Clusters** をクリックします。
2. クラスターの名前をクリックして、詳細ビューに移動します。
3. **CPU Profiles** タブをクリックし、削除する CPU プロファイルを選択します。
4. **Remove** をクリックします。
5. **OK** をクリックします。

CPU プロファイルが仮想マシンに割り当てられている場合、それらの仮想マシンには **default** CPU プロファイルが自動的に割り当てられます。

8.2.12. 既存の Red Hat Gluster Storage クラスターのインポート

Red Hat Gluster Storage クラスターおよびクラスターに属するすべてのホストを Red Hat Virtualization Manager にインポートできます。

クラスター内の任意のホストの IP アドレスまたはホスト名やパスワードなどの詳細を指定すると、SSH を介してそのホストで **gluster peer status** コマンドを実行すると、クラスターの一部であるホストの一覧が表示されます。各ホストのフィンガープリントを手動で検証し、パスワードを提供する必要があります。クラスター内のいずれかのホストが停止または到達できない場合、クラスターをインポートすることはできません。新規インポートされたホストに VDSM がインストールされていないため、ブートストラップスクリプトは、インポート後にホストに必要な VDSM パッケージをすべてインストールして再起動します。

既存の Red Hat Gluster Storage クラスターの Red Hat Virtualization Manager へのインポート

1. **Compute → Clusters** をクリックします。
2. **New** をクリックします。
3. クラスターが属する **データセンター** を選択します。
4. クラスターの **Name** および **Description** を入力します。
5. **Enable Gluster Service**チェックボックスを選択し、**Import existing gluster configuration** チェックボックスを選択します。
   Import existing gluster configuration フィールドは、**Enable Gluster Service**が選択されている場合にのみ表示されます。
6. **Hostname** フィールドには、クラスター内のサーバーのホスト名または IP アドレスを入力します。

ホストの **SSH フィンガープリント** が表示され、正しいホストに接続していることを確認します。ホストが到達不能な場合や、ネットワークエラーが発生した場合には、**Error in fetching fingerprint** のエラーが **Fingerprint** フィールドに表示されます。

7. サーバーの **Password** を入力し、**OK** をクリックします。
8. **ホストの追加** 画面が開き、クラスターの一部であるホストの一覧が表示されます。
9. 各ホストに **名前** と **root パスワード**を入力します。
10. すべてのホストに同じパスワードを使用する場合は、**Use a Common Password** チェックボックスを選択して、指定したテキストフィールドにパスワードを入力します。
Apply をクリックして、入力したパスワードをすべて設定します。

フィンガープリントが有効であることを確認し、**OK** をクリックして変更を送信します。

ブートストラップスクリプトは、インポート後にホストに必要な VDSM パッケージをすべてインストールして再起動します。既存の Red Hat Gluster Storage クラスターを Red Hat Virtualization Manager に正常にインポートできるようになりました。

8.2.13. ホストウィンドウの追加設定の説明

Add Hosts ウィンドウで、Gluster 対応クラスターの一部としてインポートされたホストの詳細を指定できます。このウィンドウは、**New Cluster** ウィンドウで **Enable Gluster Service**チェックボックスを選択し、必要なホストの詳細を提供した後に表示されます。

表8.10 Gluster ホスト設定の追加

フィールド	説明
一般的なパスワードを使用します。	このチェックボックスにチェックマークを入れ、クラスターに属する全ホストに同じパスワードを使用します。**パスワード** フィールドにパスワードを入力し、**適用** ボタンをクリックして全ホストにパスワードを設定します。
Name	ホストの名前を入力します。
ホスト名/IP	このフィールドには、**新規クラスター**ウィンドウで指定したホストの完全修飾ドメイン名または IP が自動的に設定されます。
root パスワード	各ホストに異なる root パスワードを使用するには、このフィールドにパスワードを入力します。このフィールドは、クラスター内のすべてのホストに提供される共通のパスワードを上書きします。
Fingerprint	ホストのフィンガープリントが表示され、正しいホストに接続していることを確認します。このフィールドには、**新規クラスター**ウィンドウで指定したホストのフィンガープリントが自動的に入力されます。

8.2.14. クラスターの削除

クラスターを削除する前に、すべてのホストをクラスターから移動します。

注記

デフォルト クラスターは **Blank** テンプレートを保持するため、削除できません。ただし、**Default** クラスターの名前を変更し、新しいデータセンターに追加することは可能です。

クラスターの削除

1. **Compute → Clusters** をクリックし、クラスターを選択します。
2. クラスターにホストがないことを確認します。
3. **Remove** をクリックします。
4. **OK** をクリックします。

8.2.15. メモリーの最適化

ホストの仮想マシン数を増やすには、仮想マシンに割り当てる **メモリーのオーバーコミット** を使用できます。メモリーは RAM を超え、swap 領域に依存します。

ただし、メモリーのオーバーコミットには潜在的な問題があります。

- スワップパフォーマンス - スワップ領域が遅くなり、仮想マシンのパフォーマンスに影響する RAM よりも多くの CPU リソースを消費します。スワップを過剰にすると、CPU のスラッシングにつながる可能性があります。
- OOM (Out-of-memory) killer: ホストがスワップ領域が不足すると、新規プロセスを開始できず、カーネルの OOM killer デーモンは仮想マシンゲストなどのアクティブなプロセスのシャットダウンを開始します。

これらの欠点に対処するために、以下を実行できます。

- **Memory Optimization** 設定および **Memory Overcommit Manager (MoM)** を使用してメモリーのオーバーコミットを制限します。
- 仮想メモリーの潜在的な要求に対応できるサイズに大きな swap 領域を作成し、安全マージンを残します。
- **memory ballooning** および **Kernel Same-page Merging (KSM)** を有効にして、仮想メモリーサイズを縮小します。

8.2.15.1. メモリーの最適化とメモリーオーバーコミット

Memory Optimization 設定 (**None** (0%)、**150%**、または **200%** のいずれかを選択して、メモリーのオーバーコミット量を制限できます。

各設定は、メモリーの割合を表します。たとえば、RAM が 64 GB のホストの場合、**150%** を選択すると、仮想メモリーの合計 96 GB について、メモリーを追加の 32 GB でオーバーコミットできます。ホストの合計が 4 GB を使用する場合には、残りの 92 GB が利用可能になります。仮想マシンにはほとんど (**System** タブの **Memory Size**) を割り当てることができますが、その一部を安全マージンとして割り当てることを検討してください。

仮想メモリーの要求で急増すると、MoM、メモリーバルーン、および KSM が仮想メモリーを再最適化するまでのパフォーマンスに影響する可能性があります。この影響を減らすには、実行するアプリケーションおよびワークロードの種類に適した制限を選択します。

- メモリーに対する増分増加をデマンドするワークロードの場合は、200% または 150% などの高いパーセンテージを選択します。
- より重大なアプリケーションまたはワークロードでメモリーの需要が増加する場合には 、150% や **None** (0%) などの低いパーセンテージを選択します。**None** を選択するとメモリーのオーバーコミットを防ぐことができますが、MoM、メモリーバルーンデバイス、および KSM は仮想メモリーの最適化を継続できます。

重要

設定を実稼働環境にデプロイする前に、さまざまな条件下で **メモリー最適化** の設定を必ずテストしてください。

Memory Optimization 設定を設定するには、**New Cluster** または **Edit Cluster** ウィンドウの **Optimization** タブをクリックします。「最適化設定の説明」 を参照してください。

その他のコメント:

- ホスト統計ビュー には、オーバーコミットメント率のサイズを決定するための有用な履歴情報が表示されます。
- KSM とメモリーバルーンの変更が達成するメモリーの最適化のサイズは継続的に行われるため、実際に利用可能なメモリーをリアルタイムで決定することはできません。
- 仮想マシンが仮想メモリー制限に達すると、新しいアプリを開始できません。
- ホストで実行する仮想マシンの数を計画する際には、最大仮想メモリー (物理メモリーサイズおよびメモリー **Memory Optimization** 設定) を開始点として使用します。メモリーバルーンや KSM などのメモリーの最適化により実現される、より小さい仮想メモリーに考慮しないでください。

8.2.15.2. swap 要領とメモリーオーバーコミットメント

Red Hat は、スワップ領域を設定するためにこの推奨事項 を提供しています。

これらの推奨事項を適用する場合は、最も悪いシナリオにおいて、スワップ領域のサイズを最後の作業メモリーとするガイダンスに従ってください。物理メモリーのサイズと **Memory Optimization** 設定を、仮想メモリーサイズの合計見積もるベースとして使用します。MoM、メモリーバルーン、および KSM による仮想メモリーサイズの削減を除外してください。

重要

OOM 条件を防ぐのに役立つため、最も悪いケースのシナリオを処理するのに十分な swap 領域を確保し、安全マージンを引き続き利用可能にします。実稼働環境にデプロイする前に、さまざまな条件で設定を常にテストしてください。

8.2.15.3. Memory Overcommit Manager(MoM)

Memory Overcommit Manager(MoM) は、以下の 2 つを行います。

- これは、前述のセクションで説明されているように、クラスターのホストにメモリー **Memory Optimization** 設定を適用してメモリーのオーバーコミットを制限します。
- 以下のセクションで説明されているように、**memory ballooning** と **KSM** を管理することで、メモリーを最適化します。

MoM を有効または無効にする必要はありません。

ホストの空きメモリーが 20% 未満になると、**mom.Controllers.Balloon - INFO Ballooning guest:half1 from 1096400 to 1991580** が Memory Overcommit Manager ログファイル**/var/log/vdsm/mom.log** に記録されます。

8.2.15.4. メモリーバルーニング

仮想マシンは、割り当てられた仮想メモリーの量で起動します。仮想メモリー使用量が RAM を超えると、ホストはスワップ領域に依存します。有効にすると、仮想マシンが **memory ballooning** の未使用の部分を上回ります。解放されたメモリーは、ホスト上の他のプロセスおよび仮想マシンで再利用できます。メモリーフットプリントが削減されると、スワップの可能性が低くなり、パフォーマンスが向上します。

読み込み可能なカーネルモジュール (LKM) として同梱されるメモリーバルーンデバイスとドライバーを提供する **virtio-balloon** パッケージ。デフォルトでは、自動的に読み込まれるように設定されています。モジュールをブラックリストに登録したり、バルーンスを無効にします。

メモリーバルーンデバイスは、相互に直接調整しません。各仮想マシンのニーズを継続的に監視するには、MoM(ホストの Memory Overcommit Manager) プロセスに依存して、各仮想マシンのニーズを継続的に監視し、仮想メモリーの増減を指示します。

パフォーマンスに関する考慮事項:

- Red Hat は、パフォーマンスが高く、低レイテンシーを必要とするワークロードにはメモリーバルーンおよびオーバーコミットを推奨しません。高パフォーマンスの仮想マシンテンプレートおよびプールの設定 を参照してください。
- Red Hat は、パフォーマンスよりも仮想マシンの密度 (経済性) を増やすことが、重要な場合は、メモリーバルーンを推奨します。
- メモリーバルーニングは CPU 使用率に大きな影響を及ぼしません。(KSM は一部の CPU リソースを消費しますが、消費は不足の下に保たれます。)

メモリーバルーンを有効にするには、**New Cluster** または**Edit Cluster** ウィンドウの **Optimization** タブをクリックします。次に、**Enable Memory Balloon Optimization**チェックボックスを選択します。この設定により、このクラスターのホストで実行されている仮想マシンでメモリーのオーバーコミットが有効になります。このチェックボックスを選択すると、MoM はバルーニングを開始し、可能な場合はすべての仮想マシンのメモリーサイズが保証されます。「最適化設定の説明」 を参照してください。

このクラスター内の各ホストは、ステータスが Up に変わったときにバルーンポリシーの更新を受け取ります。必要に応じて、ステータスを変更せずに、ホストのバルーンポリシーを手動で更新できます。「クラスター内のホストでの MoM ポリシーの更新」 を参照してください。

8.2.15.5. Kernel Same-page Merging (KSM)

仮想マシンの実行時には、一般的なライブラリーや使用頻度の高いデータといったアイテム向けに、重複したメモリーページが作成されることがあります。さらに、同じようなゲスト OS やアプリケーションを実行している仮想マシンでは、仮想メモリー内のメモリーページが重複してしまいます。

KSM (**Kernel Sam-page Merging**) を有効にすると、ホスト上の仮想メモリーを調査し、重複するメモ

リーページを排除して、残りのメモリーページを複数のアプリケーションや仮想マシンで共有できます。これらの共有メモリーページにはコピーオンライトのマークがついており、仮想マシンでページに変更を書き込む必要がある場合には、先にコピーを作成してからそのコピーに変更を書き込みます。

KSM が有効な間は、MoM が KSM を管理します。KSM を手動で設定制御する必要はありません。

KSM は、2 つの方法で仮想メモリーのパフォーマンスを向上させます。共有メモリーのページは使用頻度が高いため、ホストはそのページをキャッシュやメインメモリーに格納する可能性が高くなり、メモリーアクセス速度が向上します。さらに、メモリーオーバーコミットを使用することで、KSM は仮想メモリーのフットプリントを減らし、スワッピング発生の可能性を軽減してパフォーマンスを向上させます。

KSM はメモリーバルーニングよりも多くの CPU リソースを消費します。KSM の CPU 消費量は、負荷をかけても変わりません。同一の仮想マシンやアプリケーションをホスト上で実行すると、KSM は異種の仮想マシンを実行するよりもメモリーページをマージする機会が多くなります。異なる仮想マシンやアプリケーションを多く実行している場合には、KSM を使用するための CPU コストがその利点を打ち消してしまう可能性があります。

パフォーマンスに関する考慮事項:

- KSM デーモンが大量のメモリーをマージした後に、カーネルのメモリーが計算した統計値が最終的に矛盾することがあります。システムに大量の空きメモリーがある場合には、KSM を無効にするとパフォーマンスが向上することがあります。
- Red Hat は、パフォーマンスが高く、低レイテンシーを必要とするワークロードには KSM およびオーバーコミットを推奨しません。高パフォーマンスの仮想マシンテンプレートおよびプールの設定 を参照してください。
- Red Hat は、パフォーマンスよりも仮想マシンの密度 (経済性) を増やすことが、重要な場合は、KSM を推奨します。

KSM を有効にするには、**New Cluster** または**Edit Cluster** ウィンドウの **Optimization** タブをクリックします。次に、**Enable KSM**のチェックボックスを選択します。この設定をし用すると、MoM は必要なときに KSM を実行でき、CPU コストを上回るメモリー節約のメリットが得られます。「最適化設定の説明」を参照してください。

8.2.16. クラスターの互換バージョンの変更

Red Hat Virtualization のクラスターには互換バージョンがあります。クラスターの互換バージョンは、そのクラスター内の全ホストがサポートする Red Hat Virtualization の機能を示します。クラスターの互換バージョンは、そのクラスター内で最も機能性の低いホストオペレーティングシステムのバージョンに応じて設定されます。

重要

クラスターの互換バージョンを変更するには、まず、クラスター内の全ホストを更新して、必要な互換性レベルをサポートするレベルにする必要があります。更新が利用可能であることを示すアイコンがホストの横にあるかどうかを確認します。

手順

1. 管理ポータルで、**コンピュート → クラスター** をクリックします。
2. 変更を行うクラスターを選択し、**編集** をクリックします。
3. **全般** タブで **互換バージョン** を必要な値に変更します。

4. **OK** をクリックします。**クラスターの互換バージョンを変更** の確認ダイアログが開きます。

5. **OK** をクリックして確定します。

重要

一部の仮想マシンおよびテンプレートが不適切に設定されていることを警告するエラーメッセージが表示される場合があります。このエラーを修正するには、それぞれの仮想マシンを手動で編集します。**仮想マシンの編集** ウィンドウには、修正すべき項目を確認することのできる新たな検証および警告が表示されます。問題が自動的に修正され、仮想マシンの設定を再度保存するだけで十分な場合もあります。それぞれの仮想マシンを編集したら、クラスターの互換バージョンを変更することができます。

クラスターの互換バージョンを更新したら、実行中またはサスペンド中のすべての仮想マシンについてクラスターの互換バージョンを更新する必要があります。そのためには、ゲストオペレーティングシステム内からではなく、管理ポータルから、または REST API を使用して仮想マシンを再起動します。再起動が必要な仮想マシンには、変更が保留されていることを示すアイコン () が付きます。プレビュー状態にある仮想マシンスナップショットについては、クラスターの互換バージョンを変更することができません。まずコミットするか、プレビューを取り消す必要があります。

セルフホスト型エンジン環境では、Manager 仮想マシンを再起動する必要はありません。

別途適切な時期に仮想マシンを再起動することもできますが、仮想マシンで最新の設定が使用されるように、直ちに再起動することを強く推奨します。更新されていない仮想マシンは古い設定で実行され、再起動前に仮想マシンに他の変更を加えた場合には新しい設定が上書きされてしまう可能性があります。

データセンター内のすべてのクラスターと仮想マシンの互換性バージョンを更新したら、データセンター自体の互換性バージョンを変更できます。

第9章 論理ネットワーク

9.1. 論理ネットワークタスク

9.1.1. ネットワークタスクの実行

Network → Networks は、ユーザーが論理ネットワーク関連の操作を実行し、各ネットワークのプロパティーや他のリソースとの関連付けに基づいて論理ネットワークを検索するための中心的な場所を提供します。**New**、**Edit**、**Remove**ボタンで、データセンター内の論理ネットワークの作成、プロパティーの変更、削除ができます。

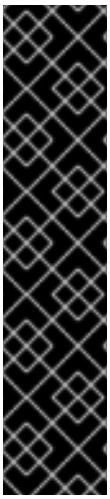
各ネットワーク名をクリックし、詳細表示のタブを使って以下の機能を実行します。

- クラスターやホストにネットワークを割り当てまたは割り当て解除する
- 仮想マシンやテンプレートからネットワークインターフェイスを削除する
- ネットワークへのアクセスや管理を行うユーザーの権限を追加、削除する

これらの機能は、それぞれのリソースからもアクセス可能です。

警告

データセンターやクラスターでは、ホストが動作している場合にはネットワークを変更しないでください。ホストに到達できなくなる危険性があります。



重要

Red Hat Virtualization ノードを使用してサービスを提供する予定の場合には、Red Hat Virtualization 環境が動作を停止すると、そのサービスが停止することに注意してください。

これはすべてのサービスに当てはまりますが、特に Red Hat Virtualization 上で以下を実行した場合の危険性に注意する必要があります。

- ディレクトリーサービス
- DNS
- ストレージ

9.1.2. 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

論理ネットワークを作成し、データセンターやデータセンター内のクラスターでの使用を定義します。

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

1. **Compute → Data Centers** または **Compute → Clusters** をクリックします。
2. データセンターまたはクラスター名をクリックして、詳細ビューを開きます。

3. Logical Networks タブをクリックします。
4. New Logical Network ウィンドウを開きます。
   - データセンターの詳細表示から、Newをクリックします。
   - クラスターの詳細表示で、Add Networkをクリックします。
5. 論理ネットワークの Name、Description、および Comment を入力します。
6. 必要に応じて、Enable VLAN tagging を有効にします。
7. 必要に応じて、VM Network を無効にします。
8. オプションで、Create on external provider チェックボックスをオンにします。これにより、Network Label、VM Network、および MTU オプションが無効になります。詳細は、14章*外部プロバイダー* を参照してください。
9. External Providerを選択します。**外部プロバイダー** のリストには、**読み取り** 専用モードの外部プロバイダーは含まれません。
   External Provider 一覧で ovirt-provider-ovn を選択し、Connect to physical network の選択は解除したままにすることで、内部分離ネットワークを作成できます。
10. Network Label テキストフィールドに、論理ネットワークの新しいラベルを入力するか、既存のラベルを選択します。
11. MTU 値を Default(1500) または Custom に設定します。
12. **外部プロバイダー**ドロップダウンリストからovirt-provider-ovnを選択した場合は、ネットワークに**セキュリティーグループ**を実装するかどうかを定義します。詳細は、「論理ネットワーク一般設定の説明」 を参照してください。
13. **クラスター**タブから、ネットワークを割り当てるクラスターを選択します。また、論理ネットワークを必須ネットワークにするかどうかも指定できます。
14. Create on external provider を選択すると、Subnet タブが表示されます。Subnetタブからを選択し、Name、CIDR、Gatewayアドレスを入力し、論理ネットワークが割り当てるサブネットのIP Versionを選択します。また、必要に応じて DNS サーバーを追加することもできます。
15. v NIC Profilesタブで、必要に応じて vNIC プロファイルを論理ネットワークに追加します。
16. OK をクリックします。

論理ネットワークにラベルを入力した場合は、そのラベルが割り当てられたすべてのホストネットワークインターフェイスに自動的に追加されます。

注記

新しい論理ネットワークを作成したり、ディスプレイネットワークとして使用されている既存の論理ネットワークに変更を加えたりする場合には、そのネットワークを使用する、稼働中の仮想マシンは、ネットワークが使用可能になる前、または変更が適用される前に、再起動する必要があります。

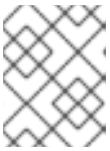
9.1.3. 論理ネットワークの編集

重要

ホスト上のネットワーク設定と同期していない場合には、論理ネットワークの編集や他のインターフェイスへの移動はできません。ネットワークの同期方法については、「ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て」 を参照してください。

論理ネットワークの編集

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **Logical Networks**タブをクリックして、論理ネットワークを選択します。
4. **Edit** をクリックします。
5. 必要な設定を編集します。



注記

デフォルトのネットワーク除き、新規または既存のネットワークの名前は、仮想マシンを停止することなく編集することができます。

6. **OK** をクリックします。

注記

マルチホストネットワーク設定では、更新されたネットワーク設定を、ネットワークが割り当てられたデータセンター内のすべてのホストに自動適用します。変更は、ネットワークを使用する仮想マシンが停止しているときにのみ適用されます。すでにホストに設定されている論理ネットワークの名前は変更できません。**VM ネットワーク**のオプションは、ネットワークを使用している仮想マシンやテンプレートを実行している間は無効にできません。

9.1.4. 論理ネットワークの削除

Network → Networks または **Compute → Data Centers** から論理ネットワークを削除できます。以下の手順では、データセンターに関連付けられた論理ネットワークを削除する方法を説明します。Red Hat Virtualization の環境では、少なくとも 1 つの論理ネットワークを**ovirtmgmt**管理ネットワークとして使用する必要があります。

論理ネットワークの削除

1. **Compute → Data Centers** をクリックします。
2. データセンターの名前をクリックして、詳細ビューを開きます。
3. **Logical Networks**タブをクリックすると、データセンター内の論理ネットワークがリストアップされます。
4. 論理ネットワークを選択し、**削除**をクリックします。
5. オプションで、**Remove external network (s) from the provider (s) as well**チェックボックスを選択すると、ネットワークが外部のプロバイダーによって提供されている場合、Manager と

外部のプロバイダーの両方から論理ネットワークを削除してください。外部プロバイダーが読み取り専用モードの場合、チェックボックスはグレーアウトされます。

6. **OK** をクリックします。

論理ネットワークが Manager から削除され、利用できなくなります。

9.1.5. 非管理者用論理ネットワークのデフォルトルートとしての設定

クラスター内のホストが使用するデフォルトのルートは、管理ネットワーク **(ovirtmgmt**) を経由します。以下の手順では、非管理者用の論理ネットワークをデフォルトルートとして設定する手順を説明します。

前提条件:

- **default_route** カスタムプロパティーを使用している場合は、接続しているすべてのホストからカスタムプロパティーの設定を解除してから、この手順を実行する必要があります。

デフォルトルートロールの設定

1. **Network → Networks** をクリックします。
2. デフォルトルートとして設定する非管理用論理ネットワークの名前をクリックすると、その詳細が表示されます。
3. **Clusters** タブをクリックします。
4. **Manage Network** をクリックして、**Manage Network** 画面を開きます。
5. 該当するクラスターの**Default Route**チェックボックスを選択します。
6. **OK** をクリックします。

ホストにネットワークが接続されている場合には、ホストのデフォルトルートは選択したネットワークに設定されます。クラスターにホストを追加する前に、デフォルトルートのロールを設定することをお勧めします。クラスターにすでにホストが含まれている場合には、変更内容をホストに同期するまで、ホストが同期しなくなる可能性があります。

IPv6 の重要な制限事項

- IPv6 については、Red Hat Virtualization でサポートされるのは静的なアドレスだけです。
- 両方のネットワークが単一のゲートウェイを共有している (同じサブネット上にある) 場合には、デフォルトルートのロールを管理ネットワーク (ovirtmgmt) から別の論理ネットワークに移動できます。
- ホストと Manager が同じサブネットにない場合には、IPv6 ゲートウェイが削除されているため、Manager はホストとの接続を失います。
- デフォルトルートのロールを非管理ネットワークに移動すると、ネットワークインターフェイスから IPv6 ゲートウェイが削除され、On cluster**clustername**the 'Default Route Role' network is no longer network ovirtmgmt という警告が表示されます。IPv6 ゲートウェイはこのネットワークから削除されています。

9.1.6. 論理ネットワークのゲートウェイの表示と編集

論理ネットワークのゲートウェイ、IP アドレス、サブネットマスクを定義できます。これは、ホスト上に複数のネットワークが存在し、トラフィックがデフォルトゲートウェイではなく、指定したネットワークを経由しなければならない場合に必要です。

ホストに複数のネットワークが存在し、ゲートウェイが定義されていない場合には、リターントラフィックはデフォルトゲートウェイを経由することになり、意図した宛先に到達しない可能性があります。これにより、ユーザーがホストに対して ping を実行できなくなります。

Red Hat Virtualization は、インターフェイスがアップダウンするたびに、複数のゲートウェイを自動的に処理します。

論理ネットワークのゲートウェイの表示と編集

1. **Compute → Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **ネットワークインターフェイス**タブをクリックすると、ホストに接続されているネットワークインターフェイスとその設定内容が一覧表示されます。
4. **Setup Host Networks** をクリックします。
5. 割り当てられた論理ネットワークにカーソルを合わせ、鉛筆アイコンをクリックして **Edit Management Network** ウィンドウを開きます。

Edit Management Network ウィンドウには、ネットワーク名、ブートプロトコル、IP、サブネットマスク、ゲートウェイの各アドレスが表示されます。アドレス情報は、**静的** ブートプロトコルを選択して手動で編集できます。

9.1.7. 論理ネットワーク一般設定の説明

以下の表では、**New Logical Network**および**Edit Logical Network**ウィンドウの**General**タブの設定について説明しています。

表9.1新しい論理ネットワーク および論理ネットワークの編集の設定

フィールド名	説明
Name	論理ネットワークの名前。このテキストフィールドには、大文字、小文字、数字、ハイフン、およびアンダースコアの組み合わせが含まれる一意の名前である必要があります。 論理ネットワークの名前は 16 文字以上、ASCII 以外の文字を含めることができますが、ホスト上の識別子**(vdsm_name)**は定義した名前とは異なりますのでご注意ください。これらの名前のマッピングを表示する手順については、Mapping VDSM Names to Logical Network Names を参照してください。
説明	論理ネットワークの説明。このテキストフィールドには 40 文字の制限があります。

フィールド名	説明
Comment	論理ネットワークに関するプレーンテキストの人間が読めるコメントを追加するためのフィールド。
外部プロバイダーでの作成	外部プロバイダーとして Manager に追加された OpenStack Networking インスタンスへの論理ネットワークを作成できます。 **External Provider**- 論理ネットワークを作成するための外部プロバイダーを選択できます。
VLAN タグの有効化	VLAN タグは、論理ネットワークで伝送されるすべてのネットワークトラフィックに特別な特性を与えるセキュリティー機能です。VLAN タグ付きのトラフィックは、その特性のないインターフェイスでは読み取れません。また、論理ネットワークに VLAN を使用すると、1 つのネットワークインターフェイスに、VLAN タグが異なる論理ネットワークを複数、関連付けることができます。VLAN タグ付けが有効な場合は、テキストエントリーフィールドに数値を入力します。
VM ネットワーク	仮想マシンのみがこのネットワークを使用する場合は、このオプションを選択します。ストレージの通信など、仮想マシンを介さないトラフィックにネットワークを使用する場合は、このチェックボックスを選択しないでください。
MTU	最大伝送単位 (MTU) を括弧 () で指定された値に設定する**デフォルト**、または論理ネットワークの**カスタム**MTU を設定するカスタムのいずれかを選択します。これを利用して、新しい論理ネットワークがサポートする MTU を、そのネットワークがインターフェイスするハードウェアがサポートする MTU に合わせることができます。**カスタム**を選択した場合は、テキスト入力フィールドに数値を入力します。
ネットワークラベル	ネットワークの新しいラベルを指定したり、ホストネットワークインターフェイスに既に取り付けられている既存のラベルを選択したりすることができます。既存のラベルを選択した場合には、そのラベルが指定されたすべてのホストネットワークインターフェイスに論理ネットワークが自動的に割り当てられます。

フィールド名	説明
セキュリティーグループ	この論理ネットワーク上のポートにセキュリティーグループを割り当てることができます。**Disabled** は、セキュリティーグループの機能を無効にします。**Enabled** は、この機能を有効にします。ポートを作成してこのネットワークに接続すると、ポートセキュリティーが有効な状態で定義されます。つまり、仮想マシンに対するアクセスには、現在プロビジョニングされているセキュリティーグループが適用されることになります。**Inherit from Configuration** では、すべてのネットワークで定義されている設定ファイルの動作をポートに継承させます。デフォルトでは、このファイルはセキュリティーグループを無効にします。詳細は、「論理ネットワークとポートへのセキュリティーグループの割り当て」 を参照してください。

9.1.8. 論理ネットワーククラスターの設定の説明

以下の表は、New Logical Network ウィンドウの Cluster タブの設定について説明しています。

表9.2 新しい論理ネットワーク 設定

フィールド名	説明
クラスターへのネットワークの接続/クラスターからのネットワークの切断	論理ネットワークをデータセンター内のクラスターにアタッチまたはデタッチでき、論理ネットワークを個々のクラスターに必要なネットワークとするかどうかを指定することができます。 Name- 設定が適用されるクラスターの名前。この値は編集できません Attach All- データセンター内のすべてのクラスターとの間で、論理ネットワークをアタッチまたはデタッチできます。また、各クラスターの名前の横にあるAttachチェックボックスを選択または選択解除して、論理ネットワークを特定のクラスターに接続したり、クラスターから分離したりすることもできます。 Required All- 論理ネットワークがすべてのクラスターで必須のネットワークであるかどうかを指定できます。また、各クラスターの名前の横にあるRequiredチェックボックスを選択または選択解除して、論理ネットワークが特定のクラスターに必要なネットワークであるかどうかを指定することもできます。

9.1.9. 論理ネットワークの vNIC プロファイル設定の説明

以下の表は、New Logical Network ウィンドウの vNIC Profiles タブの設定について説明しています。

表9.3 新しい論理ネットワーク 設定

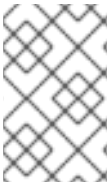
フィールド名	説明
vNIC Profiles	論理ネットワークの1つまたは複数の vNIC プロファイルを指定できます。vNIC プロファイルの横にあるプラスボタンまたはマイナスボタンをクリックして、vNIC プロファイルを論理ネットワークに追加したり、論理ネットワークから削除したりすることができます。最初のフィールドでは、vNIC プロファイルの名前を入力します。 **Public**- プロファイルをすべてのユーザーが利用できるようにするかどうかを指定できます。 **QoS**- vNIC プロファイルにネットワークの QoS(Quality of Service) プロファイルを指定できます。

9.1.10. ネットワークの管理ウィンドウでの論理ネットワークに対する特定のトラフィックタイプの指定

ネットワークのトラフィックフローを最適化するために、論理ネットワークのトラフィックタイプを指定します。

論理ネットワークのトラフィックタイプの指定

1. **Compute** → **Clusters** をクリックします。
2. クラスターの名前をクリックして、詳細ビューに移動します。
3. **Logical Networks** タブをクリックします。
4. **Manage Networks**をクリックします。
5. 適切なチェックボックスやラジオボタンを選択してください。
6. **OK** をクリックします。



注記

外部のプロバイダーが提供する論理ネットワークは、仮想マシンのネットワークとして使用する必要があり、表示や移行などの特別なクラスターのロールを割り当てることはできません。

9.1.11. ネットワーク管理画面での設定内容の説明

以下の表では、**Manage Networks**ウィンドウの設定について説明しています。

表9.4 ネットワーク設定の管理

フィールド	説明/アクション
Assign	クラスター内の全ホストへの論理ネットワークの割り当てます。
必須。	Required (必須) と表示されたネットワークは、そのネットワークに関連するホストを正しく機能させるには、常に稼働している必要があります。必要なネットワークが機能しなくなると、そのネットワークに関連するホストはすべて動作しなくなります。
VM ネットワーク	VM ネットワークとマークされている論理ネットワークは、仮想マシンのネットワークに関連するネットワークトラフィックを伝送します。
ディスプレイネットワーク	ディスプレイネットワークとマークされた論理ネットワークは、SPICE と仮想ネットワークコントローラーに関連するネットワークトラフィックを伝送します。
移行ネットワーク	Migration Network とマークされた論理ネットワークは、仮想マシンとストレージの移行トラフィックを伝送します。このネットワークに障害が発生した場合には、代わりに管理ネットワーク**ovirtmgmt** (デフォルト) が使用されます。

9.1.12. NIC の仮想機能設定の編集

注記

これは、Red Hat Virtualization で SR-IOV を準備および設定する方法を示す一連のトピックの 1 つです。詳細は、Setting Up and Configuring SR-IOV を参照してください。

Single Root I/O Virtualization (SR-IOV) では、単一の PCIe エンドポイントを複数の個別デバイスとして使用できるようになります。これは、物理機能 (PF) と仮想機能 (VF) の 2 つの PCIe 機能を導入することで実現されます。PCIe カードには 1~8 個の PF を含めることができますが、PF ごとにさらに多くの VF をサポートできます (デバイスによって異なります)。

各 NIC の VF の数を含め、Red Hat Virtualization Manager を介して SR-IOV 対応のネットワークインターフェイスコントローラー (NIC) の設定を編集し、VF へのアクセスを許可する仮想ネットワークを指定できます。

VF が作成されると、それぞれをスタンドアロンの NIC として扱うことができます。これには、1 つ以上の論理ネットワークを割り当てること、論理ネットワークとの結合インターフェイスを作成すること、および直接デバイスパススルーのために vNIC を論理ネットワークに直接割り当てることが含まれます。

VF に直接接続するには、vNIC プロパティーを有効にする必要があります。「vNIC プロファイルでのパススルーの有効化」を参照してください。

NIC の仮想機能設定の編集

1. **Compute → Hosts** をクリックします。
2. SR-IOV 対応ホストの名前をクリックし、詳細ビューを開きます。
3. **Network Interfaces** タブをクリックします。
4. **Setup Host Networks** をクリックします。
5. SR-IOV 対応の NIC ( マーク) を選択し、鉛筆アイコンをクリックします。
6. 仮想機能の数を編集するには、**Number of VFs setting**ドロップダウンボタンをクリックし、**Number of VFs**テキストフィールドを編集します。

重要

VF の数を変更すると、新しい VF を作成する前に、ネットワークインターフェイス上の以前の VF がすべて削除されます。これには、仮想マシンが直接接続されている VF が含まれます。

7. **All Networks** チェックボックスがデフォルトで選択されており、全ネットワークが Virtual Function にアクセスできるようにします。Virtual Function へのアクセスを許可する仮想ネットワークを指定するには、**Specific networks** ラジオボタンを選択してすべてのネットワークを一覧表示します。次に、任意のネットワークのチェックボックスを選択するか、**Labels** テキストフィールドを使用して 1 つ以上のネットワークラベルに基づいてネットワークを自動的に選択できます。
8. **OK** をクリックします。
9. **Setup Host Networks** ウィンドウで**OK**をクリックします。

9.2. 仮想ネットワークインターフェイスカード

9.2.1. vNIC プロファイルの概要

バーチャルネットワークインターフェイスカード (vNIC) のプロファイルは、Manager 内の個々のバーチャルネットワークインターフェイスカードに適用できる設定の集まりです。vNIC プロファイルでは、ネットワーク QoS プロファイルの vNIC への適用、ポートミラーリングの有効化/無効化、カスタムプロパティーの追加/削除が可能です。また、vNIC プロファイルは、特定のユーザーに使用 (消費) のパーミッションを与えることができるという点で、管理上の柔軟性において、追加の切り口が提供されています。このようにして、異なるユーザーが特定のネットワークから受けるサービスの質を制御できます。

9.2.2. vNIC プロファイルの作成と編集

Virtual Network Interface Controller (vNIC) のプロファイルを作成または編集して、ユーザーやグループのネットワーク帯域幅を調整できます。

注記

ポートミラーリングを有効または無効にする場合には、編集する前に、関連するプロファイルを使用しているすべての仮想マシンがダウン状態になっている必要があります。

vNIC プロファイルの作成と編集

1. **Network** → **Networks** をクリックします。
2. 論理ネットワーク名をクリックして、詳細ビューを開きます。
3. **vNIC Profiles** タブをクリックします。
4. **新規作成**または**編集**をクリックします。
5. プロファイルの **Name** および **Description** を入力します。
6. **QoS**リストから該当する Quality of Service ポリシーを選択します。
7. ドロップダウンリストから**ネットワークフィルター**を選択して、仮想マシンとの間のネットワークパケットのトラフィックを管理します。ネットワークフィルターの詳細は、**Red Hat Enterprise Linux Virtualization Deployment and Administration Guide** の Applying network filtering を参照してください。
8. vNIC のパススルーを有効にして、仮想機能のデバイスを直接割り当てるようにするには、**パススルー**びチェックボックスを選択します。パススルーのプロパティーを有効にすると、QoS、ネットワークフィルターリング、ポートミラーリングに互換性がないため、これらが無効になります。パススルーの詳細は、「vNIC プロファイルでのパススルーの有効化」 を参照してください。
9. **Passthrough**を選択した場合には、オプションで**Migratable**チェックボックスの選択を解除すると、このプロファイルを使用する vNIC の移行が無効になります。このチェックボックスを選択したままの場合は、**Virtual Machine Management Guide**の Additional Prerequisites for Virtual Machines with SR-IOV-Enabled vNICs を参照してください。
10. **Port Mirroring**と**Allow all users to use this Profile**のチェックボックスを使って、これらのオプションを切り替えます。
11. カスタムプロパティーリストからカスタムプロパティーを選択すると、デフォルトで**Please select a key...**と表示されます。**+** および **-** ボタンを使用して、カスタムプロパティーを追加または削除します。
12. **OK** をクリックします。

このプロファイルをユーザーやグループに適用して、ネットワークの帯域幅を調整します。vNIC プロファイルを編集した場合は、仮想マシンを再起動するか、ゲスト OS が vNIC のホットプラグとホットアンプラグをサポートしている場合は、ホットアンプラグしてから vNIC をホットプラグする必要があります。

9.2.3. VM インターフェイスプロファイルウィンドウの設定内容の説明

表9.5 VM インターフェイスプロファイルウィンドウ

フィールド名	説明
ネットワーク	vNIC プロファイルの適用先の利用可能なネットワークのドロップダウンリストです。
Name	vNIC プロファイルの名前。これは、1 から 50 文字までの大文字と小文字、数字、ハイフン、およびアンダースコアの組み合わせが含まれる一意の名前である必要があります。
説明	vNIC プロファイルの説明。このフィールドは推奨されますが、必須ではありません。
QoS	vNIC プロファイルに適用する、利用可能な Network Quality of Service ポリシーのドロップダウンリストです。QoS ポリシーは、vNIC のインバウンドおよびアウトバウンドのネットワークトラフィックを規制します。
ネットワークフィルター	vNIC プロファイルに適用するネットワークフィルターのドロップダウンリストです。ネットワークフィルターは、仮想マシンとの間で送信可能なパケットの種類をフィルターリングして、ネットワークセキュリティーを向上させます。デフォルトのフィルターは、**vdsm-no-mac-spoofing** で **no-mac-spoofing** と **no-arp-mac-spoofing** を組み合わせたものです。libvirt が提供するネットワークフィルターの詳細は、**Red Hat Enterprise Linux Virtualization Deployment and Administration Guide** の Pre-existing network filters セクションを参照してください。 仮想マシンの VLAN やボンドには、**<ネットワークフィルターなし>** を使用してください。信頼できる仮想マシンでネットワークフィルターを使用しない場合には、パフォーマンスが向上します。  **注記** Red Hat では、**engine-config** ツールを使用して **Enable MACAnti Spoofing Filter Rules** パラメーターを **false** に設定してフィルターを無効にするサポートはなくなりました。代わりに **<No Network Filter>** オプションを使用してください。

フィールド名	説明
passthrough	パススルーのプロパティーを切り替えるためのチェックボックス。パススルーでは、vNIC がホスト NIC の仮想機能に直接接続できるようになります。vNIC プロファイルが仮想マシンにアタッチされている場合には、パススルーのプロパティーは編集できません。 パススルーを有効にすると、vNIC プロファイルで QoS、ネットワークフィルター、ポートミラーリングが無効になります。
Migratable	このプロファイルを使用する vNIC を移行可能化どうかを切り替えるチェックボックスです。移行は、通常の vNIC プロファイルではデフォルトで有効になっています。その場合にはチェックボックスが選択されており、変更できません。**Passthrough**チェックボックスが選択されていると、**Migratable** が有効になり、必要に応じて選択を解除して、パススルー vNIC の移行を無効にできます。
Port Mirroring	ポートミラーリングを切り替えるためのチェックボックスです。ポートミラーリングは、論理ネットワーク上のレイヤー 3 のネットワークトラフィックを、仮想マシン上の仮想インターフェイスにコピーします。デフォルトでは選択されていません。詳細は、**Technical Reference** の Port Mirroring を参照してください。
Device Custom Properties	vNIC プロファイルに適用する利用可能なカスタムプロパティーを選択するためのドロップダウンメニューです。+ と - ボタンを使用してプロパティーをそれぞれ追加、削除します。
Allow all users to use this Profile	環境内の全ユーザーがプロファイルを利用できるかどうかを切り替えるためのチェックボックスです。これはデフォルトで選択されます。

9.2.4. vNIC プロファイルでのパススルーの有効化

注記

これは、Red Hat Virtualization で SR-IOV を準備および設定する方法を示す一連のトピックの 1 つです。詳細は、Setting Up and Configuring SR-IOV を参照してください。

vNIC プロファイルのパススルーのプロパティーを使用すると、SR-IOV 対応 NIC の仮想機能 (VF) に vNIC を直接接続できるようになります。次に、vNIC はソフトウェアによるネットワーク仮想化をバイパスして、VF に直接接続してデバイスを割り当てます。

vNIC プロファイルがすでに vNIC にアタッチされている場合、パススループロパティーは有効にできません。この手順では、これを避けるために新しいプロファイルを作成します。vNIC プロファイルでパ

スルーが有効になっている場合、QoS、ネットワークフィルター、およびポートミラーリングを同じプロファイルで有効にすることはできません。

SR-IOV、直接デバイスの割り当て、および Red Hat Virtualization へのこれらの実装に関するハードウェアの考慮事項は、Hardware Considerations for Implementing SR-IOV を参照してください。

パススルーの有効化

1. **Network → Networks** をクリックします。
2. 論理ネットワーク名をクリックして、詳細ビューを開きます。
3. **vNIC Profiles**タブをクリックすると、その論理ネットワークのすべての v NIC プロファイルが一覧表示されます。
4. **New** をクリックします。
5. プロファイルの **Name** および **Description** を入力します。
6. **Passthrough**チェックボックスを選択します。
7. このプロファイルを使用する vNIC の移行を無効にするには、オプションで**Migratable**チェックボックスの選択を解除します。このチェックボックスを選択したままの場合は、**Virtual Machine Management Guide** の Additional Prerequisites for Virtual Machines with SR-IOV-Enabled vNICs を参照してください。
8. 必要に応じて、**Please select a key...**とデフォルトで表示されるカスタムプロパティーリストからカスタムプロパティーを選択します。+ および - ボタンを使用して、カスタムプロパティーを追加または削除します。
9. **OK** をクリックします。

vNIC プロファイルがパススルーに対応するようになりました。このプロファイルを使用して仮想マシンを NIC または PCI VF に直接アタッチするには、論理ネットワークを NIC にアタッチして、パススルー vNIC プロファイルを使用目的の仮想マシン上に新しい**PCI パススルー** vNIC を作成します。これらのそれぞれの手順の詳細は、**Virtual Machine Management Guide**の 「ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て」 および Adding a New Network Interface を参照してください。

9.2.5. vNIC プロファイルの削除

vNIC プロファイルを削除すると、仮想化環境から削除されます。

vNIC プロファイルの削除

1. **Network → Networks** をクリックします。
2. 論理ネットワーク名をクリックして、詳細ビューを開きます。
3. **vNIC Profiles**タブをクリックすると、利用可能な vNIC プロファイルが表示されます。
4. 1 つまたは複数のプロファイルを選択し、**削除**をクリックします。
5. **OK** をクリックします。

9.2.6. vNIC プロファイルへのセキュリティーグループの割り当て

注記

この機能は、OpenStack Networking (neutron) が外部ネットワークプロバイダーとして追加される場合にのみ利用できます。セキュリティーグループは Red Hat Virtualization Manager で作成できません。OpenStack を使用してセキュリティーグループを作成する必要があります。詳細は、**Red Hat OpenStack Platform Users and Identity Management Guide** の Project Security Management を参照してください。

OpenStack Networking インスタンスからインポートされ、Open vSwitch プラグインを使用するネットワークの vNIC プロファイルに、セキュリティー グループを割り当てることができます。セキュリティーグループとは、厳密に適用されるルールの集合体であり、ネットワークインターフェイス上のインバウンドおよびアウトバウンドのトラフィックをフィルターリングできます。以下の手順では、vNIC プロファイルにセキュリティーグループをアタッチする方法について説明します。

注記

セキュリティーグループは、OpenStack Networking インスタンスに登録されているセキュリティーグループの ID を使用して識別されます。特定のテナントのセキュリティーグループ ID を特定するには、OpenStack Networking がインストールされているシステムで以下のコマンドを実行します。

```
# neutron security-group-list
```

vNIC プロファイルへのセキュリティーグループの割り当て

1. **Network → Networks** をクリックします。
2. 論理ネットワーク名をクリックして、詳細ビューを開きます。
3. **vNIC Profiles** タブをクリックします。
4. **新規作成**をクリックするか、既存の vNIC プロファイルを選択して**編集**をクリックします。
5. カスタムプロパティーのドロップダウンリストから、**Security Groups**を選択します。カスタムプロパティーのドロップダウンを空白のままにすると、デフォルトのセキュリティー設定が適用され、すべてのアウトバウンドトラフィックとの相互通信が許可されますが、デフォルトのセキュリティーグループ外からのインバウンドトラフィックはすべて拒否されます。なお、後で**Security Groups**プロパティーを削除しても、適用されたセキュリティーグループには影響しません。
6. テキストフィールドに、vNIC プロファイルにアタッチするセキュリティーグループの ID を入力します。
7. **OK** をクリックします。

vNIC プロファイルにセキュリティーグループをアタッチしました。そのプロファイルが接続されている論理ネットワークを経由するすべてのトラフィックは、そのセキュリティーグループに定義されているルールに従ってフィルターリングされます。

9.2.7. vNIC プロファイルのユーザー権限

ユーザー権限を設定して、特定の vNIC プロファイルにユーザーを割り当てます。**Vnic Profile User**ロールをユーザーに割り当ててプロファイルの使用を可能にします。特定のプロファイルに対する権限を削除して、ユーザーを制限できます。

vNIC プロファイルのユーザー権限

1. **Network → vNIC Profile** をクリックします。
2. vNIC プロファイルの名前をクリックし、詳細ビューを開きます。
3. **Permissions**タブをクリックすると、そのプロファイルの現在のユーザー権限が表示されます。
4. **追加**または**削除**をクリックして、vNIC プロファイルのユーザー権限を変更します。
5. **Add Permissions to User**ウィンドウで**My Groups**をクリックすると、ユーザーグループが表示されます。このオプションを使用して、グループ内の他のユーザーに権限を付与できます。

vNIC プロファイルのユーザー権限を設定しました。

9.2.8. UCS 統合用の vNIC プロファイルの設定

Cisco の Unified Computing System (UCS) は、コンピューティング、ネットワーク、ストレージリソースなどのデータセンターの要素を管理するために使用されます。

vdsm-hook-vmfex-dev フックを使用すると、vNIC プロファイルを設定して、仮想マシンは Cisco の UCS 定義のポートプロファイルに接続できます。UCS で定義されたポートプロファイルには、UCS で仮想インターフェイスを設定するために使用されるプロパティーと設定が含まれています。**vdsm-hook-vmfex-dev** フックは、VDSM とともにデフォルトでインストールされます。詳細は、付録A *VDSM およびフック* を参照してください。

vNIC プロファイルを使用する仮想マシンが作成されると、Cisco vNIC が使用されます。

UCS 統合用の vNIC プロファイルを設定する手順では、最初にカスタムデバイスプロパティーを設定する必要があります。カスタムデバイスプロパティーを設定すると、そこに含まれる既存の値が上書きされます。新規および既存のカスタムプロパティーを組み合わせる場合は、キーの値を設定するために使用されるコマンドのすべてのカスタムプロパティーを含める必要があります。複数のカスタムプロパティーはセミコロンで区切られます。

注記

vNIC プロファイルを設定する前に、UCS ポートプロファイルを Cisco UCS で設定する必要があります。

カスタムデバイスプロパティーの設定

1. Red Hat Virtualization Manager で、**vmfex** カスタムプロパティーを設定し、**--cver** を使用してクラスターの互換性レベルを設定します。

```
# engine-config -s CustomDeviceProperties='{type=interface;prop={vmfex=^[a-zA-Z0-9_.-]{2,32}$}}' --cver=3.6
```

2. **vmfex** カスタムデバイスプロパティーが追加されていることを確認します。

```
# engine-config -g CustomDeviceProperties
```

3. **ovirt-engine** サービスを再起動します。

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

設定する vNIC プロファイルは、新規または既存の論理ネットワークに所属させることができます。新しい論理ネットワークを設定する手順は、「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」 を参照してください。

UCS 統合用の vNIC プロファイルの設定

1. **Network → Networks** をクリックします。
2. 論理ネットワーク名をクリックして、詳細ビューを開きます。
3. **vNIC Profiles** タブをクリックします。
4. **New** をクリックするか、vNIC プロファイルを選択し、**Edit** をクリックします。
5. プロファイルの **Name** および **Description** を入力します。
6. カスタムプロパティー リストから **vmfex** カスタムプロパティーを選択し、UCS ポートプロファイル名を入力します。
7. **OK** をクリックします。

9.3. 外部プロバイダーネットワーク

9.3.1. 外部プロバイダーからのネットワークのインポート

外部ネットワークプロバイダー (OpenStack Networking または OpenStack Neutron REST API を実装するサードパーティープロバイダー) からのネットワークを使用するには、プロバイダーを Manager に登録します。詳細は、Adding an OpenStack Network Service Neutron for Network Provisioning または Adding an External Network Provider を参照してください。その後、以下の手順でプロバイダーが提供するネットワークを Manager にインポートし、仮想マシンがネットワークを使用できるようにします。

外部プロバイダーからのネットワークのインポート

1. **Network → Networks**。
2. **Import** をクリックします。
3. **Network Provider**ドロップダウンリストから、外部のプロバイダーを選択します。そのプロバイダーが提供しているネットワークが自動的に検出され、**プロバイダーネットワーク**リストに表示されます。
4. チェックボックスを使って、**Provider Networks**リストでインポートするネットワークを選択し、下矢印をクリックしてそのネットワークを**Networks to Import**リストに移動させます。
5. インポートするネットワークの名前をカスタマイズすることができます。名前をカスタマイズするには、**名前**列でネットワークの名前をクリックして、テキストを変更します。
6. **Data Center**ドロップダウンリストから、ネットワークをインポートするデータセンターを選択します。
7. オプション: 対象のネットワークをすべてのユーザーが利用できるようにするには、**Allow All**チェックボックスをオフにします。
8. **Import** をクリックします。

選択されたネットワークはターゲットデータセンターにインポートされ、仮想マシンにアタッチできるようになります。詳細は、**Virtual Machine Management Guide**の Adding a New Network Interface を参照してください。

9.3.2. 外部プロバイダーネットワークの使用に関するの制限

外部プロバイダーからインポートした論理ネットワークを Red Hat Virtualization 環境で使用する場合には、以下の制限があります。

- 外部プロバイダーが提供する論理ネットワークは、仮想マシンのネットワークとして使用する必要があり、ディスプレイネットワークとして使用できません。
- 同一の論理ネットワークを複数回インポートできますが、インポートできるのは異なるデータセンターのみです。
- 外部プロバイダーが提供する論理ネットワークを Manager で編集できません。外部のプロバイダーが提供する論理ネットワークの詳細を編集するには、対象の論理ネットワークを提供している外部のプロバイダーから直接編集する必要があります。
- 外部のプロバイダーが提供する論理ネットワークに接続された仮想ネットワークインターフェイスカードでは、ポートミラーリングは利用できません。
- 仮想マシンが外部のプロバイダーが提供する論理ネットワークを使用している場合には、その論理ネットワークが仮想マシンで使用されている間は、そのプロバイダーを Manager から削除できません。
- 外部のプロバイダーが提供するネットワークは必須ではありません。そのため、ホスト選択時には、このような論理ネットワークのインポート先のクラスターのスケジューリングでこれらの論理ネットワークは考慮されません。さらに、このような論理ネットワークのインポート先のクラスターで、ユーザーが責任を持ってホストで論理ネットワークを利用できるようにしてください。

9.3.3. 外部プロバイダーの論理ネットワークのサブネット設定

外部のプロバイダーが提供する論理ネットワークでは、その論理ネットワーク上に 1 つ以上のサブネットが定義されている場合にのみ、仮想マシンに IP アドレスを割り当てることができます。サブネットが定義されていない場合には、仮想マシンには IP アドレスが割り当てられません。サブネットが 1 つの場合には、仮想マシンにそのサブネットから IP アドレスが割り当てられ、複数のサブネットがある場合には、仮想マシンに利用可能なサブネットのいずれかから IP アドレスが割り当てられます。論理ネットワークのホスト先の外部ネットワークプロバイダーが提供する DHCP サービスが、これらの IP アドレスを割り当てます。

Red Hat Virtualization Manager は、インポートされた論理ネットワーク上で定義済みのサブネットを自動的に検出しますが、Manager 内で論理ネットワークにサブネットを追加したり、論理ネットワークからサブネットを削除したりすることもできます。

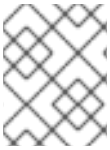
外部ネットワークプロバイダーとして Open Virtual Network (OVN)(ovirt-provider-ovn) を追加した場合には、複数のサブネットをルーターで接続できます。これらのルーターを管理するには、OpenStack Networking API v2.0 を使用できます。ただし、ovirt-provider-ovn には制限がありますのでご注意ください。ソース NAT(OpenStack API の enable_snat) は実装されていません。

9.3.4. 外部プロバイダー論理ネットワークへのサブネットの追加

外部のプロバイダーが提供する論理ネットワーク上にサブネットを作成します。

外部プロバイダー論理ネットワークへのサブネットの追加

1. **Network → Networks**。
2. 論理ネットワーク名をクリックして、詳細ビューを開きます。
3. **Subnets** タブをクリックします。
4. **New** をクリックします。
5. 新しいサブネットの**名前**と**CIDR**を入力します。
6. **IP Version** "ドロップダウンリストから、**IPv4**または**IPv6**のいずれかを選択します。
7. **OK** をクリックします。



注記

IPv6 については、Red Hat Virtualization でサポートされるのは静的なアドレスだけです。

9.3.5. 外部プロバイダー論理ネットワークからのサブネットの削除

外部プロバイダーが提供する論理ネットワークからサブネットを削除します。

外部プロバイダー論理ネットワークからのサブネットの削除

1. **Network → Networks**。
2. 論理ネットワーク名をクリックして、詳細ビューを開きます。
3. **Subnets** タブをクリックします。
4. サブネットを選択し、**削除**をクリックします。
5. **OK** をクリックします。

9.3.6. 論理ネットワークとポートへのセキュリティーグループの割り当て

注記

この機能は、Open Virtual Network (OVN) を外部ネットワークプロバイダーとして (ovirt-provider-ovn として) 追加した場合にのみ使用できます。セキュリティーグループは Red Hat Virtualization Manager で作成できません。セキュリティーグループの作成は、OpenStack Networking API v2.0 または Ansible で行う必要があります。

セキュリティーグループとは、厳密に適用されるルールの集合体であり、ネットワーク上のインバウンドおよびアウトバウンドのトラフィックをフィルターリングすることができます。また、セキュリティーグループを使って、ポートレベルでトラフィックをフィルターリングすることもできます。

Red Hat Virtualization 4.2.7 では、セキュリティーグループはデフォルトで無効になっています。

論理ネットワークへのセキュリティーグループの割り当て

1. **Compute → Clusters** をクリックします。
2. クラスター名をクリックして詳細ビューを開きます。

3. **Logical Networks** タブをクリックします。

4. **ネットワークの追加**をクリックしてプロパティーを定義し、**外部プロバイダー** ドロップダウンリストから **ovirt-provider-ovn** が選択されていることを確認します。詳細は、「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」 を参照してください。

5. **Security Group** ドロップダウンリストから **Enabled** を選択します。詳細は、「論理ネットワーク一般設定の説明」 を参照してください。

6. **OK** をクリックします。

7. OpenStack Networking API v2.0 または Ansible を使用して、セキュリティーグループを作成します。

8. OpenStack Networking API v2.0 または Ansible を使用して、セキュリティーグループのルールを作成します。

9. OpenStack Networking API v2.0 または Ansible を使用して定義したセキュリティーグループでポートを更新します。

10. オプション。セキュリティー機能をポートレベルで有効にするかどうかを定義します。現在のところ、これは OpenStack Networking API でのみ可能です。**port_security_enabled** 属性が設定されていない場合は、所属するネットワークで指定された値がデフォルトとなります。

9.4. ホストとネットワーキング

9.4.1. ホスト機能のリフレッシュ

ネットワークインターフェイスカードをホストに追加した場合、そのネットワークインターフェイスカードを Manager に表示するには、ホストの機能を更新する必要があります。

ホスト機能のリフレッシュ

1. **Compute → Hosts** をクリックし、ホストを選択します。

2. **Management → Refresh Capabilities** をクリックします。

選択したホストの **Network Interfaces** タブにあるネットワークインターフェイスカードの一覧が更新されます。新しいネットワークインターフェイスカードが、Manager で使用できるようになりました。

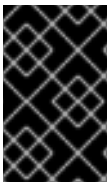
9.4.2. ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て

物理ホストネットワークインターフェイスの設定を変更したり、管理ネットワークを別の物理ホストネットワークインターフェイスに移動したり、物理ホストネットワークインターフェイスに論理ネットワークを割り当てたりすることができます。また、ブリッジや ethtool のカスタムプロパティーにも対応しています。

警告

Red Hat Virtualization でホストの IP アドレスを変更する唯一の方法は、そのホストを削除してから再度追加することです。

ホストの VLAN 設定を変更するには、「ホストの VLAN 設定の編集」 を参照してください。



重要

外部のプロバイダーが提供する論理ネットワークは、物理ホストのネットワークインターフェイスに割り当てることはできません。このような論理ネットワークは、仮想マシンが必要なときに動的にホストに割り当てられます。

注記

スイッチが LLDP(Link Layer Discovery Protocol) 情報を提供するように設定されている場合には、物理ネットワークインターフェイスにカーソルを合わせると、そのスイッチポートの現在の設定が表示されます。これにより、誤った設定を防ぐことができます。Red Hat は、論理ネットワークを割り当てる前に、以下の情報を確認することを推奨します。

- **Port Description (TLV タイプ 4)**と**System Name (TLV タイプ 5)**は、ホストのどのポート、そしてどのスイッチにパッチが当てられているかを検出するのに役立ちます。
- **Port VLAN ID**は、タグなしイーサーネットフレーム用にスイッチポートに設定されたネイティブ VLAN ID を表示します。スイッチポートに設定されているすべての VLAN が、**VLAN Name**と**VLAN ID**の組み合わせで表示されます。

ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て

1. **Compute → Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **Network Interfaces** タブをクリックします。
4. **Setup Host Networks** をクリックします。
5. 必要に応じて、ホストネットワークインターフェイスにカーソルを合わせ、スイッチが提供する設定情報を表示します。
6. 論理ネットワークを選択して、物理ホストネットワークインターフェイスの横にある**Assigned Logical Networks**エリアにドラッグして、論理ネットワークを物理ホストネットワークインターフェイスにアタッチします。

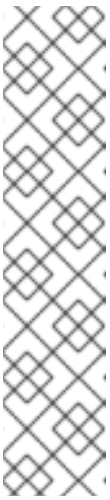
注記

1 つの NIC が複数の論理ネットワークに接続されている場合には、そのうちの 1 つのネットワークのみを VLAN 以外にすることができます。他のすべての論理ネットワークは、一意の VLAN でなければなりません。

7. 論理ネットワークの設定

   a. 割り当てられた論理ネットワークにカーソルを合わせ、鉛筆アイコンをクリックして **Edit Management Network** ウィンドウを開きます。

   b. **IPv4** タブの**Boot Protocol** で**None**、**DHCP** または **Static**を選択します。**Static**を選択した場合は、**IP**、**Netmask/Routing Prefix**、**Gateway**を入力します。



注記

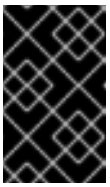
IPv6 は、静的 IPv6 アドレスのみ対応しています。論理ネットワークを設定するには、**IPv6**タブを選択し、次のように入力します。

- **Boot Protocol**を**Static**に設定します。
- **Routing Prefix**には、スラッシュと小数点を使って、接頭辞の**長さ**を入力します。例: **/48**
- **IP**: ホストのネットワークインターフェイスの完全な IPv6 アドレスです。例: **2001:db8::1:0:0:6**
- **ゲートウェイ**。送信元ルーターの IPv6 アドレス。例: **2001:db8::1:0:0:1**

注記

ホストの管理ネットワークの IP アドレスを変更した場合には、新しい IP アドレスの設定に、ホストを再インストール する必要があります。

各論理ネットワークには、管理ネットワークのゲートウェイとは別にゲートウェイを定義できます。これにより、論理ネットワークに到達したトラフィックは、管理ネットワークで使用されているデフォルトゲートウェイではなく、論理ネットワークのゲートウェイを使用して転送されます。



重要

クラスター内の**すべて** のホストが、管理ネットワークに同じ IP スタック (IPv4 または IPv6 のみ) を使用するように設定します。デュアルスタックには対応していません。

   c. **QoS**タブでは、デフォルトのホストネットワークのサービス品質を上書きします。**Override QoS** を選択し、以下のフィールドに必要な値を入力します。

- **Weight Share**: 同じ論理リンクリンクにアタッチされた他のネットワークと比較して、特定のネットワークに割り当てる必要がある論理リンクの容量を指定します。正確な共有は、そのリンクの全ネットワークの共有の合計によって異なります。デフォルトでは、この値は 1-100 の範囲の数字になります。
- **Rate Limit [Mbps]**: ネットワークによって使用される最大帯域幅。

- **Committed Rate [Mbps]**: ネットワークに必要な最小帯域幅。要求される Committed Rate は保証されず、ネットワークインフラストラクチャーおよび同じ論理リンクの他のネットワークによって要求される Commmitted Rate によって異なります。

d. ネットワークブリッジを設定するには、**Custom Properties**タブをクリックし、ドロップダウンリストから**bridge_opts**を選択します。有効なキーと値を次の構文で入力してください:**key**=**value**複数の項目を空白文字で区切ります。以下のキーが有効で、値は例として示されています。これらのパラメーターに関する詳しい情報は、「bridge_opts パラメーターの説明」 を参照してください。

```
forward_delay=1500
gc_timer=3765
group_addr=1:80:c2:0:0:0
group_fwd_mask=0x0
hash_elasticity=4
hash_max=512
hello_time=200
hello_timer=70
max_age=2000
multicast_last_member_count=2
multicast_last_member_interval=100
multicast_membership_interval=26000
multicast_querier=0
multicast_querier_interval=25500
multicast_query_interval=13000
multicast_query_response_interval=1000
multicast_query_use_ifaddr=0
multicast_router=1
multicast_snooping=1
multicast_startup_query_count=2
multicast_startup_query_interval=3125
```

e. イーサネットのプロパティーを設定するには、**Custom Properties**タブをクリックし、ドロップダウンリストから**ethtool_opts**を選択します。ethtool のコマンドライン引数の形式で、有効な値を入力してください。以下に例を示します。

```
--coalesce em1 rx-usecs 14 sample-interval 3 --offload em2 rx on lro on tso off --change em1 speed 1000 duplex half
```

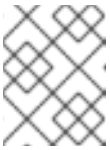
このフィールドは、ワイルドカードを使用できます。たとえば、このネットワークのすべてのインターフェイスに同じオプションを適用するには以下を使用します。

```
--coalesce * rx-usecs 14 sample-interval 3
```

ethtool_optsオプションはデフォルトでは使用できないので、エンジン設定ツールで追加する必要があります。詳細は、「Red Hat Virtualization Manager を使用するように Red Hat Virtualization Manager を設定する方法」 を参照してください。ethtool プロパティーの詳細は、コマンドラインで **man ethtool** と入力し、man ページを参照してください。

f. Fibre Channel over Ethernet (FCoE) を設定するには、**Custom Properties**タブをクリックし、ドロップダウンリストから**fcoe**を選択します。有効なキーと値を次の構文で入力してください:**key**=**value**最低でも **enable=yes** が必要です。また、 **dcb=** と **auto_vlan=[yes|no]** を追加することもできます。複数の項目を空白文字で区切りま

す。**fcoe**オプションはデフォルトでは利用できないので、エンジン設定ツールを使って追加する必要があります。詳細は、「FCoE を使用するように Red Hat Virtualization Manager を設定する方法」 を参照してください。



注記

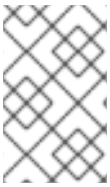
FCoE を使用する場合は、別途、専用の論理ネットワークを用意することをお勧めします。

g. ホストが使用するデフォルトネットワークを管理ネットワーク (ovirtmgmt) から非管理ネットワークに変更するには、非管理ネットワークのデフォルトルートを設定します。詳細は、「非管理者用論理ネットワークのデフォルトルートとしての設定」 を参照してください。

h. 論理ネットワークの定義がホストのネットワーク設定と同期していない場合は、**ネットワークの同期**チェックボックスを選択します。同期されていないホストとその同期方法の詳細は、「ホストネットワークの同期」 を参照してください。

8. **Verify connectivity between Host and Engine** チェックボックスを選択し、ネットワークの接続性を確認します。このアクションは、ホストがメンテナンスモードの場合にのみ機能します。

9. **OK** をクリックします。



注記

ホストのすべてのネットワークインターフェイスカードが表示されていない場合は **Management → Refresh Capabilities** をクリックして、そのホストで利用可能なネットワークインターフェイスカードのリストを更新します。

9.4.3. ホストネットワークの同期

ホスト上のインターフェイスの定義が、Manager が記憶している定義と異なる場合には、Manager はネットワークインターフェイスを **非同期** と定義します。同期がとれていないネットワークは、ホストの **Network Interfaces** タブに非同期アイコン ( ) を付けて、**Setup Host Networks** ウィンドウには アイコンをつけて表示されます。

ホストのネットワークが同期していない場合に、**Setup Host Networks**ウィンドウで同期していないネットワークに実行できるアクティビティーは、論理ネットワークをネットワークインターフェイスから切り離すか、ネットワークを同期させるかのいずれかのみです。

ホストが同期しなくなる仕組みを理解する

次のような場合には、ホストの同期が取れなくなります。

- **論理ネットワークの編集**ウィンドウなどを使わずに、ホスト上で設定を変更する。
  - 物理ホスト上の VLAN 識別子を変更する。
  - 物理ホストの**カスタム MTU**を変更する。
- ネットワーク名は同じだが、値やパラメーターが異なる別のデータセンターにホストを移動させる。

- ホストから手動でブリッジを削除してネットワークのVM Networkプロパティーを変更する。

ホストが非同期になるのを回避

以下のベストプラクティスに従うと、ホストの非同期化を回避できます。

1. ホストのローカルで変更するのではなく、管理ポータルで変更します。
2. 「ホストの VLAN 設定の編集」の説明に従って、VLAN の設定を編集します。

ホストの同期

ホストのネットワークインターフェイスの定義を同期させるには、Manager からの定義を使用してホストに適用します。これらの定義が必要でない場合は、同期後に管理ポータルからホストの定義を更新してください。ホストのネットワークを 3 つのレベルで同期させることができます。

- 論理ネットワーク別
- 1 ホスト別
- クラスター別

ホストネットワークを論理ネットワークレベルで同期させます。

1. **Compute** → **Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **Network Interfaces** タブをクリックします。
4. **Setup Host Networks** をクリックします。
5. 同期されていないネットワークにカーソルを合わせ、鉛筆アイコンをクリックして **Edit Network** ウィンドウを開きます。
6. **Sync network**のチェックボックスを選択します。
7. **OK**をクリックすると、ネットワークの変更が保存されます。
8. **OK** をクリックして **Setup Host Networks** ウィンドウを閉じます。

ホストレベルでのホストのネットワークの同期

- ホストのNetwork Interfacesタブにある**Sync All Networks**ボタンをクリックすると、ホストで同期していないネットワークインターフェイスがすべて同期されます。

クラスターレベルでのホストのネットワークの同期

- クラスターの **Logical Networks** タブの **Sync All Networks** ボタンをクリックして、クラスター全体の同期されていない論理ネットワーク定義をすべて同期します。

注記

REST API 経由でホストのネットワークを同期することもできます。**REST API Guide** の syncallnetworks を参照してください。

9.4.4. ホストの VLAN 設定の編集

ホストの VLAN 設定を変更するには、一旦 Manager からホストを削除し、再設定した後、再度 Manager に追加する必要があります。

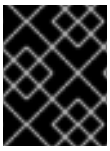
ネットワークを同期させるためには、以下を実行します。

1. ホストをメンテナンスモードにします。
2. 管理ネットワークを手動でホストから外します。これにより、ホストは新しい VLAN 上で到達可能になります。
3. ホストをクラスターに追加します。管理ネットワークに直接接続されていない仮想マシンは、ホスト間で安全に移行できます。

管理ネットワークの VLAN ID を変更すると、次のような警告メッセージが表示されます。

> Changing certain properties (e.g. VLAN, MTU) of the management network could lead to loss of connectivity to hosts in the data center, if its underlying network infrastructure isn't configured to accommodate the changes. Are you sure you want to proceed?

