Red Hat Virtualization 4.0 Virtual Machine Management Guide

Managing Virtual Machines in Red Hat Virtualization

Red Hat Virtualization Documentation Team

Red Hat Virtualization 4.0 Virtual Machine Management Guide

Managing Virtual Machines in Red Hat Virtualization

Red Hat Virtualization Documentation Team
Red Hat Customer Content Services
rhev-docs@redhat.com

Legal Notice

Abstract

This guide describes the installation, configuration, and administration of virtual machines in Red Hat Virtualization.

Table of Contents

Chapter 1. Introduction

A virtual machine is a software implementation of a computer. The Red Hat Virtualization environment enables you to create virtual desktops and virtual servers.

Virtual machines consolidate computing tasks and workloads. In traditional computing environments, workloads usually run on individually administered and upgraded servers. Virtual machines reduce the amount of hardware and administration required to run the same computing tasks and workloads.

1.1. Audience

Most virtual machine tasks in Red Hat Virtualization can be performed in both the User Portal and Administration Portal. However, the user interface differs between each portal, and some administrative tasks require access to the Administration Portal. Tasks that can only be performed in the Administration Portal will be described as such in this book. Which portal you use, and which tasks you can perform in each portal, is determined by your level of permissions. Virtual machine permissions are explained in Section 6.8, "Virtual Machines and Permissions".

The User Portal's user interface is described in the Introduction to the User Portal.

The Administration Portal's user interface is described in the Introduction to the Administration Portal.

The creation and management of virtual machines through the Red Hat Virtualization REST API is documented in the REST API Guide.

1.2. Supported Virtual Machine Operating Systems

The operating systems that can be virtualized as guest operating systems in Red Hat Virtualization are as follows:

Table 1.1. Operating systems that can be used as guest operating systems

Operating System	Architecture
Red Hat Enterprise Linux 3	32-bit, 64-bit
Red Hat Enterprise Linux 4	32-bit, 64-bit
Red Hat Enterprise Linux 5	32-bit, 64-bit
Red Hat Enterprise Linux 6	32-bit, 64-bit
Red Hat Enterprise Linux 7	64-bit
Red Hat Enterprise Linux Atomic Host 7	64-bit
SUSE Linux Enterprise Server 10 (select **Other Linux** for the guest type in the user interface)	32-bit, 64-bit
SUSE Linux Enterprise Server 11 (SPICE drivers (QXL) are not supplied by Red Hat. However, the distribution's vendor may provide SPICE drivers as part of their distribution.)	32-bit, 64-bit
Ubuntu 12.04 (Precise Pangolin LTS)	32-bit, 64-bit
Ubuntu 12.10 (Quantal Quetzal)	32-bit, 64-bit
Ubuntu 13.04 (Raring Ringtail)	32-bit, 64-bit
Ubuntu 13.10 (Saucy Salamander)	32-bit, 64-bit
Windows 7	32-bit, 64-bit

Operating System	Architecture
Windows 8	32-bit, 64-bit
Windows 8.1	32-bit, 64-bit
Windows 10	32-bit, 64-bit
Windows Server 2008	32-bit, 64-bit
Windows Server 2008 R2	64-bit
Windows Server 2012	64-bit
Windows Server 2012 R2	64-bit

Of the operating systems that can be virtualized as guest operating systems in Red Hat Virtualization, the operating systems that are supported by Global Support Services are as follows:

Table 1.2. Guest operating systems that are supported by Global Support Services

Operating System	Architecture	SPICE Support
Red Hat Enterprise Linux 3	32-bit, 64-bit	No
Red Hat Enterprise Linux 4	32-bit, 64-bit	No
Red Hat Enterprise Linux 5	32-bit, 64-bit	No
Red Hat Enterprise Linux 6	32-bit, 64-bit	Yes (on Red Hat Enterprise Linux 6.8 and above)
Red Hat Enterprise Linux 7	64-bit	Yes (on Red Hat Enterprise Linux 7.2 and above)
Red Hat Enterprise Linux Atomic Host 7	64-bit	Yes
SUSE Linux Enterprise Server 10 (select **Other Linux** for the guest type in the user interface)	32-bit, 64-bit	No
SUSE Linux Enterprise Server 11 (SPICE drivers (QXL) are not supplied by Red Hat. However, the distribution's vendor may provide SPICE drivers as part of their distribution.)	32-bit, 64-bit	No
Windows 7	32-bit, 64-bit	Yes
Windows 8	32-bit, 64-bit	No
Windows 8.1	32-bit, 64-bit	No
Windows 10	32-bit, 64-bit	No
Windows Server 2008	32-bit, 64-bit	No
Windows Server 2008 R2	64-bit	No
Windows Server 2012	64-bit	No
Windows Server 2012 R2	64-bit	No

Remote Desktop Protocol (RDP) is the default connection protocol for accessing Windows 8 and Windows 2012 guests from the user portal as Microsoft introduced changes to the Windows Display Driver Model that prevent SPICE from performing optimally.

> **Note**
>
> While Red Hat Enterprise Linux 3 and Red Hat Enterprise Linux 4 are supported, virtual machines running the 32-bit version of these operating systems cannot be shut down gracefully from the administration portal because there is no ACPI support in the 32-bit x86 kernel. To terminate virtual machines running the 32-bit version of Red Hat Enterprise Linux 3 or Red Hat Enterprise Linux 4, right-click the virtual machine and select the **Power Off** option.

> **Note**
>
> See http://www.redhat.com/resourcelibrary/articles/enterprise-linux-virtualization-support for information about up-to-date guest support.

1.3. Virtual Machine Performance Parameters

Red Hat Virtualization virtual machines can support the following parameters:

Table 1.3. Supported virtual machine parameters

Parameter	Number	Note
Virtualized CPUs	240	Per virtual machine running on a Red Hat Enterprise Linux 6 host.
Virtualized CPUs	240	Per virtual machine running on a Red Hat Enterprise Linux 7 host.
Virtualized RAM	4000 GB	For a 64 bit virtual machine.
Virtualized RAM	4 GB	Per 32 bit virtual machine. Note, the virtual machine may not register the entire 4 GB. The amount of RAM that the virtual machine recognizes is limited by its operating system.
Virtualized storage devices	8	Per virtual machine.
Virtualized network interface controllers	8	Per virtual machine.
Virtualized PCI devices	32	Per virtual machine.

1.4. Installing Supporting Components

1.4.1. Installing Console Components

A console is a graphical window that allows you to view the start up screen, shut down screen, and desktop of a virtual machine, and to interact with that virtual machine in a similar way to a physical machine. In Red Hat Virtualization, the default application for opening a console to a virtual machine is Remote Viewer, which must be installed on the client machine prior to use.

1.4.1.1. Installing Remote Viewer on Red Hat Enterprise Linux

The Remote Viewer application provides users with a graphical console for connecting to virtual machines. Once installed, it is called automatically when attempting to open a SPICE session with a virtual machine. Alternatively, it can also be used as a standalone application. Remote Viewer is included in the *virt-viewer* package provided by the base Red Hat Enterprise Linux Workstation and Red Hat Enterprise Linux Server repositories.

Procedure 1.1. Installing Remote Viewer on Linux

1. Install the *virt-viewer* package:

   ```
   # yum install virt-viewer
   ```

2. Restart your browser for the changes to take effect.

You can now connect to your virtual machines using either the SPICE protocol or the VNC protocol.

1.4.1.2. Installing Remote Viewer on Windows

The Remote Viewer application provides users with a graphical console for connecting to virtual machines. Once installed, it is called automatically when attempting to open a SPICE session with a virtual machine. Alternatively, it can also be used as a standalone application.

Procedure 1.2. Installing Remote Viewer on Windows

1. Open a web browser and download one of the following installers according to the architecture of your system.

   ≫ Virt Viewer for 32-bit Windows:

   ```
   https://your-manager-fqdn/ovirt-engine/services/files/spice/virt-
   viewer-x86.msi
   ```

   ≫ Virt Viewer for 64-bit Windows:

   ```
   https://your-manager-fqdn/ovirt-engine/services/files/spice/virt-
   viewer-x64.msi
   ```

2. Open the folder where the file was saved.

3. Double-click the file.

4. Click **Run** if prompted by a security warning.

5. Click **Yes** if prompted by User Account Control.

Remote Viewer is installed and can be accessed via **Remote Viewer** in the `VirtViewer` folder of `All Programs` in the start menu.

1.4.2. Installing usbdk on Windows

usbdk is a driver that is able to install and uninstall Windows USB drivers in guest virtual machines. Installing **usbdk** requires Administrator privileges. Note that the previously supported **USB Clerk** option has been deprecated and is no longer supported.

Procedure 1.3. Installing usbdk on Windows

1. Open a web browser and download one of the following installers according to the architecture of your system.

   ≫ **usbdk** for 32-bit Windows:

   ```
   https://[your manager's address]/ovirt-
   engine/services/files/spice/usbdk-x86.msi
   ```

   ≫ **usbdk** for 64-bit Windows:

   ```
   https://[your manager's address]/ovirt-
   engine/services/files/spice/usbdk-x64.msi
   ```

2. Open the folder where the file was saved.

3. Double-click the file.

4. Click **Run** if prompted by a security warning.

5. Click **Yes** if prompted by User Account Control.

The SPICE client sends requests to install or uninstall drivers for USB devices when users connect or disconnect USB devices to or from a guest, upon request.

Chapter 2. Installing Linux Virtual Machines

This chapter describes the steps required to install a Linux virtual machine:

1. Create a blank virtual machine on which to install an operating system.

2. Add a virtual disk for storage.

3. Add a network interface to connect the virtual machine to the network.

4. Install an operating system on the virtual machine. See your operating system's documentation for instructions.

   * Red Hat Enterprise Linux 6: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/index.html

   * Red Hat Enterprise Linux 7: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Installation_Guide/index.html

   * Red Hat Enterprise Linux Atomic Host 7: https://access.redhat.com/documentation/en/red-hat-enterprise-linux-atomic-host/7/single/installation-and-configuration-guide/

5. Register the virtual machine with the Content Delivery Network and subscribe to the relevant entitlements.

6. Install guest agents and drivers for additional virtual machine functionality.

When all of these steps are complete, the new virtual machine is functional and ready to perform tasks.

2.1. Creating a Linux Virtual Machine

Create a new virtual machine and configure the required settings.

Procedure 2.1. Creating Linux Virtual Machines

1. Click the **Virtual Machines** tab.

2. Click the **New VM** button to open the **New Virtual Machine** window.
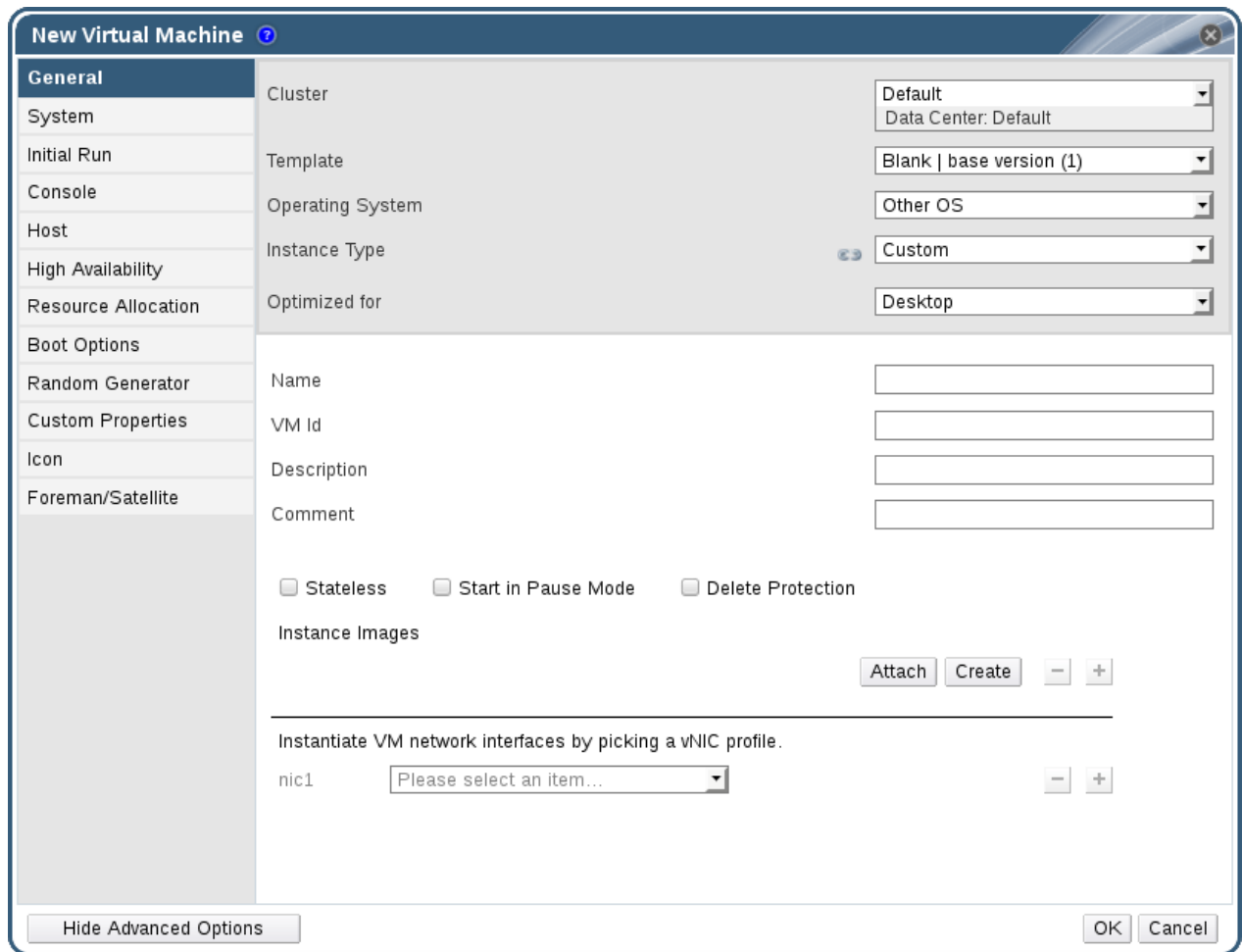
Figure 2.1. The New Virtual Machine Window

3. Select a Linux variant from the **Operating System** drop-down list.

4. Enter a **Name** for the virtual machine.

5. Add storage to the virtual machine. **Attach** or **Create** a virtual disk under **Instance Images**.

   A. Click **Attach** and select an existing virtual disk.

   B. Click **Create** and enter a **Size(GB)** and **Alias** for a new virtual disk. You can accept the default settings for all other fields, or change them if required. See Section A.3, "Explanation of Settings in the New Virtual Disk and Edit Virtual Disk Windows" for more details on the fields for all disk types.

6. Connect the virtual machine to the network. Add a network interface by selecting a vNIC profile from the **nic1** drop-down list at the bottom of the **General** tab.

7. Specify the virtual machine's **Memory Size** on the **System** tab.

8. Choose the **First Device** that the virtual machine will boot from on the **Boot Options** tab.

9. You can accept the default settings for all other fields, or change them if required. For more details on all fields in the **New Virtual Machine** window, see Section A.1, "Explanation of Settings in the New Virtual Machine and Edit Virtual Machine Windows".

10. Click **OK**.

The new virtual machine is created and displays in the list of virtual machines with a status of **Down**. Before you can use this virtual machine, you must install an operating system and register with the Content Delivery Network.

2.2. Starting the Virtual Machine

2.2.1. Starting a Virtual Machine

Procedure 2.2. Starting Virtual Machines

1. Click the `Virtual Machines` tab and select a virtual machine with a status of **Down**.

2. Click the run (▲) button.

   Alternatively, right-click the virtual machine and select **Run**.

The **Status** of the virtual machine changes to **Up**, and the operating system installation begins. Open a console to the virtual machine if one does not open automatically.

2.2.2. Opening a Console to a Virtual Machine

Use Remote Viewer to connect to a virtual machine.

Procedure 2.3. Connecting to Virtual Machines

1. Install Remote Viewer if it is not already installed. See Section 1.4.1, "Installing Console Components".

2. Click the `Virtual Machines` tab and select a virtual machine.

3. Click the console button or right-click the virtual machine and select `Console`.

4. A. If the connection protocol is set to SPICE, a console window will automatically open for the virtual machine.

   B. If the connection protocol is set to VNC, a `console.vv` file will be downloaded. Click on the file and a console window will automatically open for the virtual machine.

2.2.3. Opening a Serial Console to a Virtual Machine

Access a virtual machine's serial console from the command line, instead of opening a console from the Administration Portal or the User Portal. The serial console is emulated through VirtIO channels, using SSH and key pairs, and does not require direct access to the Manager; the Manager acts as a proxy for the connection, provides information about virtual machine placement, and stores the authentication keys. You can add public keys for each user from either the Administration Portal or the User Portal. You can access serial consoles for only those virtual machines for which you have appropriate permissions.

> **Important**
>
> To access the serial console of a virtual machine, the user must have the UserVmManager, SuperUser, or UserInstanceManager permission on that virtual machine. These permissions must be explicitly defined per user; it is not enough to assign these permissions for **Everyone**.

The serial console is accessed via TCP port 2222 on the Manager. This port is opened during **engine-setup** on new installations. The serial console relies on the *ovirt-vmconsole* package and the *ovirt-vmconsole-proxy* on the Manager, and the *ovirt-vmconsole* package and the *ovirt-vmconsole-host* package on virtualization hosts. These packages are installed by default on new installations. To install the packages on existing installations, reinstall the host. See Reinstalling Hosts in the *Administration Guide*.

Procedure 2.4. Connecting to a Virtual Machine Serial Console

1. On the client machine from which you will access the virtual machine serial console, generate an SSH key pair. The Manager supports standard SSH key types. For example, generate an RSA key:

   ```
   # ssh-keygen -t rsa -b 2048 -C "admin@internal" -f
   .ssh/serialconsolekey
   ```

   This command generates a public key and a private key.

2. In the Administration Portal or the User Portal, click the name of the signed-in user on the header bar, and then click **Options** to open the **Edit Options** window.

3. In the **User's Public Key** text field, paste the public key of the client machine that will be used to access the serial console.

4. Click the **Virtual Machines** tab and select a virtual machine.

5. Click **Edit**.

6. In the **Console** tab of the **Edit Virtual Machine** window, select the **Enable VirtIO serial console** check box.

7. On the client machine, connect to the virtual machine's serial console:

   a. If a single virtual machine is available, this command connects the user to that virtual machine:

   ```
   # ssh -t -p 2222 ovirt-vmconsole@MANAGER_IP
   Red Hat Enterprise Linux Server release 6.7 (Santiago)
   Kernel 2.6.32-573.3.1.el6.x86_64 on an x86_64
   USER login:
   ```

   If more than one virtual machine is available, this command lists the available virtual machines:

   ```
   # ssh -t -p 2222 ovirt-vmconsole@MANAGER_IP
   1. vm1 [vmid1]
   2. vm2 [vmid2]
   3. vm3 [vmid3]
   ```

```
> 2
Red Hat Enterprise Linux Server release 6.7 (Santiago)
Kernel 2.6.32-573.3.1.el6.x86_64 on an x86_64
USER login:
```

Enter the number of the machine to which you want to connect, and press **Enter**.

b. Alternatively, connect directly to a virtual machine using its unique identifier or its name:

```
# ssh -t -p 2222 ovirt-vmconsole@MANAGER_IP --vm-id vmid1
```

```
# ssh -t -p 2222 ovirt-vmconsole@MANAGER_IP --vm-name vm1
```

> **Important**
>
> If the serial console session is disconnected abnormally, a TCP timeout occurs. You will be unable to reconnect to the virtual machine's serial console until the timeout period expires.

2.3. Subscribing to the Required Entitlements

To install packages signed by Red Hat you must register the target system to the Content Delivery Network. Then, use an entitlement from your subscription pool and enable the required repositories.

Procedure 2.5. Subscribing to the Required Entitlements Using Subscription Manager

1. Register your system with the Content Delivery Network, entering your Customer Portal user name and password when prompted:

   ```
   # subscription-manager register
   ```

2. Locate the relevant subscription pools and note down the pool identifiers.

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

3. Use the pool identifiers located in the previous step to attach the required entitlements.

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

4. Disable all existing repositories:

   ```
   # subscription-manager repos --disable=*
   ```

5. When a system is subscribed to a subscription pool with multiple repositories, only the main repository is enabled by default. Others are available, but disabled. Enable any additional repositories:

```
# subscription-manager repos --enable=repository
```

6. Ensure that all packages currently installed are up to date:

```
# yum update
```

2.4. Installing Guest Agents and Drivers

2.4.1. Red Hat Virtualization Guest Agents and Drivers

The Red Hat Virtualization guest agents and drivers provide additional information and functionality for Red Hat Enterprise Linux and Windows virtual machines. Key features include the ability to monitor resource usage and gracefully shut down or reboot virtual machines from the User Portal and Administration Portal. Install the Red Hat Virtualization guest agents and drivers on each virtual machine on which this functionality is to be available.

Table 2.1. Red Hat Virtualization Guest Drivers

Driver	Description	Works on
`virtio-net`	Paravirtualized network driver provides enhanced performance over emulated devices like rtl.	Server and Desktop.
`virtio-block`	Paravirtualized HDD driver offers increased I/O performance over emulated devices like IDE by optimizing the coordination and communication between the guest and the hypervisor. The driver complements the software implementation of the virtio-device used by the host to play the role of a hardware device.	Server and Desktop.
`virtio-scsi`	Paravirtualized iSCSI HDD driver offers similar functionality to the virtio-block device, with some additional enhancements. In particular, this driver supports adding hundreds of devices, and names devices using the standard SCSI device naming scheme.	Server and Desktop.

Driver	Description	Works on
virtio-serial	Virtio-serial provides support for multiple serial ports. The improved performance is used for fast communication between the guest and the host that avoids network complications. This fast communication is required for the guest agents and for other features such as clipboard copy-paste between the guest and the host and logging.	Server and Desktop.
virtio-balloon	Virtio-balloon is used to control the amount of memory a guest actually accesses. It offers improved memory over-commitment. The balloon drivers are installed for future compatibility but not used by default in Red Hat Virtualization.	Server and Desktop.
qxl	A paravirtualized display driver reduces CPU usage on the host and provides better performance through reduced network bandwidth on most workloads.	Server and Desktop.

Table 2.2. Red Hat Virtualization Guest Agents and Tools

Guest agent/tool	Description	Works on
rhevm-guest-agent-common	Allows the Red Hat Virtualization Manager to receive guest internal events and information such as IP address and installed applications. Also allows the Manager to execute specific commands, such as shut down or reboot, on a guest. On Red Hat Enterprise Linux 6 and higher guests, the rhevm-guest-agent-common installs **tuned** on your virtual machine and configures it to use an optimized, virtualized-guest profile.	Server and Desktop.

Guest agent/tool	Description	Works on
spice-agent	The SPICE agent supports multiple monitors and is responsible for client-mouse-mode support to provide a better user experience and improved responsiveness than the QEMU emulation. Cursor capture is not needed in client-mouse-mode. The SPICE agent reduces bandwidth usage when used over a wide area network by reducing the display level, including color depth, disabling wallpaper, font smoothing, and animation. The SPICE agent enables clipboard support allowing cut and paste operations for both text and images between client and guest, and automatic guest display setting according to client-side settings. On Windows guests, the SPICE agent consists of vdservice and vdagent.	Server and Desktop.
rhev-sso	An agent that enables users to automatically log in to their virtual machines based on the credentials used to access the Red Hat Virtualization Manager.	Desktop.
rhev-usb	A component that contains drivers and services for Legacy USB support (version 3.0 and earlier) on guests. It is needed for accessing a USB device that is plugged into the client machine. **RHEV-USB Client** is needed on the client side. **Important** The Legacy USB option has been deprecated and will be removed in Red Hat Virtualization 4.1.	Desktop.

2.4.2. Installing the Guest Agents and Drivers on Red Hat Enterprise Linux

The Red Hat Virtualization guest agents and drivers are installed on Red Hat Enterprise Linux virtual machines using the **rhevm-guest-agent** package provided by the Red Hat Virtualization Agent repository.

Procedure 2.6. Installing the Guest Agents and Drivers on Red Hat Enterprise Linux

1. Log in to the Red Hat Enterprise Linux virtual machine.

2. Enable the Red Hat Virtualization Agent repository:

   A. For Red Hat Enterprise Linux 6

   ```
   # subscription-manager repos --enable=rhel-6-server-rhv-4-agent-rpms
   ```

   B. For Red Hat Enterprise Linux 7

   ```
   # subscription-manager repos --enable=rhel-7-server-rh-common-rpms
   ```

3. Install the *rhevm-guest-agent-common* package and dependencies:

   ```
   # yum install rhevm-guest-agent-common
   ```

4. Start and enable the service:

   A. For Red Hat Enterprise Linux 6

   ```
   # service ovirt-guest-agent start
   # chkconfig ovirt-guest-agent on
   ```

   B. For Red Hat Enterprise Linux 7

   ```
   # systemctl start ovirt-guest-agent.service
   # systemctl enable ovirt-guest-agent.service
   ```

5. Start and enable the **qemu-ga** service:

   A. For Red Hat Enterprise Linux 6

   ```
   # service qemu-ga start
   # chkconfig qemu-ga on
   ```

   B. For Red Hat Enterprise Linux 7

   ```
   # systemctl start qemu-guest-agent.service
   # systemctl enable qemu-guest-agent.service
   ```

The guest agent now passes usage information to the Red Hat Virtualization Manager. The Red Hat Virtualization agent runs as a service called **ovirt-guest-agent** that you can configure via the **ovirt-guest-agent.conf** configuration file in the **/etc/** directory.

Chapter 3. Installing Windows Virtual Machines

This chapter describes the steps required to install a Windows virtual machine:

1. Create a blank virtual machine on which to install an operating system.

2. Add a virtual disk for storage.

3. Add a network interface to connect the virtual machine to the network.

4. Attach the **virtio-win.vfd** diskette to the virtual machine so that VirtIO-optimized device drivers can be installed during the operating system installation.

5. Install an operating system on the virtual machine. See your operating system's documentation for instructions.

6. Install guest agents and drivers for additional virtual machine functionality.

When all of these steps are complete, the new virtual machine is functional and ready to perform tasks.

3.1. Creating a Windows Virtual Machine

Create a new virtual machine and configure the required settings.

Procedure 3.1. Creating Windows Virtual Machines

1. Click the **Virtual Machines** tab.

2. Click the **New VM** button to open the **New Virtual Machine** window.

Figure 3.1. The New Virtual Machine Window

3. Select a Windows variant from the **Operating System** drop-down list.

4. Enter a **Name** for the virtual machine.

5. Add storage to the virtual machine. **Attach** or **Create** a virtual disk under **Instance Images**.

   A. Click **Attach** and select an existing virtual disk.

   B. Click **Create** and enter a **Size(GB)** and **Alias** for a new virtual disk. You can accept the default settings for all other fields, or change them if required. See Section A.3, "Explanation of Settings in the New Virtual Disk and Edit Virtual Disk Windows" for more details on the fields for all disk types.

6. Connect the virtual machine to the network. Add a network interface by selecting a vNIC profile from the **nic1** drop-down list at the bottom of the **General** tab.

7. Specify the virtual machine's **Memory Size** on the **System** tab.

8. Choose the **First Device** that the virtual machine will boot from on the **Boot Options** tab.

9. You can accept the default settings for all other fields, or change them if required. For more details on all fields in the **New Virtual Machine** window, see Section A.1, "Explanation of Settings in the New Virtual Machine and Edit Virtual Machine Windows".

10. Click **OK**.

The new virtual machine is created and displays in the list of virtual machines with a status of **Down**. Before you can use this virtual machine, you must install an operating system and VirtIO-optimized disk and network drivers.

3.2. Starting the Virtual Machine Using the Run Once Option

3.2.1. Installing Windows on VirtIO-Optimized Hardware

Install VirtIO-optimized disk and network device drivers during your Windows installation by attaching the **virtio-win.vfd** diskette to your virtual machine. These drivers provide a performance improvement over emulated device drivers.

Use the **Run Once** option to attach the diskette in a one-off boot different from the **Boot Options** defined in the **New Virtual Machine** window. This procedure presumes that you added a **Red Hat VirtIO** network interface and a disk that uses the **VirtIO** interface to your virtual machine.

> **Note**
>
> The **virtio-win.vfd** diskette is placed automatically on ISO storage domains that are hosted on the Manager server. An administrator must manually upload it to other ISO storage domains using the **engine-iso-uploader** tool.

Procedure 3.2. Installing VirtIO Drivers during Windows Installation

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Run Once**.

3. Expand the **Boot Options** menu.

4. Select the **Attach Floppy** check box, and select **virtio-win.vfd** from the drop-down list.

5. Select the **Attach CD** check box, and select the required Windows ISO from the drop-down list.

6. Move **CD-ROM** to the top of the **Boot Sequence** field.

7. Configure the rest of your **Run Once** options as required. See Section A.5, "Explanation of Settings in the Run Once Window" for more details.

8. Click **OK**.

The **Status** of the virtual machine changes to **Up**, and the operating system installation begins. Open a console to the virtual machine if one does not open automatically.

Windows installations include an option to load additional drivers early in the installation process. Use this option to load drivers from the **virtio-win.vfd** diskette that was attached to your virtual machine as **A:**. For each supported virtual machine architecture and Windows version, there is a folder on the disk containing optimized hardware device drivers.

3.2.2. Opening a Console to a Virtual Machine

Use Remote Viewer to connect to a virtual machine.

Procedure 3.3. Connecting to Virtual Machines

1. Install Remote Viewer if it is not already installed. See Section 1.4.1, "Installing Console Components".

2. Click the **Virtual Machines** tab and select a virtual machine.

3. Click the console button or right-click the virtual machine and select **Console**.

4. A. If the connection protocol is set to SPICE, a console window will automatically open for the virtual machine.

   B. If the connection protocol is set to VNC, a **console.vv** file will be downloaded. Click on the file and a console window will automatically open for the virtual machine.

3.3. Installing Guest Agents and Drivers

3.3.1. Red Hat Virtualization Guest Agents and Drivers

The Red Hat Virtualization guest agents and drivers provide additional information and functionality for Red Hat Enterprise Linux and Windows virtual machines. Key features include the ability to monitor resource usage and gracefully shut down or reboot virtual machines from the User Portal and Administration Portal. Install the Red Hat Virtualization guest agents and drivers on each virtual machine on which this functionality is to be available.

Table 3.1. Red Hat Virtualization Guest Drivers

Driver	Description	Works on
virtio-net	Paravirtualized network driver provides enhanced performance over emulated devices like rtl.	Server and Desktop.
virtio-block	Paravirtualized HDD driver offers increased I/O performance over emulated devices like IDE by optimizing the coordination and communication between the guest and the hypervisor. The driver complements the software implementation of the virtio-device used by the host to play the role of a hardware device.	Server and Desktop.

Driver	Description	Works on
virtio-scsi	Paravirtualized iSCSI HDD driver offers similar functionality to the virtio-block device, with some additional enhancements. In particular, this driver supports adding hundreds of devices, and names devices using the standard SCSI device naming scheme.	Server and Desktop.
virtio-serial	Virtio-serial provides support for multiple serial ports. The improved performance is used for fast communication between the guest and the host that avoids network complications. This fast communication is required for the guest agents and for other features such as clipboard copy-paste between the guest and the host and logging.	Server and Desktop.
virtio-balloon	Virtio-balloon is used to control the amount of memory a guest actually accesses. It offers improved memory over-commitment. The balloon drivers are installed for future compatibility but not used by default in Red Hat Virtualization.	Server and Desktop.
qxl	A paravirtualized display driver reduces CPU usage on the host and provides better performance through reduced network bandwidth on most workloads.	Server and Desktop.

Table 3.2. Red Hat Virtualization Guest Agents and Tools

Guest agent/tool	Description	Works on

Guest agent/tool	Description	Works on
`rhevm-guest-agent-common`	Allows the Red Hat Virtualization Manager to receive guest internal events and information such as IP address and installed applications. Also allows the Manager to execute specific commands, such as shut down or reboot, on a guest. On Red Hat Enterprise Linux 6 and higher guests, the rhevm-guest-agent-common installs **tuned** on your virtual machine and configures it to use an optimized, virtualized-guest profile.	Server and Desktop.
`spice-agent`	The SPICE agent supports multiple monitors and is responsible for client-mouse-mode support to provide a better user experience and improved responsiveness than the QEMU emulation. Cursor capture is not needed in client-mouse-mode. The SPICE agent reduces bandwidth usage when used over a wide area network by reducing the display level, including color depth, disabling wallpaper, font smoothing, and animation. The SPICE agent enables clipboard support allowing cut and paste operations for both text and images between client and guest, and automatic guest display setting according to client-side settings. On Windows guests, the SPICE agent consists of vdservice and vdagent.	Server and Desktop.
`rhev-sso`	An agent that enables users to automatically log in to their virtual machines based on the credentials used to access the Red Hat Virtualization Manager.	Desktop.