続行すると、データセンター内のすべてのホストが Manager への接続を失い、新しい管理ネットワークへのホストの移行が失敗してしまいます。管理ネットワークは、非同期と報告されます。



重要

管理ネットワークの VLAN ID を変更した場合は、ホストを再インストール して新しい VLAN ID を適用する必要があります。

9.4.5. 論理ネットワークを使用した単一のネットワークインターフェイスへの複数の VLAN の追加

1 つのネットワークインターフェイスに複数の VLAN を追加し、1 つのホストのトラフィックを分離できます。

重要

複数の論理ネットワークを作成した場合には、全論理ネットワークで**New Logical Network** または **Edit Logical Network** ウィンドウで **VLAN タグを有効にする**チェックボックスをチェックしておく必要があります。

論理ネットワークを使用したネットワークインターフェイスへの複数の VLAN の追加

1. **Compute → Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **Network Interfaces** タブをクリックします。
4. **Setup Host Networks** をクリックします。
5. VLAN タグ付きの論理ネットワークを、物理ネットワークインターフェイスの横にある**Assigned Logical Networks**エリアにドラッグします。物理ネットワークインターフェイスには、VLAN タグがあるので複数の論理ネットワークを割り当てることができます。

6. 論理ネットワークを設定します。

    a. 割り当てられた論理ネットワークにカーソルを合わせ、鉛筆アイコンをクリックします。

    b. 論理ネットワークの定義がホストのネットワーク設定と同期していない場合は、**ネットワークの同期**チェックボックスを選択します。

    c. **ブートプロトコル**を選択します。

    - None
    - DHCP
    - **静的**

    d. **IP**と**サブネットマスク**を入力してください。

    e. **OK** をクリックします。

7. **Verify connectivity between Host and Engine** チェックボックスを選択すると、ネットワークチェックが実行されますが、これはホストがメンテナンスモードの場合にのみ機能します。

8. **OK** をクリックします。

クラスター内の各ホストの NIC を編集して、論理ネットワークをクラスター内の各ホストに追加します。この後、ネットワークの運用が開始されます。

この作業を複数回繰り返し、それぞれのホストで同じネットワークインターフェイスを選択、編集して、異なる VLAN タグを割り当てた論理ネットワークを 1 つのネットワークインターフェイスに追加できます。

9.4.6. ホストネットワークへの追加の IPv4 アドレスの割り当て

ovirtmgmt 管理ネットワークなどのホストネットワークは、最初にセットアップされたときに 1 つの IP アドレスのみで作成されます。つまり、NIC の設定ファイル (例: **/etc/sysconfig/network-scripts/ifcfg-eth01**) に複数の IP アドレスが設定されている場合、最初にリストアップされた IP アドレスのみがホストネットワークに割り当てられることになります。ストレージに接続する場合や、同じ NIC を使って別のプライベートサブネット上のサーバーに接続する場合は、追加の IP アドレスが必要になることがあります。

vdsm-hook-extra-ipv4-addrs フックでは、ホストネットワークに追加の IPv4 アドレスを設定することができます。フックに関する詳細は、付録A *VDSM およびフック* を参照してください。

以下の手順では、追加の IP アドレスを設定する各ホストで、ホスト固有のタスクを実行する必要があります。

ホストネットワークへの追加の IPv4 アドレスの割り当て

1. 追加の IPv4 アドレスを設定したいホストに、VDSM のフックパッケージをインストールします。Red Hat Virtualization Host ではこのパッケージがデフォルトで利用可能ですが、Red Hat Enterprise Linux ホストにインストールする必要があります。

```
# yum install vdsm-hook-extra-ipv4-addrs
```

2. Manager で以下のコマンドを実行してキーを追加します。

```
# engine-config -s 'UserDefinedNetworkCustomProperties=ipv4_addrs=.*'
```

3. **ovirt-engine** サービスを再起動します。

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

4. 管理ポータルで **コンピュート → ホスト** をクリックします。
5. ホストの名前をクリックして、詳細ビューを開きます。
6. **Network Interfaces** タブをクリックし、**Setup Host Networks** をクリックします。
7. 割り当てられた論理ネットワークの上にカーソルを置き、鉛筆のアイコンをクリックして、ホストネットワークインターフェイスを編集します。
8. **Custom Properties** のドロップダウンリストから **ipv4_addr** を選択し、IP アドレスと接頭辞を追加します (5.5.5.5/24 など)。複数の IP アドレスはコンマで区切る必要があります。
9. **OK** をクリックして、**Edit Network** ウィンドウを閉じます。
10. **OK** をクリックして **Setup Host Networks** ウィンドウを閉じます。

追加された IP アドレスは Manager には表示されませんが、ホスト上で **ip addr show** コマンドを実行することで、追加されたことを確認できます。

9.4.7. ホストネットワークインターフェイスへのネットワークラベルの追加

ネットワークラベルを使用すると、ホストネットワークインターフェイスへの論理ネットワークの割り当てに関連する管理ワークロードを大幅に簡素化できます。ロールネットワーク (たとえば、移行ネットワークやディスプレイネットワーク) にラベルを設定すると、そのネットワークがすべてのホストに大量に展開されます。このようなネットワークの大量追加は、DHCP を使って実現しています。多くの静的 IP アドレスを入力するタスクのスケーラブルでない性質のため、この大量展開の方法は、静的アドレスを入力する方法よりも選択されました。

ホストネットワークインターフェイスにラベルを追加するには 2 つの方法があります。

- 管理ポータルで手動で実行する
- LLDP Labeler サービスで自動で実行する

管理ポータルでのネットワークラベルの追加

1. **Compute → Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **Network Interfaces** タブをクリックします。
4. **Setup Host Networks** をクリックします。
5. **ラベル**をクリックし、**新規ラベル**を右クリックします。ラベルを貼る物理ネットワークインターフェイスを選択します。
6. **ラベルテキストフィールド**にネットワークラベルの名前を入力します。
7. **OK** をクリックします。

LLDP ラベルサービスを使用したネットワークラベルの追加

LLDP Labeler サービスを使用すると、設定済みのクラスターリスト内のホストネットワークインターフェイスにラベルを割り当てるプロセスを自動化できます。

デフォルトでは、LLDP Labeler は 1 時間ごとのサービスとして動作します。このオプションは、ハードウェアを変更する場合 (NIC、スイッチ、ケーブルなど)、またはスイッチ設定を変更する場合に役立ちます。

前提条件

- インターフェイスは、ジュニパー製スイッチに接続されている必要があります。
- ジュニパーのスイッチは、LLDP を使って **Port VLAN** を提供するように設定する必要があります。

手順

1. **/etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf** で **username** および **password** を設定します。
   - **username**: Manager 管理者のユーザー名。デフォルトは **admin@internal** です。
   - **password**: Manager 管理者のパスワード。デフォルトは **123456** です。
2. **etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf** で以下の値を更新して、LLDP Labeler サービスを設定します。
   - **clusters**: サービスが実行されるクラスターのコンマ区切りリスト。ワイルドカードはサポートされます。たとえば、**Cluster*** は、**Cluster** という単語から始まるすべてのクラスターで実行する LLDP ラクターを定義します。データセンター内のすべてのクラスターでサービスを実行するには、* と入力します。デフォルトは **Def*** です。
   - **api_url**: Manager の API の完全な URL。デフォルトは **https://Manager_FQDN/ovirt-engine/api** です。
   - **ca_file**: カスタム CA 証明書ファイルへのパス。カスタム証明書を使用しない場合は、この値を空欄のままにします。デフォルトは空です。
   - **auto_bonding**: LLDP ラベラーのボンディング機能を有効にします。デフォルトは **true** です。
   - **auto_labeling**: LLDP ラベラーのラベリング機能を有効にします。デフォルトは **true** です。
3. 必要に応じて、**etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-labeler.timer** の **OnUnitActiveSec** の値を変更することで、別の時間間隔でサービスを実行するように設定できます。デフォルトは **1h** です。
4. 以下のコマンドを入力して、現在およびシステムの起動時にサービスが開始するように設定します。

```
# systemctl enable --now ovirt-lldp-labeler
```

手動でサービスを呼び出すには、以下のコマンドを入力します。

```
# /usr/bin/python /usr/share/ovirt-lldp-labeler/ovirt_lldp_labeler_cli.py
```

ホストのネットワークインターフェイスにネットワークラベルを追加しました。同じラベルで新しく作成された論理ネットワークは、そのラベルを持つすべてのホストネットワークインターフェイスに自動的に割り当てられます。論理ネットワークからラベルを削除すると、そのラベルを持つすべてのホストネットワークインターフェイスからその論理ネットワークが自動的に削除されます。

9.4.8. ホストの FQDN の変更

以下の手順で、ホストの完全修飾ドメイン名を変更します。

ホストの FQDN の更新

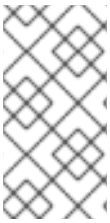
1. ホストをメンテナンスモードにして、仮想マシンが別のホストにライブマイグレーションされるようにします。詳細は、「ホストのメンテナンスモードへの切り替え」 を参照してください。あるいは、すべての仮想マシンを手動でシャットダウンするか、別のホストに移行してください。詳細は、**Virtual Machine Management Guide**の Manually Migrating Virtual Machines を参照してください。
2. **Remove** をクリックし、**OK** をクリックすると、管理ポータルからホストが削除されます。
3. **hostnamectl** ツールを使用して、ホスト名を更新します。その他のオプションについては、**Red Hat Enterprise Linux 7 Networking Guide**の Configure Host Names を参照してください。

```
# hostnamectl set-hostname NEW_FQDN
```

4. ホストを再起動します。
5. ホストをマネージャーに再登録する。詳細は、「Red Hat Virtualization Manager への通常のホストの追加」 を参照してください。

9.4.9. IPv6 ネットワーキングサポート

Red Hat Virtualization はほとんどのコンテキストで静的 IPv6 ネットワーキングをサポートします。



注記

Red Hat Virtualization では、Manager を実行しているコンピューターまたは仮想マシン (または Manager マシン) で、引き続き IPv6 を有効にする必要があります。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。

IPv6 の制限事項

- スタティックな IPv6 アドレッシングのみ対応しています。**DHCP** や **Stateless Address Autoconfiguration** による動的な IPv6 アドレスの設定はサポートしていません。
- デュアルスタックアドレッシング、IPv4 **および** IPv6 はサポートされていません。
- OVN のネットワークは、IPv4 **または** IPv6 のみで使用できます。
- クラスターの IPv4 から IPv6 への切り替えはサポートされていません。
- IPv6 では、ホストごとに 1 つのゲートウェイしか設定できません。
- 両方のネットワークが単一のゲートウェイを共有している (同じサブネット上にある) 場合に

は、デフォルトルートのロールを管理ネットワーク (ovirtmgmt) から別の論理ネットワークに移動できます。ホストと Manager は同じ IPv6 ゲートウェイを持つ必要があります。ホストと Manager が同じサブネット上にない場合、IPv6 ゲートウェイが削除されたために Manager がホストとの接続を失う可能性があります。

- IPv6 アドレスの gluster サーバーで glusterfs ストレージドメインを使用することはサポートされていません。

9.4.10. SR-IOV の設定および設定

このトピックでは、SR-IOV のセットアップと設定の手順をまとめ、各手順の詳細を説明するトピックへのリンクを掲載しています。

9.4.10.1. 前提条件

SR-IOV を実装するためのハードウェアの考慮事項 に従ってハードウェアをセットアップします

9.4.10.2. SR-IOV の設定および設定

SR-IOV をセットアップして設定するには、以下のタスクを実行します。

1. PCI パススルーのホストの設定
2. NIC の仮想機能設定の編集
3. vNIC プロファイルでのパススルーの有効化
4. 移行中のネットワーク停止を減らすための SR-IOV 対応 vNIC が設定された仮想マシンの設定

注記

- パススルーの vNIC の数は、ホスト上で利用可能な仮想機能 (VF) の数によって異なります。たとえば、3 つの SR-IOV カード (vNIC) で仮想マシン (VM) を実行するには、ホストで 3 つ以上の VF が有効になっている必要があります。
- ホットプラグとアンプラグに対応しています。
- ライブマイグレーションは RHV バージョン 4.1 以降でサポートされます。
- VM を移行するためには、移行先のホストにも VM を受け入れるのに十分な空き VF が必要です。マイグレーションの際、VM はソースホスト上のいくつかの VF を解放し、デスティネーションホスト上で同じ数の VF を占有します。
- ホストには、他のインターフェイスと同様に、デバイス、リンク、または ifcae が表示されます。そのデバイスは、VM に装着すると消え、離すと再び現れます。
- SR-IOV 機能では、ホストデバイスを VM に直接接続することは避けてください。
- 複数の VLAN を持つトランクポートとして VF を使用し、ゲスト内で VLAN を設定するには、Cannot configure VLAN on SR-IOV VF interfaces inside the Virtual Machine を参照してください。

ここでは、インターフェイスの libvirt XML がどのように見えるかの例を示します。

```
----
<interface type='hostdev'>
```

```
  <mac address='00:1a:yy:xx:vv:xx'/>
  <driver name='vfio'/>
  <source>
    <address type='pci' domain='0x0000' bus='0x05' slot='0x10' function='0x0'/>
  </source>
  <alias name='ua-18400536-5688-4477-8471-be720e9efc68'/>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x08' function='0x0'/>
</interface>
----
```

トラブルシューティング

以下の例は、インターフェイスにアタッチされている VF に関する診断情報を取得する方法を示しています。

```
# ip -s link show dev enp5s0f0

1: enp5s0f0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP mode
DEFAULT qlen 1000
    link/ether 86:e2:ba:c2:50:f0 brd ff:ff:ff:ff:ff:ff
    RX: bytes  packets  errors  dropped overrun mcast
    30931671   218401   0       0       0       19165434
    TX: bytes  packets  errors  dropped carrier collsns
    997136     13661    0       0       0       0
    vf 0 MAC 02:00:00:00:00:01, spoof checking on, link-state auto, trust off, query_rss off
    vf 1 MAC 00:1a:4b:16:01:5e, spoof checking on, link-state auto, trust off, query_rss off
    vf 2 MAC 02:00:00:00:00:01, spoof checking on, link-state auto, trust off, query_rss off
```

9.4.10.3. 関連情報

- RHV VM に SR-IOV パススルーを設定する方法
- RHV の SR-IOV VF(Virtual Function) でボンディングを設定する方法
- RHV で仮想マシンに専用の仮想 NIC を割り当てるために、ホストデバイスのパススルーと SR-IOV を有効にする方法

9.5. ネットワークボンディング

ネットワークボンディングは、複数の NIC を 1 つのボンドデバイスにまとめるもので、以下のようなメリットがあります。

- ボンディングされた NIC の伝送速度は、シングル NIC の伝送速度よりも高速です。
- ネットワークボンディングは、ボンドデバイスのすべての NIC が故障しない限り、ボンドデバイスは故障しないため、フォールトトレランスを提供します。

同じメーカー、同じモデルの NIC を使用することで、同じボンディングオプションやモードをサポートすることができます。

重要

Red Hat Virtualization のデフォルトのボンディングモードである **(Mode 4) Dynamic Link Aggregation** には、802.3ad をサポートするスイッチが必要です。

ボンディングの論理的なネットワークには互換性がなければなりません。ボンディングは、1 つの VLAN 以外の論理ネットワークのみをサポートします。残りの論理ネットワークには、固有の VLAN ID を設定する必要があります。

スイッチのポートでボンディングを有効にする必要があります。具体的な方法は、スイッチのベンダーが提供するマニュアルを参照してください。

ネットワークボンドデバイスは、以下のいずれかの方法で作成することができます。

- 管理ポータル で、特定のホストに対して手動で
- クラスターやデータセンター内の全ホストのアンボンド NIC に対して、LLDP Labeler を用いて自動的に

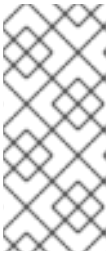
ご使用の環境で iSCSI ストレージを使用していて、冗長性を実装する場合は、iSCSI マルチパスを設定するための手順に従ってください。

9.5.1. 管理ポータルでのボンドデバイスの作成

管理ポータルで特定のホストにボンドデバイスを作成することができます。ボンドデバイスは、VLAN タグ付きのトラフィックとタグなしのトラフィックの両方を伝送することができます。

手順

1. **Compute → Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **Network Interfaces** タブをクリックすると、ホストに接続されている物理的なネットワークインターフェイスが一覧表示されます。
4. **Setup Host Networks** をクリックします。
5. スイッチの設定を確認してください。スイッチが LLDP (Link Layer Discovery Protocol) 情報を提供するように設定されている場合は、物理的な NIC の上にカーソルを置くと、スイッチポートのアグリゲーション設定が表示されます。
6. NIC を他の NIC やボンドにドラッグ&ドロップします。



注記

2 枚の NIC が新しい結合を形成します。NIC とボンドは、既存のボンドに NIC を追加します。

論理ネットワークに 互換性がない 場合、ボンディング操作はブロックされます。

7. ドロップダウンメニューから **Bond Name** および **Bonding Mode** を選択します。詳細は、「ボンディングモード」 を参照してください。

カスタム ボンディングモードを選択した場合、次の例のように、テキストフィールドにボンディングオプションを入力できます。

- ご使用の環境で **ethtool** を使用してリンク状態が報告されない場合は、**mode=1arp_interval=1arp_ip_target=192.168.0.2** と入力して ARP モニターリングを設定できます。
- **mode=1 primary=eth0** と入力すると、スループットの高い NIC をプライマリーインターフェイスとして指定できます。
  ボンディングオプションとその説明の包括的なリストについては、Kernel.org の Linux Ethernet Bonding Driver HOWTO を参照してください。

8. **OK** をクリックします。
9. 新しいボンドに論理ネットワークを付けて設定します。手順は 「ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て」 を参照してください。

注記

論理ネットワークをボンド内の個々の NIC に直接アタッチすることはできません。

10. オプションで、ホストがメンテナンスモードの場合は、**ホストとエンジン間の接続の確認** を選択できます。
11. **OK** をクリックします。

9.5.2. LLDP Labeler Service によるボンドデバイスの作成

LLDP Labeler サービスを利用すると、1 つ以上のクラスターまたはデータセンター全体のすべてのホストに対して、すべてのアンボンド NIC で自動的にボンドデバイスを作成することができます。ボンディングモードは **(モード 4) 動的リンクアグリゲーション (802.3ad)** です。

互換性のない論理ネットワーク を持つ NIC は結合できません。

デフォルトでは、LLDP Labeler は 1 時間ごとのサービスとして動作します。このオプションは、ハードウェアを変更する場合 (NIC、スイッチ、ケーブルなど)、またはスイッチ設定を変更する場合に役立ちます。

前提条件

- インターフェイスは、ジュニパー製スイッチに接続されている必要があります。
- ジュニパースイッチは、LLDP を使用してリンクアグリゲーション制御プロトコル (LACP) 用に設定する必要があります。

手順

1. **/etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf** で **username** および **password** を設定します。
   - **username**: Manager 管理者のユーザー名。デフォルトは **admin@internal** です。
   - **password**: Manager 管理者のパスワード。デフォルトは **123456** です。

2. **etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf** で以下の値を更新して、LLDP Labeler サービスを設定します。

   - **clusters**: サービスが実行されるクラスターのコンマ区切りリスト。ワイルドカードはサポートされます。たとえば、**Cluster*** は、**Cluster** という単語から始まるすべてのクラスターで実行する LLDP ラクターを定義します。データセンター内のすべてのクラスターでサービスを実行するには、* と入力します。デフォルトは **Def*** です。

   - **api_url**: Manager の API の完全な URL。デフォルトは **https://Manager_FQDN/ovirt-engine/api** です。

   - **ca_file**: カスタム CA 証明書ファイルへのパス。カスタム証明書を使用しない場合は、この値を空欄のままにします。デフォルトは空です。

   - **auto_bonding**: LLDP ラベラーのボンディング機能を有効にします。デフォルトは **true** です。

   - **auto_labeling**: LLDP ラベラーのラベリング機能を有効にします。デフォルトは **true** です。

3. 必要に応じて、**etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-labeler.timer** の **OnUnitActiveSec** の値を変更することで、別の時間間隔でサービスを実行するように設定できます。デフォルトは **1h** です。

4. 以下のコマンドを入力して、現在およびシステムの起動時にサービスが開始するように設定します。

   ```
   # systemctl enable --now ovirt-lldp-labeler
   ```

   手動でサービスを呼び出すには、以下のコマンドを入力します。

   ```
   # /usr/bin/python /usr/share/ovirt-lldp-labeler/ovirt_lldp_labeler_cli.py
   ```

5. 新しいボンドに論理ネットワークを付けて設定します。手順は 「ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て」 を参照してください。

注記

論理ネットワークをボンド内の個々の NIC に直接アタッチすることはできません。

9.5.3. ボンディングモード

パケット分散アルゴリズムは、ボンディングモードによって決定されます。(詳細は Linux Ethernet Bonding Driver HOWTO を参照してください)。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(802.3ad)** です。

Red Hat Virtualization は、仮想マシン (ブリッジド) ネットワークで使用できるため、以下のボンディングモードをサポートしています。

(Mode 1) Active-Backup

1 枚の NIC がアクティブです。アクティブな NIC が故障した場合、バックアップ NIC の 1 つがボンド内の唯一のアクティブな NIC としてその NIC を置き換えます。このボンドの MAC アドレスは、ネットワークアダプターのポートにのみ表示されます。これにより、ボンドの MAC アドレスが変更

されても、新しいアクティブな NIC の MAC アドレスが反映されるため、MAC アドレスの混乱を防ぐことができます。

(Mode 2) Load Balance (balance-xor)

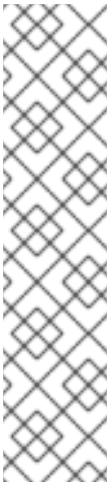
パケットを送信する NIC は、送信元 MAC アドレスと宛先 MAC アドレスに対して XOR 演算を実行し、NIC の総数の **modulo** を掛けて選択されます。このアルゴリズムでは、各宛先 MAC アドレスに対して同じ NIC が選択されるようになっています。

(Mode 3) Broadcast

パケットはすべての NIC に送信されます。

(Mode 4) Dynamic Link Aggregation(802.3ad) (デフォルト)

NIC は、同じ速度とデュプレックスの設定を共有するグループに集約されます。アクティブなアグリゲーショングループのすべての NIC が使用されます。



注記

(Mode 4) Dynamic Link Aggregation(802.3ad) は 802.3ad 対応のスイッチが必要です。

ボンディングされた NIC は、同じアグリゲーター ID を持つ必要があります。それ以外の場合、マネージャーは **ネットワークインターフェイス** タブのボンドに警告感嘆符アイコンを表示し、ボンドの **ad_partner_mac** 値は **00:00:00:00:00:00** として報告されます。以下のコマンドを入力することで、アグリゲーター ID を確認できます。

```
# cat /proc/net/bonding/bond0
```

https://access.redhat.com/solutions/67546 を参照してください。

Red Hat Virtualization はブリッジネットワークで使用できず、仮想マシンの論理ネットワークと互換性がないため、以下のボンディングモードはサポートされません。

(Mode 0) Round-Robin

NIC は、パケットを順番に送信します。パケットは、ボンド内の利用可能な最初の NIC から始まり、ボンド内の利用可能な最後の NIC で終わるループで送信されます。後続のループは、最初に利用可能な NIC から始まります。

(Mode 5) Balance-TLB (Transmit Load-Balance)

発信するトラフィックは、負荷に応じて、ボンド内のすべての NIC に分散されます。受信トラフィックは、アクティブな NIC で受信されます。受信する NIC が故障した場合、別の NIC が割り当てられます。

(Mode 6) Balance-ALB (Adaptive Load-Balance とも呼ばれる)

(Mode 5) Balance-TLB は、IPv4 トラフィックの受信負荷分散と組み合わせます。ARP ネゴシエーションは、受信負荷のバランスをとるために使用されます。

9.6. ネットワーク接続性の分析と監視

9.6.1. Skydive の導入

Skydive を使用すると、外部ネットワークプロバイダー として定義された Open Virtual Networks(OVN) を含む論理ネットワークを監視できます。Skydive は、ネットワークトポロジー、依存関係、フローのライブビューの提供、レポート生成、設定監査を実行します。

Skydive が提示するデータを使用して、以下を行うことができます。

- パケットロスの検出
- ブリッジやインターフェイスを含むクラスターのネットワークトポロジーを取得して、デプロイメントが正しく機能していることを確認します
- 予想される MTU 設定が正しく適用されているかどうかを確認します。
- 仮想マシン間、または仮想マシンとホストの間のネットワークトラフィックの取得

Skydive の機能セットの詳細は、http://skydive.network を参照してください。

注記

Skydive はテクノロジープレビュー機能です。テクノロジープレビュー機能は、Red Hat の実稼働環境でのサービスレベルアグリーメント (SLA) ではサポートされていないため、Red Hat では実稼働環境での使用を推奨していません。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。

Red Hat のテクノロジープレビュー機能のサポートについての詳細は、テクノロジープレビュー機能のサポート範囲 を参照してください。

9.6.2. Skydive のインストール

手順

1. Manager マシンに **skydive-ansible** をインストールします。

```
# yum --disablerepo="*" --enablerepo="rhel-7-server-rpms,rhel-7-server-extras-rpms,rhel-7-server-rh-common-rpms,rhel-7-server-openstack-14-rpms" install skydive-ansible
```

2. **/usr/share/ovirt-engine/playbooks/install-skydive.inventory.sample** を現在のディレクトリーにコピーし、この名前を **inventory** に変更します。

3. 以下のように **inventory/01_hosts** ファイルを変更します (完全なコンテンツについては、以下を参照してください)。

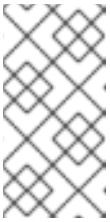
   a. **skydive_os_auth_url** を Manager の FQDN に更新します。これは、Manager と同じ FQDN を使用する OVN により使用されます。

   b. **ovn_provider_username** を OVN プロバイダーに使用されるユーザー名で更新します。デフォルトは **/etc/ovirt-provider-ovn/ovirt-provider-ovn.conf** で定義されています。

   c. **ovn_provider_password** を更新します。

   d. **[agents:children] <host_group>** で、Skydive エージェントをインストールするホスト、クラスター、またはデータセンターを定義します。
   以下を実行して有効なグループの一覧を表示できます。

```
/usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory | python -m json.tool
```



注記

各ホストを明示的に一覧表示する必要はありません。クラスターのすべてのホストにエージェントをインストールするには、**ovirt_cluster_Default** を追加します。または、データセンター内のすべてのホストにエージェントをインストールするには、**ovirt_datacenter_Default** を追加します。

インベントリーファイルのサンプル

```
[agents]
[analyzers]
[skydive:children]
 analyzers
 agents

[skydive:vars]
skydive_listen_ip=0.0.0.0
skydive_deployment_mode=package
skydive_extra_config={'agent.topology.probes': ['ovsdb', 'neutron'],
'agent.topology.neutron.ssl_insecure': true}

 skydive_fabric_default_interface=ovirtmgmt

 skydive_os_auth_url=https://MANAGERS_FQDN:35357/v2.0
 skydive_os_service_username=ovn_provider_username
 skydive_os_service_password=ovn_provider_password
 skydive_os_service_tenant_name=service
 skydive_os_service_domain_name=Default
 skydive_os_service_region_name=RegionOne

[agents:vars]
ansible_ssh_private_key_file=/etc/pki/ovirt-engine/keys/engine_id_rsa

[agents:children]
host_group

[analyzers]
localhost ansible_connection=local
```

4. Playbook を実行します。

```
# ansible-playbook -i inventory /usr/share/ovirt-engine/playbooks/install-skydive.yml
/usr/share/skydive-ansible/playbook.yml.sample
```

5. Skydive が仮想マシンのポートを認識していることを確認するには、http://**MANAGERS_FQDN**:8082 にアクセスし、仮想マシンを選択して、**Capture** タブの **Metadata** セクションで次のフィールドを確認します。

- Manager: Neutron
- networkName: **network_name**
- ipV4: **IP_address** (サブネットが使用される場合)

Skydive を使用してネットワークのアクティビティーをキャプチャーする方法の例を確認するには、「Skydive を使用したネットワーク接続のテスト」 を参照してください。

9.6.3. Skydive を使用したネットワーク接続のテスト

以下の例では、複数の IPv4 アドレスのある NIC を持つ 2 つのホスト間の接続をテストします。NIC は VLAN 4 としてタグ付けされた論理ネットワークに接続されます。IP アドレスを論理ネットワークに割り当てる方法は、「ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て」 を参照してください。

手順

1. Skydive をインストールします。
2. **http://MANAGERS_FQDN:8082** から Skydive を開きます。
3. ネットワークマップの **rhv-host1** で **network_4** を選択します。
4. **Capture** タブで **Create** をクリックし、**Start** をクリックします。
5. **rhv-host0** の **network_4** の以前の手順を繰り返します。
6. **Generate** タブをクリックします。
7. **Source** として **rhv-host0 のeth0** を選択し、**Destination** として **rhv-host1** の **eth0** を選択します。
8. **タイプ** ドロップダウンリストから **ICMPv4/Echo Request** を選択します。
9. **Inject** をクリックしてパケットを挿入します。
10. **Flows** タブを開きます。ping の結果がテーブルに表示されます。ping が成功すると、**ICMPv4** と送信元および宛先 IP アドレスを含む行が表示されます。その行にカーソルを合わせると、**network_4** がネットワークマップ上で黄色い円で強調表示されます。

Skydive の使用に関する詳細は、Skydive ドキュメント を参照してください。

For installation is :Testing!:

第10章 ホスト

10.1. ホストの HQL の概要

ホスト (ハイパーバイザーとも呼ばれる) は、仮想マシンが動作する物理サーバーです。Kernel-based Virtual Machine (KVM) と呼ばれるローダーブル Linux カーネルモジュールを使用することで、完全な仮想化が提供されます。

KVM は、Windows または Linux いずれかのオペレーティングシステムを実行する複数の仮想マシンを、同時にホストすることができます。仮想マシンはホストマシン上で個々の Linux プロセスやスレッドとして実行され、Red Hat Virtualization Manager によってリモートで管理されます。Red Hat Virtualization の環境には、それに接続された 1 つ以上のホストがあります。

Red Hat Virtualization は、ホストをインストールする 2 つの方法をサポートしています。Red Hat Virtualization Host (RHVH) のインストールメディアを使用するか、標準の Red Hat Enterprise Linux インストールにハイパーバイザーパッケージをインストールすることができます。

注記

Red Hat Virtualization Manager で個別のホストのホストタイプを特定するには、ホスト名を選択して詳細ビューを開き、**Software** で **OS Description** を確認してください。

ホストは、仮想化の最適化を提供する **tuned** プロファイルを使用します。**tuned** に関する詳細は、Red Hat Enterprise Linux 7 Performance Tuning Guide を参照してください。

Red Hat Virtualization Host は、セキュリティー機能が有効になっています。SELinux (Security Enhanced Linux) とファイアウォールは完全に設定されており、デフォルトでオンになっています。選択したホストの SELinux の状態は、詳細表示の **General** タブの **SELinux mode** で報告されます。Manager は、Red Hat Enterprise Linux ホストを環境に追加する際に、必要なポートを開くことができます。

ホストとは、Red Hat Enterprise Linux 7 AMD64/Intel 64 版が動作する Intel VT または AMD-V 拡張機能を持つ物理的な 64 ビットサーバーのことです。

Red Hat Virtualization プラットフォーム上の物理的なホストのこと。

- システム内の 1 つのクラスターにのみ属すること。
- AMD-V または Intel VT ハードウェア仮想化拡張をサポートする CPU を搭載していること。
- クラスター作成時に選択された仮想 CPU タイプで提供されるすべての機能をサポートする CPU が必要です。
- 最小 2 GB の メモリー。
- システム権限を持つシステム管理者を割り当てることができる。

管理者は Red Hat Virtualization のウォッチリストから最新のセキュリティー勧告を受け取ることができます。Red Hat Virtualization ウォッチリストに登録すると、Red Hat Virtualization 製品の新しいセキュリティー勧告を電子メールで受け取ることができます。このフォームに必要事項を入力してください。

https://www.redhat.com/mailman/listinfo/rhsa-announce

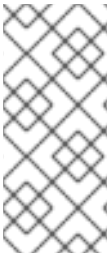
10.2. RED HAT VIRTUALIZATION HOST

Red Hat Virtualization Host (RHVH) は、仮想マシンをホストするのに必要なパッケージのみを搭載した Red Hat Enterprise Linux の特別なビルドを使用してインストールされます。Red Hat Enterprise Linux ホストで使用されているものをベースにした **Anaconda** インストールインターフェイスを使用しており、Red Hat Virtualization Manager または **yum** を通じて更新することができます。追加のパッケージをインストールして、アップグレード後もそれを維持するには、**yum** コマンドを使うのが唯一の方法です。

RHVH には、ホストのリソースを監視したり、管理作業を行うためのコックピット Web インターフェイスがあります。SSH やコンソールを介した RHVH への直接アクセスはサポートされていないため、Cockpit ウェブインターフェイスは、ネットワークの設定や、セルフホストエンジンのデプロイ、**Terminal**サブタブを介したターミナルコマンドの実行など、ホストが Red Hat Virtualization Manager に追加される前に実行されるタスクのためのグラフィカルユーザーインターフェイスを提供します。

Web ブラウザーで https://**Host FQDNor IP:9090**、コックピットのウェブインターフェイスにアクセスします。Cockpit for RHVH には、ホストのヘルスステータス、SSH ホストキー、セルフホスト型エンジンのステータス、仮想マシン、および仮想マシンの統計情報を表示するカスタム**仮想化**ダッシュボードが含まれています。

RHVH では、アプリケーションのクラッシュに関する意味のあるデバッグ情報を収集するために、ABRT (Automatic Bug Reporting Tool) を使用しています。詳細は Red Hat Enterprise Linux System Administrator's Guide を参照してください。



注記

カスタムブートカーネル引数は、**grubby** ツールを使用して Red Hat Virtualization Host に追加することができます。**grubby** ツールは、**grub.cfg** ファイルに永続的な変更を加えます。ホストの Cockpit ウェブインターフェイスの **Terminal** サブタブに移動し、**grubby** コマンドを使用します。詳細は Red Hat Enterprise Linux System Administrator's Guide を参照してください。

警告

Red Hat は、ローカルセキュリティーの脆弱性が悪用される可能性があるため、RHVH で信頼できないユーザーを作成しないことを強くお勧めします。

10.3. RED HAT ENTERPRISE LINUX ホスト

可能なハードウェアにインストールされた Red Hat Enterprise Linux 7 をホストとして使用することができます。Red Hat Virtualization は、Red Hat Enterprise Linux 7 Server AMD64/Intel 64 版の Intel VT または AMD-V 拡張を実行するホストをサポートします。Red Hat Enterprise Linux マシンをホストとして使用するには、**Red Hat Enterprise Linux Server** および **Red Hat Virtualization** のサブスクリプションもアタッチする必要があります。

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、およびブリッジの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。詳細ビューを使用して、ホストと管理システムが接続を確立する際のプロセスを監視します。

オプションで、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 をインストールできます。Cockpit ウェブインターフェイスは、ネットワークの設定や、セルフホストエンジンのデプロイ、**Terminal**サブタブを介したターミナルコマンドの実行など、ホストが Red Hat Virtualization

Manager に追加される前に実行されるタスクのためのグラフィカルユーザーインターフェイスを提供します。

重要

サードパーティーのウォッチドッグは、VDSM が提供するウォッチドッグデーモンに干渉する可能性があるため、Red Hat Enterprise Linux ホストにはインストールしないでください。

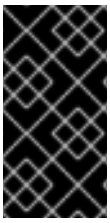
10.4. サテライトホストプロバイダーホスト

Satellite ホストプロバイダーによって提供されたホストは、Red Hat Virtualization Manager によって仮想化ホストとして使用することもできます。Satellite ホストプロバイダーが外部プロバイダーとして Manager に追加されると、そのプロバイダーが提供するホストは Red Hat Virtualization Hosts (RHVH) や Red Hat Enterprise Linux ホストと同じ方法で Red Hat Virtualization に追加して使用することができます。

10.5. ホストのタスク

10.5.1. Red Hat Virtualization Manager への通常のホストの追加

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、およびブリッジの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。



重要

静的 IPv6 アドレスを使用する管理ブリッジを作成する場合は、ホストを追加する前に、インターフェイス設定 (ifcfg) ファイルでネットワークマネージャーコントロールを無効にしてください。詳細は、https://access.redhat.com/solutions/3981311 を参照してください。

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. オプションとして、**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する。
- ホストの SSH フィンガープリントを追加し、セキュリティーを強化する。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には、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、**Administration Guide** の Host Power Management Settings Explained 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。

10.5.2. サテライトホストプロバイダーホストの追加

Satellite ホストプロバイダーのホストを追加するプロセスは、マネージャーでホストを識別する方法を除いて、Red Hat Enterprise Linux ホストを追加するプロセスとほぼ同じです。以下の手順では、サテライトホストプロバイダーが提供するホストを追加する方法について説明します。

サテライトホストプロバイダーホストの追加

1. **Compute → Hosts** をクリックします。
2. **New** をクリックします。
3. ドロップダウンメニューを使って、新しいホストの**ホストクラスター**を選択します。
4. **Foreman/Satellite** チェックボックスを選択すると、Satellite ホストプロバイダーのホストを追加するためのオプションが表示され、ホストを追加するプロバイダーを選択することができます。
5. **Discovered Hosts** または **Provisioned Hosts** のいずれかを選択します。
   - **Discovered Hosts** (デフォルトオプション) を選択します。ドロップダウンリストから、ホスト、ホストグループ、コンピュートリソースを選択します。
   - **Provisioned Hosts (プロビジョニングされたホスト)**。**Providers Hosts** ドロップダウンリストからホストを選択します。
     外部プロバイダーから取得できるホストに関する詳細は自動的に設定され、必要に応じて編集できます。
6. 新しいホストの **名前** と **SSH Port** (Provisioned Hosts のみ) を入力します。
7. ホストで使用する認証方法を選択します。
   - 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。
   - **SSH 公開鍵** 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 (プロビジョニングされたホストのみ)。
8. これで、Red Hat Enterprise Linux ホストを追加するための必須手順が完了しました。**Advanced Parameters** ドロップダウンボタンをクリックすると、ホストの詳細設定が表示されます。
   a. オプションでファイアウォールの自動設定を無効にします

b. 必要の応じてホストの SSH フィンガープリントを追加し、セキュリティーを強化する。手動での追加または自動取得が可能です。

9. **Power Management**、**SPM**、**Console**、**Network Provider** は、現在該当するタブを使用して設定することができますが、これらは Red Hat Enterprise Linux ホストを追加する上で基本的ではないため、この手順では説明しません。

10. **OK** をクリックすると、ホストが追加され、ウィンドウが閉じます。

新規ホストが **Installing** のステータスでホスト一覧に表示され、詳細ビューでインストールの進捗を表示できます。インストールが完了すると、ステータスが **Reboot** に更新されます。ステータスが **Up** に変わるには、ホストが起動している必要があります。

10.5.3. ホストの Satellite エラータ管理の設定

Red Hat Virtualization は、Red Hat Satellite からエラータを表示するように設定できます。これにより、ホスト管理者は、ホスト設定の管理に使用されるのと同じダッシュボードで、利用可能なエラータとその重要性に関する更新を受け取ることができます。Red Hat Satellite の詳細は、Red Hat Satellite Documentation を参照してください。

Red Hat Virtualization 4.3 では、Red Hat Satellite 6.5 でのエラータ管理がサポートされます。

重要

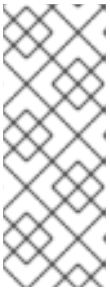
ホストは FQDN で Satellite サーバーで識別されます。IP アドレスを使用して追加したホストは、エラータを報告できません。これにより、Red Hat Virtualization で外部コンテンツホスト ID を維持する必要がなくなります。

ホストの管理に使用する Satellite アカウントにはは、Administrator 権限と、デフォルトの組織が設定されている必要があります。

ホストの Satellite エラータ管理の設定

1. 外部プロバイダーとして Satellite Server を追加します。詳細は、「ホストのプロビジョニング用の Red Hat Satellite インスタンスの追加」 を参照してください。

2. 必要なホストを Satellite サーバーに関連付けます。



注記

ホストは Satellite サーバーに登録し、そのホストに **katello-agent** パッケージをインストールしておく必要があります。

ホストの登録の設定方法、およびホストを登録し、katello-agent パッケージをインストールする方法は、Red Hat Satellite ドキュメント**Managing Hosts**の Registering Hosts を参照してください。

a. **Compute → Hosts** をクリックし、ホストを選択します。

b. **Edit** をクリックします。

c. **Use Foreman/Satellite**のチェックボックスを選択します。

d. ドロップダウンリストから、必要な Satellite サーバーを選択します。

e. **OK** をクリックします。

これで、ホストの設定を管理するための同じダッシュボードに、利用可能なエラッタとその重要性が表示されるようになりました。

10.5.4. 新規ホスト、ホスト編集ウィンドウの設定とコントロールの説明

10.5.5. ホストの一般設定の説明

これらの設定は、ホストの詳細を編集するとき、または新しい Red Hat Enterprise Linux ホストと Satellite ホストプロバイダーホストを追加するときに適用されます。

General 設定の表には、**New Host** または **Edit Host** ウィンドウの **General** タブで必要な情報が含まれています。

表10.1 一般設定

フィールド名	説明
Host Cluster	ホストが属するクラスターとデータセンター。

フィールド名	説明
Use Foreman/Satellite	このチェックボックスを選択またはクリアすると、サテライトホストプロバイダーが提供するホストを追加するためのオプションが表示または非表示になります。以下のオプションを利用できます。 **検出されたホスト** ● **Discovered Hosts (発見されたホスト)**: エンジンによって発見されたサテライトホストの名前が入力されたドロップダウンリストです。 ● **Host Groups**: 利用可能なホストグループのドロップダウンリストです。 ● **Compute Resources**: コンピュートリソースを提供するハイパーバイザのドロップダウンリストです。 **プロビジョニングされたホスト** ● **Providers Hosts**: 選択された外部プロバイダーが提供するホストの名前が入力されるドロップダウンリストです。このリストのエントリーは、**プロバイダー検索フィルター** に入力された検索クエリーに応じてフィルターリングされます。 ● **Provider search filter**: 選択された外部プロバイダーが提供するホストを検索するためのテキストフィールドです。このオプションはプロバイダー固有です。特定のプロバイダーの検索クエリーの作成の詳細については、プロバイダーのドキュメントを参照してください。利用可能なすべてのホストを表示するには、このフィールドを空白にします。
Name	ホストの名前。このテキストフィールドには 40 文字の制限があり、大文字、小文字、数字、ハイフン、およびアンダースコアの組み合わせが含まれる一意の名前である必要があります。
Comment	ホストに関するプレーンテキストで人間が読めるコメントを追加するためのフィールドです。
Hostname	ホストの IP アドレスまたは解決可能なホスト名。解決可能なホスト名を使用する場合は、すべてのアドレス (IPv4 および IPv6) で、ホストの管理ネットワークで使用される IP アドレス (IPv4 および IPv6) と一致するようにホスト名が解決されるように指定する必要があります。

フィールド名	説明
Password	ホストのルートユーザーのパスワードです。これは、ホストの追加時にのみ付与でき、後で編集することはできません。
SSH Public Key	テキストボックス内の内容をホスト上の **/root/.ssh/authorized_hosts** ファイルにコピーすることで、ホストとの認証にパスワードの代わりに Manager の SSH キーを使用することができます。
ホストのファイアウォールを自動的に設定	新しいホストを追加する際、マネージャーはホストのファイアウォールで必要なポートを開くことができます。これはデフォルトで有効になっています。これは **Advanced Parameter** です。
SSH Fingerprint	ホストの SSH フィンガープリントを**取得**し、ホストが返すと予想されるフィンガープリントと比較して、両者が一致することを確認することができます。これは **Advanced Parameter** です。

10.5.6. ホストパワーマネージメント設定の説明

Power Management 設定の表には、**新規ホスト**または**ホストの編集**ウィンドウの**Power Management** タブで必要な情報が含まれています。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には、電源管理を設定することができます。

表10.2 Power Management 設定

フィールド名	説明
電源管理の有効化	ホストの電源管理を有効にする。このチェックボックスを選択すると、**Power Management** タブの残りのフィールドが有効になります。
Kdump 統合	カーネルのクラッシュダンプの実行中にホストがフェンシングするのを防ぎ、クラッシュダンプが中断されないようにします。Red Hat Enterprise Linux 7.1 以降では、デフォルトで kdump が利用できます。kdump がホスト上で利用可能であっても、その設定が有効でない (kdump サービスが開始できない) 場合、**Kdump の統合**を有効にすると、ホストの (再) インストールが失敗します。この場合は、「fence_kdump の高度な設定」 を参照してください。

フィールド名	説明
電源管理のポリシー制御を無効にする	電源管理は、ホストの**クラスター**の**スケジューリングポリシー**によって制御されます。パワーマネージメントが有効で、定義された低使用率の値に達した場合、マネージャーはホストマシンをパワーダウンさせ、ロードバランシングが必要な場合や、クラスター内に十分な空きホストがない場合には、再びホストマシンを再起動させます。ポリシーコントロールを無効にする場合は、このチェックボックスを選択します。
順番待ちのエージェント	ホストのフェンスエージェントを一覧表示します。フェンスエージェントには、シーケンシャル、コンカレント、またはその両方の組み合わせがあります。 ● フェンスエージェントが順次使用される場合、ホストの停止または起動にはまずプライマリーエージェントが使用され、それが失敗した場合にはセカンダリーエージェントが使用されます。 ● フェンスエージェントを同時に使用する場合、両方のフェンスエージェントが Stop コマンドに反応しなければホストは停止しませんが、一方のエージェントが Start コマンドに反応すればホストは立ち上がります。 フェンスエージェントはデフォルトではシーケンシャルです。上下のボタンでフェンス剤の使用順序を変更できます。 2 つのフェンスエージェントを同時進行させるには、一方のフェンスエージェントをもう一方のフェンスエージェントの隣にある **Concurrent with** ドロップダウンリストから選択します。追加のフェンスエージェントの横にある **Concurrent with** ドロップダウンリストからグループを選択することで、同時進行のフェンスエージェントのグループに追加することができます。
フェンスエージェントの追加	+ ボタンをクリックして、新しい接続を追加します。**フェンスエージェントの編集** ウィンドウが開きます。このウィンドウのフィールドの詳細は、以下の表を参照してください。
電源管理プロキシープリファレンス	デフォルトでは、Manager がホストと同じ**クラスター** 内のフェンシングプロキシーを検索し、フェンシングプロキシーが見つからない場合は、同じ **DC** (データセンター) 内を検索するよう指定します。上下のボタンで、これらのリソースの使用順序を変更することができます。このフィールドは、**Advanced Parameters** で利用できます。

次の表は、**Edit fence agent** ウィンドウで必要な情報です。

表10.3 フェンスエージェントの編集 の設定

フィールド名	説明
Address	ホストの電源管理デバイスにアクセスするためのアドレス。解決可能なホスト名または IP アドレスのいずれか。
User Name	電源管理デバイスにアクセスするユーザーアカウント。デバイスにユーザーを設定するか、デフォルトのユーザーを使用します。
Password	電源管理デバイスにアクセスするユーザーのパスワード。
タイプ	ホストの電源管理デバイスのタイプ。以下のいずれかを選択します。 ● **apc**:APC MasterSwitch ネットワーク電源スイッチ。APC 5.x 電源スイッチデバイスとは使用しないでください。 ● **apc_snmp** - APC 5.x 電源スイッチデバイスとは使用しないでください。 ● **bladecenter**: IBM Bladecenter リモートスーパバイザアダプター。 ● **cisco_ucs**: Cisco United Computing System ● **drac5**: Dell コンピューター用の Dell Remote Access Controller です。 ● **drac7**: Dell コンピューター用の Dell Remote Access Controller です。 ● **eps**: ePower Switch 8M+ ネットワークパワースイッチ。 ● **hpblade** - HP BladeSystem. ● **ILO、ILO2、ILO3、ILO4**: HP Integrated Lights-Out。 ● **ipmilan**: Intelligent Platform Management Interface と Sun Integrated Lights Out Management デバイス。 ● **rsa**: IBM リモートスーパーバイザーアダプター。 ● **rsb**: 富士通シーメンスの RSB 管理インターフェイスです。 ● **wti**: WTI ネットワークパワースイッチ。 電源管理デバイスの詳細は、**Technical Reference** の Power Management を参照してください。

フィールド名	説明
ポート	電源管理デバイスがホストとの通信に使用するポート番号。
Slot	電源管理デバイスのブレードを識別するための番号。
Service Profile	電源管理デバイスのブレードを識別するために使用されるサービスプロファイル名です。デバイスタイプが **cisco_ucs** の場合、Slot の代わりにこのフィールドが表示されます。

フィールド名	説明
オプション	電源管理デバイス固有のオプション。これらを key=value として入力します。利用可能なオプションは、お使いのホストの電源管理デバイスのドキュメントを参照してください。 Red Hat Enterprise Linux 7 ホストで、電源管理デバイスとして cisco_ucs を使用している場合は、**Options** フィールドに **ssl_insecure=1** を追加する必要があります。
Secure	電源管理デバイスがホストに安全に接続できるようにするには、このチェックボックスを選択します。これは、電源管理エージェントに応じて、ssh、ssl、または他の認証プロトコルを介して行うことができます。

10.5.7. SPM のプライオリティー設定の説明

SPM 設定の表には、**New Host** または **Edit Host** ウィンドウの **SPM** タブに必要な情報の詳細が記載されています。

表10.4 SPM の設定

フィールド名	説明
SPM の優先度	ホストにストレージプールマネージャー (SPM) のロールが与えられる可能性を定義します。オプションは、**Low**、**Normal**、**High**の 3 つです。優先度が低いとは、ホストに SPM のロールが割り当てられる可能性が低いことを意味し、優先度が高いとは、その可能性が高いことを意味しています。デフォルト設定は Normal です。

10.5.8. ホストクラスター設定の説明

Console 設定の表には、**New Host** または **Edit Host** ウィンドウの **Console** タブに必要な情報の詳細が記載されています。

表10.5 コンソール の設定

フィールド名	説明

フィールド名	説明
表示アドレスの上書き	ホストの表示アドレスを上書きする場合は、このチェックボックスを選択します。この機能は、ホストが内部 IP で定義されており、NAT ファイアウォールの内側にある場合に有効です。ユーザーが内部ネットワークの外から仮想マシンに接続した場合、仮想マシンが動作しているホストのプライベートアドレスを返すのではなく、パブリック IP または FQDN(外部ネットワークではパブリック IP に解決される) を返します。
表示アドレス	ここで指定した表示アドレスは、このホスト上で動作するすべての仮想マシンに使用されます。アドレスは、完全修飾ドメイン名または IP の形式でなければなりません。

10.5.9. ネットワークプロバイダー設定の説明

ネットワークプロバイダー の設定の表には、**New Host** または **Edit Host** ウィンドウの**Network Provider** タブで必要な情報の詳細が記載されています。

表10.6 Network Provider設定

フィールド名	説明
外部ネットワークプロバイダー	外部ネットワークプロバイダーを追加し、ホストのネットワークを外部ネットワークプロバイダーによってプロビジョニングする場合は、リストから選択します。

10.5.10. カーネル設定の説明

Kernel 設定の表には、**New Host** または **Edit Host** ウィンドウの **Kernel** タブに必要な情報の詳細が記載されています。一般的なカーネルブートパラメーターのオプションはチェックボックスで表示されるので、簡単に選択することができます。

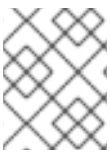
より複雑な変更を行う場合は、**Kernel command line** の横にあるフリーテキスト入力フィールドを使用して、必要な追加パラメーターを追加します。カーネルのコマンドラインパラメーターを変更した場合は、ホストを再インストール する必要があります。

重要

ホストが Manager に接続されている場合、変更する前にホストをメンテナンスモードにする必要があります。変更後に、ホストを再インストールして 変更を適用します。

表10.7 カーネル 設定

フィールド名	説明
Hostdev パススルーおよび SR-IOV	カーネルの IOMMU フラグを有効にして、デバイスが仮想マシン自体に直接アタッチされているデバイスであるかのように、ホストデバイスを仮想マシンで使用できるようにします。また、ホストのハードウェアとファームウェアも IOMMU に対応している必要があります。ハードウェア上で仮想化拡張機能と IOMMU 拡張機能が有効になっている必要があります。Configuring a Host for PCI Passthroughを参照してください。IBM POWER8 では、デフォルトで IOMMU が有効になっています。
Nested Virtualization	vmx フラグまたは svm フラグを有効にして、仮想マシン内で仮想マシンを実行できるようにします。このオプションは評価目的のみで、実稼働での使用はサポートされません。**vdsm-hook-nestedvt** フックはホストにインストールされている必要があります。
安全でない割り込み	IOMMU が有効になっているが、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は、このオプションを有効にすることを検討できます。このオプションは、ホスト上の仮想マシンが信頼できる場合にのみ有効にしてください。このオプションを有効にすると、仮想マシンからの MSI 攻撃を受ける可能性があります。このオプションは、評価目的で認定されていないハードウェアを使用する場合のみ、回避策として使用することを目的としています。
PCI 再割り当て	メモリーの問題で SR-IOV NIC が仮想機能を割り当てられない場合は、このオプションを有効にすることを検討してください。また、ホストのハードウェアとファームウェアが PCI の再配置をサポートしている必要があります。このオプションは、評価目的で認定されていないハードウェアを使用する場合のみ、回避策として使用することを目的としています。
カーネルコマンドライン	このフィールドでは、デフォルトのパラメーターにさらにカーネルパラメーターを追加することができます。



注記

カーネルブートパラメーターがグレーアウトしている場合は、**reset**ボタンをクリックすると、オプションが利用可能になります。

10.5.11. ホストエンジン設定の説明

Hosted Engine settings 表は、New Host または Edit Host ウィンドウの Hosted Engine タブで必要な情報の詳細を示します。

表10.8 Hosted Engine 設定

フィールド名	説明
ホストされたエンジンの展開方法を選択	利用可能な 3 つのオプションは以下のとおりです。 • **None**: 必要なアクションはありません。 • **Deploy**: ホストをセルフホスト型のエンジンノードとしてデプロイする場合は、このオプションを選択します。 • **Undeploy**: セルフホスト型エンジンノードの場合、このオプションを選択すると、ホストがアンデプロイされ、セルフホスト型エンジン関連の設定が削除されます。

10.5.12. ホストパワーマネージメントの設定

管理ポータルからホストのライフサイクル操作 (停止、開始、再起動) を行うために、ホストパワーマネージメントデバイスの設定を行います。

ホストの高可用性や仮想マシンの高可用性を利用するためには、ホストのパワーマネジメントを設定する必要があります。電源管理デバイスの詳細は、**Technical Reference** の Power Management を参照してください。

電源管理状態の設定

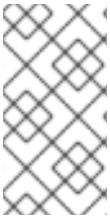
1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **管理 → メンテナーンス** をクリックし、**OK** をクリックして確定します。
3. ホストがメンテナンスモードになったら、**Remove** をクリックします。
4. **Power Management** タブをクリックします。
5. **パワーマネジメントを有効にする**チェックボックスを選択し、フィールドを有効にします。
6. **Kdump integration** チェックボックスを選択すると、カーネルクラッシュダンプの実行中にホストがフェンシングするのを防ぐことができます。

重要

既存のホストで**Kdump の統合** を有効または無効にした場合、kdump を設定するには ホストを再インストール する必要があります。

7. オプションとして、ホストの電源管理をホストの**クラスター**の**スケジューリングポリシー**で制御したくない場合は、**Disable policy control of power management**チェックボックスを選択します。
8. プラス + ボタンをクリックして、新しいパワーマネージメントデバイスを追加します。**フェンスエージェントの編集**ウィンドウが開きます。

9. パワーマネージメントデバイスの**ユーザー名**と**パスワード**を適切なフィールドに入力します。
10. ドロップダウンリストからパワーマネージメントデバイスの**種類**を選択します。
11. **アドレス**フィールドに IP アドレスを入力します。
12. 電源管理装置がホストとの通信に使用する **SSH ポート** 番号を入力します。
13. パワーマネージメントデバイスのブレードを識別するための **Slot** 番号を入力してください。
14. パワーマネージメントデバイスの**オプション**を入力します。**key=value** ペアのコンマ区切りリストを使用します。
    - IPv4 と IPv6 の両方の IP アドレスを使用できる場合 (デフォルト) は、**Options** フィールドを空白にします。
    - IPv4 の IP アドレスのみを使用する場合は、**inet4_only=1** を入力してください。
    - IPv6 の IP アドレスのみを使用する場合は、**inet6_only=1** を入力してください。
15. パワーマネージメントデバイスがホストに安全に接続できるようにするには、**Secure** チェックボックスを選択します。
16. **テスト** をクリックして、設定が正しいことを確認します。検証に成功すると **Test Succeeded、Host Status is: on** と表示されます。
17. **OK** をクリックして、**Edit fence agent** ウィンドウを閉じます。
18. **電源管理** タブで、オプションで **詳細パラメーター** を展開し、上下のボタンを使用して、マネージャーがホストの **クラスター** と **DC** (データセンター) でフェンシングプロキシーを検索する順序を指定します。
19. **OK** をクリックします。



注記

- IPv6 については、Red Hat Virtualization でサポートされるのは静的なアドレスだけです。
- **IPv4 と IPv6**のデュアルスタックアドレッシングはサポートされていません。

Management → Power Management ドロップダウン メニューは、管理者ポータルで有効化されています。

10.5.13. ホストストレージプールマネージャーの設定

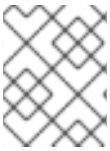
ストレージプールマネージャー (SPM) は、データセンター内のホストの 1 つに与えられた管理者のロールで、ストレージドメインへのアクセス制御を維持します。SPM は常に利用可能でなければならず、SPM ホストが利用できなくなった場合、SPM ロールは別のホストに割り当てられます。SPM ロールはホストの利用可能なリソースの一部を使用するため、リソースに余裕のあるホストを優先的に使用することが重要です。

ホストの SPM (Storage Pool Manager) 優先度の設定により、ホストが SPM ロールが割り当てられる可能性があります。SPM 優先度が高いホストには、SPM の優先度が低いホストの前に SPM ロールが割り当てられます。

SPM 設定の設定

1. **Compute → Hosts** をクリックします。
2. **Edit** をクリックします。
3. **SPM** タブをクリックします。
4. ラジオボタンで、ホストに適した SPM の優先順位を選択します。
5. **OK** をクリックします。

10.5.14. 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 を準備および設定する方法を示す一連のトピックの 1 つです。詳細は、Setting Up and Configuring SR-IOV 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することができます。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には、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

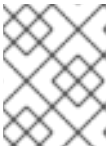
- ホストハードウェアが PCI デバイスパススルーおよび割り当ての要件を満たしていることを確認してください。詳細は、PCI Device Requirements を参照してください。

PCI パススルーを有効にするためのホストの設定

1. BIOS の仮想化拡張機能および IOMMU 拡張機能を有効化してください。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、Adding Standard Hosts to the Red Hat Virtualization Manager および Kernel Settings Explained 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、**Setting up an NVIDIA GPU for a virtual machine in Red Hat Virtualization** の GPU device passthrough: Assigning a host GPU to a single virtual machine 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、IOMMU がデフォルトで有効化されているので、この手順は飛ばしてください。

- 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=on** が機能する場合は、**iommu=pt** または **amd_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 のみが有効化されて、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は、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、ホストは仮想マシンからの MSI 攻撃に晒されることになるため、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、以下のように設定してください。

```
# 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
```

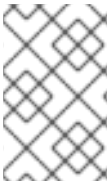
```
# reboot
```

SR-IOV を有効にして専用の仮想 NIC を仮想マシンに割り当てるには、https://access.redhat.com/articles/2335291 を参照してください。

10.5.15. ホストのメンテナンスモードへの切り替え

ネットワークの設定やソフトウェアの更新など、一般的なメンテナンス作業では、ホストをメンテナンスモードにする必要があります。ホストは、再起動、ネットワークやストレージの問題など、VDSM が正常に動作しなくなる可能性があるイベントが発生する前に、メンテナンスモードにする必要があります。

ホストがメンテナンスモードになると、Red Hat Virtualization Manager は実行中のすべての仮想マシンを代替ホストに移行しようとします。ライブマイグレーションの標準的な前提条件が適用されます。特に、移行した仮想マシンを実行する能力を持つアクティブなホストが、クラスター内に少なくとも 1 つ存在する必要があります。



注記

ホストに固定されていて移行できない仮想マシンはシャットダウンされます。どの仮想マシンがホストに固定されているかは、ホストの詳細表示の**仮想マシン**タブで**Pinned to Host**をクリックすることで確認できます。

ホストのメンテナンスモードへの移行

1. **コンピュート → ホスト** をクリックし、任意のホストを選択します。
2. **管理 → メンテナーンス** をクリックして、**メンテナーンスホスト** の確認ウィンドウを開きます。
3. オプションで、ホストをメンテナンスモードに移行させる**理由**を入力します。この理由は、ログに表示され、ホストが再びアクティブになったときに表示されます。

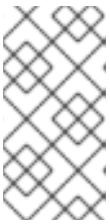
   

   注記

   ホストメンテナンス**理由**フィールドは、クラスター設定で有効になっている場合にのみ表示されます。詳細は、「一般的なクラスター設定に関する説明」 を参照してください。

4. オプションで、Gluster をサポートするホストに必要なオプションを選択します。
   デフォルトのチェックを回避するには、**Gluster クォーラムと自己修復の検証を無視する** オプションを選択します。デフォルトでは、ホストがメンテナンスモードに移行したときに、Gluster クォーラムが失われないように Manager がチェックします。また Manager は、ホストをメンテナンスモードに移行することで影響を受ける自己修復活動がないことを確認します。Gluster のクォーラムが失われる場合や、自己修復活動が影響を受ける場合、Manager はホストがメンテナンスモードになるのを防ぎます。このオプションは、ホストをメンテナンスモードにする他の方法がない場合にのみ使用してください。

   ホストをメンテナンスモードに移行する際、すべての Gluster サービスを停止するには、**Stop Gluster Service** オプションを選択します。

   注記

   これらのフィールドは、選択したホストが Gluster に対応している場合にのみ、ホストのメンテナンスウィンドウに表示されます。詳細は、**Maintaining Red Hat Hyperconverged Infrastructure** の Replacing the Primary Gluster Storage Node を参照してください。

5. **OK** をクリックしてメンテナーンスモードを開始します。

稼働中の仮想マシンはすべて代替ホストに移行されます。ホストが Storage Pool Manager (SPM) の場

合、SPM のロールは別のホストに移行されます。ホストの **Status** フィールドが **Preparing for Maintenance** に変わり、操作が正常に完了すると最終的に **Maintenance** となります。ホストがメンテナンスモードになっても、VDSM は停止しません。

注記

いずれかの仮想マシンで移行が失敗した場合は、ホストで **Management → Activate** をクリックして操作を停止し、メンテナンスモードにしてから、仮想マシンで **Cancel Migration** をクリックして移行を停止します。

10.5.16. メンテナンスモードからホストを起動する

メンテナンスモードになったホストや、最近環境に追加されたホストは、使用する前にアクティベートする必要があります。ホストの準備ができていないと、アクティベーションに失敗することがあります。ホストのアクティベーションを試みる前に、すべてのタスクが完了していることを確認してください。

メンテナンスモードからホストを起動する

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **管理 → アクティブ化** をクリックします。

操作が完了すると、ホストの状態は **Unassigned** に変わり、最後に **Up** となります。仮想マシンがホスト上で動作するようになりました。メンテナンスモード時にホストから移行された仮想マシンは、ホストが起動したときに自動的には戻ってきませんが、手動で移行することができます。メンテナンスモードに移行する前にホストがストレージプールマネージャー (SPM) であった場合、ホストがアクティブになっても SPM のロールは自動的には戻りません。

10.5.17. ホストファイアウォールルールの設定

ホストのファイアウォールルールは、Ansible を使って永続的になるように設定することができます。クラスターは、**iptables** ではなく **firewalld** を使用するように設定する必要があります。

注記

iptables は非推奨になりました。

ホストに対するファイアウォールルールの設定

1. Manager マシン上で、**ovirt-host-deploy-post-tasks.yml.example**を編集し、カスタムファイアウォールポートを追加します。

```
# vi /etc/ovirt-engine/ansible/ovirt-host-deploy-post-tasks.yml.example
---
#
# Any additional tasks required to be executing during host deploy process can
# be added below
#
- name: Enable additional port on firewalld
  firewalld:
    port: "12345/tcp"
    permanent: yes
    immediate: yes
    state: enabled
```

2. ファイルを別の場所に **ovirt-host-deploy-post-tasks.yml** として保存します。

新規ホストまたは再インストールされたホストは、更新されたファイアウォールルールで設定されます。

Installation → Reinstall をクリックし、**Automatically configure host firewall** を選択して、既存のホストを再インストールする必要があります。

10.5.18. ホストの削除

仮想化環境からホストを削除します。

ホストの削除

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **管理 → メンテナンス** をクリックします。
3. ホストがメンテナーンスモードにある場合は、**削除** をクリックして、**ホストの削除** の確認ウィンドウを開きます。
4. ホストが Red Hat Gluster Storage クラスターに含まれており、ボリュームブリックがある場合や、ホストが応答していない場合は、**Force Remove** のチェックボックスを選択します。
5. **OK** をクリックします。

10.5.19. マイナーリリース間でのホストの更新

クラスター内のすべてのホスト を更新したり、個別のホスト を更新したりできます。

10.5.19.1. クラスター内の全ホストの更新

ホストを個別に更新するのではなく、クラスター内の全ホストを更新することができます。この手法は、Red Hat Virtualization を新しいバージョンにアップグレードする際に特に役立ちます。更新の自動化に使用する Ansible ロールに関する詳細情報は、https://github.com/oVirt/ovirt-ansible-cluster-upgrade/blob/master/README.md を参照してください。

Red Hat は、一度に 1 つのクラスターを更新することをお勧めします。

制限事項

- RHVH の更新時には、**/etc** および **/var** ディレクトリーに変更されたコンテンツのみを保持します。他のパスに含まれる変更されたデータは更新時に上書きされます。
- クラスターレベリで移行が有効化されている場合には、仮想マシンはそのクラスター内の別のホストに自動的に移行されます。
- セルフホスト型エンジン環境では、Manager 用仮想マシンは同一クラスター内のセルフホスト型エンジンノード間でのみ移行が可能です。通常のホストに移行することはできません。
- ホストが属するクラスターには、ホストがメンテナンスを実行するのに十分なメモリーが確保されている必要があります。確保されていないと、仮想マシンの移行がハングして失敗してしまいます。ホストを更新する前に一部またはすべての仮想マシンをシャットダウンしておくと、ホスト更新によるメモリー使用量を低減することができます。

- ホストに固定された仮想マシン (vGPU を使用している仮想マシンなど) を別のホストに移行することはできません。ホストの更新をスキップしない限り、そのホストに固定された仮想マシンは更新中にシャットダウンされます。

手順

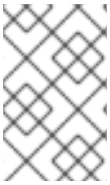
1. 管理ポータルで **コンピュート → クラスター** をクリックし、クラスターを選択します。
2. **アップグレード** をクリックします。
3. 更新するホストを選択し、次に **Next** をクリックします。
4. オプションを設定します。
   - **固定された仮想マシンの停止**: クラスター内のホストに固定された仮想マシンをシャットダウンします。このオプションは、デフォルトで選択されています。このチェックボックスの選択を解除すると、固定された仮想マシンが動作を続けられるように、それらのホストの更新をスキップすることができます (固定された仮想マシンが重要なサービスまたはプロセスを実行中で、更新中の予期せぬ時にシャットダウンされるのを避けたい場合など)。
   - **Upgrade Timeout (Minutes)**: このオプションで設定した時間内に個々のホストの更新が完了しない場合には、クラスターのアップグレードはタイムアウトで失敗します。デフォルトは **60** です。60 分では不十分と思われる大規模なクラスターの場合には、時間を延長することができます。また、ホストの更新が短時間で完了する小規模なクラスターでは、短縮することができます。
   - **Check Upgrade**: アップグレードプロセスを実行する前に、それぞれのホストで更新が利用可能かどうかを確認します。このオプションは、デフォルトでは選択されていません。ただし、Manager がホストの更新を確認する頻度をデフォルトより低く設定している状況などで、最新の更新を確実に含める必要がある場合には、このオプションを選択することができます。
   - **Reboot After Upgrade**: ホストの更新後に、それぞれのホストを再起動します。このオプションは、デフォルトで選択されています。ホストの再起動を必要とする保留中の更新がないことが明らかであれば、このチェックボックスの選択を解除してプロセスを迅速化することができます。
   - **Use Maintenance Policy**: 更新時のクラスターのスケジューリングポリシーを **cluster_maintenance** に設定します。このオプションはデフォルトで選択されています。したがって、許可される動作は限定的で、仮想マシンは高可用性でない限り起動することができません。更新中も使用を続けたいカスタムのスケジューリングポリシーがある場合には、このチェックボックスの選択を解除することができます。ただし、これにより想定外の結果を招く可能性があります。このオプションを無効にする前に、カスタムのポリシーがクラスターのアップグレード操作に対応していることを確認してください。
5. **Next** をクリックします。
6. 影響を受けるホストおよび仮想マシンの概要を確認します。
7. **アップグレード** をクリックします。

コンピュート → ホスト ビューおよび **通知ドロワー** () の **イベント** セクションで、ホスト更新の進捗を追跡することができます。

仮想マシン移行の進捗を、**コンピュート → 仮想マシン** ビューの **ステータス** 列で個々に追跡することができます。大規模な環境では、特定の仮想マシングループの結果を表示するために、結果を絞り込まなければならない場合があります。

10.5.19.2. 個々のホストの更新

ホストのアップグレードマネージャーを使用して、管理ポータルから直接個々のホストを更新します。



注記

アップグレードマネージャーが確認するのは、ステータスが **Up** または **Non-operational** のホストだけです。ステータスが **Maintenance** のホストは確認されません。

制限事項

- RHVH の更新時には、**/etc** および **/var** ディレクトリーに変更されたコンテンツのみを保持します。他のパスに含まれる変更されたデータは更新時に上書きされます。
- クラスターレベリで移行が有効化されている場合には、仮想マシンはそのクラスター内の別のホストに自動的に移行されます。使用率が比較的に低い時間帯にホストを更新してください。
- セルフホスト型エンジン環境では、Manager 用仮想マシンは同一クラスター内のセルフホスト型エンジンノード間でのみ移行が可能です。通常のホストに移行することはできません。
- ホストが属するクラスターには、ホストがメンテナンスを実行するのに十分なメモリーが確保されている必要があります。確保されていないと、仮想マシンの移行がハングして失敗してしまいます。ホストを更新する前に一部またはすべての仮想マシンをシャットダウンしておくと、ホスト更新によるメモリー使用量を低減することができます。
- すべてのホストを同時に更新しないでください。Storage Pool Manager (SPM) のタスクを実行するために、1 台のホストは使用可能な状態でなければなりません。
- ホストに固定された仮想マシン (vGPU を使用している仮想マシンなど) を別のホストに移行することはできません。ホストを更新する前に、固定された仮想マシンをシャットダウンする必要があります。

手順

1. 適切なリポジトリーが有効であることを確認します。現在有効なリポジトリーの一覧を表示するには、**yum repolist** を実行します。
   - Red Hat Virtualization Host の場合:

     ```
     # subscription-manager repos --enable=rhel-7-server-rhvh-4-rpms
     ```

   - Red Hat Enterprise Linux ホストの場合:

     ```
     # subscription-manager repos \
         --enable=rhel-7-server-rpms \
         --enable=rhel-7-server-rhv-4-mgmt-agent-rpms \
         --enable=rhel-7-server-ansible-2.9-rpms
     ```

2. 管理ポータルで **コンピュート → ホスト** をクリックし、更新するホストを選択します。
3. **インストール → アップグレードを確認** をクリックしてから **OK** をクリックします。
   通知ドロワー () を開き、**イベント** セクションを展開して結果を表示します。
4. 更新が利用可能であれば、**インストール → アップグレード** をクリックします。

5. **OK** をクリックしてホストを更新します。実行中の仮想マシンは、その移行ポリシーに従って移行されます。いずれかの仮想マシンの移行が無効になっている場合は、シャットダウンするよう求められます。
コンピュート → ホスト にホストの情報が更新され、ステータスが以下の順序で変わります。

   Maintenance > Installing > Reboot > Up

注記

更新が失敗すると、ホストのステータスは **Install Failed** に変わります。**Install Failed** のステータスから **インストール → アップグレード** を再度クリックすることができます。

Red Hat Virtualization 環境内のホストごとに同じ手順を繰り返してください。

Red Hat は、管理ポータルからホストを更新することをお勧めします。ただし、管理ポータルの代わりに **yum update** を使用してホストを更新することもできます。

10.5.19.3. ホストの手動更新

yum コマンドを使用してホストを更新できます。セキュリティーやバグに関する修正がタイムリーに適用されるように、システムを定期的に更新してください。

制限事項

- RHVH の更新時には、**/etc** および **/var** ディレクトリーに変更されたコンテンツのみを保持します。他のパスに含まれる変更されたデータは更新時に上書きされます。
- クラスターレベリで移行が有効化されている場合には、仮想マシンはそのクラスター内の別のホストに自動的に移行されます。使用率が比較的に低い時間帯にホストを更新してください。
- セルフホスト型エンジン環境では、Manager 用仮想マシンは同一クラスター内のセルフホスト型エンジンノード間でのみ移行が可能です。通常のホストに移行することはできません。
- ホストが属するクラスターには、ホストがメンテナンスを実行するのに十分なメモリーが確保されている必要があります。確保されていないと、仮想マシンの移行がハングして失敗してしまいます。ホストを更新する前に一部またはすべての仮想マシンをシャットダウンしておくと、ホスト更新によるメモリー使用量を低減することができます。
- すべてのホストを同時に更新しないでください。Storage Pool Manager (SPM) のタスクを実行するために、1 台のホストは使用可能な状態でなければなりません。
- ホストに固定された仮想マシン (vGPU を使用している仮想マシンなど) を別のホストに移行することはできません。ホストを更新する前に、固定された仮想マシンをシャットダウンする必要があります。

手順

1. 適切なリポジトリーが有効であることを確認します。**yum repolist** を実行して、現在有効なリポジトリーを確認することができます。
   - Red Hat Virtualization Host の場合:

     ```
     # subscription-manager repos --enable=rhel-7-server-rhvh-4-rpms
     ```

- Red Hat Enterprise Linux ホストの場合:

```
# subscription-manager repos \
    --enable=rhel-7-server-rpms \
    --enable=rhel-7-server-rhv-4-mgmt-agent-rpms \
    --enable=rhel-7-server-ansible-2.9-rpms
```

2. 管理ポータルで **コンピュート → ホスト** をクリックし、更新するホストを選択します。
3. **管理 → メンテナーンス** をクリックします。
4. ホストを更新します。

```
# yum update
```

5. すべての更新が正常に適用されるように、ホストを再起動します。

注記

imgbased ログを確認して、Red Hat Virtualization Host 向けの追加パッケージの更新に失敗したものがないかを確認します。更新後に一部のパッケージの再インストールに失敗した場合には、そのパッケージが **/var/imgbased/persisted-rpms** に記載されていることを確認します。足りないパッケージを追加してから **rpm -Uvh /var/imgbased/persisted-rpms/*** を実行します。

Red Hat Virtualization 環境内のホストごとに同じ手順を繰り返してください。

10.5.20. ホストの再インストール

管理ポータルから Red Hat Virtualization Host (RHVH) および Red Hat Enterprise Linux ホストを再インストールします。この手順には、ホストの停止および再起動が含まれます。

前提条件

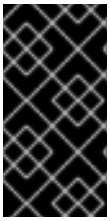
- 移行がクラスターレベルで有効にされる場合に、仮想マシンはクラスター内の別のホストに自動的に移行されので、ホストの使用が比較的低くなる場合にはホストを再インストールことをおすすめします。
- ホストによるメンテナーンス実行に十分なメモリーがクラスターに予約されていることを確認します。クラスターに十分なメモリーがない場合には、仮想マシンの移行操作がハングしてから失敗します。一部またはすべての仮想マシンをシャットダウンしてから、ホストをメンテナーンスに移行すると、この操作のメモリー使用量を減らすことができます。
- 再インストールを実行する前に、クラスターに複数のホストが含まれていることを確認してください。Storage Pool Manager (SPM) のタスクを実行するには、1 台のホストは使用可能な状態でなければならないので、すべてのホストを同時に再インストールしないでください。

手順

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **管理 → メンテナンス** をクリックします。
3. **Installation → Reinstall** をクリックして、**Install Host** ウィンドウを開きます。

4. **OK** をクリックして、ホストを再インストールします。

正常に再インストールされると、ホストのステータスが **Up** と表示されます。ホストから移行された仮想マシンはすべて、ホストに移行できるようになりました。



重要

Red Hat Virtualization Host が Red Hat Virtualization Manager に正常に登録されてから再インストールされると、ステータスが **Install Failed** の状態で、管理ポータルに誤って表示される可能性があります。**Management → アクティブ化** をクリックすると、ホストが **Up** ステータスに変わり、使用できるようになります。

10.5.21. ホスト Errata の表示

各ホストの Errata は、Red Hat Satellite サーバーから Errata 情報を受信するようにホストが設定された後に表示されます。ホストを設定してエラータ情報を取得する方法の詳細は、「ホストの Satellite エラータ管理の設定」 を参照してください。

ホスト Errata の表示

1. **Compute → Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **Errata** タブをクリックします。

10.5.22. ホストのヘルスステータスの表示

ホストは、通常の **Status** に加えて、外部の Health Status を持っています。外部の健康状態は、プラグインや外部システムから報告されたり、管理者が設定したりするもので、ホストの **Name** の左側に以下のいずれかのアイコンとして表示されます。

- **OK**: アイコンなし
- **info**: 
- **Warning**: 
- **Error**: 
- **Failure**: 

ホストのヘルスステータスの詳細を表示するには、ホスト名をクリックして詳細ビューを開き、**Events** タブをクリックします。

また、REST API を使ってホストの健康状態を確認することもできます。ホストの **GET** リクエストには、ヘルスステータスを含む **external_status** 要素が含まれます。

REST API では、**events** コレクションを介してホストの健康状態を設定できます。詳細は、**REST API Guide** の Adding Events を参照してください。

10.5.23. ホストデバイスの表示

各ホストのホストデバイスは、詳細ビューの **Host Devices** タブで確認できます。ホストにデバイスの直接割り当てが設定されている場合、これらのデバイスを仮想マシンに直接接続してパフォーマンスを向上させることができます。

デバイスの直接割り当てに関するハードウェア要件の詳細は、**Hardware Considerations for Implementing SR-IOV** の Additional Hardware Considerations for Using Device Assignment を参照してください。

直接デバイスの割り当て用にホストを設定する方法の詳細は、「PCI パススルーを有効にするためのホストの設定」を参照してください。

仮想マシンにホストデバイスを割り当てる方法の詳細は、**Virtual Machine Management Guide**の Host Devices を参照してください。

ホストデバイスの表示

1. **Compute → Hosts** をクリックします。
2. ホストの名前をクリックして、詳細ビューを開きます。
3. **ホストデバイス**"タブをクリックします。

このタブでは、仮想マシンに接続されているかどうか、その仮想マシンで現在使用されているかどうかなど、ホストデバイスの詳細が表示されます。

10.5.24. 管理ポータルからのコックピットへのアクセス

Cockpit はデフォルトで Red Hat Virtualization Hosts (RHVH) および Red Hat Enterprise Linux ホストで利用できます。コックピットの Web インターフェイスには、ブラウザーにアドレスを入力するか、アドミニストレーションポータルからアクセスすることができます。

管理ポータルからのコックピットへのアクセス

1. 管理ポータルで **コンピュート → ホスト** をクリックし、ホストを選択します。
2. **Host Console** をクリックします。

コックピットのログインページが新しいブラウザーウィンドウで開きます。

10.5.25. レガシー SPICE 暗号の設定

SPICE コンソールでは、デフォルトで FIPS 準拠の暗号化を行い、暗号文字列を使用します。デフォルトの SPICE 暗号文字列は **kECDHE+FIPS:kDHE+FIPS:kRSA+FIPS:!eNULL:!aNULL** です。

通常、この文字列で十分です。ただし、古いオペレーティングシステムまたは SPICE クライアントの仮想マシンがあり、そのうちのいずれかが FIPS 準拠の暗号化に対応していない場合は、弱い暗号文字列を使用する必要があります。そうしないと、新規クラスターまたは新規ホストを既存のクラスターにインストールし、その仮想マシンへの接続を試みると、接続のセキュリティーエラーが発生します。

Ansible Playbook を使用して暗号文字列を変更できます。

暗号文字列の変更

1. Manager マシンで、**/usr/share/ovirt-engine/playbooks** ディレクトリーにファイルを作成します。以下に例を示します。

```
# vim /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

2. ファイルに以下を入力し、保存します。

```
name: oVirt - setup weaker SPICE encryption for old clients
hosts: hostname
vars:
  host_deploy_spice_cipher_string: 'DEFAULT:-RC4:-3DES:-DES'
roles:
  - ovirt-host-deploy-spice-encryption
```

3. 作成したファイルを実行します。

```
# ansible-playbook -l hostname /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

または、以下のように変数 **host_deploy_spice_cipher_string** に **--extra-vars** オプションを指定して、Ansible Playbook **ovirt-host-deploy** でホストを再設定できます。

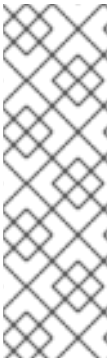
```
# ansible-playbook -l hostname \
  --extra-vars host_deploy_spice_cipher_string="DEFAULT:-RC4:-3DES:-DES" \
  /usr/share/ovirt-engine/playbooks/ovirt-host-deploy.yml
```

10.6. ホストの耐障害性

10.6.1. 高可用性

Red Hat Virtualization Manager は、クラスター内のホストの応答性を維持するためにフェンシングを使用します。**Non Responsive** ホストは、**Non Operational** ホストとは異なります。**Non Operational** ホストは、マネージャーから通信可能ですが、論理ネットワークがないなど、設定が正しくない場合があります。**Non Responsive** ホストは、マネージャーが連絡を取ることができません。

フェンシングにより、クラスターは予期せぬホストの障害に対応し、省電力、ロードバランシング、仮想マシンの可用性のポリシーを実施することができます。ホストの電源管理装置にフェンシングのパラメーターを設定し、時々その正しさをテストする必要があります。フェンシング操作では、反応しないホストが再起動され、所定の時間内にアクティブな状態に戻らない場合は、手動での介入やトラブルシューティングが行われるまで、反応しない状態が維持されます。



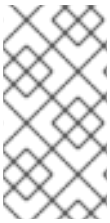
注記

フェンシングパラメーターを自動的にチェックするには、**PMHealth Check Enabled** (デフォルトでは false) と **PMHealth Check Interval In Sec** (デフォルトでは 3600 秒) の engine-config オプションを設定することができます。

PMHealth Check Enabled が true に設定されている場合、**PMHealth Check Interval In Sec** で指定された間隔で全てのホストエージェントをチェックし、問題を検出した場合は警告を発します。engine-config オプションの設定に関する詳細は、「engine-config コマンドの構文」 を参照してください。

電源管理操作は、Red Hat Virtualization Manager が再起動した後、プロキシーホストによって、または管理ポータルで手動で行うことができます。反応しないホスト上で稼働しているすべての仮想マシンを停止し、高可用性を持つ仮想マシンを別のホスト上で起動します。電源管理操作には、少なくとも 2 台のホストが必要です。

Manager の起動後、電源管理が有効になっている無反応なホストに対して、静寂時間 (デフォルトでは 5 分) が経過した後、自動的にフェンスを試みます。**Disable Fence At Startup In Sec** エンジン設定オプションを更新することで、静寂時間を設定することができます。



注記

Disable Fence At Startup In Sec エンジンコンフィグオプションは、ホストの起動時に Manager がフェンスを試みてしまうシナリオを防ぐのに役立ちます。通常、ホストのブートプロセスは Manager のブートプロセスよりも長いため、データセンターが停止した後にこのような事態が発生する可能性があります。

ホストのフェンスは、プロキシーホストが電源管理パラメーターを使って自動的に行うことも、ホストを右クリックしてメニューのオプションを使って手動で行うこともできます。

重要

ホストが高可用性を持つ仮想マシンを実行する場合、パワーマネージメントを有効にして設定する必要があります。

10.6.2. Red Hat Virtualization の Proxy による電源管理

Red Hat Virtualization Manager は、フェンスエージェントと直接通信しません。その代わりに、Manager はプロキシーを使用してホストの電源管理デバイスに電源管理コマンドを送信します。Manager は VDSM を使用してパワーマネージメントデバイスのアクションを実行するため、環境内の別のホストをフェンシングプロキシーとして使用しています。

以下で選択することができます。

- フェンシングが必要なホストと同じクラスター内の任意のホスト。
- フェンシングが必要なホストと同じデータセンターにあるすべてのホスト。

実行可能なフェンシングプロキシーホストのステータスは **UP** または **Maintenance** のいずれかです。

10.6.3. ホストでのフェンシングパラメーターの設定

ホストフェンシングのパラメーターは、**New Host** または **Edit Host** ウィンドウの **Power Management** フィールドで設定します。パワーマネジメントは、RAC (Remote Access Card) などの追加インターフェイスを使って、システムがトラブルのあるホストをフェンスすることを可能にします。

すべての電源管理操作は、Red Hat Virtualization Manager によって直接行われるのではなく、プロキシーホストを使用して行われます。電源管理操作には、少なくとも 2 台のホストが必要です。

ホストでのフェンシングパラメーターの設定

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **Edit** をクリックします。
3. **Power Management** タブをクリックします。
4. **パワーマネジメントを有効にする**チェックボックスを選択し、フィールドを有効にします。
5. **Kdump integration** チェックボックスを選択すると、カーネルクラッシュダンプの実行中にホストがフェンシングするのを防ぐことができます。

重要

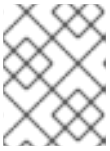
既存のホストで **Kdump 統合** を有効または無効にした場合、ホストを再インストール する必要があります。

6. オプションとして、ホストの電源管理をホストのクラスターの**スケジューリングポリシー**で制御したくない場合は、**Disable policy control of power management**チェックボックスを選択します。

7. **+** ボタンをクリックして、新しいパワーマネージメントデバイスを追加します。**フェンスエージェントの編集** ウィンドウが開きます。

8. 電源管理装置の**アドレス**、**ユーザー名**、**パスワード**を入力します。

9. ドロップダウンリストから電源管理デバイスの**種類**を選択します。



注記

カスタムの電源管理デバイスの設定方法の詳細は、https://access.redhat.com/articles/1238743 を参照してください。

10. 電源管理装置がホストとの通信に使用する **SSH ポート** 番号を入力します。

11. パワーマネージメントデバイスのブレードを識別するための **Slot** 番号を入力してください。

12. パワーマネージメントデバイスの**オプション**を入力します。**key=value** ペアのコンマ区切りリストを使用します。

13. パワーマネージメントデバイスがホストに安全に接続できるようにするには、**Secure** チェックボックスを選択します。

14. **Test** ボタンをクリックして、設定が正しいことを確認します。検証に成功すると **Test Succeeded、Host Status is: on** と表示されます。

警告

電源管理パラメーター (ユーザー ID、パスワード、オプションなど) は、Red Hat Virtualization Manager によってセットアップ時にのみテストされ、その後は手動でテストされます。不正なパラメーターに関する警告を無視したり、Red Hat Virtualization Manager で対応する変更を行わずに電源管理ハードウェアでパラメーターを変更したりすると、最も必要なときにフェンシングが失敗する可能性があります。

15. **OK** をクリックして、**Edit fence agent** ウィンドウを閉じます。

16. **電源管理** タブで、オプションで **詳細パラメーター** を展開し、上下のボタンを使用して、マネージャーがホストの **クラスター** と **DC** (データセンター) でフェンシングプロキシーを検索する順序を指定します。

17. **OK** をクリックします。

ホストのリストに戻ります。ホスト名の横にあった感嘆符が消えていることに注意してください。これはパワーマネージメントの設定が成功したことを示しています。

10.6.4. fence_kdump の高度な設定

kdump

ホスト名をクリックすると、詳細表示の **General** タブに kdump サービスの状態が表示されます。

- **Enabled**: kdump が正しく設定され、kdump サービスが実行されています。
- **Disable**: kdump サービスが実行されていません (この場合、kdump インテグレーションは正しく動作しません)。
- **Unknown**: kdump の状態を報告しない以前の VDSM のバージョンを持つホストでのみ発生します。

kdump のインストールおよび使用に関する詳細は、Red Hat Enterprise Linux 7 Kernel Crash Dump Guide を参照してください。

fence_kdump

New Host または **Edit Host** ウィンドウの **Power Management** タブで **Kdump の統合** を有効にすると、標準的な fence_kdump の設定が行われます。環境のネットワーク設定が単純で、マネージャーの FQDN がすべてのホストで解決可能な場合は、デフォルトの fence_kdump 設定で十分に使用できます。

ただし、fence_kdump の高度な設定が必要な場合もあります。ネットワークが複雑な環境では、Manager、fence_kdump リスナー、またはその両方の設定を手動で変更する必要がある場合があります。例えば、**Kdump 統合** が有効になっているすべてのホストで Manager の FQDN が解決できない場合、**engine-config** を使って適切なホスト名や IP アドレスを設定することができます。

```
engine-config -s FenceKdumpDestinationAddress=A.B.C.D
```

以下の例の場合も、設定変更が必要な場合があります。

- Manager には 2 つの NIC があり、そのうちの 1 つは公開用で、もう 1 つは fence_kdump メッセージの優先的な送信先となっています。
- fence_kdump のリスナーを別の IP やポートで実行する必要があります。
- パケットロスの可能性を防ぐために、fence_kdump の通知メッセージのカスタムインターバルを設定する必要があります。

デフォルトの設定を変更する必要があるのは、より複雑なネットワーク設定の場合のみであるため、カスタマイズされた fence_kdump 検出設定は、上級ユーザーのみに推奨されます。fence_kdump リスナーの設定オプションは、fence_kdump listener Configuration を参照してください。Manager での kdump の設定については、Configuring fence_kdump on the Manager を参照してください。

10.6.4.1. fence_kdump リスナーの設定

fence_kdump リスナーの設定を編集します。これは、デフォルトの設定では十分ではない場合にのみ必要です。

fence_kdump リスナーの手動設定

1. **etc/ovirt-engine/ovirt-fence-kdump-listener.conf.d/** に新しいファイル (例えば、**my-fence-kdump.conf**) を作成します。
2. カスタマイズした内容を **OPTION=value** の構文で入力し、ファイルを保存します。

重要

「マネージャーでの fence_kdump の設定」 の fence_kdump リスナー設定オプションの表で説明されているように、編集した値は **engine-config** でも変更する必要があります。

3. fence_kdump リスナーを再起動します。

```
# systemctl restart ovirt-fence-kdump-listener.service
```

以下のオプションは、必要に応じてカスタマイズすることができます。

表10.9 追加のリスナー設定オプション

変数	説明	デフォルト	注記
LISTENER_ADDRESS	fence_kdump メッセージを受信するための IP アドレスを定義します。	0.0.0.0	このパラメーターの値を変更する場合は、**engine-config** の **Fence Kdump Destination Address** の値と一致させる必要があります。
LISTENER_PORT	fence_kdump メッセージを受信するポートを定義します。	7410	このパラメーターの値を変更する場合は、**engine-config** の **Fence Kdump Destination Port** の値と一致させる必要があります。
HEARTBEAT_INTERVAL	リスナーのハートビート更新の間隔を秒単位で定義します。	30	このパラメーターの値を変更する場合は、**engine-config** の **Fence Kdump Listener Timeout** の値の半分以下にしなければなりません。
SESSION_SYNC_INTERVAL	リスナーのメモリー上のホストの kdumping セッションをデータベースに同期させる間隔を秒単位で定義します。	5	このパラメーターの値を変更する場合は、**engine-config** の **Kdump Started Timeout** の値の半分以下にしなければなりません。

変数	説明	デフォルト	注記
REOPEN_DB_CONNECTION_INTERVAL	以前に利用できなかったデータベース接続を再開する間隔を秒単位で定義します。	30	-
KDUMP_FINISHED_TIMEOUT	kdumping ホストからのメッセージを最後に受信してから、ホストの kdump フローが FINISHED とマークされるまでの最大タイムアウトを秒単位で定義します。	60	このパラメーターの値を変更する場合は、**engine-config** の **Fence Kdump Message Interval** の値の 2 倍以上でなければなりません。

10.6.4.2. マネージャーでの fence_kdump の設定

Manager の kdump 設定を編集します。これは、デフォルトの設定では十分ではない場合にのみ必要です。現在の設定値は以下の方法で確認できます。

```
# engine-config -g OPTION
```

engine-config での Kdump の手動設定

1. **engine-config** コマンドで kdump の設定を編集します。

   ```
   # engine-config -s OPTION=value
   ```

   重要

   編集した値は、**Kdump Configuration Options** の表に記載されているように、fence_kdump リスナー設定ファイルでも変更する必要があります。「fence_kdump リスナーの設定」 を参照してください。