Guest agent/tool	Description	Works on
rhev-usb	A component that contains drivers and services for Legacy USB support (version 3.0 and earlier) on guests. It is needed for accessing a USB device that is plugged into the client machine. **RHEV-USB Client** is needed on the client side. **Important** The Legacy USB option has been deprecated and will be removed in Red Hat Virtualization 4.1.	Desktop.

3.3.2. Installing the Guest Agents and Drivers on Windows

The Red Hat Virtualization guest agents and drivers are installed on Windows virtual machines using the **rhev-tools-setup.iso** ISO file, which is provided by the *rhev-guest-tools-iso* package installed as a dependency to the Red Hat Virtualization Manager. This ISO file is located in **/usr/share/rhev-guest-tools-iso/rhev-tools-setup.iso** on the system on which the Red Hat Virtualization Manager is installed.

Note

The **rhev-tools-setup.iso** ISO file is automatically copied to the default ISO storage domain, if any, when you run **engine-setup**, or must be manually uploaded to an ISO storage domain.

Note

Updated versions of the **rhev-tools-setup.iso** ISO file must be manually attached to running Windows virtual machines to install updated versions of the tools and drivers. If the APT service is enabled on virtual machines, the updated ISO files will be automatically attached.

> **Note**
>
> If you install the guest agents and drivers from the command line or as part of a deployment tool such as Windows Deployment Services, you can append the options **ISSILENTMODE** and **ISNOREBOOT** to **RHEV-toolsSetup.exe** to silently install the guest agents and drivers and prevent the machine on which they have been installed from rebooting immediately after installation. The machine can then be rebooted later once the deployment process is complete.
>
> ```
> D:\RHEV-toolsSetup.exe ISSILENTMODE ISNOREBOOT
> ```

Procedure 3.4. Installing the Guest Agents and Drivers on Windows

1. Log in to the virtual machine.

2. Select the CD Drive containing the **rhev-tools-setup.iso** file.

3. Double-click **RHEV-toolsSetup**.

4. Click **Next** at the welcome screen.

5. Follow the prompts on the **RHEV-Tools InstallShield Wizard** window. Ensure all check boxes in the list of components are selected.
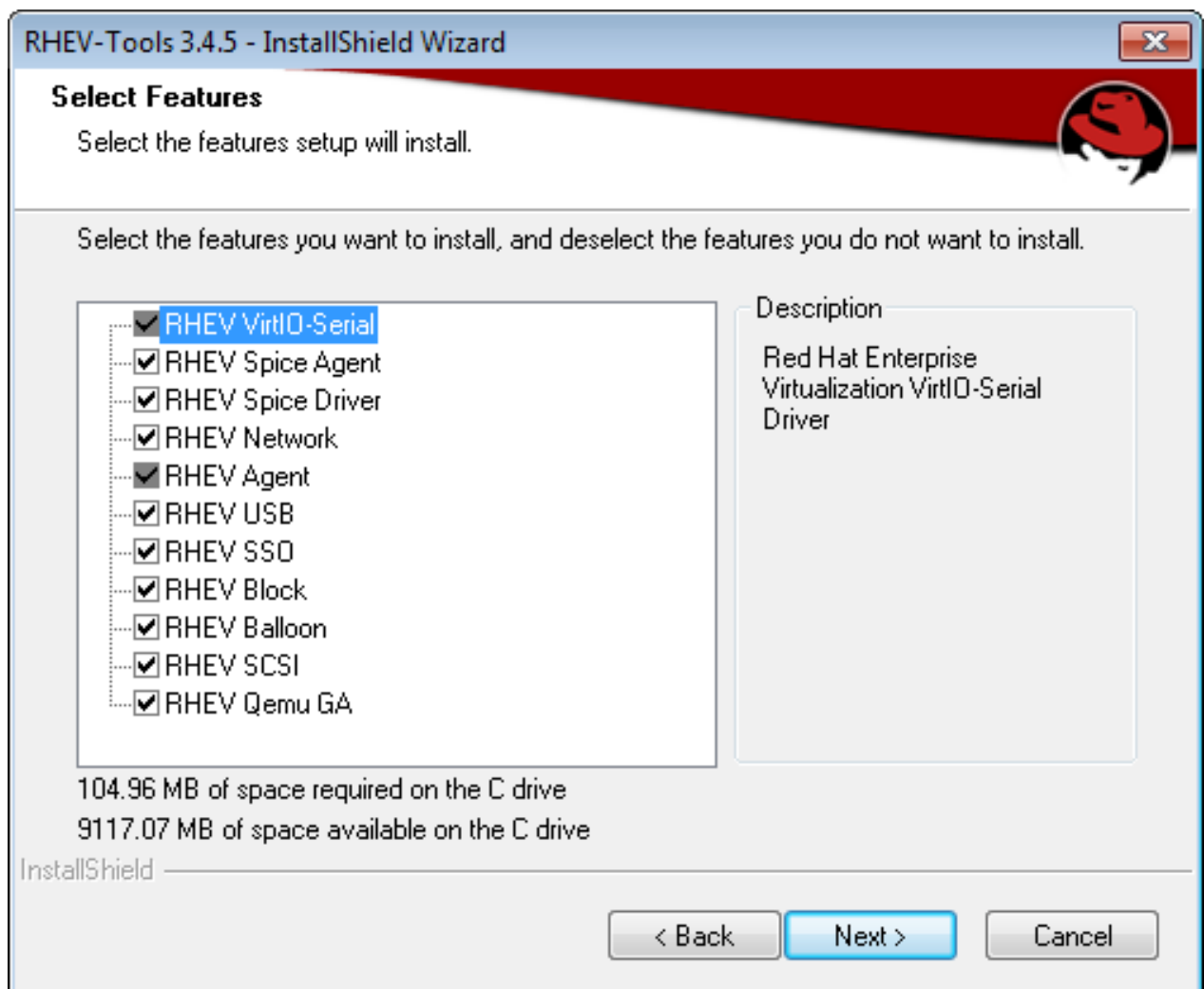
Figure 3.2. Selecting All Components of Red Hat Virtualization Tools for Installation

6. Once installation is complete, select **Yes, I want to restart my computer now** and click **Finish** to apply the changes.

The guest agents and drivers now pass usage information to the Red Hat Virtualization Manager and allow you to access USB devices, single sign-on into virtual machines and other functionality. The Red Hat Virtualization guest agent runs as a service called **RHEV Agent** that you can configure using the **rhev-agent** configuration file located in **C:\Program Files\Redhat\RHEV\Drivers\Agent**.

3.3.3. Automating Guest Additions on Windows Guests with Red Hat Virtualization Application Provisioning Tool(APT)

Red Hat Virtualization Application Provisioning Tool (APT) is a Windows service that can be installed on Windows virtual machines and templates. When the APT service is installed and running on a virtual machine, attached ISO files are automatically scanned. When the service recognizes a valid Red Hat Virtualization guest tools ISO, and no other guest tools are installed, the APT service installs the guest tools. If guest tools are already installed, and the ISO image contains newer versions of the tools, the service performs an automatic upgrade. This procedure assumes you have attached the **rhev-tools-setup.iso** ISO file to the virtual machine.

Procedure 3.5. Installing the APT Service on Windows

1. Log in to the virtual machine.

2. Select the CD Drive containing the **rhev-tools-setup.iso** file.

3. Double-click **RHEV-Application Provisioning Tool**.

4. Click **Yes** in the **User Account Control** window.

5. Once installation is complete, ensure the **Start RHEV-apt Service** check box is selected in the **RHEV-Application Provisioning Tool InstallShield Wizard** window, and click **Finish** to apply the changes.

Once the APT service has successfully installed or upgraded the guest tools on a virtual machine, the virtual machine is automatically rebooted; this happens without confirmation from the user logged in to the machine. The APT Service will also perform these operations when a virtual machine created from a template that has the APT Service already installed is booted for the first time.

> **Note**
>
> The **RHEV-apt** service can be stopped immediately after install by clearing the **Start RHEV-apt Service** check box. You can stop, start, or restart the service at any time using the **Services** window.

Chapter 4. Additional Configuration

4.1. Configuring Single Sign-On for Virtual Machines

Configuring single sign-on, also known as password delegation, allows you to automatically log in to a virtual machine using the credentials you use to log in to the User Portal. Single sign-on can be used on both Red Hat Enterprise Linux and Windows virtual machines.

> **Important**
>
> If single sign-on to the User Portal is enabled, single sign-on to virtual machines will not be possible. With single sign-on to the User Portal enabled, the User Portal does not need to accept a password, thus the password cannot be delegated to sign in to virtual machines.

4.1.1. Configuring Single Sign-On for Red Hat Enterprise Linux Virtual Machines Using IPA (IdM)

To configure single sign-on for Red Hat Enterprise Linux virtual machines using GNOME and KDE graphical desktop environments and IPA (IdM) servers, you must install the *rhevm-guest-agent* package on the virtual machine and install the packages associated with your window manager.

> **Important**
>
> The following procedure assumes that you have a working IPA configuration and that the IPA domain is already joined to the Manager. You must also ensure that the clocks on the Manager, the virtual machine and the system on which IPA (IdM) is hosted are synchronized using NTP.

Procedure 4.1. Configuring Single Sign-On for Red Hat Enterprise Linux Virtual Machines

1. Log in to the Red Hat Enterprise Linux virtual machine.

2. Enable the required channel:

   ≫ For Red Hat Enterprise Linux 6

   ```
   # subscription-manager repos --enable=rhel-6-server-rhv-4-agent-rpms
   ```

   ≫ For Red Hat Enterprise Linux 7

   ```
   # subscription-manager repos --enable=rhel-7-server-rh-common-rpms
   ```

3. Download and install the guest agent packages:

   ```
   # yum install rhevm-guest-agent-common
   ```

4. Install the single sign-on packages:

```
# yum install rhevm-guest-agent-pam-module
# yum install rhevm-guest-agent-gdm-plugin
```

5. Install the IPA packages:

```
# yum install ipa-client
```

6. Run the following command and follow the prompts to configure **ipa-client** and join the virtual machine to the domain:

```
# ipa-client-install --permit --mkhomedir
```

> **Note**
>
> In environments that use DNS obfuscation, this command should be:
>
> ```
> # ipa-client-install --domain=FQDN --server==FQDN
> ```

7. For Red Hat Enterprise Linux 7.2, run:

```
# authconfig --enablenis --update
```

> **Note**
>
> Red Hat Enterprise Linux 7.2 has a new version of the System Security Services Daemon (SSSD) which introduces configuration that is incompatible with the Red Hat Virtualization Manager guest agent single sign-on implementation. The command will ensure that single sign-on works.

8. Fetch the details of an IPA user:

```
# getent passwd IPA_user_name
```

This will return something like this:

```
some-ipa-user:*:936600010:936600001::/home/some-ipa-user:/bin/sh
```

You will need this information in the next step to create a home directory for *some-ipa-user*.

9. Set up a home directory for the IPA user:

a. Create the new user's home directory:

```
# mkdir /home/some-ipa-user
```

   b. Give the new user ownership of the new user's home directory:

```
# chown 935500010:936600001 /home/some-ipa-user
```

Log in to the User Portal using the user name and password of a user configured to use single sign-on and connect to the console of the virtual machine. You will be logged in automatically.

4.1.2. Configuring Single Sign-On for Red Hat Enterprise Linux Virtual Machines Using Active Directory

To configure single sign-on for Red Hat Enterprise Linux virtual machines using GNOME and KDE graphical desktop environments and Active Directory, you must install the *rhevm-guest-agent* package on the virtual machine, install the packages associated with your window manager and join the virtual machine to the domain.

> **Important**
>
> The following procedure assumes that you have a working Active Directory configuration and that the Active Directory domain is already joined to the Manager. You must also ensure that the clocks on the Manager, the virtual machine and the system on which Active Directory is hosted are synchronized using NTP.

Procedure 4.2. Configuring Single Sign-On for Red Hat Enterprise Linux Virtual Machines

1. Log in to the Red Hat Enterprise Linux virtual machine.

2. Enable the Red Hat Virtualization Agent channel:

    A. For Red Hat Enterprise Linux 6

    ```
    # subscription-manager repos --enable=rhel-6-server-rhv-4-agent-rpms
    ```

    B. For Red Hat Enterprise Linux 7

    ```
    # subscription-manager repos --enable=rhel-7-server-rh-common-rpms
    ```

3. Download and install the guest agent packages:

    ```
    # yum install rhevm-guest-agent-common
    ```

4. Install the single sign-on packages:

    ```
    # yum install rhev-agent-gdm-plugin-rhevcred
    ```

5. Install the Samba client packages:

    ```
    # yum install samba-client samba-winbind samba-winbind-clients
    ```

6. On the virtual machine, modify the **/etc/samba/smb.conf** file to contain the following, replacing **DOMAIN** with the short domain name and**REALM.LOCAL** with the Active Directory realm:

```
[global]
   workgroup = DOMAIN
   realm = REALM.LOCAL
   log level = 2
   syslog = 0
   server string = Linux File Server
   security = ads
   log file = /var/log/samba/%m
   max log size = 50
   printcap name = cups
   printing = cups
   winbind enum users = Yes
   winbind enum groups = Yes
   winbind use default domain = true
   winbind separator = +
   idmap uid = 1000000-2000000
   idmap gid = 1000000-2000000
   template shell = /bin/bash
```

7. Join the virtual machine to the domain:

```
net ads join -U user_name
```

8. Start the **winbind** service and ensure it starts on boot:

   A. For Red Hat Enterprise Linux 6

   ```
   # service winbind start
   # chkconfig winbind on
   ```

   B. For Red Hat Enterprise Linux 7

   ```
   # systemctl start winbind.service
   # systemctl enable winbind.service
   ```

9. Verify that the system can communicate with Active Directory:

   a. Verify that a trust relationship has been created:

   ```
   # wbinfo -t
   ```

   b. Verify that you can list users:

   ```
   # wbinfo -u
   ```

   c. Verify that you can list groups:

   ```
   # wbinfo -g
   ```

10. Configure the NSS and PAM stack:

a. Open the **Authentication Configuration** window:

```
# authconfig-tui
```

b. Select the **Use Winbind** check box, select **Next** and press **Enter**.

c. Select the **OK** button and press **Enter**.

Log in to the User Portal using the user name and password of a user configured to use single sign-on and connect to the console of the virtual machine. You will be logged in automatically.

4.1.3. Configuring Single Sign-On for Windows Virtual Machines

To configure single sign-on for Windows virtual machines, the Windows guest agent must be installed on the guest virtual machine. The **RHEV Guest Tools** ISO file provides this agent. If the **RHEV-toolsSetup.iso** image is not available in your ISO domain, contact your system administrator.

Procedure 4.3. Configuring Single Sign-On for Windows Virtual Machines

1. Select the Windows virtual machine. Ensure the machine is powered up.

2. Click **Change CD**.

3. Select **RHEV-toolsSetup.iso** from the list of images.

4. Click **OK**.

5. Click the **Console** icon and log in to the virtual machine.

6. On the virtual machine, locate the CD drive to access the contents of the guest tools ISO file and launch **RHEV-ToolsSetup.exe**. After the tools have been installed, you will be prompted to restart the machine to apply the changes.

Log in to the User Portal using the user name and password of a user configured to use single sign-on and connect to the console of the virtual machine. You will be logged in automatically.

4.1.4. Disabling Single Sign-on for Virtual Machines

The following procedure explains how to disable single sign-on for a virtual machine.

Procedure 4.4. Disabling Single Sign-On for Virtual Machines

1. Select a virtual machine and click **Edit**.

2. Click the **Console** tab.

3. Select the **Disable Single Sign On** check box.

4. Click **OK**.

4.2. Configuring USB Devices

A virtual machine connected with the SPICE protocol can be configured to connect directly to USB devices.

The USB device will only be redirected if the virtual machine is active and in focus. USB redirection can be manually enabled each time a device is plugged in or set to automatically redirect to active virtual machines in the SPICE client menu.

> **Important**
>
> Note the distinction between the client machine and guest machine. The client is the hardware from which you access a guest. The guest is the virtual desktop or virtual server which is accessed through the User Portal or Administration Portal.

4.2.1. Using USB Devices on Virtual Machines

USB redirection Native mode allows KVM/SPICE USB redirection for Linux and Windows virtual machines. Virtual (guest) machines require no guest-installed agents or drivers for native USB. On Red Hat Enterprise Linux clients, all packages required for USB redirection are provided by the *virt-viewer* package. On Windows clients, you must also install the *usbdk* package. Native USB mode is supported on the following clients and guests:

- Client

  - Red Hat Enterprise Linux 7.1 and higher

  - Red Hat Enterprise Linux 6.0 and higher

  - Windows 10

  - Windows 8

  - Windows 7

  - Windows 2008

  - Windows 2008 Server R2

- Guest

  - Red Hat Enterprise Linux 7.1 and higher

  - Red Hat Enterprise Linux 6.0 and higher

  - Windows 7

  - Windows XP

  - Windows 2008

> **Note**
>
> If you have a 64-bit architecture PC, you must use the 64-bit version of Internet Explorer to install the 64-bit version of the USB driver. The USB redirection will not work if you install the 32-bit version on a 64-bit architecture. As long as you initially install the correct USB type, you then can access USB redirection from both 32 and 64-bit browsers.

4.2.2. Using USB Devices on a Windows Client

The **usbdk** driver must be installed on the Windows client for the USB device to be redirected to the guest. Ensure the version of **usbdk** matches the architecture of the client machine. For example, the 64-bit version of **usbdk** must be installed on 64-bit Windows machines.

Procedure 4.5. Using USB Devices on a Windows Client

1. When the **usbdk** driver is installed, select a virtual machine that has been configured to use the SPICE protocol.

2. Ensure USB support is set to `Native`:

   a. Click **Edit**.

   b. Click the `Console` tab.

   c. Select `Native` from the `USB Support` drop-down list.

   d. Click `OK`.

3. Click the `Console Options` button and select the `Enable USB Auto-Share` check box.

4. Start the virtual machine and click the `Console` button to connect to that virtual machine. When you plug your USB device into the client machine, it will automatically be redirected to appear on your guest machine.

4.2.3. Using USB Devices on a Red Hat Enterprise Linux Client

The *usbredir* package enables USB redirection from Red Hat Enterprise Linux clients to virtual machines. *usbredir* is a dependency of the *virt-viewer* package, and is automatically installed together with that package.

Procedure 4.6. Using USB devices on a Red Hat Enterprise Linux client

1. Click the `Virtual Machines` tab and select a virtual machine that has been configured to use the SPICE protocol.

2. Ensure USB support is set to `Native`:

   a. Click **Edit**.

   b. Click the `Console` tab.

   c. Select `Native` from the `USB Support` drop-down list.

   d. Click `OK`.

3. Click the `Console Options` button and select the `Enable USB Auto-Share` check box.

4. Start the virtual machine and click the `Console` button to connect to that virtual machine. When you plug your USB device into the client machine, it will automatically be redirected to appear on your guest machine.

4.3. Configuring Multiple Monitors

4.3.1. Configuring Multiple Displays for Red Hat Enterprise Linux Virtual Machines

A maximum of four displays can be configured for a single Red Hat Enterprise Linux virtual

machine when connecting to the virtual machine using the SPICE protocol.

1. Start a SPICE session with the virtual machine.

2. Open the **View** drop-down menu at the top of the SPICE client window.

3. Open the **Display** menu.

4. Click the name of a display to enable or disable that display.

> **Note**
>
> By default, **Display 1** is the only display that is enabled on starting a SPICE session with a virtual machine. If no other displays are enabled, disabling this display will close the session.

4.3.2. Configuring Multiple Displays for Windows Virtual Machines

A maximum of four displays can be configured for a single Windows virtual machine when connecting to the virtual machine using the SPICE protocol.

1. Click the **Virtual Machines** tab and select a virtual machine.

2. With the virtual machine in a powered-down state, click **Edit**.

3. Click the **Console** tab.

4. Select the number of displays from the **Monitors** drop-down list.

> **Note**
>
> This setting controls the maximum number of displays that can be enabled for the virtual machine. While the virtual machine is running, additional displays can be enabled up to this number.

5. Click **Ok**.

6. Start a SPICE session with the virtual machine.

7. Open the **View** drop-down menu at the top of the SPICE client window.

8. Open the **Display** menu.

9. Click the name of a display to enable or disable that display.

> **Note**
>
> By default, **Display 1** is the only display that is enabled on starting a SPICE session with a virtual machine. If no other displays are enabled, disabling this display will close the session.

4.4. Configuring Console Options

4.4.1. Console Options

Connection protocols are the underlying technology used to provide graphical consoles for virtual machines and allow users to work with virtual machines in a similar way as they would with physical machines. Red Hat Virtualization currently supports the following connection protocols:

SPICE

Simple Protocol for Independent Computing Environments (SPICE) is the recommended connection protocol for both Linux virtual machines and Windows virtual machines. To open a console to a virtual machine using SPICE, use Remote Viewer.

VNC

Virtual Network Computing (VNC) can be used to open consoles to both Linux virtual machines and Windows virtual machines. To open a console to a virtual machine using VNC, use Remote Viewer or a VNC client.

RDP

Remote Desktop Protocol (RDP) can only be used to open consoles to Windows virtual machines, and is only available when you access a virtual machines from a Windows machine on which Remote Desktop has been installed. Before you can connect to a Windows virtual machine using RDP, you must set up remote sharing on the virtual machine and configure the firewall to allow remote desktop connections.

> **Note**
>
> SPICE is not currently supported on virtual machines running Windows 8. If a Windows 8 virtual machine is configured to use the SPICE protocol, it will detect the absence of the required SPICE drivers and automatically fall back to usingRDP.

4.4.1.1. Accessing Console Options

You can configure several options for opening graphical consoles for virtual machines, such as the method of invocation and whether to enable or disable USB redirection.

Procedure 4.7. Accessing Console Options

1. Select a running virtual machine.

2. Open the `Console Options` window.

   A. In the Administration Portal, right-click the virtual machine and click `Console Options`.

   B. In the User Portal, click the `Edit Console Options` button.
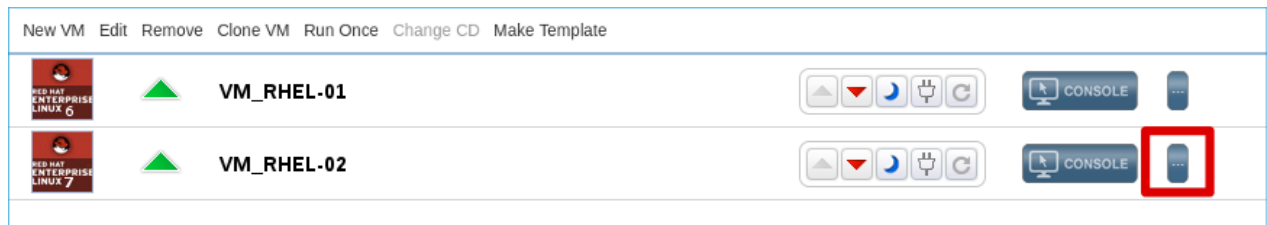
Figure 4.1. The User Portal Edit Console Options Button

> **Note**
>
> Further options specific to each of the connection protocols, such as the keyboard layout when using the VNC connection protocol, can be configured in the **Console** tab of the **Edit Virtual Machine** window.

4.4.1.2. SPICE Console Options

When the SPICE connection protocol is selected, the following options are available in the **Console Options** window.
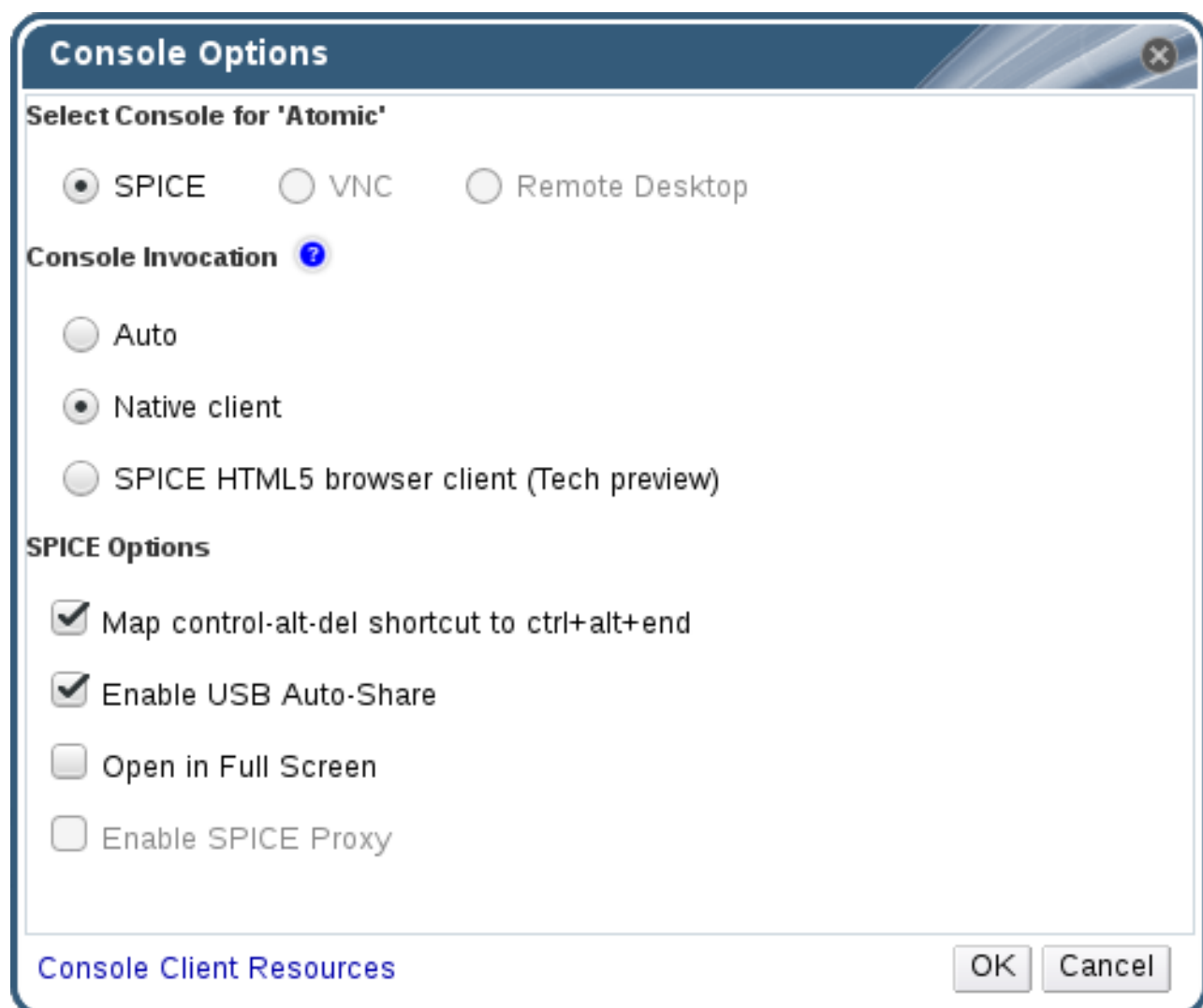
Figure 4.2. The Console Options window

Console Invocation

- **Auto**: The Manager automatically selects the method for invoking the console.

- **Native client**: When you connect to the console of the virtual machine, a file download dialog provides you with a file that opens a console to the virtual machine via Remote Viewer.

- **SPICE HTML5 browser client (Tech preview)**: When you connect to the console of the virtual machine, a browser tab is opened that acts as the console.

SPICE Options

- **Map control-alt-del shortcut to ctrl+alt+end**: Select this check box to map the **Ctrl**+**Alt**+**Del** key combination to **Ctrl**+**Alt**+**End** inside the virtual machine.

- **Enable USB Auto-Share**: Select this check box to automatically redirect USB devices to the virtual machine. If this option is not selected, USB devices will connect to the client machine instead of the guest virtual machine. To use the USB device on the guest machine, manually enable it in the SPICE client menu.

- **Open in Full Screen**: Select this check box for the virtual machine console to automatically open in full screen when you connect to the virtual machine. Press **SHIFT**+**F11** to toggle full screen mode on or off.

- **Enable SPICE Proxy**: Select this check box to enable the SPICE proxy.

- **Enable WAN options**: Select this check box to set the parameters **WANDisableEffects** and **WANColorDepth** to **animation** and **16** bits respectively on Windows virtual machines. Bandwidth in WAN environments is limited and this option prevents certain Windows settings from consuming too much bandwidth.

4.4.1.3. VNC Console Options

When the VNC connection protocol is selected, the following options are available in the **Console Options** window.
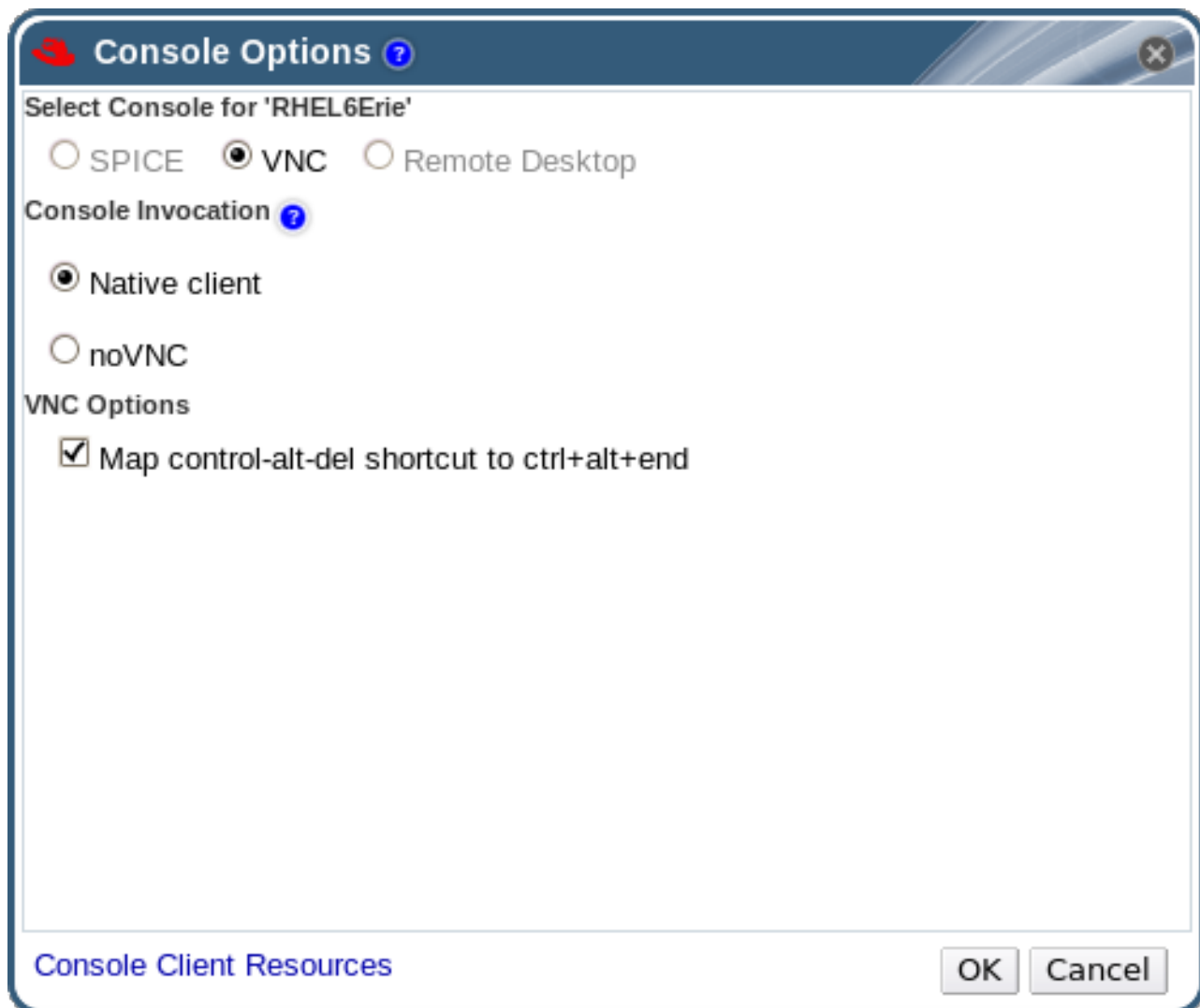
Figure 4.3. The Console Options window

Console Invocation

» **Native Client**: When you connect to the console of the virtual machine, a file download dialog provides you with a file that opens a console to the virtual machine via Remote Viewer.

» **noVNC**: When you connect to the console of the virtual machine, a browser tab is opened that acts as the console.

VNC Options

» **Map control-alt-delete shortcut to ctrl+alt+end**: Select this check box to map the **Ctrl**+**Alt**+**Del** key combination to **Ctrl**+**Alt**+**End** inside the virtual machine.