2. **ovirt-engine** サービスを再起動します。

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

3. 必要に応じて、**Kdump 統合** を有効にして、すべてのホストを再インストールします (以下の表を参照)。

engine-config では以下のオプションが設定できます。

表10.10 Kdump 設定オプション

変数	説明	デフォルト	注記

変数	説明	デフォルト	注記
FenceKdumpDestinationAddress	fence_kdump メッセージの送信先となるホスト名または IP アドレスを定義します。空の場合は、マネージャーの FQDN が使用されます。	空の文字列 (Manager FQDN が使用されます)	このパラメーターの値を変更する場合は、fence_kdump リスナー設定ファイルの **LISTENER_ADDRESS** の値と一致させる必要があり、**Kdump 統合**が有効になっているすべてのホストを再インストールする必要があります。
FenceKdumpDestinationPort	fence_kdump メッセージの送信先となるポートを定義します。	7410	このパラメーターの値を変更する場合は、fence_kdump リスナー設定ファイルの **LISTENER_PORT** の値と一致させる必要があり、**Kdump 統合**が有効になっているすべてのホストを再インストールする必要があります。
FenceKdumpMessageInterval	fence_kdump が送信するメッセージの間隔を秒単位で定義します。	5	このパラメーターの値を変更する場合は、fence_kdump リスナー設定ファイルの **KDUMP_FINISHED_TIMEOUT** の値の半分以下にし、**Kdump 統合**を有効にしているすべてのホストを再インストールする必要があります。
FenceKdumpListenerTimeout	fence_kdump リスナーが生存しているとみなす最後のハートビートからの最大タイムアウトを秒単位で定義します。	90	このパラメーターの値を変更する場合は、fence_kdump リスナー設定ファイルの **HEARTBEAT_INTERVAL** の値の 2 倍以上でなければなりません。
KdumpStartedTimeout	kdumping ホストからの最初のメッセージを受信するまで (ホストの kdump フローが開始されたことを検出するまで) の最大タイムアウトを秒単位で定義します。	30	このパラメーターの値を変更する場合は、fence_kdump リスナー設定ファイルの **SESSION_SYNC_INTERVAL** の値の 2 倍以上、**Fence Kdump Message Interval** の値を変更する必要があります。

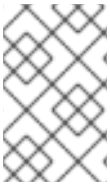
10.6.5. ソフトフェンシングホスト

ホストは予期せぬ問題で応答しなくなることがありますが、VDSM は要求に応答できないものの、VDSM に依存している仮想マシンは生きており、アクセス可能です。このような場合は、VDSM を再起動することで VDSM が応答可能な状態に戻り、この問題が解決します。

"SSH Soft Fencing "とは、応答しないホストに対して Manager が SSH 経由で VDSM の再起動を試みるプロセスのことです。Manager が SSH 経由で VDSM の再起動に失敗した場合、外部フェンシングエージェントが設定されていれば、フェンシングの責任は外部フェンシングエージェントに移ります。

SSH でのソフトフェンシングは以下のように動作します。ホストでフェンシングを設定して有効にする必要があり、有効なプロキシーホスト (データセンター内の UP 状態の 2 番目のホスト) が存在する必要があります。Manager とホストの接続がタイムアウトすると、以下のようになります。

1. 最初のネットワーク障害では、ホストの状態が接続中に変わります。
2. その後、マネージャーは VDSM にステータスの問い合わせを 3 回試みるか、ホストの負荷に応じた間隔で待機します。間隔の長さを決定する式は、設定値 TimeoutToResetVdsInSeconds (デフォルトは 60 秒) + [DelayResetPerVmInSeconds (デフォルトは 0.5 秒)]*(ホスト上で実行している仮想マシンの数)+ [DelayResetForSpmInSeconds (デフォルトは 20 秒です)] * 1 (ホストが SPM として実行している場合) または 0 (ホストが SPM として実行されていない場合)。VDSM に応答する最大時間を与えるために、Manager は上記の 2 つのオプションのうち長い方を選択します (VDSM のステータスまたは上記の式で決定された間隔を取得するための 3 回の試行)。
3. 間隔が経過したときにホストが応答しない場合には、**vdsm restart** を SSH 経由で実行します。
4. **vdsm restart** が行われても、ホストと Manager 間の接続を再度確立しない場合は、ホストのステータスが **Non Responsive** に変更になり、電源管理が設定されている場合には、フェンシングが外部フェンシングエージェントに渡されます。



注記

SSH を介したソフトフェンシングは、電源管理が設定されていないホストで実行できます。これはフェンシングとは異なります。フェンシングは、電源管理が設定されているホストでのみ実行できます。

10.6.6. ホストの電源管理機能の利用

電源管理がホストに設定されている場合は、Administration Portal インターフェイスから多くのオプションにアクセスできます。電源管理デバイスはそれぞれカスタマイズ可能な独自のオプションを持っていますが、いずれもホストの起動、停止、再起動の基本的なオプションをサポートしています。

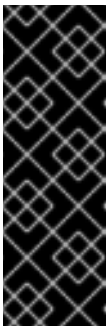
ホストの電源管理機能の利用

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **Management** ドロップダウンメニューをクリックし、以下の **Power Management** オプションを選択します。
   - **再起動**します。このオプションは、ホストを停止し、ホストのステータスが **Down** に変わるまで待機します。エージェントがホストのダウンを確認すると、クラスター内の別のホストで高可用仮想マシンが再起動されます。その後、エージェントはこのホストを再起動します。ホストが使用可能な状態になると、ステータスが **Up** と表示されます。

- **起動**します。このオプションは、ホストを起動し、クラスターに参加させます。使用可能な状態になると、ステータスが **Up** と表示されます。
- **Stop**: このオプションは、ホストの電源をオフにします。このオプションを使用する前に、ホスト上で実行されている仮想マシンがクラスター内の他のホストに移行されていることを確認してください。そうしないと、仮想マシンがクラッシュし、可用性の高い仮想マシンだけが別のホストで再起動されます。ホストが停止している場合、ステータスは **Non-Operational** と表示されます。

注記

電源管理が有効になっていない場合は、これを選択してホストを再起動または停止し、**Management** ドロップダウンメニューをクリックして、**SSH Management** オプション、**Restart** または **Stop** を選択します。



重要

1 つのホスト上に 2 つのフェンシングエージェントが定義されている場合、それらは同時または連続して使用することができます。同時進行のエージェントの場合、両方のエージェントが Stop コマンドに反応しないとホストは停止せず、一方のエージェントが Start コマンドに反応するとホストは立ち上がります。シーケンシャルエージェントの場合、ホストを起動または停止するには、まずプライマリーエージェントが使用され、それが失敗した場合はセカンダリーエージェントが使用されます。

3. **OK** をクリックします。

10.6.7. 反応しないホストを手動でフェンシングまたは隔離する方法

ハードウェアの故障などにより、ホストが予期せず非応答状態になると、環境のパフォーマンスに大きな影響を与えます。電源管理装置がない場合や、設定が間違っている場合は、手動でホストを再起動することができます。

警告

ホストを手動で再起動しない限り、**Confirm host has been rebooted** オプションは使用しないでください。ホストの実行中にこのオプションを使用すると、仮想マシンのイメージが破損する可能性があります。

反応しないホストを手動でフェンシングまたは隔離する方法

1. 管理ポータルで **Compute → Hosts** をクリックし、ホストのステータスが **Non Responsive** になっていることを確認します。
2. システムを手動で再起動します。これは物理的にラボに入り、ホストを再起動することを意味します。
3. 管理ポータルでホストを選択し、**More Actions** (⋮) をクリックしてから、**Confirm 'Host has been Rebooted'** をクリックします。

4. **Approve operation** チェックボックスを選択し、**OK** をクリックします。

5. ホストの起動に異常に長い時間がかかる場合は、**ServerRebootTimeout** を設定して、ホストが **Non Responsive** と判断するまで待機する秒数を指定できます。

```
# engine-config --set ServerRebootTimeout=integer
```

第11章 ストレージ

Red Hat Virtualization では、仮想ディスク、ISO ファイル、スナップショットのための集中型ストレージシステムを使用しています。ストレージネットワーキングは、以下の方法で実装できます。

- Network File System (NFS)
- Gluster FS のエクスポート
- その他 POSIX 準拠ファイルシステム
- Internet Small Computer System Interface (iSCSI)
- 仮想化ホストに直接接続されたローカルストレージ
- ファイバーチャネルプロトコル (FCP)
- Parallel NFS (pNFS)

ストレージドメインが接続され、アクティベートされなければデータセンターは初期化されないため、ストレージの設定は新しいデータセンターの前提条件となります。

Red Hat Virtualization のシステム管理者として、仮想化された企業のためにストレージを作成、設定、アタッチ、維持する必要があります。ストレージの種類とその使い方をよく理解しておく必要があります。ストレージアレイのベンダーガイドを読み、ストレージの概念、プロトコル、要件、または一般的な使用方法の詳細については、Red Hat Enterprise Linux Storage Administration Guide を参照してください。

ストレージドメインを追加するには、管理ポータルに正常にアクセスでき、ステータスが **Up** になっているホストが 1 台以上接続されている必要があります。

Red Hat Virtualization には、3 種類のストレージドメインがあります。

- **Data Domain:** データドメインは、データセンター内のすべての仮想マシンとテンプレートの仮想ハードディスクと OVF ファイルを保持します。さらに、仮想マシンのスナップショットもデータドメインに保存されます。
  データドメインは、データセンター間で共有することはできません。複数のタイプ (iSCSI、NFS、FC、POSIX、および Gluster) のデータドメインは、ローカルドメインではなくすべて共有されている場合、同じデータセンターに追加できます。

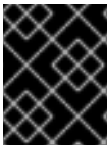
  データセンターに他のタイプのドメインをアタッチする前に、データドメインをアタッチする必要があります。

- **ISO Domain:**ISO ドメインは、仮想マシンのオペレーティングシステムとアプリケーションのインストールと起動に使用される ISO ファイル (または論理 CD) を格納します。ISO ドメインは、データセンターが物理的なメディアを必要としない。ISO ドメインは、異なるデータセンターで共有することができます。ISO ドメインは NFS ベースのものしかありません。データセンターに 1 つの ISO ドメインのみをアタッチできます。

- **Export Domain:**エクスポートドメインは、データセンターと Red Hat Virtualization 環境の間でイメージをコピーおよび移動するために使用される一時的なストレージリポジトリーです。エクスポートドメインは、仮想マシンのバックアップに使用できます。エクスポートドメインは、データセンター間で移動することができますが、同時に 1 つのデータセンターでしか有効にすることができません。エクスポートドメインは、NFS ベースでのみ可能です。データセンターに追加できるエクスポートドメインは 1 つだけです。

注記

エクスポートストレージドメインは非推奨になりました。ストレージデータドメインはデータセンターから接続を解除し、同じ環境または別の環境にある別のデータセンターにインポートすることができます。仮想マシン、フローティング仮想ディスク、およびテンプレートは、インポートされたストレージドメインからアタッチされたデータセンターにアップロードできます。ストレージドメインのインポートに関する詳細は、「既存のストレージドメインのインポート」を参照してください。



重要

Red Hat Virtualization 環境へのストレージの設定と接続は、データセンターのストレージニーズが決定してから開始してください。

11.1. ストレージドメインについて

ストレージドメインとは、共通のストレージインターフェイスを持つイメージの集合体です。ストレージドメインには、テンプレートや仮想マシンの完全なイメージ (スナップショットを含む)、または ISO ファイルが格納されています。ストレージドメインは、ブロックデバイス (SAN - iSCSI または FCP) またはファイルシステム (NAS - NFS、GlusterFS、またはその他の POSIX 準拠のファイルシステム) で設定できます。

デフォルトでは、GlusterFS ドメインとローカルストレージドメインは 4K ブロックサイズをサポートします。4K ブロックサイズを使用すると、特に大きなファイルを使用する場合などに、パフォーマンスが向上します。また、VDO などの 4K 互換性を必要とするツールを使用する場合にも必要です。

NFS では、仮想ディスク、テンプレート、スナップショットはすべてファイルです。

SAN (iSCSI/FCP) では、各仮想ディスク、テンプレート、またはスナップショットは論理ボリュームです。ブロックデバイスは、ボリュームグループと呼ばれる論理エンティティーに集約され、LVM (Logical Volume Manager) によって論理ボリュームに分割されて仮想ハードディスクとして使用されます。LVM の詳細は、**Red Hat Enterprise Linux Logical Volume Manager Administration Guide**を参照してください。

仮想ディスクは、QCOW2 または raw の 2 つの形式のいずれかを持つことができます。ストレージのタイプは、スパースまたは事前割り当てにすることができます。スナップショットは常にスパースですが、どちらのフォーマットのディスクでも取ることができます。

同じストレージドメインを共有する仮想マシンは、同じクラスターに属するホスト間で移行することができます。

11.2. NFS ストレージの準備と追加

11.2.1. 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 の設定については、**Red Hat Enterprise Linux 7 Storage Administration Guide**の Network File System (NFS) を参照してください。

NFS 共有をエクスポートする方法については、How to export 'NFS' share from NetApp Storage / EMC SAN in Red Hat Virtualization 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。

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 36
```

3. エクスポートしたディレクトリーの所有権を 36:36 に設定します。これにより、**vdsm:kvm** の所有権が与えられます。

```
# chown -R 36:36 /exports/data
```

4. 所有者に読み取り/書き込みアクセスを許可し、グループおよびその他のユーザーに読み取り/実行アクセスを許可するように、ディレクトリーのモードを変更します。

```
# chmod 0755 /exports/data
```

11.2.2. NFS ストレージの追加

この手順では、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方法について説明します。

ISO またはエクスポートドメインが必要な場合にも、この手順を使用します。ただし、**Domain Function** の一覧から **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. **新規ドメイン** 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** 一覧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)、**[2001:0:0:0:0:0:0:5db1]:/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することが可能です。

a. **Advanced Parameters** をクリックします。

b. **Warning Low Space Indicator** のフィールドに、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** の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することが可能ですが、その場合にはすでに存在していたディスクの削除後にワイププロパティーは変更されません。

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。

11.2.3. NFS ストレージの増設

NFS ストレージの容量を増やすには、新しいストレージドメインを作成して既存のデータセンターに追加するか、NFS サーバーの利用可能な空き容量を増やすことができます。以前のオプションについては、「NFS ストレージの追加」 を参照してください。以下の手順では、既存の NFS サーバーの利用可能な空き容量を増やす方法を説明しています。

既存の NFS ストレージドメインの増加

1. **Storage → Domains** をクリックします。

2. NFS ストレージドメインの名前をクリックして、詳細ビューを開きます。

3. **Data Center** タブをクリックし、**Maintenance** をクリックして、ストレージドメインをメンテナンスモードにします。これにより、既存の共有がアンマウントされ、ストレージドメインのサイズ変更が可能になります。

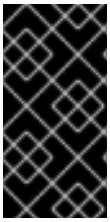
4. NFS サーバー上で、ストレージのサイズを変更する。Red Hat Enterprise Linux 6 システムについては、Red Hat Enterprise Linux 6 Storage Administration Guide を参照してください。Red Hat Enterprise Linux 7 システムについては、Red Hat Enterprise Linux 7 Storage Administration Guide を参照してください。

5. 詳細ビューで、**Data Center** タブをクリックし、**Activate** をクリックしてストレージドメインをマウントします。

11.3. ローカルストレージの準備と追加

11.3.1. ローカルストレージの準備

ホスト上にローカルストレージドメインをセットアップすることができます。ホストがローカルストレージを使用するように設定すると、そのホストは、他のホストを追加することができない新規データセンターとクラスターに自動的に追加されます。複数のホストで設定されるクラスターの場合は、全ホストが全ストレージドメインにアクセス可能である必要があり、ローカルストレージでは対応不可能です。単一ホストのクラスター内で作成された仮想マシンは、移行、フェンシング、スケジューリングはできません。



重要

Red Hat Virtualization Host (RHVH) の場合は、必ず / (root) とは異なるファイルシステム上にローカルストレージを定義する必要があります。Red Hat は、アップグレード中にデータが失われる可能性を防ぐために、別の論理ボリュームまたはディスクを使用することをお勧めします。

ローカルストレージの準備 (Red Hat Enterprise Linux ホスト向け)

1. ホストで、ローカルストレージで使用するディレクトリーを作成します。

```
# mkdir -p /data/images
```

2. **vdsm** ユーザー (UID 36) および **kvm** グループ (GID 36) がそのディレクトリーに読み取り/書き込みアクセスできるように、パーミッションを設定します。

```
# chown 36:36 /data /data/images
# chmod 0755 /data /data/images
```

ローカルストレージの準備 (Red Hat Virtualization Host 向け)

Red Hat は、次のように論理ボリューム上にローカルストレージを作成することをお勧めします。

1. ローカルストレージディレクトリーを作成します。

```
# mkdir /data
# lvcreate -L $SIZE rhvh -n data
# mkfs.ext4 /dev/mapper/rhvh-data
# echo "/dev/mapper/rhvh-data /data ext4 defaults,discard 1 2" >> /etc/fstab
# mount /data
```

2. 新しいローカルストレージをマウントし、続いてパーミッションと所有者を変更します。

```
# mount -a
# chown 36:36 /data /rhvh-data
# chmod 0755 /data /rhvh-data
```

11.3.2. ローカルストレージの追加

ホストにローカルストレージを追加すると、ホストが新規のデータセンターとクラスターに配置されます。ローカルストレージ設定ウィンドウは、データセンター、クラスター、ストレージの作成を 1 つのプロセスにまとめています。

手順

1. **コンピュート → ホスト** をクリックし、ホストを選択します。
2. **Management → Maintenance** をクリックし、**OK** をクリックします。
3. **管理 → ローカルストレージの設定** をクリックします。
4. **Data Center**、**Cluster**、および **Storage** フィールドの横にある **Edit** ボタンをクリックし、ローカルのストレージドメインを設定して名前を付けます。

5. 文字入力フィールドにローカルストレージへのパスを設定します。

6. 該当する場合には、**Optimization** タブをクリックして新規ローカルストレージクラスターのメモリー最適化ポリシーを設定します。

7. **OK** をクリックします。

ホストが、自己のデータセンター内でオンラインになります。

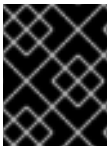
11.4. POSIX 準拠ファイルシステムストレージの準備

11.4.1. POSIX 準拠ファイルシステムストレージの準備

POSIX ファイルシステムのサポートにより、通常コマンドラインから手動でマウントするときと同じマウントオプションを使ってファイルシステムをマウントすることができます。この機能は、NFS、iSCSI、または FCP 以外を使用してマウントするストレージへのアクセスを可能にすることを目的としています。

Red Hat Virtualization でストレージドメインとして使用する POSIX 準拠のファイルシステムは、Global File System 2 (GFS2) 等のクラスター化したファイルシステムで、かつスパースファイルおよびダイレクト I/O をサポートしている必要があります。たとえば、Common Internet File System (CIFS) は、ダイレクト I/O をサポートしていないので、Red Hat Virtualization との互換性はありません。

POSIX 準拠ファイルシステムストレージの準備および設定に関する情報は、Red Hat Enterprise Linux Global File System 2 を参照してください。



重要

POSIX 準拠ファイルシステムのストレージドメインを作成して、NFS ストレージを **マウントしないでください**。必ず、NFS ストレージドメインを作成してください。

11.4.2. POSIX 準拠ファイルシステムストレージの追加

この手順では、既存の POSIX 準拠ファイルシステム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方法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. このストレージドメインと関連づける **Data Center** を選択します。選択したデータセンターのタイプは、**POSIX (POSIX compliant FS)** でなければなりません。または、**(none)** 選択します。

5. **Domain Function** ドロップダウンリストから **Data** を選択し、**Storage Type** ドロップダウンリストから **POSIX compliant FS** を選択します。
該当する場合には、ドロップダウンメニューから **Format** を選択します。

6. **Host** のドロップダウンリストからホストを選択します。

7. 通常 **mount** コマンドで指定するように、POSIX ファイルシステムへの **Path** を入力します。

8. 通常 **-t** 引数を使用して **mount** コマンドで指定するように、**VFS Type** を入力します。有効な VFS タイプの一覧は、**man mount** で確認してください。

9. 通常 **mount** コマンドに **-o** 引数を指定して指定するように、追加の **Mount Options** を入力します。このマウントオプションはコンマ区切りリストで提示してください。有効なマウントオプションの一覧については、**man mount** で確認してください。

10. オプションで、詳細パラメーターを設定することが可能です。

    a. **Advanced Parameters** をクリックします。

    b. **Warning Low Space Indicator** のフィールドに、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、ユーザーに警告の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** の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することが可能ですが、その場合にはすでに存在していたディスクの削除後にワイププロパティーは変更されません。

11. **OK** をクリックします。

11.5. ブロックストレージの準備と追加

11.5.1. 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。

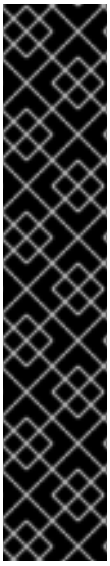
iSCSI ストレージの設定については、**Red Hat Enterprise Linux 7 Storage Administration Guide**の Online Storage Management を参照してください。

重要

ブロックストレージを使用する際、仮想マシンを Raw デバイスまたは直接 LUN にデプロイし、論理ボリュームマネージャーで管理する場合は、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。詳細は、https://access.redhat.com/solutions/2662261 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。

この状況を防ぐために、Red Hat は、接続がある場合にはブート 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て、キューに配置されるようにすることを推奨します。

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

11.5.2. iSCSI ストレージの追加

この手順では、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方法について説明します。

手順

1. **Storage** → **Domains** をクリックします。
2. **新規ドメイン** 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** に**Data** を、**Storage Type** に **iSCSI** を、それぞれ選択します。
6. **Host** に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、直接 Manager からではなく、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には、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には、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、**新規ドメイン** 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

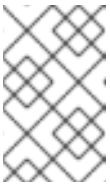
環境の外部で使用されている LUN も表示されます。

ターゲットを検出 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。

b. **アドレス** 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** の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。

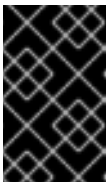


注記

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 Guide** 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には、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さらにパスを追加する方法は、現在サポートされていません。

8. 対象の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することが可能です。

a. **Advanced Parameters** をクリックします。

b. **Warning Low Space Indicator** のフィールドに、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** の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することが可能ですが、その場合にはすでに存在していたディスクの削除後にワイププロパティーは変更されません。

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には、Configuring iSCSI Multipathing に記載の手順に従って、iSCSI 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、Migrating a Logical Network to an iSCSI Bond を参照してください。

11.5.3. Configuring iSCSI Multipathing

iSCSI マルチパスでは、論理ネットワークと iSCSI ストレージ接続のグループを作成管理することができます。ホストと iSCSI ストレージの間に複数のネットワークパスがあることで、ネットワークパスの障害によるホストのダウンタイムを防ぎます。

Manager は、iSCSI ボンドの論理ネットワークに割り当てられた NIC または VLAN を使用して、データセンター内の各ホストを各ターゲットに接続します。

複数のターゲットと論理ネットワークを持つ iSCSI ボンドを作成し、冗長性を持たせることができます。

前提条件

- 1 つまたは複数の iSCSI ターゲット
- 以下の要件を満たす 1 つまたは複数の 論理ネットワーク。
  - **必須** または **VM ネットワーク** として定義されていない
  - ホストインターフェイスに割り当てられた
  - iSCSI ボンド内の他の論理ネットワークと同じ VLAN およびサブネット内の スタティック IP アドレス が割り当てられていること

手順

1. **Compute → Data Centers** をクリックします。
2. データセンター名をクリックして、詳細ビューを開きます。
3. **iSCSI Multipathing** タブで **Add** をクリックします。
4. **Add iSCSI Bond** ウィンドウで、**Name** と **Description** を入力します。
5. **Logical Networks** からは論理ネットワークを、**Storage Targets** からはストレージドメインを選択します。同じターゲットへのすべてのパスを選択する必要があります。
6. **OK** をクリックします。

データセンター内のホストは、iSCSI ボンドの論理ネットワークを介して iSCSI ターゲットに接続されています。

11.5.4. ロジカルネットワークを iSCSI ボンドに移行する

iSCSI トラフィック用に作成した論理ネットワークがあり、既存の ネットワークボンド の上に設定されている場合は、中断やダウンタイムなしに同じサブネット上の iSCSI ボンドに移行することができます。

手順

1. 現在の論理ネットワークを Required ではないものに変更する。
   a. Compute → Clusters をクリックします。
   b. クラスター名をクリックして詳細ビューを開きます。
   c. Logical Networksタブで、現在の論理ネットワーク (**net-1**) を選択し、Manage Networks をクリックします。
   d. Require チェックボックスをオフにして、OK をクリックします。
2. Required ではなく、**VM ネットワーク** でもない新しい論理ネットワークを作成します。
   a. Add Network をクリックして New Logical Network ウィンドウを開きます。
   b. General タブで、Name (**net-2**) を入力し、VM network のチェックボックスをオフにします。
   c. Cluster タブで Require のチェックボックスをオフにして OK をクリックします。
3. 現在のネットワークボンドを削除し、論理ネットワークを再割り当てする。
   a. Compute → Hosts をクリックします。
   b. ホスト名をクリックして、詳細ビューを開きます。
   c. Network Interfaces タブで Setup Host Networks をクリックします。
   d. **net-1** を右にドラッグすると、割り当てが解除されます。
   e. 現在のボンドを右にドラッグすると、ボンドが削除されます。
   f. **net-1** と **net-2** を左にドラッグして、物理インターフェイスに割り当てます。
   g. **net-2** の鉛筆アイコンをクリックして、Edit Network ウィンドウを開きます。
   h. IPV4 タブで、Static を選択します。
   i. サブネットの IP と Netmask/Routing Prefix を入力し、OK をクリックします。
4. iSCSI ボンドを作成します。
   a. Compute → Data Centers をクリックします。
   b. データセンター名をクリックして、詳細ビューを開きます。
   c. iSCSI Multipathing タブで Add をクリックします。
   d. Add iSCSI Bondウィンドウで、Name を入力し、ネットワーク **net-1** と **net-2** を選択して、OK をクリックします。

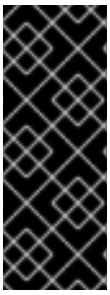
データセンターには、古い論理ネットワークと新しい論理ネットワークを含む iSCSI ボンドがあります。

11.5.5. 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方法で、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) の概念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 **Red Hat Enterprise Linux DM Multipath**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、仮想マシンを Raw デバイスまたは直接 LUN にデプロイし、論理ボリュームマネージャーで管理する場合は、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。詳細は、https://access.redhat.com/solutions/2662261 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。

この状況を防ぐために、Red Hat は、接続がある場合にはブート 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て、キューに配置されるようにすることを推奨します。

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

11.5.6. FCP ストレージの追加

この手順では、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方法について説明します。

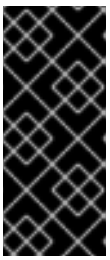
手順

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には **(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** の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なければ、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。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. **New Domain** ウィンドウで、ストレージタイプに **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** の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することが可能です。

   a. **Advanced Parameters** をクリックします。

   b. **Warning Low Space Indicator** のフィールドに、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** の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することが可能ですが、その場合にはすでに存在していたディスクの削除後にワイププロパティーは変更されません。

   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。

11.5.7. iSCSI または FCP ストレージの増設

iSCSI や FCP のストレージサイズを大きくするにはいくつかの方法があります。

- 既存の LUN を現在のストレージドメインに追加します。
- 新しい LUN を持つ新しいストレージドメインを作成し、既存のデータセンターに追加します。「iSCSI ストレージの追加」 を参照してください。

- 基盤となる LUN のサイズを変更することで、ストレージドメインを拡張します。

Red Hat Enterprise Linux 7 システムで iSCSI ストレージの作成、設定、またはサイズ変更を行う方法の詳細については、Red Hat Enterprise Linux 7 Storage Administration Guideを参照してください。

以下の手順では、既存のストレージドメインに新しい LUN を追加して、SAN(Storage Area Network) ストレージを拡張する方法を説明します。

前提条件

- ストレージドメインのステータスは、**UP** である必要があります。
- ステータスが **UP** ですべてのホストが LUN にアクセスできる必要があります。アクセスできない場合、操作は失敗し、LUN はドメインに追加されません。ただし、ホスト自身には影響はありません。新しく追加されたホスト、またはメンテナンスまたは **Non Operational** 状態になっているホストが LUN にアクセスできない場合、ホストの状態は **Non Operational** になります。

既存の iSCSI または FCP ストレージドメインの拡張

1. **Storage → Domains** をクリックして、iSCSI または FCP ドメインを選択します。
2. **Manage Domain** をクリックします。
3. **Targets > LUNs** をクリックし、**Discover Targets** 拡張ボタンをクリックします。
4. ストレージサーバーの接続情報を入力し、**Discover** をクリックして接続を開始します。
5. **LUNs > Targets** をクリックして、新しく利用可能になった LUN のチェックボックスを選択します。
6. **OK** をクリックして、選択したストレージドメインに LUN を追加します。

これにより、追加した LUN のサイズでストレージドメインが増えます。

基礎となる LUN のサイズを変更してストレージドメインを拡張する場合は、管理ポータルで LUN も更新する必要があります。

LUN サイズの更新

1. **Storage → Domains** をクリックして、iSCSI または FCP ドメインを選択します。
2. **Manage Domain** をクリックします。
3. **LUN > Targets**の順にクリックします。
4. **Additional Size** コラムで、LUN の **Add Additional_Storage_Size** ボタンをクリックして更新します。
5. **OK** をクリックして LUN を更新し、新規のストレージサイズを示します。

11.5.8. LUN の再利用

LUN をそのまま再利用して、ストレージドメインまたは仮想ディスクを作成することはできません。LUN を再利用しようとすると、管理ポータルに以下のエラーメッセージが表示されます。

> Physical device initialization failed. Please check that the device is empty and accessible by the host.

セルフホスト型エンジンは、インストール時に以下のエラーを表示します。

```
[ ERROR ] Error creating Volume Group: Failed to initialize physical device: ("
[u'/dev/mapper/000000000000000000000000000000000']",)
[ ERROR ] Failed to execute stage 'Misc configuration': Failed to initialize physical device: ("
[u'/dev/mapper/000000000000000000000000000000000']",)
```

LUN を再利用できるようにするには、古いパーティションテーブルをクリアする必要があります。

LUN からのパーティションテーブルの消去

重要

誤ってデータを破棄しないように、正しい LUN でこの手順を実行する必要があります。

再利用する LUN の ID、一度に読み書きする最大バイト数、およびコピーする入力ブロックの数を指定して **dd** コマンドを実行します。

```
# dd if=/dev/zero of=/dev/mapper/LUN_ID bs=1M count=200 oflag=direct
```

11.6. RED HAT GLUSTER STORAGE の準備と追加

11.6.1. 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 Installation Guide** を参照してください。

Red Hat Virtualization でサポートされる Red Hat Gluster Storage のバージョンについては、https://access.redhat.com/articles/2356261 を参照してください。

11.6.2. Red Hat Gluster Storage の追加

Red Hat Virtualization で Red Hat Gluster Storage を使用するには、**Configuring Red Hat Virtualization with Red Hat Gluster Storage** を参照してください。

Red Hat Virtualization でサポートされる Red Hat Gluster Storage のバージョンについては、https://access.redhat.com/articles/2356261 を参照してください。

11.7. 既存のストレージドメインのインポート

11.7.1. 既存のストレージドメインのインポートの概要

データを含まない新しいストレージドメインを追加するだけでなく、既存のストレージドメインをインポートして、その中のデータにアクセスすることもできます。ストレージドメインをインポートすることで、Manager データベースに障害が発生してもデータを復旧したり、データセンターや環境間でデータを移行したりすることができます。

以下は、各ストレージドメインタイプのインポートの概要です。

データ

既存のデータストレージドメインをインポートすると、そのデータストレージドメインに含まれるすべての仮想マシンやテンプレートにアクセスできるようになります。ストレージドメインをイン

ポートした後、仮想マシン、フローティングディスクイメージ、テンプレートをデスティネーションデータセンターに手動でインポートする必要があります。データストレージドメインが含む仮想マシンやテンプレートをインポートするプロセスは、エクスポートストレージドメインの場合と同様です。ただし、データストレージドメインには、特定のデータセンター内のすべての仮想マシンとテンプレートが含まれているため、データ復旧や、データセンターや環境間で仮想マシンを大規模に移行する場合は、データストレージドメインのインポートを推奨します。

重要

サポートされている正しい互換性レベルを持つデータセンターに接続された既存のデータストレージドメインをインポートすることができます。詳細は、Supportability and constraints regarding importing Storage Domains and Virtual Machines from older RHV versions を参照してください。

ISO

既存の ISO ストレージドメインをインポートすると、その ISO ストレージドメインに含まれるすべての ISO ファイルや仮想ディスケットにアクセスできるようになります。これらのリソースにアクセスするために、ストレージドメインをインポートした後に追加の操作をする必要はなく、必要に応じて仮想マシンにアタッチすることができます。

Export

既存のエクスポートストレージドメインをインポートすると、エクスポートストレージドメインに含まれるすべての仮想マシンイメージとテンプレートにアクセスできるようになります。エクスポートドメインは仮想マシンのイメージとテンプレートをエクスポートおよびインポートするように設計されているため、環境内または環境間で少数の仮想マシンとテンプレートを移行するには、エクスポートストレージドメインをインポートすることが推奨されます。エクスポートストレージドメインとの間で仮想マシンとテンプレートをエクスポートおよびインポートする方法については、**Virtual Machine Management Guide**の Exporting and Importing Virtual Machines and Templates を参照してください。

注記

エクスポートストレージドメインは非推奨になりました。ストレージデータドメインはデータセンターから接続を解除し、同じ環境または別の環境にある別のデータセンターにインポートすることができます。仮想マシン、フローティング仮想ディスク、およびテンプレートは、インポートされたストレージドメインからアタッチされたデータセンターにアップロードできます。

警告

ストレージドメインを宛先データセンターに接続すると、新しいストレージドメイン形式にアップグレードされ、ソースデータセンターに再接続されない場合があります。これにより、エクスポートドメインの代わりとしてのデータドメインの使用が中断されます。

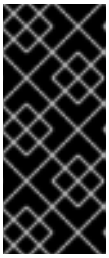
11.7.2. ストレージドメインのインポート

同じ環境または別の環境のデータセンターに以前に接続されていたストレージドメインをインポートし

ます。この手順では、データの破損を防ぐために、ストレージドメインがどの環境のどのデータセンターにも接続されていないことを前提としています。既存のデータストレージドメインをインポートしてデータセンターに接続するには、ターゲットデータセンターを初期化する必要があります。

手順

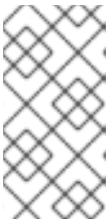
1. **Storage → Domains** をクリックします。
2. **Import Domain** をクリックします。
3. ストレージドメインをインポートする **Data Center** を選択します。
4. ストレージドメインの **Name** を入力します。
5. ドロップダウンリストから **ドメイン機能** および **ストレージタイプ** を選択します。
6. **Host** のドロップダウンリストからホストを選択します。



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。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. ストレージドメインの詳細を入力します。



注記

ストレージドメインの詳細を指定するためのフィールドは、**Domain Function** リストと **Storage Type** リストで選択した値によって異なります。これらのフィールドは、新しいストレージドメインを追加するために使用できるフィールドと同じです。

8. 選択したデータセンターに接続した後にストレージドメインをアクティブ化するには、**Activate Domain in Data Center** チェックボックスをオンにします。
9. **OK** をクリックします。

これで、仮想マシンとテンプレートをストレージドメインからデータセンターにインポートできます。

警告

ストレージドメインを宛先データセンターに接続すると、新しいストレージドメイン形式にアップグレードされ、ソースデータセンターに再接続されない場合があります。これにより、エクスポートドメインの代わりとしてのデータドメインの使用が中断されます。

関連情報

- 「インポート済みデータストレージドメインからの仮想マシンのインポート」
- 「インポートされたデータストレージドメインからのテンプレートのインポート」

11.7.3. 同じ環境内のデータセンター間でのストレージドメインの移行

同じ Red Hat Virtualization 環境内のあるデータセンターから別のデータセンターにストレージドメインを移行して、宛先データセンターがストレージドメインに含まれるデータにアクセスできるようにします。この手順では、ストレージドメインをあるデータセンターから切り離し、別のデータセンターに接続します。

手順

1. 必要なストレージドメインで実行されているすべての仮想マシンをシャットダウンします。
2. **Storage** → **Domains** をクリックします。
3. ストレージドメインの名前をクリックし、詳細ビューを開きます。
4. **Data Center** タブをクリックします。
5. **Maintenance** をクリックしてから **OK** をクリックします。
6. **Detach** をクリックしてから **OK** をクリックします。
7. **Attach** をクリックします。
8. 宛先データセンターを選択し、**OK** をクリックします。

ストレージドメインは宛先データセンターにアタッチされ、自動的にアクティブになります。これで、仮想マシンとテンプレートをストレージドメインから宛先データセンターにインポートできます。

警告

ストレージドメインを宛先データセンターに接続すると、新しいストレージドメイン形式にアップグレードされ、ソースデータセンターに再接続されない場合があります。これにより、エクスポートドメインの代わりとしてのデータドメインの使用が中断されます。

11.7.4. 異なる環境内のデータセンター間でのストレージドメインの移行

ストレージドメインをある Red Hat Virtualization 環境から別の環境に移行して、移行先環境がストレージドメインに含まれるデータにアクセスできるようにします。この手順では、1 つの Red Hat Virtualization 環境からストレージドメインを削除し、それを別の環境にインポートします。既存のデータストレージドメインをインポートして Red Hat Virtualization データセンターに接続するには、ストレージドメインのソースデータセンターに、サポートされている正しい互換性レベルが必要です。詳細は、Supportability and constraints regarding importing Storage Domains and Virtual Machines from older RHV versions を参照してください。

手順

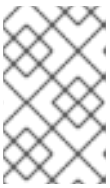
1. ソース環境の管理ポータルにログインします。
2. 必要なストレージドメインで実行されているすべての仮想マシンをシャットダウンします。
3. **Storage** → **Domains** をクリックします。
4. ストレージドメインの名前をクリックし、詳細ビューを開きます。
5. **Data Center** タブをクリックします。
6. **メンテナンス** をクリックします。

重要

Ignore OVF update failure チェックボックスを選択しないでください。ストレージドメインのメンテナーンス操作では、OVF を更新する必要があります。

OK をクリックします。

7. **Detach** をクリックしてから **OK** をクリックします。
8. **Remove** をクリックします。
9. **Remove Storage(s)** ウィンドウで、**Format Domain, i.e. Storage Content will be lost!**チェックボックスが選択されていません。この手順では、後で使用できるようにデータをストレージドメインに保存します。
10. **OK** をクリックすると、ソース環境からストレージドメインが削除されます。
11. 宛先環境の管理ポータルにログインします。
12. **Storage** → **Domains** をクリックします。
13. **Import Domain** をクリックします。
14. **Data Center** ドロップダウンリストから宛先データセンターを選択します。
15. ストレージドメインの名前を入力します。
16. 適切な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。
17. **Host** のドロップダウンリストからホストを選択します。
18. ストレージドメインの詳細を入力します。



注記

ストレージドメインの詳細を指定するためのフィールドは、**Storage Type** ドロップダウンリストで選択した値に応じて変わります。これらのフィールドは、新しいストレージドメインを追加するために使用できるフィールドと同じです。

19. ストレージドメインが接続されたときに自動的にアクティブ化するには、**Activate Domain in Data Center** チェックボックスをオンにします。
20. **OK** をクリックします。

ストレージドメインは、新しい Red Hat Virtualization 環境の宛先データセンターに接続され、自動的にアクティブ化されます。これで、インポートしたストレージドメインから宛先データセンターに、仮想マシンおよびテンプレートをインポートできます。

警告

ストレージドメインを宛先データセンターに接続すると、新しいストレージドメイン形式にアップグレードされ、ソースデータセンターに再接続されない場合があります。これにより、エクスポートドメインの代わりとしてのデータドメインの使用が中断されます。

11.7.5. インポート済みデータストレージドメインからの仮想マシンのインポート

Red Hat Virtualization 環境にインポートしたデータストレージドメインから、1つまたは複数のクラスターに仮想マシンをインポートします。この手順は、インポートされたデータストレージドメインがデータセンターに接続され、アクティブ化されていることを前提としています。

手順

1. **Storage → Domains** をクリックします。
2. インポートされたストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **VM Import** タブをクリックします。
4. インポートする1つ以上の仮想マシンを選択します。
5. **Import** をクリックします。
6. **Import Virtual Machine(s)** ウィンドウのそれぞれの仮想マシンごとに、**Cluster** 一覧で正しいターゲットクラスターが選択されていることを確認します。
7. 外部仮想マシンの vNIC プロファイルを、ターゲットクラスターに存在するプロファイルにマッピングします。
    a. **vNic Profiles Mapping** をクリックします。
    b. **Target vNic Profile** ドロップダウンリストから、使用する vNIC プロファイルを選択します。
    c. **Import Virtual Machine(s)** ウィンドウで複数のターゲットクラスターを選択した場合、**Target Cluster** のドロップダウンリストで各ターゲットクラスターを選択し、マッピングが正しいことを確認します。
    d. **OK** をクリックします。
8. MAC アドレスの競合が検出されると、仮想マシンの名前の横に感嘆符が表示されます。アイコンにマウスをかざして、発生したエラーのタイプを表示するツールチップを表示します。**Reassign Bad MACs** チェックボックスを選択して、新しい MAC アドレスを問題のあるすべての仮想マシンに再割り当てします。または、仮想マシンごとに **Reassign** のチェックボックスを選択できます。

注記

割り当てることができるアドレスがない場合、インポート操作は失敗します。ただし、クラスターの MAC アドレスプール範囲外の MAC アドレスの場合は、新しい MAC アドレスを再割り当てせずに仮想マシンをインポートすることができます。

9. **OK** をクリックします。

インポートされた仮想マシンは、**VM Import** タブの下の一覧に表示されなくなります。

11.7.6. インポートされたデータストレージドメインからのテンプレートのインポート

Red Hat Virtualization 環境にインポートしたデータストレージドメインからテンプレートをインポートします。この手順は、インポートされたデータストレージドメインがデータセンターに接続され、アクティブ化されていることを前提としています。

手順

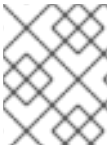
1. **Storage → Domains** をクリックします。
2. インポートされたストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **Template Import** タブをクリックします。
4. インポートする 1 つ以上のテンプレートを選択します。
5. **Import** をクリックします。
6. **Import Templates(s)** ウィンドウの各テンプレートについて、**Cluster** リストで正しいターゲットクラスターが選択されていることを確認します。
7. 外部仮想マシンの vNIC プロファイルを、ターゲットクラスターに存在するプロファイルにマッピングします。
    a. **vNic Profiles Mapping** をクリックします。
    b. **Target vNic Profile** ドロップダウンリストから、使用する vNIC プロファイルを選択します。
    c. **Import Templates** ウィンドウで複数のターゲットクラスターを選択した場合、**Target Cluster** のドロップダウンリストで各ターゲットクラスターを選択し、マッピングが正しいことを確認します。
    d. **OK** をクリックします。
8. **OK** をクリックします。

インポートされたテンプレートは、**Template Import** タブの下のリストに表示されなくなります。

11.8. ストレージタスク

11.8.1. データストレージドメインへのイメージのアップロード

管理ポータルまたは REST API を使用して、仮想ディスクイメージと ISO イメージをデータストレージドメインにアップロードできます。



注記

REST API でイメージをアップロードするには、**REST API Guide** の IMAGETRANSFERS および IMAGETRANSFER を参照してください。

QEMU 互換の仮想ディスクを仮想マシンに接続できます。仮想ディスクのタイプは、QCOW2 または raw のいずれかである必要があります。QCOW2 仮想ディスクから作成されたディスクは共有できません。また、QCOW2 仮想ディスクファイルにバッキングファイルを含めることはできません。

ISO イメージは、CDROM として仮想マシンに添付することも、仮想マシンの起動に使用することもできます。

前提条件

アップロード機能は HTML 5 API を使用します。これには、環境に次のものが必要です。

- **engine-setup** で設定されるイメージ I/O プロキシー (**ovirt-imageio-proxy**)。詳細は、Configuring the Red Hat Virtualization Manager を参照してください。
- 管理ポータルへのアクセスに使用される Web ブラウザーにインポートされた認証局。認証局をインポートするには、**https://engine_address/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA** にアクセスし、すべての信頼設定を有効にします。Firefox、Internet Explorer、または Google Chrome に認証局をインストールする手順を参照してください。
- Firefox 35、Internet Explorer 10、Chrome13 またはそれ以降などの HTML5 をサポートするブラウザー。

データストレージドメインへのイメージのアップロード

1. **Storage → Disks** をクリックします。
2. **Upload** メニューから **Start** を選択します。
3. **Choose File** をクリックして、アップロードするイメージを選択します。
4. **Disk Options** フィールドに入力します。関連するフィールドの説明は、「新しい仮想ディスクウィンドウの設定の説明」 を参照してください。
5. **OK** をクリックします。
進捗バーは、アップロードのステータスを示します。**Upload** メニューから、アップロードを一時停止、キャンセル、または再開できます。

アップロードタイムアウト値の増加

1. アップロードがタイムアウトし、メッセージ**Reason: timeout due to transfer inactivity**が表示される場合は、タイムアウト値を増やします。

```
# engine-config -s TransferImageClientInactivityTimeoutInSeconds=6000
```

2. **ovirt-engine** サービスを再起動します。

```
# systemctl restart ovirt-engine
```

-

11.8.2. ストレージドメインのメンテナンスモードへの移行

ストレージドメインを切り離して削除する前に、ストレージドメインをメンテナーンスモードにする必要があります。これは、別のデータドメインをマスターデータドメインとして再指定するために必要です。

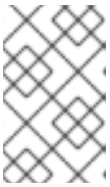
重要

仮想マシンにストレージドメインのリースがある場合、ストレージドメインをメンテナーンスモードに移行することはできません。最初に仮想マシンをシャットダウンするか、リースを削除するか、別のストレージドメインに移動する必要があります。仮想マシンのリースに関する詳細は、Virtual Machine Management Guideを参照してください。

LUN を追加して iSCSI ドメインを拡張できるのは、ドメインがアクティブな場合のみです。

ストレージドメインのメンテナーンスモードへの移行

1. ストレージドメインで実行しているすべての仮想マシンをシャットダウンします。
2. **Storage** → **Domains** をクリックします。
3. ストレージドメインの名前をクリックし、詳細ビューを開きます。
4. **Data Center** タブをクリックします。
5. **メンテナンス** をクリックします。

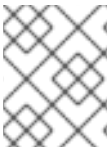


注記

Ignore OVF update failure チェックボックスをオンにすると、OVF 更新が失敗した場合でも、ストレージドメインをメンテナンスモードにすることができます。

6. **OK** をクリックします。

ストレージドメインは非アクティブ化され、結果リストに **Inactive** ステータスが表示されます。これで、データセンターから非アクティブなストレージドメインを編集、デタッチ、削除、または再アクティブ化できます。



注記

また、関連付けられているデータセンターの詳細ビューのストレージタブを使用して、ドメインをアクティブ化し、デタッチし、メンテナンスモードにすることもできます。

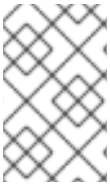
11.8.3. ストレージドメインの編集

管理ポータルからストレージドメインパラメーターを編集できます。ストレージドメインの状態 (アクティブまたは非アクティブ) に応じて、さまざまなフィールドを編集できます。**Data Center**、**Domain Function**、**Storage Type**、および **Format** などのフィールドを変更することはできません。

- **Active**: ストレージドメインがアクティブ状態の場合、**Name**, **Description**、**Comment**、**Warning Low Space Indicator (%)**、**Critical Space Action Blocker**

(GB)、**Wipe After Delete**、および **Discard After Delete** フィールドを編集できます。**Name** フィールドは、ストレージドメインがアクティブな間のみ編集できます。他のすべてのフィールドは、ストレージドメインが非アクティブのときに編集することもできます。

- **Inactive**: ストレージドメインがメンテナンスモードまたはアタッチされていない状態であるため、非アクティブ状態の場合、**Name**、**Data Center**、**Domain Function**、**Storage Type**、および **Format** を除くすべてのフィールドを編集できます。ストレージ接続、マウントオプション、およびその他の高度なパラメーターを編集するには、ストレージドメインを非アクティブにする必要があります。これは、NFS、POSIX、およびローカルストレージタイプでのみサポートされます。



注記

iSCSI ストレージ接続は、管理ポータルを介して編集することはできませんが、REST API を介して編集することができます。**REST API Guide** の Updating Storage Connections を参照してください。

アクティブなストレージドメインの編集

1. **Storage** → **Domains** をクリックし、ストレージドメインを選択します。
2. **Manage Domain** をクリックします。
3. 必要に応じて、使用可能なフィールドを編集します。
4. **OK** をクリックします。

非アクティブなストレージドメインの編集

1. **Storage** → **Domains** をクリックします。
2. ストレージドメインがアクティブな場合は、メンテナンスモードに移行します。
   a. ストレージドメインの名前をクリックし、詳細ビューを開きます。
   b. **Data Center** タブをクリックします。
   c. **メンテナンス** をクリックします。
   d. **OK** をクリックします。
3. **Manage Domain** をクリックします。
4. 必要に応じて、ストレージパスおよびその他の詳細を編集します。新しい接続の詳細は、元の接続と同じストレージタイプである必要があります。
5. **OK** をクリックします。
6. ストレージドメインをアクティブ化します。
   a. ストレージドメインの名前をクリックし、詳細ビューを開きます。
   b. **Data Center** タブをクリックします。
   c. **アクティブ化** クリックします。

11.8.4. OVF の更新

デフォルトでは、OVF は 60 分ごとに更新されます。ただし、重要な仮想マシンをインポートした場合、または重要な更新を行った場合は、OVF を手動で更新できます。

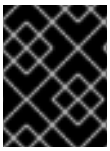
OVF の更新

1. **Storage** → **Domains** をクリックします。
2. ストレージドメインを選択し、**More Actions** (⋮) をクリックしてから、**Update OVFs** をクリックします。
   OVF が更新され、メッセージが **Events** に表示されます。

11.8.5. メンテナンスモードからのストレージドメインのアクティブ化

データセンターのストレージに変更を加えている場合は、ストレージドメインをメンテナンスモードにする必要があります。ストレージドメインをアクティブ化して、使用を再開します。

1. **Storage** → **Domains** をクリックします。
2. アクティブではないストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **Data Centers** タブをクリックします。
4. **アクティブ化** クリックします。



重要

データドメインをアクティブ化する前に ISO ドメインをアクティブ化しようとすると、エラーメッセージが表示され、ドメインはアクティブ化されません。

11.8.6. データセンターからストレージドメインをデタッチ

あるデータセンターからストレージドメインをデタッチして、別のデータセンターに移行します。

データセンターからストレージドメインをデタッチ

1. **Storage** → **Domains** をクリックします。
2. ストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **Data Center** タブをクリックします。
4. **メンテナンス** をクリックします。
5. **OK** をクリックしてメンテナーンスモードを開始します。
6. **デタッチ** をクリックします。
7. **OK** をクリックして、ストレージドメインを切り離します。

ストレージドメインがデータセンターから切り離され、別のデータセンターに接続できるようになりました。

11.8.7. ストレージドメインのデータセンターへのアタッチ

ストレージドメインをデータセンターに接続します。

ストレージドメインのデータセンターへのアタッチ

1. **Storage** → **Domains** をクリックします。
2. ストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **Data Center** タブをクリックします。
4. **Attach** をクリックします。
5. 適切なデータセンターを選択します。
6. **OK** をクリックします。

ストレージドメインはデータセンターにアタッチされ、自動的にアクティブになります。

11.8.8. ストレージドメインの削除

データセンターに、仮想化環境から削除するストレージドメインがあります。

手順

1. **Storage** → **Domains** をクリックします。
2. ストレージドメインをメンテナンスモードに移動し、デタッチします。
   a. ストレージドメインの名前をクリックし、詳細ビューを開きます。
   b. **Data Center** タブをクリックします。
   c. **Maintenance** をクリックしてから **OK** をクリックします。
   d. **Detach** をクリックしてから **OK** をクリックします。
3. **Remove** をクリックします。
4. 任意で **Format Domain, i.e. Storage Content will be lost!**チェックボックスを選択して、ドメインのコンテンツを消去します。
5. **OK** をクリックします。

ストレージドメインは環境から完全に削除されます。

11.8.9. ストレージドメインの破棄

エラーが発生したストレージドメインは、通常の手順では削除できない場合があります。ストレージドメインを破棄すると、仮想化環境からストレージドメインが強制的に削除されます。

ストレージドメインの破棄

1. **Storage** → **Domains** をクリックします。
2. ストレージドメインを選択し、**More Actions** (⋮) をクリックしてから **Destroy** をクリックします。

3. **Approve operation** チェックボックスを選択します。
4. **OK** をクリックします。

11.8.10. ディスクプロファイルの作成

ディスクプロファイルは、ストレージドメイン内の仮想ディスクのスループットの最大レベルと入出力操作の最大レベルを定義します。ディスクプロファイルは、データセンターで定義されたストレージプロファイルに基づいて作成され、プロファイルを有効にするには、個々の仮想ディスクに手動で割り当てる必要があります。

この手順は、ストレージドメインが属するデータセンターの下に 1 つ以上のストレージサービス品質エントリーがすでに定義されていることを前提としています。

ディスクプロファイルの作成

1. **Storage → Domains** をクリックします。
2. データストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **Disk Profiles** タブをクリックします。
4. **New** をクリックします。
5. ディスクプロファイルの **名前** と **説明** を入力します。
6. **QoS** 一覧からディスクプロファイルに適用するサービスの品質を選択します。
7. **OK** をクリックします。

11.8.11. ディスクプロファイルの削除

Red Hat Virtualization 環境から既存のディスクプロファイルを削除します。

ディスクプロファイルの削除

1. **Storage → Domains** をクリックします。
2. データストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **Disk Profiles** タブをクリックします。
4. 削除するディスクプロファイルを選択します。
5. **Remove** をクリックします。
6. **OK** をクリックします。

ディスクプロファイルがいずれかの仮想ディスクに割り当てられている場合、ディスクプロファイルはそれらの仮想ディスクから削除されます。

11.8.12. ストレージドメインのヘルスステータスの表示

ストレージドメインには、通常の **ステータス** に加えて、外部のヘルスステータスがあります。外部の正常状態は、プラグインや外部システムから報告されたり、管理者が設定したりするもので、ストレージドメインの **Name** の左側に以下のいずれかのアイコンとして表示されます。

- OK: アイコンなし
- info: 
- Warning: 
- Error: 
- Failure: 

ストレージドメインのヘルスステータスの詳細を表示するには、ストレージドメインの名前をクリックして詳細ビューを開き、**Events** タブをクリックします。

ストレージドメインのヘルスステータスは、REST API を使用して表示することもできます。ストレージドメインでの **GET** リクエストには、ヘルスステータスを含む **external_status** 要素が含まれます。

events コレクションを介して、REST API でストレージドメインのヘルスステータスを設定できます。詳細は、**REST API Guide** の Adding Events を参照してください。

11.8.13. ストレージドメインの削除後に破棄の設定

Discard After Delete チェックボックスがオンになっている場合は、論理ボリュームが削除されると **blkdiscard** コマンドが呼び出され、ブロックが解放されたことが基になるストレージに通知されます。ストレージアレイは、解放されたスペースを使用して、要求に応じて割り当てることができます。**Discard After Delete** は、ブロックストレージでのみ機能します。このフラグは、NFS などのファイルストレージ用の Red Hat Virtualization Manager では使用できません。

制限:

- **Discard After Delete** は、iSCSI やファイバーチャネルなどのブロックストレージドメインでのみ使用できます。
- 基盤となるストレージは **Discard** をサポートする必要があります。

Discard After Delete は、ブロックストレージドメインを作成するとき、またはブロックストレージドメインを編集するときに有効にできます。Preparing and Adding Block Storage および Editing Storage Domains を参照してください。

11.8.14. 250 を超えるホストがある環境で 4K サポートを有効にする

デフォルトでは、GlusterFS ドメインとローカルストレージドメインは、最大 250 のホストを備えた Red Hat Virtualization 環境で 4K ブロックサイズをサポートします。4K ブロックサイズを使用すると、特に大きなファイルを使用する場合などに、パフォーマンスが向上します。また、VDO などの 4K 互換性を必要とするツールを使用する場合にも必要です。

ホストの最大数がデフォルトの 250 の場合、Sanlock が割り当てるロックスペース領域は 1MB です。4K ストレージを使用するときにホストの最大数を増やすと、ロックスペース領域が大きくなります。たとえば、2000 のホストを使用する場合、ロックスペース領域は最大 8MB になる可能性があります。

エンジン設定パラメーター **MaxNumberOfHostsInStoragePool** を設定することにより、250 を超えるホストがある環境で 4K ブロックのサポートを有効にできます。

手順

1. Manager マシンで、必要な最大数のホストを有効にします。

```
# engine-config -s MaxNumberOfHostsInStoragePool=NUMBER_OF_HOSTS
```

2. JBoss Application Server を再起動します。

```
# service jboss-as restart
```

たとえば、300 のホストを持つクラスターがある場合は、次のように入力します。

```
# engine-config -s MaxNumberOfHostsInStoragePool=300
# service jboss-as restart
```

検証

Manager で **MaxNumberOfHostsInStoragePool** パラメーターの値を表示します。

```
# engine-config --get=MaxNumberOfHostsInStoragePool
MaxNumberOfHostsInStoragePool: 250 version: general
```

11.8.15. 4K サポートの無効化

デフォルトでは、GlusterFS ドメインとローカルストレージドメインは 4K ブロックサイズをサポートします。4K ブロックサイズを使用すると、特に大きなファイルを使用する場合などに、パフォーマンスが向上します。また、VDO などの 4K 互換性を必要とするツールを使用する場合にも必要です。

4K ブロックのサポートを無効にすることができます。

手順

1. 4K ブロックのサポートが有効になっていることを確認してください。

```
$ vdsm-client Host getCapabilities
…
{
   "GLUSTERFS" : [
       0,
       512,
       4096,
   ]
   …
```

2. **/etc/vdsm/vdsm.conf.d/gluster.conf** を編集し、**enable_4k_storage** を **false** に設定します。以下に例を示します。

```
$ vi  /etc/vdsm/vdsm.conf.d/gluster.conf

[gluster]
# Use to disable 4k support
# if needed.
enable_4k_storage = false
```

第12章 POOLS

12.1. 仮想マシンプールの概要

仮想マシンプールは、すべて同じテンプレートのクローンであり、特定のグループ内の任意のユーザーがオンデマンドで使用できる仮想マシンのグループです。仮想マシンプールを使用すると、管理者はユーザー向けに一連の一般化された仮想マシンを迅速に設定できます。

ユーザーは、プールから仮想マシンを取得することにより、仮想マシンプールにアクセスします。ユーザーがプールから仮想マシンを取得すると、プール内の仮想マシン (使用可能な場合) のいずれかが提供されます。その仮想マシンは、プールのベースとなったテンプレートと同じオペレーティングシステムと設定を持ちますが、ユーザーが仮想マシンを使用するたびにプールの同じメンバーを受け取るとは限りません。ユーザーは、プールの設定に応じて、同じ仮想マシンプールから複数の仮想マシンを取得することもできます。

仮想マシンプールはデフォルトでステートレスです。つまり、仮想マシンのデータと設定の変更は再起動後も永続的ではありません。ただし、プールはステートフルになるように設定できるため、前のユーザーが行った変更を保持できます。ただし、ユーザーが仮想マシンプールから取得した仮想マシンのコンソールオプションを設定する場合、それらのオプションは、その仮想マシンプールのそのユーザーのデフォルトとして設定されます。

注記

プールから取得した仮想マシンは、管理ポータルからアクセスしたときにステートレスではありません。これは、管理者が必要に応じてディスクに変更を書き込める必要があるためです。

原則として、プール内の仮想マシンは、ユーザーが取得したときに起動され、ユーザーが終了したときにシャットダウンされます。ただし、仮想マシンプールには、事前に起動した仮想マシンを含めることもできます。事前に起動した仮想マシンは稼働状態に保たれ、ユーザーが使用するまでアイドル状態のままになります。これにより、ユーザーはそのような仮想マシンの使用をすぐに開始できますが、これらの仮想マシンは、アイドル状態のために使用されていないときでもシステムリソースを消費します。

12.2. 仮想マシンプールの作成

共通のテンプレートに基づいて、複数の仮想マシンを含む仮想マシンプールを作成できます。仮想マシンのシーリングおよびテンプレートの作成についての詳細は、**Virtual Machine Management Guide**の Templates を参照してください。

Windows 仮想マシンの Sysprep ファイル設定オプション

要件に応じて、いくつかの **sysprep** ファイル設定オプションを使用できます。

プールがドメインに参加する必要がない場合は、**/usr/share/ovirt-engine/conf/sysprep/** にあるデフォルトの **sysprep** ファイルを使用できます。

プールをドメインに参加させる必要がある場合は、Windows オペレーティングシステムごとにカスタム **sysprep** を作成できます。

1. 各オペレーティングシステムに関連するセクションを **/usr/share/ovirt-engine/conf/osinfo-defaults.properties** から新しいファイルにコピーし、**99-defaults.properties** として保存します。

2. **99-defaults.properties** で、Windows プロダクトアクティベーションキーと新しいカスタム **sysprep** ファイルのパスを指定します。

```
os.operating_system.productKey.value=Windows_product_activation_key
...
os.operating_system.sysprepPath.value =
${ENGINE_USR}/conf/sysprep/sysprep.operating_system
```

3. ドメイン、ドメインパスワード、およびドメイン管理者を指定して、新しい **sysprep** ファイルを作成します。

```
<Credentials>
   <Domain>AD_Domain</Domain>
   <Password>Domain_Password</Password>
   <Username>Domain_Administrator</Username>
</Credentials>
```

Windows 仮想マシンのさまざまなプール用に異なる **sysprep** 設定が必要な場合は、管理ポータルにカスタムの **sysprep** ファイルを作成することができます (以下の Creating a Virtual Machine Pool を参照)。詳細は、**Virtual Machine Guide** の Using Sysprep to Automate the Configuration of Virtual Machines を参照してください。

仮想マシンプールの作成

1. **Compute** → **Pools** 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から **クラスター** を選択します。
4. ドロップダウンメニューから **テンプレート** とバージョンを選択します。テンプレートは、プール内のすべての仮想マシンの標準設定を提供します。
5. ドロップダウンリストから **Operating System** を選択します。
6. **Optimized for** ドロップダウンリストを使用して、**Desktop** または **Server** の仮想マシンを最適化します。

注記

高性能仮想マシンは単一のホストと具体的なリソースに固定されているため、**高性能** 最適化はプールには推奨されません。このような設定の複数の仮想マシンを含むプールは、適切に実行されません。

7. **名前** を入力し、オプションで **説明** と **コメント** を入力します。
プールの **名前** は、数値の接尾辞を付けて、プール内の各仮想マシンに適用されます。仮想マシンの番号付けは、プレースホルダーとして **?** を使用してカスタマイズできます。

例12.1 プール名と仮想マシンの番号付けの例

- プール: **MyPool**
仮想マシン: **MyPool-1**、**MyPool-2**、... **MyPool-10**
- プール: **MyPool-???**
仮想マシン: **MyPool-001**、**MyPool-002**、... **MyPool-010**

8. プールの **VM の数** を入力します。
9. **Prestarted** フィールドに、事前起動する仮想マシンの数を入力します。
10. 1 人のユーザーがセッションで実行できる **1 ユーザーあたりの仮想マシンの最大数** を 選択します。最小値は 1 です。
11. 削除保護を有効にするには、**保護の削除** チェックボックスをオンにします。
12. Windows 以外の仮想マシンのプールを作成している場合、またはデフォルトの **sysprep** を使用している場合は、この手順をスキップしてください。Windows 仮想マシンのプール用のカスタム **sysprep** ファイルを作成する場合:
    a. **Show Advanced Options** ボタンをクリックします。
    b. **Initial Run** タブをクリックし、**Use Cloud-Init/Sysprep** チェックボックスを選択します。
    c. **Authentication** 矢印をクリックして **User Name** と **Password** を入力するか、**Use already configured password** を選択します。

注記

この **User Name** は、ローカル管理者の名前です。この値は、**Authentication** セクションまたはカスタム **sysprep** ファイルでデフォルト値 (**user**) から変更できます。

    d. **Custom Script** の矢印をクリックして、**/usr/share/ovirt-engine/conf/sysprep/** にあるデフォルトの **sysprep** ファイルの内容をテキストボックスに貼り付けます。
    e. **sysprep** ファイルの次の値を変更できます。

- **Key**.事前定義された Windows アクティベーションプロダクトキーを使用しない場合は、**<![CDATA[$ProductKey$]]>** を有効なプロダクトキーに置き換えてください。

```
<ProductKey>
  <Key><![CDATA[$ProductKey$]]></Key>
</ProductKey>
```

例12.2 Windows のプロダクトキーの例

```
<ProductKey>
  <Key>0000-000-000-000</Key>
</ProductKey>
```

- Windows 仮想マシンが参加する **Domain**、ドメインの **Password**、およびドメイン管理者の **Username**:

```
<Credentials>
  <Domain>AD_Domain</Domain>
  <Password>Domain_Password</Password>
  <Username>Domain_Administrator</Username>
</Credentials>
```

例12.3 ドメイン認証情報の例

```
<Credentials>
   <Domain>addomain.local</Domain>
   <Password>12345678</Password>
   <Username>Sarah_Smith</Username>
</Credentials>
```

注記

ドメインに参加するには、**Domain**、**Password**、および **Username** が必要です。**キー** はアクティベーション用です。必ずしも両方が必要なわけではありません。

ドメインと認証情報は、**Initial Run** タブでは変更できません。

- ローカル管理者の **FullName**:

```
<UserData>
…
   <FullName>Local_Administrator</FullName>
…
</UserData>
```

- **DisplayName** とローカル管理者の **名前**:

```
<LocalAccounts>
   <LocalAccount wcm:action="add">
      <Password>
         <Value><![CDATA[$AdminPassword$]]></Value>
         <PlainText>true</PlainText>
      </Password>
      <DisplayName>Local_Administrator</DisplayName>
      <Group>administrators</Group>
      <Name>Local_Administrator</Name>
   </LocalAccount>
</LocalAccounts>
```

sysprep ファイルの残りの変数は、**Initial Run** タブで入力できます。

13. オプション。**Pool Type** を設定します。

    a. **Type** タブをクリックして、**Pool Type** を選択します。

    - **Manual** - 管理者は、仮想マシンをプールに明示的に戻す責任があります。
    - **Automatic** - 仮想マシンは自動的に仮想マシンプールに戻されます。

    b. 仮想マシンがステートフルモードで開始されるようにするには、**Stateful Pool** チェックボックスをオンにします。これにより、前のユーザーが行った変更が仮想マシンに保持されます。

    c. **OK** をクリックします。

14. オプション。SPICE プロキシーをオーバーライドします。

    a. **Console** タブで、**Override SPICE Proxy** チェックボックスをオンにします。

    b. **Overridden SPICE proxy address** テキストフィールドで、グローバル SPICE プロキシーをオーバーライドする SPICE プロキシーのアドレスを指定します。

    c. **OK** をクリックします。

15. Windows 仮想マシンのプールの場合は、**Compute → Virtual Machines** をクリックして、プールから各仮想マシンを選択し、**Run → Run Once** をクリックします。

注記

仮想マシンが起動せず、**Info [windeploy.exe] Found no unattend file** が **%WINDIR%\panther\UnattendGC\setupact.log** に表示されない場合は、作成に使用された Windows 仮想マシンのレジストリーに **UnattendFile** キーを追加して、プールのテンプレートを作成します。

1. Windows 仮想マシンに、**A:\Unattend.xml** などの無人ファイルを含むフロッピーデバイスが接続されていることを確認します。
2. **Start** をクリックし、**Run** をクリックして、**Open** テキストボックスに **regedit** と入力して、**OK** をクリックします。
3. 左側のペインで、**HKEY_LOCAL_MACHINE → SYSTEM → Setup** に移動します。
4. 右ペインを右クリックして、**New → String Value** を選択します。
5. キー名として **UnattendFile** を入力します。
6. 新しいキーをダブルクリックし、キーの値として **unattend** ファイルの名前とパス (**A:\Unattend.xml** など) を入力します。
7. レジストリーを保存し、Windows 仮想マシンを封印して、新しいテンプレートを作成します。詳細は、**Virtual Machine Management Guide**の Templates を参照してください。

指定した数の同一の仮想マシンを使用して仮想マシンプールを作成および設定しました。これらの仮想マシンは、**Compute → Virtual Machines** で表示するか、プールの名前をクリックして詳細ビューを開くことで表示できます。プール内の仮想マシンは、アイコンにより独立した仮想マシンと区別されます。

12.3. 新しいプールとプールの編集ウィンドウの設定およびコントロールの説明

12.3.1. 新しいプールと編集プールの一般設定の説明

次の表に、仮想マシンプールに固有の **New Pool** ウィンドウと **Edit Pool** ウィンドウの **General** タブに必要な情報の詳細を示します。他のすべての設定は、**New Virtual Machine** ウィンドウの設定と同じです。

表12.1一般設定

フィールド名	説明
Template	仮想マシンプールのベースとなるテンプレートおよびテンプレートサブバージョン。テンプレートの **latest** サブバージョンに基づいてプールを作成する場合、プール内のすべての仮想マシンは、再起動すると、最新のテンプレートバージョンを自動的に受け取ります。仮想マシンのテンプレートの設定に関する詳細は、**Virtual Machine Management Guide** の Virtual Machine General Settings Explained および Explanation of Settings in the New Template and Edit Template Windows を参照してください。
説明	仮想マシンプールの意味のある説明。
Comment	仮想マシンプールに関するプレーンテキストの人間が判読できるコメントを追加するフィールド。
Prestarted VMs	仮想マシンプール内の仮想マシンが取得される前に開始され、ユーザーが取得する状態に維持される仮想マシンの数を指定できます。このフィールドの値は、**0** から仮想マシンプール内の仮想マシンの総数の間でなければなりません。
VM の数/プール内の VM の数を次のように増やす	作成して仮想マシンプールで使用できるようにする仮想マシンの数を指定できます。編集ウィンドウでは、仮想マシンプール内の仮想マシンの数を指定された数だけ増やすことができます。デフォルトでは、プールに作成できる仮想マシンの最大数は 1000 です。この値は、**engine-config** コマンドの **MaxVmsInPool** キーを使用して設定できます。
ユーザーあたりの VM の最大数	1 人のユーザーが一度に仮想マシンプールから取得できる仮想マシンの最大数を指定できます。このフィールドの値は、**1** から **32,767** の間でなければなりません。
Delete Protection	プール内の仮想マシンが削除されないようにすることができます。

12.3.2. 新しいプールおよびプールタイプ設定の編集の説明

次の表に、New Pool ウィンドウと Edit Pool ウィンドウの Type タブに必要な情報の詳細を示します。

表12.2 Type 設定

フィールド名	説明

フィールド名	説明
Pool Type	このドロップダウンメニューでは、仮想マシンプールのタイプを指定できます。以下のオプションを設定できます。 ● **自動**: ユーザーが仮想マシンプールから取得した仮想マシンの使用を終了すると、その仮想マシンは自動的に仮想マシンプールに戻されます。 ● **手動**: ユーザーが仮想マシンプールから取得した仮想マシンの使用を終了した後、管理者が手動で仮想マシンを返却した場合にのみ、その仮想マシンは仮想マシンプールに返却されます。
Stateful Pool	仮想マシンが別のユーザーに渡されたときに、プール内の仮想マシンの状態を保持するかどうかを指定します。これは、前のユーザーが行った変更が仮想マシンに保持されることを意味します。

12.3.3. 新しいプールおよびプールコンソール設定の編集の説明

次の表に、仮想マシンプールに固有の **New Pool** ウィンドウまたはプールの **Edit Pool** ウィンドウの **Console** タブに必要な情報の詳細を示します。他のすべての設定は、**New Virtual Machine** ウィンドウと **Edit Virtual Machine** ウィンドウの設定と同じです。

表12.3 コンソール の設定

フィールド名	説明
Override SPICE proxy	このチェックボックスを選択し、グローバル設定で定義された SPICE プロキシーの上書きを有効にします。この機能は、ユーザー (たとえば、仮想マシンポータル経由で接続する) がホストが存在するネットワーク外にある場合に役に立ちます。
SPICE プロキシーアドレスのオーバーライド	SPICE クライアントが仮想マシンに接続するプロキシーこのプロキシーは、Red Hat Virtualization 環境用に定義されたグローバル SPICE プロキシーと、仮想マシンプールが属するクラスター用に定義された SPICE プロキシー (存在する場合) の両方をオーバーライドします。アドレスは以下の形式でなければなりません。 `protocol://host:port`

12.3.4. 仮想マシンプールのホスト設定の説明

次の表に、**New Pool** ウィンドウおよび **Edit Pool** ウィンドウの **Host** タブで使用できるオプションの詳細を示します。

表12.4 仮想マシンプール: ホスト設定

フィールド名	サブ要素	説明
Start Running On		仮想マシンを実行する優先ホストを定義します。以下のいずれかを選択します。 ● **Any Host in Cluster** - 仮想マシンは、クラスター内の利用可能な任意のホストで起動し、実行できます。 ● **Specific Host(s)** - 仮想マシンはクラスター内の特定のホストで実行を開始します。ただし、Manager または管理者は、仮想マシンの移行および高可用性設定に応じて、仮想マシンをクラスター内の別のホストに移行することができます。利用可能なホスト一覧から、特定のホストまたはホストのグループを選択します。
Migration Options	Migration mode	仮想マシンの実行および移行オプションを定義します。このオプションを使用しない場合、仮想マシンはそのクラスターのポリシーに従って実行または移行されます。 ● **Allow manual and automatic migration**: 環境のステータスに応じて自動的に、または管理者により手動で、仮想マシンをあるホストから別のホストに移行することができます。 ● **Allow manual migration only**: 仮想マシンは、管理者によって手動でしか、あるホストから別のホストに移行できません。 ● **Do not allow migration**: 仮想マシンを自動または手動いずれでも移行することはできません。

フィールド名	サブ要素	説明
	Use custom migration policy	移行コンバージェンスポリシーを定義します。チェックボックスをオフのままにすると、ホストがポリシーを決定します。 • **legacy**: 3.6 バージョンのレガシー動作**vdsm.conf** のオーバーライドは引き続き適用されます。ゲストエージェントフックメカニズムが無効になっている。 • **Minimal downtime**: 一般的な状況で仮想マシンを移行できるようにします。仮想マシンは、ダウンタイムを大幅に発生しません。移行は、長時間 (QEMU の反復により最大 500 ミリ秒) 後に仮想マシンの移行が収束されない場合に中止されます。ゲストエージェントフックメカニズムが有効になっている。 • **Suspend workload if needed**: 仮想マシンが負荷の高いワークロードを実行している場合 など、ほとんどの状況で仮想マシンを移行できるようにします。このため、仮想マシンでは、他の設定よりも大きなダウンタイムが生じる場合があります。移行は、極端なワークロードに対して中止される場合があります。ゲストエージェントフックメカニズムが有効になっている。
	Use custom migration downtime	このチェックボックスを選択すると、ライブマイグレーション中に仮想マシンがダウンできる最大期間をミリ秒単位で指定できます。ワークロードおよび SLA の要件に従って、各仮想マシンに異なる最大ダウンタイムを設定します。VDSM のデフォルト値を使用するには **0** を入力します。

フィールド名	サブ要素	説明
	自動コンバージョン	**Legacy** 移行ポリシーでのみアクティベートされます。仮想マシンのライブマイグレーション中に自動コンバージェンスが使用されるかどうかを設定できます。負荷が大きいが大きい仮想マシンでは、ライブマイグレーション中に行われる転送速度よりも速くメモリーがダーティーなり、移行が収束できなくなります。QEMU の自動調整機能を使用すると、仮想マシン移行の収束を強制的に実行できます。QEMU は、コンバージェンスの欠如を自動的に検出し、仮想マシン上の vCPU のスロットルダウンをトリガーします。オートコンバージェンスはデフォルトで無効になっています。 ● クラスターレベルで設定される自動収束設定を使用するには、**Inherit from global setting** を選択します。このオプションはデフォルトで選択されます。 ● クラスター設定またはグローバル設定を上書きし、仮想マシンの自動収束を許可するには、**Auto Converge** を選択します。 ● クラスター設定またはグローバル設定を上書きし、仮想マシンの自動収束を防ぐには、**Don't Auto Converge** を選択します。

フィールド名	サブ要素	説明
	移行圧縮の有効化	**Legacy** 移行ポリシーでのみアクティベートされます。このオプションを使用すると、仮想マシンのライブマイグレーション中に移行圧縮を使用するかどうかを設定できます。この機能は、Xor Binary Zero Run-Length-Encoding を使用して、メモリー書き込みを必要とするワークロードまたはスパースメモリー更新パターンを使用するアプリケーションに対して、仮想マシンのダウンタイムと合計移行時間を短縮します。移行圧縮は、デフォルトでは無効になっています。 ● クラスターレベルで設定される圧縮設定を使用するには、**Inherit from global setting** を選択します。このオプションはデフォルトで選択されます。 ● クラスター設定またはグローバル設定を上書きし、仮想マシンの圧縮を許可するには、**Compress** を選択します。 ● クラスター設定またはグローバル設定を上書きし、仮想マシンの圧縮を防ぐには、**Don't compress** を選択します。
	パススルーホスト CPU	このチェックボックスを選択すると、仮想マシンは配置されているホストの物理 CPU の機能を利用できます。
Configure NUMA	NUMA Node Count	仮想マシンに割り当てる仮想 NUMA ノードの数。**Tune Mode** が **Preferred** の場合、この値を **1** に設定する必要があります。

フィールド名	サブ要素	説明
	Tune Mode	メモリーを割り当てるために使用されるメソッド。 ● **Strict**: メモリーをターゲットノードに割り当てできない場合には、メモリーの割り当てに失敗します。 ● **Preferred**: メモリーは、1 つの優先ノードから割り当てられます。十分なメモリーが利用できない場合は、他のノードからメモリーを割り当てることができます。 ● **Interleave**: メモリーはラウンドロビンアルゴリズムで全ノードに割り当てられます。
	NUMA Pinning	**NUMA Topology** ウィンドウを開きます。このウィンドウには、ホストの合計 CPU、メモリー、NUMA ノード、および仮想マシンの仮想 NUMA ノードが表示されます。右側のボックスから左側の NUMA ノードに各 vNUMA をクリックアンドドラッグすることで、仮想 NUMA ノードをホストの NUMA ノードに固定します。

12.3.5. 新しいプールとプールのリソース割り当て設定の編集の説明

次の表に、仮想マシンプールに固有の **New Pool** ウィンドウと **Edit Pool** ウィンドウの **Resource Allocation** タブに必要な情報の詳細を示します。他のすべての設定は、**New Virtual Machine** ウィンドウの設定と同じです。詳細は、**Virtual Machine Management Guide**の Virtual Machine Resource Allocation Settings Explained を参照してください。

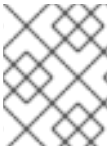
表12.5 Resource Allocation の設定

フィールド名	サブ要素	説明
Disk Allocation	Auto select target	このチェックボックスをオンにすると、空き容量が最も多いストレージドメインが自動的に選択されます。**ターゲット** フィールドと **ディスクプロファイル** フィールドは無効になっています。

フィールド名	サブ要素	説明
	形式	このフィールドは読み取り専用で、ストレージドメインタイプが OpenStack ボリューム (Cinder) でない限り常に **QCOW2** と表示します。この場合の形式は **Raw** になります。

12.4. 仮想マシンプールの編集

仮想マシンプールが作成された後、そのプロパティーを編集できます。仮想マシンプールの編集時に使用できるプロパティーは、新しい仮想マシンプールの作成時に使用できるプロパティーと同じですが、**Number of VMs** プロパティーが **Increase number of VMs in pool by**に置き換えられている点が異なります。



注記

仮想マシンプールを編集する場合、導入された変更は新しい仮想マシンにのみ影響します。導入された変更の時点ですでに存在していた仮想マシンは影響を受けません。

仮想マシンプールの編集

1. **Compute → Pools** をクリックして、仮想マシンプールを選択します。
2. **Edit** をクリックします。
3. 仮想マシンプールのプロパティーを編集します。
4. **OK** をクリックします。

12.5. プール内の仮想マシンの事前起動

仮想マシンプール内の仮想マシンは、デフォルトでパワーダウンされています。ユーザーがプールから仮想マシンを要求すると、マシンの電源がオンになり、ユーザーに割り当てられます。対照的に、事前に起動した仮想マシンはすでに実行しており、ユーザーへの割り当てを待機しているため、ユーザーがマシンにアクセスできるようになるまで待機する時間が短縮されます。事前に起動した仮想マシンがシャットダウンすると、プールに戻され、元の状態に復元されます。事前に起動した仮想マシンの最大数は、プール内の仮想マシンの数です。

事前に起動した仮想マシンは、ユーザーが特に割り当てられていない仮想マシンにすぐにアクセスする必要がある環境に適しています。自動プールのみが仮想マシンを事前に起動できます。

プール内の仮想マシンの事前起動

1. **Compute → Pools** をクリックして、仮想マシンプールを選択します。
2. **Edit** をクリックします。
3. **Prestarted VMs** フィールドに、事前起動する仮想マシンの数を入力します。

4. **Type** タブをクリックします。**Pool Type** が **Automatic** に設定されていることを確認します。
5. **OK** をクリックします。

12.6. 仮想マシンプールへの仮想マシンの追加

仮想マシンプールで最初にプロビジョニングされた数よりも多くの仮想マシンが必要な場合は、プールにマシンを追加します。

仮想マシンプールへの仮想マシンの追加

1. **Compute → Pools** をクリックして、仮想マシンプールを選択します。
2. **Edit** をクリックします。
3. **Increase number of VMs in pool by**フィールドに、追加の仮想マシンの数を入力します。
4. **OK** をクリックします。

12.7. 仮想マシンプールからの仮想マシンのデタッチ

仮想マシンを仮想マシンプールからデタッチできます。仮想マシンを切り離すと、その仮想マシンがプールから削除され、独立した仮想マシンになります。

仮想マシンプールからの仮想マシンのデタッチ

1. **Compute → Pools** をクリックします。
2. プールの名前をクリックして、詳細ビューを開きます。
3. **Virtual Machines** タブをクリックして、プール内の仮想マシンを一覧表示します。
4. 仮想マシンのステータスが **Down** であることを確認します。実行中の仮想マシンをデタッチすることはできません。
5. 1 つ以上の仮想マシンを選択し、**Detach** をクリックします。
6. **OK** をクリックします。

注記

仮想マシンは環境内に存在したままで、**Compute → Virtual Machines** から表示およびアクセスできます。アイコンが変化して、デタッチされた仮想マシンが独立した仮想マシンであることを示すことに注意してください。

12.8. 仮想マシンプールの削除

データセンターから仮想マシンプールを削除できます。最初に、プール内のすべての仮想マシンを削除またはデタッチする必要があります。プールから仮想マシンを切り離すと、それらは独立した仮想マシンとして保持されます。

仮想マシンプールの削除

1. **Compute → Pools** をクリックして、仮想マシンプールを選択します。

2. **Remove** をクリックします。

3. **OK** をクリックします。

12.9. 信頼できるコンピュートプール

信頼できるコンピュートプールは、Intel Trusted Execution Technology (Intel TXT) に基づくセキュアなクラスターです。信頼できるクラスターは、ホワイトリストデータベースに対してホストのハードウェアとソフトウェアの整合性を測定する Intel の OpenAttestation によって検証されたホストのみを許可します。信頼できるホストおよびそれらで実行されている仮想マシンには、高いレベルでセキュリティーを確保する必要があるタスクを割り当てることができます。Intel TXT、信頼できるシステム、およびテストの詳細は、https://software.intel.com/en-us/articles/intel-trusted-execution-technology-intel-txt-enabling-guide を参照してください。

信頼できるコンピュートプールを作成するには、以下の手順を行います。

- Manager が OpenAttestation サーバーと通信するように設定する。
- 信頼できるクラスターを作成して、信頼できるホストだけを実行できるようにする。
- 信頼できるホストを信頼できるクラスターに追加する。OpenAttestation サーバーによって信頼されていることを確認するには、ホストが OpenAttestation エージェントを実行している必要があります。

12.9.1. OpenAttestation サーバーの Manager への接続

信頼されたクラスターを作成する前に、Red Hat Virtualization Manager が OpenAttestation サーバーを認識するように設定する必要があります。**engine-config** を使用して OpenAttestation サーバーの FQDN または IP アドレスを追加します。

```
# engine-config -s AttestationServer=attestationserver.example.com
```

必要に応じて、以下の設定を変更することもできます。

表12.6 engine-config の OpenAttestation 設定

オプション	デフォルト値	説明
AttestationServer	oat-server	OpenAttestation サーバーの FQDN または IP アドレス。これは、Manager が OpenAttestation サーバーと通信するように設定する必要があります。
AttestationPort	8443	Manager との通信に OpenAttestation サーバーが使用するポート。
AttestationTruststore	TrustStore.jks	OpenAttestation サーバーとの通信のセキュリティーを保護するために使用されるトラストストア。

オプション	デフォルト値	説明
AttestationTruststorePass	password	トラストストアのアクセスに使用されるパスワード。
AttestationFirstStageSize	10	クイック初期化に使用されます。この値は、適切な理由なしで変更することは推奨されません。
SecureConnectionWithOATServers	true	OpenAttestation サーバーとのセキュアな通信を有効または無効にします。
PollUri	AttestationService/resources/PollHosts	OpenAttestation サービスへのアクセスに使用される URI。

12.9.2. 信頼できるクラスターの作成

信頼できるクラスターは OpenAttestation サーバーと通信して、ホストのセキュリティーを評価します。ホストが信頼できるクラスターに追加されると、OpenAttestation サーバーは、ホワイトリストデータベースに対して、ホストのハードウェアおよびソフトウェアを測定します。仮想マシンは、信頼されるクラスターの信頼済みホスト間で移行できるので、セキュアな環境での高可用性を可能にします。

信頼できるクラスターの作成

1. **Compute** → **Clusters** をクリックします。
2. **New** をクリックします。
3. クラスターの **Name** を入力します。
4. **Enable Virt Service** チェックボックスを選択します。
5. **Scheduling Policy** タブをクリックして、**Enable Trusted Service** チェックボックスを選択します。
6. **OK** をクリックします。

12.9.3. 信頼できるホストの追加

Red Hat Enterprise Linux ホストは、信頼できるクラスターに追加し、OpenAttestation サーバーがホワイトリストデータベースに対して、このホストを測定できます。OpenAttestation サーバーで信頼されるようにするには、ホストが以下の要件を満たす必要があります。

- BIOS で Intel TXT が有効になっている。
- OpenAttestation エージェントがインストールされ、実行されている。
- ホスト上で実行されるソフトウェアは、OpenAttestation サーバーのホワイトリストデータベースと一致する。

信頼できるホストの追加

1. **Compute → Hosts** をクリックします。
2. **New** をクリックします。
3. **ホスト** クラスター のドロップダウンリストから信頼できるクラスターを選択します。
4. ホストの **Name** を入力します。
5. ホストの **Hostname** を入力します。
6. ホストの root **Password** を入力します。
7. **OK** をクリックします。

ホストが信頼できるクラスターに追加されると、OpenAttestation サーバーにより評価されます。ホストが OpenAttestation サーバーで信頼されていない場合には、**Non Operational** の状態に移行するため、信頼できるクラスターから削除する必要があります。

第13章 仮想ディスク

13.1. 仮想マシンストレージを理解する

Red Hat Virtualization は、NFS、iSCSI、FCP の 3 つのストレージタイプをサポートしています。

それぞれのタイプで、Storage Pool Manager (SPM) と呼ばれるホストが、ホストとストレージ間のアクセスを管理します。SPM ホストは、ストレージプール内でフルアクセスできる唯一のノードです。SPM は、ストレージドメインのメタデータとプールのメタデータを変更できます。他のすべてのホストは、仮想マシンのハードディスクイメージデータにのみアクセスできます。

デフォルトでは、NFS、ローカル、または POSIX 準拠のデータセンターでは、SPM は、ファイルシステム内のファイルとしてシンプロビジョニングされた形式を使用して仮想ディスクを作成します。

iSCSI およびその他のブロックベースのデータセンターでは、SPM は、提供された論理ユニット番号 (LUN) の上にボリュームグループを作成し、仮想ディスクとして使用する論理ボリュームを作成します。ブロックベースのストレージ上の仮想ディスクは、デフォルトで事前に割り当てられています。

仮想ディスクが事前に割り当てられている場合は、GB 単位で指定されたサイズの論理ボリュームが作成されます。仮想マシンは、**kpartx**、**vgscan**、**vgchange**、または **mount** を使用して Red Hat Enterprise Linux サーバーにマウントし、仮想マシンのプロセスまたは問題を調査できます。

仮想ディスクがシンプロビジョニングされる場合は、1 GB の論理ボリュームが作成されます。論理ボリュームは、仮想マシンが実行しているホストによって継続的に監視されます。使用量がしきい値に近づくとすぐに、ホストは SPM に通知し、SPM は論理ボリュームを 1GB 拡張します。ホストは、論理ボリュームが拡張された後、仮想マシンを再開する責任があります。仮想マシンが一時停止状態になる場合は、SPM が時間内にディスクを拡張できなかったことを意味します。これは、SPM がビジー状態であるか、十分なストレージスペースがない場合に発生します。

事前に割り当てられた (raw) 形式の仮想ディスクは、シンプロビジョニング (QCOW2) 形式の仮想ディスクよりも書き込み速度が圧倒的に高速に行えます。シンプロビジョニングは、仮想ディスクの作成にかかる時間が大幅に短縮されます。シンプロビジョニング形式は、I/O を多用しない仮想マシンに適しています。I/O 書き込みが多い仮想マシンには、事前に割り当てられた形式が推奨されます。仮想マシンが 4 秒ごとに 1GB を超える書き込みを実行できる場合は、可能な場合は事前に割り当てられたディスクを使用してください。

13.2. 仮想ディスクの概要

Red Hat Virtualization は、**事前割り当て** (シックプロビジョニング) および **スパース** (シンプロビジョニング) ストレージオプションを備えています。

- 事前割り当て
  事前に割り当てられた仮想ディスクは、仮想マシンに必要なすべてのストレージを事前に割り当てます。たとえば、仮想マシンのデータパーティション用に事前に割り当てられた 20 GB の論理ボリュームは、作成直後に 20GB のストレージスペースを占有します。

- スパース
  スパース割り当てを使用すると、管理者は仮想マシンに割り当てるストレージの合計を定義できますが、ストレージは必要な場合にのみ割り当てられます。

  たとえば、20 GB のシンプロビジョニングされた論理ボリュームは、最初に作成されたときに 0 GB のストレージスペースを占有します。オペレーティングシステムがインストールされると、インストールされたファイルのサイズを占める可能性があり、データが最大 20GB のサイズまで追加されるにつれて大きくなり続けます。

Storage → Disks で仮想ディスクの **ID** を表示できます。**ID** は、デバイス名 (たとえば、**/dev/vda0**) が変更されてディスクが破損する可能性があるため、仮想ディスクを識別するために使用されます。**/dev/disk/by-id** で仮想ディスクの ID を表示することもできます。

ディスクの **仮想サイズ** は、**Storage → Disks** と、ストレージドメイン、仮想マシン、およびテンプレートの詳細ビューの **Disks** タブで確認できます。**仮想サイズ** は、仮想マシンが使用できるディスク容量の合計量です。これは、仮想ディスクを作成または編集するときに **Size(GB)** フィールドに入力する数値です。

ディスクの **実際のサイズ** は、ストレージドメインとテンプレートの詳細ビューの **Disks** タブで確認できます。これは、これまでに仮想マシンに割り当てられたディスク容量です。事前に割り当てられたディスクは、**Virtual Size** および **Actual Size** に同じ値を示します。スパースディスクは、割り当てられているディスク容量に応じて、異なる値を表示する場合があります。

注記

Cinder 仮想ディスクを作成する場合には、ディスクの形式およびタイプは Cinder によって内部で処理され、Red Hat Virtualization では管理されません。

次の表に、ストレージの種類と形式の可能な組み合わせを示します。

表13.1 許可されたストレージの組み合わせ

ストレージ	形式	タイプ	注記
NFS	Raw	事前割り当て	仮想ディスクに定義されたストレージの量に等しい初期サイズのファイルで、フォーマットはありません。
NFS	Raw	スパース	初期サイズがゼロに近く、フォーマットされていないファイル。
NFS	QCOW2	スパース	初期サイズがゼロに近く、QCOW2 フォーマットのファイル。後続のレイヤーは QCOW2 形式になります。
SAN	Raw	事前割り当て	仮想ディスクに定義されたストレージの量に等しい初期サイズを持ち、フォーマットがないブロックデバイス。

ストレージ	形式	タイプ	注記
SAN	QCOW2	スパース	初期サイズが仮想ディスクに定義されたサイズ (現在は 1 GB) よりもはるかに小さく、必要に応じてスペースが割り当てられる QCOW2 フォーマット (現在は 1 GB 刻み) のブロックデバイス。

13.3. 削除後に仮想ディスクをワイプするための設定

管理ポータルで **Wipe After Delete** チェックボックスとして表示される **wipe_after_delete** フラグは、仮想ディスクが削除されたときに使用済みデータをゼロに置き換えます。デフォルトである false に設定されている場合、ディスクを削除すると、それらのブロックが再利用できるようになりますが、データは消去されません。したがって、ブロックがゼロに戻されていないため、このデータが回復される可能性があります。

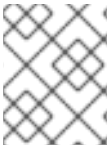
wipe_after_delete フラグは、ブロックストレージでのみ機能します。NFS などのファイルストレージでは、ファイルシステムがデータが存在しないことを確認するため、このオプションは何もしません。

仮想ディスクに対して **wipe_after_delete** を有効にすると、より安全になり、仮想ディスクに機密データが含まれている場合に推奨されます。これはより集中的な操作であり、ユーザーはパフォーマンスの低下と削除時間の延長を経験する可能性があります。

注記

削除後のワイプ機能は安全な削除と同じではなく、データがストレージから削除されることを保証することはできません。同じストレージ上に作成された新しいディスクが古いディスクからのデータを公開しないことだけです。

wipe_after_delete フラグのデフォルトは、設定プロセス時に **true** に変更できます (Configuring the Red Hat Virtualization Manager を参照)。または、Red Hat Virtualization Manager で **engine-config** ツールを使用して変更できます。設定の変更を有効にするには、**ovirt-engine** サービスを再起動します。



注記

wipe_after_delete フラグのデフォルト設定を変更しても、既存のディスクの **Wipe After Delete** プロパティーには影響しません。

エンジン設定ツールを使用して SANWipeAfterDelete をデフォルトの True に設定する

1. **--set** アクションを指定して **engine-config** ツールを実行します。

```
# engine-config --set SANWipeAfterDelete=true
```

2. 変更を反映するには、**ovirt-engine** サービスを再起動します。

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

ホストにある **/var/log/vdsm/vdsm.log** ファイルをチェックして、仮想ディスクが正常に消去および削除されたことを確認できます。

消去を成功させるために、ログファイルにはエントリーが含まれます。**storage_domain_id/volume_id was zeroed and will be deleted** はゼロにされ、削除されます。以下に例を示します。

```
a9cb0625-d5dc-49ab-8ad1-72722e82b0bf/a49351a7-15d8-4932-8d67-512a369f9d61 was zeroed and will be deleted
```

削除を成功させるために、ログファイルには **finished with VG:storage_domain_id LVs: list_of_volume_ids, img: image_id** エントリーが含まれます。以下に例を示します。

```
finished with VG:a9cb0625-d5dc-49ab-8ad1-72722e82b0bf LVs: {'a49351a7-15d8-4932-8d67-512a369f9d61': ImgsPar(imgs=['11f8b3be-fa96-4f6a-bb83-14c9b12b6e0d'], parent='00000000-0000-0000-0000-000000000000')}, img: 11f8b3be-fa96-4f6a-bb83-14c9b12b6e0d
```

消去が失敗すると、ログメッセージ **zeroing storage_domain_id/volume_id failedZero and remove this volume manually** が表示され、削除に失敗すると **Remove failed for some of VG: storage_domain_id zeroed volumes: list_of_volume_ids** が表示されます。

13.4. RED HAT VIRTUALIZATION の共有可能ディスク

一部のアプリケーションでは、サーバー間でストレージを共有する必要があります。Red Hat Virtualization を使用すると、仮想マシンのハードディスクを **Shareable** 可能としてマークし、それらのディスクを仮想マシンに接続できます。このようにして、単一の仮想ディスクを複数のクラスター対応ゲストが使用できます。

共有ディスクは、すべての状況で使用されるわけではありません。クラスター化されたデータベースサーバーやその他の高可用性サービスなどのアプリケーションには、共有ディスクが適しています。クラスターに対応していない複数のゲストに共有ディスクを接続すると、ディスクへの読み取りと書き込みが調整されていないため、データが破損する可能性があります。

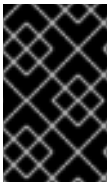
共有ディスクのスナップショットを撮ることはできません。スナップショットが取得された仮想ディスクは、後で共有可能としてマークすることはできません。

ディスクを作成するとき、または後でディスクを編集することによって、ディスクを共有可能としてマークできます。

13.5. RED HAT VIRTUALIZATION の読み取り専用ディスク

一部のアプリケーションでは、管理者が読み取り専用の権限でデータを共有する必要があります。これは、仮想マシンの詳細ビューの **Disks** タブを使用して仮想マシンに接続されたディスクを作成または編集し、**Read Only** チェックボックスをオンにするときに実行できます。これにより、管理者が書き込み権限を維持しながら、単一のディスクを複数のクラスター対応ゲストが読み取ることができます。

仮想マシンの実行中は、ディスクの読み取り専用ステータスを変更することはできません。



重要

ジャーナルファイルシステムをマウントするには、読み取り/書き込みアクセスが必要です。**Read Only** オプションの使用は、そのようなファイルシステム (**EXT3**、**EXT4**、**XFS** など) を含む仮想ディスクには適していません。

13.6. 仮想ディスクタスク

13.6.1. 仮想ディスクの作成

イメージディスクの作成は、Manager がすべて管理します。**ダイレクト LUN** ディスクには、外部で準備された、既存のターゲットが必要です。**Cinder** ディスクには、**外部プロバイダー** ウィンドウを使用して Red Hat Virtualization 環境に追加された OpenStack ボリュームのインスタンスへのアクセスが必要です。詳細は、「ストレージ管理用の OpenStack Block Storage (Cinder) インスタンスの追加」 を参照してください。

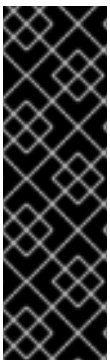
特定の仮想マシンに接続された仮想ディスクを作成できます。「新しい仮想ディスクウィンドウの設定の説明」 で記載されるているように、割り当てられた仮想ディスクを作成する際に追加のオプションが利用できます。

仮想マシンに接続された仮想ディスクの作成

1. **Compute → Virtual Machines**をクリックします。
2. 仮想マシンの名前をクリックして、詳細ビューに表示します。
3. **Disks** タブをクリックします。
4. **New** をクリックします。
5. 適切なボタンをクリックして、仮想ディスクを **イメージ** ディスクにするか、**ダイレクト LUN** ディスクにするか、**Cinder** ディスクにするかを指定します。
6. 仮想ディスクに必要なオプションを選択します。オプションは、選択したディスクタイプに基づいて変更します。各ディスクタイプの各オプションに関する詳細は、「新しい仮想ディスクウィンドウの設定の説明」 を参照してください。
7. **OK** をクリックします。

また、どの仮想マシンにも属さないフローティング仮想ディスクを作成することもできます。このディスクは、単一の仮想マシンに接続することも、ディスクが共有可能な場合は複数の仮想マシンに接続することもできます。一部のオプションは、「新しい仮想ディスクウィンドウの設定の説明」 で記載されているように、仮想ディスクの作成時には利用できません。

フローティング仮想ディスクの作成



重要

Floating 仮想ディスクの作成はテクノロジープレビュー機能です。テクノロジープレビュー機能は、Red Hat の実稼働環境でのサービスレベルアグリーメント (SLA) ではサポートされていないため、Red Hat では実稼働環境での使用を推奨していません。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。

Red Hat のテクノロジープレビュー機能のサポートについての詳細は、テクノロジープレビュー機能のサポート範囲 を参照してください。

1. **Storage** → **Disks** をクリックします。
2. **New** をクリックします。
3. 適切なボタンをクリックして、仮想ディスクを **イメージ** ディスクにするか、**ダイレクト LUN** ディスクにするか、**Cinder** ディスクにするかを指定します。
4. 仮想ディスクに必要なオプションを選択します。オプションは、選択したディスクタイプに基づいて変更します。各ディスクタイプの各オプションに関する詳細は、「新しい仮想ディスクウィンドウの設定の説明」 を参照してください。
5. **OK** をクリックします。

13.6.2. 新しい仮想ディスクウィンドウの設定の説明

フローティング仮想ディスクと接続仮想ディスクを作成するための新しい仮想ディスクウィンドウは非常に似ているため、それらの設定は 1 つのセクションで説明されています。

表13.2 New Virtual Disk および Edit Virtual Disk の設定image

フィールド名	説明
Size(GB)	新しい仮想ディスクのサイズ (GB 単位)。
エイリアス	仮想ディスクの名前。最大で 40 文字に制限されます。
Description	仮想ディスクの説明。このフィールドは推奨されますが、必須ではありません。
Interface	このフィールドは、接続されたディスクを作成する際に限り表示されます。 ディスクが仮想マシンに提示する仮想インターフェイス。**VirtIO** はより高速ですが、ドライバーが必要になります。Red Hat Enterprise Linux 5 以降にはこれらのドライバーが含まれています。Windows にはこれらのドライバーは含まれていませんが、ゲストツール ISO または仮想フロッピーディスクからインストールできます。**IDE** デバイスは特別なドライバーを必要としません。 インターフェイスタイプは、ディスクが接続されているすべての仮想マシンを停止した後に更新できます。
Data Center	このフィールドは、フローティングディスクを作成するときにのみ表示されます。 仮想ディスクが利用できるデータセンター。

フィールド名	説明
Storage Domain	仮想ディスクが保存されるストレージドメイン。ドロップダウンリストには、特定のデータセンターで使用可能なすべてのストレージドメインが表示され、ストレージドメインで使用可能な合計容量と現在使用可能な容量も表示されます。
Allocation Policy	新しい仮想ディスクのプロビジョニングポリシー。 ● **Preallocated** は、仮想ディスクの作成時に、ディスクのサイズ全体をストレージドメインに割り当てます。事前に割り当てられたディスクの仮想サイズと実際のサイズは同じです。事前に割り当てられた仮想ディスクは、シンプロビジョニングされた仮想ディスクよりも作成に時間がかかりますが、読み取りと書き込みのパフォーマンスは向上します。サーバーやその他の I/O を多用する仮想マシンには、事前に割り当てられた仮想ディスクをお勧めします。仮想マシンが 4 秒ごとに 1GB を超える書き込みを実行できる場合は、可能な場合は事前に割り当てられたディスクを使用してください。 ● **Thin Provision** は、仮想ディスクの作成時に 1GB を割り当て、ディスクを拡張できるサイズの最大制限を設定します。ディスクの仮想サイズは最大制限です。ディスクの実際のサイズは、これまでに割り当てられたスペースです。シンプロビジョニングされたディスクは、事前に割り当てられたディスクよりも作成が速く、ストレージのオーバーコミットが可能です。デスクトップには、シンプロビジョニングされた仮想ディスクが推奨されます。
ディスクプロファイル	仮想ディスクに割り当てられたディスクプロファイル。ディスクプロファイルは、ストレージドメイン内の仮想ディスクのスループットの最大量と入出力操作の最大レベルを定義します。ディスクプロファイルは、データセンター用に作成されたストレージのサービス品質エントリーに基づいて、ストレージドメインレベルで定義されます。
ディスクをアクティブ化する	このフィールドは、接続されたディスクを作成する際に限り表示されます。 作成後すぐに仮想ディスクをアクティブ化します。
Wipe After Delete	仮想ディスクが削除されたときに機密資料を削除するための強化されたセキュリティーを有効にすることができます。

フィールド名	説明
起動可能	このフィールドは、接続されたディスクを作成する際に限り表示されます。 仮想ディスクで起動可能フラグを有効にすることができます。
Shareable	一度に複数の仮想マシンに仮想ディスクを接続できます。
Read-Only	このフィールドは、接続されたディスクを作成する際に限り表示されます。 ディスクを読み取り専用として設定できます。同じディスクを読み取り専用として1つの仮想マシンに接続したり、別の仮想マシンに再書き込み可能として接続したりできます。
破棄を有効にする	このフィールドは、接続されたディスクを作成する際に限り表示されます。 仮想マシンが稼働しているときに、シンプロビジョニングされたディスクを縮小できます。ブロックストレージの場合、基盤となるストレージデバイスは破棄呼び出しをサポートする必要があり、基盤となるストレージが discard_zeroes_data プロパティーをサポートしない限り、このオプションを **Wipe After Delete** で使用することはできません。ファイルストレージの場合、基盤となるファイルシステムおよびブロックデバイスは破棄呼び出しをサポートする必要があります。すべての要件が満たされている場合、ゲスト仮想マシンから発行された SCSI UNMAP コマンドは、QEMU によって基盤となるストレージに渡され、未使用のスペースが解放されます。

Direct LUN 設定は、**Targets > LUNs** または **LUNs > Targets** のいずれかに表示できます。**Targets > LUNs** は、検出されたホストに従って使用可能な LUN をソートしますが、**LUNs > Targets** は LUN の単一のリストを表示します。

Discover Targets セクションのフィールドに入力し、**Discover** をクリックしてターゲットサーバーを検出します。次に、**Login All** ボタンをクリックして、ターゲットサーバーで使用可能な LUN を一覧表示し、各 LUN の横にあるラジオボタンを使用して、追加する LUN を選択します。

LUN を仮想マシンのハードディスクイメージとして直接使用すると、仮想マシンとそのデータの間の抽象化レイヤーが削除されます。

ダイレクト LUN を仮想マシンのハードディスクイメージとして使用する場合は、次の考慮事項を考慮する必要があります。

- ダイレクト LUN ハードディスクイメージのライブストレージ移行はサポートされていません。
- ダイレクト LUN ディスクは、仮想マシンのエクスポートには含まれません。

- ダイレクト LUN ディスクは、仮想マシンのスナップショットには含まれていません。

表13.3 New Virtual Disk および Edit Virtual Disk の設定Direct LUN

フィールド名	説明
エイリアス	仮想ディスクの名前。最大で 40 文字に制限されます。
Description	仮想ディスクの説明。このフィールドは推奨されますが、必須ではありません。デフォルトでは、LUN ID の最後の 4 文字がフィールドに挿入されます。 デフォルトの動作は、**engine-config** コマンドを使用して **PopulateDirectLUNDiskDescriptionWithLUNId** 設定キーを適切な値に設定することで設定できます。設定キーは、完全な LUN ID を使用する場合は **-1** に設定でき、この機能を無視する場合は **0** に設定できます。正の整数は、説明に LUN ID の対応する文字数を入力します。
Interface	このフィールドは、接続されたディスクを作成する際に限り表示されます。 ディスクが仮想マシンに提示する仮想インターフェイス。**VirtIO** はより高速ですが、ドライバーが必要になります。Red Hat Enterprise Linux 5 以降にはこれらのドライバーが含まれています。Windows にはこれらのドライバーは含まれていませんが、ゲストツール ISO または仮想フロッピーディスクからインストールできます。**IDE** デバイスは特別なドライバーを必要としません。 インターフェイスタイプは、ディスクが接続されているすべての仮想マシンを停止した後に更新できます。
Data Center	このフィールドは、フローティングディスクを作成するときにのみ表示されます。 仮想ディスクが利用できるデータセンター。
Host	LUN がマウントされるホスト。データセンター内の任意のホストを選択できます。
Storage Type	追加する外部 LUN のタイプ。**iSCSI** または **Fibre Channel** から選択できます。

フィールド名	説明
Discover Targets	このセクションは、iSCSI 外部 LUN を使用していて、**Targets > LUNs** が選択されている場合に展開できます。 **Address** - ターゲットサーバーのホスト名または IP アドレス。 **Port** - ターゲットサーバーへの接続を試みるためのポート。デフォルトのポートは 3260 です。 **User Authentication** - iSCSI サーバーにはユーザー認証が必要です。iSCSI 外部 LUN を使用している場合は、**User Authentication** フィールドが表示されます。 **CHAP user name** - LUN にログインする権限を持つユーザーのユーザー名。このフィールドには、**User Authentication** チェックボックスがオンになっている場合にアクセスできます。 **CHAP password** - LUN にログインする権限を持つユーザーのパスワード。このフィールドには、**User Authentication** チェックボックスがオンになっている場合にアクセスできます。
ディスクをアクティブ化する	このフィールドは、接続されたディスクを作成する際に限り表示されます。 作成後すぐに仮想ディスクをアクティブ化します。
起動可能	このフィールドは、接続されたディスクを作成する際に限り表示されます。 仮想ディスクで起動可能フラグを有効にすることができます。
Shareable	一度に複数の仮想マシンに仮想ディスクを接続できます。
Read-Only	このフィールドは、接続されたディスクを作成する際に限り表示されます。 ディスクを読み取り専用として設定できます。同じディスクを読み取り専用として 1 つの仮想マシンに接続したり、別の仮想マシンに再書き込み可能として接続したりできます。

フィールド名	説明
破棄を有効にする	このフィールドは、接続されたディスクを作成する際に限り表示されます。 仮想マシンが稼働しているときに、シンプロビジョニングされたディスクを縮小できます。このオプションを有効にすると、ゲスト仮想マシンから発行された SCSI UNMAP コマンドは、QEMU によって基盤となるストレージに渡され、未使用のスペースが解放されます。
Enable SCSI Pass-Through	このフィールドは、接続されたディスクを作成する際に限り表示されます。 **Interface** が **VirtIO-SCSI** に設定されている場合に使用できます。このチェックボックスをオンにすると、物理 SCSI デバイスを仮想ディスクにパススルーできます。SCSI パススルーが有効になっている VirtIO-SCSI インターフェイスには、SCSI 廃棄のサポートが自動的に含まれています。このチェックボックスが選択されている場合、**読み取り専用** はサポートされません。 このチェックボックスが選択されていない場合、仮想ディスクはエミュレートされた SCSI デバイスを使用します。**Read-Only** は、エミュレートされた VirtIO-SCSI ディスクでサポートされています。
Allow Privileged SCSI I/O	このフィールドは、接続されたディスクを作成する際に限り表示されます。 **Enable SCSI Pass-Through** チェックボックスがオンになっている場合に使用できます。このチェックボックスをオンにすると、フィルターリングされていない SCSI Generic I/O (SG_IO) アクセスが有効になり、ディスク上で特権 SG_IO コマンドが許可されます。これは永続的な予約に必要です。
Using SCSI Reservation	このフィールドは、接続されたディスクを作成する際に限り表示されます。 **Enable SCSI Pass-Through** および **Allow Privileged SCSI I/O** チェックボックスがオンになっている場合に使用できます。このチェックボックスをオンにすると、このディスクを使用する仮想マシンの移行が無効になり、SCSI 予約を使用する仮想マシンがディスクにアクセスできなくなるのを防ぐことができます。

該当するデータセンターにディスクを作成する権限がある OpenStack Volume のストレージドメインが利用できない場合には、**Cinder** の設定フォームが無効になります。**Cinder** ディスクには、**外部プロバ**

イダー ウィンドウを使用して Red Hat Virtualization 環境に追加された OpenStack ボリュームのインスタンスへのアクセスが必要です。詳細は、「ストレージ管理用の OpenStack Block Storage (Cinder) インスタンスの追加」 を参照してください。

表13.4 New Virtual Disk および Edit Virtual Disk の設定Cinder

フィールド名	説明
Size(GB)	新しい仮想ディスクのサイズ (GB 単位)。
エイリアス	仮想ディスクの名前。最大で 40 文字に制限されます。
Description	仮想ディスクの説明。このフィールドは推奨されますが、必須ではありません。
Interface	このフィールドは、接続されたディスクを作成する際に限り表示されます。 ディスクが仮想マシンに提示する仮想インターフェイス。**VirtIO** はより高速ですが、ドライバーが必要になります。Red Hat Enterprise Linux 5 以降にはこれらのドライバーが含まれています。Windows にはこれらのドライバーは含まれていませんが、ゲストツール ISO または仮想フロッピーディスクからインストールできます。**IDE** デバイスは特別なドライバーを必要としません。 インターフェイスタイプは、ディスクが接続されているすべての仮想マシンを停止した後に更新できます。
Data Center	このフィールドは、フローティングディスクを作成するときにのみ表示されます。 仮想ディスクが利用できるデータセンター。
Storage Domain	仮想ディスクが保存されるストレージドメイン。ドロップダウンリストには、特定のデータセンターで使用可能なすべてのストレージドメインが表示され、ストレージドメインで使用可能な合計容量と現在使用可能な容量も表示されます。
Volume Type	仮想ディスクのボリュームタイプ。ドロップダウンリストには、利用可能なすべてのボリュームタイプが表示されます。このボリュームタイプは、OpenStack Cinder で管理および設定されます。
Activate Disk(s)	このフィールドは、接続されたディスクを作成する際に限り表示されます。 作成後すぐに仮想ディスクをアクティブ化します。

フィールド名	説明
起動可能	このフィールドは、接続されたディスクを作成する際に限り表示されます。 仮想ディスクで起動可能フラグを有効にすることができます。
Shareable	一度に複数の仮想マシンに仮想ディスクを接続できます。
Read-Only	このフィールドは、接続されたディスクを作成する際に限り表示されます。 ディスクを読み取り専用として設定できます。同じディスクを読み取り専用として1つの仮想マシンに接続したり、別の仮想マシンに再書き込み可能として接続したりできます。

重要

ジャーナルファイルシステムをマウントするには、読み取り/書き込みアクセスが必要です。**Read Only** オプションの使用は、そのようなファイルシステム (**EXT3**、**EXT4**、**XFS** など) を含む仮想ディスクには適していません。

13.6.3. ライブストレージ移行の概要

仮想ディスクは、それらが接続されている仮想マシンの実行中に、あるストレージドメインから別のストレージドメインに移行できます。これは、ライブストレージ移行と呼ばれます。実行中の仮想マシンに接続されているディスクが移行されると、そのディスクのイメージチェーンのスナップショットがソースストレージドメインに作成され、イメージチェーン全体が宛先ストレージドメインに複製されます。そのため、ソースストレージドメインと宛先ストレージドメインの両方に、ディスクイメージチェーンとスナップショットの両方をホストするのに十分なストレージスペースがあることを確認してください。移行が失敗した場合でも、ライブストレージの移行が試行されるたびに新しいスナップショットが作成されます。

ライブストレージ移行を使用する場合は、次の点を考慮してください。

- 一度に複数のディスクをライブマイグレーションできます。
- 同じ仮想マシンの複数のディスクは複数のストレージドメインにまたがって存在できますが、各ディスクのイメージチェーンは単一のストレージドメインに存在する必要があります。
- 同じデータセンター内の任意の 2 つのストレージドメイン間でディスクをライブマイグレーションできます。
- ダイレクト LUN ハードディスクイメージまたは共有可能としてマークされたディスクをライブマイグレーションすることはできません。

13.6.4. 仮想ディスクの移動

仮想マシンに接続されている、またはフローティング仮想ディスクとして機能する仮想ディスクを、あるストレージドメインから別のストレージドメインに移動します。実行中の仮想マシンに接続されてい

る仮想ディスクを移動できます。これは、ライブストレージ移行と呼ばれます。または、続行する前に仮想マシンをシャットダウンします。

ディスクを移動するときは、次の点を考慮してください。

- 複数のディスクを同時に移動できます。
- 同じデータセンター内の任意の 2 つのストレージドメイン間でディスクを移動できます。
- テンプレートに基づいて作成され、シンプロビジョニングストレージ割り当てオプションを使用した仮想マシンに仮想ディスクが接続されている場合は、仮想マシンが仮想ディスクと同じストレージドメインに基づいていたテンプレートのディスクをコピーする必要があります。

仮想ディスクの移動

1. **Storage → Disks** をクリックして、移動する 1 つ以上の仮想ディスクを選択します。
2. **Move** をクリックします。
3. **Target** リストから、仮想ディスクの移動先のストレージドメインを選択します。
4. 必要に応じて、**Disk Profile** リストからディスクのプロファイルを選択します。
5. **OK** をクリックします。

仮想ディスクは、対象のストレージドメインに移動します。移動手順時に、**Status** 列には、**Locked** と表示され、また移動操作の進捗を示す進捗バーが表示されます。

13.6.5. ディスクインターフェイスタイプの変更

ユーザーは、ディスクの作成後にディスクのインターフェイスタイプを変更できます。これにより、既存のディスクを、異なるインターフェイスタイプを必要とする仮想マシンに接続できます。たとえば、**VirtIO** インターフェイスを使用するディスクは、**VirtIO-SCSI** または **IDE** インターフェイスを必要とする仮想マシンに接続できます。これにより、バックアップと復元、または障害復旧の目的でディスクを移行する柔軟性が提供されます。共有可能ディスクのディスクインターフェイスは、仮想マシンごとに更新することもできます。これは、共有ディスクを使用する各仮想マシンが異なるインターフェイスタイプを使用できることを意味します。

ディスクインターフェイスタイプを更新するには、最初にディスクを使用するすべての仮想マシンを停止する必要があります。

ディスクインターフェイスタイプの変更

1. **Compute → Virtual Machines** をクリックして、適切な仮想マシンを停止します。
2. 仮想マシンの名前をクリックして、詳細ビューに表示します。
3. **Disks** タブをクリックして、ディスクを選択します。
4. **Edit** をクリックします。
5. **Interface** リストから、新しいインターフェイスタイプを選択し、**OK** をクリックします。

別のインターフェイスタイプを必要とする別の仮想マシンにディスクを接続できます。

別のインターフェイスタイプを使用して別の仮想マシンにディスクを接続

1. **Compute** → **Virtual Machines** をクリックして、適切な仮想マシンを停止します。
2. 仮想マシンの名前をクリックして、詳細ビューに表示します。
3. **Disks** タブをクリックして、ディスクを選択します。
4. **Remove** をクリックしてから **OK** をクリックします。
5. **仮想マシン** に戻り、ディスクが割り当てられる新しい仮想マシンの名前をクリックします。
6. **Disks** タブをクリックしてから **Attach** をクリックします。
7. **Attach Virtual Disks** ウィンドウでディスクを選択し、**Interface** ドロップダウンから適切なインターフェイスを選択します。
8. **OK** をクリックします。

13.6.6. 仮想ディスクのコピー

あるストレージドメインから別のストレージドメインに仮想ディスクをコピーできます。コピーしたディスクは仮想マシンに接続できます。

仮想ディスクのコピー

1. **Storage** → **Disks** をクリックして、仮想ディスクを選択します。
2. **コピー** をクリックします。
3. 必要に応じて、**Alias** フィールドに新しい名前を入力します。
4. **Target** リストから、仮想ディスクのコピー先のストレージドメインを選択します。
5. 必要に応じて、**Disk Profile** リストからディスクのプロファイルを選択します。
6. **OK** をクリックします。

コピー中の仮想ディスクのステータスは **Locked** です。

13.6.7. データストレージドメインへのイメージのアップロード

管理ポータルまたは REST API を使用して、仮想ディスクイメージと ISO イメージをデータストレージドメインにアップロードできます。「データストレージドメインへのイメージのアップロード」 を参照してください。

13.6.8. インポートされたストレージドメインからのディスクイメージのインポート

インポートされたストレージドメインからフローティング仮想ディスクをインポートできます。

注記

Manager にインポートできるのは QEMU 互換ディスクのみです。

ディスクイメージのインポート

1. **Storage** → **Domains** をクリックします。

2. インポートしたストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **Disk Import** タブをクリックします。
4. 1つ以上のディスクを選択し、**Import** をクリックします。
5. 各ディスクに適切な **ディスクプロファイル** を選択します。
6. **OK** をクリックします。

13.6.9. インポートされたストレージドメインからの未登録のディスクイメージのインポート

ストレージドメインからフローティング仮想ディスクをインポートできます。Red Hat Virtualization 環境の外部で作成されたフローティングディスクは、Manager には登録されません。ストレージドメインをスキャンして、インポートする未登録のフローティングディスクを特定します。

注記

Manager にインポートできるのは QEMU 互換ディスクのみです。

ディスクイメージのインポート

1. **Storage → Domains** をクリックします。
2. ストレージドメインの名前をクリックし、詳細ビューを開きます。
3. **More Actions** (⋮) をクリックしてから、Manager が未登録のディスクを特定できるように、**Scan Disks** ディスクをクリックします。
4. **Disk Import** タブをクリックします。
5. 1つ以上のディスクイメージを選択し、**インポート** をクリックします。
6. 各ディスクに適切な **ディスクプロファイル** を選択します。
7. **OK** をクリックします。

13.6.10. OpenStack Image Service からの仮想ディスクのインポート

OpenStack Image サービスが外部プロバイダーとして Manager に追加されている場合には、その OpenStack Image サービスが管理する仮想ディスクを Red Hat Virtualization Manager にインポートすることができます。

1. **Storage → Domains** をクリックします。
2. OpenStack Image サービスドメイン名をクリックして、詳細ビューを開きます。
3. **Images** タブをクリックして、イメージを選択します。
4. **Import** をクリックします。
5. イメージをインポートする **データセンター** を選択します。

6. **Domain Name** ドロップダウンリストから、イメージが保存されるストレージドメインを選択します。
7. 必要に応じて、**Quota** ドロップダウンリストからイメージに適用するクォータを選択します。
8. **OK** をクリックします。

これで、ディスクを仮想マシンに接続できます。

13.6.11. OpenStack Image Service への仮想ディスクのエクスポート

仮想ディスクは、外部プロバイダーとして Manager に追加された OpenStack Image Service にエクスポートできます。

重要

仮想ディスクは、複数のボリュームがなく、シンプロビジョニングされておらず、スナップショットがない場合にのみエクスポートできます。

1. **Storage → Disks** をクリックして、エクスポートするディスクを選択します。
2. **More Actions** (⋮) をクリックしてから、**Export** をクリックします。
3. **Domain Name** ドロップダウンリストから、ディスクのエクスポート先となる OpenStack Image Service を選択します。
4. クォータを適用する場合は、**Quota** ドロップダウンリストからディスクのクォータを選択します。
5. **OK** をクリックします。

13.6.12. 仮想ディスクスペースの回収

シンプロビジョニングを使用する仮想ディスクは、ファイルを削除した後、自動的に縮小しません。たとえば、実際のディスクサイズが 100GB で、50GB のファイルを削除した場合、割り当てられたディスクサイズは 100GB のままであり、残りの 50GB はホストに返されないため、他の仮想マシンで使用できません。この未使用のディスク領域は、仮想マシンのディスクでスパース操作を実行することにより、ホストによって再利用できます。これにより、空き領域がディスクイメージからホストに転送されます。複数の仮想ディスクを並行してスパース化できます。

Red Hat は、この操作は、仮想マシンのクローンを作成する前、仮想マシンに基づいてテンプレートを作成する前、またはストレージドメインのディスク領域をクリーンアップする前に実行することを推奨します。

制限事項

- NFS ストレージドメインは、NFS バージョン 4.2 以降を使用する必要があります。
- ダイレクト LUN または Cinder を使用するディスクをスパース化することはできません。
- 事前に割り当てられた割り当てポリシーを使用するディスクをスパース化することはできません。テンプレートから仮想マシンを作成する場合は、**Storage Allocation** フィールドから **Thin** を選択する必要があります。**クローン** を選択する場合は、テンプレートがシンプロビジョニングのある仮想マシンに基づいていることを確認してください。

- アクティブなスナップショットのみをスパースできます。

Sparsifying a Disk

1. **Compute → Virtual Machines** をクリックして、必要な仮想マシンをシャットダウンします。
2. 仮想マシンの名前をクリックして、詳細ビューに表示します。
3. **Disks** タブをクリックします。ディスクのステータスが **OK** であることを確認します。
4. **More Actions** ( ) をクリックしてから、**Sparsify** をクリックします。
5. **OK** をクリックします。

Started to sparsify イベントは、スパース化操作中に **Events** タブに表示され、ディスクのステータスは **Locked** と表示されます。操作が完了すると、**Sparsified successfully** イベントが **Events** タブに表示され、ディスクのステータスが **OK** と表示されます。未使用のディスク領域はホストに戻され、他の仮想マシンで使用できるようになりました。

第14章 外部プロバイダー

14.1. RED HAT VIRTUALIZATION における外部プロバイダーの紹介

Red Hat Virtualization Manager 自体によって管理されるリソースに加えて、Red Hat Virtualization は外部ソースによって管理されるリソースを利用することもできます。外部プロバイダーと呼ばれるこれらのリソースのプロバイダーは、仮想化ホスト、仮想マシンイメージ、ネットワークなどのリソースを提供できます。

Red Hat Virtualization は現在、以下の外部プロバイダーをサポートしています。

ホストプロビジョニング用の Red Hat Satellite

Satellite は、物理ホストと仮想ホストの両方のライフサイクルのすべての側面を管理するためのツールです。Red Hat Virtualization では、Satellite によって管理されるホストを、Red Hat Virtualization Manager に仮想化ホストとして追加して使用できます。Manager に Satellite インスタンスを追加した後、新しいホストを追加するときにその Satellite インスタンスで使用可能なホストを検索することにより、Satellite インスタンスによって管理されるホストを追加できます。Red Hat Satellite のインストール、および Red Hat Satellite を使用したホストの管理に関する詳細は、Red Hat Satellite Installation Guide および Red Hat Satellite Host Configuration Guide を参照してください。

イメージ管理用の OpenStack Image Service (Glance)

OpenStack Image Service は、仮想マシンイメージのカタログを提供します。Red Hat Virtualization では、これらのイメージを Red Hat Virtualization Manager にインポートして、フローティングディスクとして使用したり、仮想マシンに接続してテンプレートに変換したりできます。OpenStack Image Service を Manager に追加すると、どのデータセンターにも接続されていないストレージドメインとして表示されます。Red Hat Virtualization 環境の仮想ディスクは、仮想ディスクとして OpenStack Image Service にエクスポートすることもできます。

ネットワークプロビジョニング用の OpenStack Networking (Neutron)

OpenStack Networking は、ソフトウェア定義ネットワークを提供します。Red Hat Virtualization では、OpenStack Networking が提供するネットワークを Red Hat Virtualization Manager にインポートして、あらゆる種類のトラフィックを伝送し、複雑なネットワークトポロジーを作成するのに使用します。OpenStack Networking を Manager に追加した後に、手動でインポートして OpenStack Networking が提供するネットワークにアクセスすることができます。

ストレージ管理用の OpenStack ボリューム (Cinder)

OpenStack Volume は、仮想ハードドライブの永続的なブロックストレージ管理機能を提供します。OpenStack Cinder ボリュームは Ceph Storage によりプロビジョニングされます。Red Hat Virtualization では、OpenStack ボリュームストレージ上にディスクを作成することができます。このストレージは、フローティングディスクとして使用したり、仮想マシンにアタッチしたりできます。OpenStack ボリュームを Manager に追加した後に、OpenStack ボリュームが提供するストレージにディスクを作成することができます。

仮想マシンプロビジョニング用の VMware

VMware で作成された仮想マシンは、V2V (**virt-v2v**) を使用して変換し、Red Hat Virtualization 環境にインポートできます。VMware プロバイダーを Manager に追加した後、それが提供する仮想マシンをインポートできます。V2V 変換は、インポート操作の一部として、指定されたプロキシーホストで実行されます。

仮想マシンプロビジョニング用の RHEL 5 Xen

RHEL 5 Xen で作成された仮想マシンは、V2V (**virt-v2v**) を使用して変換し、Red Hat Virtualization 環境にインポートできます。RHEL 5 Xen ホストを Manager に追加した後、それが提供する仮想マシンをインポートできます。V2V 変換は、インポート操作の一部として、指定されたプロキシーホストで実行されます。

仮想マシンプロビジョニング用の KVM

KVM で作成された仮想マシンは、Red Hat Virtualization 環境にインポートできます。KVM ホストを Manager に追加した後、KVM ホストが提供する仮想マシンをインポートできます。

ネットワークプロビジョニング用の Open Virtual Network (OVN)

Open Virtual Network (OVN) は、ソフトウェア定義のネットワークを提供する Open vSwitch (OVS) 拡張機能です。Manager に OVN を追加した後、既存の OVN ネットワークをインポートし、Manager から新しい OVN ネットワークを作成できます。**engine-setup** を使用して、Manager に OVN を自動的にインストールすることもできます。

ネットワークプロビジョニング用の外部ネットワークプロバイダー

サポート対象の外部ソフトウェア定義のネットワークプロバイダーには、OpenStack REST API を実装するプロバイダーが含まれます。OpenStack Networking (Neutron) とは異なり、Neutron エージェントはホスト上の仮想インターフェイスドライバーの実装としては使用されません。代わりに、仮想インターフェイスドライバーは、外部ネットワークプロバイダーの実装者が提供する必要があります。

すべての外部リソースプロバイダーは、入力にあった単一のウィンドウを使用して追加されます。Red Hat Virtualization 環境で提供されるリソースを使用するには、リソースプロバイダーを追加する必要があります。

14.2. 外部プロバイダーの追加

14.2.1. ホストのプロビジョニング用の Red Hat Satellite インスタンスの追加

ホストプロビジョニング用の Satellite インスタンスを Red Hat Virtualization Manager に追加します。Red Hat Virtualization 4.2 は、Red Hat Satellite 6.1 でサポートされています。

ホストプロビジョニング用の Satellite インスタンスの追加

1. **Administration → Providers** をクリックします。
2. **Add** をクリックします。
3. **Name** および **Description** を入力します。
4. **Type** ドロップダウンリストから **Foreman/Satellite** を選択します。
5. Satellite インスタンスがインストールされているマシンの URL または完全修飾ドメイン名を **Provider URL** テキストフィールドに入力します。ポート番号を指定する必要はありません。

重要

IP アドレスを使用して Satellite インスタンスを追加することはできません。

6. **Requires Authentication** チェックボックスをオンにします。
7. Satellite インスタンスの **ユーザー名** と **パスワード** を入力します。Satellite プロビジョニングポータルへのログインに使用するのと同じユーザー名とパスワードを使用する必要があります。
8. 認証情報をテストします。
   a. **Test** をクリックして、提供された認証情報を使用して Satellite インスタンスで正常に認証できるかどうかをテストします。

b. Satellite インスタンスが SSL を使用している場合は、**Import provider certificates** ウィンドウが開きます。**OK** をクリックして、Satellite インスタンスが提供する証明書をインポートし、Manager がインスタンスと通信できるようにします。

9. **OK** をクリックします。

14.2.2. イメージ管理用の OpenStack Image (Glance) インスタンスの追加

Red Hat Virtualization Manager にイメージ管理用の OpenStack Image (Glance) インスタンスを追加します。

イメージ管理用の OpenStack Image (Glance) インスタンスの追加

1. **Administration → Providers** をクリックします。
2. **Add** をクリックし、**General Settings** タブに詳細を入力します。これらのフィールドの詳細は、「プロバイダーの一般設定の説明を追加」 を参照してください。
3. **Name** および **Description** を入力します。
4. **Type** ドロップダウンリストから **OpenStack Image** を選択します。
5. OpenStack Image インスタンスがインストールされているマシンの URL または完全修飾ドメイン名を **Provider URL** テキストフィールドに入力します。
6. 必要に応じて、**Requires Authentication** チェックボックスを選択し、Keystone に登録されている OpenStack Image インスタンスユーザーの **Username** 名と **Password** を入力します。**Protocol** (**HTTP** である必要があります)、**Hostname**、および **API Port.** を定義して Keystone サーバーの認証 URL を定義する必要もあります。
   OpenStack Image インスタンスの **Tenant** を入力します。
7. 認証情報をテストします。
   a. **Test** をクリックして、提供された認証情報を使用して OpenStack Image インスタンスで正常に認証できるかどうかをテストします。
   b. OpenStack Image インスタンスが SSL を使用している場合、**Import provider certificates** ウィンドウが開きます。**OK** をクリックして、OpenStack Image インスタンスが提供する証明書をインポートし、Manager がインスタンスと通信できるようにします。
8. **OK** をクリックします。

14.2.3. ネットワークプロビジョニング用の OpenStack Networking (Neutron) インスタンスの追加

ネットワークプロビジョニング用の OpenStack Networking (neutron) インスタンスを Red Hat Virtualization Manager に追加します。OpenStack Neutron REST API を実装する他のサードパーティーネットワークプロバイダーを追加するには、「外部ネットワークプロバイダーの追加」 を参照してください。

重要

Red Hat Virtualization は、外部ネットワークプロバイダーとして Red Hat OpenStack Platform バージョン 10、13、14 をサポートします。

- OpenStack 10 は OVS ドライバーと共にデプロイする必要があります。
- OpenStack 13 は、OVS、OVN、または ODL ドライバーと共にデプロイする必要があります。
- OpenStack 14 は、OVN または ODL ドライバーと共にデプロイする必要があります。

neutron ネットワークを使用するには、ホストに neutron エージェントが設定されている必要があります。ネットワークノードをホストとして Manager に追加する前に、エージェントを手動で設定するか、Red Hat OpenStack Platform director を使用して Networker ロールをデプロイしてください。director の使用が推奨されます。**New Host** ウィンドウの **Network Provider** タブを使用した neutron エージェントの自動デプロイメントには対応していません。

ネットワークノードと通常のホストを同じクラスターで使用できますが、neutron ネットワークを使用する仮想マシンはネットワークノードでのみ実行できます。

ネットワークノードのホストとしての追加

1. Red Hat OpenStack Platform director を使用して、ネットワークノードに Networker ロールをデプロイします。**Red Hat OpenStack Platform Advanced Overcloud Customization**Creating a New Role と Networker を参照してください。
2. 必要なリポジトリーを有効にします。
   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のユーザー名とパスワードを入力します。

   ```
   # subscription-manager register
   ```

   b. **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
   ```

   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=poolid
   ```

   d. リポジトリーを設定します。

   ```
   # subscription-manager repos \
       --disable='*' \
       --enable=rhel-7-server-rpms \
       --enable=rhel-7-server-rhv-4-mgmt-agent-rpms \
       --enable=rhel-7-server-ansible-2.9-rpms
   ```

   e. 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。

   ```
   # yum update
   ```

f. いずれかのカーネルのパッケージを更新した場合には、マシンを再起動してください。

3. Openstack Networking フックをインストールします。

```
# yum install vdsm-hook-openstacknet
```

4. ネットワークノードを Manager にホストとして追加します。「Red Hat Virtualization Manager への通常のホストの追加」を参照してください。

重要

Network Provider タブから OpenStack Networking プロバイダーを選択しないでください。これは現在サポートされていません。

ネットワークプロビジョニング用の OpenStack Networking (Neutron) インスタンスの追加

1. **Administration → Providers** をクリックします。

2. **Add** をクリックし、**General Settings** タブに詳細を入力します。これらのフィールドの詳細は、「プロバイダーの一般設定の説明を追加」を参照してください。

3. **Name** および **Description** を入力します。

4. **Type** ドロップダウンリストから **OpenStack Networking** を選択します。

5. **Networking Plugin** フィールドで **Open vSwitch** が選択されていることを確認します。

6. 必要に応じて、**Automatic Synchronization** チェックボックスをオンにします。これにより、外部ネットワークプロバイダーと既存のネットワークの自動同期が可能になります。

7. OpenStack Networking インスタンスがインストールされているマシンの URL または完全修飾ドメイン名を **Provider URL** テキストフィールドに入力し、その後にポート番号を入力します。デフォルトでは、**Read-Only** チェックボックスがオンになっています。これにより、ユーザーが OpenStack Networking インスタンスを変更できなくなります。

重要

セットアップが Red Hat でサポートされるようにするには、**Read-Only** チェックボックスをオンのままにしておく必要があります。

8. 必要に応じて、**Requires Authentication** チェックボックスを選択し、Keystone に登録されている OpenStack Networking ユーザーの **Username** 名と **Password** を入力します。**Protocol, Hostname, API Port,** および **API Version** を定義して Keystone サーバーの認証 URL を定義する必要もあります。
API バージョン 2.0 の場合には、OpenStack Networking インスタンスの **Tenant** を入力します。API バージョン 3 の場合には、**User Domain Name**、**Project Name**、および **Project Domain Name** を入力します。

9. 認証情報をテストします。

a. **Test** をクリックして、提供された認証情報を使用して OpenStack Networking インスタンスで正常に認証できるかどうかをテストします。

b. OpenStack Networking インスタンスが SSL を使用する場合は、**Import provider**

certificates ウィンドウが開きます。**OK** をクリックして OpenStack Networking インスタンスが提供する証明書をインポートし、Manager がインスタンスと通信できるようにします。

10. **Agent Configuration** タブをクリックします。

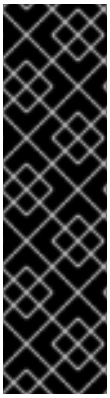
警告

以下の手順は、テクノロジープレビューとしてのみ提供されます。Red Hat Virtualization がサポートするのは、事前設定済みの neutron ホストだけです。

11. **Interface Mappings** フィールドに、Open vSwitch エージェントのインターフェイスマッピングのコンマ区切りリストを入力します。
12. **Broker Type** リストから OpenStack Networking インスタンスが使用するメッセージブローカータイプを選択します。
13. メッセージブローカーが **Host** フィールドにホストされるホストの URL または完全修飾ドメイン名を入力します。
14. メッセージブローカーに接続する **Port** を入力します。メッセージブローカーが SSL を使用するように設定されていない場合は、このポート番号は 5762 で、SSL を使用するように設定されている場合は 5761 となります。
15. メッセージブローカーインスタンスに登録された OpenStack Networking ユーザーの **Username** および **Password** を入力します。
16. **OK** をクリックします。

OpenStack Networking インスタンスが Red Hat Virtualization Manager に追加されました。提供されているネットワークを使用する前に、ネットワークを Manager にインポートしてください。「外部プロバイダーからのネットワークのインポート」を参照してください。

14.2.4. ストレージ管理用の OpenStack Block Storage (Cinder) インスタンスの追加



重要

ストレージ管理に OpenStack Block Storage (Cinder) インスタンスを使用する機能はテクノロジープレビュー機能です。テクノロジープレビュー機能は、Red Hat の実稼働環境でのサービスレベルアグリーメント (SLA) ではサポートされていないため、Red Hat では実稼働環境での使用を推奨していません。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。

Red Hat のテクノロジープレビュー機能のサポートについての詳細は、テクノロジープレビュー機能のサポート範囲 を参照してください。

Red Hat Virtualization Manager に、ストレージ管理用の OpenStack Block Storage (Cinder) インスタンスを追加します。OpenStack Cinder ボリュームは Ceph Storage によりプロビジョニングされます。

ストレージ管理用の OpenStack Block Storage (Cinder) インスタンスの追加

1. **Administration → Providers** をクリックします。
2. **Add** をクリックし、**General Settings** タブに詳細を入力します。これらのフィールドの詳細は、「プロバイダーの一般設定の説明を追加」 を参照してください。
3. **Name** および **Description** を入力します。
4. **Type** ドロップダウンリストから **OpenStack Block Storage** を選択します。
5. OpenStack Block Storage ボリュームの接続先となる **データセンター** を選択します。
6. OpenStack Block Storage インスタンスがインストールされているマシンの URL または完全修飾ドメイン名を入力し、その後に **Provider URL** のテキストフィールドにポート番号を入力します。
7. 必要に応じて、**Requires Authentication** チェックボックスを選択し、Keystone に登録されている OpenStack Block Storage インスタンスユーザーの **Username** 名と **Password** を入力します。**Protocol** (**HTTP** である必要があります)、**Hostname**、および **API Port** を定義して Keystone サーバーの認証 URL を定義する必要もあります。
   OpenStack Block Storage インスタンスの **Tenant** を入力します。
8. **Test** をクリックして、提供された認証情報を使用して OpenStack Block Storage インスタンスで正常に認証できるかどうかをテストします。
9. **OK** をクリックします。
10. クライアント Ceph 認証 (**cephx**) が有効になっている場合は、以下の手順も完了する必要があります。**cephx** プロトコルはデフォルトで有効になっています。
    a. Ceph サーバーで、**ceph auth get-or-create** コマンドを使用して **client.cinder** ユーザーの新しい秘密鍵を作成します。**cephx** に関する詳細は、Cephx 設定リファレンス を、新規ユーザーのキーを作成する情報は 新規ユーザーのキー作成 を参照してください。**client.cinder** ユーザーにキーがすでに存在する場合は、同じコマンドを使用してキーを取得します。
    b. 管理ポータルで、**Providers** 一覧から新たに作成された Cinder 外部プロバイダーを選択します。
    c. **Authentication Keys** タブをクリックします。
    d. **New** をクリックします。
    e. **Value** フィールドに秘密鍵を入力します。
    f. 自動生成された **UUID** をコピーするか、テキストフィールドに既存の UUID を入力します。
    g. Cinder サーバーで、前の手順の **cinder** ユーザーの UUID を **/etc/cinder/cinder.conf** に追加します。

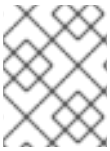
```
rbd_secret_uuid = UUID
rbd_user = cinder
```

OpenStack Block Storage (Cinder) ディスクの作成に関する詳細は、「仮想ディスクの作成」 を参照してください。

14.2.5. 仮想マシンプロバイダーとしての VMware インスタンスの追加

VMware vCenter インスタンスを追加して、仮想マシンを VMware から Red Hat Virtualization Manager にインポートします。

Red Hat Virtualization は、V2V を使用して、VMware 仮想マシンをインポートする前に正しい形式に変換します。**virt-v2v** パッケージが、1 つ以上のホストにインストールされている必要があります。Red Hat Virtualization Host (RHVH) では、**virt-v2v** パッケージがデフォルトで利用でき、Red Hat Virtualization 環境に追加されると、Red Hat Enterprise Linux ホストに VDSM の依存関係としてインストールされます。Red Hat Enterprise Linux ホストは、Red Hat Enterprise Linux 7.2 以降である必要があります。



注記

ppc64le アーキテクチャーでは **virt-v2v** パッケージが利用できず、これらのホストはプロキシーホストとして使用できません。

仮想マシンプロバイダーとしての VMware vCenter インスタンスの追加

1. **Administration → Providers** をクリックします。
2. **Add** をクリックします。
3. **Name** および **Description** を入力します。
4. **Type** ドロップダウンリストから **VMware** を選択します。
5. VMware 仮想マシンをインポートする **データセンター** を選択するか、**任意のデータセンター** を選択して、個々のインポート操作中に宛先データセンターを指定します。
6. **vCenter** フィールドに VMware vCenter インスタンスの IP アドレスまたは完全修飾ドメイン名を入力します。
7. **ESXi** フィールドに仮想マシンをインポートするホストの IP アドレスまたは完全修飾ドメイン名を入力します。
8. 指定した ESXi ホストが存在するデータセンターの名前を **Data Center** フィールドに入力します。
9. ESXi ホストと Manager との間で SSL 証明書を交換した場合は、**Verify server's SSL certificate** チェックボックスを選択したままにして、ESXi ホストの証明書を確認します。交換していない場合は、チェックボックスの選択を解除します。
10. 仮想マシンのインポート操作中に **Proxy Host** として機能するように、**virt-v2v** がインストールされている、選択したデータセンター内のホストを選択します。このホストは、VMware vCenter 外部プロバイダーのネットワークに接続できる必要もあります。上記の **Any Data Center** を選択した場合は、ここでホストを選択することはできませんが、代わりに、個別のインポート操作時にホストを指定できます。
11. VMware vCenter インスタンスの **Username** および **Password** を入力します。ユーザーは、仮想マシンが置かれている VMware データセンターおよび ESXi ホストにアクセスできる必要があります。
12. 認証情報をテストします。
    a. **Test** をクリックして、提供された認証情報を使用して VMware vCenter インスタンスで正常に認証できるかどうかをテストします。

b. VMware vCenter インスタンスが SSL を使用している場合は、**Import provider certificates** ウィンドウが開きます。**OK** をクリックして、VMware vCenter インスタンスが提供する証明書をインポートし、Manager がインスタンスと通信できるようにします。

13. **OK** をクリックします。

VMware 外部プロバイダーから仮想マシンをインポートするには、**Virtual Machine Management Guide** の Importing a Virtual Machine from a VMware Provider を参照してください。

14.2.6. RHEL 5 Xen ホストの仮想マシンプロバイダーとしての追加

RHEL 5 Xen ホストを追加して、仮想マシンを Xen から Red Hat Virtualization にインポートします。

Red Hat Virtualization は、V2V を使用して、RHEL 5 Xen 仮想マシンをインポートする前に正しい形式に変換します。**virt-v2v** パッケージが、1 つ以上のホストにインストールされている必要があります。Red Hat Virtualization Host (RHVH) では、**virt-v2v** パッケージがデフォルトで利用でき、Red Hat Virtualization 環境に追加されると、Red Hat Enterprise Linux ホストに VDSM の依存関係としてインストールされます。Red Hat Enterprise Linux ホストは、Red Hat Enterprise Linux 7.2 以降である必要があります。

注記

ppc64le アーキテクチャーでは **virt-v2v** パッケージが利用できず、これらのホストはプロキシーホストとして使用できません。

RHEL 5 Xen インスタンスの仮想マシンプロバイダーとしての追加

1. プロキシーホストと RHEL 5 ホスト間の公開鍵認証を有効にします。

   a. プロキシーホストにログインし、**vdsm** ユーザーの SSH キーを生成します。

   ```
   # sudo -u vdsm ssh-keygen
   ```

   b. **vdsm** ユーザーの公開鍵を RHEL 5 Xen ホストにコピーします。プロキシーホストの **known_hosts** ファイルも更新され、RHEL 5 Xen ホストのホストキーが追加されます。

   ```
   # sudo -u vdsm ssh-copy-id root@xenhost.example.com
   ```

   c. RHEL 5 Xen ホストにログインして、ログインが正常に機能していることを確認します。

   ```
   # sudo -u vdsm ssh root@xenhost.example.com
   ```

2. **Administration → Providers** をクリックします。

3. **Add** をクリックします。

4. **Name** および **Description** を入力します。

5. **Type** ドロップダウンリストから **XEN** を選択します。

6. Xen 仮想マシンをインポートする **データセンター** を選択するか、**任意のデータセンター** を選択して、個々のインポート操作中に宛先データセンターを指定します。

7. **URI** フィールドに RHEL 5 Xen ホストの URI を入力します。

8. 仮想マシンのインポート操作中に **Proxy Host** として機能するように、**virt-v2v** がインストールされている、選択したデータセンター内のホストを選択します。このホストは、RHEL 5 Xen 外部プロバイダーのネットワークにも接続できる必要があります。上記の **Any Data Center** を選択した場合は、ここでホストを選択することはできませんが、代わりに、個別のインポート操作時にホストを指定できます。

9. **Test** をクリックして、RHEL 5 Xen ホストで正常に認証できるかどうかをテストします。

10. **OK** をクリックします。

RHEL 5 Xen 外部プロバイダーから仮想マシンをインポートするには、**Virtual Machine Management Guide** の Importing a Virtual Machine from a RHEL 5 Xen Host を参照してください。

14.2.7. KVM ホストの仮想マシンプロバイダーとしての追加

KVM ホストを追加して、仮想マシンを KVM から Red Hat VirtualizationManager にインポートします。

KVM ホストの仮想マシンプロバイダーとしての追加

1. プロキシーホストと KVM ホスト間の公開鍵認証を有効にします。

   a. プロキシーホストにログインし、**vdsm** ユーザーの SSH キーを生成します。

   ```
   # sudo -u vdsm ssh-keygen
   ```

   b. **vdsm** ユーザーの公開鍵を KVM ホストにコピーします。プロキシーホストの **known_hosts** ファイルも更新され、KVM ホストのホストキーが追加されます。

   ```
   # sudo -u vdsm ssh-copy-id root@kvmhost.example.com
   ```

   c. KVM ホストにログインして、ログインが正常に機能していることを確認します。

   ```
   # sudo -u vdsm ssh root@kvmhost.example.com
   ```

2. **Administration → Providers** をクリックします。

3. **Add** をクリックします。

4. **Name** および **Description** を入力します。

5. **Type** ドロップダウンリストから **KVM** を選択します。

6. KVM 仮想マシンをインポートする **データセンター** を選択するか、**任意のデータセンター** を選択して、個々のインポート操作中に宛先データセンターを指定します。

7. **URI** フィールドに KVM ホストの URI を入力します。

   ```
   qemu+ssh://root@host.example.com/system
   ```

8. 選択したデータセンターで、仮想マシンのインポート操作中に **プロキシーホスト** として機能するホストを選択します。このホストは、KVM 外部プロバイダーのネットワークにも接続できる必要があります。上記の **Data Center** フィールドで **Any Data Center** を選択した場合、ここでホストを選択することはできません。フィールドはグレー表示され、**Any Host in Data Center** が表示されます。代わりに、個別のインポート操作中にホストを指定できます。

9. 必要に応じて、**Requires Authentication** チェックボックスを選択し、KVM ホストの **Username** 名と **Password** を入力します。ユーザーは、仮想マシンが存在する KVM ホストにアクセスできる必要があります。

10. **Test** をクリックし、提供された認証情報を使用して、KVM ホストで正常に認証できるかどうかをテストします。

11. **OK** をクリックします。

KVM 外部プロバイダーから仮想マシンをインポートするには、**Virtual Machine Management Guide** の Importing a Virtual Machine from a KVM Host を参照してください。

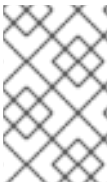
14.2.8. 外部ネットワークプロバイダーとしてのオープン仮想ネットワーク (OVN) の追加

Open Virtual Network (OVN) を使用すると、VLAN を追加したりインフラストラクチャーを変更したりせずにネットワークを作成できます。OVN は Open vSwitch(OVS) の拡張機能で、仮想 L2 および L3 オーバーレイのネイティブ OVS サポートを追加することで仮想ネットワークをサポートします。

新しい OVN ネットワークプロバイダーをインストール することも、既存の OVN ネットワークプロバイダーを追加 することもできます。

OVN ネットワークをネイティブの Red Hat Virtualization ネットワークに接続することもできます。詳細は、「OVN ネットワークを物理ネットワークに接続」 を参照してください。この機能は、テクノロジープレビューとしてのみ利用できます。

Neutron と同様の REST API は **ovirt-provider-ovn** によって公開され、ネットワーク、サブネット、ポート、およびルーターを作成できます (詳細は、**OpenStack Networking API v2.0** を参照してください)。これらのオーバーレイネットワークは、仮想マシン間の通信を可能にします。



注記

OVN は、OpenStack (Neutron) API を使用して CloudForms によって外部プロバイダーとしてサポートされます。詳細は、**Red Hat CloudForms: Managing Providers** の Network Managers を参照してください。

OVS および OVN の詳細は、OVS ドキュメントの http://docs.openvswitch.org/en/latest/ および http://openvswitch.org/support/dist-docs/ を参照してください。

14.2.8.1. 新しい OVN ネットワークプロバイダーのインストール

警告

openvswitch パッケージがすでにインストールされていて、バージョンが 1:2.6.1 (バージョン 2.6.1 エポック 1) の場合、最新の **openvswitch** パッケージをインストールしようとすると、OVN のインストールは失敗します。詳細と回避策は、BZ#1505398 の Doc Text を参照してください。

engine-setup を使用して OVN をインストールする場合、以下の手順は自動化されます。

- Manager マシンに OVN 中央サーバーをセットアップします。
- 外部ネットワークプロバイダーとして OVN を Red Hat Virtualization に追加します。
- **Default** クラスターのデフォルトネットワークプロバイダーを **ovirt-provider-ovn** に設定します。
- クラスターへの追加時に OVN と通信するようにホストを設定します。

engine-setup で事前設定された応答ファイルを使用する場合は、次のエントリーを追加して OVN をインストールできます。

```
OVESETUP_OVN/ovirtProviderOvn=bool:True
```

新しい OVN ネットワークプロバイダーのインストール

1. engine-setup を使用して Manager に OVN をインストールします。インストール時に、**engine-setup** は以下の質問をします。

   ```
   # Install ovirt-provider-ovn(Yes, No) [Yes]?:
   ```

   - **Yes** の場合には、engine-setup は **ovirt-provider-ovn** をインストールします。**engine-setup** がシステムを更新する場合に、このプロンプトは **ovirt-provider-ovn** が以前にインストールされていない場合にのみ表示されます。
   - **No** の場合は、**engine-setup** を次回実行しても再度尋ねられることはありません。このオプションを表示するには、**engine-setup --reconfigure-optional-components** を実行します。

     ```
     # Use default credentials (admin@internal) for ovirt-provider-ovn(Yes, No) [Yes]?:
     ```

     Yes の場合、**engine-setup** は、セットアッププロセスの前半で指定されたデフォルトのエンジンユーザーとパスワードを使用します。このオプションは、新規インストール時にのみ使用できます。

     ```
     # oVirt OVN provider user[admin]:
     # oVirt OVN provider password[empty]:
     ```

     デフォルト値を使用するか、oVirt OVN プロバイダーのユーザーとパスワードを指定できます。

     注記

     後で認証方法を変更するには、**/etc/ovirt-provider-ovn/conf.d/10_engine_setup.conf** ファイルを編集するか、新しい **/etc/ovirt-provider-ovn/conf.d/20_engine_setup.conf** ファイルを作成します。変更を有効にするには、**ovirt-provider-ovn** サービスを再起動します。OVN 認証に関する詳細は、https://github.com/oVirt/ovirt-provider-ovn/blob/master/README.adoc を参照してください。

2. **Default** クラスターにホストを追加します。このクラスターに追加されたホストは、OVN と通信するように自動的に設定されます。新規のホストを追加するには、「Red Hat Virtualization Manager への通常のホストの追加」 を参照してください。

ホストが既存のデフォルト以外のネットワークを使用するように設定するには、「OVN トンネルネットワークのホストの設定」を参照してください。

3. ネットワークを **Default** クラスターに追加します。「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」を参照して、**Create on external provider** チェックボックスを選択します。デフォルトでは、**ovirt-provider-ovn** が選択されています。

4. OVN ネットワークをネイティブの Red Hat Virtualization ネットワークに接続するには、**物理ネットワークに接続する** チェックボックスをオンにして、使用する Red Hat Virtualization ネットワークを指定します。詳細および前提条件については、「OVN ネットワークを物理ネットワークに接続」を参照してください。

5. ネットワークが **Security Groups** のドロップダウンからのセキュリティーグループを使用するかどうかを定義します。利用可能なオプションの詳細は、「論理ネットワーク一般設定の説明」を参照してください。これで、OVN ネットワークを使用する仮想マシンを作成できます。

14.2.8.2. 既存の OVN ネットワークプロバイダーの追加

Red Hat Virtualization で外部ネットワークプロバイダーとして既存の OVN セントラルサーバーを追加するには、次の重要な手順が必要です。

- Manager が OVN と対話するために使用するプロキシーである OVN プロバイダーをインストールします。OVN プロバイダーは任意のマシンにインストールできますが、OVN 中央サーバーおよび Manager と通信できる必要があります。
- OVN プロバイダーを外部ネットワークプロバイダーとして Red Hat Virtualization に追加します。
- デフォルトのネットワークプロバイダーとして OVN を使用する新しいクラスターを作成します。このクラスターに追加されたホストは、OVN と通信するように自動的に設定されます。

前提条件

次のパッケージは OVN プロバイダーに必要であり、プロバイダーマシンで使用可能である必要があります。

- openvswitch-ovn-central
- openvswitch
- openvswitch-ovn-common
- python-openvswitch

これらのパッケージがプロバイダーマシンですでに有効になっているリポジトリーから利用できない場合は、OVS の Web サイト (http://openvswitch.org/download/) からダウンロードできます。

既存の OVN ネットワークプロバイダーの追加

1. OVN プロバイダーをインストールして設定します。

   a. プロバイダーをプロバイダーマシンにインストールします。

   ```
   # yum install ovirt-provider-ovn
   ```

   b. Manager と同じマシンにプロバイダーをインストールしない場合は、次のエントリーを

b. Manager と同じマシンにプロバイダーをインストールしない場合は、次のエントリーを **/etc/ovirt-provider-ovn/conf.d/10_engine_setup.conf** ファイルに追加します (このファイルがまだ存在しない場合は作成します)。

```
[OVIRT]
ovirt-host=https://Manager_host_name
```

認証が有効になっている場合、これは認証に使用されます。

c. プロバイダーを OVN 中央サーバーと同じマシンにインストールしない場合は、次のエントリーを **/etc/ovirt-provider-ovn/conf.d/10_engine_setup.conf** ファイルに追加します (このファイルがまだ存在しない場合は作成します)。

```
[OVN REMOTE]
ovn-remote=tcp:OVN_central_server_IP:6641
```

d. ファイアウォールのポート 9696、6641、および 6642 を開いて、OVN プロバイダー、OVN 中央サーバー、および Manager 間の通信を許可します。これは、手動で行うことも、**ovirt-provider-ovn** および **ovirt-provider-ovn-central** サービスを適切なゾーンに追加することによって行うこともできます。

```
# firewall-cmd --zone=ZoneName --add-service=ovirt-provider-ovn --permanent
# firewall-cmd --zone=ZoneName --add-service=ovirt-provider-ovn-central --permanent
# firewall-cmd --reload
```

e. サービスを開始して有効にします。

```
# systemctl start ovirt-provider-ovn
# systemctl enable ovirt-provider-ovn
```

f. ポート 6642 および 6641 からの要求をリッスンするように OVN 中央サーバーを設定します。

```
# ovn-sbctl set-connection ptcp:6642
# ovn-nbctl set-connection ptcp:6641
```

2. 管理ポータルで、**Administration → Providers** をクリックします。

3. **Add** をクリックし、**General Settings** タブに詳細を入力します。これらのフィールドの詳細は、「プロバイダーの一般設定の説明を追加」 を参照してください。

4. **Name** および **Description** を入力します。

5. **Type** リストから、**External Network Provider** を選択します。

6. **Networking Plugin** テキストボックスをクリックし、ドロップダウンメニューから **oVirt Network Provider for OVN** を選択します。

7. 必要に応じて、**Automatic Synchronization** チェックボックスをオンにします。これにより、外部ネットワークプロバイダーと既存のネットワークの自動同期が可能になります。

注記

自動同期は、**engine-setup** ツールによって作成された **ovirt-provider-ovn** ネットワークプロバイダーでデフォルトで有効になっています。

8. OVN プロバイダーの URL または完全修飾ドメイン名を **Provider URL** テキストフィールドに入力し、その後にポート番号を入力します。OVN プロバイダーと OVN 中央サーバーが別々のマシン上にある場合、これはプロバイダーマシンの URL であり、中央サーバーではありません。OVN プロバイダーが Manager と同じマシン上にある場合、URL はデフォルトの **http://localhost:9696** のままにすることができます。

9. Red Hat Virtualization Manager から新しい OVN ネットワークを作成できるようにするには、**Read-Only** チェックボックスをオフにします。

10. 必要に応じて、**Requires Authentication** チェックボックスをオンにし、Keystone に登録されている外部ネットワークプロバイダーユーザーの **ユーザー名** と **パスワード** を入力します。**Protocol**、**Hostname**、および **API Port** を定義して Keystone サーバーの認証 URL を定義する必要もあります。
必要に応じて、外部ネットワークプロバイダーの **Tenant** を入力します。

    認証方法は、**/etc/ovirt-provider-ovn/conf.d/10_engine_setup.conf** ファイルで設定する必要があります (このファイルがまだ存在しない場合は作成してください)。変更を有効にするには、**ovirt-provider-ovn** サービスを再起動します。OVN 認証に関する詳細は、https://github.com/oVirt/ovirt-provider-ovn/blob/master/README.adoc を参照してください。

11. 認証情報をテストします。

    a. **Test** をクリックし、提供された認証情報を使用して、OVN で正常に認証できるかどうかをテストします。

    b. OVN インスタンスが SSL を使用している場合は、**Import provider certificates** ウィンドウが開きます。**OK** をクリックして、OVN インスタンスが提供する証明書をインポートし、Manager がインスタンスと通信できるようにします。

12. **OK** をクリックします。

13. デフォルトのネットワークプロバイダーとして OVN を使用する新しいクラスターを作成します。「新規クラスターの作成」 を参照して、**Default Network Provider** ドロップダウンリストから OVN ネットワークプロバイダーを選択します。

14. ホストをクラスターに追加します。このクラスターに追加されたホストは、OVN と通信するように自動的に設定されます。新規のホストを追加するには、「Red Hat Virtualization Manager への通常のホストの追加」 を参照してください。

15. OVN ネットワークを新しいクラスターにインポートまたは追加します。ネットワークをインポートするには、Importing Networks を参照してください。OVN を使用して新規のネットワークを作成するには、Creating a new logical network in a data center or cluster を参照し、**Create on external provider** チェックボックスを選択します。デフォルトでは、**ovirt-provider-ovn** が選択されています。
ホストが既存のデフォルト以外のネットワークを使用するように設定するには、「OVN トンネルネットワークのホストの設定」 を参照してください。

    OVN ネットワークをネイティブの Red Hat Virtualization ネットワークに接続するには、**物理ネットワークに接続する** チェックボックスをオンにして、使用する Red Hat Virtualization ネットワークを指定します。詳細および前提条件については、「OVN ネットワークを物理ネットワークに接続」 を参照してください。

これで、OVN ネットワークを使用する仮想マシンを作成できます。

14.2.8.3. Ansible Playbook を使用して OVN トンネルネットワークの変更

ovirt-provider-ovn-driver Ansible Playbook を使用して、長い名前を使用して OVN コントローラーのトンネルネットワークを変更できます。

OVN トンネルネットワークを変更するための Ansible Playbook

```
# ansible-playbook --key-file <path_to_key_file> -i <path_to_inventory> --extra-vars " cluster_name=<cluster_name> ovn_central=<ovn_central_ip_address> ovirt_network=<ovirt network name> ovn_tunneling_interface=<vdsm_network_name>" ovirt-provider-ovn-driver.yml
```

パラメーター

key-file

ホストにログインするためのキーファイル。デフォルトのキーファイルは、通常、**/etc/pki/ovirt-engine/keys** ディレクトリーにあります。

inventory

oVirt VM インベントリー。在庫値を見つけるには、**/usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory** スクリプトを使用します。

cluster_name

名前を更新するクラスターの名前。

ovn_central

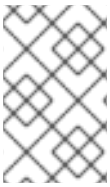
OVN 中央サーバーへの IP アドレス。この IP アドレスは、すべてのホストがアクセスできる必要があります。

ovirt_network

oVirt ネットワーク名。

ovn_tunneling_interface

VDSM ネットワーク名。



注記

ovirt-provider-ovn-driver Ansible Playbook は、**ovirt_network** パラメーターまたは **ovn_tunneling_interface** パラメーターのいずれかの使用をサポートしています。両方のパラメーターが同じ Playbook に存在する場合、この Playbook は失敗します。

ovirt_network パラメーターを使用した Playbook

```
# ansible-playbook --key-file /etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars " cluster_name=test-cluster ovn_central=192.168.200.2 ovirt_network=\"Long\ Network\ Name\ with\ \Ascii\ character\ \☺\"" ovirt-provider-ovn-driver.yml
```

ovn_tunneling_interface パラメーターを使用した Playbook