4.4.1.4. RDP Console Options

When the RDP connection protocol is selected, the following options are available in the **Console Options** window.
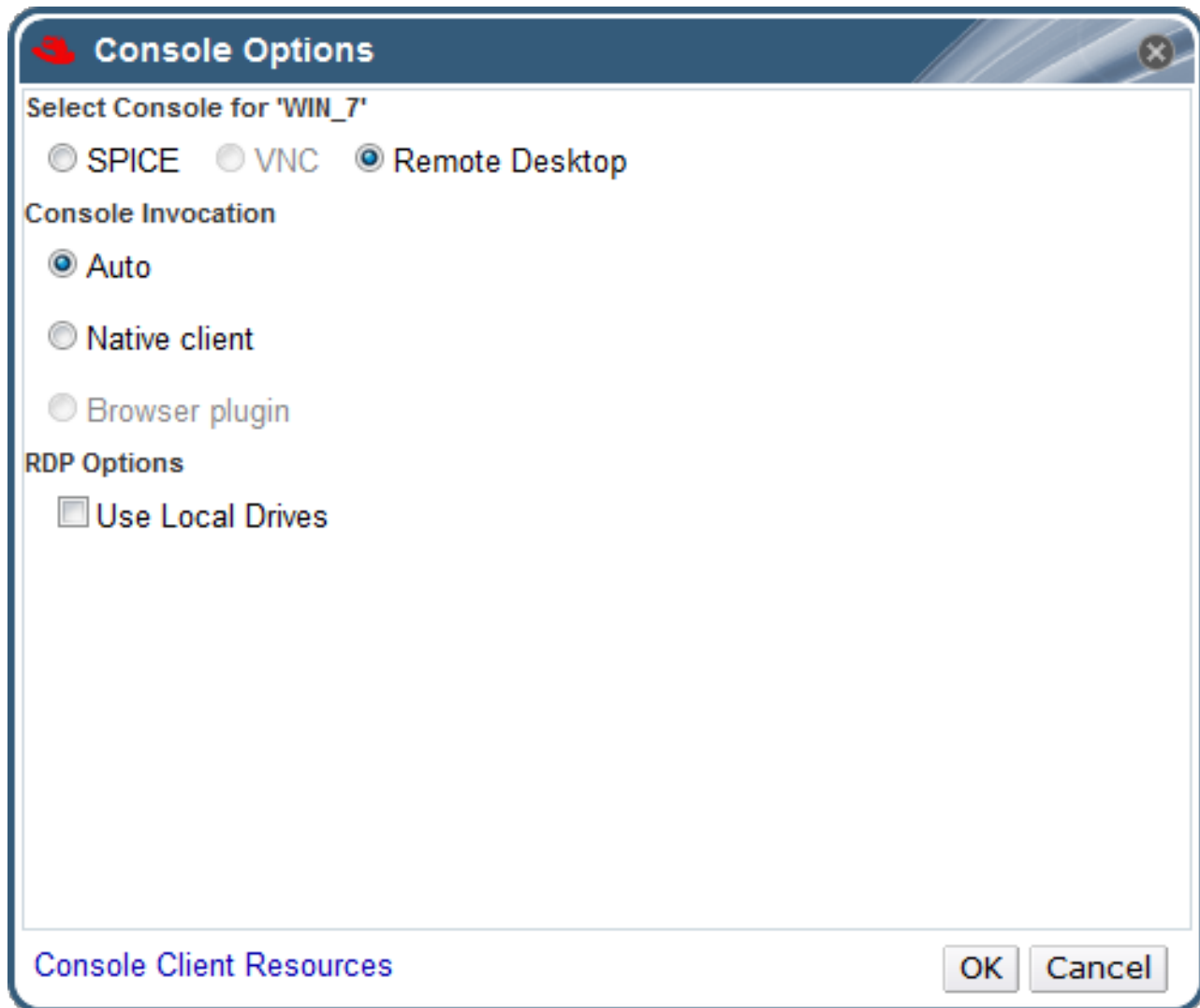
Figure 4.4. The Console Options window

Console Invocation

» **Auto**: The Manager automatically selects the method for invoking the console.

» **Native client**: When you connect to the console of the virtual machine, a file download dialog provides you with a file that opens a console to the virtual machine via Remote Desktop.

RDP Options

» **Use Local Drives**: Select this check box to make the drives on the client machine accessible on the guest virtual machine.

4.4.2. Remote Viewer Options

4.4.2.1. Remote Viewer Options

When you specify the **Native client** console invocation option, you will connect to virtual machines using Remote Viewer. The Remote Viewer window provides a number of options for interacting with the virtual machine to which it is connected.
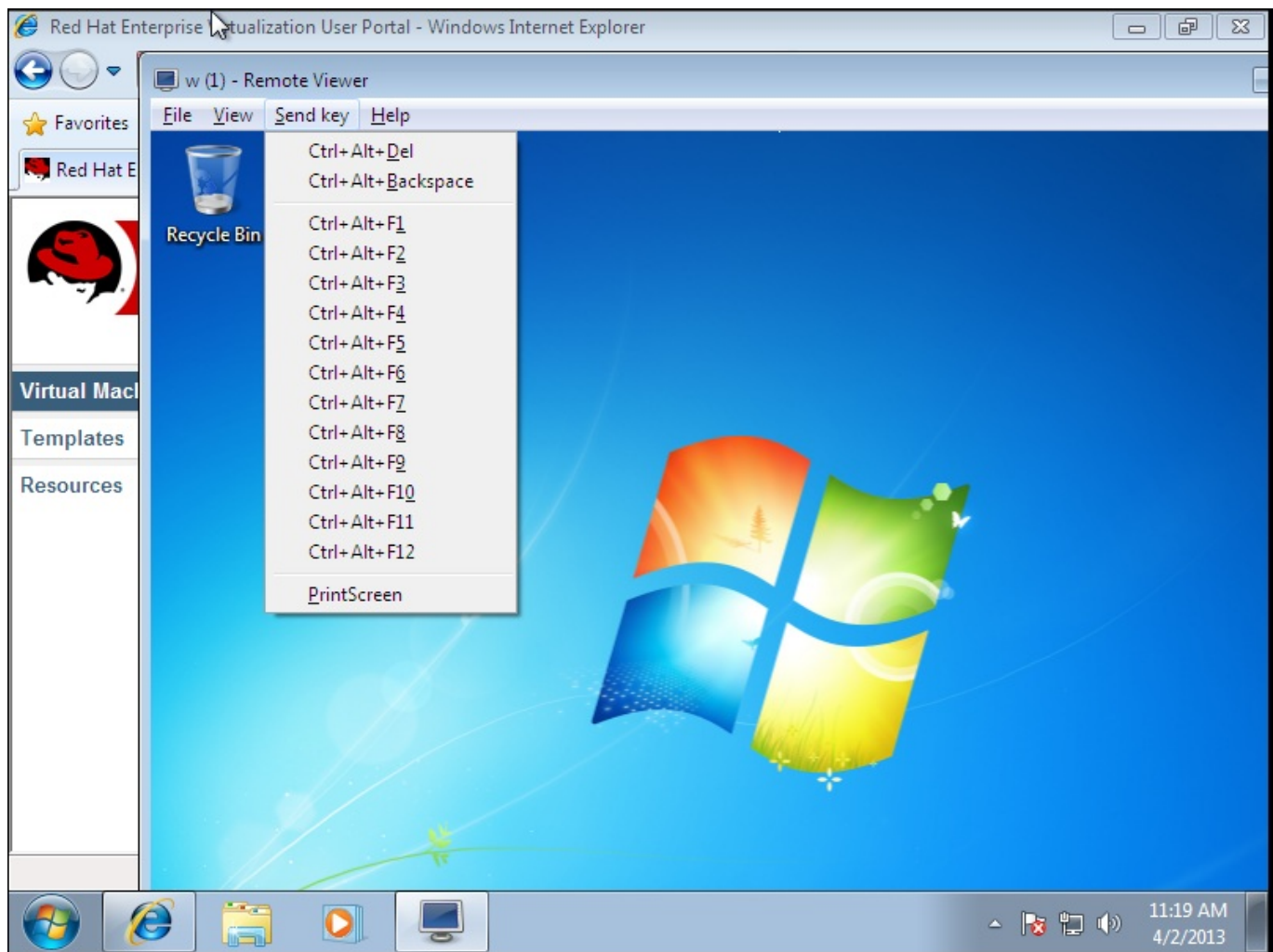
Figure 4.5. The Remote Viewer connection menu

Table 4.1. Remote Viewer Options

Option	Hotkey
File	» **Screenshot**: Takes a screen capture of the active window and saves it in a location of your specification. » **USB device selection**: If USB redirection has been enabled on your virtual machine, the USB device plugged into your client machine can be accessed from this menu. » **Quit**: Closes the console. The hot key for this option is **Shift**+**Ctrl**+**Q**.

Option	Hotkey
View	» **Full screen**: Toggles full screen mode on or off. When enabled, full screen mode expands the virtual machine to fill the entire screen. When disabled, the virtual machine is displayed as a window. The hot key for enabling or disabling full screen is **SHIFT**+**F11**. » **Zoom**: Zooms in and out of the console window. **Ctrl++** zooms in, **Ctrl+-** zooms out, and **Ctrl+0** returns the screen to its original size. » **Automatically resize**: Tick to enable the guest resolution to automatically scale according to the size of the console window. » **Displays**: Allows users to enable and disable displays for the guest virtual machine.
Send key	» **Ctrl**+**Alt**+**Del**: On a Red Hat Enterprise Linux virtual machine, it displays a dialog with options to suspend, shut down or restart the virtual machine. On a Windows virtual machine, it displays the task manager or Windows Security dialog. » **Ctrl**+**Alt**+**Backspace**: On a Red Hat Enterprise Linux virtual machine, it restarts the X sever. On a Windows virtual machine, it does nothing. » **Ctrl**+**Alt**+**F1** » **Ctrl**+**Alt**+**F2** » **Ctrl**+**Alt**+**F3** » **Ctrl**+**Alt**+**F4** » **Ctrl**+**Alt**+**F5** » **Ctrl**+**Alt**+**F6** » **Ctrl**+**Alt**+**F7** » **Ctrl**+**Alt**+**F8** » **Ctrl**+**Alt**+**F9** » **Ctrl**+**Alt**+**F10** » **Ctrl**+**Alt**+**F11** » **Ctrl**+**Alt**+**F12** » **Printscreen**: Passes the **Printscreen** keyboard option to the virtual machine.
Help	The **About** entry displays the version details of Virtual Machine Viewer that you are using.
Release Cursor from Virtual Machine	**SHIFT**+**F12**

4.4.2.2. Remote Viewer Hotkeys

You can access the hotkeys for a virtual machine in both full screen mode and windowed mode. If you are using full screen mode, you can display the menu containing the button for hotkeys by moving the mouse pointer to the middle of the top of the screen. If you are using windowed mode, you can access the hotkeys via the **Send key** menu on the virtual machine window title bar.

> **Note**
>
> If **vdagent** is not running on the client machine, the mouse can become captured in a virtual machine window if it is used inside a virtual machine and the virtual machine is not in full screen. To unlock the mouse, press **Shift+F12**.

4.4.2.3. Manually Associating console.vv Files with Remote Viewer

If you are prompted to download a **console.vv** file when attempting to open a console to a virtual machine using the native client console option, and Remote Viewer is already installed, then you can manually associate **console.vv** files with Remote Viewer so that Remote Viewer can automatically use those files to open consoles.

Procedure 4.8. Manually Associating console.vv Files with Remote Viewer

1. Start the virtual machine.

2. Open the **Console Options** window.

   A. In the Administration Portal, right-click the virtual machine and click **Console Options**.

   B. In the User Portal, click the **Edit Console Options** button.
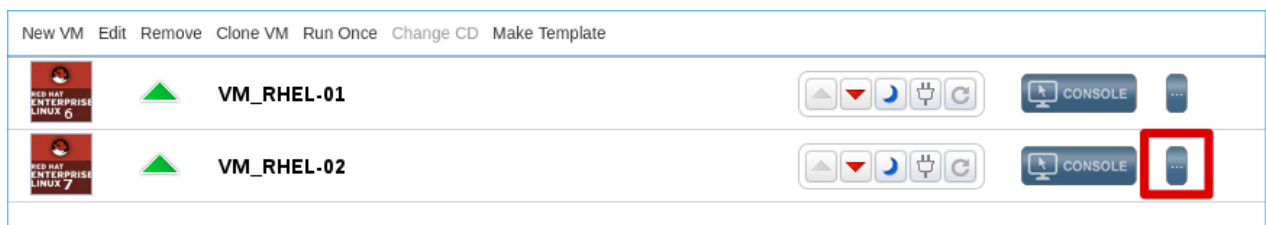
Figure 4.6. The User Portal Edit Console Options Button

3. Change the console invocation method to **Native client** and click **OK**.

4. Attempt to open a console to the virtual machine, then click **Save** when prompted to open or save the **console.vv** file.

5. Navigate to the location on your local machine where you saved the file.

6. Double-click the **console.vv** file and select **Select a program from a list of installed programs** when prompted.

7. In the **Open with** window, select **Always use the selected program to open this kind of file** and click the **Browse** button.

8. Navigate to the **C:\Users\[user name]\AppData\Local\virt-viewer\bin** directory and select **remote-viewer.exe**.

9. Click **Open** and then click **OK**.

When you use the native client console invocation option to open a console to a virtual machine, Remote Viewer will automatically use the **console.vv** file that the Red Hat Virtualization Manager provides to open a console to that virtual machine without prompting you to select the application to use.

4.5. Configuring a Watchdog

4.5.1. Adding a Watchdog Card to a Virtual Machine

You can add a watchdog card to a virtual machine to monitor the operating system's responsiveness.

Procedure 4.9. Adding Watchdog Cards to Virtual Machines

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **High Availability** tab.

4. Select the watchdog model to use from the **Watchdog Model** drop-down list.

5. Select an action from the **Watchdog Action** drop-down list. This is the action that the virtual machine takes when the watchdog is triggered.

6. Click **OK**.

4.5.2. Installing a Watchdog

To activate a watchdog card attached to a virtual machine, you must install the *watchdog* package on that virtual machine and start the **watchdog** service.

Procedure 4.10. Installing Watchdogs

1. Log in to the virtual machine on which the watchdog card is attached.

2. Install the *watchdog* package and dependencies:

   ```
   # yum install watchdog
   ```

3. Edit the **/etc/watchdog.conf** file and uncomment the following line:

   ```
   watchdog-device = /dev/watchdog
   ```

4. Save the changes.

5. Start the **watchdog** service and ensure this service starts on boot:

   A. Red Hat Enterprise Linux 6:

      ```
      # service watchdog start
      # chkconfig watchdog on
      ```

B. Red Hat Enterprise Linux 7:

```
# systemctl start watchdog.service
# systemctl enable watchdog.service
```

4.5.3. Confirming Watchdog Functionality

Confirm that a watchdog card has been attached to a virtual machine and that the **watchdog** service is active.

> ⚠️ **Warning**
>
> This procedure is provided for testing the functionality of watchdogs only and must not be run on production machines.

Procedure 4.11. Confirming Watchdog Functionality

1. Log in to the virtual machine on which the watchdog card is attached.

2. Confirm that the watchdog card has been identified by the virtual machine:

   ```
   # lspci | grep watchdog -i
   ```

3. Run one of the following commands to confirm that the watchdog is active:

   A. Trigger a kernel panic:

      ```
      # echo c > /proc/sysrq-trigger
      ```

   B. Terminate the **watchdog** service:

      ```
      # kill -9 `pgrep watchdog`
      ```

The watchdog timer can no longer be reset, so the watchdog counter reaches zero after a short period of time. When the watchdog counter reaches zero, the action specified in the **Watchdog Action** drop-down menu for that virtual machine is performed.

4.5.4. Parameters for Watchdogs in watchdog.conf

The following is a list of options for configuring the **watchdog** service available in the **/etc/watchdog.conf** file. To configure an option, you must uncomment that option and restart the **watchdog** service after saving the changes.

> 📝 **Note**
>
> For a more detailed explanation of options for configuring the **watchdog** service and using the **watchdog** command, see the **watchdog** man page.

Table 4.2. watchdog.conf variables

Variable name	Default Value	Remarks
`ping`	N/A	An IP address that the watchdog attempts to ping to verify whether that address is reachable. You can specify multiple IP addresses by adding additional **ping** lines.
`interface`	N/A	A network interface that the watchdog will monitor to verify the presence of network traffic. You can specify multiple network interfaces by adding additional **interface** lines.
`file`	`/var/log/messages`	A file on the local system that the watchdog will monitor for changes. You can specify multiple files by adding additional **file** lines.
`change`	**1407**	The number of watchdog intervals after which the watchdog checks for changes to files. A **change** line must be specified on the line directly after each **file** line, and applies to the **file** line directly above that **change** line.
`max-load-1`	**24**	The maximum average load that the virtual machine can sustain over a one-minute period. If this average is exceeded, then the watchdog is triggered. A value of **0** disables this feature.
`max-load-5`	**18**	The maximum average load that the virtual machine can sustain over a five-minute period. If this average is exceeded, then the watchdog is triggered. A value of **0** disables this feature. By default, the value of this variable is set to a value approximately three quarters that of **max-load-1**.
`max-load-15`	**12**	The maximum average load that the virtual machine can sustain over a fifteen-minute period. If this average is exceeded, then the watchdog is triggered. A value of **0** disables this feature. By default, the value of this variable is set to a value approximately one half that of **max-load-1**.
`min-memory`	**1**	The minimum amount of virtual memory that must remain free on the virtual machine. This value is measured in pages. A value of **0** disables this feature.

Variable name	Default Value	Remarks
repair-binary	/usr/sbin/repair	The path and file name of a binary file on the local system that will be run when the watchdog is triggered. If the specified file resolves the issues preventing the watchdog from resetting the watchdog counter, then the watchdog action is not triggered.
test-binary	N/A	The path and file name of a binary file on the local system that the watchdog will attempt to run during each interval. A test binary allows you to specify a file for running user-defined tests.
test-timeout	N/A	The time limit, in seconds, for which user-defined tests can run. A value of **0** allows user-defined tests to continue for an unlimited duration.
temperature-device	N/A	The path to and name of a device for checking the temperature of the machine on which the **watchdog** service is running.
max-temperature	**120**	The maximum allowed temperature for the machine on which the **watchdog** service is running. The machine will be halted if this temperature is reached. Unit conversion is not taken into account, so you must specify a value that matches the watchdog card being used.
admin	**root**	The email address to which email notifications are sent.
interval	**10**	The interval, in seconds, between updates to the watchdog device. The watchdog device expects an update at least once every minute, and if there are no updates over a one-minute period, then the watchdog is triggered. This one-minute period is hard-coded into the drivers for the watchdog device, and cannot be configured.
logtick	**1**	When verbose logging is enabled for the **watchdog** service, the **watchdog** service periodically writes log messages to the local system. The **logtick** value represents the number of watchdog intervals after which a message is written.

Variable name	Default Value	Remarks
`realtime`	`yes`	Specifies whether the watchdog is locked in memory. A value of **yes** locks the watchdog in memory so that it is not swapped out of memory, while a value of **no** allows the watchdog to be swapped out of memory. If the watchdog is swapped out of memory and is not swapped back in before the watchdog counter reaches zero, then the watchdog is triggered.
`priority`	`1`	The schedule priority when the value of `realtime` is set to `yes`.
`pidfile`	`/var/run/syslogd.pid`	The path and file name of a PID file that the watchdog monitors to see if the corresponding process is still active. If the corresponding process is not active, then the watchdog is triggered.

4.6. Configuring Virtual NUMA

In the Administration Portal, you can configure virtual NUMA nodes on a virtual machine and pin them to physical NUMA nodes on a host. The host's default policy is to schedule and run virtual machines on any available resources on the host. As a result, the resources backing a large virtual machine that cannot fit within a single host socket could be spread out across multiple NUMA nodes, and over time may be moved around, leading to poor and unpredictable performance. Configure and pin virtual NUMA nodes to avoid this outcome and improve performance.

Configuring virtual NUMA requires a NUMA-enabled host. To confirm whether NUMA is enabled on a host, log in to the host and run `numactl --hardware`. The output of this command should show at least two NUMA nodes. You can also view the host's NUMA topology in the Administration Portal by selecting the host from the **Hosts** tab and clicking **NUMA Support**. This button is only available when the selected host has at least two NUMA nodes.

Procedure 4.12. Configuring Virtual NUMA

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **Host** tab.

4. Select the **Specific** radio button and select a host from the list. The selected host must have at least two NUMA nodes.

5. Select **Do not allow migration** from the **Migration Options** drop-down list.

6. Enter a number into the **NUMA Node Count** field to assign virtual NUMA nodes to the virtual machine.

7. Select **Strict**, **Preferred**, or **Interleave** from the **Tune Mode** drop-down list. If the selected mode is **Preferred**, the **NUMA Node Count** must be set to **1**.

8. Click **NUMA Pinning**.
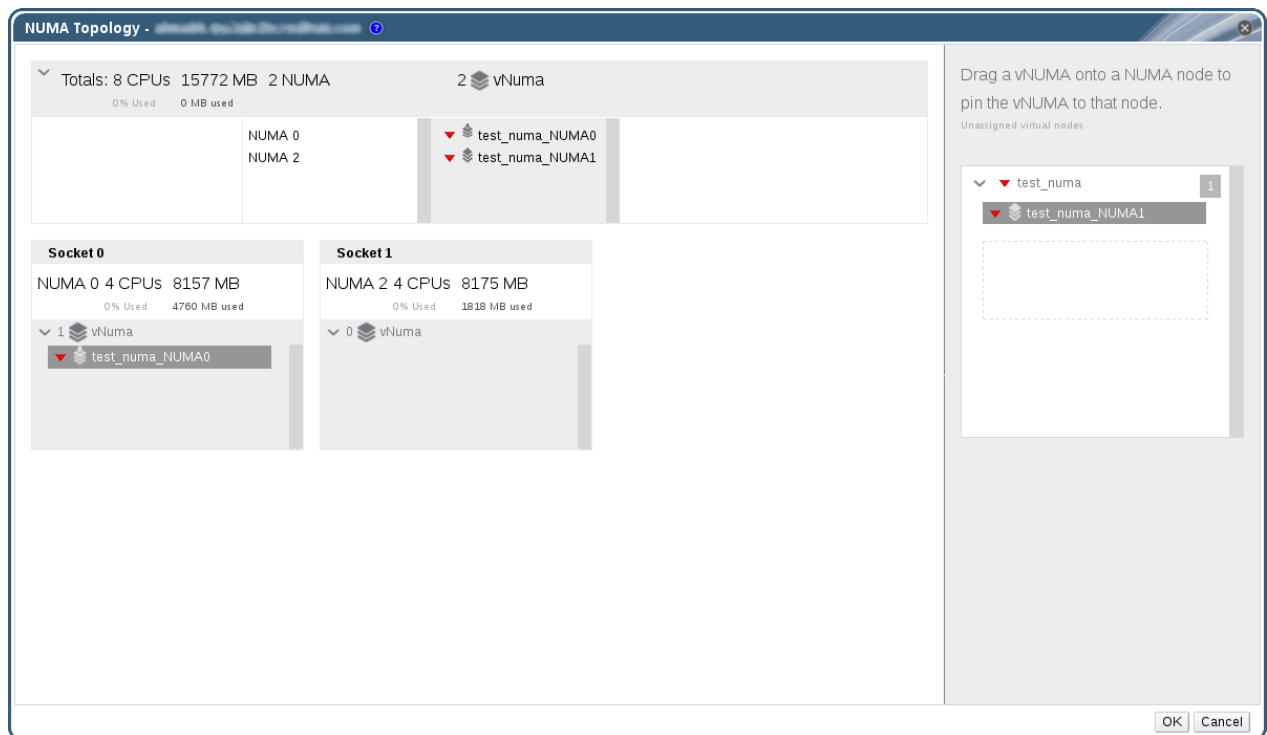
Figure 4.7. The NUMA Topology Window

9. In the **NUMA Topology** window, click and drag virtual NUMA nodes from the box on the right to host NUMA nodes on the left as required, and click **OK**.

10. Click **OK**.

> **Note**
>
> Automatic NUMA balancing is available in Red Hat Enterprise Linux 7, but is not currently configurable through the Red Hat Virtualization Manager.

4.7. Configuring Red Hat Satellite Errata Management for a Virtual Machine

In the Administration Portal, you can configure a virtual machine to display the available errata. The virtual machine needs to be associated with a Red Hat Satellite server to show available errata.

Red Hat Virtualization 4.0 supports errata management with Red Hat Satellite 6.1.

The following prerequisites apply:

❧ The host that the virtual machine runs on also needs to be configured to receive errata information from Satellite. See Configuring Satellite Errata Management for a Host in the *Administration Guide* for more information.

❧ The virtual machine must have the rhevm-guest-agent package installed. This package allows the virtual machine to report its host name to the Red Hat Virtualization Manager.

This allows the Red Hat Satellite server to identify the virtual machine as a content host and report the applicable errata. For more information on installing the ovirt-guest-agent package see Section 2.4.2, "Installing the Guest Agents and Drivers on Red Hat Enterprise Linux" for Red Hat Enterprise Linux virtual machines and Section 3.3.2, "Installing the Guest Agents and Drivers on Windows" for Windows virtual machines.

> **Important**
>
> Virtual machines are identified in the Satellite server by their FQDN. This ensures that an external content host ID does not need to be maintained in Red Hat Virtualization.

Procedure 4.13. Configuring Red Hat Satellite Errata Management

> **Note**
>
> The virtual machine must be registered to the Satellite server as a content host and have the katello-agent package installed.
>
> For more information on how to configure a host registration see Configuring a Host for Registration in the *Red Hat Satellite User Guide* and for more information on how to register a host and install the katello-agent package see Registration in the *Red Hat Satellite User Guide*

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **Foreman/Satellite** tab.

4. Select the required Satellite server from the **Provider** drop-down list.

5. Click **OK**.

Chapter 5. Editing Virtual Machines

5.1. Editing Virtual Machine Properties

Changes to storage, operating system, or networking parameters can adversely affect the virtual machine. Ensure that you have the correct details before attempting to make any changes. Virtual machines can be edited while running, and some changes (listed in the procedure below) will be applied immediately. To apply all other changes, the virtual machine must be shut down and restarted.

Procedure 5.1. Editing Virtual Machines

1. Select the virtual machine to be edited.

2. Click **Edit**.

3. Change settings as required.

   Changes to the following settings are applied immediately:

   » **Name**

   » **Description**

   » **Comment**

   » **Optimized for** (Desktop/Server)

   » **Delete Protection**

   » **Network Interfaces**

   » **Memory Size** (Edit this field to hot plug virtual memory. See Section 5.4, "Hot Plugging Virtual Memory".)

   » **Virtual Sockets** (Edit this field to hot plug CPUs. See Section 5.5, "Hot Plugging Virtual CPUs".)

   » **Use custom migration downtime**

   » **Highly Available**

   » **Priority for Run/Migration queue**

   » **Disable strict user checking**

   » **Icon**

4. Click **OK**.

5. If the **Next Start Configuration** pop-up window appears, click **OK**.

Changes from the list in step 3 are applied immediately. All other changes are applied when you shut down and restart your virtual machine. Until then, an orange icon ( ) appears as a reminder of the pending changes.

5.2. Network Interfaces

5.2.1. Adding a New Network Interface

You can add multiple network interfaces to virtual machines. Doing so allows you to put your virtual machine on multiple logical networks.

Procedure 5.2. Adding Network Interfaces to Virtual Machines

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Network Interfaces** tab in the details pane.

3. Click **New**.
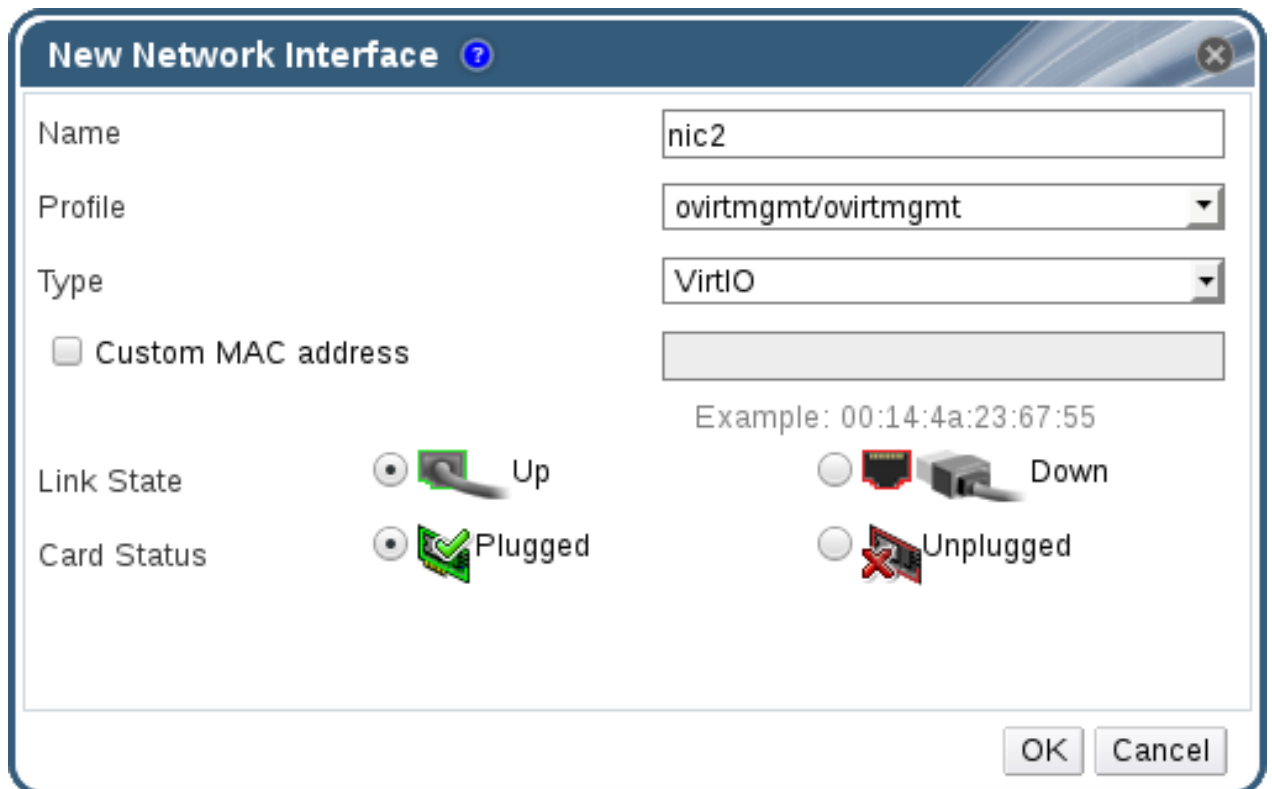
Figure 5.1. New Network Interface window

4. Enter the **Name** of the network interface.

5. Use the drop-down lists to select the **Profile** and the **Type** of the network interface. The **Profile** and **Type** drop-down lists are populated in accordance with the profiles and network types available to the cluster and the network interface cards available to the virtual machine.

6. Select the **Custom MAC address** check box and enter aMAC address for the network interface card as required.

7. Click **OK**.

The new network interface is listed in the **Network Interfaces** tab in the details pane of the virtual machine. The **Link State** is set to **Up** by default when the network interface card is defined on the virtual machine and connected to the network.

For more details on the fields in the **New Network Interface** window, see Section A.2, "Explanation of Settings in the New Network Interface and Edit Network Interface Windows".

5.2.2. Editing a Network Interface

In order to change any network settings, you must edit the network interface. This procedure can be performed on virtual machines that are running, but some actions can be performed only on virtual machines that are not running.

Procedure 5.3. Editing Network Interfaces

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Network Interfaces** tab in the details pane and select the network interface to edit.

3. Click **Edit**. The **Edit Network Interface** window contains the same fields as the **New Network Interface** window.

4. Change settings as required.

5. Click **OK**.

5.2.3. Hot Plugging a Network Interface

You can hot plug network interfaces. Hot plugging means enabling and disabling devices while a virtual machine is running.

> **Note**
>
> The guest operating system must support hot plugging network interfaces.

Procedure 5.4. Hot Plugging Network Interfaces

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Network Interfaces** tab in the details pane and select the network interface to hot plug.

3. Click **Edit**.

4. Set the **Card Status** to **Plugged** to enable the network interface, or set it to **Unplugged** to disable the network interface.

5. Click **OK**.

5.2.4. Removing a Network Interface

Procedure 5.5. Removing Network Interfaces

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Network Interfaces** tab in the details pane and select the network interface to remove.

3. Click **Remove**.

4. Click **OK**.

5.3. Virtual Disks

5.3.1. Adding a New Virtual Disk

You can add multiple virtual disks to a virtual machine.

Image is the default type of disk. You can also add a **Direct LUN** disk or a **Cinder** (OpenStack Volume) disk. **Image** disk creation is managed entirely by the Manager. **Direct LUN** disks require externally prepared targets that already exist. **Cinder** disks require access to an instance of OpenStack Volume that has been added to the Red Hat Virtualization environment using the **External Providers** window; see Adding an OpenStack Volume (Cinder) Instance for Storage Management for more information. Existing disks are either floating disks or shareable disks attached to virtual machines.

Procedure 5.6. Adding Disks to Virtual Machines

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Disks** tab in the details pane.

3. Click **New**.

Figure 5.2. The New Virtual Disk Window

4. Use the appropriate radio buttons to switch between **Image**, **Direct LUN**, or **Cinder**. Virtual disks added in the User Portal can only be **Image** disks. **Direct LUN** and **Cinder** disks can be added in the Administration Portal.

5. Enter a **Size(GB)**, **Alias**, and **Description** for the new disk.

6. Use the drop-down lists and check boxes to configure the disk. See Section A.3, "Explanation of Settings in the New Virtual Disk and Edit Virtual Disk Windows" for more details on the fields for all disk types.

7. Click **OK**.

The new disk appears in the details pane after a short time.

5.3.2. Attaching an Existing Disk to a Virtual Machine

Floating disks are disks that are not associated with any virtual machine.

Floating disks can minimize the amount of time required to set up virtual machines. Designating a floating disk as storage for a virtual machine makes it unnecessary to wait for disk preallocation at the time of a virtual machine's creation.

Floating disks can be attached to a single virtual machine, or to multiple virtual machines if the disk is shareable. Each virtual machine that uses the shared disk can use a different disk interface type.

Once a floating disk is attached to a virtual machine, the virtual machine can access it.

Procedure 5.7. Attaching Virtual Disks to Virtual Machines

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Disks** tab in the details pane.

3. Click **Attach**.
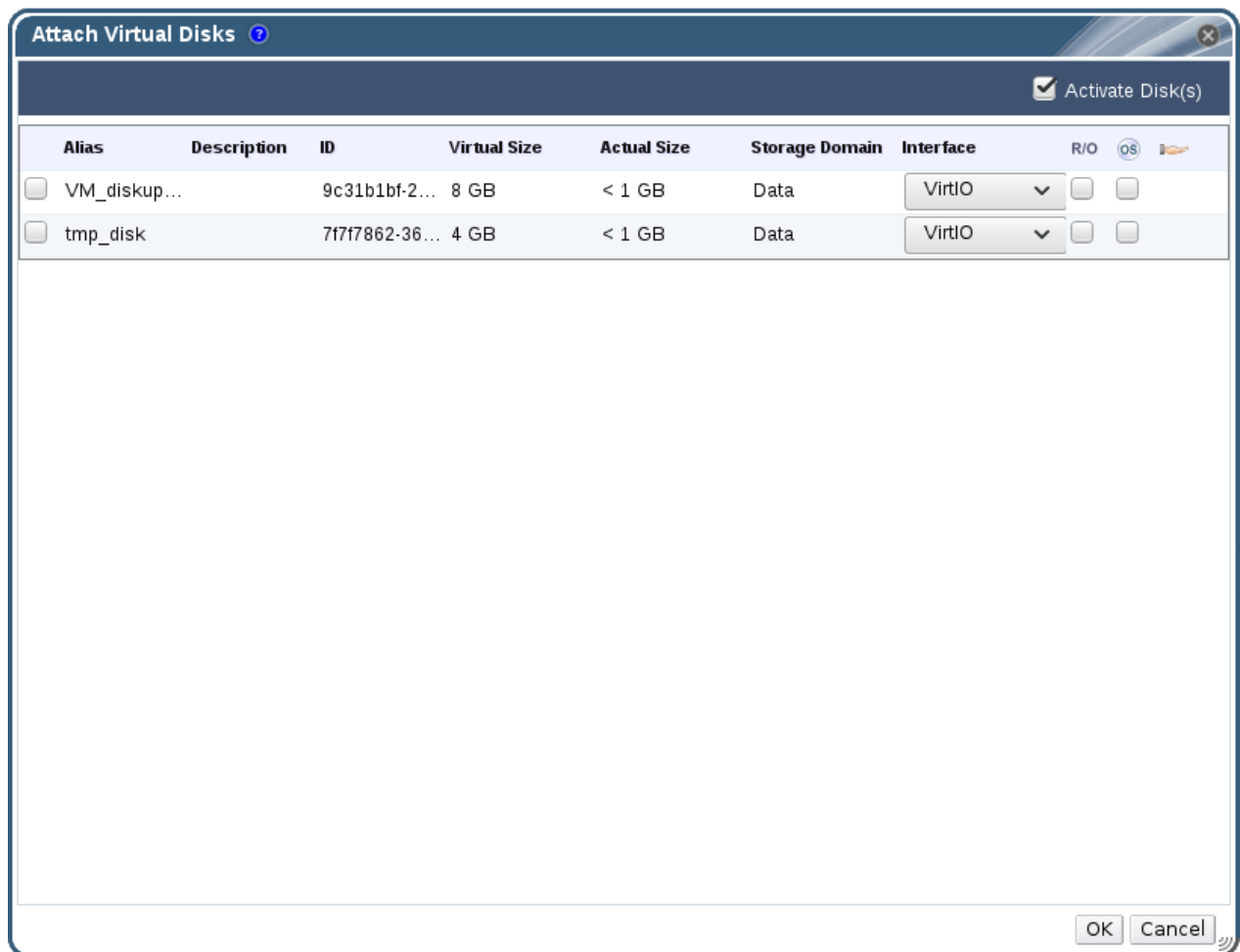
Figure 5.3. The Attach Virtual Disks Window

4. Select one or more virtual disks from the list of available disks and select the required interface from the **Interface** drop-down.

5. Click **OK**.

> ### Note
>
> No Quota resources are consumed by attaching virtual disks to, or detaching virtual disks from, virtual machines.

5.3.3. Extending the Available Size of a Virtual Disk

You can extend the available size of a virtual disk while the virtual disk is attached to a virtual machine. Resizing a virtual disk does not resize the underlying partitions or file systems on that virtual disk. Use the **fdisk** utility to resize the partitions and file systems as required. See How to Resize a Partition using fdisk for more information.

Procedure 5.8. Extending the Available Size of Virtual Disks

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Disks** tab in the details pane and select the disk to edit.

3. Click **Edit**.

4. Enter a value in the **Extend size by(GB)** field.

5. Click **OK**.

The target disk's status becomes **locked** for a short time, during which the drive is resized. When the resizing of the drive is complete, the status of the drive becomes **OK**.

5.3.4. Hot Plugging a Virtual Disk

You can hot plug virtual machine disks. Hot plugging means enabling or disabling devices while a virtual machine is running.

> **Note**
>
> The guest operating system must support hot plugging virtual disks.

Procedure 5.9. Hot Plugging Virtual Disks

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Disks** tab in the details pane and select the virtual disk to hot plug.

3. Click **Activate** to enable the disk, or click**Deactivate** to disable the disk.

4. Click **OK**.

5.3.5. Removing a Virtual Disk from a Virtual Machine

Procedure 5.10. Removing Virtual Disks From Virtual Machines

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Disks** tab in the details pane and select the virtual disk to remove.

3. Click **Deactivate**.

4. Click **OK**.

5. Click **Remove**.

6. Optionally, select the **Remove Permanently** check box to completely remove the virtual disk from the environment. If you do not select this option - for example, because the disk is a shared disk - the virtual disk will remain in the **Disks** resource tab.

7. Click **OK**.

If the disk was created as block storage, for example iSCSI, and the **Wipe After Delete** check box was selected when creating the disk, you can view the log file on the host to confirm that the data has been wiped after permanently removing the disk. See Settings to Wipe Virtual Disks After Deletion in the*Administration Guide*.

5.3.6. Importing a Disk Image from an Imported Storage Domain

Import floating virtual disks from an imported storage domain using the **Disk Import** tab of the

details pane.

This procedure requires access to the Administration Portal

> **Note**
>
> Only QEMU-compatible disks can be imported into the Manager.

Procedure 5.11. Importing a Disk Image

1. Select a storage domain that has been imported into the data center.

2. In the details pane, click **Disk Import**.

3. Select one or more disk images and click **Import** to open the **Import Disk(s)** window.

4. Select the appropriate **Disk Profile** for each disk.

5. Click **OK** to import the selected disks.

5.3.7. Importing an Unregistered Disk Image from an Imported Storage Domain

Import floating virtual disks from a storage domain using the **Disk Import** tab of the details pane. Floating disks created outside of a Red Hat Virtualization environment are not registered with the Manager. Scan the storage domain to identify unregistered floating disks to be imported.

This procedure requires access to the Administration Portal

> **Note**
>
> Only QEMU-compatible disks can be imported into the Manager.

Procedure 5.12. Importing a Disk Image

1. Select a storage domain that has been imported into the data center.

2. Right-click the storage domain and select **Scan Disks** so that the Manager can identify unregistered disks.

3. In the details pane, click **Disk Import**.

4. Select one or more disk images and click **Import** to open the **Import Disk(s)** window.

5. Select the appropriate **Disk Profile** for each disk.

6. Click **OK** to import the selected disks.

5.4. Hot Plugging Virtual Memory

You can hot plug virtual memory. Hot plugging means enabling or disabling devices while a

virtual machine is running. Each time memory is hot plugged, it appears as a new memory device in the **Vm Devices** tab in the details pane, up to a maximum of 16. When the virtual machine is shut down and restarted, these devices are cleared from the **Vm Devices** tab without reducing the virtual machine's memory, allowing you to hot plug more memory devices.

> **Important**
>
> Hot unplugging virtual memory is not currently supported in Red Hat Virtualization.

Procedure 5.13. Hot Plugging Virtual Memory

1. Click the **Virtual Machines** tab and select a running virtual machine.

2. Click **Edit**.

3. Click the **System** tab.

4. Edit the **Memory Size** as required. Memory can be added in multiples of 256 MB.

5. Click **OK**.

   This action opens the **Next Start Configuration** window, as the **MemSizeMb** value will not change until the virtual machine is restarted. However, the hot plug action is triggered by the change to the **memory** value, which can be applied immediately.
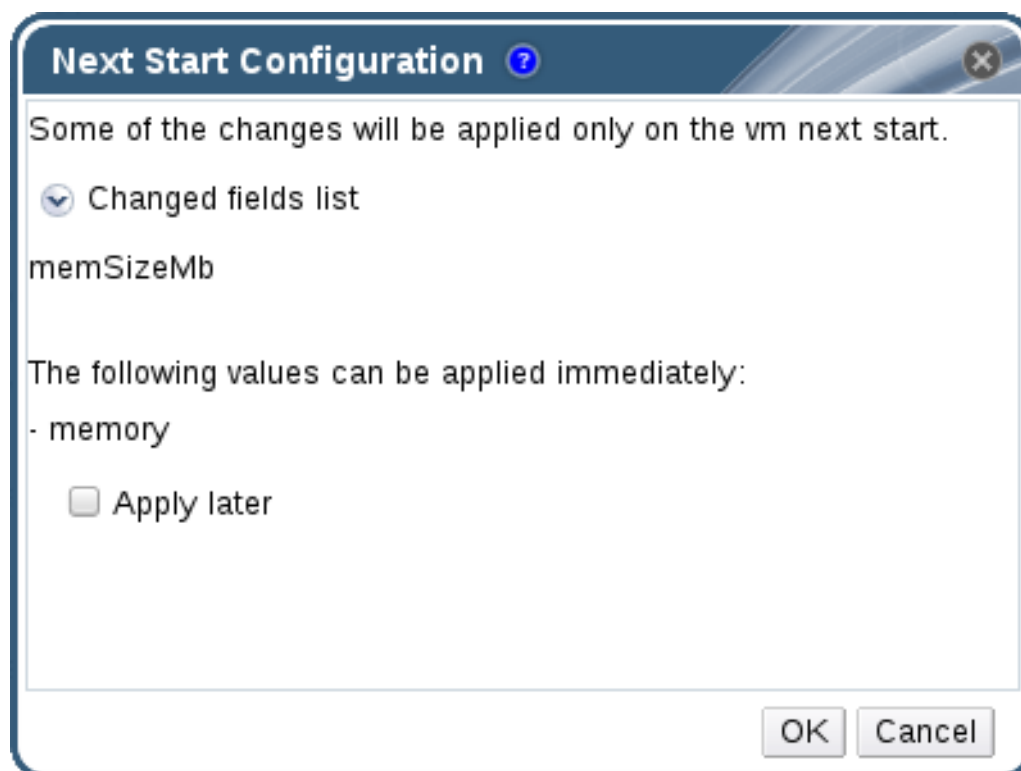
Figure 5.4. Hot Plug Virtual Memory

6. Clear the **Apply later** check box to apply the change immediately.

7. Click **OK**.

The virtual machine's **Defined Memory** is updated in the **General** tab in the details pane. You can see the newly added memory device in the **Vm Devices** tab in the details pane.

5.5. Hot Plugging Virtual CPUs

You can hot plug virtual CPUs. Hot plugging means enabling or disabling devices while a virtual machine is running.

The following prerequisites apply:

» The virtual machine's **Operating System** must be explicitly set in the **New Virtual Machine** window.

» The virtual machine's operating system must support CPU hot plug. See the table below for support details.

» Windows virtual machines must have the guest agents installed. See Section 3.3.2, "Installing the Guest Agents and Drivers on Windows".

> ⭐ **Important**
>
> Hot unplugging virtual CPUs is not currently supported in Red Hat Virtualization.

Procedure 5.14. Hot Plugging Virtual CPUs

1. Click the **Virtual Machines** tab and select a running virtual machine.

2. Click **Edit**.

3. Click the **System** tab.

4. Change the value of **Virtual Sockets** as required.

5. Click **OK**.

Table 5.1. Operating System Support Matrix for vCPU Hot Plug

Operating System	Version	Architecture	Hot Plug Supported
Red Hat Enterprise Linux 6.3+		x86	Yes
Red Hat Enterprise Linux 7.0+		x86	Yes
Microsoft Windows Server 2008	All	x86	No
Microsoft Windows Server 2008	Standard, Enterprise	x64	No
Microsoft Windows Server 2008	Datacenter	x64	Yes
Microsoft Windows Server 2008 R2	All	x86	No
Microsoft Windows Server 2008 R2	Standard, Enterprise	x64	No

Operating System	Version	Architecture	Hot Plug Supported
Microsoft Windows Server 2008 R2	Datacenter	x64	Yes
Microsoft Windows Server 2012	All	x64	Yes
Microsoft Windows Server 2012 R2	All	x64	Yes
Microsoft Windows 7	All	x86	No
Microsoft Windows 7	Starter, Home, Home Premium, Professional	x64	No
Microsoft Windows 7	Enterprise, Ultimate	x64	Yes
Microsoft Windows 8.x	All	x86	Yes
Microsoft Windows 8.x	All	x64	Yes

5.6. Pinning a Virtual Machine to Multiple Hosts

Virtual machines can be pinned to multiple hosts. Multi-host pinning allows a virtual machine to run on a specific subset of hosts within a cluster, instead of one specific host or all hosts in the cluster. The virtual machine cannot run on any other hosts in the cluster even if all of the specified hosts are unavailable. Multi-host pinning can be used to limit virtual machines to hosts with, for example, the same physical hardware configuration.

A virtual machine that is pinned to multiple hosts cannot be live migrated, but in the event of a host failure, any virtual machine configured to be highly available is automatically restarted on one of the other hosts to which the virtual machine is pinned.

> **Note**
>
> High availability is not supported for virtual machines that are pinned to a single host.

Procedure 5.15. Pinning Virtual Machines to Multiple Hosts

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **Host** tab.

4. Select the **Specific** radio button under **Start Running On** and select two or more hosts from the list.

5. Select **Do not allow migration** from the **Migration Options** drop-down list.

6. Click the **High Availability** tab.

7. Select the **Highly Available** check box.

8. Select **Low**, **Medium**, or **High** from the **Priority** drop-down list. When migration is triggered, a queue is created in which the high priority virtual machines are migrated first. If a cluster is running low on resources, only the high priority virtual machines are migrated.

9. Click **OK**.

5.7. Changing the CD for a Virtual Machine

You can change the CD accessible to a virtual machine while that virtual machine is running.

> **Note**
>
> You can only use ISO files that have been added to the ISO domain of the virtual machine's cluster.

Procedure 5.16. Changing the CD for a Virtual Machine

1. Click the **Virtual Machines** tab and select a running virtual machine.

2. Click **Change CD**.

3. Select an option from the drop-down list:

   A. Select an ISO file from the list to eject the CD currently accessible to the virtual machine and mount that ISO file as a CD.

   B. Select **[Eject]** from the list to eject the CD currently accessible to the virtual machine.

4. Click **OK**.

5.8. Smart Card Authentication

Smart cards are an external hardware security feature, most commonly seen in credit cards, but also used by many businesses as authentication tokens. Smart cards can be used to protect Red Hat Virtualization virtual machines.

Procedure 5.17. Enabling Smart Cards

1. Ensure that the smart card hardware is plugged into the client machine and is installed according to manufacturer's directions.

2. Click the **Virtual Machines** tab and select a virtual machine.

3. Click **Edit**.

4. Click the **Console** tab and select the **Smartcard enabled** check box.

5. Click **OK**.

6. Connect to the running virtual machine by clicking the **Console** icon. Smart card authentication is now passed from the client hardware to the virtual machine.

> **Important**
>
> If the Smart card hardware is not correctly installed, enabling the Smart card feature will result in the virtual machine failing to load properly.

Procedure 5.18. Disabling Smart Cards

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **Console** tab, and clear the **Smartcard enabled** check box.

4. Click **OK**.

Procedure 5.19. Configuring Client Systems for Smart Card Sharing

1. Smart cards may require certain libraries in order to access their certificates. These libraries must be visible to the NSS library, which **spice-gtk** uses to provide the smart card to the guest. NSS expects the libraries to provide the PKCS #11 interface.

2. Make sure that the module architecture matches **spice-gtk/remote-viewer**'s architecture. For instance, if you have only the 32b PKCS #11 library available, you must install the 32b build of virt-viewer in order for smart cards to work.

Procedure 5.20. Configuring RHEL clients with CoolKey Smart Card Middleware

» CoolKey Smart Card middleware is a part of Red Hat Enterprise Linux. Install the **Smart card support** group. If the Smart Card Support group is installed on a Red Hat Enterprise Linux system, smart cards are redirected to the guest when Smart Cards are enabled. The following command installs the **Smart card support** group:

```
# yum groupinstall "Smart card support"
```

Procedure 5.21. Configuring RHEL clients with Other Smart Card Middleware

» Register the library in the system's NSS database. Run the following command as root:

```
# modutil -dbdir /etc/pki/nssdb -add "module name" -libfile
/path/to/library.so
```

Procedure 5.22. Configuring Windows Clients

» Red Hat does not provide PKCS #11 support to Windows clients. Libraries that provide PKCS #11 support must be obtained from third parties. When such libraries are obtained, register them by running the following command as a user with elevated privileges:

```
modutil -dbdir %PROGRAMDATA%\pki\nssdb -add "module name" -libfile
C:\Path\to\module.dll
```

Chapter 6. Administrative Tasks

6.1. Shutting Down a Virtual Machine

Procedure 6.1. Shutting Down a Virtual Machine

1. Click the **Virtual Machines** tab and select a running virtual machine.

2. Click the shut down ( ) button.

   Alternatively, right-click the virtual machine and select **Shutdown**.

3. Optionally in the Administration Portal, enter a **Reason** for shutting down the virtual machine in the **Shut down Virtual Machine(s)** confirmation window. This allows you to provide an explanation for the shutdown, which will appear in the logs and when the virtual machine is powered on again.

   > **Note**
   >
   > The virtual machine shutdown **Reason** field will only appear if it has been enabled in the cluster settings. For more information, see Explanation of Settings and Controls in the New Cluster and Edit Cluster Windows in the *Administration Guide*.

4. Click **OK** in the **Shut down Virtual Machine(s)** confirmation window.

The virtual machine shuts down gracefully and the **Status** of the virtual machine changes to **Down**.

6.2. Suspending a Virtual Machine

Suspending a virtual machine is equal to placing that virtual machine into *Hibernate* mode.

Procedure 6.2. Suspending a Virtual Machine

1. Click the **Virtual Machines** tab and select a running virtual machine.

2. Click the Suspend ( ) button.

   Alternatively, right-click the virtual machine and select **Suspend**.

The **Status** of the virtual machine changes to **Suspended**.

6.3. Rebooting a Virtual Machine

Procedure 6.3. Rebooting a Virtual Machine

1. Click the **Virtual Machines** tab and select a running virtual machine.

2. Click the Reboot ( ) button.

   Alternatively, right-click the virtual machine and select **Reboot**.

3. Click **OK** in the **Reboot Virtual Machine(s)** confirmation window.

The **Status** of the virtual machine changes to **Reboot In Progress** before returning to **Up**.

6.4. Removing a Virtual Machine

> **Important**
>
> The **Remove** button is disabled while virtual machines are running; you must shut down a virtual machine before you can remove it.

Procedure 6.4. Removing Virtual Machines

1. Click the **Virtual Machines** tab and select the virtual machine to remove.

2. Click **Remove**.

3. Optionally, select the **Remove Disk(s)** check box to remove the virtual disks attached to the virtual machine together with the virtual machine. If the **Remove Disk(s)** check box is cleared, then the virtual disks remain in the environment as floating disks.

4. Click **OK**.

6.5. Cloning a Virtual Machine

You can clone virtual machines without having to create a template or a snapshot first.

> **Important**
>
> The **Clone VM** button is disabled while virtual machines are running; you must shut down a virtual machine before you can clone it.

Procedure 6.5. Cloning Virtual Machines

1. Click the **Virtual Machines** tab and select the virtual machine to clone.

2. Click **Clone VM**.

3. Enter a **Clone Name** for the new virtual machine.

4. Click **OK**.

6.6. Updating Virtual Machine Guest Agents and Drivers

6.6.1. Updating the Guest Agents and Drivers on Red Hat Enterprise Linux

Update the guest agents and drivers on your Red Hat Enterprise Linux virtual machines to use the latest version.

Procedure 6.6. Updating the Guest Agents and Drivers on Red Hat Enterprise Linux

1. Log in to the Red Hat Enterprise Linux virtual machine.

2. Update the *rhevm-guest-agent-common* package:

   ```
   # yum update rhevm-guest-agent-common
   ```

3. Restart the service:

   ▸ For Red Hat Enterprise Linux 6

      ```
      # service ovirt-guest-agent restart
      ```

   ▸ For Red Hat Enterprise Linux 7

      ```
      # systemctl restart ovirt-guest-agent.service
      ```

6.6.2. Updating the Guest Agents and Drivers on Windows

The guest tools comprise software that allows Red Hat Virtualization Manager to communicate with the virtual machines it manages, providing information such as the IP addresses, memory usage, and applications installed on those virtual machines. The guest tools are distributed as an ISO file that can be attached to guests. This ISO file is packaged as an RPM file that can be installed and upgraded from the machine on which the Red Hat Virtualization Manager is installed.

Procedure 6.7. Updating the Guest Agents and Drivers on Windows

1. On the Red Hat Virtualization Manager, update the Red Hat Virtualization Guest Tools to the latest version:

   ```
   # yum update -y rhev-guest-tools-iso*
   ```

2. Upload the ISO file to your ISO domain, replacing *[ISODomain]* with the name of your ISO domain:

   ```
   engine-iso-uploader --iso-domain=[ISODomain] upload /usr/share/rhev-
   guest-tools-iso/rhev-tools-setup.iso
   ```

   > **Note**
   >
   > The **rhev-tools-setup.iso** file is a symbolic link to the most recently updated ISO file. The link is automatically changed to point to the newest ISO file every time you update the *rhev-guest-tools-iso* package.

3. In the Administration or User Portal, if the virtual machine is running, use the **Change CD** button to attach the latest **rhev-tools-setup.iso** file to each of your virtual machines. If the virtual machine is powered off, click the **Run Once** button and attach the ISO as a CD.

4. Select the CD Drive containing the updated ISO and execute the **RHEV-ToolsSetup.exe** file.

6.7. Viewing Red Hat Satellite Errata for a Virtual Machine

Errata for each virtual machine can be viewed after the Red Hat Virtualization virtual machine has been configured to receive errata information from the Red Hat Satellite server.

For more information on configuring a virtual machine to display available errata see Section 4.7, "Configuring Red Hat Satellite Errata Management for a Virtual Machine"

Procedure 6.8. Viewing Red Hat Satellite Errata

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Errata** tab in the details pane.

6.8. Virtual Machines and Permissions

6.8.1. Managing System Permissions for a Virtual Machine

As the **SuperUser**, the system administrator manages all aspects of the Administration Portal. More specific administrative roles can be assigned to other users. These restricted administrator roles are useful for granting a user administrative privileges that limit them to a specific resource. For example, a **DataCenterAdmin** role has administrator privileges only for the assigned data center with the exception of the storage for that data center, and a **ClusterAdmin** has administrator privileges only for the assigned cluster.

A **UserVmManager** is a system administration role for virtual machines in a data center. This role can be applied to specific virtual machines, to a data center, or to the whole virtualized environment; this is useful to allow different users to manage certain virtual resources.

The user virtual machine administrator role permits the following actions:

» Create, edit, and remove virtual machines.

» Run, suspend, shutdown, and stop virtual machines.

> **Note**
>
> You can only assign roles and permissions to existing users.

Many end users are concerned solely with the virtual machine resources of the virtualized environment. As a result, Red Hat Virtualization provides several user roles which enable the user to manage virtual machines specifically, but not other resources in the data center.

6.8.2. Virtual Machines Administrator Roles Explained

The table below describes the administrator roles and privileges applicable to virtual machine administration.

Table 6.1. Red Hat Virtualization System Administrator Roles

Role	Privileges	Notes
DataCenterAdmin	Data Center Administrator	Possesses administrative permissions for all objects underneath a specific data center except for storage.
ClusterAdmin	Cluster Administrator	Possesses administrative permissions for all objects underneath a specific cluster.
NetworkAdmin	Network Administrator	Possesses administrative permissions for all operations on a specific logical network. Can configure and manage networks attached to virtual machines. To configure port mirroring on a virtual machine network, apply the **NetworkAdmin** role on the network and the **UserVmManager** role on the virtual machine.

6.8.3. Virtual Machine User Roles Explained

The table below describes the user roles and privileges applicable to virtual machine users. These roles allow access to the User Portal for managing and accessing virtual machines, but they do not confer any permissions for the Administration Portal.

Table 6.2. Red Hat Virtualization System User Roles

Role	Privileges	Notes
UserRole	Can access and use virtual machines and pools.	Can log in to the User Portal and use virtual machines and pools.
PowerUserRole	Can create and manage virtual machines and templates.	Apply this role to a user for the whole environment with the **Configure** window, or for specific data centers or clusters. For example, if a PowerUserRole is applied on a data center level, the PowerUser can create virtual machines and templates in the data center. Having a **PowerUserRole** is equivalent to having the **VmCreator**, **DiskCreator**, and **TemplateCreator** roles.
UserVmManager	System administrator of a virtual machine.	Can manage virtual machines and create and use snapshots. A user who creates a virtual machine in the User Portal is automatically assigned the UserVmManager role on the machine.

Role	Privileges	Notes
UserTemplateBasedVm	Limited privileges to only use Templates.	Level of privilege to create a virtual machine by means of a template.
VmCreator	Can create virtual machines in the User Portal.	This role is not applied to a specific virtual machine; apply this role to a user for the whole environment with the **Configure** window. When applying this role to a cluster, you must also apply the **DiskCreator** role on an entire data center, or on specific storage domains.
VnicProfileUser	Logical network and network interface user for virtual machines.	If the **Allow all users to use this Network** option was selected when a logical network is created, **VnicProfileUser** permissions are assigned to all users for the logical network. Users can then attach or detach virtual machine network interfaces to or from the logical network.

6.8.4. Assigning Virtual Machines to Users

If you are creating virtual machines for users other than yourself, you have to assign roles to the users before they can use the virtual machines. Note that permissions can only be assigned to existing users. See Users and Roles in the *Red Hat Virtualization Administration Guide* for details on creating user accounts.

The User Portal supports three default roles: User, PowerUser and UserVmManager. However, customized roles can be configured via the Administration Portal. The default roles are described below.

» A **User** can connect to and use virtual machines. This role is suitable for desktop end users performing day-to-day tasks.

» A **PowerUser** can create virtual machines and view virtual resources. This role is suitable if you are an administrator or manager who needs to provide virtual resources for your employees.

» A **UserVmManager** can edit and remove virtual machines, assign user permissions, use snapshots and use templates. It is suitable if you need to make configuration changes to your virtual environment.

When you create a virtual machine, you automatically inherit **UserVmManager** privileges. This enables you to make changes to the virtual machine and assign permissions to the users you manage, or users who are in your Identity Management (IdM) or RHDS group. See *Administration Guide* for more information.

Procedure 6.9. Assigning Permissions to Users

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Permissions** tab on the details pane.

3. Click **Add**.

4. Enter a name, or user name, or part thereof in the **Search** text box, and click **Go**. A list of possible matches display in the results list.

5. Select the check box of the user to be assigned the permissions.

6. Select **UserRole** from the **Role to Assign** drop-down list.

7. Click **OK**.

The user's name and role display in the list of users permitted to access this virtual machine.

> **Note**
>
> If a user is assigned permissions to only one virtual machine, single sign-on (SSO) can be configured for the virtual machine. With single sign-on enabled, when a user logs in to the User Portal, and then connects to a virtual machine through, for example, a SPICE console, users are automatically logged in to the virtual machine and do not need to type in the user name and password again. Single sign-on can be enabled or disabled on a per virtual machine basis. See Section 4.1, "Configuring Single Sign-On for Virtual Machines" for more information on how to enable and disable single sign-on for virtual machines.

6.8.5. Removing Access to Virtual Machines from Users

Procedure 6.10. Removing Access to Virtual Machines from Users

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Permissions** tab on the details pane.

3. Click **Remove**. A warning message displays, asking you to confirm removal of the selected permissions.

4. To proceed, click **OK**. To abort, click **Cancel**.

6.9. Snapshots

6.9.1. Creating a Snapshot of a Virtual Machine

A snapshot is a view of a virtual machine's operating system and applications on any or all available disks at a given point in time. Take a snapshot of a virtual machine before you make a change to it that may have unintended consequences. You can use a snapshot to return a virtual machine to a previous state.

> **Important**
>
> Before taking a live snapshot of a virtual machine using OpenStack Volume (Cinder) disks, you must freeze and thaw the guest filesystem manually. This cannot be done with the Manager, and must be executed using the REST API. See Freeze Virtual Machine Filesystems Action in the *Red Hat Virtualization REST API Guide* for more information.

Procedure 6.11. Creating a Snapshot of a Virtual Machine

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Snapshots** tab in the details pane and click **Create**.

Figure 6.1. Create snapshot

3. Enter a description for the snapshot.

4. Select **Disks to include** using the check boxes.

5. Use the **Save Memory** check box to denote whether to include the virtual machine's memory in the snapshot.

6. Click **OK**.

> **Note**
>
> If you are taking a snapshot of a virtual machine with an OpenStack Volume (Cinder) disk, you must thaw the guest filesystem when the snapshot is complete using the REST API. See Thaw Virtual Machine Filesystems Action in the *REST API Guide* for instructions.

The virtual machine's operating system and applications on the selected disk(s) are stored in a snapshot that can be previewed or restored. The snapshot is created with a status of **Locked**, which changes to **Ok**. When you click on the snapshot, its details are shown on the **General**, **Disks**, **Network Interfaces**, and **Installed Applications** tabs in the right side-pane of the details pane.

6.9.2. Using a Snapshot to Restore a Virtual Machine

A snapshot can be used to restore a virtual machine to its previous state.

Procedure 6.12. Using Snapshots to Restore Virtual Machines

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Snapshots** tab in the details pane to list the available snapshots.