```
# ansible-playbook --key-file /etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars " cluster_name=test-cluster ovn_central=192.168.200.2 ovn_tunneling_interface=on703ea21ddbc34" ovirt-provider-ovn-driver.yml
```

Manager マシンで、**/usr/share/ovirt-engine/playbooks** ディレクトリーに移動して、Ansible Playbook を実行します。

14.2.8.4. OVN トンネルネットワークのホストの設定

ovirt-provider-ovn-driver Ansible Playbook を使用して、デフォルトの **ovirtmgmt** ネットワーク以外の既存のネットワークを使用するようにホストを設定できます。ネットワークは、クラスター内のすべてのホストからアクセス可能である必要があります。

注記

ovirt-provider-ovn-driver Ansible Playbook は、既存のホストを更新します。クラスターに新しいホストを追加する場合は、Playbook を再度実行する必要があります。

OVN トンネルネットワークのホストの設定

1. Manager マシンで、**playbooks** ディレクトリーに移動します。

```
# cd /usr/share/ovirt-engine/playbooks
```

2. 次のパラメーターを指定して **ansible-playbook** コマンドを実行します。

```
# ansible-playbook --private-key=/etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-
engine-metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars
" cluster_name=Cluster_Name ovn_central=OVN_Central_IP
ovn_tunneling_interface=VDSM_Network_Name" ovirt-provider-ovn-driver.yml
```

以下に例を示します。

```
# ansible-playbook --private-key=/etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-
engine-metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars
" cluster_name=MyCluster ovn_central=192.168.0.1 ovn_tunneling_interface=MyNetwork"
ovirt-provider-ovn-driver.yml
```

注記

OVN_Central_IP は新しいネットワーク上に配置できますが、これは必須ではありません。OVN_Central_IP は、すべてのホストからアクセス可能である必要があります。

VDSM_Network_Name は 15 文字に制限されています。15 文字より長い、または非 ASCII 文字を含む論理ネットワーク名を定義した場合は、15 文字の名前が自動的に生成されます。これらの名前のマッピングを表示する手順については、Mapping VDSM Names to Logical Network Names を参照してください。

単一ホスト上の OVN トンネルネットワークの更新

vdsm-tool を使用して、単一のホスト上の OVN トンネルネットワークを更新できます。

```
# vdsm-tool ovn-config OVN_Central_IP Tunneling_IP_or_Network_Name
```

例14.1 vdsm-tool を使用したホストの更新

```
# vdsm-tool ovn-config 192.168.0.1 MyNetwork
```

14.2.8.5. OVN ネットワークを物理ネットワークに接続

重要

この機能は、Red Hat Virtualization のテクノロジープレビューとしてのみ利用可能な Open vSwitch サポートに依存しています。テクノロジープレビュー機能は、Red Hat の実稼働環境でのサービスレベルアグリーメント (SLA) ではサポートされておらず、機能的に完全ではない可能性があるため、Red Hat では実稼働環境での使用を推奨していません。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。

Red Hat のテクノロジープレビュー機能のサポートについての詳細は、テクノロジープレビュー機能のサポート範囲 を参照してください。

ネイティブの Red Hat Virtualization ネットワークをオーバーレイする外部プロバイダーネットワークを作成して、それぞれの仮想マシンが同じサブネットを共有しているように見せることができます。

重要

OVN ネットワークのサブネットを作成した場合、そのネットワークを使用する仮想マシンはそこから IP アドレスを受け取ります。物理ネットワークに IP アドレスを割り当てたい場合は、OVN ネットワークのサブネットを作成しないでください。

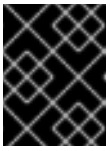
前提条件

- クラスターでは、**スイッチタイプ** として **OVS** が選択されている必要があります。このクラスターに追加されたホストには、**ovirtmgmt** ブリッジなどの既存の Red Hat Virtualization ネットワークを設定してはなりません。
- 物理ネットワークはホストで利用可能である必要があります。これを実施するには、(**Manage Networks** ウィンドウ、または **New Logical Network** ウィンドウの **Cluster** タブで) クラスターに必要な物理ネットワークを設定します。

物理ネットワークに接続された新規外部ネットワークの作成

1. **Compute** → **Clusters** をクリックします。
2. クラスターの名前をクリックして、詳細ビューに移動します。
3. **Logical Networks** タブをクリックし、**Add Network** をクリックします。
4. ネットワークの **Name** を入力します。
5. **Create on external provider** チェックボックスをオンにします。デフォルトでは、**ovirt-provider-ovn** が選択されています。
6. デフォルトでまだ選択されていない場合は、**Connect to physical network** チェックボックスをオンにします。
7. 新しいネットワークを接続する物理ネットワークを選択します。
   - **Data Center Network** ラジオボタンをクリックし、ドロップダウンリストから物理ネットワークを選択します。これは推奨されるオプションです。

- **Custom** ラジオボタンをクリックして、物理ネットワークの名前を入力します。物理ネットワークで VLAN タギングが有効になっている場合は、**Enable VLAN tagging** チェックボックスをオンにして、物理ネットワークの VLAN タグも入力する必要があります。



重要

物理ネットワークの名前は 15 文字を超えてはならず、特殊文字を含んでいてはなりません。

8. **OK** をクリックします。

14.2.9. 外部ネットワークプロバイダーの追加

OpenStack Neutron REST API を実装する任意のネットワークプロバイダーを Red Hat Virtualization に追加できます。仮想インターフェイスドライバーは、外部ネットワークプロバイダーの実装者が提供する必要があります。ネットワークプロバイダーおよび仮想インターフェイスドライバーの参照実装は、https://github.com/mmirecki/ovirt-provider-mock および https://github.com/mmirecki/ovirt-provider-mock/blob/master/docs/driver_instalation から利用できます。

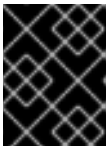
ネットワークプロビジョニング用の外部ネットワークプロバイダーの追加

1. **Administration → Providers** をクリックします。
2. **Add** をクリックし、**General Settings** タブに詳細を入力します。これらのフィールドの詳細は、「プロバイダーの一般設定の説明を追加」 を参照してください。
3. **Name** および **Description** を入力します。
4. **タイプ** ドロップダウンリストから **外部ネットワークプロバイダー** を選択します。
5. 必要に応じて、**Networking Plugin** テキストボックスをクリックし、ドロップダウンメニューから適切なドライバーを選択します。
6. 必要に応じて、**Automatic Synchronization** チェックボックスをオンにします。これにより、外部ネットワークプロバイダーと既存のネットワークの自動同期が可能になります。この機能は、外部ネットワークプロバイダーを追加する際にデフォルトで無効にされます。

注記

自動同期は、**engine-setup** ツールによって作成された **ovirt-provider-ovn** ネットワークプロバイダーでデフォルトで有効になっています。

7. 外部ネットワークプロバイダーがインストールされているマシンの URL または完全修飾ドメイン名を **Provider URL** テキストフィールドに入力し、その後にポート番号を入力します。デフォルトでは、**Read-Only** チェックボックスがオンになっています。これにより、ユーザーは外部ネットワークプロバイダーを変更できなくなります。



重要

セットアップが Red Hat でサポートされるようにするには、**Read-Only** チェックボックスをオンのままにしておく必要があります。

8. 必要に応じて、**Requires Authentication** チェックボックスをオンにし、Keystone に登録されている外部ネットワークプロバイダーユーザーの **ユーザー名** と **パスワード** を入力しま

す。**Protocol**、**Hostname**、および **API Port** を定義して Keystone サーバーの認証 URL を定義する必要もあります。
必要に応じて、外部ネットワークプロバイダーの **Tenant** を入力します。

9. 認証情報をテストします。

   a. **Test** をクリックし、提供された認証情報を使用して、外部ネットワークプロバイダーで正常に認証できるかどうかをテストします。

   b. 外部ネットワークプロバイダーが SSL を使用している場合は、**Import provider certificates** ウィンドウが開きます。**OK** をクリックして、外部ネットワークプロバイダーが提供する証明書をインポートし、Manager がインスタンスと通信できるようにします。

10. **OK** をクリックします。

このプロバイダーのネットワークを使用する前に、ホストに仮想インターフェイスドライバーをインストールし、ネットワークをインポートする必要があります。ネットワークをインポートするには、「外部プロバイダーからのネットワークのインポート」 を参照してください。

14.2.10. プロバイダーの一般設定の説明を追加

Add Provider ウィンドウの **General** タブでは、外部プロバイダーのコアの詳細を登録できます。

表14.1 プロバイダーの追加: 一般設定

設定	説明
Name	Manager でプロバイダーを表す名前。
説明	プロバイダーのプレーンテキストで人間が読める形式の説明。
タイプ	外部プロバイダーのタイプ。この設定を変更すると、プロバイダーの設定に使用できるフィールドが変更されます。 **Foreman/Satellite** ● **Provider URL**: Satellite インスタンスをホストするマシンの URL または完全修飾ドメイン名。URL または完全修飾ドメイン名の末尾にポート番号を追加する必要はありません。 ● **Requires Authentication**: プロバイダーに認証が必要かどうかを指定できます。**Foreman/Satellite** が選択されている場合には、認証は必須です。 ● **Username**: Satellite インスタンスに接続するためのユーザー名。このユーザー名は、Satellite インスタンスのプロビジョニングポータルへのログインに使用されるユーザー名でなければなりません。 ● **Password**: 上記のユーザー名が認証されるパスワード。このパスワードは、Satellite インスタンスのプロビジョニングポータル

設定	説明
	へのログインに使用するパスワードでなければなりません。 **OpenStack Image** ● **Provider URL**: OpenStack Image Service がホストされているマシンの URL または完全修飾ドメイン名。OpenStack Image Service のポート番号を URL または完全修飾ドメイン名の末尾に追加する必要があります。デフォルトでは、このポート番号は 9292 です。 ● **Requires Authentication**: OpenStack Image サービスにアクセスするために認証が必要であるかどうかを指定できます。 ● **Username**: Keystone サーバーに接続するためのユーザー名このユーザー名は、OpenStack Image サービスが所属する Keystone インスタンスに登録されている OpenStack Image サービスのユーザー名でなければなりません。 ● **Password**: 上記のユーザー名が認証されるパスワード。このパスワードは、OpenStack Image サービスが所属する Keystone インスタンスに登録されている OpenStack Image サービスのパスワードでなければなりません。 ● **Protocol**: Keystone サーバーと通信するために使用するプロトコル。**HTTP** に設定する必要があります。 ● **Hostname**: Keystone サーバーの IP アドレスまたはホスト名。 ● **API port**: Keystone サーバーの API ポート番号。 ● **API Version**: Keystone サービスのバージョン。値は **v2.0** で、フィールドは無効になっています。 ● **Tenant Name**: OpenStack Image サービスが所属する OpenStack テナントの名前。 **OpenStack ネットワーキング** ● **Networking Plugin**: OpenStack Networking サーバーに接続するためのネットワークプラグイン。OpenStack Networking の場合、**Open vSwitch** は唯一のオプションで、デフォルトで選択されています。 ● **Automatic Synchronization**: プロバイダーが既存のネットワークと自動的に同期されるかどうかを指定できます。 ● **Provider URL**: OpenStack Networking インスタンスがホストされるマシンの URL または完全修飾ドメイン名。OpenStack Networking インスタンスのポート番号を、

設定	説明
	URL または完全修飾ドメイン名の末尾に追加する必要があります。デフォルトでは、このポート番号は 9696 です。 ● **Read Only**: OpenStack Networking インスタンスを管理ポータルから変更できるかどうかを指定できます。 ● **Requires Authentication**: OpenStack Networking サービスへのアクセスに認証が必要かどうかを指定できます。 ● **Username**: OpenStack Networking インスタンスに接続するためのユーザー名。このユーザー名は、OpenStack Networking インスタンスが所属する Keystone インスタンスに登録されている OpenStack Networking のユーザー名でなければなりません。 ● **Password**: 上記のユーザー名が認証されるパスワード。このパスワードは、OpenStack Networking インスタンスが所属する Keystone インスタンスに登録されている OpenStack Networking のパスワードでなければなりません。 ● **Protocol**: Keystone サーバーと通信するために使用するプロトコル。デフォルトは **HTTPS** です。 ● **Hostname**: Keystone サーバーの IP アドレスまたはホスト名。 ● **API port**: Keystone サーバーの API ポート番号。 ● **API Version**: Keystone サーバーのバージョン。これは URL に表示されます。v2.0 が表示されたら、**v2.0** を選択します。v3 が表示されたら、**v3** を選択します。 **API Version** フィールドで **v3** を選択すると、以下のフィールドが表示されます。 ● **User Domain Name**: ドメインで定義されたユーザーの名前。 Keystone API v3 では、ドメインを使用して、OpenStack のサービスエンティティーの管理上の境界を決定します。ドメインを使用すると、ドメイン固有の設定やセキュリティーオプションの設定など、さまざまな目的でユーザーをグループ化できます。詳細は、Red Hat OpenStack Platform **Architecture Guide** の OpenStack Identity (keystone) を参照してください。 ● **Project Name**: OpenStack Identity API v3 のプロジェクト名を定義します。 ● **Project Domain Name**: OpenStack Identity API v3 のプロジェクトのドメイン名を定義します。 **API Version** フィールドから **v2.0** を選択すると、次のフィールドが表示されます。

設定	説明
	• **Tenant Name**: **API Version** フィールドから v2 が選択されている場合にのみ表示されます。OpenStack Networking インスタンスが所属する OpenStack テナントの名前。 **OpenStack ボリューム** • **Data Center**: OpenStack ボリュームストレージボリュームが接続されるデータセンター。 • **Provider URL**: OpenStack Volume インスタンスがホストされるマシンの URL または完全修飾ドメイン名。OpenStack Volume インスタンスのポート番号を、URL または完全修飾ドメイン名の末尾に追加する必要があります。デフォルトでは、このポート番号は 8776 です。 • **Requires Authentication**: OpenStack ボリュームサービスへのアクセスに認証が必要であるかどうかを指定できます。 • **Username**: Keystone サーバーに接続するためのユーザー名このユーザー名は、OpenStack Volume インスタンスが所属する Keystone インスタンスに登録されている OpenStack Volume のユーザー名でなければなりません。 • **Password**: 上記のユーザー名が認証されるパスワード。このパスワードは、OpenStack Volume インスタンスが所属する Keystone インスタンスに登録されている OpenStack Volume のパスワードでなければなりません。 • **Protocol**: Keystone サーバーと通信するために使用するプロトコル。**HTTP** に設定する必要があります。 • **Hostname**: Keystone サーバーの IP アドレスまたはホスト名。 • **API port**: Keystone サーバーの API ポート番号。 • **API Version**: Keystone サーバーのバージョン。値は **v2.0** で、フィールドは無効になっています。 • **テナント名** :OpenStack Volume インスタンスがメンバーになっている OpenStack テナントの名前。 **VMware** • **Data Center**: VMware 仮想マシンがインポートされるデータセンターを指定するか、**Any Data Center** を選択して、(**Virtual Machines** タブの **Import** 機能を使用して) 個々のインポート操作中に宛先データセンターを指定します。

設定	説明
	• **vCenter**: VMware vCenter インスタンスの IP アドレスまたは完全修飾ドメイン名。 • **ESXi**: 仮想マシンのインポート元となるホストの IP アドレスまたは完全修飾ドメイン名。 • **Data Center**: 指定された ESXi ホストが存在するデータセンターの名前。 • **Cluster**: 指定された ESXi ホストが存在するクラスターの名前。 • **Verify server's SSL certificate**: 接続時に ESXi ホストの証明書を確認するかどうかを指定します。 • **Proxy Host** 仮想マシンのインポート操作中にホストとして機能するように、選択したデータセンターの **virt-v2v** をインストールしたホストを選択します。このホストは、VMware vCenter 外部プロバイダーのネットワークに接続できる必要もあります。**Any Data Center** を選択した場合は、ここでホストを選択することはできませんが、個別のインポート操作時にホストを指定できます (**Virtual Machines** タブの **Import** 機能を使用)。 • **ユーザー名**: VMware vCenter インスタンスに接続するためのユーザー名。ユーザーは、仮想マシンが置かれている VMware データセンターおよび ESXi ホストにアクセスできる必要があります。 • **Password**: 上記のユーザー名が認証されるパスワード。 RHEL 5 Xen • **Data Center**: Xen 仮想マシンがインポートされるデータセンターを指定するか、**Any Data Center** を選択して、(**Virtual Machines** タブの **Import** 機能を使用して) 個々のインポート操作中に宛先データセンターを指定します。 • **URI**: RHEL 5 Xen ホストの URI を入力します。 • **Proxy Host** 仮想マシンのインポート操作中にホストとして機能するように、選択したデータセンターの **virt-v2v** をインストールしたホストを選択します。このホストは、RHEL 5 Xen 外部プロバイダーのネットワークにも接続できる必要があります。**Any Data Center** を選択した場合は、ここでホストを選択することはできませんが、代わりに、個別のインポート操作時にホストを指定できます (**Virtual Machines** タブの **Import** 機能を使用)。 KVM • **Data Center**: KVM 仮想マシンがインポート

設定	説明
	されるデータセンターを指定するか、**Any Data Center** を選択して、(**Virtual Machines** タブの **Import** 機能を使用して) 個々のインポート操作中に宛先データセンターを指定します。 ● **URI** :KVM ホストの URI。 ● **Proxy Host**: 選択したデータセンターで、仮想マシンのインポート操作中にホストとして機能するホストを選択します。このホストは、KVM 外部プロバイダーのネットワークにも接続できる必要があります。**Any Data Center** を選択した場合は、ここでホストを選択することはできませんが、代わりに、個別のインポート操作時にホストを指定できます (**Virtual Machines** タブの **Import** 機能を使用)。 ● **Requires Authentication**: KVM ホストにアクセスするために認証が必要であるかどうかを指定できます。 ● **Username**: KVM ホストに接続するためのユーザー名 ● **Password**: 上記のユーザー名が認証されるパスワード。 **外部ネットワークプロバイダー** ● **Networking Plugin**: NIC 操作を処理するのにホストで使用されるドライバーの実装を決定します。**oVirt Network Provider for OVN** プラグインを備えた外部ネットワークプロバイダーがクラスターのデフォルトネットワークプロバイダーとして追加された場合は、これにより、クラスターに追加されたホストにインストールされるドライバーも決まります。 ● **Automatic Synchronization**: プロバイダーが既存のネットワークと自動的に同期されるかどうかを指定できます。 ● **Provider URL**: 外部ネットワークプロバイダーがホストされるマシンの URL または完全修飾ドメイン名。外部ネットワークプロバイダーのポート番号を URL または完全修飾ドメイン名の末尾に追加する必要があります。デフォルトでは、このポート番号は 9696 です。 ● **Read Only**: 管理ポータルから外部ネットワークプロバイダーを変更できるかどうかを指定します。 ● **Requires Authentication**: 外部ネットワークプロバイダーにアクセスするために認証が必要であるかどうかを指定できます。 ● **Username**: 外部ネットワークプロバイダーに接続するためのユーザー名。Active Directory で認証する場合は、ユーザー名の形式は、デフォルトの **username@domain**

設定	説明
	ではなく、**username@domain@auth_profile** の形式にする必要があります。 • **Password**: 上記のユーザー名が認証されるパスワード。 • **Protocol**: Keystone サーバーと通信するために使用するプロトコル。デフォルトは **HTTPS** です。 • **Hostname**: Keystone サーバーの IP アドレスまたはホスト名。 • **API port**: Keystone サーバーの API ポート番号。 • **API Version**: Keystone サーバーのバージョン。値は **v2.0** で、フィールドは無効になっています。 • **Tenant Name**: 任意。外部ネットワークプロバイダーがメンバーになっているテナントの名前。
Test	ユーザーが指定の認証情報をテストすることを許可します。このボタンは、すべてのプロバイダータイプで使用できます。

14.2.11. プロバイダーエージェントの設定設定に関する説明の追加

Add Provider ウィンドウの **Agent Configuration** タブを使用すると、ユーザーはネットワークプラグインの詳細を登録できます。このタブは **OpenStack Networking** プロバイダータイプでのみ使用できます。

表14.2 プロバイダーの追加: エージェント設定設定

設定	説明
インターフェイスマッピング	**label:interface** 形式のマッピングのコンマ区切りのリスト。
ブローカータイプ	OpenStack Networking インスタンスが使用するメッセージブローカータイプ。**RabbitMQ** または **Qpid** を選択します。
ホスト	メッセージブローカーがインストールされているマシンの URL または完全修飾ドメイン名。

設定	説明
ポート	上記のホストとの接続を確立するためのリモートポート。デフォルトでは、SSL がホストで有効でない場合は、このポートは 5762 に、SSL が有効な場合は 5761 になります。
Username	上記のメッセージブローカーを使用して OpenStack Networking インスタンスを認証するためのユーザー名。デフォルトでは、このユーザー名は **neutron** です。
Password	上記のユーザー名が認証されるパスワード。

14.3. 外部プロバイダーの編集

外部プロバイダーの編集

1. **Administration** → **Providers** をクリックし、編集する外部プロバイダーを選択します。
2. **Edit** をクリックします。
3. プロバイダーの現在の値を推奨値に変更します。
4. **OK** をクリックします。

14.4. 外部プロバイダーの削除

外部プロバイダーの削除

1. **Administration** → **Providers** をクリックし、削除する外部プロバイダーを選択します。
2. **Remove** をクリックします。
3. **OK** をクリックします。

パート III. 環境の管理

第15章 セルフホストエンジンの管理

15.1. セルフホストエンジンの保守

セルフホストエンジンのメンテナーンスモード

メンテナンスモードを使用すると、高可用性エージェントからの干渉を受けずに Manager 仮想マシンを起動、停止、および変更したり、Manager に干渉することなく環境内のセルフホスト型エンジンノードを再起動および変更したりできます。

適用できるメンテナーンスモードが 3 つあります。

- **global** - クラスター内のすべての高可用性エージェントは、Manager 仮想マシンの状態を監視できなくなります。グローバルメンテナンスモードは、Red Hat Virtualization の新しいバージョンへのアップグレードなど、**ovirt-engine** サービスの停止を必要とするセットアップまたはアップグレード操作に適用する必要があります。

- **local** - コマンドを発行しているノードの高可用性エージェントは、Manager 仮想マシンの状態を監視できません。ローカルメンテナンスモードでは、ノードは Manager 仮想マシンのホスティングを免除されます。このモードに設定されたときに Manager 仮想マシンをホストしている場合、使用可能なノードがあれば、Manager は別のノードに移行します。セルフホスト型エンジンノードにシステムの変更または更新を適用する場合は、ローカルメンテナンスモードをお勧めします。

- **none** - メンテナンスモードを無効にして、高可用性エージェントが動作していることを確認します。

ローカルメンテナーンスの設定

単一のセルフホストエンジンノードで高可用性エージェントを停止します。

管理ポータルからのローカルメンテナンスモードの設定

1. セルフホスト型エンジンノードをローカルメンテナンスモードにします。

   a. 管理ポータルで、**Compute → Hosts** をクリックし、セルフホスト型エンジンノードを選択します。

   b. **管理 → メンテナンス** をクリックします。そのノードに対してローカルメンテナンスモードが自動的にトリガーされます。

2. メンテナンスタスクを完了したら、メンテナンスモードを無効にします。

   a. 管理ポータルで **Compute → Hosts** をクリックし、セルフホスト型エンジンノードを選択します。

   b. **管理 → アクティブ化** をクリックします。

コマンドラインからローカルメンテナンスモードを設定

1. セルフホスト型エンジンノードにログインし、ローカルメンテナンスモードにします。

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

2. メンテナンスタスクを完了したら、メンテナンスモードを無効にします。

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

グローバルメンテナーンスの設定
クラスター内のすべてのセルフホストエンジンノードで高可用性エージェントを停止します。

管理ポータルからグローバルメンテナンスモードを設定

1. すべてのセルフホスト型エンジンノードをグローバルメンテナンスモードにします。
   a. 管理ポータルで **Compute → Hosts** をクリックし、セルフホスト型エンジンノードを選択します。
   b. **More Actions** (⋮) をクリックしてから、**Enable Global HA Maintenance** をクリックします。
2. メンテナンスタスクを完了したら、メンテナンスモードを無効にします。
   a. 管理ポータルで **Compute → Hosts** をクリックし、セルフホスト型エンジンノードを選択します。
   b. **More Actions** (⋮) をクリックしてから、**Disable Global HA Maintenance** をクリックします。

コマンドラインからグローバルメンテナンスモードの設定

1. セルフホスト型エンジンノードにログインし、グローバルメンテナンスモードにします。

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

2. メンテナンスタスクを完了したら、メンテナンスモードを無効にします。

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

15.2. MANAGER 仮想マシンの管理

hosted-engine ユーティリティーは、Manager 仮想マシンの管理に役立つ多くのコマンドを提供します。**hosted-engine** は、任意のセルフホスト型エンジンノードで実行できます。利用可能なコマンドをすべて表示するには、**hosted-engine --help** を実行します。特定のコマンドの詳細については、**hosted-engine --command --help** を実行してください。

15.2.1. セルフホスト型エンジン設定の更新

セルフホスト型エンジン設定を更新するには、**hosted-engine --set-shared-config** コマンドを使用します。このコマンドは、初期デプロイ後に共有ストレージドメインのセルフホスト型エンジン設定を更新します。

現在の設定値を表示するには、**hosted-engine --get-shared-config** コマンドを使用します。

利用可能なすべての設定キーの一覧とそれに対応するタイプを表示するには、以下のコマンドを入力します。

```
# hosted-engine --set-shared-config key --type=type --help
```

type は次のいずれかです。

he_local	ローカルホストの **etc/ovirt-hosted-engine/hosted-engine.conf** のローカルインスタンスに値を設定し、そのホストのみが新しい値を使用するようにします。新しい値を有効にするには、ovirt-ha-agent サービスおよび ovirt-ha-broker サービスを再起動します。
he_shared	共有ストレージの **etc/ovirt-hosted-engine/hosted-engine.conf** に値を設定するため、設定の変更後にデプロイされるすべてのホストがこれらの値を使用します。ホストで新しい値を有効にするには、そのホストを再デプロイします。
ha	ローカルストレージの **/var/lib/ovirt-hosted-engine-ha/ha.conf** に値を設定します。新しい設定はすぐに有効になります。
broker	ローカルストレージの **/var/lib/ovirt-hosted-engine-ha/broker.conf** に値を設定します。ovirt-ha-broker サービスを再起動して、新しい設定を有効にします。

15.2.2. メール通知の設定

セルフホスト型エンジンノードの HA 状態遷移に対して、SMTP を使用して電子メール通知を設定できます。更新できるキーには、**smtp-server**、**smtp-port**、**source-email**、**destination-emails**、および **state_transition** が含まれます。

電子メール通知の設定:

1. セルフホスト型エンジンノードで、**smtp-server** キーを目的の SMTP サーバーアドレスに設定します。

```
# hosted-engine --set-shared-config smtp-server smtp.example.com --type=broker
```

注記

セルフホスト型エンジン設定ファイルが更新されたことを確認するには、次のコマンドを実行します。

```
# hosted-engine --get-shared-config smtp-server --type=broker
broker : smtp.example.com, type : broker
```

2. デフォルトの SMTP ポート (ポート 25) が設定されていることを確認します。

```
# hosted-engine --get-shared-config smtp-port --type=broker
broker : 25, type : broker
```

3. SMTP サーバーが電子メール通知の送信に使用する電子メールアドレスを指定します。指定できるアドレスは 1 つだけです。

```
# hosted-engine --set-shared-config source-email source@example.com --type=broker
```

4. 電子メール通知を受け取る宛先電子メールアドレスを指定します。複数のメールアドレスを指定するには、各アドレスをコンマで区切ります。

```
# hosted-engine --set-shared-config destination-emails
destination1@example.com,destination2@example.com --type=broker
```

SMTP がセルフホスト型エンジン環境用に適切に設定されていることを確認するには、セルフホスト型エンジンノードの HA 状態を変更し、電子メール通知が送信されたかどうかを確認します。たとえば、HA エージェントをメンテナンスモードにすることで、HA の状態を変更できます。詳細は、「セルフホストエンジンの保守」 を参照してください。

15.3. 追加ホストでセルフホスト型エンジン用に予約されたメモリースロットの設定

Manager 用仮想マシンがシャットダウンするか、または移行する必要がある場合は、Manager 用仮想マシンが再起動または移行できるように、セルフホスト型エンジンノードに十分なメモリーが必要です。このメモリーは、スケジューリングポリシーを使用して、複数のセルフホスト型エンジンノードで予約できます。スケジューリングポリシーは、仮想マシンを起動または移行する前に、Manager 仮想マシンを起動するのに十分なメモリーが指定された数の追加のセルフホスト型エンジンノードに残っているかどうかを確認します。スケジューリングポリシーについての詳細は、**Administration Guide** の Creating a Scheduling Policy を参照してください。

Red Hat Virtualization Manager へ他のセルフホストエンジンノードを追加するには「Red Hat Virtualization Manager へのセルフホスト型エンジンノードの追加」 を参照してください。

追加ホストでセルフホスト型エンジン用に予約されたメモリースロットの設定

1. クラスターの **Compute** → **Clusters** をクリックして、セルフホスト型エンジンノードを含むクラスターを選択します。
2. **Edit** をクリックします。
3. **Scheduling Policy** タブをクリックします。
4. **+** をクリックして、**HeSparesCount** を選択します。
5. Manager 仮想マシンを起動するのに十分な空きメモリーを予約する追加のセルフホスト型エンジンノードの数を入力します。
6. **OK** をクリックします。

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

セルフホストエンジンノードは、通常のホストと同じ方法で追加することができますが、セルフホスト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することができます。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することもできますが、Manager 用仮想マシンをホストすることはできません。Red Hat は、Manager 仮想マシンの高可用性を確保するために、少なくとも 2 つのセルフホストエンジンノードを用意することをお勧めします。追加ホストは、REST API を使用して追加することもできます。**REST API Guide** の Hosts を参照してください。

前提条件

- セルフホスト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エンジン設定を削除してください。Removing a Host from a Self-Hosted Engine Environment を参照してください。

重要

静的 IPv6 アドレスを使用する管理ブリッジを作成する場合は、ホストを追加する前に、インターフェイス設定 (ifcfg) ファイルでネットワークマネージャーコントロールを無効にしてください。詳細は、https://access.redhat.com/solutions/3981311 を参照してください。

手順

1. 管理ポータルで **コンピュート → ホスト** 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、**Administration Guide** の Explanation of Settings and Controls in the New Host and Edit Host Windows 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には、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、**Administration Guide** の Host Power Management Settings Explained を参照してください。
7. **ホストエンジン** タブをクリックします。
8. **デプロイ** を選択します。
9. **OK** をクリックします。

15.5. 既存のホストのセルフホスト型エンジンノードとしての再インストール

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

手順

1. **コンピュート → ホスト** をクリックし、ホストを選択します。
2. **管理 → メンテナーンス** をクリックし、**OK** をクリックします。
3. **Installation → Reinstall** をクリックします。
4. **Hosted Engine** タブをクリックし、ドロップダウンリストから **DEPLOY** を選択します。
5. **OK** をクリックします。

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

15.6. MANAGER 仮想マシンをレスキューモードで起動

このトピックでは、Manager 仮想マシンが起動しないときにレスキューモードで起動する方法について説明します。詳細は、**Red Hat Enterprise Linux System 管理者ガイド**の レスキューモードでの起動 を参照してください。

1. ホストエンジンノードの 1 つに接続します。

```
$ ssh root@host_address
```

2. セルフホスト型エンジンをグローバルメンテナンスモードにします。

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

3. Manager 仮想マシンの実行中のインスタンスがすでに存在するかどうかを確認します。

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

Manager 仮想マシンインスタンスが実行されている場合は、そのホストに接続します。

```
# ssh root@host_address
```

4. 仮想マシンをシャットダウンします:

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

注記

仮想マシンがシャットダウンしない場合は、次のコマンドを実行します。

```
# hosted-engine --vm-poweroff
```

5. Manager 仮想マシンを一時停止モードで起動します。

```
hosted-engine --vm-start-paused
```

6. 一時的な VNC パスワードを設定します。

```
hosted-engine --add-console-password
```

このコマンドは、VNC を使用して Manger 仮想マシンにログインするために必要な情報を出力します。

7. VNC で Manager 用仮想マシンにログインします。Manager 仮想マシンはまだ一時停止しているため、フリーズしているように見えます。

8. ホストで次のコマンドを使用して、Manager 仮想マシンを再開します。

警告

次のコマンドを実行すると、ブートローダーメニューが表示されます。ブートローダーが通常のブートプロセスを続行する前に、レスキューモードに入る必要があります。このコマンドを続行する前に、レスキューモードに入ることに関する次の手順をお読みください。

```
# /usr/bin/virsh -c qemu:///system?authfile=/etc/ovirt-hosted-engine/virsh_auth.conf resume
HostedEngine
```

9. Manager 仮想マシンをレスキューモードで起動します。
10. グローバルメンテナンスモードを無効にする

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

これで、Manager 仮想マシンでレスキュータスクを実行できます。

15.7. セルフホスト型エンジン環境からのホストの削除

セルフホスト型エンジンノードを環境から削除するには、ノードをメンテナンスモードにし、ノードをアンデプロイし、オプションでノードを削除します。HA サービスが停止され、セルフホスト型エンジン設定ファイルが削除された後、ノードは通常のホストとして管理できます。

セルフホスト型エンジン環境からのホストの削除

1. 管理ポータルで **Compute → Hosts** をクリックし、セルフホスト型エンジンノードを選択します。
2. **管理 → メンテナーンス** をクリックし、**OK** をクリックします。
3. **Installation → Reinstall** をクリックします。
4. **Hosted Engine** タブをクリックし、ドロップダウンリストから **UNDEPLOY** を選択します。このアクションにより、**ovirt-ha-agent** および **ovirt-ha-broker** サービスが停止し、セルフホスト型エンジン設定ファイルが削除されます。
5. **OK** をクリックします。
6. 必要に応じて、**Remove** をクリックして、Remove Host(s) 確認ウィンドウを開き、**OK** をクリックします。

15.8. セルフホスト型エンジンの更新

セルフホストエンジンを現在お使いの 4.3 バージョンから最新の 4.3 バージョンに更新するには、環境をグローバルメンテナーンスモードに切り替え、続いてマイナーバージョン間の標準更新手順に従う必要があります。

グローバルメンテナーンスモードの有効化

Manager 用仮想マシンの設定またはアップグレード作業を実施する前に、セルフホスト型エンジン環境をグローバルメンテナンスモードに切り替える必要があります。

手順

1. セルフホスト型エンジンノードのいずれかにログインして、グローバルメンテナンスモードを有効にします。

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

2. 作業を進める前に、環境がメンテナーンスモードにあることを確認します。

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

クラスターがメンテナーンスモードにあることを示すメッセージが表示されるはずです。

Red Hat Virtualization Manager の更新

Red Hat Virtualization Manager の更新は、コンテンツ配信ネットワーク (CDN) 経由でリリースされます。

手順

1. Manager 用仮想マシンにログインします。
2. 更新されたパッケージが利用可能かどうかを確認します。

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

3. setup のパッケージを更新します。

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

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

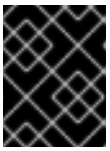
```
# engine-setup
```

スクリプトが正常に完了すると、以下のメッセージが表示されます。

```
Execution of setup completed successfully
```

注記

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



重要

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

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

```
# yum update
```

重要

カーネルパッケージが更新された場合は、以下を実行します。

1. グローバルメンテナンスモードを無効にする
2. マシンを再起動して更新を完了する

関連情報

グローバルメンテナーンスモードの無効化

グローバルメンテナーンスモードの無効化

手順

1. Manager 用仮想マシンにログインし、シャットダウンします。
2. セルフホスト型エンジンノードのいずれかにログインして、グローバルメンテナンスモードを無効にします。

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

グローバルメンテナンスモードを終了すると、ovirt-ha-agent が Manager 用仮想マシンを起動し、続いて Manager が自動的に起動します。Manager が起動するまでに最大で 10 分程度かかる場合があります。

3. 環境が動作していることを確認します。

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

情報の一覧に、**Engine status** が含まれます。**Engine status** の値は、以下のようになるはずです。

```
{"health": "good", "vm": "up", "detail": "Up"}
```

注記

仮想マシンが起動中で Manager がまだ動作していない場合、**Engine status** は以下のようになります。

```
{"reason": "bad vm status", "health": "bad", "vm": "up", "detail": "Powering up"}
```

このような場合には、数分間待ってからやり直してください。

15.9. セルフホスト型エンジンでの MANAGER の FQDN の変更

ovirt-engine-rename コマンドを使用して、Manager の完全修飾ドメイン名 (FQDN) のレコードを更新できます。

詳細は、「oVirt Engine Rename Tool を使って Manager の名前を変更」 を参照してください。

第16章 バックアップおよび移行

16.1. RED HAT VIRTUALIZATION MANAGER のバックアップおよび復元

16.1.1. Red Hat Virtualization Manager のバックアップ - 概要

engine-backup ツールを使用して、Red Hat Virtualization Manager の定期的なバックアップを作成します。このツールは、エンジンデータベースおよび設定ファイルを 1 つのファイルにバックアップし、**ovirt-engine** サービスを中断することなく実行できます。

16.1.2. engine-backup コマンドの構文

engine-backup コマンドは、次の 2 つの基本モードのいずれかで機能します。

```
# engine-backup --mode=backup
```

```
# engine-backup --mode=restore
```

これらの 2 つのモードは、バックアップの範囲とエンジンデータベースのさまざまな認証情報を指定できる一連のパラメーターによってさらに拡張されます。パラメーターとその機能の完全なリストについては、**engine-backup --help** を実行してください。

Basic Options

--mode

コマンドがバックアップ操作を実行するか、復元操作を実行するかを指定します。**backup** と **restore** の 2 つのオプションを使用できます。これは必須パラメーターです。

--file

バックアップモードでバックアップを取得するファイルのパスおよび名前、ならびに復元モードでバックアップデータを読み取るファイルのパスおよび名前を指定します。これは、バックアップモードと復元モードの両方で必須のパラメーターです。

--log

バックアップまたは復元操作のログが書き込まれるファイルのパスおよび名前を指定します。このパラメーターは、バックアップモードと復元モードの両方で必要です。

--scope

バックアップまたは復元操作の範囲を指定します。4 つのオプションがあります。**all** は、すべてのデータベースおよび設定データをバックアップまたは復元します。**files** は、システム上のファイルのみをバックアップまたは復元します。**db** は、Manager データベースのみをバックアップまたは復元します。**dwhdb** は、Data Warehouse データベースのみをバックアップまたは復元します。デフォルトのスコープは **all** です。
--scope パラメーターは、同じ **engine-backup** コマンドで複数回指定できます。

Manager Database Options

次のオプションは、**restore** モードで **engine-backup** コマンドを使用する場合にのみ使用できます。以下のオプション構文は、Manager データベースの復元に適用されます。Data Warehouse データベースを復元するための同じオプションがあります。Data Warehouse オプションの構文は、**engine-backup --help** を参照してください。

--provision-db

復元先の Manager データベースバックアップ用の PostgreSQL データベースを作成します。これは、PostgreSQL データベースがまだ設定されていないリモートホストまたは新規インストールでバックアップを復元する場合に必要なパラメーターです。

--change-db-credentials

バックアップ自体に保存されている認証情報以外の認証情報を使用して、Manager データベースを復元するための代替認証情報を指定できます。このパラメーターで必要な追加パラメーターは、**engine-backup --help** を参照してください。

--restore-permissions または --no-restore-permissions

データベースユーザーの権限を復元します (または復元しません)。バックアップを復元するときは、これらのパラメーターの 1 つが必要です。

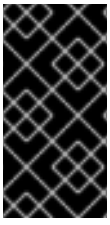
注記

バックアップに追加のデータベースユーザーの許可が含まれている場合、**--restore-permissions** および **--provision-db** (または **--provision-dwh-db**) オプションを使用してバックアップを復元すると、ランダムなパスワードを持つ追加のユーザーが作成されます。追加のユーザーが復元したシステムにアクセスする必要がある場合は、これらのパスワードを手動で変更する必要があります。https://access.redhat.com/articles/2686731 を参照してください。

16.1.3. engine-backup コマンドを使用したバックアップの作成

Red Hat Virtualization Manager は、Manager がアクティブな場合に **engine-backup** コマンドを使用してバックアップできます。以下のオプションのいずれかを **--scope** に追加し、実行するバックアップを指定します。

- **all**: Manager 上のすべてのデータベースおよび設定ファイルの完全バックアップ
- **files**: システム上のファイルのみのバックアップ
- **db**: Manager データベースのみのバックアップ
- **dwhdb**: データウェアハウスデータベースのみのバックアップ



重要

データベースを Red Hat Virtualization Manager の新規インストールに復元するには、データベースのバックアップだけでは不十分です。Manager では設定ファイルへのアクセスも必要です。デフォルト以外のスコープを指定するバックアップ。**all** は **files** スコープまたはファイルシステムのバックアップを一緒に指定する必要があります。

engine-backup コマンドの使用例

1. Red Hat Virtualization Manager を実行しているマシンにログインします。
2. バックアップを作成します。

例16.1 完全バックアップの作成

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

例16.2 Manager データベースのバックアップの作成

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

db オプションは、Data Warehouse データベースをバックアップする **dwhdb** に置き換えます。

バックアップを含む **tar** ファイルは、指定したパスとファイル名を使用して作成されます。

バックアップを含む **tar** ファイルを使用して、環境を復元できるようになりました。

16.1.4. engine-backup コマンドを使用したバックアップの復元

engine-backup コマンドを使用してバックアップを復元するには、復元先によっては、バックアップを作成するよりも多くの手順が必要です。たとえば、**engine-backup** コマンドを使用して、Red Hat Virtualization の既存のインストールに加えて、ローカルまたはリモートのデータベースを使用して、Red Hat Virtualization の新規インストールにバックアップを復元できます。

重要

バックアップは、バックアップと同じメジャーリリースの環境にのみ復元できます。たとえば、Red Hat Virtualization バージョン 4.2 環境のバックアップは、別の Red Hat Virtualization バージョン 4.2 環境にのみ復元できます。バックアップファイルに含まれている Red Hat Virtualization のバージョンを表示するには、バックアップファイルを解凍し、解凍されたファイルのルートディレクトリーにある **version** ファイルの値を読み取ります。

16.1.5. バックアップを新規インストールに復元する

engine-backup コマンドを使用して、Red Hat Virtualization Manager の新規インストールにバックアップを復元できます。以下の手順は、ベースオペレーティングシステムがインストールされ、Red Hat Virtualization Manager に必要なパッケージがインストールされているが、**engine-setup** コマンドがまだ実行されていないマシンで実行する必要があります。この手順は、バックアップを復元するマシンから 1 つまたは複数のバックアップファイルにアクセスできることを前提としています。

バックアップを新規インストールに復元する

1. Manager マシンにログインします。エンジンデータベースをリモートホストに復元する場合は、そのホストにログオンして、関連するアクションを実行する必要があります。同様に、Data Warehouse をリモートホストにも復元する場合は、そのホストにログオンして、関連するアクションを実行する必要があります。

2. 完全バックアップまたはデータベースのみのバックアップを復元します。

   - 完全バックアップを復元します。

     ```
     # engine-backup --mode=restore --file=file_name --log=log_file_name --provision-db --
     restore-permissions
     ```

     完全バックアップの一部として Data Warehouse も復元される場合は、追加のデータベースをプロビジョニングします。

```
engine-backup --mode=restore --file=file_name --log=log_file_name --provision-db --
provision-dwh-db --restore-permissions
```

- 設定ファイルとデータベースバックアップを復元して、データベースのみのバックアップを復元します。

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

上記の例では、Manager データベースのバックアップを復元します。

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

上記の例では、Data Warehouse データベースのバックアップを復元します。

成功すると、次の出力が表示されます。

```
You should now run engine-setup.
Done.
```

3. 次のコマンドを実行し、プロンプトに従って復元された Manager を設定します。

```
# engine-setup
```

Red Hat Virtualization Manager は、バックアップに保存されているバージョンに復元されました。新しい Red Hat Virtualization システムの完全修飾ドメイン名を変更するには、「oVirt エンジンの名前変更ツール」 を参照してください。

16.1.6. バックアップを復元して既存のインストールを上書き

engine-backup コマンドを使用すると、Red Hat Virtualization Manager がすでにインストールおよび設定されているマシンにバックアップを復元できます。これは、環境のバックアップを取り、その環境で変更を実行し、バックアップから環境を復元して変更を元に戻したい場合に役立ちます。

ホストの追加や削除など、バックアップの作成後に環境に加えられた変更は、復元された環境には表示されません。これらの変更をやり直す必要があります。

手順

1. Manager マシンにログインします。
2. 設定ファイルを削除し、Manager に関連付けられているデータベースをクリーンアップします。

```
# engine-cleanup
```

engine-cleanup コマンドは、Manager データベースのみを削除します。データベースを削除したり、そのデータベースを所有しているユーザーを削除したりすることはありません。

3. フルバックアップまたはデータベースのみのバックアップを復元します。ユーザーとデータベースがすでに存在しているので、新規のデータベースを作成したり、データベースの認証情報を指定する必要はありません。

- 完全バックアップを復元します。

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

- 設定ファイルおよびデータベースバックアップを復元して、データベースのみのバックアップを復元します。

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

注記

Manager データベースのみを復元するには (たとえば、Data Warehouse データベースが別のマシンにある場合)、**-scope=dwhdb** パラメーターを省略できます。

成功すると、次の出力が表示されます。

```
You should now run engine-setup.
Done.
```

4. Manager を再設定します。

```
# engine-setup
```

16.1.7. 異なる認証情報を使用したバックアップの復元

engine-backup コマンドは、Red Hat Virtualization Manager がすでにインストールされセットアップされているマシンにバックアップを復元することができますが、バックアップ内のデータベースの認証情報は、バックアップを復元するマシン上のデータベースの認証情報とは異なっています。これは、インストールのバックアップを取り、バックアップから別のシステムにインストールを復元する場合に役立ちます。

重要

バックアップを復元して既存のインストールを上書きする場合は、**engine-backup** コマンドを使用する前に、**engine-cleanup** コマンドを実行して既存のインストールをクリーンアップする必要があります。**engine-cleanup** コマンドは、エンジンデータベースをクリーンアップするだけで、データベースを削除したり、そのデータベースを所有しているユーザーを削除したりすることはありません。したがって、新規のデータベースを作成したり、データベースの認証情報を指定する必要はありません。ただし、エンジンデータベースの所有者の認証情報がわからない場合は、バックアップを復元する前に認証情報を変更する必要があります。

異なる認証情報を使用したバックアップの復元

1. Red Hat Virtualization Manager マシンにログインします。
2. 次のコマンドを実行し、プロンプトに従って Manager の設定ファイルを削除し、Manager のデータベースをクリーンアップします。

```
# engine-cleanup
```

3. **engine** データベースの所有者のパスワードがわからない場合は、そのユーザーのパスワードを変更します。

   a. postgresql コマンドラインを入力します。

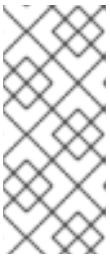
   ```
   # su - postgres -c 'scl enable rh-postgresql10 -- psql'
   ```

   b. **engine** データベースを所有するユーザーのパスワードを変更します。

   ```
   postgres=# alter role user_name encrypted password 'new_password';
   ```

   必要に応じて、**ovirt_engine_history** データベースを所有するユーザーに対してこれを繰り返します。

4. **--change-db-credentials** パラメーターを使用して完全バックアップまたはデータベースのみのバックアップを復元し、新しいデータベースの認証情報を渡します。Manager にローカルなデータベースの database_location は **localhost** です。



注記

次の例では、パスワードを指定せずにデータベースごとに **--*password** オプションを使用します。これにより、データベースごとにパスワードの入力を求められます。または、データベースごとに **--*passfile=password_file** オプションを使用して、対話型プロンプトを必要とせずに、パスワードを **engine-backup** ツールに安全に渡すことができます。

- 完全バックアップを復元します。

```
# engine-backup --mode=restore --file=file_name --log=log_file_name --change-db-credentials --db-host=database_location --db-name=database_name --db-user=engine --db-password --no-restore-permissions
```

Data Warehouse も完全バックアップの一部として復元される場合は、追加のデータベースの改訂された認証情報を含めます。

```
engine-backup --mode=restore --file=file_name --log=log_file_name --change-db-credentials --db-host=database_location --db-name=database_name --db-user=engine --db-password --change-dwh-db-credentials --dwh-db-host=database_location --dwh-db-name=database_name --dwh-db-user=ovirt_engine_history --dwh-db-password --no-restore-permissions
```

- 設定ファイルおよびデータベースバックアップを復元して、データベースのみのバックアップを復元します。

```
# engine-backup --mode=restore --scope=files --scope=db --file=file_name --log=log_file_name --change-db-credentials --db-host=database_location --db-name=database_name --db-user=engine --db-password --no-restore-permissions
```

上記の例では、Manager データベースのバックアップを復元します。

```
# engine-backup --mode=restore --scope=files --scope=dwhdb --file=file_name --
```

```
log=log_file_name --change-dwh-db-credentials --dwh-db-host=database_location --
dwh-db-name=database_name --dwh-db-user=ovirt_engine_history --dwh-db-password
--no-restore-permissions
```

上記の例では、Data Warehouse データベースのバックアップを復元します。

成功すると、次の出力が表示されます。

```
You should now run engine-setup.
Done.
```

5. 次のコマンドを実行し、プロンプトに従ってファイアウォールを再設定し、**ovirt-engine** サービスが正しく設定されていることを確認します。

```
# engine-setup
```

16.1.8. セルフホスト型エンジンのバックアップおよび復元

セルフホスト型エンジンをバックアップして、新しいセルフホスト環境に復元できます。この手順は、環境を別のストレージタイプの新しいセルフホスト型エンジンストレージドメインに移行するなどのタスクに使用します。

デプロイメント中にバックアップファイルを指定すると、バックアップは、新しいセルフホスト型エンジンストレージドメインを使用して、新しい Manager 仮想マシンに復元されます。古い Manager が削除され、古いセルフホスト型エンジンストレージドメインの名前が変更され、新しい環境が正しく機能していることを確認した後、手動で削除できます。新規ホストにデプロイすることを強く推奨します。デプロイメント用のホストがバックアップ環境に存在している場合は、新しい環境で競合を回避するために、復元されたデータベースから削除されます。

バックアップと復元の操作には、次の主要なアクションが含まれます。

1. **engine-backup** ツールを使用して元の Manager をバックアップします。
2. 新しいセルフホスト型エンジンをデプロイし、バックアップを復元します。
3. 新しい Manager 用仮想マシンで Manager のリポジトリーを有効にします。
4. セルフホスト型エンジンノードを再インストールして、設定を更新します。
5. 古いセルフホスト型エンジンストレージドメインを削除します。

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

前提条件

- Manager とホスト用に用意された完全修飾ドメイン名。正引き (フォワードルックアップ) と逆引き (リバースルックアップ) の記録は両方とも DNS で設定する必要があります。新しい Manager は、元の Manager と同じ完全修飾ドメイン名を持っている必要があります。
- 元の Manager を最新のマイナーバージョンに更新する必要があります。バックアップファイルの Manager バージョンは、新しい Manager のバージョンと一致する必要があります。**Upgrade Guide** の Updating the Red Hat Virtualization Manager を参照してください。
- 環境内に少なくとも 1 つの通常のホストが存在する必要があります。このホスト (およびその他

の通常のホスト) は、SPM ロールおよび実行中の仮想マシンをホストするためにアクティブなままになります。通常のホストがまだ SPM でない場合は、バックアップを作成する前に、通常のホストを選択し、**Management → Select as SPM** をクリックして、SPM のロールを移動します。
通常のホストが利用できない場合、1 つを追加する方法は 2 つあります。

- セルフホスト型エンジン設定をノードから削除します (ただし、ノードを環境から削除しないでください)。「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。
- 新しい通常のホストを追加します。「Red Hat Virtualization Manager への通常のホストの追加」 を参照してください。

16.1.8.1. 元の Manager のバックアップ

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

engine-backup --mode=backup オプションの詳細は、**Administration Guide** の Backing Up and Restoring the Red Hat Virtualization Manager を参照してください。

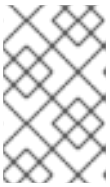
手順

1. セルフホスト型エンジンノードの 1 つにログインし、環境をグローバルメンテナンスモードに移行します。

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

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

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



注記

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

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

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

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

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

5. 他の目的で Manager マシンが必要ない場合は、Red Hat Subscription Manager から登録を解除します。

```
# subscription-manager unregister
```

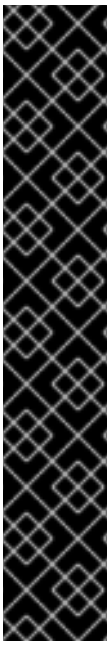
6. セルフホスト型エンジンノードの 1 つにログインし、元の Manager 仮想マシンをシャットダウンします。

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

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

16.1.8.2. 新しいセルフホスト型エンジンでのバックアップの復元

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



重要

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

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

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

手順

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

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

2. 新しいホストにログインします。Red Hat Virtualization ホストにデプロイする場合、デフォルトでセルフホストエンジンデプロイメントツールを使用できます。Red Hat Enterprise Linux にデプロイする場合は、以下のパッケージをインストールする必要があります。

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

3. Red Hat は、ネットワークまたはターミナルが中断した場合にセッションが失われないように、**screen** ウィンドウマネージャーを使用してスクリプトを実行することをお勧めします。**screen** をインストールして実行します。

```
# yum install screen
# screen
```

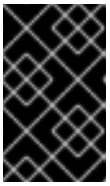
セッションのタイムアウトまたは接続の中断が発生した場合は、**screen-d-r** を実行してデプロイメントセッションを復元します。

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

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

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

5. **Yes** を選択してデプロイメントを開始します。
6. ネットワークを設定します。スクリプトにより、環境の管理ブリッジとして使用する NIC 候補が検出されます。
7. 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。
8. Manager 仮想マシンの完全修飾ドメイン名 (FQDN) を指定します。
9. Manager の root パスワードを入力します。
10. root ユーザーとして Manager にログインできる SSH 公開鍵を入力し、root ユーザーの SSH アクセスを有効にするかどうかを指定します。
11. 仮想マシンの CPU およびメモリー設定を入力します。
12. Manager 用仮想マシンの MAC アドレスを入力するか、無作為に生成される MAC アドレスを適用します。Manager 用仮想マシンへの IP アドレス割り当てに DHCP を使用するには、この MAC アドレスに有効な DHCP 予約があることを確認してください。デプロイメントスクリプトは、DHCP サーバーの設定は行いません。
13. 仮想マシンのネットワーク情報を入力します。**Static** を指定する場合には、Manager の IP アドレスを入力します。



重要

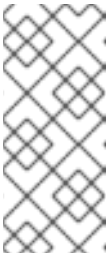
静的 IP アドレスは、ホストと同じサブネットに属している必要があります。たとえばホストが 10.1.1.0/24 内にある場合、Manager 用仮想マシンの IP は同じサブネット範囲 (10.1.1.1-254/24) になければなりません。

14. Manager 用仮想マシンおよびベースホストのエントリーを仮想マシンの **/etc/hosts** ファイルに追加するかどうかを指定します。ホスト名は解決可能でなければなりません。
15. SMTP サーバーの名前と TCP ポート番号、メール通知を送信するメールアドレス、メール通知を受信するメールアドレス (複数ある場合はコンマ区切りリスト) を指定します。
16. 管理ポータルにアクセスするための **admin@internal** ユーザーのパスワードを入力します。スクリプトにより仮想マシンが作成されます。RHV-M Appliance をインストールする必要がある場合は、時間がかかる場合があります。
17. 使用するストレージのタイプを選択します。
    - NFS の場合は、バージョン、完全なアドレス、およびストレージへのパスならびにマウントオプションを入力します。

警告

仮想マシンのデータが失われるリスクがあるため、古いセルフホスト型エンジンストレージドメインのマウントポイントを新しいストレージドメインに使用しないでください。

- 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することができます。



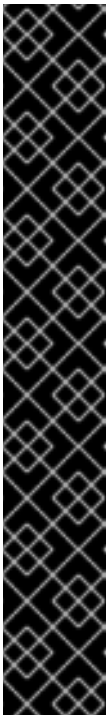
注記

複数の iSCSI ターゲットを指定するには、セルフホスト型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 Multipath** を参照してください。Multipath Helper ツールを使用して、さまざまなオプションでマルチパスをインストールおよび設定するスクリプトを生成することもできます。

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

警告

仮想マシンのデータが失われるリスクがあるため、古いセルフホスト型エンジンストレージドメインのマウントポイントを新しいストレージドメインに使用しないでください。



重要

レプリカ 3 Gluster ストレージのみがサポートされています。次の設定になっていることを確認してください。

- 3 つの Gluster サーバーすべての**/etc/glusterfs/glusterd.vol** ファイルで、**rpc-auth-allow-insecure** を **on** に設定します。

```
option rpc-auth-allow-insecure on
```

- 次のようにボリュームを設定します。

```
gluster volume set _volume_ cluster.quorum-type auto
gluster volume set _volume_ network.ping-timeout 10
gluster volume set _volume_ auth.allow \*
gluster volume set _volume_ group virt
gluster volume set _volume_ storage.owner-uid 36
gluster volume set _volume_ storage.owner-gid 36
gluster volume set _volume_ server.allow-insecure on
```

- 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、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、**Administration Guide** の Reusing LUNs を参照してください。

18. Manager のディスクサイズを入力します。
スクリプトはデプロイメントが完了するまで続行されます。

19. デプロイメントプロセスでは Manager の SSH キーが変更されます。クライアントマシンが SSH エラーなしで新規の Manager にアクセスできるようにするには、元の Manager にアクセスするクライアントマシンの **.ssh/known_hosts** ファイルから元の Manager のエントリーを削除します。

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

16.1.8.3. Red Hat Virtualization Manager リポジトリーの有効化

システムを Red Hat Subscription Manager に登録し、**Red Hat Virtualization Manager** のサブスクリプションをアタッチして Manager のリポジトリーを有効にします。

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。

```
# yum repolist
```

4. リポジトリーを設定します。

```
# subscription-manager repos \
    --disable='*' \
    --enable=rhel-7-server-rpms \
    --enable=rhel-7-server-supplementary-rpms \
    --enable=rhel-7-server-rhv-4.3-manager-rpms \
    --enable=rhel-7-server-rhv-4-manager-tools-rpms \
    --enable=rhel-7-server-ansible-2.9-rpms \
    --enable=jb-eap-7.2-for-rhel-7-server-rpms
```

Manager とそのリソースは、新しいセルフホスト環境で実行されています。セルフホスト型エンジンノードは、セルフホスト型エンジン設定を更新するために Manager に再インストールする必要があります。標準ホストは影響を受けません。セルフホスト型エンジンノードごとに次の手順を実行します。

16.1.8.4. ホストの再インストール

管理ポータルから Red Hat Virtualization Host (RHVH) および Red Hat Enterprise Linux ホストを再インストールします。この手順には、ホストの停止および再起動が含まれます。

前提条件

- 移行がクラスターレベルで有効にされる場合に、仮想マシンはクラスター内の別のホストに自動的に移行されので、ホストの使用が比較的低くなる場合にはホストを再インストールことをおすすめします。
- ホストによるメンテナーンス実行に十分なメモリーがクラスターに予約されていることを確認します。クラスターに十分なメモリーがない場合には、仮想マシンの移行操作がハングしてから失敗します。一部またはすべての仮想マシンをシャットダウンしてから、ホストをメンテナーンスに移行すると、この操作のメモリー使用量を減らすことができます。

- 再インストールを実行する前に、クラスターに複数のホストが含まれていることを確認してください。Storage Pool Manager (SPM) のタスクを実行するには、1台のホストは使用可能な状態でなければならないので、すべてのホストを同時に再インストールしないでください。

手順

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **管理 → メンテナンス** をクリックします。
3. **Installation → Reinstall** をクリックして、**Install Host** ウィンドウを開きます。
4. **Hosted Engine** タブをクリックし、ドロップダウンリストから **DEPLOY** を選択します。
5. **OK** をクリックして、ホストを再インストールします。

正常に再インストールされると、ホストのステータスが **Up** と表示されます。ホストから移行された仮想マシンはすべて、ホストに移行できるようになりました。

重要

Red Hat Virtualization Host が Red Hat Virtualization Manager に正常に登録されてから再インストールされると、ステータスが **Install Failed** の状態で、管理ポータルに誤って表示される可能性があります。**Management → アクティブ化** をクリックすると、ホストが **Up** ステータスに変わり、使用できるようになります。

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

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

復元中に、古いセルフホスト型エンジンのストレージドメインの名前が変更されましたが、復元に問題があった場合に備えて、新しい環境から削除されませんでした。環境が正常に実行されていることを確認したら、古いセルフホスト型エンジンストレージドメインを削除できます。

16.1.8.5. ストレージドメインの削除

データセンターに、仮想化環境から削除するストレージドメインがあります。

手順

1. **Storage → Domains** をクリックします。
2. ストレージドメインをメンテナンスモードに移動し、デタッチします。
   a. ストレージドメインの名前をクリックし、詳細ビューを開きます。
   b. **Data Center** タブをクリックします。
   c. **Maintenance** をクリックしてから **OK** をクリックします。
   d. **Detach** をクリックしてから **OK** をクリックします。
3. **Remove** をクリックします。

4. 任意で **Format Domain, i.e. Storage Content will be lost!**チェックボックスを選択して、ドメインのコンテンツを消去します。

5. **OK** をクリックします。

ストレージドメインは環境から完全に削除されます。

16.1.9. 既存のバックアップからのセルフホスト型エンジンの復元

修復できない問題のためにセルフホスト型エンジンが使用できない場合は、問題が発生する前に作成したバックアップを使用して、新しいセルフホスト環境でエンジンを復元できます (使用可能な場合)。

デプロイメント中にバックアップファイルを指定すると、バックアップは、新しいセルフホスト型エンジンストレージドメインを使用して、新しい Manager 仮想マシンに復元されます。古い Manager が削除され、古いセルフホスト型エンジンストレージドメインの名前が変更され、新しい環境が正しく機能していることを確認した後、手動で削除できます。新規ホストにデプロイすることを強く推奨します。デプロイメント用のホストがバックアップ環境に存在している場合は、新しい環境で競合を回避するために、復元されたデータベースから削除されます。

セルフホスト型エンジンの復元には、次の主要なアクションが含まれます。

1. 新しいセルフホスト型エンジンをデプロイし、バックアップを復元します。
2. 新しい Manager 用仮想マシンで Manager のリポジトリーを有効にします。
3. セルフホスト型エンジンノードを再インストールして、設定を更新します。
4. 古いセルフホスト型エンジンストレージドメインを削除します。

この手順は、元の Manager にアクセスできず、新しいホストがバックアップファイルにアクセスできることを前提としています。

前提条件

- Manager とホスト用に用意された完全修飾ドメイン名。正引き (フォワードルックアップ) と逆引き (リバースルックアップ) の記録は両方とも DNS で設定する必要があります。新しい Manager は、元の Manager と同じ完全修飾ドメイン名を持っている必要があります。

16.1.9.1. 新しいセルフホスト型エンジンでのバックアップの復元

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

重要

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

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

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

手順

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

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

2. 新しいホストにログインします。Red Hat Virtualization ホストにデプロイする場合、デフォルトでセルフホストエンジンデプロイメントツールを使用できます。Red Hat Enterprise Linux にデプロイする場合は、以下のパッケージをインストールする必要があります。

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

3. Red Hat は、ネットワークまたはターミナルが中断した場合にセッションが失われないように、**screen** ウィンドウマネージャーを使用してスクリプトを実行することをお勧めします。**screen** をインストールして実行します。

```
# yum install screen
# screen
```

セッションのタイムアウトまたは接続の中断が発生した場合は、**screen-d-r** を実行してデプロイメントセッションを復元します。

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

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

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

5. **Yes** を選択してデプロイメントを開始します。

6. ネットワークを設定します。スクリプトにより、環境の管理ブリッジとして使用する NIC 候補が検出されます。

7. 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。

8. Manager 仮想マシンの完全修飾ドメイン名 (FQDN) を指定します。

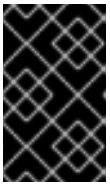
9. Manager の root パスワードを入力します。

10. root ユーザーとして Manager にログインできる SSH 公開鍵を入力し、root ユーザーの SSH アクセスを有効にするかどうかを指定します。

11. 仮想マシンの CPU およびメモリー設定を入力します。

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

13. 仮想マシンのネットワーク情報を入力します。**Static** を指定する場合には、Manager の IP アドレスを入力します。



重要

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

14. Manager 用仮想マシンおよびベースホストのエントリーを仮想マシンの **/etc/hosts** ファイルに追加するかどうかを指定します。ホスト名は解決可能でなければなりません。

15. SMTP サーバーの名前と TCP ポート番号、メール通知を送信するメールアドレス、メール通知を受信するメールアドレス (複数ある場合はコンマ区切りリスト) を指定します。

16. 管理ポータルにアクセスするための **admin@internal** ユーザーのパスワードを入力します。スクリプトにより仮想マシンが作成されます。RHV-M Appliance をインストールする必要がある場合は、時間がかかる場合があります。

17. 使用するストレージのタイプを選択します。

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

警告

仮想マシンのデータが失われるリスクがあるため、古いセルフホスト型エンジンストレージドメインのマウントポイントを新しいストレージドメインに使用しないでください。

    - 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す

ることができます。

注記

複数の iSCSI ターゲットを指定するには、セルフホスト型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 Multipath** を参照してください。Multipath Helper ツールを使用して、さまざまなオプションでマルチパスをインストールおよび設定するスクリプトを生成することもできます。

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

警告

仮想マシンのデータが失われるリスクがあるため、古いセルフホスト型エンジンストレージドメインのマウントポイントを新しいストレージドメインに使用しないでください。

重要

レプリカ 3 Gluster ストレージのみがサポートされています。次の設定になっていることを確認してください。

- 3 つの Gluster サーバーすべての**/etc/glusterfs/glusterd.vol** ファイルで、**rpc-auth-allow-insecure** を **on** に設定します。

```
option rpc-auth-allow-insecure on
```

- 次のようにボリュームを設定します。

```
gluster volume set _volume_ cluster.quorum-type auto
gluster volume set _volume_ network.ping-timeout 10
gluster volume set _volume_ auth.allow \*
gluster volume set _volume_ group virt
gluster volume set _volume_ storage.owner-uid 36
gluster volume set _volume_ storage.owner-gid 36
gluster volume set _volume_ server.allow-insecure on
```

- 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、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、**Administration Guide** の Reusing LUNs を参照してください。

18. Manager のディスクサイズを入力します。
スクリプトはデプロイメントが完了するまで続行されます。

19. デプロイメントプロセスでは Manager の SSH キーが変更されます。クライアントマシンが SSH エラーなしで新規の Manager にアクセスできるようにするには、元の Manager にアクセ

スするクライアントマシンの **.ssh/known_hosts** ファイルから元の Manager のエントリーを削除します。

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

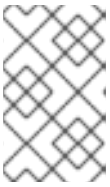
16.1.9.2. Red Hat Virtualization Manager リポジトリーの有効化

システムを Red Hat Subscription Manager に登録し、**Red Hat Virtualization Manager** のサブスクリプションをアタッチして Manager のリポジトリーを有効にします。

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

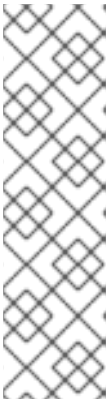
IPv6 ネットワークを使用している場合は、IPv6 移行メカニズムを使用して、コンテンツ配信ネットワークおよびサブスクリプションマネージャーにアクセスします。

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。

```
# yum repolist
```

4. リポジトリーを設定します。

```
# subscription-manager repos \
    --disable='*' \
    --enable=rhel-7-server-rpms \
    --enable=rhel-7-server-supplementary-rpms \
    --enable=rhel-7-server-rhv-4.3-manager-rpms \
```

```
--enable=rhel-7-server-rhv-4-manager-tools-rpms \
--enable=rhel-7-server-ansible-2.9-rpms \
--enable=jb-eap-7.2-for-rhel-7-server-rpms
```

Manager とそのリソースは、新しいセルフホスト環境で実行されています。セルフホスト型エンジンノードは、セルフホスト型エンジン設定を更新するために Manager に再インストールする必要があります。標準ホストは影響を受けません。セルフホスト型エンジンノードごとに次の手順を実行します。

16.1.9.3. ホストの再インストール

管理ポータルから Red Hat Virtualization Host (RHVH) および Red Hat Enterprise Linux ホストを再インストールします。この手順には、ホストの停止および再起動が含まれます。

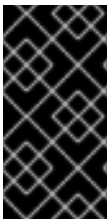
前提条件

- 移行がクラスターレベルで有効にされる場合に、仮想マシンはクラスター内の別のホストに自動的に移行されので、ホストの使用が比較的低くなる場合にはホストを再インストールことをおすすめします。
- ホストによるメンテナーンス実行に十分なメモリーがクラスターに予約されていることを確認します。クラスターに十分なメモリーがない場合には、仮想マシンの移行操作がハングしてから失敗します。一部またはすべての仮想マシンをシャットダウンしてから、ホストをメンテナーンスに移行すると、この操作のメモリー使用量を減らすことができます。
- 再インストールを実行する前に、クラスターに複数のホストが含まれていることを確認してください。Storage Pool Manager (SPM) のタスクを実行するには、1 台のホストは使用可能な状態でなければならないので、すべてのホストを同時に再インストールしないでください。

手順

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **管理 → メンテナンス** をクリックします。
3. **Installation → Reinstall** をクリックして、**Install Host** ウィンドウを開きます。
4. **Hosted Engine** タブをクリックし、ドロップダウンリストから **DEPLOY** を選択します。
5. **OK** をクリックして、ホストを再インストールします。

正常に再インストールされると、ホストのステータスが **Up** と表示されます。ホストから移行された仮想マシンはすべて、ホストに移行できるようになりました。



重要

Red Hat Virtualization Host が Red Hat Virtualization Manager に正常に登録されてから再インストールされると、ステータスが **Install Failed** の状態で、管理ポータルに誤って表示される可能性があります。**Management → アクティブ化** をクリックすると、ホストが **Up** ステータスに変わり、使用できるようになります。

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

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

復元中に、古いセルフホスト型エンジンのストレージドメインの名前が変更されましたが、復元に問題があった場合に備えて、新しい環境から削除されませんでした。環境が正常に実行されていることを確認したら、古いセルフホスト型エンジンストレージドメインを削除できます。

16.1.9.4. ストレージドメインの削除

データセンターに、仮想化環境から削除するストレージドメインがあります。

手順

1. **Storage** → **Domains** をクリックします。
2. ストレージドメインをメンテナンスモードに移動し、デタッチします。
   a. ストレージドメインの名前をクリックし、詳細ビューを開きます。
   b. **Data Center** タブをクリックします。
   c. **Maintenance** をクリックしてから **OK** をクリックします。
   d. **Detach** をクリックしてから **OK** をクリックします。
3. **Remove** をクリックします。
4. 任意で **Format Domain, i.e. Storage Content will be lost!**チェックボックスを選択して、ドメインのコンテンツを消去します。
5. **OK** をクリックします。

ストレージドメインは環境から完全に削除されます。

16.1.10. 既存のバックアップからのセルフホスト型エンジンの上書き

セルフホスト型エンジンにアクセスできるが、データベースの破損や設定エラーなどロールバックが困難な問題が発生した場合は、問題が発生する前に取ったバックアップがあれば、それを使用して以前の状態に環境を復元することができます。

セルフホスト型エンジンの以前の状態を復元するには、次の手順が必要です。

1. 環境をグローバルメンテナンスモードに切り替える
2. Manager 仮想マシンでバックアップを復元します。
3. グローバルメンテナーンスモードを無効にする

engine-backup --mode=restore オプションの詳細は、「Red Hat Virtualization Manager のバックアップおよび復元」 を参照してください。

16.1.10.1. グローバルメンテナーンスモードの有効化

Manager 用仮想マシンの設定またはアップグレード作業を実施する前に、セルフホスト型エンジン環境をグローバルメンテナンスモードに切り替える必要があります。

手順

1. セルフホスト型エンジンノードのいずれかにログインして、グローバルメンテナンスモードを有効にします。

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

2. 作業を進める前に、環境がメンテナーンスモードにあることを確認します。

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

クラスターがメンテナーンスモードにあることを示すメッセージが表示されるはずです。

16.1.10.2. バックアップを復元して既存のインストールを上書き

engine-backup コマンドを使用すると、Red Hat Virtualization Manager がすでにインストールおよび設定されているマシンにバックアップを復元できます。これは、環境のバックアップを取り、その環境で変更を実行し、バックアップから環境を復元して変更を元に戻したい場合に役立ちます。

ホストの追加や削除など、バックアップの作成後に環境に加えられた変更は、復元された環境には表示されません。これらの変更をやり直す必要があります。

手順

1. Manager マシンにログインします。
2. 設定ファイルを削除し、Manager に関連付けられているデータベースをクリーンアップします。

```
# engine-cleanup
```

engine-cleanup コマンドは、Manager データベースのみを削除します。データベースを削除したり、そのデータベースを所有しているユーザーを削除したりすることはありません。

3. フルバックアップまたはデータベースのみのバックアップを復元します。ユーザーとデータベースがすでに存在しているので、新規のデータベースを作成したり、データベースの認証情報を指定する必要はありません。

- 完全バックアップを復元します。

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

- 設定ファイルおよびデータベースバックアップを復元して、データベースのみのバックアップを復元します。

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

注記

Manager データベースのみを復元するには (たとえば、Data Warehouse データベースが別のマシンにある場合)、**-scope=dwhdb** パラメーターを省略できます。

成功すると、次の出力が表示されます。

```
You should now run engine-setup.
Done.
```

4. Manager を再設定します。

```
# engine-setup
```

16.1.10.3. グローバルメンテナーンスモードの無効化

手順

1. Manager 用仮想マシンにログインし、シャットダウンします。
2. セルフホスト型エンジンノードのいずれかにログインして、グローバルメンテナンスモードを無効にします。

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

グローバルメンテナンスモードを終了すると、ovirt-ha-agent が Manager 用仮想マシンを起動し、続いて Manager が自動的に起動します。Manager が起動するまでに最大で 10 分程度かかる場合があります。

3. 環境が動作していることを確認します。

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

情報の一覧に、**Engine status** が含まれます。**Engine status** の値は、以下のようになるはずです。

```
{"health": "good", "vm": "up", "detail": "Up"}
```

注記

仮想マシンが起動中で Manager がまだ動作していない場合、**Engine status** は以下のようになります。

```
{"reason": "bad vm status", "health": "bad", "vm": "up", "detail": "Powering up"}
```

このような場合には、数分間待ってからやり直してください。

環境が再び実行している場合は、停止した仮想マシンを起動して、環境内のリソースが期待どおりに動作していることを確認できます。

16.2. RED HAT VIRTUALIZATION データベースのリモートサーバーへの移行

16.2.1. Manager データベースのリモートサーバーへの移行

Red Hat Virtualization Manager の初期設定の後に、Manager (**engine**) データベースをリモートのデータベースサーバーに移行することができます。**engine-backup** を使用してデータベースのバックアップを作成し、新しいデータベースサーバーに復元します。

新規データベースサーバーに Red Hat Enterprise Linux 7 をインストールし、必要なリポジトリーを有効にする必要があります。

16.2.1.1. Red Hat Virtualization Manager リポジトリーの有効化

システムを Red Hat Subscription Manager に登録し、**Red Hat Virtualization Manager** のサブスクリプションをアタッチして Manager のリポジトリーを有効にします。

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。

```
# yum repolist
```

4. リポジトリーを設定します。

```
# subscription-manager repos \
    --disable='*' \
    --enable=rhel-7-server-rpms \
    --enable=rhel-7-server-supplementary-rpms \
    --enable=rhel-7-server-rhv-4.3-manager-rpms \
    --enable=rhel-7-server-rhv-4-manager-tools-rpms \
    --enable=rhel-7-server-ansible-2.9-rpms \
    --enable=jb-eap-7.2-for-rhel-7-server-rpms
```

Manager データベースのリモートサーバーへの移行

1. Red Hat Virtualization Manager マシンにログインし、**ovirt-engine** サービスを停止して、エンジンのバックアップに干渉しないようにします。

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

2. **engine** データベースのバックアップを作成します。

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

3. バックアップファイルを新規データベースサーバーにコピーします。

```
# scp /tmp/engine.dump root@new.database.server.com:/tmp
```

4. 新しいデータベースサーバーにログインし、**engine-backup** をインストールします。

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

5. 新しいデータベースサーバーで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=db --file=file_name --
log=log_file_name --provision-db --no-restore-permissions
```

6. データベースが移行されたので、**ovirt-engine** サービスを開始します。

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

16.2.2. リモートサーバーへのセルフホストエンジンデータベースの移行

Red Hat Virtualization Manager の初期設定の後に、セルフホストエンジンの **engine** データベースをリモートのデータベースサーバーに移行することができます。**engine-backup** を使用してデータベースのバックアップを作成し、新しいデータベースサーバーに復元します。

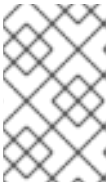
新規データベースサーバーに Red Hat Enterprise Linux 7 をインストールし、必要なリポジトリーを有効にする必要があります。

Red Hat Virtualization Manager リポジトリーの有効化
システムを Red Hat Subscription Manager に登録し、**Red Hat Virtualization Manager** のサブスクリプションをアタッチして Manager のリポジトリーを有効にします。

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

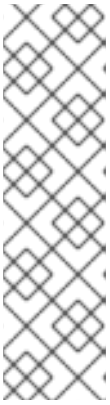
IPv6 ネットワークを使用している場合は、IPv6 移行メカニズムを使用して、コンテンツ配信ネットワークおよびサブスクリプションマネージャーにアクセスします。

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。

```
# yum repolist
```

4. リポジトリーを設定します。

```
# subscription-manager repos \
    --disable='*' \
    --enable=rhel-7-server-rpms \
    --enable=rhel-7-server-supplementary-rpms \
    --enable=rhel-7-server-rhv-4.3-manager-rpms \
    --enable=rhel-7-server-rhv-4-manager-tools-rpms \
    --enable=rhel-7-server-ansible-2.9-rpms \
    --enable=jb-eap-7.2-for-rhel-7-server-rpms
```

リモートサーバーへのセルフホストエンジンデータベースの移行

1. セルフホストエンジンノードにログインし、環境を **global** メンテナーンスモードにします。これにより、高可用性エージェントを無効化して、この手順の実行中に Manager 用仮想マシンが移行されないようにします。

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

2. Red Hat Virtualization Manager マシンにログインし、**ovirt-engine** サービスを停止して、エンジンのバックアップに干渉しないようにします。

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

3. **engine** データベースのバックアップを作成します。

```
# engine-backup --scope=files --scope=db --mode=backup --file=file_name --
log=backup_log_name
```

4. バックアップファイルを新規データベースサーバーにコピーします。

```
# scp /tmp/engine.dump root@new.database.server.com:/tmp
```

5. 新しいデータベースサーバーにログインし、**engine-backup** をインストールします。

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

6. 新しいデータベースサーバーで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=db --file=file_name --
log=restore_log_name --provision-db --no-restore-permissions
```

7. データベースが移行されたので、**ovirt-engine** サービスを開始します。

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

8. セルフホストエンジンノードにログインし、メンテナーンスモードをオフにして、高可用性エージェントを有効にします。

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

16.2.3. 別のマシンへの Data Warehouse の移行

本セクションでは、Data Warehouse データベースおよびサービスを Red Hat Virtualization Manager 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ことで、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。

Data Warehouse サービスを移行して既存の Data Warehouse データベース (**ovirt_engine_history**) に接続するか、Data Warehouse サービスを移行する前に Data Warehouse データベースを別のマシンに移行することができます。Data Warehouse データベースが Manager 上でホストされている場合、Data Warehouse サービスに加えてデータベースを移行することで、Manager マシンのリソースの競合がさらに減少します。データベースは、Data Warehouse サービスを移行するのと同じマシンに移行することも、Manager マシンと新しい Data Warehouse サービスマシンの両方とは別のマシンに移行することもできます。

16.2.3.1. 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** の詳細は、**engine-backup --help** を実行してください。

Data Warehouse サービスのみを移行するには、「別のマシンへの Data Warehouse サービスの移行」を参照してください。

新規データベースサーバーに Red Hat Enterprise Linux 7 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リポジトリーを有効にします。

Red Hat Virtualization Manager リポジトリーの有効化
システムを Red Hat Subscription Manager に登録し、**Red Hat Virtualization Manager** のサブスクリプションをアタッチして Manager のリポジトリーを有効にします。

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

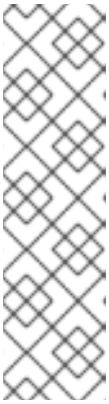
IPv6 ネットワークを使用している場合は、IPv6 移行メカニズムを使用して、コンテンツ配信ネットワークおよびサブスクリプションマネージャーにアクセスします。

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。

```
# yum repolist
```

4. リポジトリーを設定します。

```
# subscription-manager repos \
    --disable='*' \
    --enable=rhel-7-server-rpms \
    --enable=rhel-7-server-supplementary-rpms \
    --enable=rhel-7-server-rhv-4.3-manager-rpms \
    --enable=rhel-7-server-rhv-4-manager-tools-rpms \
    --enable=rhel-7-server-ansible-2.9-rpms \
    --enable=jb-eap-7.2-for-rhel-7-server-rpms
```

別のマシンへの Data Warehouse データベースの移行

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=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。

```
# yum 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。

```
# yum install rh-postgresql10 rh-postgresql10-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。