3. Select a snapshot to restore in the left side-pane. The snapshot details display in the right side-pane.

4. Click the drop-down menu beside **Preview** to open the **Custom Preview Snapshot** window.
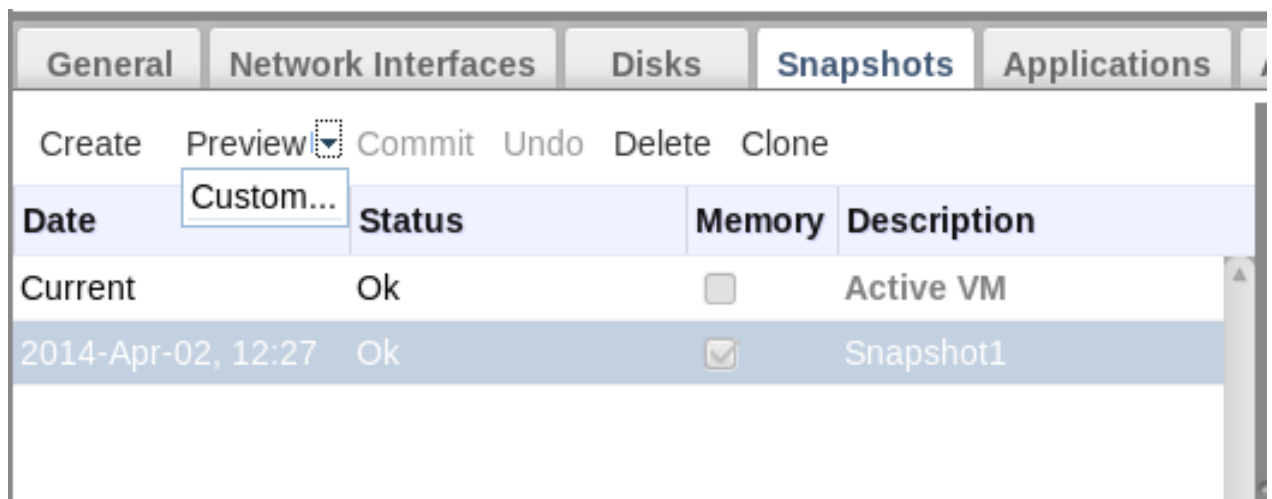
Figure 6.2. Custom Preview Snapshot

5. Use the check boxes to select the **VM Configuration**, **Memory**, and disk(s) you want to restore, then click **OK**. This allows you to create and restore from a customized snapshot using the configuration and disk(s) from multiple snapshots.
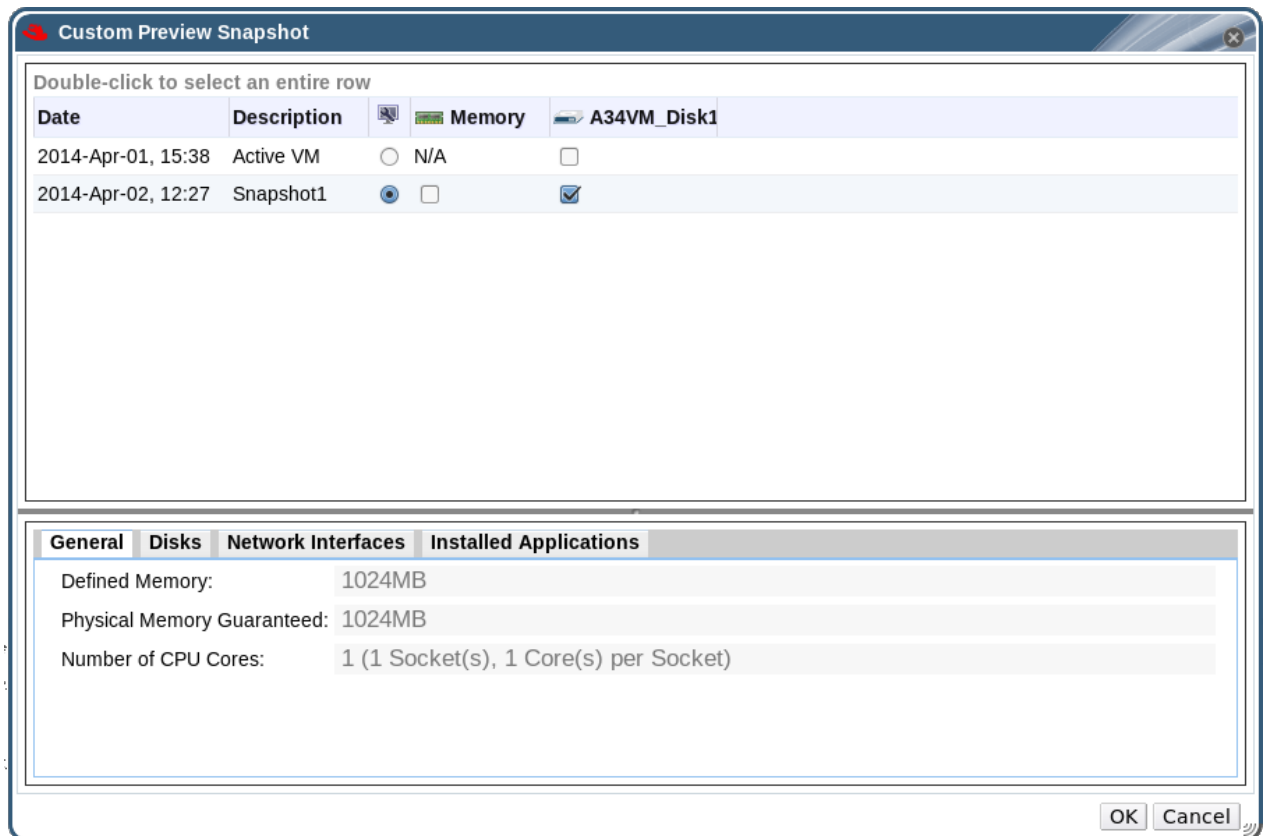
Figure 6.3. The Custom Preview Snapshot Window

The status of the snapshot changes to **Preview Mode**. The status of the virtual machine briefly changes to **Image Locked** before returning to **Down**.

6. Start the virtual machine; it runs using the disk image of the snapshot.

7. Click **Commit** to permanently restore the virtual machine to the condition of the snapshot. Any subsequent snapshots are erased.

   Alternatively, click the **Undo** button to deactivate the snapshot and return the virtual machine to its previous state.

6.9.3. Creating a Virtual Machine from a Snapshot

You have created a snapshot from a virtual machine. Now you can use that snapshot to create another virtual machine.

Procedure 6.13. Creating a virtual machine from a snapshot

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Snapshots** tab in the details pane to list the available snapshots.

3. Select a snapshot in the list displayed and click **Clone**.

4. Enter the **Name** and **Description** for the virtual machine.

Figure 6.4. Clone a Virtual Machine from a Snapshot

5. Click **OK**.

After a short time, the cloned virtual machine appears in the **Virtual Machines** tab in the navigation pane with a status of **Image Locked**. The virtual machine will remain in this state until Red Hat Virtualization completes the creation of the virtual machine. A virtual machine with a preallocated 20 GB hard drive takes about fifteen minutes to create. Sparsely-allocated virtual disks take less time to create than do preallocated virtual disks.

When the virtual machine is ready to use, its status changes from **Image Locked** to **Down** in the **Virtual Machines** tab in the navigation pane.

6.9.4. Deleting a Snapshot

You can delete a virtual machine snapshot and permanently remove it from your Red Hat Virtualization environment. This operation is supported on a running virtual machine and does not require the virtual machine to be in a down state.

> **Important**
>
> When you delete a snapshot from an image chain, one of three things happens:
>
> » If the snapshot being deleted is contained in a RAW (preallocated) base image, a new volume is created that is the same size as the base image.
> » If the snapshot being deleted is contained in a QCOW2 (thin provisioned) base image, the volume subsequent to the volume containing the snapshot being deleted is extended to the cumulative size of the successor volume and the base volume.
> » If the snapshot being deleted is contained in a QCOW2 (thin provisioned), non-base image hosted on internal storage, the successor volume is extended to the cumulative size of the successor volume and the volume containing the snapshot being deleted.
>
> The data from the two volumes is merged in the new or resized volume. The new or resized volume grows to accommodate the total size of the two merged images; the new volume size will be, at most, the sum of the two merged images. To delete a snapshot, you must have enough free space in the storage domain to temporarily accommodate both the original volume and the newly merged volume. Otherwise, snapshot deletion will fail and you will need to export and re-import the volume to remove snapshots. For detailed information on snapshot deletion for all disk formats, see https://access.redhat.com/solutions/527613.

Procedure 6.14. Deleting a Snapshot

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Snapshots** tab in the details pane to list the snapshots for that virtual machine.

Figure 6.5. Snapshot List

3. Select the snapshot to delete.

4. Click **Delete**.

5. Click **OK**.

6.10. Host Devices

6.10.1. Adding a Host Device to a Virtual Machine

Virtual machines can be directly attached to the host devices for improved performance if a compatible host has been configured for direct device assignment.

Procedure 6.15. Adding Host Devices to a Virtual Machine

1. Select a virtual machine and click the **Host Devices** tab in the details pane to list the host devices already attached to this virtual machine. A virtual machine can only have devices attached from the same host. If a virtual machine has attached devices from one host, and you attach a device from another host, the attached devices from the previous host will be automatically removed.

   Attaching host devices to a virtual machine requires the virtual machine to be in a **Down** state. If the virtual machine is running, the changes will not take effect until after the virtual machine has been shut down.

2. Click **Add device** to open the **Add Host Devices** window.

3. Use the **Pinned Host** dropdown menu to select a host.

4. Use the **Capability** dropdown menu to list the **pci**, **scsi**, or **usb_device** host devices.

5. Select the check boxes of the devices to attach to the virtual machine from the **Available Host Devices** pane and click the directional arrow button to transfer these devices to the **Host Devices to be attached** pane, creating a list of the devices to attach to the virtual machine.

6. When you have transferred all desired host devices to the **Host Devices to be attached** pane, click **OK** to attach these devices to the virtual machine and close the window.

These host devices will be attached to the virtual machine when the virtual machine is next powered on.

6.10.2. Removing Host Devices from a Virtual Machine

Remove a host device from a virtual machine to which it has been directly attached using the details pane of the virtual machine.

If you are removing all host devices directly attached to the virtual machine in order to add devices from a different host, you can instead add the devices from the desired host, which will automatically remove all of the devices already attached to the virtual machine.

Procedure 6.16. Removing a Host Device from a Virtual Machine

1. Select the virtual machine and click the **Host Devices** tab in the details pane to list the host devices attached to the virtual machine.

2. Select the host device to detach from the virtual machine, or hold **Ctrl** to select multiple devices, and click **Remove device** to open the **Remove Host Device(s)** window.

3. Click **OK** to confirm and detach these devices from the virtual machine.

6.10.3. Pinning a Virtual Machine to Another Host

You can use the **Host Devices** tab in the details pane of a virtual machine to pin it to a specific host.

If the virtual machine has any host devices attached to it, pinning it to another host will automatically remove the host devices from the virtual machine.

Procedure 6.17. Pinning a Virtual Machine to a Host

1. Select a virtual machine and click the **Host Devices** tab in the details pane.

2. Click **Pin to another host** to open the **Pin VM to Host** window.

3. Use the **Host** drop-down menu to select a host.

4. Click **OK** to pin the virtual machine to the selected host.

6.11. Affinity Groups

Virtual machine affinity allows you to define sets of rules that specify whether certain virtual machines run together on the same host or run separately on different hosts. This allows you to create advanced workload scenarios for addressing challenges such as strict licensing requirements and workloads demanding high availability.

Virtual machine affinity is applied to virtual machines by adding virtual machines to one or more affinity groups. An affinity group is a group of two or more virtual machines for which a set of identical parameters and conditions apply. These parameters include positive (run together) affinity that ensures the virtual machines in an affinity group run on the same host, and negative (run independently) affinity that ensures the virtual machines in an affinity group run on different hosts.

A further set of conditions can then be applied to these parameters. For example, you can apply hard enforcement, which is a condition that ensures the virtual machines in the affinity group run on the same host or different hosts regardless of external conditions, or soft enforcement, which is a condition that indicates a preference for virtual machines in an affinity group to run on the same host or different hosts when possible.

The combination of an affinity group, its parameters, and its conditions is known as an affinity policy. Affinity policies are applied to running virtual machines immediately, without having to restart.

> **Note**
>
> Affinity groups are applied to virtual machines on the cluster level. When a virtual machine is moved from one cluster to another, that virtual machine is removed from all affinity groups in the source cluster.

> **Important**
>
> Affinity groups will only take effect when the **VmAffinityGroups** filter module or weights module is enabled in the scheduling policy applied to clusters in which affinity groups are defined. The **VmAffinityGroups** filter module is used to implement hard enforcement, and the **VmAffinityGroups** weights module is used to implement soft enforcement.

6.11.1. Creating an Affinity Group

You can create new affinity groups in the Administration Portal.

Procedure 6.18. Creating Affinity Groups

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Affinity Groups** tab in the details pane.

3. Click **New**.

4. Enter a **Name** and **Description** for the affinity group.

5. Select the **Positive** check box to apply positive affinity, or ensure this check box is cleared to apply negative affinity.

6. Select the **Enforcing** check box to apply hard enforcement, or ensure this check box is cleared to apply soft enforcement.

7. Use the drop-down list to select the virtual machines to be added to the affinity group. Use the **+** and **-** buttons to add or remove additional virtual machines.

8. Click **OK**.

6.11.2. Editing an Affinity Group

Procedure 6.19. Editing Affinity Groups

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Affinity Groups** tab in the details pane.

3. Click **Edit**.

4. Change the **Positive** and **Enforcing** check boxes to the preferred values and use the **+** and **-** buttons to add or remove virtual machines to or from the affinity group.

5. Click **OK**.

6.11.3. Removing an Affinity Group

Procedure 6.20. Removing Affinity Groups

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click the **Affinity Groups** tab in the details pane.

3. Click **Remove**.

4. Click **OK**.

The affinity policy that applied to the virtual machines that were members of that affinity group no longer applies.

6.12. Exporting and Importing Virtual Machines and Templates

> **Note**
>
> The export storage domain is deprecated. Storage data domains can be unattached from a data center and imported to another data center in the same environment, or in a different environment. Virtual machines, floating virtual disk images, and templates can then be uploaded from the imported storage domain to the attached data center. See the Importing Existing Storage Domains section in the *Red Hat Virtualization Administration Guide* for information on importing storage domains.

Virtual machines and templates stored in Open Virtual Machine Format (OVF) can be exported from and imported to data centers in the same or different Red Hat Virtualization environment.

To export or import virtual machines and templates, an active export domain must be attached to the data center containing the virtual machine or template to be exported or imported. An export domain acts as a temporary storage area containing two directories for each exported virtual machine or template. One directory contains the OVF files for the virtual machine or template. The other directory holds the disk image or images for the virtual machine or template.

There are three stages to exporting and importing virtual machines and templates:

1. Export the virtual machine or template to an export domain.

2. Detach the export domain from one data center, and attach it to another. You can attach it to a different data center in the same Red Hat Virtualization environment, or attach it to a data center in a separate Red Hat Virtualization environment that is managed by another installation of the Red Hat Virtualization Manager.

> **Note**
>
> An export domain can only be active in one data center at a given time. This means that the export domain must be attached to either the source data center or the destination data center.

3. Import the virtual machine or template into the data center to which the export domain is attached.

When you export or import a virtual machine or template, properties including basic details such as the name and description, resource allocation, and high availability settings of that virtual machine or template are preserved. Specific user roles and permissions, however, are not preserved during the export process. If certain user roles and permissions are required to access the virtual machine or template, they will need to be set again after the virtual machine or template is imported.

You can also use the V2V feature to import virtual machines from other virtualization providers, such as Xen or VMware, or import Windows virtual machines. V2V converts virtual machines so that they can be hosted by Red Hat Virtualization. For more information on installing and using V2V, see Converting Virtual Machines from Other Hypervisors to KVM with virt-v2v

> ⭐ **Important**
>
> Virtual machines must be shut down before being exported or imported.

6.12.1. Graphical Overview for Exporting and Importing Virtual Machines and Templates

This procedure provides a graphical overview of the steps required to export a virtual machine or template from one data center and import that virtual machine or template into another data center.

Procedure 6.21. Exporting and Importing Virtual Machines and Templates

1. Attach the export domain to the source data center.
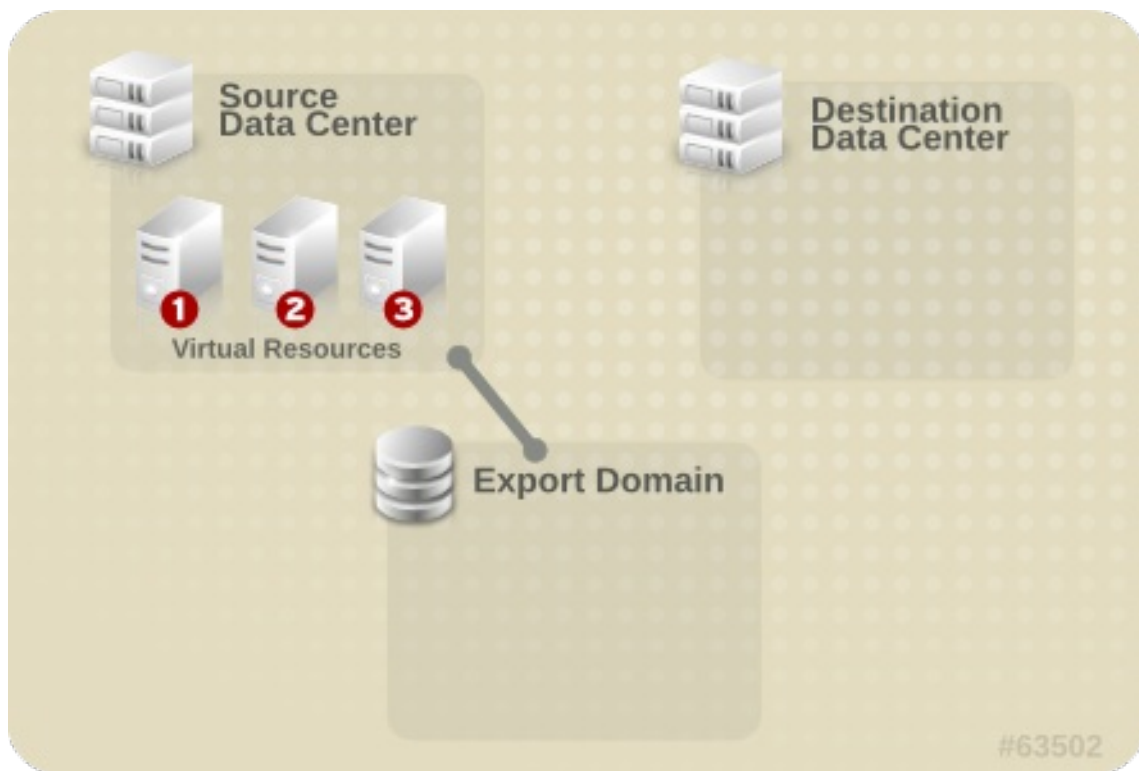
   Figure 6.6. Attach Export Domain

2. Export the virtual machine or template to the export domain.
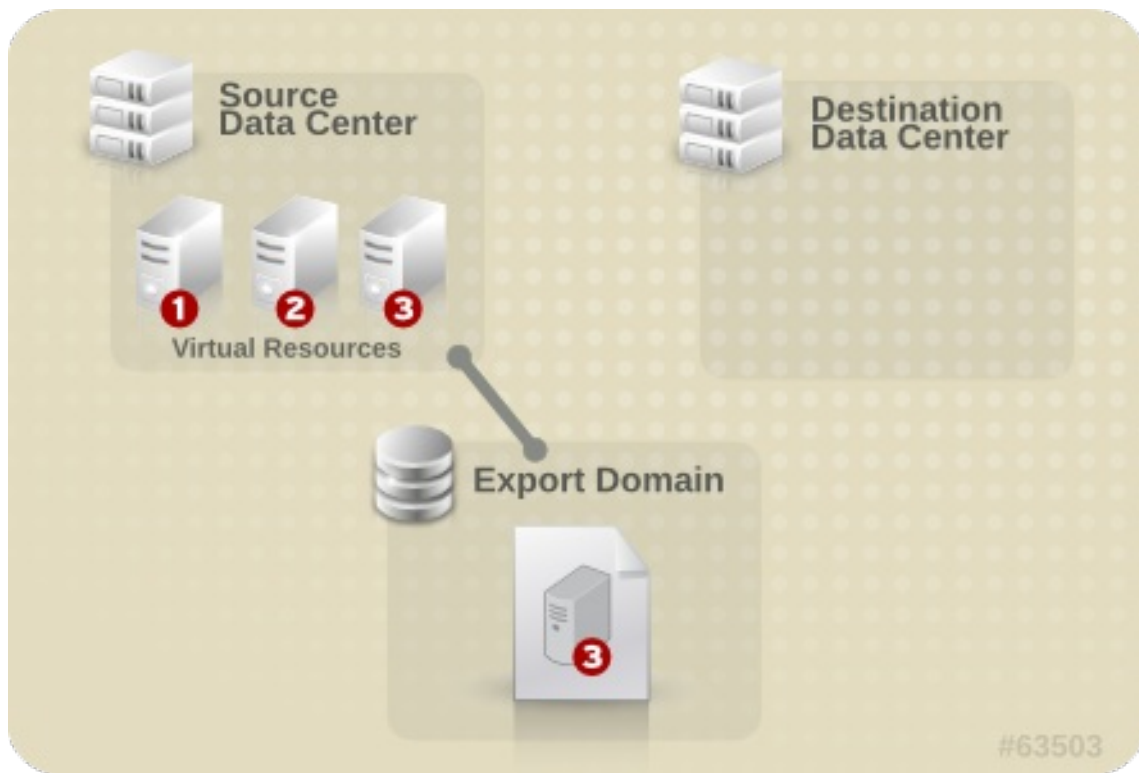
Figure 6.7. Export the Virtual Resource

3. Detach the export domain from the source data center.
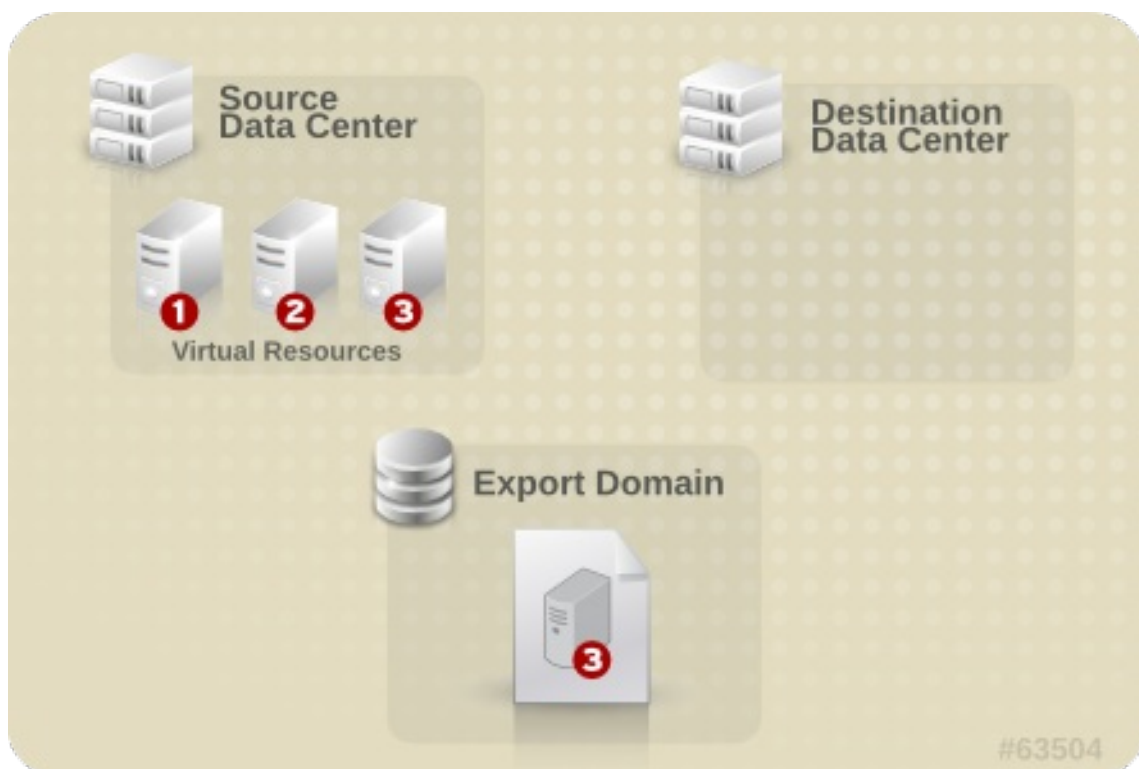
Figure 6.8. Detach Export Domain

4. Attach the export domain to the destination data center.
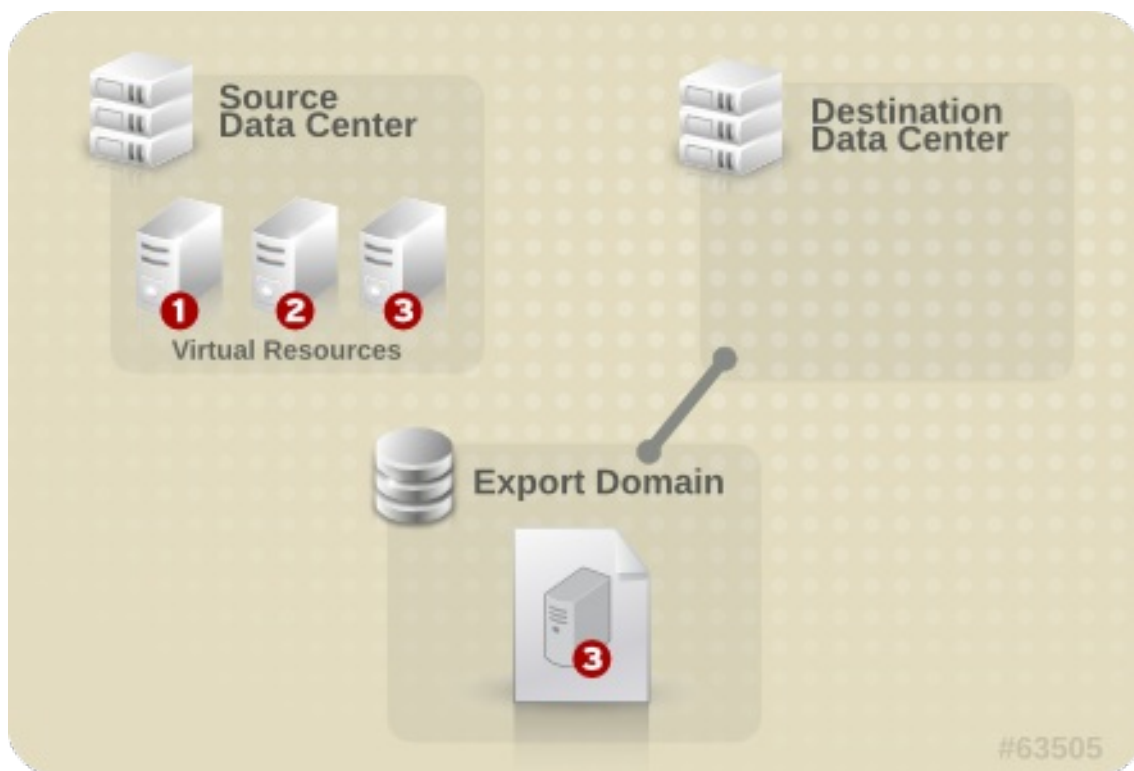
Figure 6.9. Attach the Export Domain

5. Import the virtual machine or template into the destination data center.
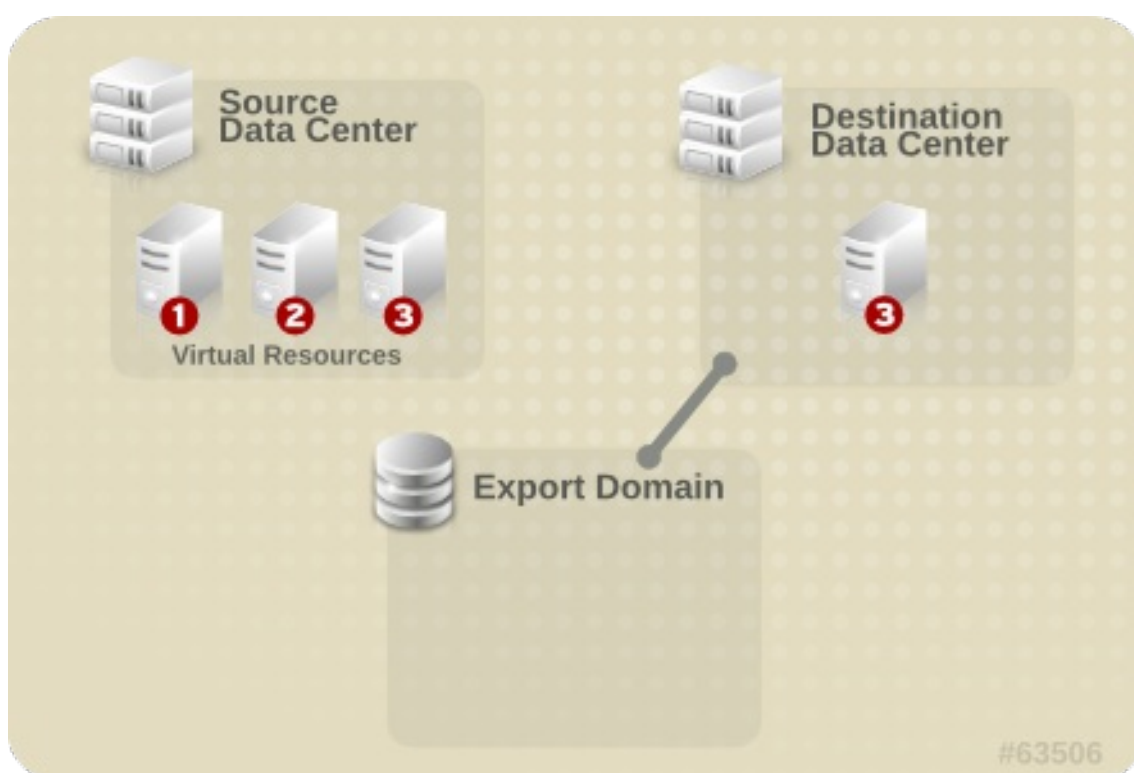
Figure 6.10. Import the virtual resource

6.12.2. Exporting a Virtual Machine to the Export Domain

Export a virtual machine to the export domain so that it can be imported into a different data center. Before you begin, the export domain must be attached to the data center that contains the virtual machine to be exported.

> ⚠️ **Warning**
>
> The virtual machine must be shut down before being exported.

Procedure 6.22. Exporting a Virtual Machine to the Export Domain

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Export**.

3. Optionally select the following check boxes:

   - **Force Override**: overrides existing images of the virtual machine on the export domain.

   - **Collapse Snapshots**: creates a single export volume per disk. This option removes snapshot restore points and includes the template in a template-based virtual machine, and removes any dependencies a virtual machine has on a template. For a virtual machine that is dependent on a template, either select this option, export the template with the virtual machine, or make sure the template exists in the destination data center.

   > 📝 **Note**
   >
   > When you create a virtual machine from a template, two storage allocation options are available under **New Virtual Machine** → **Resource Allocation** → **Storage Allocation**.
   >
   > - If **Clone** was selected, the virtual machine is not dependent on the template. The template does not have to exist in the destination data center.
   > - If **Thin** was selected, the virtual machine is dependent on the template, so the template must exist in the destination data center or be exported with the virtual machine. Alternatively, select the **Collapse Snapshots** check box to collapse the template disk and virtual machine disk into a single disk.
   >
   > To check which option was selected, select a virtual machine and click the **General** tab in the details pane.

4. Click **OK**.

The export of the virtual machine begins. The virtual machine displays in the **Virtual Machines** results list with an **Image Locked** status while it is exported. Depending on the size of your virtual machine hard disk images, and your storage hardware, this can take up to an hour. Use the **Events** tab to view the progress. When complete, the virtual machine has been exported to the export domain and displays on the **VM Import** tab of the export domain's details pane.

6.12.3. Importing a Virtual Machine into the Destination Data Center

You have a virtual machine on an export domain. Before the virtual machine can be imported to a new data center, the export domain must be attached to the destination data center.

Procedure 6.23. Importing a Virtual Machine into the Destination Data Center

1. Click the **Storage** tab, and select the export domain in the results list. The export domain must have a status of **Active**.

2. Select the **VM Import** tab in the details pane to list the available virtual machines to import.

3. Select one or more virtual machines to import and click **Import**.
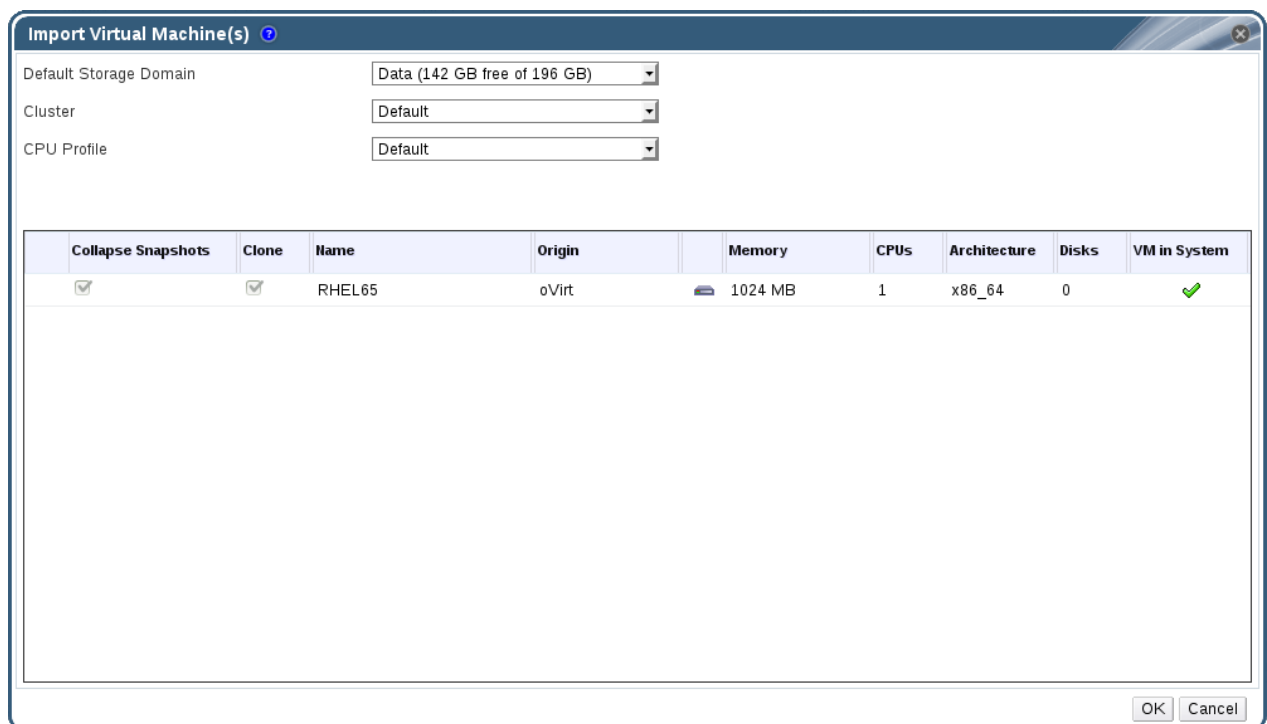
Figure 6.11. Import Virtual Machine

4. Select the **Default Storage Domain** and **Cluster**.

5. Select the **Collapse Snapshots** check box to remove snapshot restore points and include templates in template-based virtual machines.

6. Click the virtual machine to be imported and click on the **Disks** sub-tab. From this tab, you can use the **Allocation Policy** and **Storage Domain** drop-down lists to select whether the disk used by the virtual machine will be thinly provisioned or preallocated, and can also select the storage domain on which the disk will be stored. An icon is also displayed to indicate which of the disks to be imported acts as the boot disk for that virtual machine.

7. Click **OK** to import the virtual machines.

   The **Import Virtual Machine Conflict** window opens if the virtual machine exists in the virtualized environment.
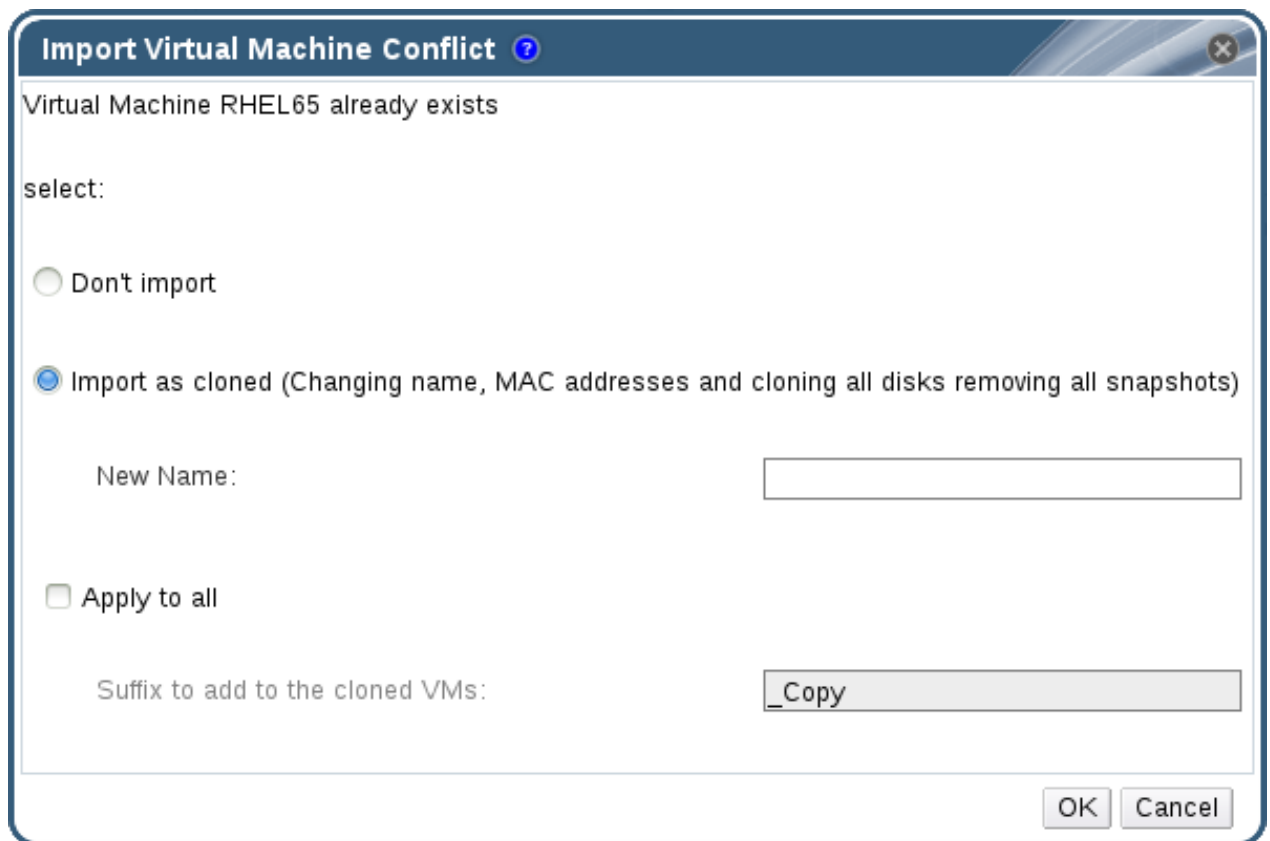
Figure 6.12. Import Virtual Machine Conflict Window

8. Choose one of the following radio buttons:

   » **Don't import**

   » **Import as cloned** and enter a unique name for the virtual machine in the **New Name** field.

9. Optionally select the **Apply to all** check box to import all duplicated virtual machines with the same suffix, and then enter a suffix in the **Suffix to add to the cloned VMs** field.

10. Click **OK**.

> ⭐ **Important**
>
> During a single import operation, you can only import virtual machines that share the same architecture. If any of the virtual machines to be imported have a different architecture to that of the other virtual machines to be imported, a warning will display and you will be prompted to change your selection so that only virtual machines with the same architecture will be imported.

6.12.4. Importing a Virtual Machine from a VMware Provider

Import virtual machines from a VMware vCenter provider to your Red Hat Virtualization environment. You can import from a VMware provider by entering its details in the **Import Virtual Machine(s)** window during each import operation, or you can add the VMware provider as an external provider, and select the preconfigured provider during import

operations. To add an external provider, see Adding a VMware Instance as a Virtual Machine Provider.

Red Hat Virtualization uses V2V to convert VMware virtual machines to the correct format before they are imported. You must install the *virt-v2v* package on a least one Red Hat Enterprise Linux 7 host before proceeding. This package is available in the base **rhel-7-server-rpms** repository.

> **Warning**
>
> The virtual machine must be shut down before being imported. Starting the virtual machine through VMware during the import process can result in data corruption.

Procedure 6.24. Importing a Virtual Machine from VMware

1. In the **Virtual Machines** tab, click **Import** to open the **Import Virtual Machine(s)** window.

Figure 6.13. The Import Virtual Machine(s) Window

2. Select **VMware** from the **Source** list.

3. If you have configured a VMware provider as an external provider, select it from the **External Provider** list. Verify that the provider credentials are correct. If you did not specify a destination data center or proxy host when configuring the external provider, select those options now.

4. If you have not configured a VMware provider, or want to import from a new VMware provider, provide the following details:

   a. Select from the list the **Data Center** in which the virtual machine will be available.

   b. Enter the IP address or fully qualified domain name of the VMware vCenter instance in the **vCenter** field.

   c. Enter the IP address or fully qualified domain name of the host from which the virtual machines will be imported in the **ESXi** field.

   d. Enter the name of the data center and the cluster in which the specified ESXi host resides in the **Data Center** field.

   e. If you have exchanged the SSL certificate between the ESXi host and the Manager, leave **Verify server's SSL certificate** checked to verify the ESXi host's certificate. If not, uncheck the option.

   f. Enter the **Username** and **Password** for the VMware vCenter instance. The user must have access to the VMware data center and ESXi host on which the virtual machines reside.

   g. Select a host in the chosen data center with *virt-v2v* installed to serve as the **Proxy Host** during virtual machine import operations. This host must also be able to connect to the network of the VMware vCenter external provider.

5. Click **Load** to generate a list of the virtual machines on the VMware provider.

6. Select one or more virtual machines from the **Virtual Machines on Source** list, and use the arrows to move them to the **Virtual Machines to Import** list. Click **Next**.

> **Important**
>
> An import operation can only include virtual machines that share the same architecture. If any virtual machine to be imported has a different architecture, a warning will display and you will be prompted to change your selection to include only virtual machines with the same architecture.

> **Note**
>
> If a virtual machine's network device uses the driver type e1000 or rtl8139, the virtual machine will use the same driver type after it has been imported to Red Hat Virtualization.
>
> If required, you can change the driver type to VirtIO manually after the import. To change the driver type after a virtual machine has been imported, see Section 5.2.2, "Editing a Network Interface". If the network device uses driver types other than e1000 or rtl8139, the driver type is changed to VirtIO automatically during the import. The **Attach VirtIO-drivers** option allows the VirtIO drivers to be injected to the imported virtual machine files so that when the driver is changed to VirtIO, the device will be properly detected by the operating system.

Figure 6.14. The Import Virtual Machine(s) Window

7. Select the **Cluster** in which the virtual machines will reside.

8. Select a **CPU Profile** for the virtual machines.

9. Select the **Collapse Snapshots** check box to remove snapshot restore points and include templates in template-based virtual machines.

10. Select the **Clone** check box to change the virtual machine name and MAC addresses, and clone all disks, removing all snapshots. If a virtual machine appears with a warning symbol beside its name or has a tick in the **VM in System** column, you must clone the virtual machine and change its name.

11. Click on each virtual machine to be imported and click on the **Disks** sub-tab. Use the **Allocation Policy** and **Storage Domain** lists to select whether the disk used by the virtual machine will be thinly provisioned or preallocated, and select the storage domain

on which the disk will be stored. An icon is also displayed to indicate which of the disks to be imported acts as the boot disk for that virtual machine.

> **Note**
>
> The target storage domain must be a filed-based domain. Due to current limitations, specifying a block-based domain causes the V2V operation to fail.

12. If you selected the **Clone** check box, change the name of the virtual machine in the **General** sub-tab.

13. Click **OK** to import the virtual machines.

6.12.5. Importing a Virtual Machine from a Xen Host

Import virtual machines from Xen on Red Hat Enterprise Linux 5 to your Red Hat Virtualization environment. Red Hat Virtualization uses V2V to convert Xen virtual machines to the correct format before they are imported. You must install the *virt-v2v* package on at least one Red Hat Enterprise Linux 7 host in the destination data center before proceeding (this host is referred to in the following procedure as the V2V host). The *virt-v2v* package is available in the base **rhel-7-server-rpms** repository.

> **Warning**
>
> The virtual machine must be shut down before being imported. Starting the virtual machine through Xen during the import process can result in data corruption.

Procedure 6.25. Importing a Virtual Machine from Xen

1. Enable public key authentication between the V2V host and the Xen host:

   a. Log in to the V2V host and generate SSH keys for the **vdsm** user.

   ```
   # sudo -u vdsm ssh-keygen
   ```

   b. Copy the **vdsm** user's public key to the Xen host.

   ```
   # sudo -u vdsm ssh-copy-id root@xenhost.example.com
   ```

   c. Log in to the Xen host to add it to the V2V host's **known_hosts** file.

   ```
   # sudo -u vdsm ssh root@xenhost.example.com
   ```

2. Log in to the Administration Portal. In the **Virtual Machines** tab, click **Import** to open the **Import Virtual Machine(s)** window.

Figure 6.15. The Import Virtual Machine(s) Window

3. Select the **Data Center** that contains the V2V host.

4. Select **XEN (via RHEL)** from the **Source** drop-down list.

5. Enter the **URI** of the Xen host. The required format is pre-filled; you must replace **<hostname>** with the host name of the Xen host.

6. Select the V2V host from the **Proxy Host** drop-down list.

7. Click **Load** to generate a list of the virtual machines on the Xen host.

8. Select one or more virtual machines from the **Virtual Machines on Source** list, and use the arrows to move them to the **Virtual Machines to Import** list.

> **Note**
>
> Due to current limitations, Xen virtual machines with block devices do not appear in the **Virtual Machines on Source** list, and cannot be imported to Red Hat Virtualization.

9. Click **Next**.

> **Important**
>
> An import operation can only include virtual machines that share the same architecture. If any virtual machine to be imported has a different architecture, a warning will display and you will be prompted to change your selection to include only virtual machines with the same architecture.
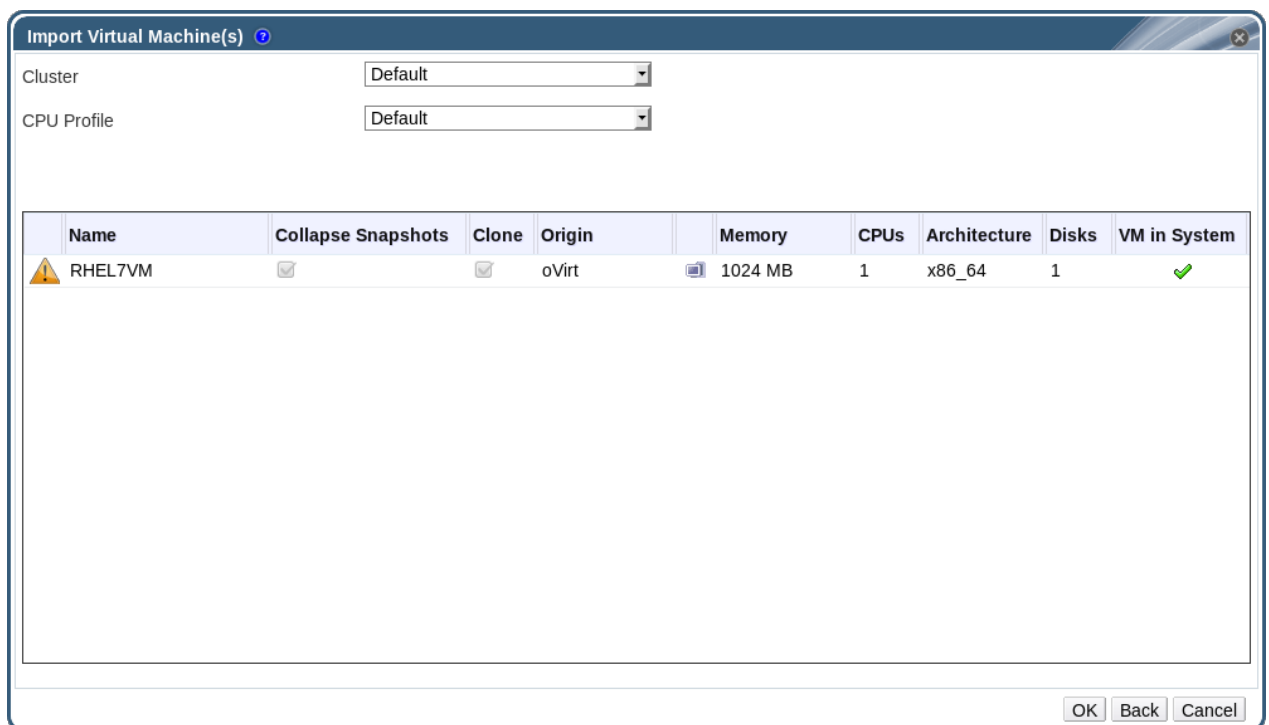

Figure 6.16. The Import Virtual Machine(s) Window

10. Select the **Cluster** in which the virtual machines will reside.

11. Select a **CPU Profile** for the virtual machines.

12. Select the **Collapse Snapshots** check box to remove snapshot restore points and include templates in template-based virtual machines.

13. Select the **Clone** check box to change the virtual machine name and MAC addresses, and clone all disks, removing all snapshots. If a virtual machine appears with a warning symbol beside its name or has a tick in the **VM in System** column, you must clone the virtual machine and change its name.

14. Click on each virtual machine to be imported and click on the **Disks** sub-tab. Use the **Allocation Policy** and **Storage Domain** lists to select whether the disk used by the virtual machine will be thinly provisioned or preallocated, and select the storage domain on which the disk will be stored. An icon is also displayed to indicate which of the disks to be imported acts as the boot disk for that virtual machine.

> **Note**
>
> The target storage domain must be a filed-based domain. Due to current limitations, specifying a block-based domain causes the V2V operation to fail.

15. If you selected the **Clone** check box, change the name of the virtual machine in the **General** sub-tab.

16. Click **OK** to import the virtual machines.

6.12.6. Importing a Virtual Machine from a KVM Host

Import virtual machines from KVM to your Red Hat Virtualization environment. Red Hat Virtualization converts KVM virtual machines to the correct format before they are imported. You must enable public key authentication between the KVM host and at least one host in the destination data center (this host is referred to in the following procedure as the proxy host).

> **Warning**
>
> The virtual machine must be shut down before being imported. Starting the virtual machine through KVM during the import process can result in data corruption.

Procedure 6.26. Importing a Virtual Machine from KVM

1. Enable public key authentication between the proxy host and the KVM host:

    a. Log in to the proxy host and generate SSH keys for the **vdsm** user.

    ```
    # sudo -u vdsm ssh-keygen
    ```

    b. Copy the **vdsm** user's public key to the KVM host.

    ```
    # sudo -u vdsm ssh-copy-id root@kvmhost.example.com
    ```

    c. Log in to the KVM host to add it to the proxy host's **known_hosts** file.

    ```
    # sudo -u vdsm ssh root@kvmhost.example.com
    ```

2. Log in to the Administration Portal. In the **Virtual Machines** tab, click **Import** to open the **Import Virtual Machine(s)** window.

Figure 6.17. The Import Virtual Machine(s) Window

3. Select the **Data Center** that contains the proxy host.

4. Select **KVM (via Libvirt)** from the **Source** drop-down list.

5. Enter the **URI** of the KVM host in the following format:

   ```
   qemu+ssh://root@kvmhost.example.com
   ```

6. Keep the **Requires Authentication** check box selected.

7. Enter **root** in the **Username** field.

8. Enter the **Password** of the KVM host's root user.

9. Select the **Proxy Host** from the drop-down list.

10. Click **Load** to generate a list of the virtual machines on the KVM host.

11. Select one or more virtual machines from the **Virtual Machines on Source** list, and use the arrows to move them to the **Virtual Machines to Import** list.

12. Click **Next**.

> **Important**
>
> An import operation can only include virtual machines that share the same architecture. If any virtual machine to be imported has a different architecture, a warning will display and you will be prompted to change your selection to include only virtual machines with the same architecture.
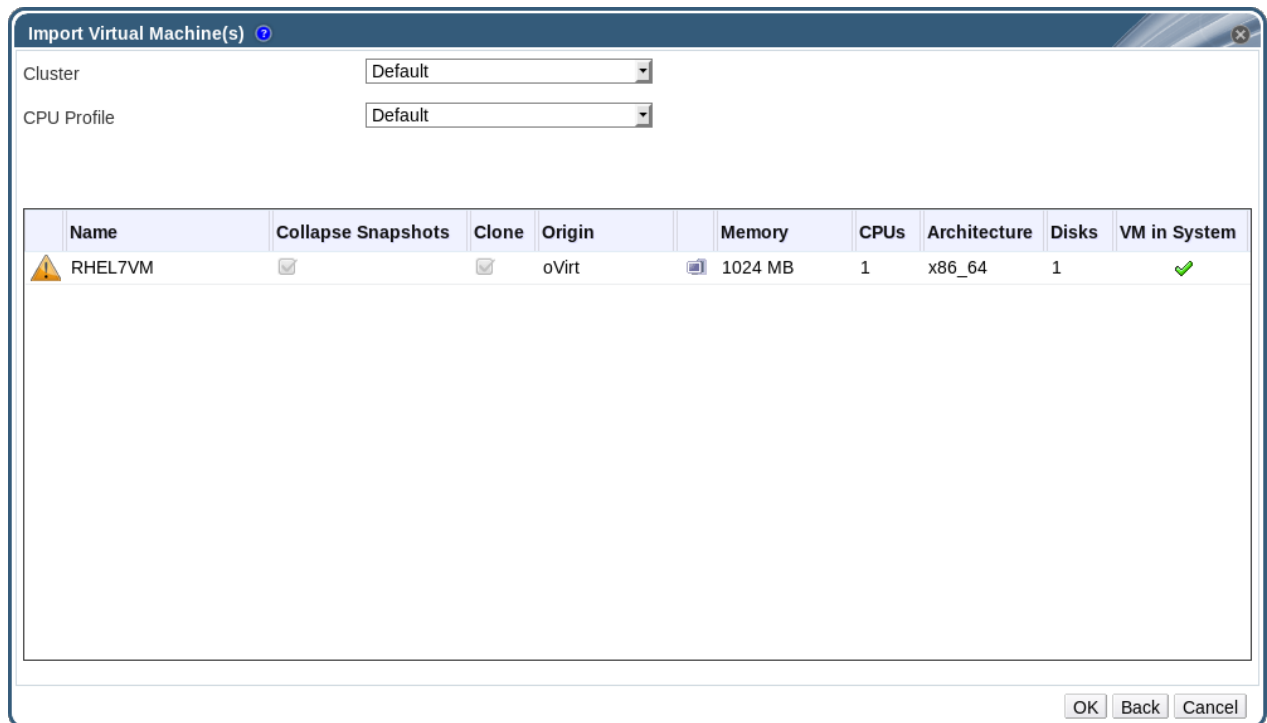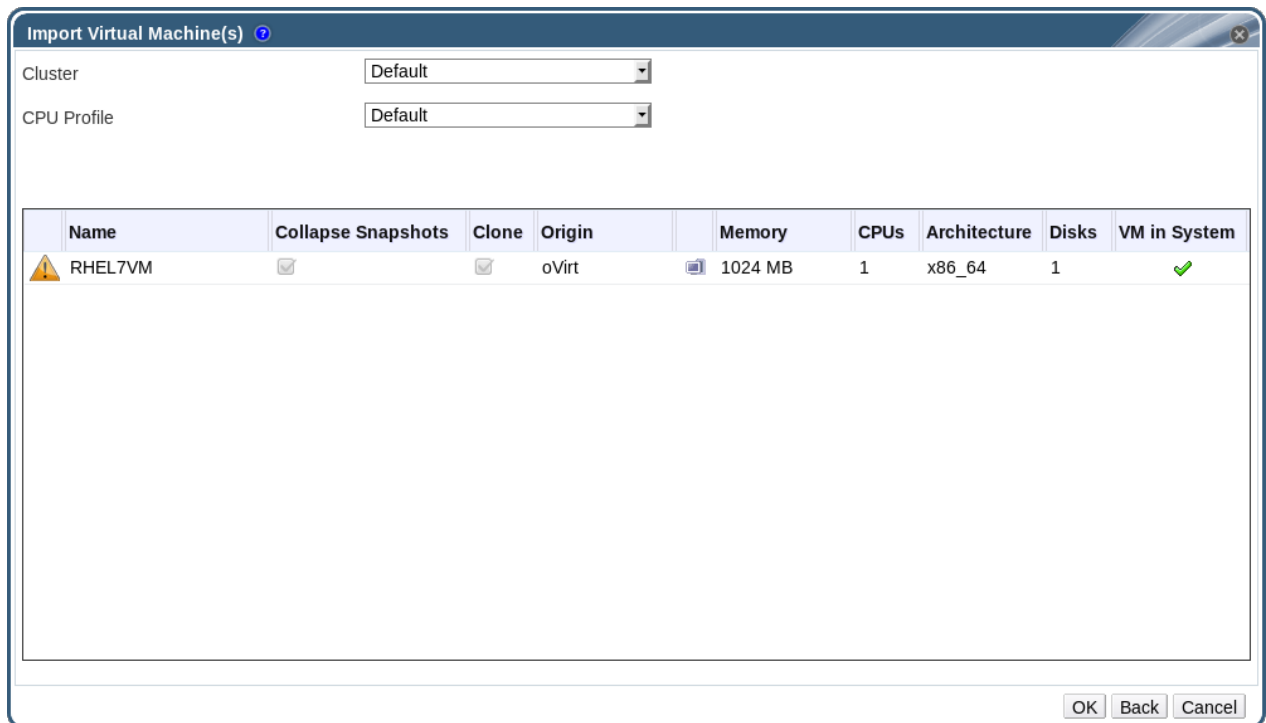

Figure 6.18. The Import Virtual Machine(s) Window

13. Select the **Cluster** in which the virtual machines will reside.

14. Select a **CPU Profile** for the virtual machines.

15. Select the **Collapse Snapshots** check box to remove snapshot restore points and include templates in template-based virtual machines.

16. Select the **Clone** check box to change the virtual machine name and MAC addresses, and clone all disks, removing all snapshots. If a virtual machine appears with a warning symbol beside its name or has a tick in the **VM in System** column, you must clone the virtual machine and change its name.

17. Click on each virtual machine to be imported and click on the **Disks** sub-tab. Use the **Allocation Policy** and **Storage Domain** lists to select whether the disk used by the virtual machine will be thinly provisioned or preallocated, and select the storage domain on which the disk will be stored. An icon is also displayed to indicate which of the disks to be imported acts as the boot disk for that virtual machine.

> **Note**
>
> The target storage domain must be a filed-based domain. Due to current limitations, specifying a block-based domain causes the operation to fail.

18. If you selected the **Clone** check box, change the name of the virtual machine in the **General** sub-tab.

19. Click **OK** to import the virtual machines.

6.13. Migrating Virtual Machines Between Hosts

Live migration provides the ability to move a running virtual machine between physical hosts with no interruption to service. The virtual machine remains powered on and user applications continue to run while the virtual machine is relocated to a new physical host. In the background, the virtual machine's RAM is copied from the source host to the destination host. Storage and network connectivity are not altered.

6.13.1. Live Migration Prerequisites

Live migration is used to seamlessly move virtual machines to support a number of common maintenance tasks. Ensure that your Red Hat Virtualization environment is correctly configured to support live migration well in advance of using it.

At a minimum, for successful live migration of virtual machines to be possible:

» The source and destination host should both be members of the same cluster, ensuring CPU compatibility between them.

> **Note**
>
> Live migrating virtual machines between different clusters is generally not recommended. The currently only supported use case is documented at https://access.redhat.com/articles/1390733.

» The source and destination host must have a status of **Up**.

» The source and destination host must have access to the same virtual networks and VLANs.

» The source and destination host must have access to the data storage domain on which the virtual machine resides.

» There must be enough CPU capacity on the destination host to support the virtual machine's requirements.

» There must be enough RAM on the destination host that is not in use to support the virtual machine's requirements.

» The migrating virtual machine must not have the **cache!=none** custom property set.

In addition, for best performance, the storage and management networks should be split to avoid network saturation. Virtual machine migration involves transferring large amounts of data between hosts.

Live migration is performed using the management network. Each live migration event is limited to a maximum transfer speed of 30 MBps, and the number of concurrent migrations supported is also limited by default. Despite these measures, concurrent migrations have the potential to saturate the management network. It is recommended that separate logical networks are created for storage, display, and virtual machine data to minimize the risk of network saturation.

6.13.2. Optimizing Live Migration

Live virtual machine migration can be a resource-intensive operation. The following two options can be set globally for every virtual machine in the environment, at the cluster level, or at the individual virtual machine level to optimize live migration.

The **Auto Converge migrations** option allows you to set whether auto-convergence is used during live migration of virtual machines. Large virtual machines with high workloads can dirty memory more quickly than the transfer rate achieved during live migration, and prevent the migration from converging. Auto-convergence capabilities in QEMU allow you to force convergence of virtual machine migrations. QEMU automatically detects a lack of convergence and triggers a throttle-down of the vCPUs on the virtual machine.

The **Enable migration compression** option allows you to set whether migration compression is used during live migration of the virtual machine. This feature uses Xor Binary Zero Run-Length-Encoding to reduce virtual machine downtime and total live migration time for virtual machines running memory write-intensive workloads or for any application with a sparse memory update pattern.

Both options are disabled globally by default.

Procedure 6.27. Configuring Auto-convergence and Migration Compression for Virtual Machine Migration

1. Configure the optimization settings at the global level:

   a. Enable auto-convergence at the global level:

   ```
   # engine-config -s DefaultAutoConvergence=True
   ```

   b. Enable migration compression at the global level:

   ```
   # engine-config -s DefaultMigrationCompression=True
   ```

   c. Restart the **ovirt-engine** service to apply the changes:

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

2. Configure the optimization settings at the cluster level:

   a. Select a cluster.

   b. Click **Edit**.

   c. Click the **Scheduling Policy** tab.

   d. From the **Auto Converge migrations** list, select **Inherit from global setting**, **Auto Converge**, or **Don't Auto Converge**.

   e. From the **Enable migration compression** list, select **Inherit from global**

setting, **Compress**, or **Don't Compress**.

3. Configure the optimization settings at the virtual machine level:

   a. Select a virtual machine.

   b. Click **Edit**.

   c. Click the **Host** tab.

   d. From the **Auto Converge migrations** list, select **Inherit from cluster setting**, **Auto Converge**, or **Don't Auto Converge**.

   e. From the **Enable migration compression** list, select **Inherit from cluster setting**, **Compress**, or **Don't Compress**.

6.13.3. Automatic Virtual Machine Migration

Red Hat Virtualization Manager automatically initiates live migration of all virtual machines running on a host when the host is moved into maintenance mode. The destination host for each virtual machine is assessed as the virtual machine is migrated, in order to spread the load across the cluster.

The Manager automatically initiates live migration of virtual machines in order to maintain load balancing or power saving levels in line with scheduling policy. While no scheduling policy is defined by default, it is recommended that you specify the scheduling policy which best suits the needs of your environment. You can also disable automatic, or even manual, live migration of specific virtual machines where required.

6.13.4. Preventing Automatic Migration of a Virtual Machine

Red Hat Virtualization Manager allows you to disable automatic migration of virtual machines. You can also disable manual migration of virtual machines by setting the virtual machine to run only on a specific host.

The ability to disable automatic migration and require a virtual machine to run on a particular host is useful when using application high availability products, such as Red Hat High Availability or Cluster Suite.