```
# scl enable rh-postgresql10 -- postgresql-setup --initdb
# systemctl enable rh-postgresql10-postgresql
# systemctl start rh-postgresql10-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=dwhdb --file=file_name --
log=log_file_name --provision-dwh-db --no-restore-permissions
```

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。

16.2.3.2. 別のマシンへの Data Warehouse サービスの移行

Red Hat Virtualization Manager マシンにインストールおよび設定した Data Warehouse サービスを、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**) を移行する場合は、「別のマシンへの Data Warehouse データベースの移行」 を参照してください。

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必要があります。
- 新たな Data Warehouse マシンを設定するには、以下の項目が必要です。
  - Manager **/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。「別のマシンへの Data Warehouse データベースの移行」 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、そのマシンでデータベースの設定中に定義した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

16.2.3.2.1. 新たな Data Warehouse マシンの準備

Red Hat Virtualization のリポジトリーを有効にし、Red Hat Enterprise Linux 7 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. 必要なリポジトリーを有効にします。
   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のユ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. リポジトリーを設定します。

   ```
   # subscription-manager repos \
       --disable='*' \
       --enable=rhel-7-server-rpms \
       --enable=rhel-7-server-supplementary-rpms \
       --enable=rhel-7-server-rhv-4.3-manager-rpms \
       --enable=rhel-7-server-rhv-4-manager-tools-rpms \
       --enable=rhel-7-server-ansible-2.9-rpms \
       --enable=jb-eap-7.2-for-rhel-7-server-rpms
   ```

2. 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。

   ```
   # yum update
   ```

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

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

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

1. Data Warehouse サービスを停止します。

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

2. データベースがリモートマシンでホストされる場合には、postgres.conf ファ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/opt/rh/rh-postgresql10/lib/pgsql/data/postgresql.conf** ファイルを編集し、listen_addresses 行を次のように変更します。

```
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には、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には、デフォルトでアクセス権限が付与されます。

Data Warehouse データベースの設定および移行方法に関する詳細情報は、「別のマシンへの Data Warehouse データベースの移行」 を参照してください。

3. postgresql サービスを再起動します。

```
# systemctl restart rh-postgresql10-postgresql
```

16.2.3.2.3. 新たな Data Warehouse マシンの設定

本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。

1. **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
```

2. **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
```

3. **Enter** を押して、Data Warehouse を設定します。

```
Configure Data Warehouse on this host (Yes, No) [Yes]:
```

4. **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]:
```

5. **Enter** キーを押してファイアウォールを自動設定するか、**No** と入力して **Enter** キーを押し、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ければ、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。

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

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
```

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。

16.3. バックアップストレージドメインを使用した仮想マシンのバックアップと復元

16.3.1. バックアップストレージドメインの説明

バックアップストレージドメインは、災害復旧、移行、その他のバックアップ/復旧の使用モデルのために、仮想マシンおよび仮想マシンテンプレートの保存と移行に特化して使用できるものです。バックアップドメインは、バックアップドメイン上のすべての仮想マシンがパワーダウン状態にあるという点で、非バックアップドメインとは異なります。仮想マシンはバックアップドメインで実行できません。

任意のデータストレージドメインをバックアップドメインとして設定できます。Manage Domain ダイアログボックスのチェックボックスを選択または選択解除することで、この設定を有効または無効にできます。この設定を有効にできるのは、そのストレージドメイン上のすべての仮想マシンが停止した後でのみです。

バックアップドメインに保存されている仮想マシンを起動することはできません。Manager は、これと、バックアップを無効にする可能性のあるその他の操作をブロックします。ただし、仮想マシンのディスクがバックアップドメインの一部でない場合は、バックアップドメインに保存されているテンプレートに基づいて仮想マシンを実行できます。

他のタイプのストレージドメインと同様に、バックアップドメインをデータセンターに接続したり、データセンターから切り離したりできます。そのため、バックアップの保存に加えて、バックアップドメインを使用してデータセンター間で仮想マシンを移行できます。

メリット

エクスポートドメインではなくバックアップドメインを使用するいくつかの理由を以下に示します。

- エクスポートドメインを 1 つだけにするのではなく、データセンターに複数のバックアップストレージドメインを含めることができます。
- バックアップと障害復旧に使用するバックアップストレージドメインを専用にすることができます。

- 仮想マシン、テンプレート、またはスナップショットのバックアップをバックアップストレージドメインに転送できます
- 多数の仮想マシン、テンプレート、または OVF ファイルの移行は、エクスポートドメインよりもバックアップドメインの方が圧倒的に高速に行えます。
- バックアップドメインは、エクスポートドメインよりも効率的にディスクスペースを使用します。
- バックアップドメインは、ファイルストレージ (NFS、Gluster) とブロックストレージ (ファイバーチャネルと iSCSI) の両方をサポートします。これは、ファイルストレージのみをサポートするエクスポートドメインとは対照的です。
- 制限を考慮して、ストレージドメインのバックアップ設定を動的に有効または無効にすることができます。

制約

- _backup ドメイン上のすべての仮想マシンまたはテンプレートは、同じドメイン上にすべてのディスクを持っている必要があります。
- ストレージドメインをバックアップドメインとして設定する前に、ストレージドメイン上のすべての仮想マシンの電源を切る必要があります。
- バックアップドメインに保存されている仮想マシンを実行することはできません。実行すると、ディスクのデータが操作される可能性があるためです。
- メモリーボリュームはアクティブな仮想マシンでのみサポートされているため、バックアップドメインをメモリーボリュームのターゲットにすることはできません。
- バックアップドメインで仮想マシンをプレビューすることはできません。
- 仮想マシンをバックアップドメインにライブ移行することはできません。
- バックアップドメインをマスタードメインとして設定することはできません。
- セルフホスト型エンジンのドメインをバックアップドメインとして設定することはできません。
- デフォルトのストレージドメインをバックアップドメインとして使用しないでください。

16.3.2. データストレージドメインをバックアップドメインに設定

前提条件

- ストレージドメイン上の仮想マシンまたはテンプレートに属するすべてのディスクは、同じドメイン上にある必要があります。
- ドメイン上のすべての仮想マシンの電源を切る必要があります。

手順

1. 管理ポータルで、**Storage → Domains** を選択します。
2. 新しいストレージドメインを作成するか、既存のストレージドメインを選択して、**ドメインの管理** をクリックします。ドメインの管理ダイアログボックスが開きます。

3. **詳細パラメーター** で、**Backup** チェックボックスを選択します。

これで、ドメインはバックアップドメインになります。

16.3.3. バックアップドメインを使用した仮想マシンまたはスナップショットのバックアップまたは復元

電源がオフになっている仮想マシンまたはスナップショットをバックアップできます。その後、バックアップを同じデータセンターに保存して必要に応じて復元したり、別のデータセンターに移行したりできます。

手順: 仮想マシンのバックアップ

1. バックアップドメインを作成します。「データストレージドメインをバックアップドメインに設定」 を参照してください。
2. バックアップする仮想マシンに基づいて、新しい仮想マシンを作成します。
   - スナップショットをバックアップするには、最初にスナップショットから仮想マシンを作成します。**Virtual Machine Management Guide**の Creating a Virtual Machine from a Snapshot を参照してください。
   - 仮想マシンをバックアップするには、最初に仮想マシンのクローンを作成します。**Virtual Machine Management Guide** の Cloning a Virtual Machine を参照してください。続行する前に、クローンの電源がオフになっていることを確認してください。
3. 新しい仮想マシンをバックアップドメインにエクスポートします。**Virtual Machine Management Guide** の Exporting a Virtual Machine to a Data Domain を参照してください。

手順: 仮想マシンの復元

1. 仮想マシンのバックアップを保存するバックアップストレージドメインがデータセンターに接続されていることを確認してください。
2. バックアップドメインから仮想マシンをインポートします。「インポート済みデータストレージドメインからの仮想マシンのインポート」を参照してください。

関連情報

- 「ストレージドメインのインポート」
- 「同じ環境内のデータセンター間でのストレージドメインの移行」
- 「異なる環境内のデータセンター間でのストレージドメインの移行」
- 「インポート済みデータストレージドメインからの仮想マシンのインポート」

16.4. バックアップおよび RESTORE API を使用した仮想マシンのバックアップおよび復元

16.4.1. バックアップおよび Restore API

バックアップおよび Restore API は、仮想マシンのフルバックアップまたはファイルレベルのバックアップ、および復元を実行できるようにする機能のコレクションです。API は、ライブスナップショッ

トや REST API などの Red Hat Virtualization のいくつかのコンポーネントを組み合わせて、独立したソフトウェアプロバイダーが提供するバックアップソフトウェアを含む仮想マシンに接続できる一時ボリュームを作成して操作します。

サポートされているサードパーティーのバックアップベンダーについては、Red Hat VirtualizationEcosystem を 参照してください。

16.4.2. 仮想マシンのバックアップ

バックアップおよび Restore API を使用して、仮想マシンをバックアップします。この手順では、バックアップを作成する仮想マシンと、バックアップを管理するためのソフトウェアがインストールされている仮想マシンの 2 つの仮想マシンがあることを前提としています。

仮想マシンのバックアップ

1. REST API を使用して、バックアップを作成する仮想マシンのスナップショットを作成します。

```
POST /api/vms/{vm:id}/snapshots/ HTTP/1.1
Accept: application/xml
Content-type: application/xml

<snapshot>
   <description>BACKUP</description>
</snapshot>
```

注記

- ここで、**{vm:id}** を、スナップショットを作成している仮想マシンの VM ID に置き換えます。この ID は、**管理ポータル** および **VM ポータル** の **New Virtual Machine** および **Edit Virtual Machine** ウィンドウの **General** タブから利用できます。
- 仮想マシンのスナップショットを作成すると、スナップショットの下の **initialization** にある **configuration** 属性の **data** 属性に現在の設定データが格納されます。

重要

共有可能としてマークされたディスク、または直接 LUN ディスクに基づくディスクのスナップショットを作成することはできません。

2. スナップショットの下の **data** 属性から仮想マシンの設定データを取得します。

```
GET /api/vms/{vm:id}/snapshots/{snapshot:id} HTTP/1.1
All-Content: true
Accept: application/xml
Content-type: application/xml
```

注記

- ここで、**{vm:id}** を、以前にスナップショットを作成した仮想マシンの ID に置き換えます。**{snapshot:id}** をスナップショット ID に置き換えます。
- **All-Content: true** ヘッダーを追加して、応答内の追加の OVF データを取得します。XML 応答の OVF データは、VM 設定要素 **<initialization> <configuration>** 内にあります。その後、このデータを使用して仮想マシンを復元します。

3. スナップショット ID を取得します。

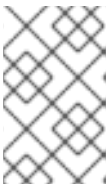
```
GET /api/vms/{vm:id}/snapshots/ HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

4. スナップショットのディスク ID を特定します。

```
GET /api/vms/{vm:id}/snapshots/{snapshot:id}/disks HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

5. スナップショットを、正しいインターフェイスタイプ (たとえば、**virtio_scsi**) を使用して、アクティブディスクアタッチメントとしてバックアップ仮想マシンにアタッチします。

```
POST /api/vms/{vm:id}/diskattachments/ HTTP/1.1
Accept: application/xml
Content-type: application/xml

<disk_attachment>
 <active>true</active>
 <interface>_virtio_scsi_</interface>
 <disk id="{disk:id}">
 <snapshot id="{snapshot:id}"/>
 </disk>
</disk_attachment>
```



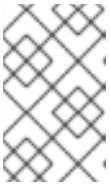
注記

ここで、**{vm:id}** を、以前にスナップショットを作成した仮想マシンではなく、**バックアップ** 仮想マシンの ID に置き換えます。**{disk:id}** をディスク ID に置き換えます。**{snapshot:id}** をスナップショット ID に置き換えます。

6. バックアップ仮想マシンのバックアップソフトウェアを使用して、スナップショットディスク上のデータをバックアップします。

7. バックアップ仮想マシンからスナップショットディスクアタッチメントを削除します。

```
DELETE /api/vms/{vm:id}/diskattachments/{snapshot:id} HTTP/1.1
Accept: application/xml
Content-type: application/xml
```



注記

ここで、**{vm:id}** を、以前にスナップショットを作成した仮想マシンではなく、**バックアップ** 仮想マシンの ID に置き換えます。**{snapshot:id}** をスナップショット ID に置き換えます。

8. 必要に応じて、スナップショットを削除します。

```
DELETE /api/vms/{vm:id}/snapshots/{snapshot:id} HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

注記

ここで、**{vm:id}** を、以前にスナップショットを作成した仮想マシンの ID に置き換えます。**{snapshot:id}** をスナップショット ID に置き換えます。

別の仮想マシンにインストールされたバックアップソフトウェアを使用して、一定の時点における仮想マシン状態のバックアップを作成しました。

16.4.3. 仮想マシンの復元

バックアップおよび Restore API を使用してバックアップされた仮想マシンを復元します。この手順は、前のバックアップの管理に使用されたソフトウェアがインストールされているバックアップ仮想マシンがあることを前提としています。

仮想マシンの復元

1. 管理ポータルで、バックアップを復元するフローティングディスクを作成します。フローティングディスクの作成方法の詳細は、「仮想ディスクの作成」 を参照してください。

2. ディスクをバックアップ仮想マシンに接続します。

```
POST /api/vms/{vm:id}/disks/ HTTP/1.1
Accept: application/xml
Content-type: application/xml

<disk id="{disk:id}">
</disk>
```

注記

ここで、**{vm:id}** を、以前にスナップショットを作成した仮想マシンではなく、この **バックアップ** 仮想マシンの ID に置き換えます。**{disk:id}** は、仮想マシンのバックアップ時に取得したディスク ID に置き換えます。

3. バックアップソフトウェアを使用して、バックアップをディスクに復元します。

4. バックアップ仮想マシンからディスクの割り当てを解除します。

```
DELETE /api/vms/{vm:id}/disks/{disk:id} HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

```
<action>
   <detach>true</detach>
</action>
```

+ 注: ここで、**{vm:id}** は、以前にスナップショットを作成した仮想マシンではなく、この **バックアップ** 仮想マシンの ID に置き換えます。**{disk:id}** をディスク ID に置き換えます。

5. 復元される仮想マシンの設定データを使用して、新しい仮想マシンを作成します。

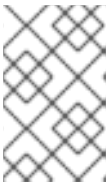
```
POST /api/vms/ HTTP/1.1
Accept: application/xml
Content-type: application/xml

<vm>
   <cluster>
      <name>cluster_name</name>
   </cluster>
   <name>_NAME_</name>
    <initialization>
    <configuration>
 <data>
 <!-- omitting long ovf data -->
 </data>
    <type>ovf</type>
    </configuration>
    </initialization>
   ...
</vm>
```

注記

仮想マシンの作成時に ovf の任意の値を上書きするには、**initialization** 要素の**前** または **後** に要素を再定義します。initialization 要素内では定義しません。

6. ディスクを新規の仮想マシンにアタッチします。

```
POST /api/vms/{vm:id}/disks/ HTTP/1.1
Accept: application/xml
Content-type: application/xml

<disk id="{disk:id}">
</disk>
```



注記

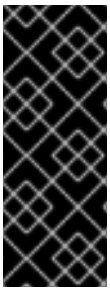
ここで、**{vm:id}** は、以前にスナップショットを作成した仮想マシンではなく、**新しい** 仮想マシンの ID に置き換えます。**{disk:id}** をディスク ID に置き換えます。

バックアップおよび Restore API を使用して作成されたバックアップを使用して、仮想マシンを復元しました。

第17章 RED HAT SATELLITE でのエラータ管理

Red Hat Virtualization は、Red Hat Virtualization Manager の Red Hat Satellite からエラータを表示するように設定できます。これにより、管理者は、ホスト、仮想マシン、および Manager が Red Hat Satellite プロバイダーに関連付けられた後に、それらのエラータとその重要性に関する更新を受け取ることができます。管理者は、必要なホスト、仮想マシン、または Manager で更新を実行して、それらの更新を適用できます。Red Hat Satellite の詳細は、Red Hat Satellite Documentation を参照してください。

Red Hat Virtualization 4.3 では、Red Hat Satellite 6.5 でのエラータ管理がサポートされます。



重要

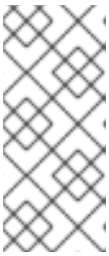
Satellite サーバーでは、Manager、ホスト、および仮想マシンは、FQDN で識別されます。これにより、外部コンテンツのホスト ID を Red Hat Virtualization で維持する必要がなくなります。

Manager、ホスト、および仮想マシンを管理していた Satellite アカウントには、管理者権限とデフォルトの組織セットが必要です。

Red Hat Virtualization エラータの設定

Manager、ホスト、および仮想マシンを Red Hat Satellite プロバイダーに関連付けるには、最初に Manager をプロバイダーに関連付ける必要があります。次に、ホストは同じプロバイダーに関連付けられ、設定されます。最後に、仮想マシンは同じプロバイダーに関連付けられ、設定されます。

1. 必要な Satellite サーバーを外部プロバイダーとして追加して Manager を関連付けます。詳細は、「ホストのプロビジョニング用の Red Hat Satellite インスタンスの追加」 を参照してください。



注記

Manager は、コンテンツホストとして Satellite サーバーに登録され、katello-agent パッケージがインストールされている必要があります。

ホスト登録の設定方法の詳細は、**Red Hat Satellite ホストの管理**の ホストの登録 を参照してください。

2. オプションで、使用可能なエラータを表示するように必要なホストを設定します。詳細は、「ホストの Satellite エラータ管理の設定」 を参照してください。

3. オプションで、使用可能なエラータを表示するように必要な仮想マシンを設定します。必要な仮想マシンを設定する前に、関連するホストを設定する必要があります。詳細は、**Virtual Machine Management Guide**の Configuring Red Hat Satellite Errata Management for a Virtual Machine を参照してください。

Red Hat Virtualization Manager エラータの表示

1. **Administration → Errata** をクリックします。

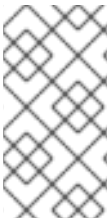
2. これらのエラータタイプのみを表示するには、**Security**、**Bugs**、または **Enhancements** チェックボックスをオンにします。

ホストで利用可能なエラータの表示に関する詳細は、「ホスト Errata の表示」 を、仮想マシンについては、**Virtual Machine Management Guide**の Viewing Red Hat Satellite Errata for a Virtual Machine を参照してください。

第18章 ANSIBLE を使用した設定タスクの自動化

Ansible は、システムの設定、ソフトウェアのデプロイ、ローリング更新の実行に使用する自動化ツールです。Ansible には Red Hat Virtualization のサポートが含まれ、Ansible モジュールを使用することで、データセンターの設定夜行性、ユーザーの管理、仮想マシン操作などのインストール後のタスクを自動化できます。

Ansible は、REST API や SDK と比較して、Red Hat Virtualization 設定を自動化する簡単な方法を提供し、他の Ansible モジュールと統合できます。Red Hat Virtualization で利用可能な Ansible モジュールの詳細は、Ansible ドキュメントの Ovirt モジュール を参照してください。



注記

Ansible Tower は、Ansible の Web インターフェイスと REST API を介してアクセスできるグラフィカルに有効化されたフレームワークです。Ansible Tower のサポートが必要な場合は、Red Hat Virtualization サブスクリプションの一部ではない Ansible Tower ライセンスが必要です。

Ansible は Red Hat Virtualization に同梱されています。Ansible をインストールするには、Manager マシンで以下のコマンドを実行します。

```
# yum install ansible
```

代替のインストール手順、および Ansible の使用に関する情報については、Ansible のドキュメント を参照してください。

注記

Ansible Playbook の実行時に Manager の詳細レベルを永続的に増やすには、以下の行を **/etc/ovirt-engine/engine.conf.d/** に設定ファイルを作成します。

ANSIBLE_PLAYBOOK_VERBOSE_LEVEL=4

ファイルの作成後に Manager を再起動するには、**systemctl restart ovirt-engine** を実行します。

18.1. ANSIBLE ロール

Red Hat Virtualization インフラストラクチャーのさまざまな部分を設定して管理するために、複数の Ansible ロールを使用できます。Ansible のロールは、大きな Playbook を他のユーザーと共有できる小さな再利用可能なファイルに分割することで、Ansible コードをモジュール化する方法を提供します。

Red Hat Virtualization で利用可能な Ansible ロールは、さまざまなインフラストラクチャーコンポーネントによって分類されます。Ansible ロールの詳細は、oVirt Ansible Roles のドキュメントを参照してください。Ansible ロールでインストールされるドキュメントは、「Ansible ロールのインストール」 を参照してください。

18.1.1. Ansible ロールのインストール

Red Hat Virtualization Manager リポジトリーから Red Hat Virtualization の Ansible ロールをインストールできます。以下のコマンドを使用して、Manager マシンに Ansible ロールをインストールします。

```
# yum install ovirt-ansible-roles
```

デフォルトでは、ロールは **/usr/share/ansible/roles** にインストールされます。**ovirt-ansible-roles** パッケージの構造は以下のとおりです。

- **/usr/share/ansible/roles**: ロールを保存します。
- **/usr/share/doc/ovirt-ansible-roles/**: 例、基本の概要、およびライセンスを保存します。
- **/usr/share/doc/ansible/roles/role_name**: ロール固有のドキュメントを保存します。

18.1.2. Ansible ロールを使用した Red Hat Virtualization の設定

次の手順では、Ansible ロールを使用して Red Hat Virtualization を設定する Playbook を作成および実行する方法について説明します。この例では、Ansible を使用してローカルマシンのマネージャーに接続し、新しいデータセンターを作成します。

前提条件

- **/etc/ansible/ansible.cfg** の **roles_path** オプションが Ansible ロールの場所 (**/usr/share/ansible/roles**) を参照していることを確認します。
- Playbook を実行しているマシンに Python SDK がインストールされていることを確認してください。

Ansible ロールを使用した Red Hat Virtualization の設定

1. 作業ディレクトリーにファイルを作成し、Red Hat Virtualization Manager ユーザーのパスワードを保存します。

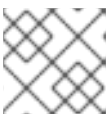
```
# cat passwords.yml
  ---
engine_password: youruserpassword
```

2. ユーザーパスワードを暗号化します。Vault パスワードが要求されます。

```
# ansible-vault encrypt passwords.yml
New Vault password:
Confirm New Vault password:
```

3. URL、証明書の場所、ユーザーなどの Manager の詳細を保存するファイルを作成します。

```
# cat engine_vars.yml
---
engine_url: https://example.engine.redhat.com/ovirt-engine/api
engine_user: admin@internal
engine_cafile: /etc/pki/ovirt-engine/ca.pem
```



注記

必要に応じて、これらの変数を Playbook に直接追加できます。

4. Playbook を作成します。これを簡素化するには、**/usr/share/doc/ovirt-ansible-roles/examples** の例をコピーして変更できます。

```
# cat rhv_infra.yml
```

```
---
- name: RHV infrastructure
  hosts: localhost
  connection: local
  gather_facts: false

  vars_files:
    # Contains variables to connect to the Manager
    - engine_vars.yml
    # Contains encrypted engine_password variable using ansible-vault
    - passwords.yml

  pre_tasks:
    - name: Login to RHV
      ovirt_auth:
        url: "{{ engine_url }}"
        username: "{{ engine_user }}"
        password: "{{ engine_password }}"
        ca_file: "{{ engine_cafile | default(omit) }}"
        insecure: "{{ engine_insecure | default(true) }}"
      tags:
        - always

  vars:
    data_center_name: mydatacenter
    data_center_description: mydatacenter
    data_center_local: false
    compatibility_version: 4.1

  roles:
    - ovirt-datacenters

  post_tasks:
    - name: Logout from RHV
      ovirt_auth:
        state: absent
        ovirt_auth: "{{ ovirt_auth }}"
      tags:
        - always
```

5. Playbook を実行します。

```
# ansible-playbook --ask-vault-pass rhv_infra.yml
```

ovirt-datacenters Ansible ロールを使用して **mydatacenter** という名前のデータセンターを作成できました。

第19章 ユーザーとロール

19.1. ユーザーの概要

Red Hat Virtualization には、ローカルドメインと外部ドメインの 2 種類のユーザードメインがあります。Manager のインストールプロセス中に、**内部** ドメインと呼ばれるデフォルトのローカルドメインとデフォルトのユーザー **admin** が作成されます。

ovirt-aaa-jdbc-tool を使用して、**internal** ドメインに追加のユーザーを作成できます。ローカルドメインに作成されたユーザーアカウントは、ローカルユーザーとして知られています。また、Red Hat Directory Server、Active Directory、OpenLDAP、その他多くのサポート対象オプションなどの外部 Directory Server を Red Hat Virtualization 環境にアタッチし、外部ドメインとして使用することも可能です。外部ドメインに作成されたユーザーアカウントは、ディレクトリーユーザーとして知られています。

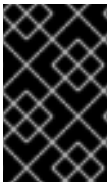
ローカルユーザーとディレクトリーユーザーの両方が、環境で機能する前に、管理ポータルを介して適切なロールおよび権限を割り当てる必要があります。ユーザーロールには、エンドユーザーと管理者の 2 つの主要なタイプがあります。エンドユーザーのロールは、VM ポータルからの仮想リソースを使用および管理します。管理者のロールは、管理ポータルを使用してシステムインフラストラクチャーを維持します。ロールは、仮想マシンやホストなどの個々のリソースのユーザーに割り当てることも、クラスターやデータセンターなどのオブジェクトの階層に割り当てることもできます。

19.2. DIRECTORY SERVER の概要

インストール中に、Red Hat Virtualization Manager は **内部** ドメインに **管理** ユーザーを作成します。このユーザーは、**admin@internal** とも呼ばれます。このアカウントは、環境の初期設定およびトラブルシューティングに使用することを目的としています。外部 Directory Server を接続し、ディレクトリーユーザーを追加し、適切なロールと権限を割り当てた後、必要がない場合は **admin@internal** ユーザーを無効にすることができます。サポートされる Directory Server は次のとおりです。

- 389ds
- 389ds RFC-2307 Schema
- Active Directory
- IBM Security Directory Server
- IBM Security Directory Server RFC-2307 Schema
- FreeIPA
- iDM
- Novell eDirectory RFC-2307 スキーマ
- OpenLDAP RFC-2307 Schema
- OpenLDAP Standard Schema
- Oracle Unified Directory RFC-2307 スキーマ
- RFC-2307 スキーマ (汎用)
- Red Hat Directory Server (RHDS)

- Red Hat Directory Server (RHDS) RFC-2307 Schema
- iPlanet



重要

Red Hat Virtualization Manager (**rhevm**) と IdM (**ipa-server**) を同じシステムにインストールすることはできません。IdM は、Red Hat Virtualization Manager で必要な **mod_ssl** パッケージと互換性がありません。

重要

ディレクトリーサーバーとして Active Directory を使用していて、テンプレートと仮想マシンの作成に **sysprep** を使用する場合は、Red Hat Virtualization の管理ユーザーにドメインの制御を委任する必要があります。

- **コンピューターをドメインに参加させる**
- **グループのメンバーシップを変更する**

Active Directory でユーザーアカウントを作成する方法の詳細は、http://technet.microsoft.com/en-us/library/cc732336.aspx を参照してください。

Active Directory での制御の委任に関する情報は、http://technet.microsoft.com/en-us/library/cc732524.aspx を参照してください。

19.3. 外部 LDAP プロバイダーの設定

19.3.1. 外部 LDAP プロバイダーの設定 (対話型セットアップ)

ovirt-engine-extension-aaa-ldap 拡張機能を使用すると、ユーザーは外部ディレクトリーの設定を簡単にカスタマイズできます。**ovirt-engine-extension-aaa-ldap** 拡張機能は多くの異なる LDAP サーバータイプをサポートし、ほとんどの LDAP タイプのセットアップを支援するインタラクティブなセットアップスクリプトが提供されています。

LDAP サーバーの種類が対話型セットアップスクリプトにリストされていない場合、またはさらにカスタマイズしたい場合は、設定ファイルを手動で編集できます。詳細は、「外部 LDAP プロバイダーの設定 (手動による方法)」 を参照してください。

Active Directory の例は、「Active Directory の接続」 を参照してください。

前提条件:

- DNS または LDAP サーバーのドメイン名を知っている必要があります。
- LDAP サーバーとマネージャーの間に安全な接続を設定するには、PEM でエンコードされた CA 証明書が準備されていることを確認してください。
- LDAP サーバーへの検索およびログインクエリーを実行するために、少なくとも 1 セットのアカウント名とパスワードを用意します。

外部 LDAP プロバイダーの設定

1. Red Hat Virtualization Manager で、LDAP 拡張パッケージをインストールします。

```
# yum install ovirt-engine-extension-aaa-ldap-setup
```

2. **ovirt-engine-extension-aaa-ldap-setup** を実行して、対話型セットアップを開始します。

```
# ovirt-engine-extension-aaa-ldap-setup
```

3. 対応する番号を入力して、LDAP タイプを選択します。LDAP サーバーがどのスキーマであるかわからない場合は、LDAP サーバータイプの標準スキーマを選択してください。Active Directory の場合は、「Active Directory の接続」 の手順に従います。

```
Available LDAP implementations:
 1 - 389ds
 2 - 389ds RFC-2307 Schema
 3 - Active Directory
 4 - IBM Security Directory Server
 5 - IBM Security Directory Server RFC-2307 Schema
 6 - IPA
 7 - Novell eDirectory RFC-2307 Schema
 8 - OpenLDAP RFC-2307 Schema
 9 - OpenLDAP Standard Schema
10 - Oracle Unified Directory RFC-2307 Schema
11 - RFC-2307 Schema (Generic)
12 - RHDS
13 - RHDS RFC-2307 Schema
14 - iPlanet
Please select:
```

4. **Enter** を押してデフォルトを受け入れ、LDAP サーバー名のドメイン名解決を設定します。

```
It is highly recommended to use DNS resolution for LDAP server.
If for some reason you intend to use hosts or plain address disable DNS usage.
Use DNS (Yes, No) [Yes]:
```

5. DNS ポリシー方式を選択します。

- オプション 1 の場合、**/etc/resolv.conf** にリストされている DNS サーバーを使用して IP アドレスを解決します。**/etc/resolv.conf** ファイルが正しい DNS サーバーで更新されていることを確認します。

- オプション 2 には、完全修飾ドメイン名 (FQDN) または LDAP サーバーの IP アドレスを入力します。SRV レコードで **dig** コマンドを使用して、ドメイン名を見つけることができます。SRV レコードは次の形式を取ります。

  ```
  _service._protocol.domain_name
  ```

  例: **dig _ldap._tcp.redhat.com SRV**。

- オプション 3 には、LDAP サーバーのスペース区切りのリストを入力します。サーバーの FQDN または IP アドレスのいずれかを使用します。このポリシーは、LDAP サーバー間の負荷分散を提供します。クエリーは、ラウンドロビンアルゴリズムに従ってすべての LDAP サーバーに分散されます。

- オプション 4 には、スペースで区切られた LDAP サーバーのリストを入力します。サーバーの FQDN または IP アドレスのいずれかを使用します。このポリシーは、クエリーに応

答するデフォルトの LDAP サーバーとなる最初の LDAP サーバーを定義します。最初のサーバーが使用できない場合、クエリーはリストの次の LDAP サーバーに移動します。

```
1 - Single server
2 - DNS domain LDAP SRV record
3 - Round-robin between multiple hosts
4 - Failover between multiple hosts
Please select:
```

6. LDAP サーバーがサポートする安全な接続方法を選択し、PEM でエンコードされた CA 証明書を取得する方法を指定します。

   - **File** を使用すると、証明書へのフルパスを指定できます。
   - **URL** を使用すると、証明書の URL を指定できます。
   - **Inline** を使用すると、証明書の内容をターミナルに貼り付けることができます。
   - **System** では、すべての CA ファイルのデフォルトの場所を指定できます。
   - **Insecure** は証明書の検証をスキップしますが、接続は引き続き TLS を使用して暗号化されます。

   ```
   NOTE:
   It is highly recommended to use secure protocol to access the LDAP server.
   Protocol startTLS is the standard recommended method to do so.
   Only in cases in which the startTLS is not supported, fallback to non standard ldaps protocol.
   Use plain for test environments only.
   Please select protocol to use (startTLS, ldaps, plain) [startTLS]: startTLS
   Please select method to obtain PEM encoded CA certificate (File, URL, Inline, System, Insecure):
   Please enter the password:
   ```

   注記

   LDAPS は、セキュアソケットリンクを介したライトウェイトディレクトリーアクセスプロトコルの略です。SSL 接続の場合は、**ldaps** オプションを選択します。

7. 検索ユーザーの識別名 (DN) を入力します。ユーザーには、Directory Server 上のすべてのユーザーとグループを参照するためのアクセス許可が必要です。検索ユーザーは、LDAP アノテーションで指定する必要があります。匿名検索が許可されている場合は、入力せずに **Enter** を押します。

   ```
   Enter search user DN (for example uid=username,dc=example,dc=com or leave empty for anonymous): uid=user1,ou=Users,ou=department-1,dc=example,dc=com
   Enter search user password:
   ```

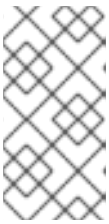
8. ベース DN を入力します。

   ```
   Please enter base DN (dc=redhat,dc=com) [dc=redhat,dc=com]: ou=department-1,dc=redhat,dc=com
   ```

9. 仮想マシン用に Single Sign-On を設定する場合は、**Yes** を選択します。この機能は、管理ポータル機能に対する Single Sign-On では使用できないことに注意してください。スクリプトは、プロファイル名がドメイン名と一致する必要があることを通知します。**仮想マシン管理ガイド** の 仮想マシン用の Single Sign-On の設定 の手順に従う必要があります。

> Are you going to use Single Sign-On for Virtual Machines (Yes, No) [Yes]:

10. プロファイル名を指定します。プロファイル名は、ログインページでユーザーに表示されます。この例では、**redhat.com** を使用しています。

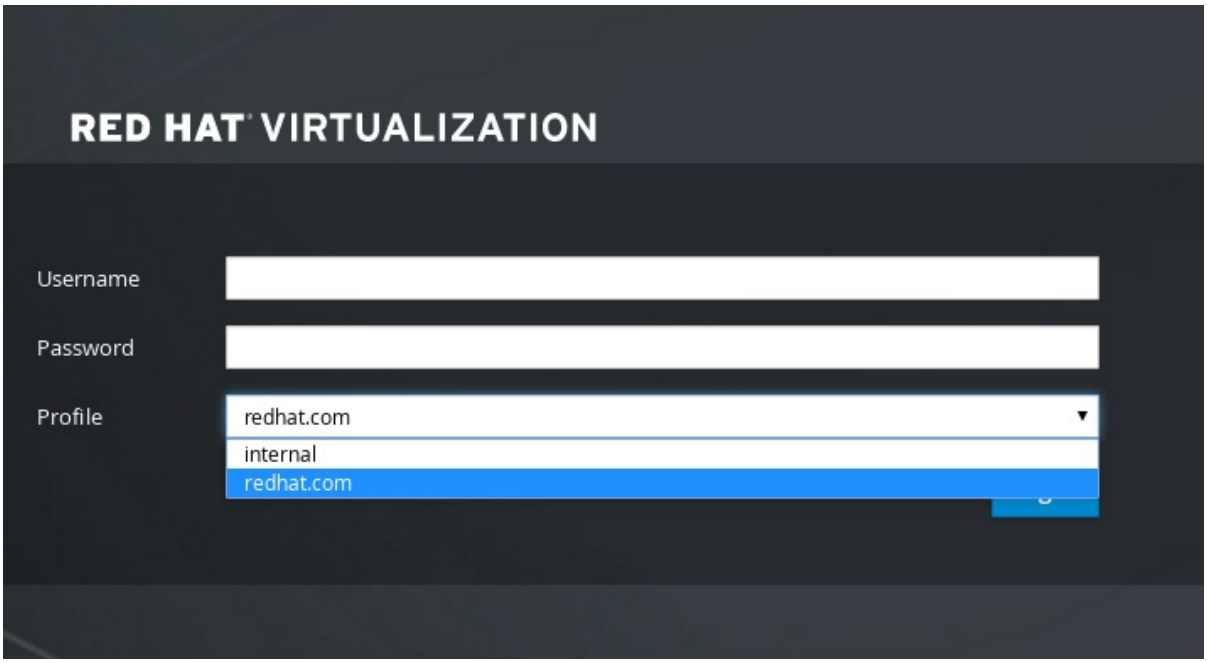


注記

ドメインの設定後にプロファイルの名前を変更するには、**/etc/ovirt-engine/extensions.d/redhat.com-authn.properties** ファイルの **ovirt.engine.aaa.authn.profile.name** 属性を編集します。変更を反映するには、**ovirt-engine** サービスを再起動します。

> Please specify profile name that will be visible to users: **redhat.com**

図19.1 管理ポータルのログインページ



注記

ユーザーは、初めてログインするときにドロップダウンリストからプロファイルを選択する必要があります。情報はブラウザーの Cookie に保存され、ユーザーが次にログインしたときに事前に選択されます。

11. ログイン機能をテストして、LDAP サーバーが Red Hat Virtualization 環境に正しく接続されていることを確認します。ログインクエリーには、**user name** および **password** を入力します。

> NOTE:
> It is highly recommended to test drive the configuration before applying it into engine.
> Login sequence is executed automatically, but it is recommended to also execute Search sequence manually after successful Login sequence.

```
Please provide credentials to test login flow:
Enter user name:
Enter user password:
[ INFO  ] Executing login sequence...
...
[ INFO  ] Login sequence executed successfully
```

12. ユーザーの詳細が正しいことを確認してください。ユーザーの詳細が正しくない場合は、**Abort** を選択します。

```
Please make sure that user details are correct and group membership meets expectations
(search for PrincipalRecord and GroupRecord titles).
Abort if output is incorrect.
Select test sequence to execute (Done, Abort, Login, Search) [Abort]:
```

13. 検索機能を手動でテストすることが推奨されます。検索クエリーで、ユーザーアカウントの場合は **Principal**、グループアカウントの場合は **Group** を選択します。ユーザーアカウントのグループアカウント情報を返す場合は、**Resolve Groups** で **Yes** を選択します。3 つの設定ファイルが作成され、画面出力に表示されます。

```
Select test sequence to execute (Done, Abort, Login, Search) [Search]: Search
Select entity to search (Principal, Group) [Principal]:
Term to search, trailing '*' is allowed: testuser1
Resolve Groups (Yes, No) [No]:
```

14. **Done** を選択してセットアップを完了します。

```
Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Done
[ INFO  ] Stage: Transaction setup
[ INFO  ] Stage: Misc configuration
[ INFO  ] Stage: Package installation
[ INFO  ] Stage: Misc configuration
[ INFO  ] Stage: Transaction commit
[ INFO  ] Stage: Closing up
CONFIGURATION SUMMARY
Profile name is: redhat.com
The following files were created:
    /etc/ovirt-engine/aaa/redhat.com.properties
    /etc/ovirt-engine/extensions.d/redhat.com.properties
    /etc/ovirt-engine/extensions.d/redhat.com-authn.properties
[ INFO  ] Stage: Clean up
Log file is available at /tmp/ovirt-engine-extension-aaa-ldap-setup-20171004101225-
mmneib.log:
[ INFO  ] Stage: Pre-termination
[ INFO  ] Stage: Termination
```

15. **ovirt-engine** サービスを再起動します。作成したプロファイルは、管理ポータルおよび仮想マシンポータルのログインページで利用できるようになります。LDAP サーバー上のユーザーアカウントに適切なロールとパーミッションを割り当て、たとえば、VM ポータルにログインするには、「管理ポータルからのユーザータスクの管理」 を参照してください。

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

注記

詳細は、**/usr/share/doc/ovirt-engine-extension-aaa-ldap-version** の LDAP 認証および承認拡張 README ファイルを参照してください。

19.3.2. Active Directory の接続

前提条件

- Active Directory フォレスト名を知っている必要があります。フォレスト名は、ルートドメイン名とも呼ばれます。

注記

ovirt-engine-extension-aaa-ldap-setup ツールで設定できない、最も一般的な Active Directory の設定例は **/usr/share/ovirt-engine-extension-aaa-ldap/examples/README.md** に記載されています。

- Active Directory フォレスト名を解決できる DNS サーバーを Manager の **/etc/resolv.conf** ファイルに追加するか、Active Directory DNS サーバーを書き留めて、対話型セットアップスクリプトのプロンプトが表示されたら入力する必要があります。
- LDAP サーバーと Manager の間に安全な接続を設定するには、PEM でエンコードされた CA 証明書が準備されていることを確認してください。詳細は、「Manager と LDAP サーバー間の暗号化通信の設定」 を参照してください。
- 匿名検索がサポートされていない限り、検索ユーザーとして使用するには、すべてのユーザーとグループを参照する権限を持つユーザーが Active Directory で利用可能である必要があります。検索ユーザーの識別名 (DN) を書き留めます。Active Directory の管理ユーザーを使用しないでください。
- Active Directory への検索およびログインクエリーを実行するには、アカウント名とパスワードを 少なくとも 1 つ用意しておく必要があります。
- Active Directory の展開が複数のドメインにまたがる場合は、**/usr/share/ovirt-engine-extension-aaa-ldap/profiles/ad.properties** ファイルに記載されている制限に注意してください。

外部 LDAP プロバイダーの設定

1. Red Hat Virtualization Manager で、LDAP 拡張パッケージをインストールします。

```
# yum install ovirt-engine-extension-aaa-ldap-setup
```

2. **ovirt-engine-extension-aaa-ldap-setup** を実行して、対話型セットアップを開始します。

```
# ovirt-engine-extension-aaa-ldap-setup
```

3. 対応する番号を入力して、LDAP タイプを選択します。この手順の後の LDAP 関連の質問は、LDAP タイプごとに異なります。

```
Available LDAP implementations:
 1 - 389ds
 2 - 389ds RFC-2307 Schema
 3 - Active Directory
```

```
 4 - IBM Security Directory Server
 5 - IBM Security Directory Server RFC-2307 Schema
 6 - IPA
 7 - Novell eDirectory RFC-2307 Schema
 8 - OpenLDAP RFC-2307 Schema
 9 - OpenLDAP Standard Schema
10 - Oracle Unified Directory RFC-2307 Schema
11 - RFC-2307 Schema (Generic)
12 - RHDS
13 - RHDS RFC-2307 Schema
14 - iPlanet
Please select: 3
```

4. Active Directory フォレスト名を入力します。フォレスト名が Manager の DNS で解決できない場合、スクリプトは、スペースで区切られた Active Directory サーバー名のリストを入力するように求めます。

```
Please enter Active Directory Forest name: ad-example.redhat.com
[ INFO  ] Resolving Global Catalog SRV record for ad-example.redhat.com
[ INFO  ] Resolving LDAP SRV record for ad-example.redhat.com
```

5. LDAP サーバーがサポートする安全な接続方法を選択し、PEM でエンコードされた CA 証明書を取得する方法を指定します。ファイルオプションを使用すると、証明書へのフルパスを指定できます。URL オプションを使用すると、証明書への URL を指定できます。インラインオプションを使用して、証明書の内容をターミナルに貼り付けます。システムオプションを使用すると、すべての CA ファイルの場所を指定できます。安全でないオプションを使用すると、startTLS を安全でないモードで使用できます。

```
NOTE:
It is highly recommended to use secure protocol to access the LDAP server.
Protocol startTLS is the standard recommended method to do so.
Only in cases in which the startTLS is not supported, fallback to non standard ldaps protocol.
Use plain for test environments only.
Please select protocol to use (startTLS, ldaps, plain) [startTLS]: startTLS
Please select method to obtain PEM encoded CA certificate (File, URL, Inline, System,
Insecure): File
Please enter the password:
```

注記

LDAPS は、セキュアソケットリンクを介したライトウェイトディレクトリーアクセスプロトコルの略です。SSL 接続の場合は、**ldaps** オプションを選択します。

PEM でエンコードされた CA 証明書の作成に関する詳細は、「Manager と LDAP サーバー間の暗号化通信の設定」 を参照してください。

6. 検索ユーザーの識別名 (DN) を入力します。ユーザーには、Directory Server 上のすべてのユーザーとグループを参照するためのアクセス許可が必要です。検索ユーザーは LDAP アノテーションである必要があります。匿名検索が許可されている場合は、入力せずに **Enter** を押します。

```
Enter search user DN (empty for anonymous):
cn=user1,ou=Users,dc=test,dc=redhat,dc=com
Enter search user password:
```

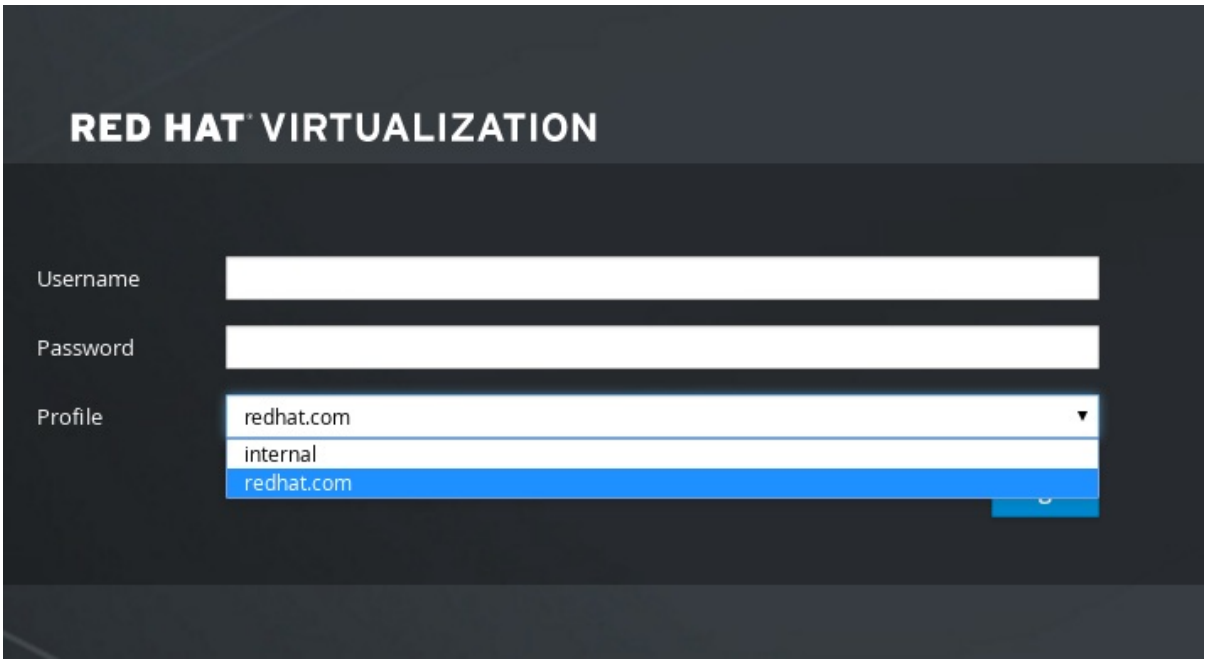
7. 仮想マシンに Single Sign-On を使用するかどうかを指定します。この機能はデフォルトで有効になっていますが、管理ポータルへの Single Sign-On が有効になっている場合は使用できません。スクリプトは、プロファイル名がドメイン名と一致する必要があることを通知します。**仮想マシン管理ガイド** の 仮想マシン用の Single Sign-On の設定 の手順に従う必要があります。

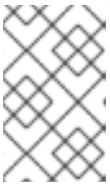
```
Are you going to use Single Sign-On for Virtual Machines (Yes, No) [Yes]:
```

8. プロファイル名を指定します。プロファイル名は、ログインページでユーザーに表示されます。この例では、**redhat.com** を使用しています。

```
Please specify profile name that will be visible to users:redhat.com
```

図19.2 管理ポータルのログインページ





注記

ユーザーは、初めてログインするときに、ドロップダウンリストから目的のプロファイルを選択する必要があります。その後、情報はブラウザーの Cookie に保存され、ユーザーが次にログインしたときに事前に選択されます。

9. 検索およびログイン機能をテストして、LDAP サーバーが Red Hat Virtualization 環境に正しく接続されていることを確認します。ログインクエリーには、アカウント名とパスワードを入力します。検索クエリーで、ユーザーアカウントの場合は **Principal** を選択し、グループアカウントの場合は **Group** を選択します。ユーザーアカウントのグループアカウント情報を返す場合は、**Resolve Groups** に **Yes** を入力します。**Done** を選択してセットアップを完了します。3 つの設定ファイルが作成され、画面出力に表示されます。

```
NOTE:
It is highly recommended to test drive the configuration before applying it into engine.
Login sequence is executed automatically, but it is recommended to also execute Search
sequence manually after successful Login sequence.
```

```
Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Login
Enter search user name: testuser1
Enter search user password:
[ INFO  ] Executing login sequence...
...
Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Search
Select entity to search (Principal, Group) [Principal]:
Term to search, trailing '*' is allowed: testuser1
Resolve Groups (Yes, No) [No]:
[ INFO  ] Executing login sequence...
...
Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Done
[ INFO  ] Stage: Transaction setup
[ INFO  ] Stage: Misc configuration
[ INFO  ] Stage: Package installation
[ INFO  ] Stage: Misc configuration
[ INFO  ] Stage: Transaction commit
[ INFO  ] Stage: Closing up
        CONFIGURATION SUMMARY
        Profile name is: redhat.com
        The following files were created:
          /etc/ovirt-engine/aaa/redhat.com.properties
          /etc/ovirt-engine/extensions.d/redhat.com-authz.properties
          /etc/ovirt-engine/extensions.d/redhat.com-authn.properties
[ INFO  ] Stage: Clean up
        Log file is available at /tmp/ovirt-engine-extension-aaa-ldap-setup-20160114064955-
1yar9i.log:
[ INFO  ] Stage: Pre-termination
[ INFO  ] Stage: Termination
```

10. 作成したプロファイルは、管理ポータルおよび仮想マシンポータルのログインページで利用できるようになります。LDAP サーバー上のユーザーアカウントに適切なロールとパーミッションを割り当て、たとえば、VM ポータルにログインするには、「管理ポータルからのユーザータスクの管理」 を参照してください。

注記

詳細は、**/usr/share/doc/ovirt-engine-extension-aaa-ldap-version** の LDAP 認証および承認拡張 README ファイルを参照してください。

19.3.3. 外部 LDAP プロバイダーの設定 (手動による方法)

ovirt-engine-extension-aaa-ldap 拡張機能は、LDAP プロトコルを使用してディレクトリーサーバーにアクセスし、完全にカスタマイズ可能です。仮想マシンポータルまたは管理ポータル機能への Single Sign-On を有効にする場合を除いて、Kerberos 認証は必要ありません。

前のセクションの対話型セットアップ方法でユースケースがカバーされていない場合は、設定ファイルを手動で変更して LDAP サーバーを接続できます。次の手順では、一般的な詳細を使用します。具体的な値は、設定によって異なります。

外部 LDAP プロバイダーの設定 (手動による方法)

1. Red Hat Virtualization Manager で、LDAP 拡張パッケージをインストールします。

```
# yum install ovirt-engine-extension-aaa-ldap
```

2. LDAP 設定テンプレートファイルを **/etc/ovirt-engine** ディレクトリーにコピーします。テンプレートファイルは、アクティブなディレクトリー (**ad**) およびその他のディレクトリータイプ (**simple**) で使用できます。この例では、単純な設定テンプレートを使用しています。

```
# cp -r /usr/share/ovirt-engine-extension-aaa-ldap/examples/simple/. /etc/ovirt-engine
```

3. 管理ポータルおよび仮想マシンポータルのログインページでユーザーに表示するプロファイル名と一致するように、設定ファイルの名前を変更します。

```
# mv /etc/ovirt-engine/aaa/profile1.properties /etc/ovirt-engine/aaa/example.properties
# mv /etc/ovirt-engine/extensions.d/profile1-authn.properties /etc/ovirt-engine/extensions.d/example-authn.properties
# mv /etc/ovirt-engine/extensions.d/profile1-authz.properties /etc/ovirt-engine/extensions.d/example-authz.properties
```

4. LDAP サーバーの種類のコメントを解除し、ドメインとパスワードのフィールドを更新して、LDAP プロパティー設定ファイルを編集します。

```
#  vi /etc/ovirt-engine/aaa/example.properties
```

例19.1 プロファイルの例: LDAP サーバーセクション

```
# Select one
#
include = <openldap.properties>
#include = <389ds.properties>
#include = <rhds.properties>
#include = <ipa.properties>
#include = <iplanet.properties>
#include = <rfc2307-389ds.properties>
#include = <rfc2307-rhds.properties>
#include = <rfc2307-openldap.properties>
#include = <rfc2307-edir.properties>
#include = <rfc2307-generic.properties>

# Server
#
vars.server = ldap1.company.com

# Search user and its password.
#
vars.user = uid=search,cn=users,cn=accounts,dc=company,dc=com
vars.password = 123456

pool.default.serverset.single.server = ${global:vars.server}
pool.default.auth.simple.bindDN = ${global:vars.user}
pool.default.auth.simple.password = ${global:vars.password}
```

TLS または SSL プロトコルを使用して LDAP サーバーと対話するには、LDAP サーバーのルート CA 証明書を取得し、それを使用して公開鍵ストアファイルを作成します。次の行のコメントを解除し、公開キーストアファイルへのフルパスとファイルにアクセスするためのパスワードを指定します。

注記

パブリックキーストアファイルの作成方法の詳細は、「Manager と LDAP サーバー間の暗号化通信の設定」を参照してください。

例19.2 プロファイルの例: キーストアセクション

```
# Create keystore, import certificate chain and uncomment
# if using tls.
pool.default.ssl.startTLS = true
pool.default.ssl.truststore.file = /full/path/to/myrootca.jks
pool.default.ssl.truststore.password = password
```

5. 認証設定ファイルを確認します。管理ポータルおよび仮想マシンポータルのログインページでユーザーに表示されるプロファイル名は、**ovirt.engine.aaa.authn.profile.name** によって定義されます。設定プロファイルの場所は、LDAP 設定ファイルの場所と一致する必要があります。すべてのフィールドをデフォルトのままにすることができます。

```
# vi /etc/ovirt-engine/extensions.d/example-authn.properties
```

例19.3 認証設定ファイルの例

```
ovirt.engine.extension.name = example-authn
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.ldap
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.ldap.AuthnExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authn
ovirt.engine.aaa.authn.profile.name = example
ovirt.engine.aaa.authn.authz.plugin = example-authz
config.profile.file.1 = ../aaa/example.properties
```

6. 許可設定ファイルを確認してください。設定プロファイルの場所は、LDAP 設定ファイルの場所と一致する必要があります。すべてのフィールドをデフォルトのままにすることができます。

```
# vi /etc/ovirt-engine/extensions.d/example-authz.properties
```

例19.4 許可設定ファイルの例

```
ovirt.engine.extension.name = example-authz
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.ldap
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.ldap.AuthzExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authz
config.profile.file.1 = ../aaa/example.properties
```

7. 設定プロファイルの所有権および権限が適切であることを確認してください。

```
# chown ovirt:ovirt /etc/ovirt-engine/aaa/example.properties
# chmod 600 /etc/ovirt-engine/aaa/example.properties
```

8. エンジンサービスを再起動します。

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

9. 作成した **example** プロファイルは、管理ポータルおよび仮想マシンポータルのログインページで利用できるようになります。LDAP サーバー上のユーザーアカウントに適切なパーミッションを割り当て、たとえば、VM ポータルにログインするには、「管理ポータルからのユーザータスクの管理」 を参照してください。

注記

詳細は、**/usr/share/doc/ovirt-engine-extension-aaa-ldap-version** の LDAP 認証および承認拡張 README ファイルを参照してください。

19.3.4. 外部 LDAP プロバイダーの削除

この手順では、外部で設定された LDAP プロバイダーとそのユーザーを削除する方法を示します。

外部 LDAP プロバイダーの削除

1. LDAP プロバイダー設定ファイルを削除し、デフォルト名 **profile1** を置き換えます。

```
# rm /etc/ovirt-engine/extensions.d/profile1-authn.properties
# rm /etc/ovirt-engine/extensions.d/profile1-authz.properties
# rm /etc/ovirt-engine/aaa/profile1.properties
```

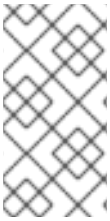
2. **ovirt-engine** サービスを再起動します。

```
# systemctl restart ovirt-engine
```

3. 管理ポータルの **Users** リソースタブで、このプロバイダーのユーザー (**Authorization provider** が **profile1**-authz であるユーザー) を選択し、**Remove** をクリックします。

19.4. SINGLE SIGN-ON 用の LDAP および KERBEROS の設定

Single Sign-On を使用すると、ユーザーはパスワードを再入力せずに VM ポータルまたは管理ポータルにログインできます。認証情報は Kerberos サーバーから取得されます。管理ポータルと仮想マシンポータルへの Single Sign-On を設定するには、**ovirt-engine-extension-aaa-misc** および **ovirt-engine-extension-aaa-ldap** の 2 つの拡張機能と、2 つの Apache モジュール **mod_auth_gssapi** および d **mod_session** を設定する必要があります。Kerberos を含まない Single Sign-On を設定できますが、これはこのドキュメントの範囲外です。



注記

VM ポータルへのシングルサインオンが有効になっている場合に、仮想マシンへのシングルサインオンはできません。VM ポータルへのシングルサインオンが有効になっている場合には、VM ポータルはパスワードを受け入れる必要がないので、パスワードを委任して仮想マシンにサインインすることはできません。

この例では、以下を前提としています。

- 既存の KeyDistributionCenter (KDC) サーバーは、MIT バージョンの Kerberos5 を使用します。
- KDC サーバーに対する管理者権限があります。
- Kerberos クライアントは、Red Hat Virtualization Manager とユーザーマシンにインストールされます。
- **kadmin** ユーティリティーは、Kerberos サービスプリンシパルと **keytab** ファイルを作成するために使用されます。

この手順には、次のコンポーネントが含まれます。

On the KDC server

- Red Hat Virtualization Manager 上で Apache サービス用のサービスプリンシパルと **keytab** ファイルを作成します。

Red Hat Virtualization Manager の場合

- 認証および許可拡張パッケージと Apache Kerberos 認証モジュールをインストールします。
- 拡張ファイルを設定します。

Apache サービス用の Kerberos の設定

1. KDC サーバーで、**kadmin** ユーティリティーを使用して、Red Hat Virtualization Manager で Apache サービスのサービスプリンシパルを作成します。サービスプリンシパルは、Apache サービスの KDC への参照 ID です。

```
# kadmin
kadmin> addprinc -randkey HTTP/fqdn-of-rhevm@REALM.COM
```

2. Apache サービスの **keytab** ファイルを生成します。**keytab** ファイルには、共有秘密鍵が格納されています。

注記

engine-backup コマンドには、バックアップおよび復元時にファイル **/etc/httpd/http.keytab** が含まれます。**keytab** ファイルに別の名前を使用する場合は、必ずバックアップして復元してください。

```
kadmin> ktadd -k /tmp/http.keytab HTTP/fqdn-of-rhevm@REALM.COM
kadmin> quit
```

3. KDC サーバーから Red Hat Virtualization Manager に **keytab** ファイルをコピーします。

```
# scp /tmp/http.keytab root@rhevm.example.com:/etc/httpd
```

仮想マシンポータルまたは管理ポータルへの Single Sign-On の設定

1. Red Hat Virtualization Manager で、キータブの所有権と権限が適切であることを確認します。

```
# chown apache /etc/httpd/http.keytab
# chmod 400 /etc/httpd/http.keytab
```

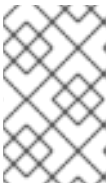
2. 認証拡張パッケージ、LDAP 拡張パッケージ、および **mod_auth_gssapi** および **mod_sessionApache** モジュールをインストールします。

```
# yum install ovirt-engine-extension-aaa-misc ovirt-engine-extension-aaa-ldap
mod_auth_gssapi mod_session
```

3. SSO 設定テンプレートファイルを **/etc/ovirt-engine** ディレクトリーにコピーします。テンプレートファイルは、Active Directory (**ad-sso**) およびその他のディレクトリータイプ (**simple-sso**) で使用できます。この例では、単純な SSO 設定テンプレートを使用しています。

```
# cp -r /usr/share/ovirt-engine-extension-aaa-ldap/examples/simple-sso/. /etc/ovirt-engine
```

4. **ovirt-sso.conf** を Apache 設定ディレクトリーに移動します。



注記

engine-backup コマンドは、バックアップとリストアの際に、**/etc/httpd/conf.d/ovirt-sso.conf** ファイルを含めます。このファイルに別の名前を使用する場合は、必ずバックアップして復元してください。

```
# mv /etc/ovirt-engine/aaa/ovirt-sso.conf /etc/httpd/conf.d
```

5. 認証方法ファイルを確認します。レルムは **keytab** ファイルから自動的にフェッチされるため、このファイルを編集する必要はありません。

```
# vi /etc/httpd/conf.d/ovirt-sso.conf
```

例19.5 認証方法ファイルの例

```
<LocationMatch ^/ovirt-engine/sso/(interactive-login-negotiate|oauth/token-http-
auth)|^/ovirt-engine/api>
  <If "req('Authorization') !~ /^(Bearer|Basic)/i">
    RewriteEngine on
    RewriteCond %{LA-U:REMOTE_USER} ^(.*)$
    RewriteRule ^(.*)$ - [L,NS,P,E=REMOTE_USER:%1]
    RequestHeader set X-Remote-User %{REMOTE_USER}s

    AuthType GSSAPI
    AuthName "Kerberos Login"

    # Modify to match installation
    GssapiCredStore keytab:/etc/httpd/http.keytab
    GssapiUseSessions On
    Session On
    SessionCookieName ovirt_gssapi_session path=/private;httponly;secure;

    Require valid-user
    ErrorDocument 401 "<html><meta http-equiv=\"refresh\" content=\"0; url=/ovirt-
engine/sso/login-unauthorized\"/><body><a href=\"/ovirt-engine/sso/login-
```

```
unauthorized\">Here</a></body></html>"
  </If>
</LocationMatch>
```

6. 管理ポータルおよび仮想マシンポータルのログインページでユーザーに表示するプロファイル名と一致するように、設定ファイルの名前を変更します。

```
# mv /etc/ovirt-engine/aaa/profile1.properties /etc/ovirt-engine/aaa/example.properties
```

```
# mv /etc/ovirt-engine/extensions.d/profile1-http-authn.properties /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

```
# mv /etc/ovirt-engine/extensions.d/profile1-http-mapping.properties /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

```
# mv /etc/ovirt-engine/extensions.d/profile1-authz.properties /etc/ovirt-engine/extensions.d/example-authz.properties
```

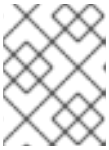
7. LDAP サーバーの種類のコメントを解除し、ドメインとパスワードのフィールドを更新して、LDAP プロパティー設定ファイルを編集します。

```
#  vi /etc/ovirt-engine/aaa/example.properties
```

例19.6 プロファイルの例: LDAP サーバーセクション

```
# Select one
include = <openldap.properties>
#include = <389ds.properties>
#include = <rhds.properties>
#include = <ipa.properties>
#include = <iplanet.properties>
#include = <rfc2307-389ds.properties>
#include = <rfc2307-rhds.properties>
#include = <rfc2307-openldap.properties>
#include = <rfc2307-edir.properties>
#include = <rfc2307-generic.properties>

# Server
#
vars.server = ldap1.company.com

# Search user and its password.
#
vars.user = uid=search,cn=users,cn=accounts,dc=company,dc=com
vars.password = 123456

pool.default.serverset.single.server = ${global:vars.server}
pool.default.auth.simple.bindDN = ${global:vars.user}
pool.default.auth.simple.password = ${global:vars.password}
```

TLS または SSL プロトコルを使用して LDAP サーバーと対話するには、LDAP サーバーのルート CA 証明書を取得し、それを使用して公開鍵ストアファイルを作成します。次の行のコメントを解除し、公開キーストアファイルへのフルパスとファイルにアクセスするためのパスワードを指定します。



注記

パブリックキーストアファイルの作成方法の詳細は、「Manager と LDAP サーバー間の暗号化通信の設定」 を参照してください。

例19.7 プロファイルの例: キーストアセクション

```
# Create keystore, import certificate chain and uncomment
# if using ssl/tls.
pool.default.ssl.startTLS = true
pool.default.ssl.truststore.file = /full/path/to/myrootca.jks
pool.default.ssl.truststore.password = password
```

8. 認証設定ファイルを確認します。管理ポータルおよび仮想マシンポータルのログインページでユーザーに表示されるプロファイル名は、**ovirt.engine.aaa.authn.profile.name** によって定義されます。設定プロファイルの場所は、LDAP 設定ファイルの場所と一致する必要があります。すべてのフィールドをデフォルトのままにすることができます。

```
# vi /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

例19.8 認証設定ファイルの例

```
ovirt.engine.extension.name = example-http-authn
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.http.AuthnExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authn
ovirt.engine.aaa.authn.profile.name = example-http
ovirt.engine.aaa.authn.authz.plugin = example-authz
ovirt.engine.aaa.authn.mapping.plugin = example-http-mapping
config.artifact.name = HEADER
config.artifact.arg = X-Remote-User
```

9. 許可設定ファイルを確認してください。設定プロファイルの場所は、LDAP 設定ファイルの場所と一致する必要があります。すべてのフィールドをデフォルトのままにすることができます。

```
#  vi /etc/ovirt-engine/extensions.d/example-authz.properties
```

例19.9 許可設定ファイルの例

```
ovirt.engine.extension.name = example-authz
ovirt.engine.extension.bindings.method = jbossmodule
```

```
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.ldap
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.ldap.AuthzExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authz
config.profile.file.1 = ../aaa/example.properties
```

10. 認証マッピング設定ファイルを確認します。設定プロファイルの場所は、LDAP 設定ファイルの場所と一致する必要があります。設定プロファイルの拡張名は、認証設定ファイルの **ovirt.engine.aaa.authn.mapping.plugin** 値と一致させる必要があります。すべてのフィールドをデフォルトのままにすることができます。

```
# vi /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

例19.10 認証マッピング設定ファイルの例

```
ovirt.engine.extension.name = example-http-mapping
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.mapping.MappingExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Mapping
config.mapAuthRecord.type = regex
config.mapAuthRecord.regex.mustMatch = true
config.mapAuthRecord.regex.pattern = ^(?<user>.*?)((\\\\(?<at>@)(?<suffix>.*?)@.*)|(?
<realm>@.*))$
config.mapAuthRecord.regex.replacement = ${user}${at}${suffix}
```

11. 設定ファイルの所有権および権限が適切であることを確認してください。

```
# chown ovirt:ovirt /etc/ovirt-engine/aaa/example.properties
```

```
# chown ovirt:ovirt /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

```
# chown ovirt:ovirt /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

```
# chown ovirt:ovirt /etc/ovirt-engine/extensions.d/example-authz.properties
```

```
# chmod 600 /etc/ovirt-engine/aaa/example.properties
```

```
# chmod 640 /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

```
# chmod 640 /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

```
# chmod 640 /etc/ovirt-engine/extensions.d/example-authz.properties
```

12. Apache サービスおよび **ovirt-engine** サービスを再起動します。

```
# systemctl restart httpd.service
# systemctl restart ovirt-engine.service
```

19.5. RED HAT SINGLE SIGN-ON のインストールおよび設定

Red HadSingle Sign-On を認証方法として使用するには、次のことを行う必要があります。

- Install Red Hat SSO.
- LDAP グループマッパーを設定します。
- Manager で Apache を設定します。
- OVN プロバイダーの認証情報を設定します。

注記

Red Hat SSO が設定されている場合、一度に使用できる認証プロトコルは 1 つだけであるため、以前の LDAP サインオンは機能しません。

19.5.1. Red Hat Single Sign-On のインストール

Red Hat Single Sign-On をインストールするには、ZIP ファイルをダウンロードして解凍するか、RPM ファイルを使用します。

Red Hat SSO インストール のインストール手順に従います

次の情報を準備します。

- **Open ID Connect** サーバーのパス/場所。
- 正しいリポジトリーのサブスクリプションチャネルです。
- 有効な Red Hat サブスクリプションのログイン認証情報。

19.5.2. LDAP グループマッパーの設定

1. 以下の情報とともに LDAP グループマッパーを追加します。
   - **Name**: ldapgroups
   - **Mapper Type**: group-ldap-mapper
   - **LDAP グループ DN**: ou=groups,dc=example,dc=com
   - **Group Object Classes**: groupofuniquenames (**LDAP サーバーの設定に応じてこのクラスを適合させます**)
   - **Membership LDAP Attribute**: uniquemember (**LDAP サーバーの設定に応じてこのクラスを適合させます**)
2. **Save** をクリックします。
3. **LDAP グループを KeyCloak に同期** をクリックします。

4. **User Federation Provider** ページの下部で、**Synchronize all users** をクリックします。
5. **Clients** タブの **Add Client** で、**Client ID** に **ovirt-engine** を追加し、**Root URL** にエンジンの URL を入力します。
6. **Client Protocol** を **openid-connect** に変更し、**Access Type** を **confidential** に変更します。
7. **Clients** タブの **Ovirt-engine** > **Advanced Settings** で、**Access Token Lifespan** の有効期間を延長します。
8. **https://rhvm.example.com:443/*** を有効なリダイレクト URI として追加します。
9. クライアントシークレットが生成され、認証情報タブで確認することができます。
10. **Create Mapper Protocol** の下の **Clients** タブで、以下の設定でマッパーを作成します。
    - **Name**: グループ
    - **Mapper Type**: グループメンバーシップ
    - **Token Claim Name**: グループ
    - **Full group path**: **ON**
    - **Add to ID token**: **ON**
    - **Add to access token**: **ON**
    - **Add to userinfo**: **ON**
11. **username** に **Builtin Protocol Mapper** を追加します。
12. **ovirt-engine**、**ovirt-app-api**、および **ovirt-app-admin** で必要なスコープを作成します。
13. 前の手順で作成したスコープを使用して、ovirt-engine クライアントのオプションのクライアントスコープを設定します。

19.5.3. Manager での Apache の設定

1. Manager で Apache を設定します。

```
# yum install mod_auth_openidc
```

2. **/etc/httpd/conf.d** に、以下の内容で新しい **httpd** 設定ファイル **ovirt-openidc.conf** を作成します。

```
LoadModule auth_openidc_module modules/mod_auth_openidc.so

OIDCProviderMetadataURL https://SSO.example.com/auth/realms/master/.well-
known/openid-configuration
OIDCSSLValidateServer Off

OIDCClientID ovirt-engine
OIDCClientSecret <client_SSO _generated_key>
OIDCRedirectURI https://rhvm.example.com/ovirt-engine/callback
OIDCDefaultURL https://rhvm.example.com/ovirt-engine/login?scope=ovirt-app-admin+ovirt-
```

```
app-portal+ovirt-ext%3Dauth%3Asequence-priority%3D%7E

# maps the prefered_username claim to the REMOTE_USER environment variable:

OIDCRemoteUserClaim <preferred_username>
OIDCCryptoPassphrase <random1234>

<LocationMatch ^/ovirt-engine/sso/(interactive-login-negotiate|oauth/token-http-auth)|^/ovirt-
engine/callback>
   <If "req('Authorization') !~ /^(Bearer|Basic)/i">

      Require valid-user
      AuthType openid-connect

      ErrorDocument 401 "<html><meta http-equiv=\"refresh\" content=\"0; url=/ovirt-
engine/sso/login-unauthorized\"/><body><a href=\"/ovirt-engine/sso/login-
unauthorized\">Here</a></body></html>"
   </If>
</LocationMatch>

OIDCOAuthIntrospectionEndpoint
https://SSO.example.com/auth/realms/master/protocol/openid-connect/token/introspect
OIDCOAuthSSLValidateServer    Off
OIDCOAuthIntrospectionEndpointParams token_type_hint=access_token
OIDCOAuthClientID ovirt-engine
OIDCOAuthClientSecret <client_SSO _generated_key>
OIDCOAuthRemoteUserClaim sub

<LocationMatch ^/ovirt-engine/(api$|api/)>
   AuthType oauth20
   Require valid-user
</LocationMatch>
```

3. 設定変更を保存するには、**httpd** および **ovirt-engine** を再起動します。

```
# systemctl restart httpd
# systemctl restart ovirt-engine
```

4. 以下の内容で、**/etc/ovirt-engine/extensions.d/** に **openidc-authn.properties** ファイルを作成します。

```
ovirt.engine.extension.name = openidc-authn
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.http.AuthnExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authn
ovirt.engine.aaa.authn.profile.name = openidchttp
ovirt.engine.aaa.authn.authz.plugin = openidc-authz
ovirt.engine.aaa.authn.mapping.plugin = openidc-http-mapping
config.artifact.name = HEADER
config.artifact.arg = OIDC_CLAIM_preferred_username
```

5. 以下の内容で、**/etc/ovirt-engine/extensions.d/** に **openidc-http-mapping.properties** ファイルを作成します。

```
ovirt.engine.extension.name = openidc-http-mapping
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.mapping.MappingExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Mapping
config.mapAuthRecord.type = regex
config.mapAuthRecord.regex.mustMatch = false
config.mapAuthRecord.regex.pattern = ^(?<user>.*?)((\\\\(?<at>@)(?<suffix>.*?)@.*)|(?
<realm>@.*))$
config.mapAuthRecord.regex.replacement = ${user}${at}${suffix}
```

6. 以下の内容で、**/etc/ovirt-engine/extensions.d/** に **openidc-authz.properties** ファイルを作成します。

```
ovirt.engine.extension.name = openidc-authz
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.http.AuthzExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authz
config.artifact.name.arg = OIDC_CLAIM_preferred_username
config.artifact.groups.arg = OIDC_CLAIM_groups
```

7. 以下の内容で、**/etc/ovirt-engine/engine.conf.d/** に **99-enable-external-auth.conf** ファイルを作成します。

```
ENGINE_SSO_ENABLE_EXTERNAL_SSO=true
ENGINE_SSO_EXTERNAL_SSO_LOGOUT_URI="${ENGINE_URI}/callback"
EXTERNAL_OIDC_USER_INFO_END_POINT=https://SSO.example.com/auth/realms/master
/protocol/openid-connect/userinfo
EXTERNAL_OIDC_TOKEN_END_POINT=https://SSO.example.com/auth/realms/master/prot
ocol/openid-connect/token
EXTERNAL_OIDC_LOGOUT_END_POINT=https://SSO.example.com/auth/realms/master/pr
otocol/openid-connect/logout
EXTERNAL_OIDC_CLIENT_ID=ovirt-engine
EXTERNAL_OIDC_CLIENT_SECRET="<client_SSO _generated_key>"
EXTERNAL_OIDC_HTTPS_PKI_TRUST_STORE="/etc/pki/java/cacerts"
EXTERNAL_OIDC_HTTPS_PKI_TRUST_STORE_PASSWORD=""
EXTERNAL_OIDC_SSL_VERIFY_CHAIN=false
EXTERNAL_OIDC_SSL_VERIFY_HOST=false
```

19.5.4. OVN の設定

Manager で **ovirt-ovn-provider** を設定した場合は、OVN プロバイダーの認証情報を設定する必要があります。

1. 以下の内容で **/etc/ovirt-provider-ovn/conf.d/** に **20-setup-ovirt-provider-ovn.conf** ファイルを作成します。ここで、**user1** は LDAP グループ **ovirt-administrator** に属し、**openidchttp** は **aaa-ldap-misc** 用に設定されたプロファイルです。

```
[OVIRT]
# ovirt-admin-user-name=user1@openidchttp
```

2. **ovirt-provider-ovn** を再起動します。

```
# systemctl restart ovirt-provider-ovn
```

3. 管理ポータルにログインし、**Administration → Providers** に移動して、**ovirt-provider-ovn** を選択し、**Edit** をクリックして ovn プロバイダーのパスワードを更新します。

19.6. ユーザーの承認

19.6.1. ユーザー認証モデル

Red Hat Virtualization は、次の 3 つのコンポーネントの組み合わせに基づいて承認制御を適用します。

- アクションを実行するユーザー
- 実行されているアクションのタイプ
- アクションが実行されるオブジェクト

19.6.2. ユーザーアクション

アクションを正常に実行するには、**user** は、アクションの対象となる **object** に対する適切な **permission** を持っている必要があります。各アクションの種類には、対応する **パーミッション** があります。

いくつかのアクションは、複数のオブジェクトに対して実行されます。たとえば、テンプレートを別のストレージドメインにコピーすると、テンプレートと宛先ストレージドメインの両方に影響します。アクションを実行するユーザーは、アクションが影響を与えるすべてのオブジェクトに対して適切な権限を持っている必要があります。

19.7. 管理ポータルからのユーザータスクの管理

19.7.1. ユーザーの追加と VM ポータルの権限付与

ユーザーを追加してロールおよび権限を割り当てる前に、ユーザーをすでに作成しておく必要があります。この手順で割り当てられたロールと権限により、ユーザーは VM ポータルにログインして仮想マシンの作成を開始することができます。この手順は、グループアカウントにも適用されます。

ユーザーの追加と VM ポータルの権限付与

1. ヘッダーバーで **Administration → Configure**をクリックして **Configure** ウィンドウを開きます。
2. **System Permissions** をクリックします。
3. **Add** をクリックして、**Add System Permission to User** ウィンドウを開きます。
4. **Search** でプロファイルを選択します。プロファイルは、検索するドメインです。検索テキストフィールドに名前または名前の一部を入力し、**GO** をクリックします。または、**GO** をクリックして、すべてのユーザーとグループのリストを表示します。
5. 適切なユーザーまたはグループのチェックボックスを選択します。

6. **Role to Assign** で、割り当てる適切なロールを選択します。**UserRole** ロールは、ユーザーアカウントに VM ポータルにログインするためのアクセス許可を付与します。

7. **OK** をクリックします。

VM ポータルにログインして、ユーザーアカウントにログインする権限があることを確認します。

19.7.2. ユーザー情報の表示

ユーザー情報の表示

1. **Administration → Users** をクリックして、承認されたユーザーの一覧を表示します。

2. ユーザーの名前をクリックして詳細ビューを開くと、通常は **General** タブにドメイン名、電子メール、ユーザーのステータスなどの一般情報が表示されます。

3. 他のタブでは、ユーザーのグループ、権限、割り当て、およびイベントを表示できます。

たとえば、ユーザーが属するグループを表示するには、**Directory Groups** タブをクリックします。

19.7.3. リソースでのユーザーパーミッションの表示

ユーザーには、特定のリソースまたはリソースの階層のパーミッションを割り当てることができます。各リソースに割り当てられたユーザーとそのパーミッションを表示できます。

リソースでのユーザーパーミッションの表示

1. リソースの名前を見つけ、クリックして詳細ビューを開きます。

2. **Permissions** タブをクリックして、割り当てられたユーザー、ユーザーのロール、および選択したリソースの継承されたパーミッションを一覧表示します。

19.7.4. ユーザーの削除

ユーザーアカウントが必要なくなったら、Red Hat Virtualization から削除します。

ユーザーの削除

1. **Administration → Users** をクリックして、承認されたユーザーの一覧を表示します。

2. 削除するユーザーを選択します。ユーザーが仮想マシンを実行していないことを確認します。

3. **Remove** をクリックしてから **OK** をクリックします。

ユーザーは Red Hat Virtualization から削除されますが、外部ディレクトリーからは削除されません。

19.7.5. ログインしたユーザーの表示

現在ログインしているユーザーを、セッション時間やその他の詳細とともに表示できます。**Administration → Active User Sessions** をクリックして、ログインしている各ユーザーの **Session DB ID**、**User Name**、**Authorization provider**、**User id**、**Source IP**、**Session Start Time**、および **Session Last Active Time** を表示します。

19.7.6. ユーザーセッションの終了

現在ログインしているユーザーのセッションを終了することができます。

ユーザーセッションの終了

1. **Administration → Active User Sessions** をクリックします。
2. 終了するユーザーセッションを選択します。
3. **セッションの終了** をクリックします。
4. **OK** をクリックします。

19.8. コマンドラインからのユーザータスクの管理

ovirt-aaa-jdbc-tool ツールを使用して、内部ドメインのユーザーアカウントを管理できます。このツールを使用して行った変更はすぐに反映され、**ovirt-engine** サービスを再起動する必要はありません。ユーザーオプションの完全なリストについては、**ovirt-aaa-jdbc-tooluser--help** を実行してください。このセクションでは、一般的な例を示します。

重要

Manager マシンにログインする必要があります。

19.8.1. 新しいユーザーの作成

新しいユーザーアカウントを作成できます。オプションの **--attribute** コマンドは、アカウントの詳細を指定します。オプションの完全なリストについては、**ovirt-aaa-jdbc-tool user add --help** を実行してください。

```
# ovirt-aaa-jdbc-tool user add test1 --attribute=firstName=John --attribute=lastName=Doe
adding user test1...
user added successfully
```

管理ポータルで新しく作成されたユーザーを追加し、ユーザーに適切なロールと権限を割り当てることができます。詳細は、「ユーザーの追加と VM ポータルの権限付与」 を参照してください。

19.8.2. ユーザーパスワードの設定

パスワードを作成できます。**--password-valid-to** の値を設定する必要があります。設定しないと、パスワードの有効期限はデフォルトで現在の時刻になります。日付形式は **yyyy-MM-dd HH:mm:ssX** です。この例では、**-0800** は GMT から 8 時間を引いたものを表します。その他のオプションについては、**ovirt-aaa-jdbc-tool user password-reset --help** を実行してください。

```
# ovirt-aaa-jdbc-tool user password-reset test1 --password-valid-to="2025-08-01 12:00:00-0800"
Password:
updating user test1...
user updated successfully
```

注記

デフォルトでは、内部ドメインのユーザーアカウントのパスワードポリシーには次の制限があります。

- 6 文字以上。
- パスワード変更中は、過去に使用した 3 つのパスワードを再度設定することはできません。

パスワードポリシーおよびその他のデフォルト設定の詳細は、**ovirt-aaa-jdbc-tool settings show** を実行してください。

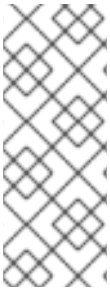
19.8.3. ユーザータイムアウトの設定

ユーザータイムアウト期間を設定できます。

```
# engine-config --set UserSessionTimeOutInterval=integer
```

19.8.4. ユーザーパスワードの事前暗号化

ovirt-engine-crypto-tool スクリプトを使用して、事前に暗号化されたユーザーパスワードを作成できます。このオプションは、スクリプトを使用してデータベースにユーザーとパスワードを追加する場合に役立ちます。



注記

パスワードは、暗号化された形式で Manager データベースに保存されます。すべてのパスワードを同じアルゴリズムで暗号化する必要があるため、**ovirt-engine-crypto-tool** スクリプトが使用されます。