Procedure 6.28. Preventing Automatic Migration of Virtual Machine

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.
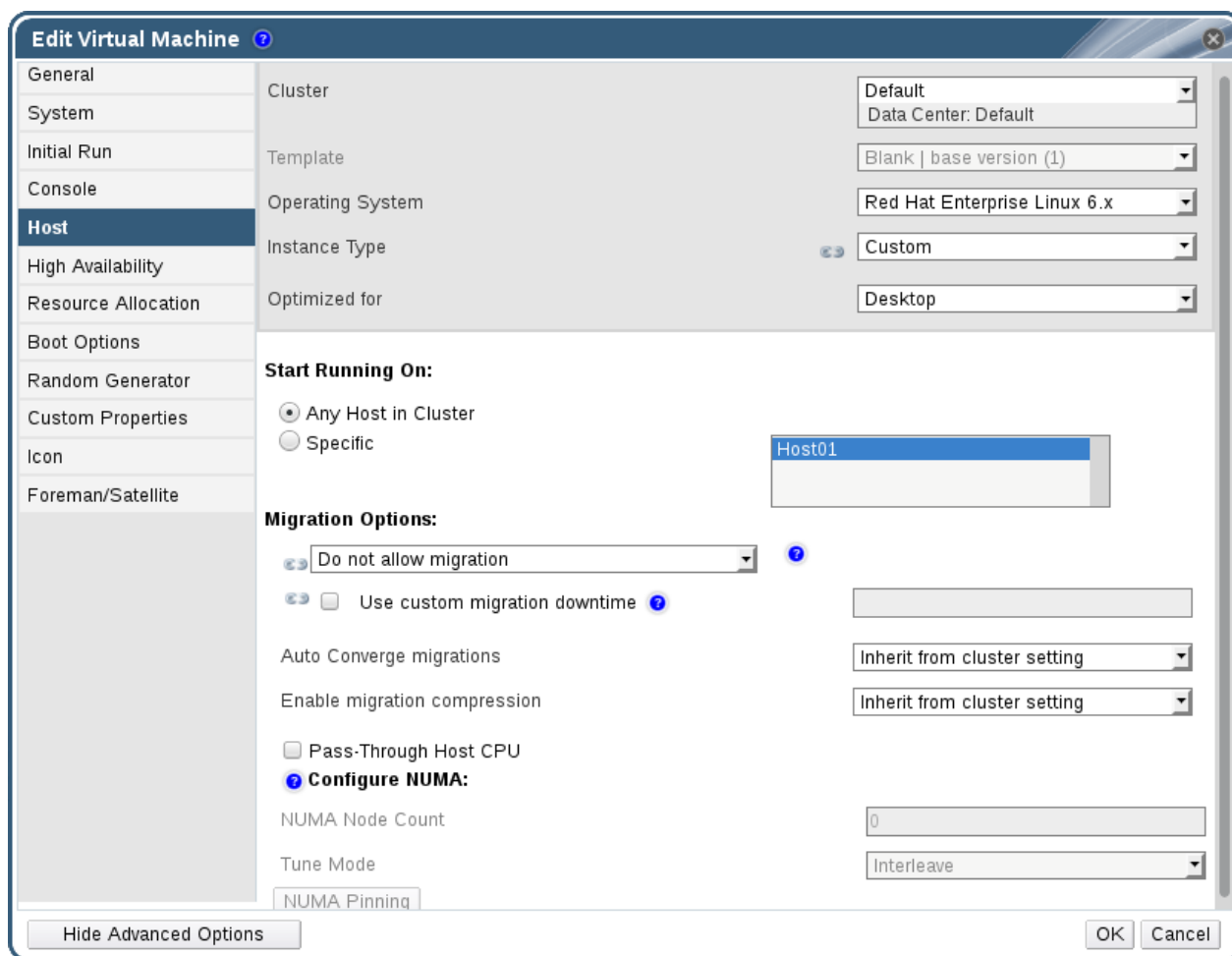
Figure 6.19. The Edit Virtual Machine Window

3. Click the **Host** tab.

4. Use the **Start Running On** radio buttons to designate the virtual machine to run on **Any Host in Cluster** or a **Specific** host. If applicable, select a specific host or group of hosts from the list.

> ⚠️ **Warning**
>
> Explicitly assigning a virtual machine to one specific host and disabling migration is mutually exclusive with Red Hat Virtualization high availability. Virtual machines that are assigned to one specific host can only be made highly available using third party high availability products like Red Hat High Availability. This restriction does not apply to virtual machines that are assigned to multiple specific hosts.

> ⭐ **Important**
>
> If the virtual machine has host devices directly attached to it, and a different host is specified, the host devices from the previous host will be automatically removed from the virtual machine.

5. Select **Allow manual migration only** or **Do not allow migration** from the **Migration Options** drop-down list.

6. Optionally, select the **Use custom migration downtime** check box and specify a value in milliseconds.

7. Click **OK**.

6.13.5. Manually Migrating Virtual Machines

A running virtual machine can be live migrated to any host within its designated host cluster. Live migration of virtual machines does not cause any service interruption. Migrating virtual machines to a different host is especially useful if the load on a particular host is too high. For live migration prerequisites, see Section 6.13.1, "Live Migration Prerequisites".

> **Note**
>
> When you place a host into maintenance mode, the virtual machines running on that host are automatically migrated to other hosts in the same cluster. You do not need to manually migrate these virtual machines.

> **Note**
>
> Live migrating virtual machines between different clusters is generally not recommended. The currently only supported use case is documented at https://access.redhat.com/articles/1390733.

Procedure 6.29. Manually Migrating Virtual Machines

1. Click the **Virtual Machines** tab and select a running virtual machine.

2. Click **Migrate**.

3. Use the radio buttons to select whether to **Select Host Automatically** or to **Select Destination Host**, specifying the host using the drop-down list.

   > **Note**
   >
   > When the **Select Host Automatically** option is selected, the system determines the host to which the virtual machine is migrated according to the load balancing and power management rules set up in the scheduling policy.

4. Click **OK**.

During migration, progress is shown in the **Migration** progress bar. Once migration is complete the **Host** column will update to display the host the virtual machine has been migrated to.

6.13.6. Setting Migration Priority

Red Hat Virtualization Manager queues concurrent requests for migration of virtual machines

off of a given host. The load balancing process runs every minute. Hosts already involved in a migration event are not included in the migration cycle until their migration event has completed. When there is a migration request in the queue and available hosts in the cluster to action it, a migration event is triggered in line with the load balancing policy for the cluster.

You can influence the ordering of the migration queue by setting the priority of each virtual machine; for example, setting mission critical virtual machines to migrate before others. Migrations will be ordered by priority; virtual machines with the highest priority will be migrated first.

Procedure 6.30. Setting Migration Priority

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Select the **High Availability** tab.

4. Select **Low**, **Medium**, or **High** from the **Priority** drop-down list.

5. Click **OK**.

6.13.7. Canceling Ongoing Virtual Machine Migrations

A virtual machine migration is taking longer than you expected. You'd like to be sure where all virtual machines are running before you make any changes to your environment.

Procedure 6.31. Canceling Ongoing Virtual Machine Migrations

1. Select the migrating virtual machine. It is displayed in the **Virtual Machines** resource tab with a status of **Migrating from**.

2. Click **Cancel Migration**.

The virtual machine status returns from **Migrating from** to **Up**.

6.13.8. Event and Log Notification upon Automatic Migration of Highly Available Virtual Servers

When a virtual server is automatically migrated because of the high availability function, the details of an automatic migration are documented in the **Events** tab and in the engine log to aid in troubleshooting, as illustrated in the following examples:

Example 6.1. Notification in the Events Tab of the Web Admin Portal

Highly Available *Virtual_Machine_Name* failed. It will be restarted automatically.

Virtual_Machine_Name was restarted on Host*Host_Name*

Example 6.2. Notification in the Manager engine.log

This log can be found on the Red Hat Virtualization Manager at **/var/log/ovirt-engine/engine.log**:

Failed to start Highly Available VM. Attempting to restart. VM Name: *Virtual_Machine_Name*, VM Id:*Virtual_Machine_ID_Number*

6.14. Improving Uptime with Virtual Machine High Availability

6.14.1. What is High Availability?

High availability means that a virtual machine will be automatically restarted if its process is interrupted. This happens if the virtual machine is terminated by methods other than powering off from within the guest or sending the shutdown command from the Manager. When these events occur, the highly available virtual machine is automatically restarted, either on its original host or another host in the cluster.

High availability is possible because the Red Hat Virtualization Manager constantly monitors the hosts and storage, and automatically detects hardware failure. If host failure is detected, any virtual machine configured to be highly available is automatically restarted on another host in the cluster.

With high availability, interruption to service is minimal because virtual machines are restarted within seconds with no user intervention required. High availability keeps your resources balanced by restarting guests on a host with low current resource utilization, or based on any workload balancing or power saving policies that you configure. This ensures that there is sufficient capacity to restart virtual machines at all times.

6.14.2. Why Use High Availability?

High availability is recommended for virtual machines running critical workloads.

High availability can ensure that virtual machines are restarted in the following scenarios:

 » When a host becomes non-operational due to hardware failure.

 » When a host is put into maintenance mode for scheduled downtime.

 » When a host becomes unavailable because it has lost communication with an external storage resource.

A high availability virtual machine is automatically restarted, either on its original host or another host in the cluster.

6.14.3. High Availability Considerations

A highly available host requires a power management device and its fencing parameters configured. In addition, for a virtual machine to be highly available when its host becomes non-operational, it needs to be started on another available host in the cluster. To enable the migration of highly available virtual machines:

 » Power management must be configured for the hosts running the highly available virtual machines.

 » The host running the highly available virtual machine must be part of a cluster which has other available hosts.

 » The destination host must be running.

 » The source and destination host must have access to the data domain on which the virtual machine resides.

» The source and destination host must have access to the same virtual networks and VLANs.

» There must be enough CPUs on the destination host that are not in use to support the virtual machine's requirements.

» There must be enough RAM on the destination host that is not in use to support the virtual machine's requirements.

6.14.4. Configuring a Highly Available Virtual Machine

High availability must be configured individually for each virtual machine.

Procedure 6.32. Configuring a Highly Available Virtual Machine

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **High Availability** tab.
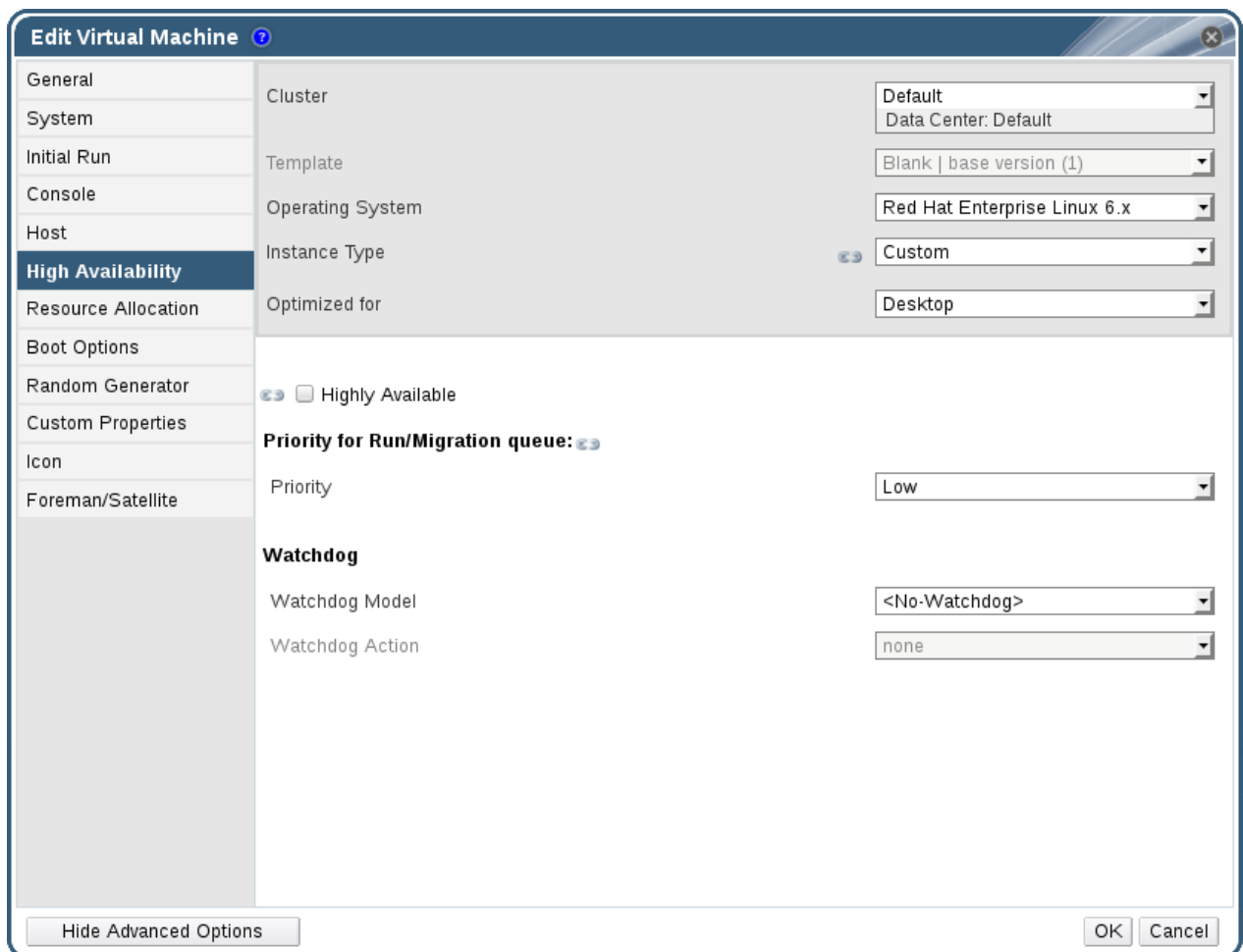
Figure 6.20. The High Availability Tab

4. Select the **Highly Available** check box to enable high availability for the virtual machine.

5. Select **Low**, **Medium**, or **High** from the **Priority** drop-down list. When migration is triggered, a queue is created in which the high priority virtual machines are migrated first. If a cluster is running low on resources, only the high priority virtual machines are

migrated.

6. Click **OK**.

6.15. Other Virtual Machine Tasks

6.15.1. Enabling SAP Monitoring

Enable SAP monitoring on a virtual machine through the Administration Portal.

Procedure 6.33. Enabling SAP Monitoring on Virtual Machines

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **Custom Properties** tab.
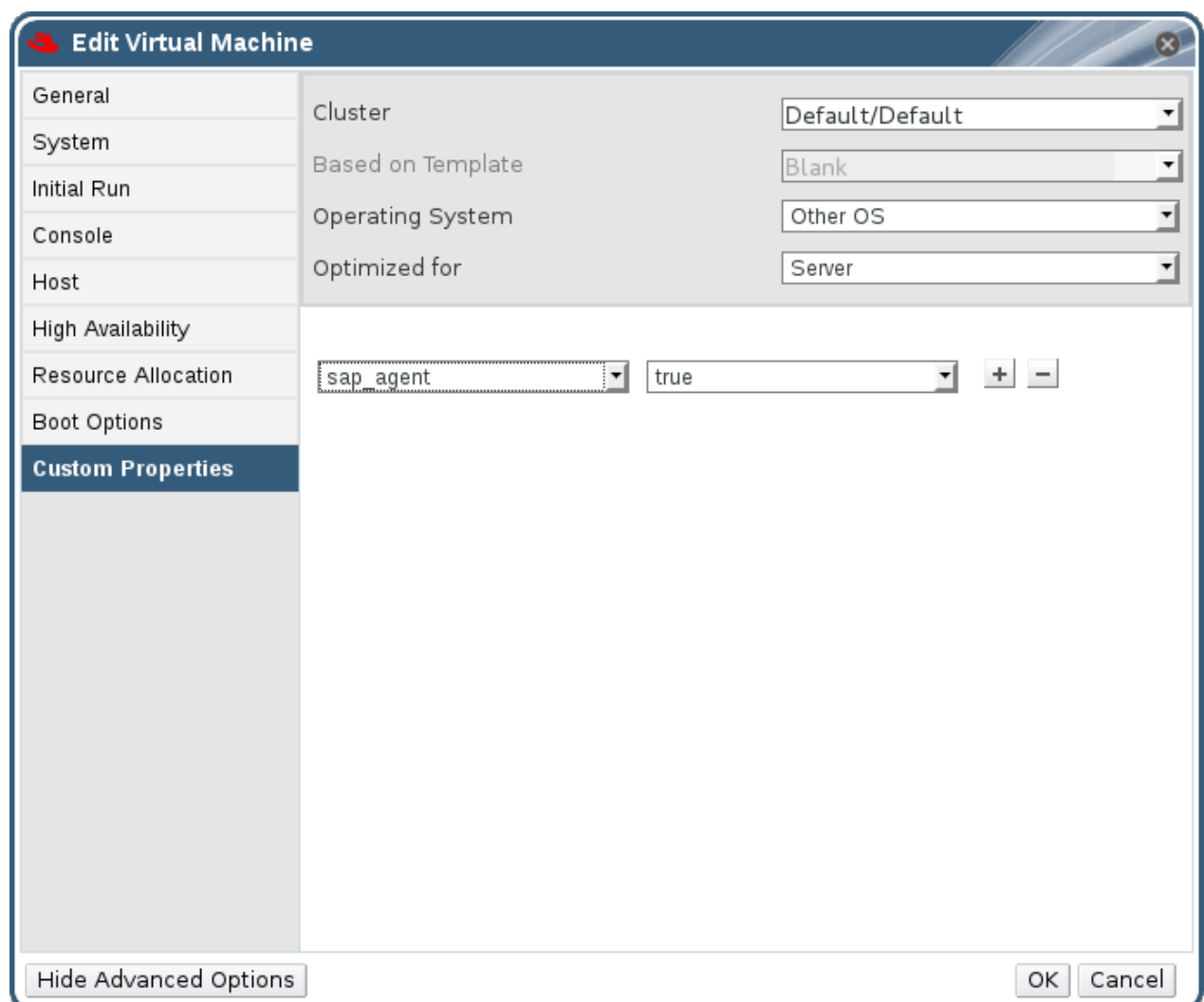
Figure 6.21. Enable SAP

4. Select **sap_agent** from the drop-down list. Ensure the secondary drop-down menu is set to **True**.

> If previous properties have been set, select the plus sign to add a new property rule and select **sap_agent**.

5. Click **OK**.

6.15.2. Configuring Red Hat Enterprise Linux 5.4 and Higher Virtual Machines to use SPICE

SPICE is a remote display protocol designed for virtual environments, which enables you to view a virtualized desktop or server. SPICE delivers a high quality user experience, keeps CPU consumption low, and supports high quality video streaming.

Using SPICE on a Linux machine significantly improves the movement of the mouse cursor on the console of the virtual machine. To use SPICE, the X-Windows system requires additional QXL drivers. The QXL drivers are provided with Red Hat Enterprise Linux 5.4 and newer. Older versions are not supported. Installing SPICE on a virtual machine running Red Hat Enterprise Linux significantly improves the performance of the graphical user interface.

> **Note**
>
> Typically, this is most useful for virtual machines where the user requires the use of the graphical user interface. System administrators who are creating virtual servers may prefer not to configure SPICE if their use of the graphical user interface is minimal.

6.15.2.1. Installing and Configuring QXL Drivers

You must manually install QXL drivers on virtual machines running Red Hat Enterprise Linux 5.4 or higher. This is unnecessary for virtual machines running Red Hat Enterprise Linux 6 or Red Hat Enterprise Linux 7 as the QXL drivers are installed by default.

Procedure 6.34. Installing QXL Drivers

1. Log in to a Red Hat Enterprise Linux virtual machine.

2. Install the QXL drivers:

   ```
   # yum install xorg-x11-drv-qxl
   ```

You can configure QXL drivers using either a graphical interface or the command line. Perform only one of the following procedures.

Procedure 6.35. Configuring QXL drivers in GNOME

1. Click **System**.

2. Click **Administration**.

3. Click **Display**.

4. Click the **Hardware** tab.

5. Click **Video Cards Configure**.

6. Select **qxl** and click **OK**.

7. Restart X-Windows by logging out of the virtual machine and logging back in.

Procedure 6.36. Configuring QXL drivers on the command line:

1. Back up **/etc/X11/xorg.conf**:

```
# cp /etc/X11/xorg.conf /etc/X11/xorg.conf.$$.backup
```

2. Make the following change to the Device section of **/etc/X11/xorg.conf**:

```
Section  "Device"
Identifier "Videocard0"
Driver  "qxl"
Endsection
```

6.15.2.2. Configuring a Virtual Machine's Tablet and Mouse to use SPICE

Edit the **/etc/X11/xorg.conf** file to enable SPICE for your virtual machine's tablet devices.

Procedure 6.37. Configuring a Virtual Machine's Tablet and Mouse to use SPICE

1. Verify that the tablet device is available on your guest:

```
# /sbin/lsusb -v | grep 'QEMU USB Tablet'
```

If there is no output from the command, do not continue configuring the tablet.

2. Back up **/etc/X11/xorg.conf**:

```
# cp /etc/X11/xorg.conf /etc/X11/xorg.conf.$$.backup
```

3. Make the following changes to **/etc/X11/xorg.conf**:

```
Section "ServerLayout"
Identifier     "single head configuration"
Screen      0  "Screen0" 0 0
InputDevice    "Keyboard0" "CoreKeyboard"
InputDevice    "Tablet" "SendCoreEvents"
InputDevice    "Mouse" "CorePointer"
EndSection

Section "InputDevice"
Identifier  "Mouse"
Driver      "void"
#Option      "Device" "/dev/input/mice"
#Option      "Emulate3Buttons" "yes"
EndSection

Section "InputDevice"
Identifier  "Tablet"
Driver      "evdev"
Option      "Device" "/dev/input/event2"
Option "CorePointer" "true"
EndSection
```

4. Log out and log back into the virtual machine to restart X-Windows.

6.15.3. KVM virtual machine timing management

Virtualization poses various challenges for virtual machine time keeping. Virtual machines which use the Time Stamp Counter (TSC) as a clock source may suffer timing issues as some CPUs do not have a constant Time Stamp Counter. Virtual machines running without accurate timekeeping can have serious affects on some networked applications as your virtual machine will run faster or slower than the actual time.

KVM works around this issue by providing virtual machines with a paravirtualized clock. The KVM **pvclock** provides a stable source of timing for KVM guests that support it.

Presently, only Red Hat Enterprise Linux 5.4 and higher virtual machines fully support the paravirtualized clock.

Virtual machines can have several problems caused by inaccurate clocks and counters:

» Clocks can fall out of synchronization with the actual time which invalidates sessions and affects networks.

» Virtual machines with slower clocks may have issues migrating.

These problems exist on other virtualization platforms and timing should always be tested.

> ### ⭐ Important
>
> The Network Time Protocol (NTP) daemon should be running on the host and the virtual machines. Enable the **ntpd** service and add it to the default startup sequence:
>
> » For Red Hat Enterprise Linux 6
>
> ```
> # service ntpd start
> # chkconfig ntpd on
> ```
>
> » For Red Hat Enterprise Linux 7
>
> ```
> # systemctl start ntpd.service
> # systemctl enable ntpd.service
> ```
>
> Using the **ntpd** service should minimize the affects of clock skew in all cases.
>
> The NTP servers you are trying to use must be operational and accessible to your hosts and virtual machines.

Determining if your CPU has the constant Time Stamp Counter

Your CPU has a constant Time Stamp Counter if the **constant_tsc** flag is present. To determine if your CPU has the **constant_tsc** flag run the following command:

```
$ cat /proc/cpuinfo | grep constant_tsc
```

If any output is given your CPU has the **constant_tsc** bit. If no output is given follow the instructions below.

Configuring hosts without a constant Time Stamp Counter

Systems without constant time stamp counters require additional configuration. Power management features interfere with accurate time keeping and must be disabled for virtual machines to accurately keep time with KVM.

> ### ⭐ Important
>
> These instructions are for AMD revision F CPUs only.

If the CPU lacks the **constant_tsc** bit, disable all power management features (BZ#513138). Each system has several timers it uses to keep time. The TSC is not stable on the host, which is sometimes caused by **cpufreq** changes, deep C state, or migration to a host with a faster TSC. Deep C sleep states can stop the TSC. To prevent the kernel using deep C states append "**processor.max_cstate=1**" to the kernel boot options in the **grub.conf** file on the host:

```
term Red Hat Enterprise Linux Server (2.6.18-159.el5)
        root (hd0,0)
 kernel /vmlinuz-2.6.18-159.el5 ro root=/dev/VolGroup00/LogVol00 rhgb quiet
processor.max_cstate=1
```

Disable **cpufreq** (only necessary on hosts without the **constant_tsc**) by editing the **/etc/sysconfig/cpuspeed** configuration file and change the **MIN_SPEED** and **MAX_SPEED** variables to the highest frequency available. Valid limits can be found in the **/sys/devices/system/cpu/cpu*/cpufreq/scaling_available_frequencies** files.

Using the `engine-config` tool to receive alerts when hosts drift out of sync.

You can use the **engine-config** tool to configure alerts when your hosts drift out of sync.

There are 2 relevant parameters for time drift on hosts: **EnableHostTimeDrift** and **HostTimeDriftInSec**. **EnableHostTimeDrift**, with a default value of false, can be enabled to receive alert notifications of host time drift. The **HostTimeDriftInSec** parameter is used to set the maximum allowable drift before alerts start being sent.

Alerts are sent once per hour per host.

Using the paravirtualized clock with Red Hat Enterprise Linux virtual machines

For certain Red Hat Enterprise Linux virtual machines, additional kernel parameters are required. These parameters can be set by appending them to the end of the /kernel line in the /boot/grub/grub.conf file of the virtual machine.

> **Note**
>
> The process of configuring kernel parameters can be automated using the **ktune** package

The **ktune** package provides an interactive Bourne shell script,**fix_clock_drift.sh**. When run as the superuser, this script inspects various system parameters to determine if the virtual machine on which it is run is susceptible to clock drift under load. If so, it then creates a new **grub.conf.kvm** file in the **/boot/grub/** directory. This file contains a kernel boot line with additional kernel parameters that allow the kernel to account for and prevent significant clock drift on the KVM virtual machine. After running **fix_clock_drift.sh** as the superuser, and once the script has created the **grub.conf.kvm** file, then the virtual machine's current **grub.conf** file should be backed up manually by the system administrator, the new **grub.conf.kvm** file should be manually inspected to ensure that it is identical to**grub.conf** with the exception of the additional boot line parameters, the **grub.conf.kvm** file should finally be renamed **grub.conf**, and the virtual machine should be rebooted.

The table below lists versions of Red Hat Enterprise Linux and the parameters required for virtual machines on systems without a constant Time Stamp Counter.

Red Hat Enterprise Linux	Additional virtual machine kernel parameters
5.4 AMD64/Intel 64 with the paravirtualized clock	Additional parameters are not required
5.4 AMD64/Intel 64 without the paravirtualized clock	notsc lpj=n
5.4 x86 with the paravirtualized clock	Additional parameters are not required
5.4 x86 without the paravirtualized clock	clocksource=acpi_pm lpj=n
5.3 AMD64/Intel 64	notsc
5.3 x86	clocksource=acpi_pm
4.8 AMD64/Intel 64	notsc
4.8 x86	clock=pmtmr
3.9 AMD64/Intel 64	Additional parameters are not required
3.9 x86	Additional parameters are not required

Chapter 7. Templates

A template is a copy of a virtual machine that you can use to simplify the subsequent, repeated creation of similar virtual machines. Templates capture the configuration of software, configuration of hardware, and the software installed on the virtual machine on which the template is based. The virtual machine on which a template is based is known as the source virtual machine.

When you create a template based on a virtual machine, a read-only copy of the virtual machine's disk is created. This read-only disk becomes the base disk image of the new template, and of any virtual machines created based on the template. As such, the template cannot be deleted while any virtual machines created based on the template exist in the environment.

Virtual machines created based on a template use the same NIC type and driver as the original virtual machine, but are assigned separate, unique MAC addresses.

You can create a virtual machine directly from the **Templates** tab, as well as from the **Virtual Machines** tab. In the **Templates** tab, right-click the required template and select **New VM**. For more information on selecting the settings and controls for the new virtual machine see Section A.1.1, "Virtual Machine General Settings Explained".

7.1. Sealing Virtual Machines in Preparation for Deployment as Templates

This section describes procedures for sealing Linux virtual machines and Windows virtual machines. Sealing is the process of removing all system-specific details from a virtual machine before creating a template based on that virtual machine. Sealing is necessary to prevent the same details from appearing on multiple virtual machines created based on the same template. It is also necessary to ensure the functionality of other features, such as predictable vNIC order.

7.1.1. Sealing a Linux Virtual Machine for Deployment as a Template

There are two main methods for sealing a Linux virtual machine in preparation for using that virtual machine to create a template: manually, or using the **sys-unconfig** command. Sealing a Linux virtual machine manually requires you to create a file on the virtual machine that acts as a flag for initiating various configuration tasks the next time you start that virtual machine. The **sys-unconfig** command allows you to automate this process. However, both of these methods also require you to manually delete files on the virtual machine that are specific to that virtual machine or might cause conflicts amongst virtual machines created based on the template you will create based on that virtual machine. As such, both are valid methods for sealing a Linux virtual machine and will achieve the same result.

7.1.1.1. Sealing a Linux Virtual Machine Manually for Deployment as a Template

You must generalize (seal) a Linux virtual machine before creating a template based on that virtual machine.

Procedure 7.1. Sealing a Linux Virtual Machine

1. Log in to the virtual machine.

2. Flag the system for re-configuration:

```
# touch /.unconfigured
```

3. Remove SSH host keys:

```
# rm -rf /etc/ssh/ssh_host_*
```

4. Change the host name to **localhost.localdomain**.

   A. For Red Hat Enterprise Linux 6, edit the **HOSTNAME** value in **/etc/sysconfig/network**:

   ```
   HOSTNAME=localhost.localdomain
   ```

   B. For Red Hat Enterprise Linux 7, use the **hostnamectl** command:

   ```
   # hostnamectl set-hostname localhost.localdomain
   ```

5. Remove **/etc/udev/rules.d/70-***:

```
# rm -rf /etc/udev/rules.d/70-*
```

6. Remove the **HWADDR** line and **UUID** line from **/etc/sysconfig/network-scripts/ifcfg-eth***.

7. Optionally, delete all the logs from **/var/log** and build logs from **/root**.

8. Shut down the virtual machine:

```
# poweroff
```

The virtual machine is sealed and can be made into a template. You can deploy Linux virtual machines from this template without experiencing configuration file conflicts.

The steps provided are the minimum steps required to seal a Red Hat Enterprise Linux virtual machine for use as a template. Additional host and site-specific custom steps are available.

7.1.1.2. Sealing a Linux Virtual Machine for Deployment as a Template using sys-unconfig

You must generalize (seal) a Linux virtual machine before creating a template based on that virtual machine.

Procedure 7.2. Sealing a Linux Virtual Machine using sys-unconfig

1. Log in to the virtual machine.

2. Remove SSH host keys:

```
# rm -rf /etc/ssh/ssh_host_*
```

3. Change the host name to **localhost.localdomain**.

   A. For Red Hat Enterprise Linux 6, edit the **HOSTNAME** value in **/etc/sysconfig/network**:

```
HOSTNAME=localhost.localdomain
```

   B. For Red Hat Enterprise Linux 7, use the **hostnamectl** command:

```
# hostnamectl set-hostname localhost.localdomain
```

4. Remove the **HWADDR** line and **UUID** line from **/etc/sysconfig/network-scripts/ifcfg-eth***.

5. Optionally, delete all the logs from **/var/log** and build logs from **/root**.

6. Run the following command:

```
# sys-unconfig
```

The virtual machine shuts down; it is now sealed and can be made into a template. You can deploy Linux virtual machines from this template without experiencing configuration file conflicts.

7.1.2. Sealing a Windows Virtual Machine for Deployment as a Template

A template created for Windows virtual machines must be generalized (sealed) before being used to deploy virtual machines. This ensures that machine-specific settings are not reproduced in the template.

The **Sysprep** tool is used to seal Windows templates before use.

> **Important**
>
> Do not reboot the virtual machine during this process.

Before starting the **Sysprep** process, verify that the following settings are configured:

» The Windows **Sysprep** parameters have been correctly defined.

   If not, click **Edit** and enter the required information in the **Operating System** and **Domain** fields.

» The correct product key has been defined in an override file on the Manager.

   The override file needs to be created under **/etc/ovirt-engine/osinfo.conf.d/**, have a filename that puts it after **/etc/ovirt-engine/osinfo.conf.d/00-defaults.properties**, and end in **.properties**. For example, **/etc/ovirt-engine/osinfo.conf.d/10-productkeys.properties**. The last file will have precedent and override any other previous file.

   If not, copy the default values for your Windows operating system from **/etc/ovirt-engine/osinfo.conf.d/00-defaults.properties** into the override file, and input your values in the **productKey.value** and **sysprepPath.value** fields.

   Example 7.1. Windows 7 Default Configuration Values

```
# Windows7(11, OsType.Windows, false),false
```

```
os.windows_7.id.value = 11
os.windows_7.name.value = Windows 7
os.windows_7.derivedFrom.value = windows_xp
os.windows_7.sysprepPath.value = ${ENGINE_USR}/conf/sysprep/sysprep.w7
os.windows_7.productKey.value =
os.windows_7.devices.audio.value = ich6
os.windows_7.devices.diskInterfaces.value.3.3 = IDE, VirtIO_SCSI, VirtIO
os.windows_7.devices.diskInterfaces.value.3.4 = IDE, VirtIO_SCSI, VirtIO
os.windows_7.devices.diskInterfaces.value.3.5 = IDE, VirtIO_SCSI, VirtIO
os.windows_7.isTimezoneTypeInteger.value = false
```

7.1.2.1. Sealing a Windows 7, Windows 2008, or Windows 2012 Template

Seal a Windows 7, Windows 2008, or Windows 2012 template before using the template to deploy virtual machines.

Procedure 7.3. Sealing a Windows 7, Windows 2008, or Windows 2012 Template

1. Launch **Sysprep** from **C:\Windows\System32\sysprep\sysprep.exe**.

2. Enter the following information into the **Sysprep** tool:

   - Under **System Cleanup Action**, select **Enter System Out-of-Box-Experience (OOBE)**.

   - Select the **Generalize** check box if you need to change the computer's system identification number (SID).

   - Under **Shutdown Options**, select **Shutdown**.