パスワードがあらかじめ暗号化されている場合、パスワードの有効性テストは行えません。パスワードは、パスワード検証ポリシーに準拠していなくても受け入れられます。

1. 以下のコマンドを実行します。

   ```
   # /usr/share/ovirt-engine/bin/ovirt-engine-crypto-tool.sh pbe-encode
   ```

   スクリプトは、パスワードの入力を促します。

   また、**--password=file:file** オプションを使用すると、ファイルの先頭行として表示される 1 つのパスワードを暗号化することができます。このオプションは自動化に役立ちます。次の例では、**file** は暗号化用の単一のパスワードを含むテキストファイルです。

   ```
   # /usr/share/ovirt-engine/bin/ovirt-engine-crypto-tool.sh pbe-encode --password=file:file
   ```

2. **ovirt-aaa-jdbc-tool** スクリプトで、**--encrypted** オプションを使用して新しいパスワードを設定します。

   ```
   # ovirt-aaa-jdbc-tool user password-reset test1 --password-valid-to="2025-08-01 12:00:00-0800" --encrypted
   ```

3. 暗号化されたパスワードを入力して確認します。

```
Password:
Reenter password:
updating user test1...
user updated successfully
```

19.8.5. ユーザー情報の表示

詳細なユーザーアカウント情報を表示できます。

```
# ovirt-aaa-jdbc-tool user show test1
```

このコマンドにより、管理ポータルの**Administration → Users** 画面により多くの情報が表示されます。

19.8.6. ユーザー情報の編集

メールアドレスなどのユーザー情報を更新できます。

```
# ovirt-aaa-jdbc-tool user edit test1 --attribute=email=jdoe@example.com
```

19.8.7. ユーザーの削除

ユーザーアカウントを削除できます。

```
# ovirt-aaa-jdbc-tool user delete test1
```

管理ポータルからユーザーを削除します。詳細は、「ユーザーの削除」 を参照してください。

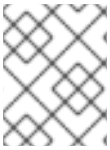
19.8.8. 内部管理ユーザーの無効化

engine-setup 中に作成された **admin@internal** ユーザーを含む、ローカルドメインのユーザーを無効にすることができます。デフォルトの **admin** ユーザーを無効にする前に、完全な管理者権限を持つ環境に少なくとも 1 人のユーザーがいることを確認してください。

内部管理ユーザーの無効化

1. Red Hat Virtualization Manager がインストールされているマシンにログインします。
2. **SuperUser** ロールを持つ別のユーザーが環境に追加されていることを確認してください。詳細は、「ユーザーの追加と VM ポータルの権限付与」 を参照してください。
3. デフォルトの **admin** ユーザーを無効にします。

```
# ovirt-aaa-jdbc-tool user edit admin --flag=+disabled
```



注記

無効になっているユーザーを有効にするには、**ovirt-aaa-jdbc-tool user edit username --flag=-disabled** を実行します。

19.8.9. グループの管理

ovirt-aaa-jdbc-tool ツールを使用して、内部ドメインのグループアカウントを管理できます。グループ

アカウントの管理は、ユーザーアカウントの管理に似ています。グループオプションの全リストは、**ovirt-aaa-jdbc-tool group --help** を実行してください。このセクションでは、一般的な例を示します。

グループの作成

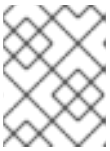
この手順では、グループアカウントを作成し、ユーザーをグループに追加し、グループの詳細を表示する方法を示します。

1. Red Hat Virtualization Manager がインストールされているマシンにログインします。
2. 新規グループを作成します。

```
# ovirt-aaa-jdbc-tool group add group1
```

3. ユーザーをグループに追加します。ユーザーが作成されている必要があります。

```
# ovirt-aaa-jdbc-tool group-manage useradd group1 --user=test1
```



注記

group-manage オプションの全リストは、**ovirt-aaa-jdbc-tool group-manage --help** を実行してください。

4. グループアカウントの詳細を表示します。

```
# ovirt-aaa-jdbc-tool group show group1
```

5. 新しく作成したグループを管理ポータルに追加し、グループに適切なロールと権限を割り当てます。グループ内のユーザーは、グループのロールおよび権限を継承します。詳細は、「ユーザーの追加と VM ポータルの権限付与」 を参照してください。

ネストされたグループの作成

この手順では、グループ内にグループを作成する方法を示します。

1. Red Hat Virtualization Manager がインストールされているマシンにログインします。
2. 最初のグループを作成します。

```
# ovirt-aaa-jdbc-tool group add group1
```

3. 2 番目のグループを作成します。

```
# ovirt-aaa-jdbc-tool group add group1-1
```

4. 2 番目のグループを最初のグループに追加します。

```
# ovirt-aaa-jdbc-tool group-manage groupadd group1 --group=group1-1
```

5. 管理ポータルに最初のグループを追加し、グループに適切なロールおよび権限を割り当てます。詳細は、「ユーザーの追加と VM ポータルの権限付与」 を参照してください。

19.8.10. ユーザーおよびグループのクエリー

クエリー モジュールを使用すると、ユーザーおよびグループの情報をクエリーできます。オプションの完全なリストについては、**ovirt-aaa-jdbc-tool query --help** を実行してください。

すべてのユーザーまたはグループアカウントの詳細の一覧表示

この手順では、すべてのアカウント情報を一覧表示する方法を示します。

1. Red Hat Virtualization Manager がインストールされているマシンにログインします。
2. アカウントの詳細を一覧表示します。
   - すべてのユーザーアカウントの詳細:

     ```
     # ovirt-aaa-jdbc-tool query --what=user
     ```

   - すべてのグループアカウントの詳細:

     ```
     # ovirt-aaa-jdbc-tool query --what=group
     ```

フィルターリングされたアカウントの詳細の一覧表示

この手順では、アカウント情報を一覧表示するときにフィルターを適用する方法を示します。

1. Red Hat Virtualization Manager がインストールされているマシンにログインします。
2. **--pattern** パラメーターを使用して、アカウントの詳細をフィルターリングします。
   - 文字 **j** で始まる名前でユーザーアカウントの詳細を一覧表示します。

     ```
     # ovirt-aaa-jdbc-tool query --what=user --pattern="name=j*"
     ```

   - 部門属性が **マーケティング** に設定されているグループを一覧表示します。

     ```
     # ovirt-aaa-jdbc-tool query --what=group --pattern="department=marketing"
     ```

19.8.11. アカウント設定の管理

デフォルトのアカウント設定を変更するには、**ovirt-aaa-jdbc-tool settings** モジュールを使用します。

アカウント設定の更新

この手順では、デフォルトのアカウント設定を更新する方法を示します。

1. Red Hat Virtualization Manager がインストールされているマシンにログインします。
2. 次のコマンドを実行して、使用可能なすべての設定を表示します。

   ```
   # ovirt-aaa-jdbc-tool settings show
   ```

3. 必要な設定を変更します。
   - この例では、すべてのユーザーアカウントのデフォルトのログインセッション時間を 60 分に更新します。デフォルト値は 10080 分です。

```
# ovirt-aaa-jdbc-tool settings set --name=MAX_LOGIN_MINUTES --value=60
```

- この例では、ユーザーアカウントがロックされる前に、ユーザーが実行できるログイン試行の失敗回数を更新します。デフォルト値は 5 です。

```
# ovirt-aaa-jdbc-tool settings set --name=MAX_FAILURES_SINCE_SUCCESS --value=3
```

注記

ロックされたユーザーアカウントを解除するには、**ovirt-aaa-jdbc-tool user unlock test1** を実行します。

19.9. 追加のローカルドメインの設定

デフォルトの **内部** ドメイン以外の追加のローカルドメインの作成もサポートされています。これは、**ovirt-engine-extension-aaa-jdbc** 拡張機能を使用して行うことができ、外部のディレクトリーサーバーを取り付けることなく複数のドメインを作成することができますが、エンタープライズ環境ではこのユースケースは一般的ではないでしょう。

追加で作成されたローカルドメインは、標準の Red Hat Virtualization アップグレード中に自動的にアップグレードされることはなく、将来のリリースごとに手動でアップグレードする必要があります。追加のローカルドメインの作成と、ドメインのアップグレード方法に関する詳細は、**/usr/share/doc/ovirt-engine-extension-aaa-jdbc-version/README.admin** の README ファイルを参照してください。

第20章 クォータとサービスレベル契約ポリシー

20.1. クォータの概要

クォータは、Red Hat Virtualization で提供されるリソース制限ツールです。クォータは、ユーザー権限によって設定された制限のレイヤーの上にある制限のレイヤーと考えることができます。

クォータはデータセンターオブジェクトです。

Quota を使用すると、Red Hat Virtualization 環境の管理者は、メモリー、CPU、およびストレージへのユーザーアクセスを制限できます。クォータは、管理者がユーザーに割り当てることができるメモリーリソースとストレージリソースを定義します。その結果、ユーザーは自分に割り当てられたリソースのみを利用できます。クォータリソースが使い果たされると、Red Hat Virtualization はそれ以上のユーザーアクションを許可しません。

クォータには 2 つの異なる種類があります。

表20.1 2 種類のクォータ

クォータタイプ	定義
実行時クォータ	このクォータは、CPU やメモリーなどのランタイムリソースの消費を制限するものです。
ストレージクォータ	このクォータは、利用可能なストレージの量を制限します。

SELinux などのクォータには 3 つのモードがあります。

表20.2 クォータモード

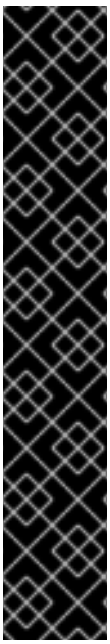
クォータモード	機能
Enforced	このモードでは、監査モードで設定したクォータが有効になり、クォータの影響を受けるグループまたはユーザーにリソースが制限されます。
Audit	このモードは、ユーザーをブロックせずにクォータ違反をログに記録し、クォータのテストに使用できます。監査モードでは、ランタイムクォータの量と、影響を受けるユーザーが利用できるストレージクォータの量を増減できます。
Disabled	このモードは、クォータによって定義されたランタイムとストレージの制限をオフにします。

ユーザーが仮想マシンを実行しようとすると、仮想マシンの仕様が、該当するクォータに設定されているストレージ許容量およびランタイム許容量と比較されます。

仮想マシンを起動すると、クォータの対象となる実行中のすべての仮想マシンの集約リソースがクォータで定義された許容量を超える場合、Manager は仮想マシンの実行を拒否します。

ユーザーが新しいディスクを作成すると、要求されたディスクサイズが、該当するクォータでカバーされる他のすべてのディスクの合計ディスク使用量に追加されます。新しいディスクが、クォータで許可されている量を超える合計ディスク使用量を取得した場合、ディスクの作成は失敗します。

クォータでは、同じハードウェアのリソース共有が可能です。ハードとソフトのしきい値をサポートします。管理者は、クォータを使用してリソースにしきい値を設定できます。これらのしきい値は、ユーザーの観点からは、そのリソースの 100% 使用率として表示されます。お客様が予期せずに、このしきい値を超えた場合に発生する障害を防ぐため、インターフェイスは、しきい値を一時的に超えることができる猶予量をサポートしています。しきい値を超えると、お客様に警告が送信されます。



重要

クォータは、仮想マシンの実行時に制限をいくつか課します。これらの制限を無視すると、仮想マシンと仮想ディスクを使用できない状況が発生する可能性があります。

クォータが強制モードで実行している場合、クォータが割り当てられていない仮想マシンとディスクは使用できません。

仮想マシンの電源をオンにするには、その仮想マシンにクォータを割り当てる必要があります。

仮想マシンのスナップショットを作成するには、仮想マシンに関連付けられているディスクにクォータが割り当てられている必要があります。

仮想マシンからテンプレートを作成する場合、テンプレートで使用するクォータを選択するように求められます。これにより、テンプレート (およびテンプレートから作成される将来のすべてのマシン) が、テンプレートの生成元の仮想マシンおよびディスクとは異なるクォータを消費するように設定できます。

20.2. 共有クォータおよび個別定義されたクォータ

SuperUser 権限を持つユーザーは、個々のユーザーのクォータまたはグループのクォータを作成できます。

Active Directory ユーザーに対してグループクォータを設定することができます。10 人のユーザーのグループに 1 TB のストレージのクォータが与えられている場合、10 人のユーザーのうち 1 人がテラバイト全体を満たしてしまうと、グループ全体がクォータを超過してしまい、10 人のユーザーのうち誰も自分のグループに関連するストレージを使用することができなくなるのです。

個人ユーザーの割り当ては、個人に対してのみ設定されます。個々のユーザーが自分のストレージまたはランタイムクォータをすべて使い切ると、ユーザーはクォータを超え、ユーザーは自分のクォータに関連付けられたストレージを使用できなくなります。

20.3. クォータアカウンティング

クォータがコンシューマーまたはリソースに割り当てられると、そのコンシューマーによる、またはストレージ、vCPU、またはメモリーに関連するリソースに対する各アクションにより、クォータの消費またはクォータの解放が発生します。

クォータは、ユーザーのリソースへのアクセスを制限する上限として機能するため、クォータの計算は、ユーザーの実際の現在の使用とは異なる場合があります。クォータは、現在の使用量ではなく、最大増加の可能性に対して計算されます。

例20.1 アカウンティングの例

> ユーザーは、1 つの vCPU と 1024 MB のメモリーを備えた仮想マシンを実行します。このアクションは、そのユーザーに割り当てられた 1 つの vCPU と 1024 MB のクォータを消費します。仮想マシンが停止すると、1 つの vCPU と 1024 MB の RAM が解放され、そのユーザーに割り当てられたクォータに戻ります。実行時のクォータ消費は、コンシューマーの実際の実行時にのみ考慮されます。
>
> あるユーザーが 10GB の仮想シンプロビジョンディスクを作成します。実際のディスク使用量は、そのディスクのうち 3GB しか使用されていないことを示している場合があります。ただし、クォータ消費量は 10 GB であり、そのディスクの最大拡張可能性です。

20.4. データセンターにおけるクォータモードの有効化および変更

この手順では、データセンターのクォータモードを有効または変更します。クォータを定義する前に、クォータモードを選択する必要があります。この手順の手順を実行するには、管理ポータルにログインする必要があります。

監査 モードを使用して、クォータをテストし、期待通りに動作することを確認します。クォータを作成または変更するには、クォータが **Audit** モードである必要はありません。

データセンターでのクォータの有効化および変更

1. データセンターの **Compute** → **Data Centers** をクリックして、データセンターを選択します。
2. **Edit** をクリックします。
3. **Quota Mode** ドロップダウンリストで、クォータモードを **Enforced** に変更します。
4. **OK** をクリックします。

テスト中にクォータモードを **Audit** に設定した場合、クォータ設定を有効にするには、クォータモードを **Enforced** に変更する必要があります。

20.5. 新しいクォータポリシーの作成

監査モードまたは強制モードのいずれかでクォータモードを有効にしました。データセンターのリソース使用量を管理するためのクォータポリシーを定義する必要があります。

新しいクォータポリシーの作成

1. **Administration** → **Quota** をクリックします。
2. **Add** をクリックします。
3. **Name** フィールドおよび **Description** フィールドに入力します。
4. **Data Center** を選択します。
5. **Memory & CPU**セクションで、緑のスライダーを使用して **Cluster Threshold** を設定します。
6. **Memory & CPU**セクションで、青のスライダーを使用して **Cluster Grace** を設定します。
7. **All Clusters** または **Specific Clusters** ラジオボタンをクリックします。**Specific Clusters** を選択した場合は、クォータポリシーを追加するクラスターのチェックボックスをオンにします。
8. **Edit** をクリックして **Edit Quota** ウィンドウを開きます。

a. **Memory** フィールドで、**Unlimited** ラジオボタン (クラスター内のメモリーリソースを無制限に使用できるようにする) を選択するか、**limit to** ラジオボタンを選択して、このクォータで設定されるメモリーの量を設定します。**limit to** ラジオボタンを選択した場合は、**MB** フィールドにメモリークォータをメガバイト (MB) 単位で入力します。

b. **CPU** フィールドで、**Unlimited** ラジオボタンまたは **limit to** ラジオボタンのいずれかを選択して、このクォータで設定される CPU の量を設定します。**limit to** ラジオボタンを選択した場合は、**vCpus** フィールドに vCPU の数を入力します。

c. **Edit Quota** ウィンドウで **OK** をクリックします。

9. **ストレージ** セクションで、緑色のスライダーを使用して **ストレージのしきい値** を設定します。

10. **Storage** セクションで、青のスライダーを使用して **ストレージの猶予** を設定します。

11. **All Storage Domains** または **Specific Storage Domains** ラジオボタンをクリックします。**Specific Storage Domains** を選択した場合は、クォータポリシーを追加するストレージドメインのチェックボックスをオンにします。

12. **Edit** をクリックして **Edit Quota** ウィンドウを開きます。

a. **Storage Quota** フィールドで、**Unlimited** ラジオボタン (ストレージを無制限に使用できるようにする) または **limit to** ラジオボタンを選択して、クォータがユーザーを制限するストレージの量を設定します。**limit to** ラジオボタンを選択した場合は、**GB** フィールドにストレージクォータサイズをギガバイト (GB) で入力します。

b. **Edit Quota** ウィンドウで **OK** をクリックします。

13. **New Quota** ウィンドウで **OK** をクリックします。

20.6. クォータしきい値の設定の説明

表20.3 クォータしきい値および猶予期間

設定	定義
クラスターしきい値	データセンターごとに利用可能なクラスターリソースの量。
クラスターの猶予	データセンターのクラスターしきい値を使い果たした後、データセンターで使用可能なクラスターの量。
ストレージのしきい値	データセンターごとに利用可能なストレージリソースの量。
ストレージの猶予	データセンターのストレージしきい値を使い切った後のデータセンターで使用可能なストレージの量。

クォータが 20% の猶予で 100 GB に設定されている場合、使用者は 120 GB のストレージを使用した後、ストレージの使用をブロックされます。同じクォータのしきい値が 70% に設定されている場合、使用者は 70 GB のストレージ消費量を超えると警告を受け取ります (ただし、120 GB のストレージ消費量に達するまでストレージを消費できます)。 しきい値および猶予の両方は、クォータを基準にして

設定されます。しきい値はソフト制限と考えることができ、それを超えると警告が生成されます。猶予はハード制限として考えられ、それを超えると、これ以上ストレージリソースを消費できなくなります。

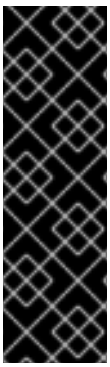
20.7. オブジェクトへのクォータの割り当て

仮想マシンへのクォータの割り当て

1. **Compute → Virtual Machines** をクリックし、仮想マシンを選択します。
2. **Edit** をクリックします。
3. **Quota** ドロップダウンリストから、仮想マシンが消費するクォータを選択します。
4. **OK** をクリックします。

仮想ディスクへのクォータの割り当て

1. **Compute → Virtual Machines**をクリックします。
2. 仮想マシン名をクリックして、詳細ビューに移動します。
3. **Disks** タブをクリックし、クォータに関連付ける予定のディスクを選択します。
4. **Edit** をクリックします。
5. **Quota** ドロップダウンリストから、仮想ディスクが消費するクォータを選択します。
6. **OK** をクリックします。



重要

仮想マシンを機能させるには、仮想マシンに関連付けられているすべてのオブジェクトに対してクォータを選択する必要があります。仮想マシンに関連付けられているオブジェクトのクォータを選択しないと、仮想マシンは機能しません。この状況で Manager が出力するエラーは一般的なものであり、仮想マシンに関連付けられているすべてのオブジェクトにクォータを関連付けていないため、エラーが出力されたかどうかを判断するのは困難です。クォータが割り当てられていない仮想マシンのスナップショットを作成することはできません。仮想ディスクにクォータが割り当てられていない仮想マシンのテンプレートを作成することはできません。

20.8. クォータを使用してユーザーごとにリソースを制限する

この手順では、クォータを使用して、ユーザーがアクセスできるリソースを制限する方法について説明します。

クォータへのユーザーの割り当て

1. **Administration → Quota** をクリックします。
2. ターゲットクォータの名前をクリックし、詳細ビューを開きます。
3. **Consumers** タブをクリックします。
4. **Add** をクリックします。

5. **Search** フィールドに、クォータに関連付けるユーザーの名前を入力します。
6. **GO** をクリックします。
7. ユーザー名の横にあるチェックボックスを選択します。
8. **OK** をクリックします。

しばらくすると、ユーザーは詳細ビューの **Consumers** タブに表示されます。

20.9. クォータの編集

この手順では、既存のクォータを変更する方法について説明します。

クォータの編集

1. **Administration → Quota** をクリックして、クォータを選択します。
2. **Edit** をクリックします。
3. 必要に応じてフィールドを編集します。
4. **OK** をクリックします。

20.10. クォータの削除

この手順では、クォータを削除する方法について説明します。

クォータの削除

1. **Administration → Quota** をクリックして、クォータを選択します。
2. **Remove** をクリックします。
3. **OK** をクリックします。

20.11. サービスレベルアグリーメントポリシーの実施

この手順では、サービスレベルアグリーメントの CPU 機能を設定する方法について説明します。

サービスレベルアグリーメントの CPU ポリシーの設定

1. **Compute → Virtual Machines**をクリックします。
2. **New** をクリックするか、仮想マシンを選択して **Edit** をクリックします。
3. **Resource Allocation** タブをクリックします。
4. **CPU Shares** を指定します。可能なオプションは、**Low**、**Medium**、**High**、**Custom**、および **Disabled** です。**High** に設定された仮想マシンは、**Medium** の 2 倍の共有を受け取り、**Medium** に設定された仮想マシンは、**Low** に設定された仮想マシンの 2 倍の共有を受け取ります。**Disabled** は、VDSM が共有の払い出しを決定するために古いアルゴリズムを使用するように指示します。通常、この条件で払い出される共有数は 1020 です。

ユーザーの CPU 消費は、設定したポリシーで制御されるようになりました。

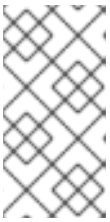
第21章 イベント通知

21.1. 管理ポータルでのイベント通知の設定

Red Hat Virtualization Manager は、Red Hat Virtualization Manager が管理する環境で特定のイベントが発生したときに、指定されたユーザーに電子メールで通知できます。この機能を使用するには、メッセージを配信するようにメール転送エージェントを設定する必要があります。管理ポータルから設定できるのは、メール通知のみです。SNMP トラップは Manager マシンで設定する必要があります。

イベント通知の設定

1. Manager からの自動メッセージを受け入れ、それらを配布リストに配信できる電子メールサーバーにアクセスできることを確認してください。
2. **Administration → Users** をクリックして、ユーザーを選択します。
3. ユーザーの **ユーザー名** をクリックすると、詳細ページが表示されます。
4. **Event Notifier** タブで、**Manage Events** をクリックします。
5. イベントを表示するには、**Expand All** ボタン、または件名別の展開ボタンを使用します。
6. 適切なチェックボックスを選択します。
7. **Mail Recipient** 欄にメールアドレスを入力します。



注記

電子メールアドレスは、テキストメッセージの電子メールアドレス (たとえば、**1234567890@carrierdomainname.com**)、または電子メールアドレスとテキストメッセージの電子メールアドレスを含む電子メールグループアドレスにすることができます。

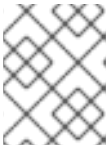
8. **OK** をクリックします。
9. Manager マシンで、**ovirt-engine-notifier.conf** を **90-email-notify.conf** という名前の新しいファイルにコピーします。

```
# cp /usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf /etc/ovirt-engine/notifier/notifier.conf.d/90-email-notify.conf
```

10. **90-email-notify.conf** を編集し、**EMAIL Notifications** セクション以外を削除します。
11. 次の例のように、正しい電子メール変数を入力します。このファイルは、元の **ovirt-engine-notifier.conf** ファイルの値を上書きします。

```
#---------------------#
# EMAIL Notifications #
#---------------------#

# The SMTP mail server address. Required.
MAIL_SERVER=myemailserver.example.com

# The SMTP port (usually 25 for plain SMTP, 465 for SMTP with SSL, 587 for SMTP with
```

```
TLS)
MAIL_PORT=25

# Required if SSL or TLS enabled to authenticate the user. Used also to specify 'from' user
address if mail server
# supports, when MAIL_FROM is not set. Address is in RFC822 format
MAIL_USER=

# Required to authenticate the user if mail server requires authentication or if SSL or TLS is
enabled
SENSITIVE_KEYS="${SENSITIVE_KEYS},MAIL_PASSWORD"
MAIL_PASSWORD=

# Indicates type of encryption (none, ssl or tls) should be used to communicate with mail
server.
MAIL_SMTP_ENCRYPTION=none

# If set to true, sends a message in HTML format.
HTML_MESSAGE_FORMAT=false

# Specifies 'from' address on sent mail in RFC822 format, if supported by mail server.
MAIL_FROM=rhevm2017@example.com

# Specifies 'reply-to' address on sent mail in RFC822 format.
MAIL_REPLY_TO=

# Interval to send smtp messages per # of IDLE_INTERVAL
MAIL_SEND_INTERVAL=1

# Amount of times to attempt sending an email before failing.
MAIL_RETRIES=4
```



注記

その他のオプションについては、**/etc/ovirt-engine/notifier/notifier.conf.d/README** を参照してください。

12. **ovirt-engine-notifier** サービスを有効にして再起動し、行った変更をアクティブにします。

```
# systemctl daemon-reload
# systemctl enable ovirt-engine-notifier.service
# systemctl restart ovirt-engine-notifier.service
```

指定されたユーザーは、Red Hat Virtualization 環境のイベントに基づいて電子メールを受け取るようになりました。選択されたイベントは、そのユーザーの **Event Notifier** タブに表示されます。

21.2. 管理ポータルでのイベント通知のキャンセル

ユーザーが不要な電子メール通知を設定していて、それらをキャンセルしたいと考えています。

イベント通知の取り消し

1. **Administration → Users** をクリックします。

2. ユーザーの **User Name** をクリックし、詳細ビューを開きます。
3. **Event Notifier** タブをクリックして、ユーザーが電子メール通知を受け取るイベントを一覧表示します。
4. **Manage Events** をクリックします。
5. イベントを表示するには、**Expand All** ボタン、または件名別の展開ボタンを使用します。
6. 該当するチェックボックスをオフにすると、そのイベントの通知が解除されます。
7. **OK** をクリックします。

21.3. OVIRT-ENGINE-NOTIFIER.CONF のイベント通知のパラメーター

イベント通知機能の設定ファイルは、**/usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf** にあります。

表21.1 ovirt-engine-notifier.conf variables

変数名	デフォルト	備考
SENSITIVE_KEYS	none	ログに記録されないキーのコンマ区切りのリスト。
JBOSS_HOME	/opt/rh/eap7/root/usr/share/wildfly/	Manager が使用する JBoss アプリケーションサーバーの場所。
ENGINE_ETC	/etc/ovirt-engine	Manager が使用する **etc** ディレクトリーの場所。
ENGINE_LOG	/var/log/ovirt-engine	Manager が使用する **logs** ディレクトリーの場所。
ENGINE_USR	/usr/share/ovirt-engine	Manager が使用する **usr** ディレクトリーの場所。
ENGINE_JAVA_MODULEPATH	${ENGINE_USR}/modules	JBoss モジュールが追加されるファイルパス。
NOTIFIER_DEBUG_ADDRESS	none	通知機能が使用する Java 仮想マシンのリモートデバッグを実行するために使用できるマシンのアドレス。
NOTIFIER_STOP_TIME	30	サービスがタイムアウトするまでの時間 (秒単位)。
NOTIFIER_STOP_INTERVAL	1	タイムアウトカウンターをインクリメントする時間 (秒)。

変数名	デフォルト	備考
INTERVAL_IN_SECONDS	120	サブスクライバーにメッセージをディスパッチするインスタンス間の間隔 (秒単位)。
IDLE_INTERVAL	30	低優先度タスクが実行される間隔 (秒単位)。
DAYS_TO_KEEP_HISTORY	0	この変数は、ディスパッチされたイベントが履歴テーブルに保持される日数を設定します。この変数が設定されていない場合、イベントは履歴テーブルに無期限に残ります。
FAILED_QUERIES_NOTIFICATION_THRESHOLD	30	通知メールが送信された後に失敗したクエリーの数。通知メールは、通知のフェッチに最初に失敗した後、この変数で指定された失敗の数に到達するたびに 1 回送信されます。**0** または **1** を指定した場合は、失敗するたびに電子メールが送信されます。
FAILED_QUERIES_NOTIFICATION_RECIPIENTS	none	通知メールの送信先となる受信者のメールアドレス。メールアドレスはコンマで区切る必要があります。この項目は、**FILTER** 変数によって非推奨とされました。
DAYS_TO_SEND_ON_STARTUP	0	通知機能の開始時に処理および送信される古いイベントの日数。
FILTER	exclude:*	電子メール通知のトリガーと受信者を決定するために使用されるアルゴリズム。この変数の値は、**include** または **exclude**、event、および recipient の組み合わせから設定される。たとえば、**include:VDC_START(smtp:mail@example.com) ${FILTER}** ようになります。
MAIL_SERVER	none	SMTP メールサーバーアドレス。必須。

変数名	デフォルト	備考
MAIL_PORT	25	通信に使用されるポート。使用できる値には、プレーン SMTP の場合は **25**、SSL を使用した SMTP の場合は **465**、および TLS を使用した SMTP の場合は **587** が含まれます。
MAIL_USER	none	ユーザー認証のために SSL が有効な場合は、この変数を設定する必要があります。この変数は、MAIL_FROM 変数が設定されていない場合に from ユーザーアドレスを指定するためにも使用されます。一部のメールサーバーでは、この機能をサポートしていません。アドレスは RFC822 形式です。
SENSITIVE_KEYS	${SENSITIVE_KEYS},MAIL_PASSWORD	メールサーバーで認証が必要な場合、または SSL または TLS が有効になっている場合は、ユーザーを認証するために必要です。
MAIL_PASSWORD	none	メールサーバーで認証が必要な場合、または SSL または TLS が有効になっている場合は、ユーザーを認証するために必要です。
MAIL_SMTP_ENCRYPTION	none	通信に使用する暗号の種類を指定します。可能な値は **none**、**ssl**、**tls** です。
HTML_MESSAGE_FORMAT	false	この変数が **true** に設定されている場合、メールサーバーは HTML フォーマットでメッセージを送信します。
MAIL_FROM	none	この変数は、メールサーバーでサポートされている場合、RFC822 形式で送信者アドレスを指定します。
MAIL_REPLY_TO	none	この変数は、メールサーバーでサポートされている場合、送信メールの返信先アドレスを RFC822 形式で指定します。
MAIL_SEND_INTERVAL	1	IDLE_INTERVAL ごとに送信される SMTP メッセージの数

変数名	デフォルト	備考
MAIL_RETRIES	4	失敗する前に電子メールの送信を試行する回数。
SNMP_MANAGER	none	SNMP マネージャーとして機能するマシンの IP アドレスまたは完全修飾ドメイン名。エントリーはスペースで区切る必要があり、ポート番号を含めることができます。たとえば、**manager1.example.com manager2.example.com:164** です。
SNMP_COMMUNITY	public	デフォルトの SNMP コミュニティー。
SNMP_OID	1.3.6.1.4.1.2312.13.1.1	アラートのデフォルトのトラップオブジェクト識別子。この OID が定義されると、すべてのトラップタイプがイベント情報とともに SNMP マネージャーに送信されます。デフォルトのトラップを変更すると、生成されたトラップが Manager の管理情報ベースに準拠できなくなることに注意してください。
ENGINE_INTERVAL_IN_SECONDS	300	Manager がインストールされているマシンを監視する間隔 (秒単位)。間隔は、監視が完了した時点から測定されます。
ENGINE_MONITOR_RETRIES	3	通知機能が、障害後の指定された間隔で Manager がインストールされているマシンのステータスを監視しようとする回数。
ENGINE_TIMEOUT_IN_SECONDS	30	通知機能が障害後に指定された間隔で Manager がインストールされているマシンの状況を監視しようとするまで待機する時間 (秒単位)。
IS_HTTPS_PROTOCOL	false	JBoss がセキュアモードで実行されている場合、このエントリーは **true** に設定する必要があります。

変数名	デフォルト	備考
SSL_PROTOCOL	TLS	SSL が有効な場合に JBoss Configuration コネクターが使用するプロトコル。
SSL_IGNORE_CERTIFICATE_ERRORS	false	JBoss がセキュアモードで実行しており、SSL エラーを無視する場合は、この値を **true** に設定する必要があります。
SSL_IGNORE_HOST_VERIFICATION	false	JBoss がセキュアモードで実行しており、ホスト名の検証が無視される場合は、この値を **true** に設定する必要があります。
REPEAT_NON_RESPONSIVE_NOTIFICATION	false	この変数は、Manager がインストールされているマシンが応答しない場合に、繰り返し失敗メッセージをサブスクライバーに送信するかどうかを指定します。
ENGINE_PID	/var/lib/ovirt-engine/ovirt-engine.pid	Manager の PID のパスとファイル名です。

21.4. SNMP トラップを送信するための RED HAT VIRTUALIZATION MANAGER の設定

Simple Network Management Protocol トラップを 1 つ以上の外部 SNMP マネージャーに送信するように Red Hat Virtualization Manager を設定します。SNMP トラップには、システムイベント情報が含まれています。これらは、Red Hat Virtualization 環境を監視するのに使用されます。SNMP マネージャーに送信されるトラップの数とタイプは、Red Hat Virtualization Manager 内で定義できます。

この手順では、トラップを受信するように 1 つ以上の外部 SNMP マネージャーを設定し、以下の詳細があることを前提としています。

- SNMP マネージャーとして機能するマシンの IP アドレスまたは完全修飾ドメイン名。必要に応じて、Manager がトラップ通知を受信するポートを決定します。デフォルトでは UDP ポート 162 になります。
- SNMP コミュニティー。複数の SNMP マネージャーが単一のコミュニティーに属することができます。管理システムとエージェントは、同じコミュニティー内にある場合にのみ通信できます。デフォルトのコミュニティーは **public** です。
- アラートのトラップオブジェクト識別子。Red Hat Virtualization Manager は、デフォルトの OID 1.3.6.1.4.1.2312.13.1.1 を提供します。この OID が定義されると、すべてのトラップタイプがイベント情報とともに SNMP マネージャーに送信されます。デフォルトのトラップを変更すると、生成されたトラップが Manager の管理情報ベースに準拠できなくなることに注意してください。

注記

Red Hat Virtualization Manager は、**/usr/share/doc/ovirt-engine/mibs/OVIRT-MIB.txt** および **/usr/share/doc/ovirt-engine/mibs/REDHAT-MIB.txt** に管理情報ベースを提供します。SNMP マネージャーに MIB を読み込ませてから作業を進めます。

デフォルトの SNMP 設定値は、Manager のイベント通知デーモン設定ファイル **/usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf** にあります。次の手順で概説する値は、このファイルで提供されているデフォルト値またはサンプル値に基づいています。**ovirt-engine-notifier.conf** ファイルを編集して、システムのアップグレード後も設定オプションが保持されるように、オーバーライドファイルを定義することを推奨します。

Manager での SNMP トラップの設定

1. Manager で SNMP 設定ファイルを作成します。

```
# vi /etc/ovirt-engine/notifier/notifier.conf.d/20-snmp.conf
```

2. SNMP マネージャー、SNMP コミュニティー、および OID を次の形式で指定します。

```
SNMP_MANAGERS="manager1.example.com manager2.example.com:162"
SNMP_COMMUNITY=public
SNMP_OID=1.3.6.1.4.1.2312.13.1.1
```

3. SNMP マネージャーに送信するイベントを定義します。

例21.1 イベントの例

すべてのイベントをデフォルトの SNMP プロファイルに送信します。

```
FILTER="include:*(snmp:) ${FILTER}"
```

重要度 **ERROR** または **ALERT** のすべてのイベントをデフォルトの SNMP プロファイルに送信します。

```
FILTER="include:*:ERROR(snmp:) ${FILTER}"
```

```
FILTER="include:*:ALERT(snmp:) ${FILTER}"
```

VDC_START のイベントを指定された電子メールアドレスに送信します。

```
FILTER="include:VDC_START(snmp:mail@example.com) ${FILTER}"
```

VDC_START 以外のすべてのイベントをデフォルトの SNMP プロファイルに送信します。

```
FILTER="exclude:VDC_START include:*(snmp:) ${FILTER}"
```

これは、**ovirt-engine-notifier.conf** で定義されているデフォルトのフィルターです。このフィルターを無効にしないか、オーバーライドフィルターを適用しない場合、通知は送信されません。

```
FILTER="exclude:*"
```

VDC_START は、使用可能な監査ログメッセージの例です。監査ログメッセージの完全なリストは、**/usr/share/doc/ovirt-engine/AuditLogMessages.properties** にあります。または、SNMP マネージャー内で結果をフィルター処理します。

4. ファイルを保存します。

5. **ovirt-engine-notifier** サービスを開始し、このサービスが起動時に開始することを確認します。

```
# systemctl start ovirt-engine-notifier.service
# systemctl enable ovirt-engine-notifier.service
```

SNMP マネージャーをチェックして、トラップを受け取っていることを確認します。

注記

通知サービスを実行するには、**SNMP_MANAGERS**、**MAIL_SERVER**、またはその両方を **/usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf** またはオーバーライドファイルで適切に定義する必要があります。

第22章 ユーティリティー

22.1. OVIRT エンジンの名前変更ツール

22.1.1. oVirt エンジンの名前変更ツール

engine-setup コマンドをクリーンな環境で実行すると、このコマンドは、セットアッププロセス中に提供された Manager の完全修飾ドメイン名を使用する多数の証明書およびキーを生成します。Manager の完全修飾ドメイン名を後で変更する必要がある場合 (たとえば、Manager をホストしているマシンを別のドメインに移行したため)、完全修飾ドメイン名のレコードを更新して、新しい名前を反映する必要があります。**ovirt-engine-rename** コマンドは、この作業を自動化します。

ovirt-engine-rename コマンドは、次の場所にある Manager の完全修飾ドメイン名のレコードを更新します。

- /etc/ovirt-engine/engine.conf.d/10-setup-protocols.conf
- /etc/ovirt-engine/isouploader.conf.d/10-engine-setup.conf
- /etc/ovirt-engine/logcollector.conf.d/10-engine-setup.conf
- /etc/pki/ovirt-engine/cert.conf
- /etc/pki/ovirt-engine/cert.template
- /etc/pki/ovirt-engine/certs/apache.cer
- /etc/pki/ovirt-engine/keys/apache.key.nopass
- /etc/pki/ovirt-engine/keys/apache.p12

警告

ovirt-engine-rename コマンドは、Manager が実行している Web サーバーの新しい証明書を作成しますが、Manager または認証局の証明書には影響しません。このため、特に Red Hat Enterprise Virtualization 3.2 以前からアップグレードされた環境では、**ovirt-engine-rename** コマンドの使用に伴うリスクがあります。したがって、可能な場合は、**engine-cleanup** および **engine-setup** を実行して、Manager の完全修飾ドメイン名を変更することが推奨されます。

警告

アップグレードプロセス中、古いホスト名は解決可能である必要があります。oVirt Engine RenameTool が失敗して **[ERROR] Host name is not valid: <OLD FQDN> did not resolve into an IP address** が発生した場合は、古いホスト名を **/etc/hosts** ファイルに追加し、oVirt Engine Rename Tool を使用します。次に、**/etc/hosts** ファイルから古いホスト名を削除します。

22.1.2. oVirt Engine Rename コマンドの構文

ovirt-engine-rename コマンドの基本的な構文は次のとおりです。

```
# /usr/share/ovirt-engine/setup/bin/ovirt-engine-rename
```

このコマンドは、次のオプションも受け入れます。

--newname=[new name]
ユーザーの操作なしで、Manager の新しい完全修飾ドメイン名を指定できます。

--log=[file]
名前変更操作のログが書き込まれるファイルのパスおよび名前を指定できます。

--config=[file]
名前変更操作にロードする設定ファイルのパスおよびファイル名を指定できます。

--config-append=[file]
名前変更操作に追加する設定ファイルのパスおよびファイル名を指定できます。このオプションを使用して、既存の応答ファイルのパスおよびファイル名を指定し、名前変更操作を自動化できます。

--generate-answer=[file]
回答と **ovirt-engine-rename** コマンドで変更された値が記録されるファイルのパスおよびファイル名を指定できます。

22.1.3. oVirt Engine Rename Tool を使って Manager の名前を変更

ovirt-engine-rename コマンドを使用して、Manager の完全修飾ドメイン名 (FQDN) のレコードを更新できます。

重要

ovirt-engine-rename コマンドでは、**imageio-proxy** または **websocket-proxy** などの SSL 証明書は更新されません。これらは、**ovirt-engine-rename** を実行した後に手動で更新する必要があります。以下の SSL 証明書の更新 を参照してください。

このツールは、Manager がローカル ISO またはデータストレージドメインを提供しているかどうかを確認します。その場合、ツールは、操作を続行する前に、ストレージに接続されている仮想マシンまたはストレージドメインをイジェクト、シャットダウン、またはメンテナンスモードにするようにユーザーに促します。これにより、仮想マシンが仮想ディスクとの接続を失うことがなくなり、名前の変更プロセス中に ISO ストレージドメインが接続を失うのを防ぐことができます。

oVirt エンジンの名前変更ツールの使用

1. 新しい FQDN のすべての DNS およびその他の関連レコードを準備します。
2. DHCP を使用している場合は、DHCP サーバーの設定を更新します。
3. Manager のホスト名を更新します。
4. 以下のコマンドを実行します。

```
# /usr/share/ovirt-engine/setup/bin/ovirt-engine-rename
```

5. プロンプトが表示されたら、**Enter** キーを押してエンジンサービスを停止してください。

```
During execution engine service will be stopped (OK, Cancel) [OK]:
```

6. プロンプトが表示されたら、マネージャーの新しい FQDN を入力します。

```
New fully qualified server name:new_engine_fqdn
```

ovirt-engine-rename コマンドは、Manager の FQDN のレコードを更新します。

セルフホスト型エンジンの場合は、次の追加手順を実行します。

1. 既存のすべてのセルフホスト型エンジンノードで次のコマンドを実行します。

```
# hosted-engine --set-shared-config fqdn new_engine_fqdn --type=he_local
```

このコマンドは、各セルフホストエンジンノードのローカルコピー **/etc/ovirt-hosted-engine-ha/hosted-engine.conf** の FQDN を変更します

2. セルフホスト型エンジンノードの 1 つで次のコマンドを実行します。

```
# hosted-engine --set-shared-config fqdn new_engine_fqdn --type=he_shared
```

このコマンドは、共有ストレージドメイン上の **/etc/ovirt-hosted-engine-ha/hosted-engine.conf** のマスターコピーの FQDN を変更します。

現在、すべての新規および既存のセルフホスト型エンジンノードは新しい FQDN を使用します。

SSL 証明書の更新

ovirt-engine-rename コマンドの後に以下のコマンドを実行し、SSL 証明書を更新します。

```
1. # names="websocket-proxy imageio-proxy"
```

```
2. # subject="$(\
    openssl x509 \
    -in /etc/pki/ovirt-engine/certs/apache.cer \
    -noout \
    -subject | \
        sed \
            's;subject= \(.*\);\1;'
  )"
```

```
3. # . /usr/share/ovirt-engine/bin/engine-prolog.sh
```

```
4. # for name in $names; do
    /usr/share/ovirt-engine/bin/pki-enroll-pkcs12.sh \
        --name="${name}" \
        --password=mypass \
        --subject="${subject}" \
        --keep-key \
        --san=DNS:"${ENGINE_FQDN}"
  done
```

22.2. エンジン設定ツール

22.2.1. エンジン設定ツール

エンジン設定ツールは、Red Hat Virtualization 環境のグローバル設定を設定するためのコマンドラインユーティリティーです。このツールは、エンジンデータベースに格納されているキーと値のマッピングのリストと対話し、個々のキーの値を取得して設定し、使用可能なすべての設定キーおよび値のリストを取得できるようにします。さらに、Red Hat Virtualization 環境の設定レベルごとに異なる値を保管できます。

注記

設定キーの値を取得または設定するために、Red Hat Virtualization Manager も Red Hat JBoss Enterprise Application Platform も実行している必要はありません。設定キー値とキーのマッピングはエンジンデータベースに保存されるため、**postgresql** サービスの実行中に更新することができます。その後、**ovirt-engine** サービスが再起動されたときに変更が適用されます。

22.2.2. engine-config コマンドの構文

Red Hat Virtualization Manager がインストールされているマシンからエンジン設定ツールを実行することができます。使用方法の詳細については、そのコマンドのヘルプ出力を印刷してください。

```
# engine-config --help
```

一般的なタスク:

- 使用可能な設定キーを一覧表示します

  ```
  # engine-config --list
  ```

- 使用可能な設定値を一覧表示します。

  ```
  # engine-config --all
  ```

- 設定キーの値を取得します。

  ```
  # engine-config --get KEY_NAME
  ```

KEY_NAME を優先するキーの名前に置き換えて、与えられたバージョンのキーの値を取得します。取得する値の設定バージョンを指定する場合は、**-cver** パラメーターを使用します。バージョンを指定しない場合は、既存のすべてのバージョンの値が返されます。

- キーの設定値を設定します。

```
# engine-config --set KEY_NAME=KEY_VALUE --cver=VERSION
```

KEY_NAME は設定するキーの名前に、KEY_VALUE は設定する値に置き換えてください。複数の設定バージョンがある環境では、**VERSION を** 指定する必要があります。

- 変更を読み込むために ovirt-engine サービスを再起動します。
変更を有効にするには、**ovirt-engine** サービスを再起動する必要があります。

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

22.3. USB フィルターエディター

22.3.1. USB Filter Editor のインストール

USB Filter Editor は、ポリシーファイル **usbfilter.txt** を設定するために使用する Windows ツールです。このファイルで定義されたポリシールールは、クライアントマシンから Red Hat Virtualization Manager を使用して管理される仮想マシンへの特定の USB デバイスの自動パススルーを許可または拒否します。ポリシーファイルは、Red Hat Virtualization Manager 上の **/etc/ovirt-engine/usbfilter.txt** に存在します。 USB フィルターポリシーの変更は、Red Hat Virtualization Manager 上の **ovirt-engine** サービスが再起動されない限り、有効にはなりません。

Red Hat Virtualization Manager 用のインストーラーとイメージ から **USB Filter Editor** インストーラーをダウンロードします。

USB Filter Editor のインストール

1. Windows マシンで、**USB Filter Editor** 用にダウンロードした **.msi** ファイルを実行します。
2. インストールウィザードの手順に従います。特に指定がない限り、USB フィルターエディターは、Windows のバージョンに応じて、デフォルトで **C:\Program Files\RedHat\USB Filter Editor** または **C:\Program Files(x86)\RedHat\USB Filter Editor** のいずれかにインストールされます。
3. デスクトップに USB フィルターエディターのショートカットアイコンが作成されます。

重要

セキュアコピー (SCP) クライアントを使用して、Red Hat Virtualization Manager からフィルターポリシーをインポートおよびエクスポートします。Windows マシン用のセキュアコピーツールは WinSCP (http://winscp.net) です。

デフォルトの USB デバイスポリシーは、仮想マシンに USB デバイスへの基本的なアクセスを提供します。ポリシーを更新して、追加の USB デバイスの使用を許可します。

22.3.2. USB フィルターエディターインターフェイス

デスクトップの USB Filter Editor ショートカットアイコンをダブルクリックします。

Red Hat USB Filter Editorインターフェイスには、各 USB デバイスの **クラス**、**ベンダー**、**製品**、**リビジョン** 、および **アクション** が表示されます。許可された USB デバイスは、 **Action** 列で **Allow** に設定されています。禁止されているデバイスは **Block** に設定されています。

表22.1 USB エディターフィールド

Name	説明
クラス	USB デバイスのタイプ (プリンター、大容量ストレージコントローラーなど)。
Vendor	選択したタイプの製造元。
製品	特定の USB デバイスモデル。
リビジョン	製品の改訂。
アクション	指定されたデバイスを許可またはブロックします。

USB デバイスポリシールールは、リストされている順序で処理されます。**上** ボタンと **下** ボタンを使用して、ルールをリストの上下に移動します。USB Filter Editor で明示的に許可されていない限り、すべての USB デバイスが拒否されるようにするには、ユニバーサルな **ブロック** ルールを最下位のエントリーとして残す必要があります。

22.3.3. USB ポリシーの追加

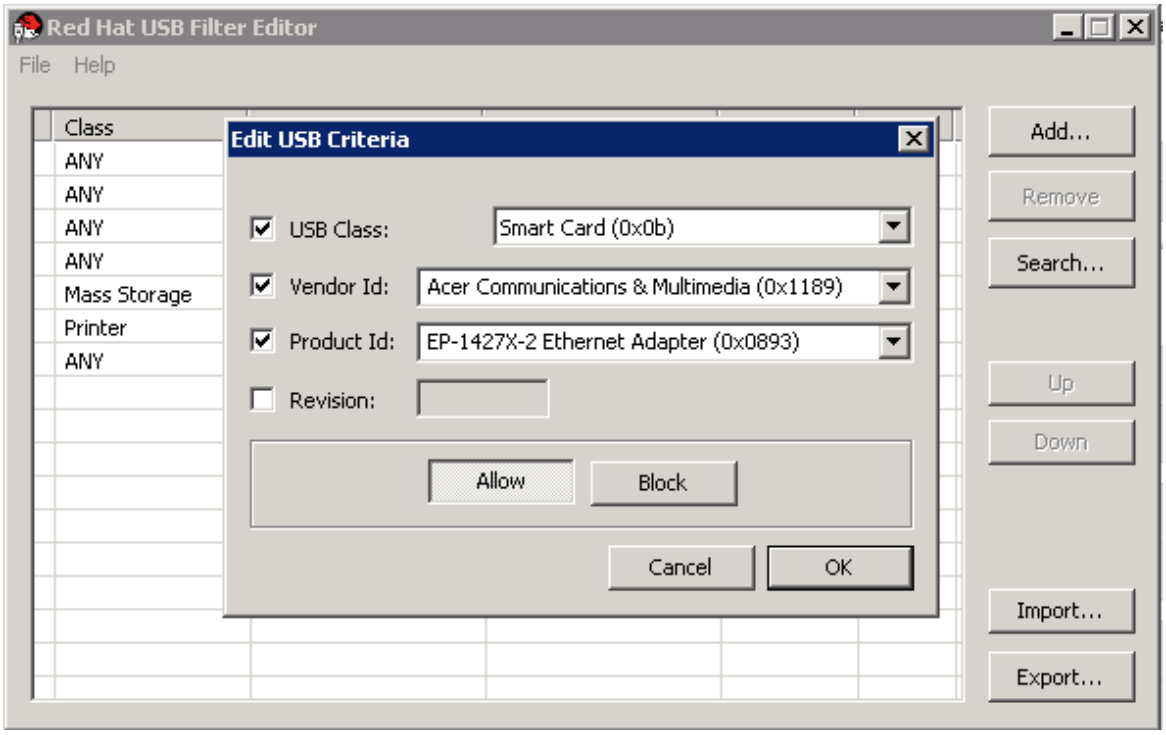
デスクトップの USB Filter Editor ショートカットアイコンをダブルクリックして、エディターを開きます。

USB ポリシーの追加

1. **Add** をクリックします。
2. **USB Class**、**Vendor ID**、**Product ID**、および **Revision**チェックボックスおよびリストを使用して、デバイスを指定します。
   Allow ボタンをクリックして、仮想マシンによる USB デバイスの使用を許可します。 **Block** ボタンをクリックして、仮想マシンの USB デバイスを禁止します。

   OK をクリックすると、選択したフィルタールールがリストに追加され、ウィンドウが閉じます。

   > **例22.1 デバイスの追加**
   >
   > 以下は、メーカーの **Acer Communications & Multimedia** の USB クラス **Smartcard** デバイス **EP-1427X-2 Ethernet Adapter** を許可されたデバイスのリストに追加する方法の例です。



3. **File → Save** をクリックして、変更を保存します。

USB Filter Editor に USB ポリシーが追加されました。USB フィルターポリシーを有効にするには、Red Hat Virtualization Manager にエクスポートする必要があります。

22.3.4. USB ポリシーの削除

デスクトップの USB Filter Editor ショートカットアイコンをダブルクリックして、エディターを開きます。

USB ポリシーの削除

1. 削除するポリシーを選択します。
2. **Remove** をクリックします。ポリシーを削除することを確認するメッセージが表示されます。
3. **Yes** をクリックして、ポリシーを削除することを確認します。
4. **File → Save** をクリックして、変更を保存します。

USB フィルターエディターから USB ポリシーを削除しました。USB フィルターポリシーを有効にするには、Red Hat Virtualization Manager にエクスポートする必要があります。

22.3.5. USB デバイスポリシーの検索

接続されている USB デバイスを検索して、USB フィルターエディターで許可またはブロックします。

デスクトップの USB Filter Editor ショートカットアイコンをダブルクリックして、エディターを開きます。

USB デバイスポリシーの検索

1. **Search** をクリックします。**Attached USB Devices** ウィンドウには、接続されているすべてのデバイスのリストが表示されます。

2. デバイスを選択し、必要に応じて **Allow** または **Block** をクリックします。選択したデバイスをダブルクリックして、ウィンドウを閉じます。デバイスのポリシールールがリストに追加されます。

3. **Up** ボタンと **Down** ボタンを使用して、リスト内の新しいポリシールールの位置を変更します。

4. **File → Save** をクリックして、変更を保存します。

接続されている USB デバイスを検索しました。USB フィルターポリシーを有効にするには、Red Hat Virtualization Manager にエクスポートする必要があります。

22.3.6. USB ポリシーのエクスポート

更新されたポリシーを有効にするには、USB デバイスポリシーの変更をエクスポートして Red Hat Virtualization Manager にアップロードする必要があります。ポリシーをアップロードし、**ovirt-engine** サービスを再起動します。

デスクトップの USB Filter Editor ショートカットアイコンをダブルクリックして、エディターを開きます。

USB ポリシーのエクスポート

1. **Export** をクリックします。**Save As** ウィンドウが開きます。

2. ファイル名を **usbfilter.txt** として保存してください。

3. WinSCP などのセキュアコピークライアントを使用して、**usbfilter.txt** ファイルを Red Hat Virtualization Manager を実行しているサーバーにアップロードします。ファイルは、サーバー上の **/etc/ovirt-engine/** ディレクトリーに配置する必要があります。

4. Red Hat Virtualization Manager を実行しているサーバーで **root** ユーザーとして、**ovirt-engine** サービスを再起動します。

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

22.3.7. USB ポリシーのインポート

編集する前に、既存の USB デバイスポリシーをダウンロードして USB フィルターエディターにインポートする必要があります。

USB ポリシーのインポート

1. WinSCP などの Secure Copy クライアントを使用して、Red Hat Virtualization Manager を実行しているサーバーから **usbfilter.txt** ファイルをダウンロードします。このファイルは、サーバー上の **/etc/ovirt-engine/** ディレクトリーにあります。

2. デスクトップの USB Filter Editor ショートカットアイコンをダブルクリックして、エディターを開きます。

3. **Import** をクリックして **Open** ウィンドウを開きます。

4. サーバーからダウンロードした **usbfilter.txt** ファイルを開きます。

22.4. ログコレクターツール

22.4.1. ログコレクター

ログ収集ツールは、Red Hat Virtualization Manager に含まれています。これにより、サポートを要求するときに、Red Hat Virtualization 環境全体から関連するログを簡単に収集できます。

ログ収集コマンドは、**ovirt-log-collector** です。**root** ユーザーとしてログインし、Red Hat Virtualization 環境の管理認証情報を提供する必要があります。**ovirt-log-collector -h** コマンドは、**ovirt-log-collector** コマンドのすべての有効なオプションのリストを含む使用法情報を表示します。

22.4.2. ovirt-log-collector コマンドの構文

ログコレクターコマンドの基本構文は以下の通りです。

```
# ovirt-log-collector options  list all|clusters|datacenters
# ovirt-log-collector options collect
```

サポートされている 2 つの操作モードは、**list** と **collect** です。

- **list** パラメーターは、Red Hat Virtualization Manager に接続されているホスト、クラスター、またはデータセンターのいずれかをリストします。リストされたオブジェクトに基づいてログコレクションをフィルターリングできます。
- **collect** パラメーターは、Red Hat Virtualization Manager からのログ収集を実行します。収集されたログは、**/tmp/logcollector** ディレクトリーの下のアーカイブファイルに配置されます。**ovirt-log-collector** コマンドは、各ログに特定のファイル名を割り当てます。

別のパラメーターが指定されていない限り、使用可能なホストを、それらが属するデータセンターおよびクラスターと一緒にリストすることがデフォルトのアクションとなります。特定のログを取得するために、ユーザー名とパスワードを入力するように求められます。

ovirt-log-collector コマンドをさらに改良するための多数のパラメーターがあります。

一般的なオプション

--version
使用中のコマンドのバージョン番号を表示し、プロンプトに戻る。

-h、--help
コマンドの使用情報を表示し、プロンプトに戻ります。

--conf-file=PATH
ツールが使用する設定ファイルとして **PATH** を設定します。

--local-tmp=PATH
ログを保存するディレクトリーとして **PATH** を設定します。デフォルトのディレクトリーは **/tmp/logcollector** です。

--ticket-number=TICKET
SOS レポートに関連付けるチケット、またはケース番号として **TICKET** を設定します。

--upload=FTP_SERVER
取得したログを FTP で送信する際の送信先を **FTP_SERVER** に設定します。
Red Hat のサポート担当者からアドバイスがない限り、このオプションは使用しないでください。

--log-file=PATH

コマンドがログ出力に使用する特定のファイル名として **PATH** を設定します。

--quiet

quiet モードを設定し、コンソール出力を最小限に抑えます。デフォルトでは、quiet モードはオフになっています。

-v、--verbose

verbose モードを設定し、より多くのコンソール出力を提供します。デフォルトでは、verbose モードはオフになっています。

--time-only

完全な SOS レポートを生成せずに、ホスト間の時差に関する情報のみを表示します。

Red Hat Virtualization Manager のオプション

これらのオプションはログコレクションをフィルターリングし、Red Hat Virtualization Manager の認証の詳細を指定します。

これらのパラメーターを、特定のコマンドと組み合わせることができます。たとえば、**ovirt-log-collector --user=admin@internal --cluster ClusterA,ClusterB --hosts "SalesHost"*** はユーザーを **admin@internal** として指定し、ログコレクションをクラスター **A** および **B** の **SalesHost** ホストのみに制限します。

--no-hypervisors

ログコレクションから仮想化ホストを省略します。

--one-hypervisor-per-cluster

各クラスターから 1 台のホスト (存在する場合は SPM) のログを収集します。

-u USER、--user=USER

ログイン用のユーザー名を設定します。**USER** は、**user@domain** の形式で指定されます。ここで、**user** はユーザー名で、**domain** は使用中のディレクトリーサービスドメインになります。ユーザーはディレクトリーサービスに存在し、Red Hat Virtualization Manager に認識されている必要があります。

-r FQDN, --rhevm=FQDN

ログの収集元となる Red Hat Virtualization Manager の完全修飾ドメイン名を設定します。**FQDN** は、Manager の完全修飾ドメイン名に置き換えられます。ログコレクターは、Red Hat Virtualization Manager と同じローカルホストで実行していると想定されています。デフォルト値は **localhost** です。

-c CLUSTER、--cluster=CLUSTER

Red Hat Virtualization Manager からのログに加えて、指定された **CLUSTER** 内の仮想化ホストからログを収集します。含めるクラスターは、クラスター名または一致パターンのコンマ区切りリストで指定する必要があります。

-d DATACENTER, --data-center=DATACENTER

Red Hat Virtualization Manager からのログに加えて、指定された **DATACENTER** の仮想化ホストからログを収集します。含めるデータセンターは、データセンター名または一致パターンのコンマ区切りリストで指定する必要があります。

-H HOSTS_LIST, --hosts=HOSTS_LIST

Red Hat Virtualization Manager からのログに加えて、指定された **HOSTS_LIST** 内の仮想化ホストからログを収集します。含めるホストは、ホスト名、完全修飾ドメイン名、または IP アドレスのコンマ区切りリストで指定する必要があります。一致パターンも有効です。

SSH 設定

--ssh-port=PORT

PORT を、仮想化ホストでの SSH 接続に使用するポートとして設定します。

-k KEYFILE、--key-file=KEYFILE

仮想ホストへのアクセスに使用される公開 SSH 鍵として KEYFILE を設定します。

--max-connections=MAX_CONNECTIONS

仮想化ホストからのログに対する最大同時 SSH 接続として MAX_CONNECTIONS を設定します。デフォルトは **10** です。

PostgreSQL データベースのオプション

データベースユーザー名とデータベース名は、デフォルト値から変更されている場合は、**pg-user** と **dbname** パラメーターを使用して指定する必要があります。

データベースがローカルホスト上にない場合は、**pg-dbhost** パラメーターを使用します。オプションの **pg-host-key** パラメーターを使用して、リモートログを収集します。リモートログ収集を成功させるには、PostgreSQL SOS プラグインをデータベースサーバーにインストールする必要があります。

--no-postgresql

データベースのコレクションを無効にします。**--no-postgresql** パラメーターが指定されていない限り、ログコレクターは Red Hat Virtualization Manager PostgreSQL データベースに接続し、データをログレポートに含めます。

--pg-user=USER

データベースサーバーとの接続に使用するユーザー名として USER を設定します。デフォルトは postgres です。

--pg-dbname=DBNAME

データベースサーバーへの接続に使用するデータベース名として DBNAME を設定します。デフォルトは rhevm です。

--pg-dbhost=DBHOST

データベースサーバーのホスト名として DBHOST を設定します。デフォルトは localhost です。

--pg-host-key=KEYFILE

データベースサーバーの公開 ID ファイル (秘密鍵) として、KEYFILE を設定します。この値はデフォルトでは設定されていません。データベースがローカルホストに存在しない場合にのみ必要です。

22.4.3. ログコレクターの基本的な使用法

追加のパラメーターを指定せずに **ovirt-log-collector** コマンドを実行すると、デフォルトの動作では、Red Hat Virtualization Manager とそれに接続されているホストからすべてのログが収集されます。また、**--no-postgresql** パラメーターを追加しない限り、データベースのログを収集します。次の例では、ログコレクターを実行して、Red Hat Virtualization Manager と接続されている 3 つのホストからすべてのログを収集します。

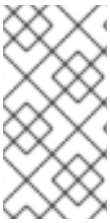
例22.2 ログコレクターの使用法

```
# ovirt-log-collector
INFO: Gathering oVirt Engine information...
INFO: Gathering PostgreSQL the oVirt Engine database and log files from localhost...
Please provide REST API password for the admin@internal oVirt Engine user (CTRL+D to abort):
About to collect information from 3 hypervisors. Continue? (Y/n):
INFO: Gathering information from selected hypervisors...
INFO: collecting information from 192.168.122.250
INFO: collecting information from 192.168.122.251
```

```
INFO: collecting information from 192.168.122.252
INFO: finished collecting information from 192.168.122.250
INFO: finished collecting information from 192.168.122.251
INFO: finished collecting information from 192.168.122.252
Creating compressed archive...
INFO Log files have been collected and placed in /tmp/logcollector/sosreport-rhn-account-20110804121320-ce2a.tar.xz.
The MD5 for this file is 6d741b78925998caff29020df2b2ce2a and its size is 26.7M
```

22.5. ISO アップローダーツール

22.5.1. ISO アップローダーツール



注記

ISO アップローダーツールが非推奨になりました。Red Hat は、管理ポータルまたは REST API を使用して、データドメインに ISO イメージをアップロードすることを推奨します。詳細は、「データストレージドメインへのイメージのアップロード」 を参照してください。

ISO アップローダーは、ISO イメージを ISO 保存領域にアップロードするためのツールです。これは、Red Hat Virtualization Manager の一部としてインストールされます。

ISO アップローダーのコマンドは **engine-iso-uploader** です。このコマンドを使用するには、**root** ユーザーとしてログインし、Red Hat Virtualization 環境の管理認証情報を提供する必要があります。**engine-iso-uploader -h** コマンドは、**engine-iso-uploader** コマンドの有効なオプション一覧などの使用法に関する情報を表示します。

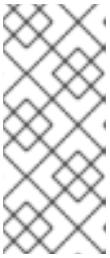
22.5.2. engine-iso-uploader コマンドの構文

ISO uploader コマンドの基本的な構文は、以下のとおりです。

```
# engine-iso-uploader options list
# engine-iso-uploader options upload file file file
```

ISO uploader コマンドは、**list** および **upload** の 2 つのアクションをサポートします。

- **list** アクションは、ISO ファイルをアップロードできる ISO ストレージドメインを一覧表示します。Red Hat Virtualization Manager は、インストールプロセス中に Manager がインストールされているマシンにこの一覧を作成します。
- **upload** アクションは、スペースで区切られた単一の ISO ファイルまたは複数の ISO ファイルを指定された ISO ストレージドメインにアップロードします。デフォルトでは NFS が使用されますが、SSH も利用可能です。



注記

SSH を使用する場合、SSH ユーザーパスワードの入力を複数回求められます。これらのプロンプトを回避するには、ISO ファイルをアップロードする前に公開 SSH キーを iso ドメインサーバーにアップロードしてから、**--key-file=KEYFILE** オプションを使用します。公開 SSH 鍵をアップロードする方法の 1 つとして、**ssh-copy-id -i ~/.ssh/mykey user@host** コマンドを使用できます。

ISO uploader コマンドを使用する場合は、上記のアクションのいずれかを指定する必要があります。さらに、**upload** アクションを使用するには、少なくとも 1 つのローカルファイルを指定する必要があります。

engine-iso-uploader コマンドをさらに絞り込むパラメーターがいくつかあります。

一般的なオプション

--version

ISO uploader コマンドのバージョンを表示します。

-h、--help

ISO uploader コマンドの使用方法に関する情報を表示します。

--conf-file=PATH

コマンドが使用する設定ファイルとして **PATH** を設定します。デフォルトは **/etc/ovirt-engine/isouploader.conf** です。

--log-file=PATH

コマンドがログ出力の書き込みに使用する特定のファイル名として**PATH**を設定します。デフォルトは **/var/log/ovirt-engine/ovirt-iso-uploader/ovirt-iso-uploader_date.log** です。

--cert-file=PATH

エンジンを検証するための証明書として **PATH** を設定します。デフォルトは **/etc/pki/ovirt-engine/ca.pem** です。

--insecure

エンジンの検証を試行しないことを指定します。

--nossl

エンジンへの接続に SSL を使用しないことを指定します。

--quiet

quiet モードを設定し、コンソール出力を最小限に抑えます。

-v、--verbose

verbose モードを設定し、より多くのコンソール出力を提供します。

-f、--force

アップロードするソースファイルのファイル名が宛先 ISO ドメインの既存のファイルと同じである場合は、強制モードが必要です。このオプションでは、既存のファイルを強制的に上書きします。

Red Hat Virtualization Manager のオプション

-u USER、--user=USER

コマンドの実行に認証情報が使用されるユーザーを指定します。**USER** は、**username@domain** 形式で指定されます。ユーザーは指定のドメインに存在し、Red Hat Virtualization Manager に認識されている必要があります。

-r FQDN、--engine=FQDN

イメージのアップロード元の Red Hat Virtualization Manager の IP アドレスまたは完全修飾ドメイン名を指定します。イメージアップローダーは、Red Hat Virtualization Manager がインストールされているのと同じマシンから実行されていると想定されています。デフォルト値は **localhost:443** です。

ISO ストレージドメインのオプション

次のオプションは、イメージがアップロードされる ISO ドメインを指定します。これらのオプションを一緒に使用することはできません。**-i** オプションまたは **-n** オプションのいずれかを使用する必要があります。

-i、--iso-domain=ISODOMAIN

アップロードの宛先としてストレージドメイン **ISODOMAIN** を設定します。

-n、--nfs-server=NFSSERVER

アップロードの宛先として NFS パス **NFSSERVER** を設定します。

接続オプション

ISO アップローダーは、デフォルトで NFS を使用してファイルをアップロードします。これらのオプションは、代わりに SSH ファイル転送を指定します。

--ssh-user=USER

アップロードに使用する SSH ユーザー名として **USER** を設定します。デフォルトは **root** です。

--ssh-port=PORT

SSH への接続時に使用するポートとして **PORT** を設定します。

-k KEYFILE、--key-file=KEYFILE

SSH 認証に使用する秘密鍵として **KEYFILE** を設定します。鍵を設定しない場合、**--ssh-user=USER** で指定されたユーザーのパスワードを入力するように求められます。ISO ファイルをアップロードする前に、公開 SSH 鍵を ISO ドメインサーバーにアップロードする必要があります。これを実行する 1 つの方法として、**ssh-copy-id -i ~/.ssh/mykey user@host** コマンドを使用できます。

22.5.3. NFS サーバーの指定

例22.3 NFS サーバーへのアップロード

```
# engine-iso-uploader --nfs-server=storage.demo.redhat.com:/iso/path upload RHEL6.0.iso
```

22.5.4. 基本的な ISO アップローダーの使用法

以下の例は、ISO アップローダーとリストパラメーターを示しています。最初のコマンドは、利用可能な ISO ストレージドメインを一覧表示します。コマンドでユーザーが指定されていないため、**admin@internal** ユーザーが使用されます。2 番目のコマンドは、指定された ISO ドメインに NFS 経由で ISO ファイルをアップロードします。

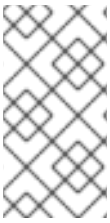
例22.4 ドメインの一覧表示およびイメージのアップロード

```
# engine-iso-uploader list
Please provide the REST API password for the admin@internal oVirt Engine user (CTRL+D to abort):
```

```
ISO Storage Domain Name   | Datacenter          | ISO Domain Status
ISODomain                 | Default             | active
```

```
# engine-iso-uploader --iso-domain=[ISODomain] upload [RHEL6.iso]
Please provide the REST API password for the admin@internal oVirt Engine user (CTRL+D to
abort):
```

22.5.5. VirtIO およびゲストツールのイメージファイルの ISO ストレージドメインへのアップロード



注記

ISO ドメインは、非推奨のストレージドメインタイプです。ISO アップローダーツールが非推奨になりました。Red Hat は、管理ポータルまたは REST API を使用して、データドメインに ISO イメージをアップロードすることを推奨します。詳細は、「データストレージドメインへのイメージのアップロード」 を参照してください。

virtio-win.iso イメージおよび Virtual Floppy Drive (VFD) イメージには Windows 仮想マシンの VirtIO ドライバーが含まれ、**RHV-toolsSetup_version.iso** には、Windows 仮想マシンの Red Hat Virtualization ゲストツールが含まれます。これらのイメージファイルは、仮想マシンにインストールし、パフォーマンスとユーザービリティーを向上させるソフトウェアを提供します。

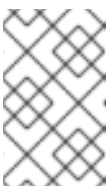
最新バージョンの **virtio-win** ファイルおよび **RHV-toolsSetup_version.iso** ファイルをインストールしてアップロードするには、以下を実行します。

1. Manager マシンにイメージファイルをインストールします。

   ```
   # yum -y install virtio-win rhv-guest-tools-iso*
   ```

   Manager マシンにインストールすると、これらのイメージファイルは以下のようになります。

   - **/usr/share/virtio-win/virtio-win_amd64.vfd** (シンボリックリンク)
   - **/usr/share/virtio-win/virtio-win_servers_amd64.vfd** (シンボリックリンク)
   - **/usr/share/virtio-win/virtio-win_servers_x86.vfd** (シンボリックリンク)
   - **/usr/share/virtio-win/virtio-win_x86.vfd** (シンボリックリンク)
   - **/usr/share/virtio-win/virtio-win.iso** (シンボリックリンク)
   - **/usr/share/rhv-guest-tools-iso/RHV-toolsSetup_version.iso**

   注記

   上記のシンボリックリンクは、名前にバージョンが含まれるファイルを参照しています。これらをコピーするか、rsync などのツールを使用する場合は、シンボリックリンクのターゲットをコピーします。

2. インストール中にローカルで作成されなかった ISO ストレージドメインにイメージファイルをアップロードします。

```
# yum -y install virtio-win rhv-guest-tools-iso*
# engine-iso-uploader --iso-domain=ISODomain upload  \
    /usr/share/virtio-win/virtio-win_amd64.vfd \
    /usr/share/virtio-win/virtio-win_servers_amd64.vfd \
    /usr/share/virtio-win/virtio-win_servers_x86.vfd \
    /usr/share/virtio-win/virtio-win_x86.vfd \
    /usr/share/virtio-win/virtio-win.iso \
    /usr/share/rhv-guest-tools-iso/RHV-toolsSetup_version.iso
```

3. イメージファイルを仮想マシンに接続します。

これで、仮想マシンは virtio ドライバーとゲストツールを使用できるようになります。

イメージファイルを仮想マシンにアタッチする方法の詳細は、**Virtual Machine Management Guide**の Installing the Guest Agents, Tools, and Drivers on Windows を参照してください。

22.6. エンジンバキュームツール

22.6.1. エンジンバキュームツール

Engine Vacuum ツールは、テーブルを更新してデッド行を削除することで PostgreSQL データベースを維持し、ディスクスペースを再利用できるようにします。**VACUUM** コマンドとそのパラメーターに関する詳細は、PostgreSQL documentation を参照してください。

Engine Vacuum コマンドは **engine-vacuum** です。**root** ユーザーとしてログインし、Red Hat Virtualization 環境の管理認証情報を提供する必要があります。

また、**engine-setup** コマンドを使用しながら Engine Vacuum ツールを実行することで、既存のインストールをカスタマイズすることも可能です。

```
$ engine-setup
...
[ INFO  ] Stage: Environment customization
...
Perform full vacuum on the engine database engine@localhost?
This operation may take a while depending on this setup health and the
configuration of the db vacuum process.
See https://www.postgresql.org/docs/10/static/sql-vacuum.html
(Yes, No) [No]:
```

Yes オプションは、Engine Vacuum ツールをフルバキュームバーボーズモードで実行します。

22.6.2. エンジンバキュームモード

エンジンバキュームには 2 つのモードがあります。

Standard Vacuum

頻繁に標準的なバキュームを行うことをお勧めします。
標準のバキュームは、テーブルとインデックスのデッドローバージョンを削除し、スペースを将来の再利用に使用できるものとしてマークします。頻繁に更新されるテーブルは、定期的にバキュームする必要があります。しかし、標準的なバキュームでは、そのスペースを OS に戻すことはできません。

パラメーターのない標準バキュームは、現在のデータベース内のすべてのテーブルを処理します。

Full Vacuum

日常的な使用には完全バキュームは推奨されませんが、テーブル内からかなりの量のスペースを再利用する必要がある場合にのみ実行する必要があります。
完全バキュームは、デッドスペースのないテーブルファイルの新しいコピーを書き込むことによってテーブルを圧縮し、それによってオペレーティングシステムがスペースを再利用できるようにします。フルバキュームには時間がかかる場合があります。

完全バキュームでは、操作が完了して古いコピーが削除されるまで、テーブルの新しいコピー用に追加のディスクスペースが必要です。フルバキュームにはテーブルの排他的ロックが必要なため、テーブルの他の使用と並行して実行することはできません。

22.6.3. engine-vacuum コマンドの構文

engine-vacuum コマンドの基本構文は以下の通りです。

```
# engine-vacuum
```

```
# engine-vacuum option
```

engine-vacuum コマンドをオプションなしで実行すると、標準的なバキュームが実行されます。

engine-vacuum コマンドをさらに洗練させるためのいくつかのパラメーターがあります。

一般的なオプション

-h --help
engine-vacuum コマンドの使用方法に関する情報を表示します。

-a
標準のバキュームを実行し、データベースを分析して、オプティマイザーの統計を更新します。

-A
バキューム処理を行わずに、データベースを解析し、オプティマイザーの統計情報を更新します。

-f
完全な vacuum を実行します。

-v
詳細モードで実行して、より多くのコンソール出力を提供します。

-t table_name
特定の 1 つテーブルまたは複数のテーブルを vacuum します。

```
# engine-vacuum -f -v -t vm_dynamic -t vds_dynamic
```

22.7. VDSM からネットワーク名へのマッピングツール

22.7.1. VDSM 名の論理ネットワーク名へのマッピング

論理ネットワークの名前が 15 文字を超えるか、非 ASCII 文字が含まれる場合、システムはホスト上の識別子 (**vdsm_name**) 名を自動的に生成します。これは、**on** の文字とネットワークの一意の識別子の最初の 13 文字 (例: **ona1b2c3d4e5f6g**) で設定されます。ホストのログファイルに表示されるのは、この

名前です。論理ネットワーク名とその自動生成ネットワーク名の一覧を表示するには、**/usr/share/ovirt-engine/bin/** にある VDSM-to-Network-Name マッピングツールを使用します。

手順

1. ツールを初めて実行するときは、PASSWORD 環境変数を定義します。これは、Manager データベースへの読み取りアクセス権を持つデータベースユーザーのパスワードです。たとえば、以下を実行します。

```
# export PASSWORD=DatabaseUserPassword
```

2. VDSM からネットワーク名へのマッピングツールを実行します。

```
# vdsm_to_network_name_map  --user USER
```

ここで、**USER** は、Manager データベースへの読み取りアクセス権を持つデータベースユーザーであり、そのパスワードは PASSWORD 環境変数に割り当てられています。

このツールは、同等のオンホスト識別子にマップされている論理ネットワーク名のリストを表示します。

その他のフラグ

以下のフラグを指定してツールを実行できます。

--host は、データベースサーバーのホスト名/IP アドレスです。デフォルト値は **localhost** です。

--port はデータベースサーバーのポート番号です。デフォルト値は **5432** です。**--database** はデータベースの名前です。デフォルト値は **engine** で、これは Manager データベースです。

--secure は、データベースとの安全な接続を有効にします。デフォルトでは、ツールは安全な接続なしで実行されます。

パート IV. 環境に関する情報の収集

第23章 RED HAT VIRTUALIZATION MANAGER で INSIGHT を展開

Red Hat Virtualization Manager がインストールされている既存の Red Hat Enterprise Linux (RHEL) システムに Red Hat Insights をデプロイするには、以下のタスクを実行します。

- Red Hat Insights アプリケーションにシステムを登録します。
- Red Hat Virtualization 環境からのデータ収集を有効にします。

システムを Red Hat Insights に登録します。

Red Hat Insights サービスと通信し、Red Hat Insights コンソールに表示される結果を表示するには、システムを登録します。

```
[root@server ~]# insights-client --register
```

Red Hat Virtualization 環境からのデータ収集を有効にする

/etc/ovirt-engine/rhv-log-collector-analyzer/rhv-log-collector-analyzer.conf ファイルを変更して、次の行を含めます。

```
upload-json=True
```

Insights の結果を Insights コンソールで確認

システムおよびインフラストラクチャーの結果は、Insights コンソールで 確認することができます。**概要** タブには、インフラストラクチャーに対する現在のリスクのダッシュボードビューが表示されます。この開始点から、特定のルールがシステムにどのように影響しているかを調査したり、システムベースのアプローチを使用して、システムにリスクをもたらすすべてのルールの一致を表示したりできます。

1. **Rule** hits by severity を選択し、インフラストラクチャーにもたらす **Total Risk** でルールを表示します (**Critical**、**Important**、**Moderate**、または **Low**)。
   または、以下を実行します。
2. **Rule** hits by category を選択し、インフラストラクチャーにもたらすリスクの種類を表示します (**Availability**、**Stability**、**Performance**、または **Security**)。
3. 特定のルールを名前で検索したり、ルールの一覧をスクロールして、Ansible Playbook のリスク、システム公開、および可用性に関するハイレベルな情報を確認して修正を自動化します。
4. ルールをクリックして、ルールの説明を表示し、関連するナレッジベースの記事から詳細を確認し、影響を受けるシステムのリストを表示します。
5. システムをクリックすると、検出された問題と手順に関する特定の情報を表示して、問題を解決します。

第24章 ログファイル

24.1. MANAGER のインストールログファイル

表24.1 インストール

ログファイル	説明
/var/log/ovirt-engine/engine-cleanup_yyyy_mm_dd_hh_mm_ss.log	**engine-cleanup** コマンドからログに記録します。これは、Red Hat Virtualization Manager のインストールをリセットするために使用されるコマンドです。コマンドが実行するたびにログが生成されます。実行の日付と時刻は、複数のログが存在できるようにするためにファイル名で使用されます。
/var/log/ovirt-engine/engine-db-install-yyyy_mm_dd_hh_mm_ss.log	**engine-setup** コマンドから、engine データベースの作成および設定の詳細をログに記録します。
/var/log/ovirt-engine/ovirt-engine-dwh-setup-yyyy_mm_dd_hh_mm_ss.log	**ovirt-engine-dwh-setup** コマンドからログに記録します。これは、レポート用の **ovirt_engine_history** データベースを作成するのに使用されるコマンドです。コマンドが実行するたびにログが生成されます。実行の日付と時刻は、複数のログが同時に存在できるようにするためにファイル名で使用されます。
/var/log/ovirt-engine/setup/ovirt-engine-setup-yyyymmddhhmmss.log	**engine-setup** コマンドからログに記録します。コマンドが実行するたびにログが生成されます。実行の日付と時刻は、複数のログが同時に存在できるようにするためにファイル名で使用されます。

24.2. RED HAT VIRTUALIZATION MANAGER ログファイル

表24.2 サービス活動

ログファイル	説明
/var/log/ovirt-engine/engine.log	すべての Red Hat Virtualization Manager GUI クラッシュ、Active Directory ルックアップ、データベースの問題、およびその他のイベントを反映します。
/var/log/ovirt-engine/host-deploy	Red Hat Virtualization Manager からデプロイされたホストのログファイル。
/var/lib/ovirt-engine/setup-history.txt	Red Hat Virtualization Manager に関連付けられているパッケージのインストールおよびアップグレードを追跡します。

ログファイル	説明
/var/log/httpd/ovirt-requests-log	HTTPS 経由で Red Hat Virtualization Manager に行われたリクエストのログファイルで、各リクエストにかかった時間などが記録されています。 **Correlation-Id** ヘッダーが含まれているため、ログファイルを **/var/log/ovirt-engine/engine.log** と比較するときにリクエストを比較できます。
/var/log/ovn-provider/ovirt-provider-ovn.log	OVN プロバイダーのアクティビティーをログに記録します。Open vSwitch のログに関する情報は、Open vSwitch のドキュメント を参照してください。

24.3. SPICE ログファイル

SPICE のログファイルは、SPICE の接続に関する問題をトラブルシューティングする際に有用です。SPICE デバッグを開始するには、ログレベルを **debugging** に変更します。次に、ログの場所を特定します。

ゲストマシンへのアクセスに使用されるクライアントおよびゲストマシン自体の両方に、SPICE ログファイルがあります。クライアント側のログの場合、**console.vv** ファイルがダウンロードされているネイティブクライアントを使用して SPICE クライアントを起動した場合は、**remote-viewer** コマンドを使用してデバッグを有効にし、ログ出力を生成します。

24.3.1. ハイパーバイザー SPICE サーバーの SPICE ログ

表24.3 ハイパーバイザー SPICE サーバーの SPICE ログ

ログタイプ	ログの場所	ログレベルを変更するには、以下を実行します。
ホスト/ハイパーバイザー SPICE サーバー	/var/log/libvirt/qemu/(guest_name).log	ゲストを起動する前に、ホスト/ハイパーバイザーで **export SPICE_DEBUG_LEVEL=5** を実行します。この変数は QEMU によって解析され、システム全体で実行すると、システム上のすべての仮想マシンのデバッグ情報が出力されます。このコマンドは、クラスター内の各ホストで実行する必要があります。このコマンドは、各クラスターではなく、各ホスト/ハイパーバイザーにのみ機能します。

24.3.2. ゲストマシンの SPICE ログ

表24.4 ゲストマシンの spice-vdagent ログ

ログタイプ	ログの場所	ログレベルを変更するには、以下を実行します。
Windows ゲスト	C:\Windows\Temp\vdagent.log C:\Windows\Temp\vdservice.log	該当なし

Red Hat Enterprise Linux ゲスト	**journalctl** を root ユーザーとして使用します。	**spice-vdagentd** サービスをデバッグモードで実行するには、root ユーザーとして、**SPICE_VDAGENTD_EXTRA_ARGS="-d -d"** のエントリーで **/etc/sysconfig/spice-vdagentd** ファイルを作成します。
		
		コマンドラインからデバッグモードで **spice-vdagent** を実行するには、次の手順に従います。
		
		`$ killall - u $USER spice-vdagent`
		`$ spice-vdagent -x -d [-d] [ |& tee spice-vdagent.log ]`

24.3.3. console.vv ファイルを使用して起動された SPICE クライアントの SPICE ログ

Linux クライアントマシンの場合:

1. **--spice-debug** オプションを指定して **remote-viewer** コマンドを実行し、SPICE デバッグを有効にします。プロンプトが表示されたら、接続 URL を入力します (例: spice://**virtual_machine_IP**:**port**)。

```
#  remote-viewer --spice-debug
```

2. デバッグパラメーターを指定して SPICE クライアントを実行し、それに .vv ファイルを渡すには、**console.vv** ファイルをダウンロードし、**--spice-debug** オプションを指定して **remote-viewer** コマンドを実行し、**console.vv** ファイルへのフルパスを指定します。

```
# remote-viewer --spice-debug /path/to/console.vv
```

Windows クライアントマシンの場合:

1. **virt-viewer** 2.0-11.el7ev 以降のバージョンでは、**virt-viewer.msi** は **virt-viewer** と **debug-viewer.exe** をインストールします。

2. **spice-debug** 引数を指定して **remote-viewer** コマンドを実行し、コマンドをコンソールへのパスに送信します。

```
remote-viewer --spice-debug path\to\console.vv
```

3. ログを表示するには、仮想マシンに接続します。GDB を実行しているコマンドプロンプトが表示され、**remote-viewer** の標準出力および標準エラーが出力されます。

24.4. ホストログファイル

ログファイル	説明
/var/log/messages	**libvirt** によって使用されるログファイル。**journalctl** を使用してログを表示します。ログを表示するには、**adm**、**systemd-journal**、または **wheel** グループのメンバーである必要があります。
/var/log/vdsm/spm-lock.log	Storage Pool Manager のロールでリースを取得するホストの機能の詳細を示すログファイル。ホストがリースを取得、解放、更新、または更新に失敗したときのログの詳細。
/var/log/vdsm/vdsm.log	ホスト上のマネージャーの Manager である VDSM のログファイルです。
/tmp/ovirt-host-deploy-Date.log	ホストが正常にデプロイされた後、**/var/log/ovirt-engine/host-deploy/ovirt-Date-Host-Correlation_ID**.log としてマネージャーにコピーされるホストデプロイメントログ。
/var/log/vdsm/import/import-UUID-Date.log	KVM ホスト、VMWare プロバイダー、または RHEL 5 Xen ホストからの仮想マシンのインポートの詳細を示すログファイル (インポートの失敗情報を含む)。**UUID** は、インポートされた仮想マシンの UUID であり、**Date** はインポートが開始した日時です。
/var/log/vdsm/supervdsm.log	スーパーユーザー権限で実行された VDSM タスクをログに記録します。
/var/log/vdsm/upgrade.log	VDSM は、ホストのアップグレード時にこのログファイルを使用して、設定の変更を記録します。
/var/log/vdsm/mom.log	VDSM のメモリーオーバーコミットメントマネージャーのアクティビティーをログに記録します。

24.5. ホストロギングサーバーのセットアップ

ホストはログファイルを生成および更新し、その動作や問題点を記録しています。これらのログファイルを一元的に収集すると、デバッグが簡素化されます。

この手順は、集中ログサーバーで使用する必要があります。別のロギングサーバーを使用するか、この手順を使用して Red Hat Virtualization Manager でホストロギングを有効にすることができます。

ホストロギングサーバーのセットアップ

1. ファイアウォールが **UDP 514** ポートでのトラフィックを許可し、**syslog** サービストラフィックに対してオープンであるかどうかを確認します。

```
# firewall-cmd --query-service=syslog
```

出力が **no** の場合は、**UDP 514** ポートで次のトラフィックを許可します。

```
# firewall-cmd --add-service=syslog --permanent
# firewall-cmd --reload
```

2. syslog サーバーに新しい **.conf** ファイル (例: **/etc/rsyslog.d/from_remote.conf**) を作成し、次の行を追加します。

```
template(name="DynFile" type="string"
string="/var/log/%HOSTNAME%/%PROGRAMNAME%.log")
RuleSet(name="RemoteMachine"){  action(type="omfile" dynaFile="DynFile")  }
Module(load="imudp")
Input(type="imudp" port="514" ruleset="RemoteMachine")
```

3. **rsyslog** サービスを再起動します。

```
# systemctl restart rsyslog.service
```

4. ハイパーバイザーにログインし、**/etc/rsyslog.conf** に以下の行を追加します。

```
*.info;mail.none;authpriv.none;cron.none @<syslog-FQDN>:514
```

5. ハイパーバイザーで rsyslog サービスを再起動します。

```
# systemctl restart rsyslog.service
```

これで、集中ログサーバーは、仮想化ホストから **messages** ログおよび **secure** ログを受け取って保存するように設定されました。

24.6. OVIRT ENGINE EXTENSION LOGGER LOG4J の有効化

ロガー実装には ovirt-engine-extension-logger-log4j パッケージが必要です。この実装により、Red Had Virtualization Manager はレコードを log4j に委譲します。Log4j は、SNMP や syslog などのさまざまなテクノロジーのアペンダーを提供するカスタマイズ可能なフレームワークです。

oVirt Engine Extension Logger log4j は **engine.log** ファイルを既存の syslog サーバーに渡します。設定手順は、ホストロギングサーバーの設定 と重複します。

中央の syslog ログサーバーでこの手順を使用します。別のログサーバーを使用するか、この手順を使用して、**engine.log** ファイルを Manager から syslog サーバーに渡すことができます。

注記

この拡張機能の syslog サーバーを定義するには、**/etc/ovirt-engine/extensions.d** ディレクトリーに移動し、**Log4jLogger.properties** ファイルの **log4j.appender.myappender.SyslogHost** の値を編集します。

syslog 機能を定義するには、**/etc/ovirt-engine/extensions.d** ディレクトリーに移動し、**Log4jLogger.properties** ファイルで **log4j.appender.myappender.Facility** の値を編集します。たとえば、**log4j.appender.myappender.Facility=local1** です。

oVirt Engine Extension Logger log4j の設定

1. 拡張をインストールします。

```
# yum install ovirt-engine-extension-logger-log4j
```

2. **/etc/ovirt-engine/extensions.d/** ディレクトリーに **Log4jLogger.properties** ファイルを作成し、以下の内容を追加します。

```
ovirt.engine.extension.name = log4jlogger
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.logger.Logger
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.logger.log4j
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.logger.log4j.Log4jLogger
log4j.rootLogger=DEBUG, myappender
log4j.appender.myappender=org.apache.log4j.net.SyslogAppender
log4j.appender.myappender.SyslogHost=localhost
log4j.appender.myappender.layout=org.apache.log4j.PatternLayout
log4j.appender.myappender.layout.ConversionPattern=[%c] %m%n
```

3. rsyslog をインストールして設定します。

```
# yum install rsyslog
```

4. **rsyslog** トラフィックを許可するように SELinux を設定します。

```
# semanage port -a -t syslogd_port_t -p udp 514
```

5. **/etc/rsyslog.conf** を編集し、以下の行を追加します。

```
$template TmplAuth, "/var/log/%fromhost%/secure"
$template TmplMsg, "/var/log/%fromhost%/messages"

$RuleSet remote
authpriv.*   ?TmplAuth
*.info,mail.none;authpriv.none,cron.none   ?TmplMsg
$RuleSet RSYSLOG_DefaultRuleset
$InputUDPServerBindRuleset remote
```

6. 次の 2 行のコメントを解除します。

```
#$ModLoad imudp
#$UDPServerRun 514
```

7. **rsyslog** サービスを再起動します。

```
# systemctl restart rsyslog.service
```

8. ファイアウォールが有効でアクティブな場合は、次のコマンドを実行して、Firewalld で rsyslog ポートを開くために必要なルールを追加します。

```
# firewall-cmd --permanent --add-port=514/udp
# firewall-cmd --reload
```

9. Red Hat Virtualization Manager を再起動します。

```
# restart ovirt-engine
```

これで、既存の syslog サーバーは **engine.log** ファイルを受け取って保存できるようになります。

付録A VDSM およびフック

A.1. VDSM

VDSM サービスは、Red Hat Virtualization Hosts (RHVH) および Red Hat Enterprise Linux ホストを管理するために Red Hat Virtualization Manager により使用されます。VDSM は、ホストのストレージ、メモリー、ネットワークリソースを管理監視します。また、仮想マシンの作成、統計収集、ログ収集、およびその他のホスト管理タスクを調整します。VDSM は、Red Hat Virtualization Manager により管理される各ホスト上でデーモンとして実行されます。クライアントからの XML-RPC 呼び出しに応答します。Red Hat Virtualization Manager は VDSM クライアントとして機能します。

A.2. VDSM フック

VDSM はフックを介して拡張可能です。フックは、重要なイベントが発生したときにホスト上で実行されるスクリプトです。サポートされているイベントが発生すると、VDSM はホスト上の **/usr/libexec/vdsm/hooks/nn_event-name/** にある実行可能フックスクリプトを英数字順に実行します。慣例により、各フックスクリプトには、ファイル名の前に含まれる 2 桁の番号が割り当てられ、スクリプトが実行される順序が明確になります。任意のプログラミング言語でフックスクリプトを作成できますが、この章に含まれる例では Python が使用されます。

イベントのホストで定義されたすべてのスクリプトが実行されることに注意してください。特定のフックをホスト上で実行される仮想マシンのサブセットに対してのみ実行する必要がある場合は、仮想マシンに関連付けられた **カスタムプロパティー** を評価することにより、フックスクリプト自体がこの要件を処理することを確認する必要があります。

警告

VDSM フックは Red Hat Virtualization の動作に干渉する可能性があります。VDSM フックのバグは、仮想マシンのクラッシュやデータの損失を引き起こす可能性があります。VDSM フックは注意して実装し、厳密にテストする必要があります。Hooks API は新しく、将来大幅に変更される可能性があります。

A.3. フックを使用した VDSM の拡張

この章では、イベント駆動型フックを使用して VDSM を拡張する方法について説明します。フックを使用して VDSM を拡張することは実験的な技術であり、この章は経験豊富な開発者を対象としています。仮想マシンにカスタムプロパティーを設定することで、フックスクリプトに仮想マシン固有のパラメーターを追加で渡すことができます。

A.4. サポートされている VDSM イベント

表A.1 サポートされている VDSM イベント

Name	説明
before_vm_start	仮想マシンが起動する前。

Name	説明
after_vm_start	仮想マシンの起動後。
before_vm_cont	仮想マシンが続行する前。
after_vm_cont	仮想マシンが続行した後。
before_vm_pause	仮想マシンが一時停止する前。
after_vm_pause	仮想マシンが一時停止した後。
before_vm_hibernate	仮想マシンが休止状態になる前。
after_vm_hibernate	仮想マシンが休止した後。
before_vm_dehibernate	仮想マシンが休止状態でなくなる前。
after_vm_dehibernate	仮想マシンが休止状態になった後。
before_vm_migrate_source	仮想マシンを移行する前に、移行が行われているソースホストで実行します。
after_vm_migrate_source	仮想マシンの移行後、移行が行われているソースホストで実行します。
before_vm_migrate_destination	仮想マシンを移行する前に、移行が行われている移行先ホストで実行します。
after_vm_migrate_destination	仮想マシンの移行後、移行が行われている移行先ホストで実行します。
after_vm_destroy	仮想マシンの破棄後。
before_vdsm_start	VDSM がホストで開始される前。**before_vdsm_start** フックはユーザー root として実行され、VDSM プロセスの環境を継承しません。
after_vdsm_stop	VDSM がホストで停止した後。**after_vdsm_stop** フックはユーザー root として実行され、VDSM プロセスの環境を継承しません。
before_nic_hotplug	NIC が仮想マシンにホットプラグされる前。
after_nic_hotplug	NIC が仮想マシンにホットプラグされた後。
before_nic_hotunplug	NIC が仮想マシンからホットプラグを抜かれる前。

Name	説明
after_nic_hotunplug	NIC が仮想マシンからホットプラグを抜かれる後。
after_nic_hotplug_fail	NIC を仮想マシンにホットプラグが失敗した後。
after_nic_hotunplug_fail	NIC が仮想マシンからホットプラグを抜かれた後。
before_disk_hotplug	ディスクが仮想マシンにホットプラグされる前。
after_disk_hotplug	ディスクが仮想マシンにホットプラグされた後。
before_disk_hotunplug	ディスクが仮想マシンからのホットプラグを抜かれる前。
after_disk_hotunplug	ディスクが仮想マシンからのホットプラグを抜かれた後。
after_disk_hotplug_fail	ディスクを仮想マシンにホットプラグが失敗した後。
after_disk_hotunplug_fail	ディスクが仮想マシンからホットプラグを抜かれた後。
before_device_create	カスタムプロパティーをサポートするデバイスを作成する前。
after_device_create	カスタムプロパティーをサポートするデバイスを作成した後。
before_update_device	カスタムプロパティーをサポートするデバイスを更新する前。
after_update_device	カスタムプロパティーをサポートするデバイスを更新した後。
before_device_destroy	カスタムプロパティーをサポートするデバイスを破棄する前。
after_device_destroy	カスタムプロパティーをサポートするデバイスを破棄した後。
before_device_migrate_destination	デバイスを移行する前に、移行が行われている宛先ホストで実行します。
after_device_migrate_destination	デバイスの移行後、移行が行われている移行先ホストで実行します。

Name	説明
before_device_migrate_source	デバイスを移行する前に、移行が行われているソースホストで実行します。
after_device_migrate_source	デバイスの移行後、移行が行われているソースホストで実行します。
after_network_setup	ホストマシンの起動時にネットワークを設定した後。
before_network_setup	ホストマシンを起動するときにネットワークを設定する前。

A.5. VDSM フック環境

ほとんどのフックスクリプトは **vdsm** ユーザーとして実行され、VDSM プロセスの環境を継承します。例外は、**before_vdsm_start** イベントと **after_vdsm_stop** イベントによってトリガーされるフックスクリプトです。これらのイベントによってトリガーされるフックスクリプトは **root** ユーザーとして実行され、VDSM プロセスの環境を継承しません。

A.6. VDSM フックドメイン XML オブジェクト

フックスクリプトが開始されると、_hook_domxml 変数が環境に追加されます。この変数には、関連する仮想マシンの libvirt ドメイン XML 表現のパスが含まれています。以下に概説するように、いくつかのフックはこのルールの例外です。次のフックの _hook_domxml 変数には、仮想マシンではなく NIC の XML 表現が含まれています。

- *_nic_hotplug_*
- *_nic_hotunplug_*
- *_update_device
- *_device_create
- *_device_migrate_*

重要

before_migration_destination フックと **before_dehibernation** フックは現在、ソースホストからドメインの XML を受け取ります。配信先のドメインの XML には様々な差異が生じます。

VDSM では、仮想マシンの定義に libvirt domain XML 形式を使用します。libvirt ドメインの XML 形式の詳細は、http://libvirt.org/formatdomain.html を参照してください。仮想マシンの UUID は、ドメイン XML から推測できますが、環境変数 vmId としても使用できます。

A.7. カスタムプロパティーの定義

Red Hat Virtualization Manager によって受け入れられ (次にカスタムフックに渡される) カスタムプロパティーは、**engine-config** コマンドを使用して定義されます。このコマンドは、Red Hat Virtualization Manager がインストールされているホストで **root** ユーザーとして実行します。

UserDefinedVMProperties および **CustomDeviceProperties** 設定キーは、サポートされているカスタムプロパティーの名前を格納するために使用されます。名前付きの各カスタムプロパティーの有効な値を定義する正規表現も、これらの設定キーに含まれています。

複数のカスタムプロパティーはセミコロンで区切られます。設定キーを設定すると、そこに含まれている既存の値が上書きされることに注意してください。新規および既存のカスタムプロパティーを組み合わせる場合は、キーの値を設定するために使用されるコマンドのすべてのカスタムプロパティーを含める必要があります。

設定キーが更新されたら、新しい値を有効にするために **ovirt-engine** サービスを再起動する必要があります。

例A.1 仮想マシンのプロパティー - smartcard カスタムプロパティーの定義

1. 次のコマンドを使用して、**UserDefinedVMProperties** 設定キーによって定義された既存のカスタムプロパティーを確認します。

```
# engine-config -g UserDefinedVMProperties
```

以下の出力が示すように、カスタムプロパティー **memory** が既に定義されています。正規表現 **^[0-9]+$** は、カスタムプロパティーに数字のみが含まれるようにします。

```
# engine-config -g UserDefinedVMProperties
UserDefinedVMProperties:  version: 3.6
UserDefinedVMProperties:  version: 4.0
UserDefinedVMProperties : memory=^[0-9]+$ version: 4.0
```

2. **memory** カスタムプロパティーは **UserDefinedVMProperties** 設定キーですでに定義されているため、新しいカスタムプロパティーを追加する必要があります。追加のカスタムプロパティーである **smartcard** が、設定キーの値に追加されます。新しいカスタムプロパティーは、**true** または **false** の値を保持できます。

```
# engine-config -s UserDefinedVMProperties='memory=^[0-9]+$;smartcard=^(true|false)$'
--cver=4.0
```

3. **UserDefinedVMProperties** 設定キーで定義されたカスタムプロパティーが正しく更新されていることを確認します。

```
# engine-config -g UserDefinedVMProperties
UserDefinedVMProperties:  version: 3.6
UserDefinedVMProperties:  version: 4.0
UserDefinedVMProperties : memory=^[0-9]+$;smartcard=^(true|false)$ version: 4.0
```

4. 最後に、設定の変更を有効にするには、**ovirt-engine** サービスを再起動する必要があります。

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

例A.2 デバイスプロパティー - interface カスタムプロパティーの定義

1. 次のコマンドを使用して、**CustomDeviceProperties** 設定キーで定義されている既存のカスタムプロパティーを確認します。

```
# engine-config -g CustomDeviceProperties
```

以下の出力に示されるように、カスタムプロパティーはまだ定義されていません。

```
# engine-config -g CustomDeviceProperties
CustomDeviceProperties:  version: 3.6
CustomDeviceProperties:  version: 4.0
```

2. **interface** カスタムプロパティーはまだ存在しないため、そのまま追加できます。この例では、**speed** サブプロパティーの値は 0 ~ 99999 の範囲に設定され、 **duplex** サブプロパティーの値は **full** または **half** のいずれかの選択に設定されます。

```
# engine-config -s CustomDeviceProperties="{type=interface;prop={speed=^([0-9]
{1,5})$;duplex=^(full|half)$}}" --cver=4.0
```

3. **CustomDeviceProperties** 設定キーで定義されたカスタムプロパティーが正しく更新されていることを確認します。

```
# engine-config -g CustomDeviceProperties
UserDefinedVMProperties:  version: 3.6
UserDefinedVMProperties:  version: 4.0
UserDefinedVMProperties : {type=interface;prop={speed=^([0-9]
{1,5})$;duplex=^(full|half)$}} version: 4.0
```

4. 最後に、設定の変更を有効にするには、**ovirt-engine** サービスを再起動する必要があります。

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

A.8. 仮想マシンのカスタムプロパティーの設定

Red Hat Virtualization Manager でカスタムプロパティーを定義したら、仮想マシンでの設定を開始できます。カスタムプロパティーは、管理ポータルの **New Virtual Machine** ウィンドウおよび**Edit Virtual Machine** ウィンドウの **Custom Properties** タブで設定されます。

Run Virtual Machine(s) ダイアログボックスからカスタムプロパティーを設定することもできます。**Run Virtual Machine(s)** ダイアログボックスから設定されたカスタムプロパティーは、次にシャットダウンされるまで仮想マシンにのみ適用されます。

Custom Properties タブには、定義済みのカスタムプロパティーのリストから選択するための機能があります。カスタムプロパティーキーを選択すると、追加のフィールドが表示され、そのキーの値を入力できます。**+** ボタンをクリックしてキーと値のペアを追加し、**-** ボタンをクリックしてそれらを削除します。

A.9. VDSM フックでの仮想マシンのカスタムプロパティーの評価

仮想マシンの **Custom Properties** フィールドに設定された各キーは、フックスクリプトを呼び出すときに環境変数として追加されます。**Custom Properties** フィールドの検証に使用される正規表現はある程度の保護を提供しますが、提供された入力が期待に一致することもスクリプトで検証する必要があり

ます。

例A.3 カスタムプロパティーの評価

この短い Python の例では、カスタムプロパティー **key1** の存在を確認します。カスタムプロパティーが設定されている場合は、その値が標準エラーに出力されます。カスタムプロパティーが設定されていないと、アクションは実行されません。

```
#!/usr/bin/python

import os
import sys

if os.environ.has_key('key1'):
 sys.stderr.write('key1 value was : %s\n' % os.environ['key1'])
else:
    sys.exit(0)
```

A.10. VDSM フックモジュールの使用

VDSM には Python フックモジュールが付属しており、VDSM フックスクリプトのヘルパー関数を提供します。このモジュールは例として提供されており、Python で記述された VDSM フックにのみ関連しています。

フックモジュールは、仮想マシンの libvirt XML の DOM オブジェクトへの読み込みをサポートします。フックスクリプトは、Python に組み込まれている **xml.dom** ライブラリー (http://docs.python.org/release/2.6/library/xml.dom.html) を使用してオブジェクトを操作できます。

変更されたオブジェクトは、フックモジュールを使用して libvirt XML に保存して戻すことができます。フックモジュールは、フック開発をサポートするために次の機能を提供します。

表A.2 フックモジュール機能

Name	引数	説明
tobool	string	文字列 "true" または "false" をブール値に変換
read_domxml	-	仮想マシンの libvirt XML を DOM オブジェクトに読み込みます
write_domxml	DOM オブジェクト	DOM オブジェクトから仮想マシンの libvirt XML を書き込みます

A.11. VDSM フックの実行

before_vm_start スクリプトは、ドメイン XML を編集して、仮想マシンが libvirt に到達する前に仮想マシンの VDSM の定義を変更できます。その際には注意が必要です。フックスクリプトは VDSM の動作を混乱させる可能性があり、バグのあるスクリプトは Red Hat Virtualization 環境の停止につながる可能性があります。特に、ドメインの UUID を変更しないようにし、十分な背景知識がない限り、ドメインからデバイスを削除しようとしないでください。

before_vdsm_start と after_vdsm_stop の両方のフックスクリプトが root ユーザーとして実行されます。システムへの root アクセスを必要とするその他のフックスクリプトは、特権の昇格に **sudo** コマンドを使用するように作成する必要があります。これをサポートするには、/etc/sudoers を更新して、vdsm ユーザーがパスワードを再入力せずに **sudo** を使用できるようにする必要があります。これは、フックスクリプトが非対話的に実行されるために必要です。

例A.4 VDSM フックの sudo の設定

この例では、**sudo** コマンドは、vdsm ユーザーが root として /**bin**/**chown** コマンドを実行できるように設定されます。

1. root として仮想化ホストにログインします。
2. テキストエディターで /etc/sudoers ファイルを開きます。
3. 次の行をファイルに追加します。

```
vdsm ALL=(ALL) NOPASSWD: /bin/chown
```

これは、vdsm ユーザーが root ユーザーとして /**bin**/**chown** コマンドを実行できることを指定します。**NOPASSWD** パラメーターは、ユーザーが **sudo** を呼び出すときにパスワードの入力を求められないことを示しています。

この設定変更が行われると、VDSM フックは **sudo** コマンドを使用して /**bin**/**chown** を root として実行できるようになります。この Python コードは、**sudo** を使用して、ファイル /my_file の root として /**bin**/**chown** を実行します。

```
retcode = subprocess.call( ["/usr/bin/sudo", "/bin/chown", "root", "/my_file"] )
```

フックスクリプトの標準エラーストリームは、VDSM のログに収集されます。この情報は、フックスクリプトをデバッグするのに使用されます。

A.12. VDSM フックの戻りコード

フックスクリプトは、表A.3「フックリターンコード」 に表示される戻りコードの 1 つを返す必要があります。戻りコードは、さらにフックスクリプトが VDSM によって処理されるかどうかを判別します。

表A.3 フックリターンコード

コード	説明
0	フックスクリプトは正常に終了しました
1	フックスクリプトが失敗しました。他のフックを処理する必要があります
2	フックスクリプトが失敗しました。これ以上フックを処理する必要はありません
>2	予約

A.13. VDSM フックの例

このセクションで提供されているフックスクリプトの例は、Red Hat では厳密にはサポートされていません。ソースに関係なく、システムにインストールするすべてのフックスクリプトが、環境に対して徹底的にテストされていることを確認する必要があります。

例A.5 NUMA ノードのチューニング

目的:

このフックスクリプトを使用すると、**numaset** カスタムプロパティーに基づいて NUMA ホストのメモリー割り当てを調整できます。カスタムプロパティーが設定されていない場合、アクションは実行されません。

Configuration String:

```
numaset=^(interleave|strict|preferred):[\^]?\d+(-\d+)?(,[\^]?\d+(-\d+)?)*$
```

使用される正規表現により、特定の仮想マシンの **numaset** カスタムプロパティーで、割り当てモード (**interleave**、**strict**、**preferred**) と使用するノードの両方を指定できます。2 つの値はコロン (**:**) で区切られます。正規表現を使用すると、**nodeset** を次のように指定できます。

- 特定のノード (ノード 1 のみを使用することを指定する **numaset=strict:1**)、または
- ノードの範囲が使用される (ノード 1 から 4 が使用されることを指定する **numaset=strict:1-4**)、または
- 特定のノードが使用されていないこと (ノード 3 が使用されていないことを指定する **numaset = strict:^ 3**)、または
- 上記のコンマ区切りの組み合わせ (**numaset=strict:1-4,6** は、ノード 1 から 4、および 6 を使用することを指定します)。

スクリプト:

/usr/libexec/vdsm/hooks/before_vm_start/50_numa

```
#!/usr/bin/python

import os
import sys
import hooking
import traceback

'''
numa hook
=========
add numa support for domain xml:

<numatune>
   <memory mode="strict" nodeset="1-4,^3" />
</numatune>

memory=interleave|strict|preferred

numaset="1" (use one NUMA node)
```

```
numaset="1-4" (use 1-4 NUMA nodes)
numaset="^3" (don't use NUMA node 3)
numaset="1-4,^3,6" (or combinations)

syntax:
    numa=strict:1-4
'''

if os.environ.has_key('numa'):
    try:
        mode, nodeset = os.environ['numa'].split(':')

        domxml = hooking.read_domxml()

        domain = domxml.getElementsByTagName('domain')[0]
        numas = domxml.getElementsByTagName('numatune')

        if not len(numas) > 0:
            numatune = domxml.createElement('numatune')
            domain.appendChild(numatune)

            memory = domxml.createElement('memory')
            memory.setAttribute('mode', mode)
            memory.setAttribute('nodeset', nodeset)
            numatune.appendChild(memory)

            hooking.write_domxml(domxml)
        else:
            sys.stderr.write('numa: numa already exists in domain xml')
            sys.exit(2)
    except:
        sys.stderr.write('numa: [unexpected error]: %s\n' % traceback.format_exc())
        sys.exit(2)
```

付録B カスタムネットワークプロパティー

B.1. BRIDGE_OPTS パラメーターの説明

表B.1 bridge_opts パラメーター

パラメーター	説明
forward_delay	ブリッジがリスニング状態とラーニング状態で費やす時間をデシ秒単位で設定します。この時点でスイッチングループが検出されない場合、ブリッジは転送状態になります。これにより、通常のネットワーク操作の前に、ネットワークのトラフィックおよびレイアウトを検査する時間ができます。
gc_timer	ガベージコレクションの時間をデシ秒単位で設定します。その後、転送データベースがチェックされ、タイムアウトしたエントリーが消去されます。
group_addr	一般的なクエリーを送信する場合は、0 に設定します。グループ固有のクエリー、またはグループとソース固有のクエリーを送信する場合は、IP マルチキャストアドレスに設定します。
group_fwd_mask	ブリッジがリンクローカルグループアドレスを転送できるようにします。この値をデフォルトから変更すると、非標準のブリッジ動作が可能になります。
hash_elasticity	ハッシュテーブルで許可される最大チェーンの長さ。次の新しいマルチキャストグループが追加されるまで有効になりません。再ハッシュ後にこれが満たされない場合は、ハッシュの競合が発生し、スヌーピングが無効になります。
hash_max	ハッシュテーブルのバケット数の最大値。これはすぐに有効になり、現在のマルチキャストグループエントリーの数より少ない値に設定することはできません。値は 2 の累乗でなければなりません。
hello_time	hello メッセージを送信してからネットワークトポロジー内のブリッジの位置を通知するまでの時間間隔をデシ秒単位で設定します。このブリッジがスパニングツリールートブリッジである場合にのみ適用されます。
hello_timer	最後の hello メッセージが送信されてからの時間 (デシ秒単位)。

パラメーター	説明
max_age	他のルートブリッジから hello メッセージを受け取ってから、そのブリッジがデッドとなったとみなされ、引き継ぎが開始されるまでの最大時間をデシ秒単位で設定します。
multicast_last_member_count	ホストから leave group メッセージを受け取った後、マルチキャストグループに送信する last member クエリーの回数を設定します。
multicast_last_member_interval	最後のメンバークエリー間の時間をデシ秒単位で設定します。
multicast_membership_interval	ブリッジがホストへのマルチキャストトラフィックの送信を停止する前に、ブリッジがマルチキャストグループのメンバーからの応答を待機する時間をデシ秒単位で設定します。
multicast_querier	ブリッジがマルチキャストクエリーをアクティブに実行するかどうかを設定します。ブリッジが他のネットワークホストからマルチキャストホストメンバーシップクエリーを受信すると、そのホストはクエリーを受け取った時刻にマルチキャストクエリー間隔を加えた時間に基づいて追跡されます。ブリッジが後でそのマルチキャストメンバーシップのトラフィックを転送しようとした場合、またはクエリーを実行しているマルチキャストルーターと通信している場合は、このタイマーはクエリーアの有効性を確認します。有効な場合、マルチキャストトラフィックは、ブリッジの既存のマルチキャストメンバーシップテーブルを介して配信されます。有効でなくなると、トラフィックはすべてのブリッジポートを介して送信されます。マルチキャストメンバーシップを持つ、またはマルチキャストメンバーシップを期待しているブロードキャストドメインは、パフォーマンスを向上させるために少なくとも 1 つのマルチキャストクエリーアを実行する必要があります。
multicast_querier_interval	ホストから受け取った最後のマルチキャストホストメンバーシップクエリー間の最大時間をデシ秒単位で設定して、それがまだ有効であることを確認します。
multicast_query_use_ifaddr	ブール値。デフォルトは 0 です。この場合、クエリーアは IPv4 メッセージの送信元アドレスとして 0.0.0.0 を使用します。これを変更すると、ブリッジ IP が送信元アドレスとして設定されます。

パラメーター	説明
multicast_query_interval	マルチキャストメンバーシップの有効性を確保するために、ブリッジによって送信されるクエリーメッセージ間の時間をデシ秒単位で設定します。このとき、またはブリッジがそのメンバーシップのマルチキャストクエリーを送信するように要求された場合、ブリッジは、チェックが要求された時間と multicast_query_interval に基づいて、自身のマルチキャストクエリーアの状態をチェックします。このメンバーシップのマルチキャストクエリーが最後の multicast_query_interval 内に送信された場合、それは再度送信されません。
multicast_query_response_interval	送信されたクエリーにホストが応答できる時間の長さ (デシ秒)。multicast_query_interval の値以下である必要があります。
multicast_router	マルチキャストルーターが接続されているポートの有効/無効を設定します。1つ以上のマルチキャストルーターを備えたポートは、すべてのマルチキャストトラフィックを受信します。値 0 は完全に無効になり、値 1 はシステムがクエリーに基づいてルーターの存在を自動的に検出できるようにし、値 2 はポートが常にすべてのマルチキャストトラフィックを受信できるようにします。
multicast_snooping	スヌーピングの有効/無効を切り替えます。スヌーピングを使用すると、ブリッジはルーターとホスト間のネットワークトラフィックをリッスンして、適切なリンクへのマルチキャストトラフィックをフィルターリングするためのマップを維持できます。このオプションを使用すると、ユーザーは、ハッシュの衝突によって自動的に無効になった場合にスヌーピングを再度有効にできます。ハッシュの衝突が解決されていない場合は、再度有効にしないでください。
multicast_startup_query_count	メンバーシップ情報を決定するのに起動時に送信されるクエリーの数を設定します。
multicast_startup_query_interval	メンバーシップ情報を決定するために起動時に送信されるクエリー間の時間をデシ秒単位で設定します。

B.2. RED HAT VIRTUALIZATION MANAGER を使用するように RED HAT VIRTUALIZATION MANAGER を設定する方法

管理ポータルから、ホストネットワークインターフェイスカードの ethtool プロパティーを設定できま

す。**ethtool_opts** キーはデフォルトでは使用できないため、エンジン設定ツールを使用して Manager に追加する必要があります。ホストに必要な VDSM フックパッケージもインストールする必要があります。

ethtool_opts キーの Manager への追加

1. Manager で以下のコマンドを実行してキーを追加します。

```
# engine-config -s UserDefinedNetworkCustomProperties=ethtool_opts=.* --cver=4.0
```

2. **ovirt-engine** サービスを再起動します。

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

3. ethtool プロパティーを設定するホストに、VDSM フックパッケージをインストールします。Red Hat Virtualization Host ではこのパッケージがデフォルトで利用可能ですが、Red Hat Enterprise Linux ホストにインストールする必要があります。

```
# yum install vdsm-hook-ethtool-options
```

ethtool_opts キーが管理ポータルで利用できるようになりました。ethtool プロパティーを論理ネットワークに適用するには、「ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て」 を参照してください。

B.3. FCOE を使用するように RED HAT VIRTUALIZATION MANAGER を設定する方法

管理ポータルから、ホストネットワークインターフェイスカードの Fibre Channel over Ethernet (FCoE) プロパティーを設定できます。**fcoe** キーはデフォルトでは使用できないため、エンジン設定ツールを使用して Manager に追加する必要があります。次のコマンドを実行して、**fcoe** がすでに有効になっているかどうかを確認できます。

```
# engine-config -g UserDefinedNetworkCustomProperties
```

ホストに必要な VDSM フックパッケージもインストールする必要があります。ホストの FCoE カードによっては、特別な設定が必要になる場合があります。**Red Hat Enterprise Linux Storage Administration Guide** の Configuring a Fibre Channel over Ethernet Interface を参照してください。

Manager への fcoe キーの追加

1. Manager で以下のコマンドを実行してキーを追加します。

```
# engine-config -s UserDefinedNetworkCustomProperties='fcoe=^((enable|dcb|auto_vlan)=(yes|no),?)*$'
```

2. **ovirt-engine** サービスを再起動します。

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

3. FCoE プロパティーを設定する各 Red Hat Enterprise Linux ホストに VDSM フックパッケージをインストールします。Red Hat Virtualization Host (RHVH) では、デフォルトでパッケージが利用可能です。

```
# yum install vdsm-hook-fcoe
```

fcoe キーが管理ポータルで使用できるようになりました。FCoE プロパティーを論理ネットワークに適用するには、「ホストのネットワークインターフェイスの編集とホストへの論理ネットワークの割り当て」 を参照してください。

付録C RED HAT VIRTUALIZATION ユーザーインターフェイスプラグイン

C.1. RED HAT VIRTUALIZATION ユーザーインターフェイスプラグイン

Red Hat Virtualization は、非標準の機能を公開するプラグインをサポートしています。これにより、Red Hat Virtualization 管理ポータルを使用して他のシステムと統合することが容易になります。各インターフェイスプラグインは、Red Hat Virtualization で使用するためにパッケージ化および配布できるユーザーインターフェイス拡張機能のセットを表します。

Red Hat Virtualization のユーザーインターフェイスプラグインは、JavaScript プログラミング言語を使用して、クライアント上で直接管理ポータルと統合されます。プラグインは管理ポータルにより呼び出され、Web ブラウザーの JavaScript ランタイムで実行されます。User Interface プラグインは、JavaScript 言語とそのライブラリーを使用することができます。

実行時の主要なイベントで、管理ポータルは、管理ポータルからプラグインへの通信を表すイベントハンドラー関数を介して個々のプラグインを呼び出します。管理ポータルは複数のイベントハンドラー関数をサポートしていますが、プラグインはその実装にのみ関係する関数を宣言します。各プラグインは、プラグインを管理ポータルで使用する前に、関連するイベントハンドラー関数をプラグインブートストラップシーケンスの一部として登録する必要があります。

ユーザーインターフェイス拡張機能を駆動するプラグインから管理ポータルへの通信を容易にするために、管理ポータルはプラグイン API を個々のプラグインが使用できるグローバル (トップレベル) の pluginApi JavaScript オブジェクトとして公開します。各プラグインは個別の pluginApi インスタンスを取得し、管理ポータルがプラグインのライフサイクルに関して各プラグインのプラグイン API 関数の呼び出しを制御できるようにします。

C.2. RED HAT VIRTUALIZATION USER INTERFACE PLUGIN LIFECYCLE

C.2.1. Red Hat Virtualization User Interface Plug-in のライフサイクル

User Interface Plug-in の基本的なライフサイクルは、3 つのステージに分けられます。

- プラグインの検出。
- プラグインの読み込み。
- プラグインブートストラップ。

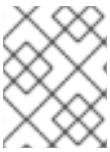
C.2.2. Red Hat Virtualization ユーザーインターフェイスプラグインの検出

プラグイン記述子の作成は、プラグイン検出プロセスの最初のステップです。プラグイン記述子には、重要なプラグインメタデータおよびオプションのデフォルトのプラグイン固有の設定が含まれています。

管理ポータルの HTML ページ要求 (**HTTP GET**) の処理の一部として、ユーザーインターフェイスプラグインインフラストラクチャーは、ローカルファイルシステムからプラグイン記述子を検出してロードしようとします。プラグイン記述子ごとに、インフラストラクチャーは、デフォルトのプラグイン固有の設定 (存在する場合) をオーバーライドし、プラグインの実行時の動作を微調整するのに使用される、対応するプラグインユーザー設定もロードしようとします。プラグインのユーザー設定は任意です。記述子と対応するユーザー設定ファイルをロードした後、oVirt Engine はユーザーインターフェイスプラグインデータを集約し、ランタイム評価のために管理ポータルの HTML ページに埋め込みます。

デフォルトでは、プラグイン記述子は **$ENGINE_USR/ui-plug-ins** にあり、oVirt Engine ローカル設定で定義されている **ENGINE_USR=/usr/share/ovirt-engine** のデフォルトマッピングがあります。プラグイン記述子は JSON 形式の仕様に準拠することが期待されていますが、プラグイン記述子では、JSON 形式の仕様に加えて (/* と // の両方の) Java/C++ スタイルのコメントを使用できます。

デフォルトでは、プラグインユーザー設定ファイルは **$ENGINE_ETC/ui-plug-ins** にあり、oVirt Engine ローカル設定で定義されている **ENGINE_ETC=/etc/ovirt-engine** のデフォルトマッピングがあります。プラグインのユーザー設定ファイルは、プラグイン記述子と同じコンテンツ形式の規則に準拠する必要があります。



注記

プラグインのユーザー設定ファイルは、通常、**<descriptorFileName>-config.json** の命名規則に従います。

C.2.3. Red Hat Virtualization User Interface Plug-in のロード

プラグインが検出され、そのデータが管理ポータルの HTML ページに埋め込まれた後、管理ポータルは、アプリケーションの起動の一部としてプラグインを読み込もうとします (アプリケーションの起動の一部として読み込まれないように設定した場合を除く)。

検出されたプラグインごとに、管理ポータルはホストページの読み込みに使用される HTML iframe 要素を作成します。プラグインホストページは、プラグインブートストラッププロセスを開始するために必要です。このプロセス (ブートストラッププロセス) は、プラグインの iframe 要素のコンテキストでプラグインコードを評価するのに使用されます。ユーザーインターフェイスプラグインインフラストラクチャーは、ローカルファイルシステムからのプラグインリソースファイル (プラグインホストページなど) の提供をサポートします。プラグインホストページが iframe 要素に読み込まれ、プラグインコードが評価されます。プラグインコードが評価された後、プラグインはプラグイン API を使用して管理ポータルと通信します。

C.2.4. Red Hat Virtualization ユーザーインターフェイスプラグインブートストラップ

一般的なプラグインブートストラップシーケンスは、次の手順で設定されます。

プラグインブートストラップシーケンス

1. 指定されたプラグインの pluginApi インスタンスを取得します
2. ランタイムプラグイン設定オブジェクトを取得 (オプション)
3. 関連するイベントハンドラー関数の登録
4. UI プラグインインフラストラクチャーにプラグインの初期化を進めるよう通知します。

次のコードは、上記の手順を実際に示すものです。

```
// Access plug-in API using 'parent' due to this code being evaluated within the context of an iframe element.
// As 'parent.pluginApi' is subject to Same-Origin Policy, this will only work when WebAdmin HTML page and plug-in
// host page are served from same origin. WebAdmin HTML page and plug-in host page will always be on same origin
// when using UI plug-in infrastructure support to serve plug-in resource files.
var api = parent.pluginApi('MyPlugin');
```

```
// Runtime configuration object associated with the plug-in (or an empty object).
var config = api.configObject();

// Register event handler function(s) for later invocation by UI plug-in infrastructure.
api.register({
    // UiInit event handler function.
  UiInit: function() {
    // Handle UiInit event.
    window.alert('Favorite music band is ' + config.band);
        }
});

// Notify UI plug-in infrastructure to proceed with plug-in initialization.
api.ready();
```

C.3. ユーザーインターフェイスプラグイン関連のファイルとその場所

表C.1 UI プラグイン関連のファイルとその場所

File	場所	備考
プラグイン記述子ファイル (メタデータ)	**/usr/share/ovirt-engine/ui-plugins/my-plugin.json**	
プラグインユーザー設定ファイル	**/etc/ovirt-engine/ui-plugins/my-plugin-config.json**	
プラグインリソースファイル	**/usr/share/ovirt-engine/ui-plugins/<resourcePath>/PluginHostPage.html**	**<resourcePath>** は、プラグイン記述子の対応する属性によって定義されます。

C.4. ユーザーインターフェイスプラグインのデプロイメント例

次の手順に従って、**Hello World!** を実行するユーザーインターフェイスプラグインを作成します。Red Hat Virtualization Manager 管理ポータルにサインインするときにプログラムします。

Hello World! のデプロイ Plug-in

1. **/usr/share/ovirt-engine/ui-plugins/helloWorld.json** の Manager で次のファイルを作成して、プラグイン記述子を作成します。

```
{
    "name": "HelloWorld",
    "url": "/ovirt-engine/webadmin/plugin/HelloWorld/start.html",
    "resourcePath": "hello-files"
}
```

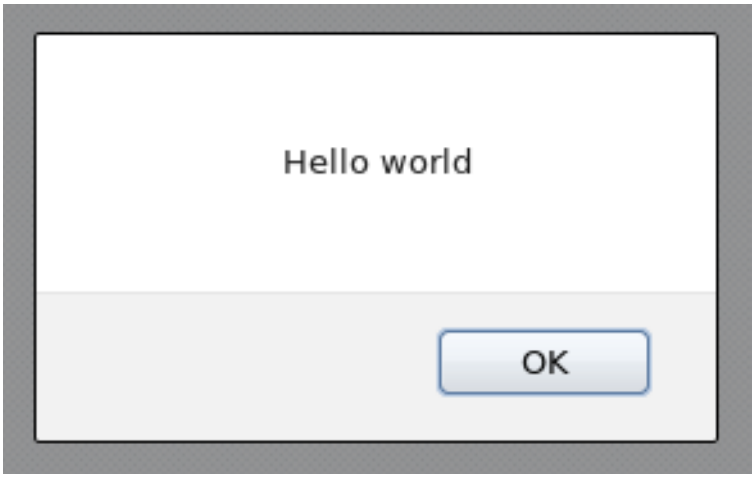
2. **/usr/share/ovirt-engine/ui-plugins/hello-files/start.html** の Manager で次のファイルを作成して、プラグインホストページを作成します。

```
<!DOCTYPE html><html><head>
<script>
```

```
    var api = parent.pluginApi('HelloWorld');
    api.register({
 UiInit: function() { window.alert('Hello world'); }
    });
    api.ready();
</script>
</head><body></body></html>
```

Hello World! プラグインが正常に実装されている場合は、管理ポータルにログインすると、この画面が表示されます。

図C.1Hello World! の実装の成功プラグイン



付録D RED HAT VIRTUALIZATION および暗号化された通信

D.1. RED HAT VIRTUALIZATION MANAGER CA 証明書の置き換え

警告

/etc/pki ディレクトリーまたはサブディレクトリーの権限と所有権を変更しないでください。**/etc/pki** および **/etc/pki/ovirt-engine** ディレクトリーの権限は、デフォルトの **755** のままにする必要があります。

HTTPS 経由で接続しているユーザーに対して Red Hat Virtualization Manager を識別するように、組織のサードパーティー CA 証明書を設定できます。

注記

HTTPS 接続にサードパーティーの CA 証明書を使用しても、Manager とホストとの間の認証に使用される証明書には影響しません。マネージャーによって生成された自己署名証明書を引き続き使用します。

前提条件

- サードパーティーの CA 証明書。これは、使用する証明書を発行した CA(認証局) の証明書です。**PEM** ファイルとして提供されます。証明書チェーンは、ルート証明書まで完全である必要があります。チェーンの順序は重要であり、最後の中間証明書からルート証明書までででなければなりません。この手順は、サードパーティーの CA 証明書が **/tmp/3rd-party-ca-cert.pem** で提供されていることを前提としています。
- Apache httpd で使用する秘密鍵パスワードを含めることはできません。この手順では、**/tmp/apache.key** にあることを前提としています。
- CA が発行する証明書。この手順では、**/tmp/apache.cer** にあることを前提としています。

CA から秘密鍵と証明書を P12 ファイルで受け取った場合は、次の手順を使用してそれらを抽出します。その他のファイル形式については、CA にお問い合わせください。秘密鍵と証明書を抽出した後、Red Hat Virtualization Manager Apache CA 証明書の置き換え に進みます。

P12 バンドルからの証明書および秘密鍵の抽出

内部 CA は、内部で生成されたキーおよび証明書を **/etc/pki/ovirt-engine/keys/apache.p12** の P12 ファイルに保存します。Red Hat では、新しいファイルを同じ場所に保存することを推奨します。以下の手順では、新しい P12 ファイルが **/tmp/apache.p12** にあると仮定しています。

1. 現在の **apache.p12** ファイルをバックアップします。

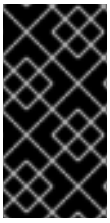
```
# cp -p /etc/pki/ovirt-engine/keys/apache.p12 /etc/pki/ovirt-engine/keys/apache.p12.bck
```

2. 現在のファイルを新しいファイルに置き換えます。

```
# cp /tmp/apache.p12 /etc/pki/ovirt-engine/keys/apache.p12
```

3. 秘密鍵と証明書を必要な場所に抽出します。ファイルがパスワードで保護されている場合は、**-passin pass:_password_** を追加し、**password** を必要なパスワードに置き換える必要があります。

```
# openssl pkcs12 -in /etc/pki/ovirt-engine/keys/apache.p12 -nocerts -nodes >
/tmp/apache.key
# openssl pkcs12 -in /etc/pki/ovirt-engine/keys/apache.p12 -nokeys > /tmp/apache.cer
```



重要

Red Hat Virtualization の新規インストールでは、この手順のすべてのステップを完了する必要があります。商用署名証明書が設定されている Red Hat Enterprise Virtualization 3.6 環境からアップグレードした場合は、実行する必要がある手順は 1、8、および 9 のみです。

Red Hat Virtualization Manager の Apache CA 証明書の置き換え

1. セルフホストエンジンを使用している場合は、環境をグローバルメンテナーンスモードにします。

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

詳細は、「セルフホストエンジンの保守」 を参照してください。

2. CA 証明書をホスト全体のトラストストアに追加します。

```
# cp /tmp/3rd-party-ca-cert.pem /etc/pki/ca-trust/source/anchors
# update-ca-trust
```

3. Manager は、**/etc/pki/ovirt-engine/ca.pem** にシンボリックリンクされている **/etc/pki/ovirt-engine/apache-ca.pem** を使用するように設定されています。シンボリックリンクを削除します。

```
# rm /etc/pki/ovirt-engine/apache-ca.pem
```

4. CA 証明書を **/etc/pki/ovirt-engine/apache-ca.pem** として保存します。

```
# cp /tmp/3rd-party-ca-cert.pem /etc/pki/ovirt-engine/apache-ca.pem
```

5. 既存の秘密鍵と証明書をバックアップします。

```
# cp /etc/pki/ovirt-engine/keys/apache.key.nopass /etc/pki/ovirt-
engine/keys/apache.key.nopass.bck
# cp /etc/pki/ovirt-engine/certs/apache.cer /etc/pki/ovirt-engine/certs/apache.cer.bck
```

6. 秘密鍵を必要な場所にコピーします。

```
# cp /tmp/apache.key /etc/pki/ovirt-engine/keys/apache.key.nopass
```

7. 秘密鍵の所有者を root に設定し、パーミッションを **0640** に設定します。

```
# chown root:ovirt  /etc/pki/ovirt-engine/keys/apache.key.nopass
# chmod 640 /etc/pki/ovirt-engine/keys/apache.key.nopass
```

8. 証明書を必要な場所にコピーします。

```
# cp /tmp/apache.cer /etc/pki/ovirt-engine/certs/apache.cer
```

9. Apache サーバーを再起動します。

```
# systemctl restart httpd.service
```

10. 次のパラメーターを使用して、新しいトラストストア設定ファイル **/etc/ovirt-engine/engine.conf.d/99-custom-truststore.conf** を作成します。

```
ENGINE_HTTPS_PKI_TRUST_STORE="/etc/pki/java/cacerts"
ENGINE_HTTPS_PKI_TRUST_STORE_PASSWORD=""
```

11. **/etc/ovirt-engine/ovirt-websocket-proxy.conf.d/10-setup.conf** ファイルをコピーし、10 より大きいインデックス番号 (たとえば、**99-setup.conf**) に名前を変更します。新しいファイルに以下のパラメーターを追加します。

```
SSL_CERTIFICATE=/etc/pki/ovirt-engine/certs/apache.cer
SSL_KEY=/etc/pki/ovirt-engine/keys/apache.key.nopass
```

12. **websocket-proxy** サービスを再起動します。

```
# systemctl restart ovirt-websocket-proxy.service
```

13. **/etc/ovirt-provider-ovn/conf.d/10-setup-ovirt-provider-ovn.conf** ファイルを手動で変更した場合、または古いインストールの設定ファイルを使用している場合は、Manager がまだ、**/etc/pki/ovirt-engine/apache-ca.pem** を証明書ソースとして使用するように設定されていることを確認してください。

14. 次の内容の新しいファイル **/etc/ovirt-engine-backup/engine-backup-config.d/update-system-wide-pki.sh** を作成して、**engine-backup** を有効にして復元時にシステムを更新します。

```
BACKUP_PATHS="${BACKUP_PATHS}
/etc/ovirt-engine-backup"
cp -f /etc/pki/ovirt-engine/apache-ca.pem
/etc/pki/ca-trust/source/anchors/3rd-party-ca-cert.pem
update-ca-trust
```

15. **ovirt-provider-ovn** サービスを再起動します。

```
# systemctl restart ovirt-provider-ovn.service
```

16. **ovirt-engine** サービスを再起動します。

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

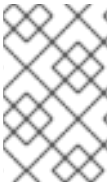
17. セルフホストエンジンを使用している場合は、グローバルメンテナーンスモードをオフにします。

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

これで、HTTPS トラフィックの暗号化に使用される証明書の信頼性に関する警告が表示されることなく、ユーザーは管理ポータルと VM ポータルに接続できます。

D.2. MANAGER と LDAP サーバー間の暗号化通信の設定

Red Hat Virtualization Manager と LDAP サーバー間の暗号化された通信をセットアップするには、LDAP サーバーのルート CA 証明書を取得し、ルート CA 証明書を Manager にコピーして、PEM でエンコードされた CA 証明書を作成します。キーストアタイプは、Java でサポートされている任意のタイプになります。以下の手順では、Java KeyStore (JKS) 形式を使用します。



注記

PEM でエンコードされた CA 証明書の作成および証明書のインポートに関する詳細は、**/usr/share/doc/ovirt-engine-extension-aaa-ldap-version** にある README ファイルの **X.509 CERTIFICATE TRUST STORE** セクションを参照してください。

PEM でエンコードされた CA 証明書の作成

1. Red Hat Virtualization Manager で、LDAP サーバーの root CA 証明書を **/tmp** ディレクトリーにコピーし、**keytool** を使用して root CA 証明書をインポートして、PEM でエンコードされた CA 証明書を作成します。以下のコマンドは、**/tmp/myrootca.pem** の root CA 証明書をインポートし、**/etc/ovirt-engine/aaa/** の下に PEM でエンコードされた CA 証明書 **myrootca.jks** を作成します。証明書の場所とパスワードを書き留めます。インタラクティブセットアップツールを使用している場合は、これが必要なすべての情報です。LDAP サーバーを手動で設定している場合は、残りの手順に従って設定ファイルを更新してください。

```
$ keytool -importcert -noprompt -trustcacerts -alias myrootca -file /tmp/myrootca.pem -
keystore /etc/ovirt-engine/aaa/myrootca.jks -storepass password
```

2. **/etc/ovirt-engine/aaa/profile1.properties** ファイルを証明書情報で更新します。

注記

${local:_basedir}** は、LDAP プロパティー設定ファイルが存在するディレクトリーであり、**/etc/ovirt-engine/aaa** ディレクトリーを指します。PEM でエンコードされた CA 証明書を別のディレクトリーに作成した場合は、**${local:_basedir} を証明書へのフルパスに置き換えます。

- startTLS (推奨) を使用するには、以下を行います。

```
# Create keystore, import certificate chain and uncomment
pool.default.ssl.startTLS = true
pool.default.ssl.truststore.file = ${local:_basedir}/myrootca.jks
pool.default.ssl.truststore.password = password
```

- SSL を使用するには、以下を行います。

```
# Create keystore, import certificate chain and uncomment
```

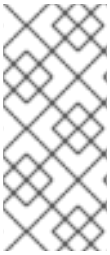
```
pool.default.serverset.single.port = 636
pool.default.ssl.enable = true
pool.default.ssl.truststore.file = ${local:_basedir}/myrootca.jks
pool.default.ssl.truststore.password = password
```

外部 LDAP プロバイダーの設定を続行するには、Configuring an External LDAP Provider を参照してください。シングルサインオン用に LDAP および Kerberos の設定を続行するには、Configuring LDAP and Kerberos for Single Sign-on を参照してください。

D.3. VDSM の暗号化通信の手動設定

VDSM の暗号化された通信を Manager や他の VDSM インスタンスに手動で設定できます。

クラスターレベル 3.6、4.0、および 4.1 のクラスターのホストのみには、手動設定が必要です。レベル 4.2 のクラスター内のホストは、ホストの再インストール中に自動的に暗号化の強化の再設定が行われます。



注記

RHVH 3.6、4.0、および 4.1 のホストは、強力な暗号化に対応していません。RHVH 4.2 および RHEL ホストは対応します。

RHVH 4.2 ホストを使用する 3.6、4.0、または 4.1 クラスターがある場合には、強力な暗号化を使用できます。

手順

1. **Compute → Hosts** をクリックし、ホストを選択します。
2. **管理 → メンテナーンス** をクリックして、**メンテナーンスホスト** の確認ウィンドウを開きます。
3. **OK** をクリックしてメンテナーンスモードを開始します。
4. ホストで、**/etc/vdsm/vdsm.conf.d/99-custom-ciphers.conf** を作成して、以下の設定を指定します。

   ```
   [vars]
   ssl_ciphers = HIGH
   ```

   詳細は、OpenSSL Cipher Strings を参照してください。
5. VDSM を再起動します。

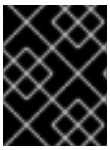
   ```
   # systemctl restart vdsm
   ```
6. **Compute → Hosts** をクリックし、ホストを選択します。
7. **Management → Activate** をクリックして、ホストを再度アクティブにします。

付録E プロキシー

E.1. SPICE プロキシー

E.1.1. SPICE プロキシーの概要

SPICE プロキシーは、SPICE クライアントがハイパーバイザーを接続するネットワークの外部にある場合に SPICE クライアントを仮想マシンに接続するために使用されるツールです。SPICE プロキシーの設定は、マシンに **Squid** をインストールし、プロキシートラフィックを許可するようにファイアウォールを設定することで設定されます。SPICE プロキシーをオンにするには、Manager で **engine-config** を使用して、キー **SpiceProxyDefault** をプロキシーの名前およびポートで設定される値に設定します。SPICE プロキシーをオフにするには、Manager で **engine-config** を使用して、キー **SpiceProxyDefault** が設定されている値を削除します。



重要

SPICE プロキシーは、スタンドアロン SPICE クライアントと組み合わせてのみ使用でき、noVNC を使用して仮想マシンに接続するために使用することはできません。

E.1.2. SPICE プロキシーマシンの設定

この手順では、マシンを SPICE プロキシーとして設定する方法について説明します。SPICE プロキシーを使用すると、ネットワークの外部から Red Hat Virtualization ネットワークに接続できます。この手順で **Squid** を使用してプロキシーサービスを提供します。

Red Hat Enterprise Linux へのSquid のインストール

1. プロキシーマシンに **Squid** をインストールします。

```
# yum install squid
```

2. **/etc/squid/squid.conf** を開きます。以下を、

```
http_access deny CONNECT !SSL_ports
```

以下のように変更します。

```
http_access deny CONNECT !Safe_ports
```

3. squid サービスを起動し、再起動後に自動的に実行されるようにします。

```
# systemctl enable squid.service --now
```

4. デフォルトの firewalld ゾーンで squid サービスへの着信リクエストを有効にします。

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

5. このファイアウォールルールをランタイム設定で永続的にします。

```
# firewall-cmd --reload
```

6. ファイアウォールのサービス一覧に squid サービスが表示されていることを確認します。

```
# firewall-cmd --list-services
ssh dhcpv6-client squid
```

これで、マシンを SPICE プロキシーとして設定できました。ネットワークの外部から Red Hat Virtualization ネットワークに接続する前に、SPICE プロキシーをアクティブ化します。

E.1.3. SPICE プロキシーをオンにする

この手順では、SPICE プロキシーをアクティブ化 (またはオン) する方法について説明します。

SPICE プロキシーのアクティブ化

1. Manager で engine-config ツールを使用してプロキシーを設定します。

```
# engine-config -s SpiceProxyDefault=someProxy
```

2. **ovirt-engine** サービスを再起動します。

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

プロキシーには以下の形式が必要です。

```
protocol://[host]:[port]
```

注記

Red Hat Enterprise Linux 6.7、Red Hat Enterprise Linux 7.2 以降に同梱されている SPICE クライアントのみが、HTTPS プロキシーをサポートします。以前のクライアントは HTTP のみをサポートしています。以前のクライアントに HTTPS が指定されている場合、クライアントはプロキシー設定を無視し、ホストへの直接接続を試みます。

SPICE Proxy がアクティブ (オン) になりました。SPICE プロキシーを介して Red Hat Virtualization ネットワークに接続できるようになりました。

E.1.4. SPICE プロキシーをオフにする

この手順では、SPICE プロキシーをオフに (非アクティブ化) する方法を説明します。

SPICE プロキシーをオフにする

1. Manager にログインします。

```
$ ssh root@[IP of Manager]
```

2. 以下のコマンドを実行し、SPICE プロキシーをクリアします。

```
# engine-config -s SpiceProxyDefault=""
```

3. Manager を再起動します。

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

SPICE プロキシーが非アクティブ (オフ) になりました。SPICE プロキシーを介して Red Hat Virtualization ネットワークに接続できなくなりました。

E.2. SQUID プロキシー

E.2.1. Squid プロキシーのインストールおよび設定

本セクションでは、VM ポータルに Squid プロキシーをインストールして設定する方法を説明します。Squid プロキシーサーバーは、コンテンツアクセラレーターとして使用されます。頻繁にビューされたコンテンツをキャッシュし、帯域幅を削減し、応答時間を改善します。

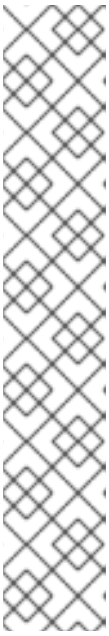
Squid プロキシーの設定

1. Squid プロキシーサーバーの HTTPS ポートのキーペアと証明書を取得します。このキーペアは、別の SSL/TLS サービスのキーペアを取得するのと同じ方法で取得できます。キーペアは、秘密キーと署名付き証明書を含む 2 つの PEM ファイルの形式です。この手順では、**proxy.key** および **proxy.cer** という名前が付けられていることを前提としています。

注記

キーペアと証明書は、エンジンの認証局を使用して生成することもできます。プロキシーの秘密鍵および証明書がすでにあり、エンジン認証局でそれを生成したくない場合は、次の手順にスキップしてください。

2. プロキシーのホスト名を選択します。次に、プロキシーの証明書の識別名の他のコンポーネントを選択します。



注記

エンジン自体が使用するのと同じ国と同じ組織名を使用することが推奨されます。Manager がインストールされているマシンにログインし、次のコマンドを実行して、この情報を見つけます。

```
# openssl x509 -in /etc/pki/ovirt-engine/ca.pem -noout -subject
```

このコマンドは次のようなものを出力します。

```
subject= /C=US/O=Example Inc./CN=engine.example.com.81108
```

ここでの関連部分は**/C=US/O=Example Inc.** です。これを使用して、プロキシーの証明書の完全な識別名を作成します。

```
/C=US/O=Example Inc./CN=proxy.example.com
```

3. プロキシーマシンにログインし、証明書署名要求を生成します。

```
# openssl req -newkey rsa:2048 -subj '/C=US/O=Example Inc./CN=proxy.example.com' -
nodes -keyout proxy.key -out proxy.req
```

重要

証明書の識別名の周りにある引用符を含める必要があります。**-nodes** オプションは、秘密鍵が暗号化されていないことを保証します。これは、プロキシーサーバーを起動するためにパスワードを入力する必要がないことを意味します。

このコマンドは、**proxy.key** および **proxy.req** の 2 つのファイルを生成します。**proxy.key** は秘密鍵です。このファイルを安全に保持します。**proxy.req** は証明書署名要求です。**proxy.req** には、特別な保護は必要ありません。

4. 署名付き証明書を生成するには、証明書署名要求ファイルをプロキシーマシンからマネージャーマシンにコピーします。

```
# scp proxy.req engine.example.com:/etc/pki/ovirt-engine/requests/.
```

5. Manager マシンにログインし、証明書に署名します。

```
# /usr/share/ovirt-engine/bin/pki-enroll-request.sh --name=proxy --days=3650 --
subject='/C=US/O=Example Inc./CN=proxy.example.com'
```

これにより、証明書が署名され、10 年間 (3650 日) 有効になります。必要に応じて、証明書の有効期限が早くなるように設定します。

6. 生成された証明書ファイルは **/etc/pki/ovirt-engine/certs** ディレクトリーで利用可能で、**proxy.cer** という名前を指定する必要があります。プロキシーマシンで、このファイルを Manager マシンから現在のディレクトリーにコピーします。

```
# scp engine.example.com:/etc/pki/ovirt-engine/certs/proxy.cer .
```

7. **proxy.key** と **proxy.cer** の両方がプロキシーマシンに存在することを確認します。

```
# ls -l proxy.key proxy.cer
```

8. Squid プロキシーサーバーパッケージをプロキシーマシンにインストールします。

```
# yum install squid
```

9. 秘密鍵と署名済み証明書を、プロキシーがアクセスできる場所 (例: **/etc/squid** ディレクトリー) に移動します。

```
# cp proxy.key proxy.cer /etc/squid/.
```

10. **squid** ユーザーが、これらのファイルを読み取れるようにパーミッションを設定します。

```
# chgrp squid /etc/squid/proxy.*
# chmod 640 /etc/squid/proxy.*
```

11. Squid プロキシーは、エンジンが使用する証明書を検証する必要があります。Manager 証明書をプロキシーマシンにコピーします。この例では、ファイルパス **/etc/squid** を使用します。

```
# scp engine.example.com:/etc/pki/ovirt-engine/ca.pem /etc/squid/.
```

注記

デフォルトの CA 証明書は、Manager マシンの **/etc/pki/ovirt-engine/ca.pem** にあります。

12. **squid** ユーザーが、証明書ファイルを読み取れるようにパーミッションを設定します。

```
# chgrp squid /etc/squid/ca.pem
# chmod 640 /etc/squid/ca.pem
```

13. SELinux が Enforcing モードの場合は、**semanage** ツールを使用して、ポート 443 のコンテキストを変更して、Squid がポート 443 を使用できるようにします。

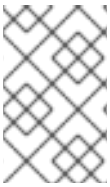
```
# yum install policycoreutils-python
# semanage port -m -p tcp -t http_cache_port_t 443
```

14. 既存の Squid 設定ファイルを以下に置き換えます。

```
https_port 443 key=/etc/squid/proxy.key cert=/etc/squid/proxy.cer ssl-bump
defaultsite=engine.example.com
cache_peer engine.example.com parent 443 0 no-query originserver ssl
sslcafile=/etc/squid/ca.pem name=engine login=PASSTHRU
cache_peer_access engine allow all
ssl_bump allow all
http_access allow all
```

15. Squid プロキシーサーバーを再起動します。

```
# systemctl restart squid.service
```



注記

デフォルト設定の Squid プロキシーは、15 分のアイドル時間後に接続を終了します。Squid Proxy がアイドル状態の接続を終了するまでの時間を増やすには、**squid.conf** の **read_timeout** オプションを調整します (例: **read_timeout 10 hours**)。

E.3. WEBSOCKET プロキシー

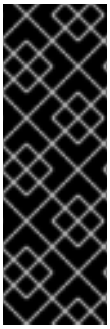
E.3.1. WebSocket プロキシーの概要

Websocket プロキシーにより、ユーザーは noVNC コンソールを介して仮想マシンに接続することができます。

WebSocket プロキシーは、初期設定時に Red Hat Virtualization Manager マシンにも (Red Hat Virtualization Manager の設定 を参照)、別のマシンにも (別のマシンへの Websocket プロキシーのインストールを 参照) インストールして設定することができます。

Websocket プロキシーは、Manager マシンから別のマシンに移行することもできます。「Websocket プロキシーを別のマシンに移行する」を参照してください。

E.3.2. Websocket プロキシーを別のマシンに移行する



重要

Websocket プロキシーおよび noVNC は、テクノロジープレビュー機能としてのみ提供されています。テクノロジープレビューの機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat では実稼働環境での使用を推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、開発プロセスの中でお客様に機能性のテストとフィードバックをしていただくことを目的としています。詳しい情報は、テクノロジープレビュー機能のサポート範囲 を参照してください。

セキュリティーまたはパフォーマンス上の理由から、WebSocket プロキシーは Red Hat Virtualization Manager を実行しない別のマシンで実行できます。WebSocket プロキシーを Manager マシンから別のマシンに移行する手順では、Manager マシンから WebSocket プロキシー設定を削除してから、別のマシンにプロキシーをインストールします。

engine-cleanup コマンドを使用して、Manager マシンから WebSocket プロキシーを削除できます。

Manager マシンからの Websocket プロキシーの削除

1. Manager マシンで、**engine-cleanup** を実行して、必要な設定を削除します。

```
# engine-cleanup
```

2. すべてのコンポーネントを削除するかどうかを問われたら、**No** と入力して **Enter** を押します。

```
Do you want to remove all components? (Yes, No) [Yes]: No
```

3. エンジンを削除するかどうかを問われたら **No** と入力し、**Enter** を押します。

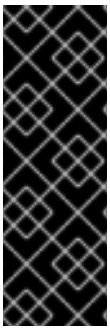
```
Do you want to remove the engine? (Yes, No) [Yes]: No
```

4. WebSocket プロキシーを削除するかどうかを問われたら、**Yes** と入力して **Enter** を押します。

```
Do you want to remove the WebSocket proxy? (Yes, No) [No]: Yes
```

   他のコンポーネントを削除するかどうかを問われたら、**No** を選択します。

別のマシンへの Websocket プロキシーのインストール



重要

Websocket プロキシーおよび noVNC は、テクノロジープレビュー機能としてのみ提供されています。テクノロジープレビューの機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat では実稼働環境での使用を推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、開発プロセスの中でお客様に機能性のテストとフィードバックをしていただくことを目的としています。詳しい情報は、テクノロジープレビュー機能のサポート範囲 を参照してください。

Websocket プロキシーにより、ユーザーは noVNC コンソールを介して仮想マシンに接続することができます。noVNC クライアントは Websocket を使用して VNC データを渡します。ただし、QEMU の VNC サーバーには Websocket サポートがないため、Websocket プロキシーはクライアントと VNC

サーバーの間に配置する必要があります。Websocket プロキシーは、ネットワークへのアクセスがあるすべてのマシン (Manager マシンを含む) で実行可能です。

セキュリティーおよびパフォーマンスの理由から、ユーザーには別のマシンで Websocket プロキシーを設定することを推奨します。

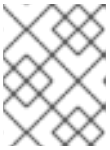
手順

1. Websocket プロキシーをインストールします。

```
# yum install ovirt-engine-websocket-proxy
```

2. **engine-setup** コマンドを実行して、WebSocket プロキシーを設定します。

```
# engine-setup
```



注記

rhvm パッケージもインストールされている場合は、このホストで Manager (**Engine**) を設定するかどうかを問われたら、**No** を選択します。

3. **Enter** を押して、**engine-setup** がマシン上に WebSocket プロキシーサーバーを設定できるようにします。

```
Configure WebSocket Proxy on this machine? (Yes, No) [Yes]:
```

4. **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。仮想化ホストを使用している場合には、自動的に検出されたホスト名が間違っている可能性がある点に注意してください。

```
Host fully qualified DNS name of this server [host.example.com]:
```

5. **Enter** を押して、**engine-setup** がファイアウォールを設定し、外部通信に必要なポートを開くことを許可します。**engine-setup** でファイアウォール設定を変更できない場合は、必要なポートを手動で開く必要があります。

```
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]:
```

6. Manager マシンの FQDN を入力し、**Enter** を押します。

```
Host fully qualified DNS name of the engine server []: manager.example.com
```

7. **Enter** を押して、**engine-setup** が Manager マシンでアクションを実行できるようにするか、**2** を押して手動でアクションを実行します。

```
Setup will need 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]:
```

a. **Enter** を押して、デフォルトの SSH ポート番号を受け入れるか、Manager マシンのポート番号を入力します。

```
ssh port on remote engine server [22]:
```

b. root パスワードを入力して Manager マシンにログインし、**Enter** を押します。

```
root password on remote engine server engine_host.example.com:
```

8. iptables ルールが現在の設定と異なる場合に、それらを確認するかどうかを選択します。

```
Generated iptables rules are different from current ones.
Do you want to review them? (Yes, No) [No]:
```

9. **Enter** を押して、設定を確定します。

```
--== CONFIGURATION PREVIEW ==--

Firewall manager                    : iptables
Update Firewall                     : True
Host FQDN                           : host.example.com
Configure WebSocket Proxy           : True
Engine Host FQDN                    : engine_host.example.com

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

Manager マシンが設定済みの Websocket プロキシーを使用するように設定するための説明が表示されます。

```
Manual actions are required on the engine host
in order to enroll certs for this host and configure the engine about it.

Please execute this command on the engine host:
   engine-config -s WebSocketProxy=host.example.com:6100
and than restart the engine service to make it effective
```

10. Manager マシンへログインして、表示された説明に沿って操作を行います。

```
# engine-config -s WebSocketProxy=host.example.com:6100
# systemctl restart ovirt-engine.service
```

付録F ブランド化

F.1. ブランド化

F.1.1. Manager の再ブランド化

ポップアップウィンドウで使用されるアイコンや表示されるテキスト、Welcome ページに表示されるリンクなど、Red Hat Virtualization Manager のさまざまな側面をカスタマイズできます。これにより、Manager のブランドを変更し、管理者およびユーザーに表示される最終的なルックアンドフィールを細かく制御することができます。

Manager のカスタマイズに必要なファイルは、Manager がインストールされているシステムの **/etc/ovirt-engine/branding/** ディレクトリーにあります。ファイルは、グラフィカルユーザーインターフェイスの様々な側面をスタイルするために使用されるカスケードスタイルシートファイルのセットと、Manager の様々なコンポーネントに組み込まれるメッセージとリンクを含むプロパティーファイルのセットで設定されています。

コンポーネントをカスタマイズするには、そのコンポーネントのファイルを編集して変更を保存します。次にそのコンポーネントを開いたり更新したりすると、変更が適用されます。

F.1.2. ログイン画面

ログイン画面は、管理ポータルと VM ポータルの両方が使用するログイン画面です。カスタマイズできるログイン画面の要素は次のとおりです。

- ボーダー
- 左側のヘッダーイメージ
- 右側のヘッダーイメージ
- ヘッダーテキスト

ログイン画面のクラスは **common.css** にあります。

F.1.3. 管理ポータルの画面

管理ポータルの画面は、管理ポータルにログインする際に表示されるメイン画面です。カスタマイズできる管理ポータル画面の要素は、以下のとおりです。

- ロゴ
- 左側のバックグラウンドイメージ
- センターのバックグラウンドイメージ
- 右側のバックグラウンドイメージ
- ロゴの右側にあるテキスト

管理ポータル画面のクラスは **web_admin.css** にあります。

F.1.4. VM ポータル画面

VM ポータル画面は、VM ポータルにログインする際に表示される画面です。カスタマイズが可能な VM ポータル画面の要素は、以下のとおりです。

- ロゴ
- センターのバックグラウンドイメージ
- 右側のバックグラウンドイメージ
- メイングリッド周辺の境界線
- **Logged in user** ラベルの上のテキスト

VM ポータル画面のクラスは **user_portal.css** にあります。

F.1.5. ポップアップウィンドウ

ポップアップウィンドウは、ホストまたは仮想マシンなどのエンティティーの作成、編集、または更新を可能にする Manager のすべてのウィンドウです。カスタマイズできるポップアップウィンドウの要素は次のとおりです。

- ボーダー
- 左側のヘッダーイメージ
- ヘッダーセンターイメージ (繰り返し)

ポップアップウィンドウのクラスは **common.css** にあります。

F.1.6. タブ

管理ポータルの多くのポップアップウィンドウにはタブが含まれています。カスタマイズ可能なこれらのタブの要素は次のとおりです。

- Active
- 非アクティブ

タブのクラスは **common.css** および **user_portal.css** にあります。

F.1.7. Welcome ページ

Welcome ページは、Manager のホームページにアクセスする際に最初に表示されるページです。全体的なルックアンドフィールをカスタマイズするだけでなく、テンプレートファイルを編集して、追加のドキュメントや内部 Web サイトのページへのリンクを追加するなどの変更を加えることもできます。カスタマイズできる Welcome Page の要素は次のとおりです。

- ページタイトル
- ヘッダー (左、中央、右)
- エラーメッセージ
- 転送するリンクおよびそのリンクに関連するメッセージ

Welcome ページのクラスは **welcome_style.css** にあります。

テンプレートファイル

Welcome Page のテンプレートファイルは、**HTML**、**HEAD**、または **BODY** タグが含まれない **welcome_page.template** の名前の通常の HTML ファイルです。このファイルはウェルカムページ自体に直接挿入され、ウェルカムページに表示されるコンテンツのコンテナーとして機能します。そのため、このファイルを編集して、新しいリンクを追加したり、コンテンツ自体を変更したりする必要があります。テンプレートファイルのもう 1 つの機能は、ウェルカムページの処理時に **messages.properties** ファイル内の対応するテキストに置き換えられる **{user_portal}** などのプレースホルダーテキストが含まれていることです。

F.1.8. Page Not Found ページ

Page Not Found ページは、Red Hat Virtualization Manager で見つからないページへのリンクを開くと表示されるページです。カスタマイズできる Page Not Found ページの要素は以下のとおりです。

- ページタイトル
- ヘッダー (左、中央、右)
- エラーメッセージ
- 転送するリンクおよびそのリンクに関連するメッセージ

Page Not Found ページのクラスは **welcome_style.css** にあります。

付録G システムアカウント

G.1. システムアカウント

G.1.1. Red Hat Virtualization Manager のユーザーアカウント

rhevm パッケージがインストールされると、Red Hat Virtualization をサポートするために多数のシステムユーザーアカウントが作成されます。各システムユーザーには、デフォルトのユーザー ID (UID) があります。作成されるシステムユーザーアカウントは、以下のとおりです。

- **vdsm** ユーザー (UID **36**)。NFS ストレージドメインをマウントおよびアクセスするサポートツールに必要です。
- **ovirt** ユーザー (UID **108**)。**ovirt-engine** Red Hat JBoss Enterprise Application Platform インスタンスの所有者。
- **ovirt-vmconsole** ユーザー (UID **498**)ゲストのシリアルコンソールに必要です。

G.1.2. Red Hat Virtualization Manager グループ

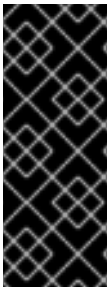
rhevm パッケージがインストールされると、Red Hat Virtualization をサポートするために多数のシステムユーザーグループが作成されます。各システムユーザーグループには、デフォルトのグループ ID (GID) があります。作成されるシステムユーザーグループは、以下のとおりです。

- **kvm** グループ (GID **36**).グループメンバーには以下が含まれます。
- **vdsm** ユーザー。
- **ovirt** グループ (GID **108**).グループメンバーには以下が含まれます。
- **ovirt** ユーザー。
- **ovirt-vmconsole** グループ (GID **498**).グループメンバーには以下が含まれます。
- **ovirt-vmconsole** ユーザー。

G.1.3. 仮想化ホストのユーザーアカウント

vdsm および **qemu-kvm-rhev** パッケージがインストールされると、仮想化ホスト上に多数のシステムユーザーアカウントが作成されます。各システムユーザーには、デフォルトのユーザー ID (UID) があります。作成されるシステムユーザーアカウントは、以下のとおりです。

- **vdsm** ユーザー (UID **36**)。
- **qemu** ユーザー (UID **107**).
- **sanlock** ユーザー (UID **179**).
- **ovirt-vmconsole** ユーザー (UID **498**)



重要

割り当てられるユーザー ID (UID) およびグループ ID (GID) は、システムによって異なる場合があります。**vdsm** ユーザーは **36** の UID に固定され、 **kvm** グループは **36** の GID に固定されます。

UID **36** または GID **36** がシステムの別のアカウントで既に使用されている場合は、**vdsm** および **qemu-kvm-rhev** パッケージのインストール時に競合が発生します。

G.1.4. 仮想化ホストグループ

vdsm および **qemu-kvm-rhev** パッケージがインストールされると、仮想化ホスト上に多数のシステムユーザーグループが作成されます。各システムユーザーグループには、デフォルトのグループ ID (GID) があります。作成されるシステムユーザーグループは、以下のとおりです。

- kvm グループ (GID **36**).グループメンバーには以下が含まれます。
- qemu ユーザー。
- sanlock ユーザー。
- qemu グループ (GID **107**).グループメンバーには以下が含まれます。
- vdsm ユーザー。
- sanlock ユーザー。
- ovirt-vmconsole グループ (GID **498**).グループメンバーには以下が含まれます。
- ovirt-vmconsole ユーザー。

重要

割り当てられるユーザー ID (UID) およびグループ ID (GID) は、システムによって異なる場合があります。**vdsm** ユーザーは **36** の UID に固定され、 **kvm** グループは **36** の GID に固定されます。

UID **36** または GID **36** がシステムの別のアカウントで既に使用されている場合は、**vdsm** および **qemu-kvm-rhev** パッケージのインストール時に競合が発生します。