3. Click **OK** to complete the sealing process; the virtual machine shuts down automatically upon completion.

The Windows 7, Windows 2008, or Windows 2012 template is sealed and ready for deploying virtual machines.

7.2. Creating a Template

Create a template from an existing virtual machine to use as a blueprint for creating additional virtual machines.

> ⭐ **Important**
>
> Before you create a template, you must seal the source virtual machine to ensure all system-specific details are removed from the virtual machine. This is necessary to prevent the same details from appearing on multiple virtual machines created based on the same template. See Section 7.1.1, "Sealing a Linux Virtual Machine for Deployment as a Template".

When creating a template, you can choose the format of the disk: RAW or QCOW2. QCOW2 always implies that a disk is thin provisioned. RAW on file storage implies thin provisioned, while RAW on block storage implies preallocated virtual disks.

Procedure 7.4. Creating a Template

1. Click the **Virtual Machines** tab and select the source virtual machine.

2. Ensure the virtual machine is powered down and has a status of **Down**.

3. Click **Make Template**.

Figure 7.1. The New Template window

4. Enter a **Name**, **Description**, and **Comment** for the template.

5. Select the cluster with which to associate the template from the **Cluster** drop-down list. By default, this is the same as that of the source virtual machine.

6. Optionally, select a CPU profile for the template from the **CPU Profile** drop-down list.

7. Optionally, select the **Create as a Sub Template version** check box, select a **Root Template**, and enter a **Sub Version Name** to create the new template as a sub template of an existing template.

8. In the **Disks Allocation** section, enter an alias for the disk in the **Alias** text field. Select the disk format in the **Format** drop-down, the storage domain on which to store the disk from the **Target** drop-down, and the disk profile in the **Disk Profile** drop-down. By default, these are the same as those of the source virtual machine.

9. Select the **Allow all users to access this Template** check box to make the template public.

10. Select the **Copy VM permissions** check box to copy the permissions of the source virtual machine to the template.

11. Click **OK**.

The virtual machine displays a status of **Image Locked** while the template is being created. The process of creating a template may take up to an hour depending on the size of the virtual machine disk and the capabilities of your storage hardware. When complete, the template is added to the **Templates** tab. You can now create new virtual machines based on the template.

> **Note**
>
> When a template is made, the virtual machine is copied so that both the existing virtual machine and its template are usable after template creation.

7.3. Editing a Template

Once a template has been created, its properties can be edited. Because a template is a copy of a virtual machine, the options available when editing a template are identical to those in the **Edit Virtual Machine** window.

Procedure 7.5. Editing a Template

1. Click the **Templates** tab and select a template.

2. Click **Edit**.

3. Change the necessary properties.

4. Click **OK**.

7.4. Deleting a Template

If you have used a template to create a virtual machine using the thin provisioning storage allocation option, the template cannot be deleted as the virtual machine needs it to continue running. However, cloned virtual machines do not depend on the template they were cloned from and the template can be deleted.

Procedure 7.6. Deleting a Template

1. Click the **Templates** tab and select a template.

2. Click **Remove**.

3. Click **OK**.

7.5. Exporting Templates

7.5.1. Migrating Templates to the Export Domain

> **Note**
>
> The export storage domain is deprecated. Storage data domains can be unattached from a data center and imported to another data center in the same environment, or in a different environment. Virtual machines, floating virtual disk images, and templates can then be uploaded from the imported storage domain to the attached data center. See the Importing Existing Storage Domains section in the *Red Hat Virtualization Administration Guide* for information on importing storage domains.

Export templates into the export domain to move them to another data domain, either in the same Red Hat Virtualization environment, or another one. This procedure requires access to the Administration Portal.

Procedure 7.7. Exporting Individual Templates to the Export Domain

1. Click the **Templates** tab and select a template.

2. Click **Export**.

3. Select the **Force Override** check box to replace any earlier version of the template on the export domain.

4. Click **OK** to begin exporting the template; this may take up to an hour, depending on the virtual machine disk image size and your storage hardware.

Repeat these steps until the export domain contains all the templates to migrate before you start the import process.

Click the **Storage** tab, select the export domain, and click the **Template Import** tab in the details pane to view all exported templates in the export domain.

7.5.2. Copying a Template's Virtual Hard Disk

If you are moving a virtual machine that was created from a template with the thin provisioning storage allocation option selected, the template's disks must be copied to the same storage domain as that of the virtual machine disk. This procedure requires access to the Administration Portal.

Procedure 7.8. Copying a Virtual Hard Disk

1. Click the **Disks** tab and select the template disk(s) to copy.

2. Click **Copy**.

3. Select the **Target** data domain from the drop-down list(s).

4. Click **OK**.

A copy of the template's virtual hard disk has been created, either on the same, or a different, storage domain. If you were copying a template disk in preparation for moving a virtual hard disk, you can now move the virtual hard disk.

7.6. Importing Templates

7.6.1. Importing a Template into a Data Center

> **Note**
>
> The export storage domain is deprecated. Storage data domains can be unattached from a data center and imported to another data center in the same environment, or in a different environment. Virtual machines, floating virtual disk images, and templates can then be uploaded from the imported storage domain to the attached data center. See the Importing Existing Storage Domains section in the *Red Hat Virtualization Administration Guide* for information on importing storage domains.

Import templates from a newly attached export domain. This procedure requires access to the Administration Portal.

Procedure 7.9. Importing a Template into a Data Center

1. Click the **Storage** tab and select the newly attached export domain.

2. Click the **Template Import** tab in the details pane and select a template.

3. Click **Import**.

4. Select the templates to import.

5. Use the drop-down lists to select the **Destination Cluster** and **Storage** domain. Alter the **Suffix** if applicable.

   Alternatively, clear the **Clone All Templates** check box.

6. Click **OK** to import templates and open a notification window. Click **Close** to close the notification window.

The template is imported into the destination data center. This can take up to an hour, depending on your storage hardware. You can view the import progress in the **Events** tab.

Once the importing process is complete, the templates will be visible in the **Templates** resource tab. The templates can create new virtual machines, or run existing imported virtual machines based on that template.

7.6.2. Importing a Virtual Disk Image from an OpenStack Image Service as a Template

Virtual disk images managed by an OpenStack Image Service can be imported into the Red Hat Virtualization Manager if that OpenStack Image Service has been added to the Manager as an external provider. This procedure requires access to the Administration Portal.

1. Click the **Storage** tab and select the OpenStack Image Service domain.

2. Click the **Images** tab in the details pane and select the image to import.

3. Click **Import**.

4. Select the **Data Center** into which the virtual disk image will be imported.

5. Select the storage domain in which the virtual disk image will be stored from the **Domain Name** drop-down list.

6. Optionally, select a **Quota** to apply to the virtual disk image.

7. Select the **Import as Template** check box.

8. Select the **Cluster** in which the virtual disk image will be made available as a template.

9. Click **OK**.

The image is imported as a template and is displayed in the **Templates** tab. You can now create virtual machines based on the template.

7.7. Templates and Permissions

7.7.1. Managing System Permissions for a Template

As the **SuperUser**, the system administrator manages all aspects of the Administration Portal. More specific administrative roles can be assigned to other users. These restricted administrator roles are useful for granting a user administrative privileges that limit them to a specific resource. For example, a **DataCenterAdmin** role has administrator privileges only for the assigned data center with the exception of the storage for that data center, and a **ClusterAdmin** has administrator privileges only for the assigned cluster.

A template administrator is a system administration role for templates in a data center. This role can be applied to specific virtual machines, to a data center, or to the whole virtualized environment; this is useful to allow different users to manage certain virtual resources.

The template administrator role permits the following actions:

» Create, edit, export, and remove associated templates.

» Import and export templates.

> **Note**
>
> You can only assign roles and permissions to existing users.

7.7.2. Template Administrator Roles Explained

The table below describes the administrator roles and privileges applicable to template administration.

Table 7.1. Red Hat Virtualization System Administrator Roles

Role	Privileges	Notes
TemplateAdmin	Can perform all operations on templates.	Has privileges to create, delete and configure a template's storage domain and network details, and to move templates between domains.
NetworkAdmin	Network Administrator	Can configure and manage networks attached to templates.

7.7.3. Template User Roles Explained

The table below describes the user roles and privileges applicable to using and administrating templates in the User Portal.

Table 7.2. Red Hat Virtualization Template User Roles

Role	Privileges	Notes
TemplateCreator	Can create, edit, manage and remove virtual machine templates within assigned resources.	The **TemplateCreator** role is not applied to a specific template; apply this role to a user for the whole environment with the **Configure** window. Alternatively apply this role for specific data centers, clusters, or storage domains.
TemplateOwner	Can edit and delete the template, assign and manage user permissions for the template.	The **TemplateOwner** role is automatically assigned to the user who creates a template. Other users who do not have **TemplateOwner** permissions on a template cannot view or use the template.
UserTemplateBasedVm	Can use the template to create virtual machines.	Cannot edit template properties.
VnicProfileUser	Logical network and network interface user for templates.	If the **Allow all users to use this Network** option was selected when a logical network is created, **VnicProfileUser** permissions are assigned to all users for the logical network. Users can then attach or detach template network interfaces to or from the logical network.

7.7.4. Assigning an Administrator or User Role to a Resource

Assign administrator or user roles to resources to allow users to access or manage that resource.

Procedure 7.10. Assigning a Role to a Resource

1. Use the resource tabs, tree mode, or the search function to find and select the resource in the results list.

2. Click the **Permissions** tab in the details pane to list the assigned users, the user's role, and the inherited permissions for the selected resource.

3. Click **Add**.

4. Enter the name or user name of an existing user into the **Search** text box and click **Go**. Select a user from the resulting list of possible matches.

5. Select a role from the **Role to Assign:** drop-down list.

6. Click **OK**.

You have assigned a role to a user; the user now has the inherited permissions of that role enabled for that resource.

7.7.5. Removing an Administrator or User Role from a Resource

Remove an administrator or user role from a resource; the user loses the inherited permissions associated with the role for that resource.

Procedure 7.11. Removing a Role from a Resource

1. Use the resource tabs, tree mode, or the search function to find and select the resource in the results list.

2. Click the **Permissions** tab in the details pane to list the assigned users, the user's role, and the inherited permissions for the selected resource.

3. Select the user to remove from the resource.

4. Click **Remove**. The **Remove Permission** window opens to confirm permissions removal.

5. Click **OK**.

You have removed the user's role, and the associated permissions, from the resource.

7.8. Using Cloud-Init to Automate the Configuration of Virtual Machines

Cloud-Init is a tool for automating the initial setup of virtual machines such as configuring the host name, network interfaces, and authorized keys. It can be used when provisioning virtual machines that have been deployed based on a template to avoid conflicts on the network.

To use this tool, the *cloud-init* package must first be installed on the virtual machine. Once installed, the Cloud-Init service starts during the boot process to search for instructions on what to configure. You can then use options in the **Run Once** window to provide these instructions one time only, or options in the **New Virtual Machine**, **Edit Virtual Machine** and **Edit Template** windows to provide these instructions every time the virtual machine starts.

7.8.1. Cloud-Init Use Case Scenarios

Cloud-Init can be used to automate the configuration of virtual machines in a variety of scenarios. Several common scenarios are as follows:

Virtual Machines Created Based on Templates

You can use the Cloud-Init options in the **Initial Run** section of the **Run Once** window to initialize a virtual machine that was created based on a template. This allows you to customize the virtual machine the first time that virtual machine is started.

Virtual Machine Templates

You can use the **Use Cloud-Init/Sysprep** options in the **Initial Run** tab of the **New Template** and **Edit Template** windows to specify options for customizing virtual machines created based on that template.

Virtual Machine Pools

You can use the **Use Cloud-Init/Sysprep** options in the **Initial Run** tab of the **New Pool** window to specify options for customizing virtual machines taken from that virtual machine pool. This allows you to specify a set of standard settings that will be applied every time a virtual machine is taken from that virtual machine pool. You can inherit or override the options specified for the template on which the virtual machine is based, or specify options for the virtual machine pool itself.

7.8.2. Installing Cloud-Init

This procedure describes how to install Cloud-Init on a virtual machine. Once Cloud-Init is installed, you can create a template based on this virtual machine. Virtual machines created based on this template can leverage Cloud-Init functions, such as configuring the host name, time zone, root password, authorized keys, network interfaces, DNS service, etc on boot.

Procedure 7.12. Installing Cloud-Init

1. Log on to the virtual machine.

2. Enable the required repositories:

   A. Red Hat Enterprise Linux 6:

      ```
      # subscription-manager repos --enable=rhel-6-server-rpms
      # subscription-manager repos --enable=rhel-6-server-rh-common-rpms
      ```

   B. Red Hat Enterprise Linux 7:

      ```
      # subscription-manager repos --enable=rhel-7-server-rpms
      # subscription-manager repos --enable=rhel-7-server-rh-common-rpms
      ```

3. Install the *cloud-init* package and dependencies:

   ```
   # yum install cloud-init
   ```

7.8.3. Using Cloud-Init to Prepare a Template

As long as the *cloud-init* package is installed on a Linux virtual machine, you can use the virtual machine to make a cloud-init enabled template. Specify a set of standard settings to be included in a template as described in the following procedure or, alternatively, skip the Cloud-Init settings steps and configure them when creating a virtual machine based on this template.

> **Note**
>
> While the following procedure outlines how to use Cloud-Init when preparing a template, the same settings are also available in the **New Virtual Machine**, **Edit Template**, and **Run Once** windows.

Procedure 7.13. Using Cloud-Init to Prepare a Template

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Edit**.

3. Click the **Initial Run** tab and select the **Use Cloud-Init/Sysprep** check box.

4. Enter a host name in the **VM Hostname** text field.

5. Select the **Configure Time Zone** check box and select a time zone from the **Time Zone** drop-down list.

6. Expand the **Authentication** section and select the **Use already configured password** check box to user the existing credentials, or clear that check box and enter a root password in the **Password** and **Verify Password** text fields to specify a new root password.

7. Enter any SSH keys to be added to the authorized hosts file on the virtual machine in the **SSH Authorized Keys** text area.

8. Select the **Regenerate SSH Keys** check box to regenerate SSH keys for the virtual machine.

9. Expand the **Networks** section and enter any DNS servers in the **DNS Servers** text field.

10. Enter any DNS search domains in the **DNS Search Domains** text field.

11. Select the **Network** check box and use the **+** and **-** buttons to add or remove network interfaces to or from the virtual machine.

12. Expand the **Custom Script** section and enter any custom scripts in the **Custom Script** text area.

13. Click **Ok**.

14. Click **Make Template** and enter the fields as necessary.

15. Click **Ok**.

You can now provision new virtual machines using this template.

7.8.4. Using Cloud-Init to Initialize a Virtual Machine

Use Cloud-Init to automate the initial configuration of a Linux virtual machine. You can use the Cloud-Init fields to configure a virtual machine's host name, time zone, root password,

authorized keys, network interfaces, and DNS service. You can also specify a custom script, a script in YAML format, to run on boot. The custom script allows for additional Cloud-Init configuration that is supported by Cloud-Init but not available in the Cloud-Init fields. For more information on custom script examples, see Cloud config examples.

Procedure 7.14. Using Cloud-Init to Initialize a Virtual Machine

This procedure starts a virtual machine with a set of Cloud-Init settings. If the relevant settings are included in the template the virtual machine is based on, review the settings, make changes where appropriate, and click **OK** to start the virtual machine.

1. Click the **Virtual Machines** tab and select a virtual machine.

2. Click **Run Once**.

3. Expand the **Initial Run** section and select the **Cloud-Init** check box.

4. Enter a host name in the **VM Hostname** text field.

5. Select the **Configure Time Zone** check box and select a time zone from the **Time Zone** drop-down menu.

6. Select the **Use already configured password** check box to use the existing credentials, or clear that check box and enter a root password in the **Password** and **Verify Password** text fields to specify a new root password.

7. Enter any SSH keys to be added to the authorized hosts file on the virtual machine in the **SSH Authorized Keys** text area.

8. Select the **Regenerate SSH Keys** check box to regenerate SSH keys for the virtual machine.

9. Enter any DNS servers in the **DNS Servers** text field.

10. Enter any DNS search domains in the **DNS Search Domains** text field.

11. Select the **Network** check box and use the **+** and **-** buttons to add or remove network interfaces to or from the virtual machine.

12. Enter a custom script in the **Custom Script** text area. Make sure the values specified in the script are appropriate. Otherwise, the action will fail.

13. Click **OK**.

> **Note**
>
> To check if a virtual machine has Cloud-Init installed, select a virtual machine and click the **Applications** sub-tab. Only shown if the guest agent is installed.

7.9. Creating a Virtual Machine Based on a Template

Create a virtual machine from a template to enable the virtual machines to be pre-configured with an operating system, network interfaces, applications and other resources.

> **Note**
>
> Virtual machines created from a template depend on that template. This means that you cannot remove a template from the Manager if a virtual machine was created from that template. However, you can clone a virtual machine from a template to remove the dependency on that template. See Section 7.10, "Creating a Cloned Virtual Machine Based on a Template" for more information.

When creating a virtual machine from a template, you can choose the format of the disk: either RAW or QCOW2. If the **Storage Allocation** is **Thin**, the format of the disk will be marked as QCOW2 and you will not be able to change it. If the **Storage Allocation** is Clone, you can select either QCOW2 or RAW.

Procedure 7.15. Creating a Virtual Machine Based on a Template

1. Click the **Virtual Machines** tab.

2. Click **New VM**.

3. Select the **Cluster** on which the virtual machine will run.

4. Select a template from the **Based on Template** list.

5. Enter a **Name**, **Description**, and any **Comments**, and accept the default values inherited from the template in the rest of the fields. You can change them if needed.

6. Click the **Resource Allocation** tab.

7. Select the **Thin** radio button in the **Storage Allocation** area.

8. Use the **Target** drop-down list to select the storage domain on which the virtual machine's virtual disk will be stored.

9. Click **OK**.

The virtual machine is displayed in the **Virtual Machines** tab.

7.10. Creating a Cloned Virtual Machine Based on a Template

Cloned virtual machines are based on templates and inherit the settings of the template. A cloned virtual machine does not depend on the template on which it was based after it has been created. This means the template can be deleted if no other dependencies exist.

> **Note**
>
> If you clone a virtual machine from a template, the name of the template on which that virtual machine was based is displayed in the **General** tab of the **Edit Virtual Machine** window for that virtual machine. If you change the name of that template, the name of the template in the **General** tab will also be updated. However, if you delete the template from the Manager, the original name of that template will be displayed instead.

When creating a virtual machine from a template, you can choose the format of the disk: either RAW or QCOW2. If the **Storage Allocation** is **Thin**, the format of the disk will be marked as QCOW2 and you will not be able to change it. If the **Storage Allocation** is Clone, you can select either QCOW2 or RAW.

Procedure 7.16. Cloning a Virtual Machine Based on a Template

1. Click the **Virtual Machines** tab.

2. Click **New VM**.

3. Select the **Cluster** on which the virtual machine will run.

4. Select a template from the **Based on Template** drop-down menu.

5. Enter a **Name**, **Description** and any **Comments**. You can accept the default values inherited from the template in the rest of the fields, or change them if required.

6. Click the **Resource Allocation** tab.

7. Select the **Clone** radio button in the **Storage Allocation** area.

8. Select the disk format from the **Format** drop-down list. This affects the speed of the clone operation and the amount of disk space the new virtual machine initially requires.

   ▹ Selecting **QCOW2** results in a faster clone operation and provides optimized usage of storage capacity. Disk space is allocated only as it is required. This is the default selection.

   ▹ Selecting **Raw** results in a slower clone operation and provides optimized virtual machine read and write operations. All disk space requested in the template is allocated at the time of the clone operation.

9. Use the **Target** drop-down menu to select the storage domain on which the virtual machine's virtual disk will be stored.

10. Click **OK**.

> **Note**
>
> Cloning a virtual machine may take some time. A new copy of the template's disk must be created. During this time, the virtual machine's status is first **Image Locked**, then **Down**.

The virtual machine is created and displayed in the **Virtual Machines** tab. You can now assign users to it, and can begin using it when the clone operation is complete.

Appendix A. Reference: Settings in Administration Portal and User Portal Windows

A.1. Explanation of Settings in the New Virtual Machine and Edit Virtual Machine Windows

A.1.1. Virtual Machine General Settings Explained

The following table details the options available on the **General** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows.

Table A.1. Virtual Machine: General Settings

Field Name	Description
Cluster	The name of the host cluster to which the virtual machine is attached. Virtual machines are hosted on any physical machine in that cluster in accordance with policy rules.
Based on Template	The template on which the virtual machine can be based. This field is set to **Blank** by default, which allows you to create a virtual machine on which an operating system has not yet been installed.
Template Sub Version	The version of the template on which the virtual machine can be based. This field is set to the most recent version for the given template by default. If no versions other than the base template are available, this field is set to **base template** by default. Each version is marked by a number in brackets that indicates the relative order of the versions, with higher numbers indicating more recent versions.
Operating System	The operating system. Valid values include a range of Red Hat Enterprise Linux and Windows variants.

Field Name	Description
`Instance Type`	The instance type on which the virtual machine's hardware configuration can be based. This field is set to **Custom** by default, which means the virtual machine is not connected to an instance type. The other options available from this drop down menu are **Large**, **Medium**, **Small**, **Tiny**, **XLarge**, and any custom instance types that the Administrator has created. Other settings that have a chain link icon next to them are pre-filled by the selected instance type. If one of these values is changed, the virtual machine will be detached from the instance type and the chain icon will appear broken. However, if the changed setting is restored to its original value, the virtual machine will be reattached to the instance type and the links in the chain icon will rejoin.
`Optimized for`	The type of system for which the virtual machine is to be optimized. There are two options: **Server**, and **Desktop**; by default, the field is set to **Server**. Virtual machines optimized to act as servers have no sound card, use a cloned disk image, and are not stateless. In contrast, virtual machines optimized to act as desktop machines do have a sound card, use an image (thin allocation), and are stateless.
`Name`	The name of the virtual machine. The name must be a unique name within the data center and must not contain any spaces, and must contain at least one character from A-Z or 0-9. The maximum length of a virtual machine name is 255 characters. The name can be re-used in different data centers in the environment.
`VM Id`	The virtual machine ID. The virtual machine's creator can set a custom ID for that virtual machine. If no ID is specified during creation a UUID will be automatically assigned. For both custom and automatically-generated IDs, changes are not possible after virtual machine creation.
`Description`	A meaningful description of the new virtual machine.
`Comment`	A field for adding plain text human-readable comments regarding the virtual machine.

Field Name	Description
Stateless	Select this check box to run the virtual machine in stateless mode. This mode is used primarily for desktop virtual machines. Running a stateless desktop or server creates a new COW layer on the VM hard disk image where new and changed data is stored. Shutting down the stateless VM deletes the new COW layer, which returns the VM to its original state. Stateless virtual machines are useful when creating machines that need to be used for a short time, or by temporary staff.
Start in Pause Mode	Select this check box to always start the virtual machine in pause mode. This option is suitable for virtual machines which require a long time to establish a SPICE connection; for example, virtual machines in remote locations.
Delete Protection	Select this check box to make it impossible to delete the virtual machine. It is only possible to delete the virtual machine if this check box is not selected.
Instance Images	Click **Attach** to attach a floating disk to the virtual machine, or click **Create** to add a new virtual disk. Use the plus and minus buttons to add or remove additional virtual disks. Click **Edit** to reopen the **Attach Virtual Disks** or **New Virtual Disk** window. This button appears after a virtual disk has been attached or created.
Instantiate VM network interfaces by picking a vNIC profile.	Add a network interface to the virtual machine by selecting a vNIC profile from the **nic1** drop-down list. Use the plus and minus buttons to add or remove additional network interfaces.

A.1.2. Virtual Machine System Settings Explained

The following table details the options available on the **System** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows.

Table A.2. Virtual Machine: System Settings

Field Name	Description

Field Name	Description
Memory Size	The amount of memory assigned to the virtual machine. When allocating memory, consider the processing and storage needs of the applications that are intended to run on the virtual machine. Maximum guest memory is constrained by the selected guest architecture and the cluster compatibility level.
Total Virtual CPUs	The processing power allocated to the virtual machine as CPU Cores. Do not assign more cores to a virtual machine than are present on the physical host.
Virtual Sockets	The number of CPU sockets for the virtual machine. Do not assign more sockets to a virtual machine than are present on the physical host.
Cores per Virtual Socket	The number of cores assigned to each virtual socket.
Threads per Core	The number of threads assigned to each core. Increasing the value enables simultaneous multi-threading (SMT). IBM POWER8 supports up to 8 threads per core. For x86 (Intel and AMD) CPU types, the recommended value is 1.
Custom Emulated Machine	This option allows you to specify the machine type. If changed, the virtual machine will only run on hosts that support this machine type. Defaults to the cluster's default machine type.
Custom CPU Type	This option allows you to specify a CPU type. If changed, the virtual machine will only run on hosts that support this CPU type. Defaults to the cluster's default CPU type.
Time Zone	This option sets the time zone offset of the guest hardware clock. For Windows, this should correspond to the time zone set in the guest. Most default Linux installations expect the hardware clock to be GMT+00:00.
Provide custom serial number policy	This check box allows you to specify a serial number for the virtual machine. Select either: » **Host ID**: Sets the host's UUID as the virtual machine's serial number. » **Vm ID**: Sets the virtual machine's UUID as its serial number. » **Custom serial number**: Allows you to specify a custom serial number.

A.1.3. Virtual Machine Initial Run Settings Explained

The following table details the options available on the **Initial Run** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows. The settings in this table are only visible if the **Use Cloud-Init/Sysprep** check box is selected, and certain options are only visible when either a Linux-based or Windows-based option has been selected in the **Operating System** list in the **General** tab, as outlined below.

Table A.3. Virtual Machine: Initial Run Settings

Field Name	Operating System	Description
Use Cloud-Init/Sysprep	Linux, Windows	This check box toggles whether Cloud-Init or Sysprep will be used to initialize the virtual machine.
VM Hostname	Linux, Windows	The host name of the virtual machine.
Domain	Windows	The Active Directory domain to which the virtual machine belongs.
Organization Name	Windows	The name of the organization to which the virtual machine belongs. This option corresponds to the text field for setting the organization name displayed when a machine running Windows is started for the first time.
Active Directory OU	Windows	The organizational unit in the Active Directory domain to which the virtual machine belongs.
Configure Time Zone	Linux, Windows	The time zone for the virtual machine. Select this check box and select a time zone from the **Time Zone** list.
Admin Password	Windows	The administrative user password for the virtual machine. Click the disclosure arrow to display the settings for this option. » **Use already configured password**: This check box is automatically selected after you specify an initial administrative user password. You must clear this check box to enable the **Admin Password** and **Verify Admin Password** fields and specify a new password. » **Admin Password**: The administrative user password for the virtual machine. Enter the password in this text field and the **Verify Admin Password** text field to verify the password.

Field Name	Operating System	Description
`Authentication`	Linux	The authentication details for the virtual machine. Click the disclosure arrow to display the settings for this option. » **Use already configured password**: This check box is automatically selected after you specify an initial root password. You must clear this check box to enable the **Password** and **Verify Password** fields and specify a new password. » **Password**: The root password for the virtual machine. Enter the password in this text field and the **Verify Password** text field to verify the password. » **SSH Authorized Keys**: SSH keys to be added to the authorized keys file of the virtual machine. You can specify multiple SSH keys by entering eachSSH key on a new line. » **Regenerate SSH Keys**: Regenerates SSH keys for the virtual machine.
`Custom Locale`	Windows	Custom locale options for the virtual machine. Locales must be in a format such as **en-US**. Click the disclosure arrow to display the settings for this option. » **Input Locale**: The locale for user input. » **UI Language**: The language used for user interface elements such as buttons and menus. » **System Locale**: The locale for the overall system. » **User Locale**: The locale for users.

Field Name	Operating System	Description

`Networks`	Linux	Network-related settings for the virtual machine. Click the disclosure arrow to display the settings for this option.
		
		  » **DNS Servers**: The DNS servers to be used by the virtual machine.
		  » **DNS Search Domains**: The DNS search domains to be used by the virtual machine.
		  » **Network**: Configures network interfaces for the virtual machine. Select this check box and click **+** or **-** to add or remove network interfaces to or from the virtual machine. When you click **+**, a set of fields becomes visible that can specify whether to use DHCP, and configure an IP address, netmask, and gateway, and specify whether the network interface will start on boot.

`Custom Script`	Linux	Custom scripts that will be run on the virtual machine when it starts. The scripts entered in this field are custom YAML sections that are added to those produced by the Manager, and allow you to automate tasks such as creating users and files, configuring **yum** repositories and running commands. For more information on the format of scripts that can be entered in this field, see the Custom Script documentation.
`Sysprep`	Windows	A custom Sysprep definition. The definition must be in the format of a complete unattended installation answer file. You can copy and paste the default answer files in the **/usr/share/ovirt-engine/conf/sysprep/** directory on the machine on which the Red Hat Virtualization Manager is installed and alter the fields as required.

A.1.4. Virtual Machine Console Settings Explained

The following table details the options available on the `Console` tab of the `New Virtual Machine` and `Edit Virtual Machine` windows.

Table A.4. Virtual Machine: Console Settings

Field Name	Description
`Graphics protocol`	Defines which display protocol to use. **SPICE** is the recommended protocol as it supports more features. **VNC** is an alternative option and requires a VNC client to connect to a virtual machine. Select **SPICE + VNC** for the most flexible option.

Field Name	Description
`VNC Keyboard Layout`	Defines the keyboard layout for the virtual machine. This option is only available when using the VNC protocol.
`USB Support`	Defines whether USB devices can be used on the virtual machine. This option is only available for virtual machines using the SPICE protocol. Select either: » `Disabled` - Does not allow USB redirection from the client machine to the virtual machine. » `Legacy` - Enables the SPICE USB redirection policy used in Red Hat Enterprise Virtualization 3.0. This option can only be used on Windows virtual machines, and will not be supported in future versions of Red Hat Virtualization. **Important** The Legacy USB option has been deprecated and will be removed in Red Hat Virtualization 4.1. » `Native` - Enables native KVM/SPICE USB redirection for Linux and Windows virtual machines. Virtual machines do not require any in-guest agents or drivers for native USB.
`Monitors`	The number of monitors for the virtual machine. This option is only available for virtual desktops using the SPICE display protocol. You can choose **1**, **2** or **4**. Note that multiple monitors are not supported for Windows 8 and Windows Server 2012 virtual machines.
`Smartcard Enabled`	Smart cards are an external hardware security feature, most commonly seen in credit cards, but also used by many businesses as authentication tokens. Smart cards can be used to protect Red Hat Virtualization virtual machines. Tick or untick the check box to activate and deactivate Smart card authentication for individual virtual machines.

Field Name	Description
`Disable strict user checking`	Click the **Advanced Parameters** arrow and select the check box to use this option. With this option selected, the virtual machine does not need to be rebooted when a different user connects to it.
	By default, strict checking is enabled so that only one user can connect to the console of a virtual machine. No other user is able to open a console to the same virtual machine until it has been rebooted. The exception is that a **SuperUser** can connect at any time and replace a existing connection. When a **SuperUser** has connected, no normal user can connect again until the virtual machine is rebooted.
	Disable strict checking with caution, because you can expose the previous user's session to the new user.
`Soundcard Enabled`	A sound card device is not necessary for all virtual machine use cases. If it is for yours, enable a sound card here.
`Enable VirtIO serial console`	The VirtIO serial console is emulated through VirtIO channels, using SSH and key pairs, and allows you to access a virtual machine's serial console directly from a client machine's command line, instead of opening a console from the Administration Portal or the User Portal. The serial console requires direct access to the Manager, since the Manager acts as a proxy for the connection, provides information about virtual machine placement, and stores the authentication keys. Select the check box to enable the VirtIO console on the virtual machine.
`Enable SPICE file transfer`	Defines whether a user is able to drag and drop files from an external host into the virtual machine's SPICE console. This option is only available for virtual machines using the SPICE protocol. This check box is selected by default.
`Enable SPICE clipboard copy and paste`	Defines whether a user is able to copy and paste content from an external host into the virtual machine's SPICE console. This option is only available for virtual machines using the SPICE protocol. This check box is selected by default.

A.1.5. Virtual Machine Host Settings Explained

The following table details the options available on the **Host** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows.

Table A.5. Virtual Machine: Host Settings

Field Name	Sub-element	Description
`Start Running On`		Defines the preferred host on which the virtual machine is to run. Select either: » **`Any Host in Cluster`** - The virtual machine can start and run on any available host in the cluster. » **`Specific`** - The virtual machine will start running on a particular host in the cluster. However, the Manager or an administrator can migrate the virtual machine to a different host in the cluster depending on the migration and high-availability settings of the virtual machine. Select the specific host or group of hosts from the list of available hosts.
`Migration Options`	`Migration mode`	Defines options to run and migrate the virtual machine. If the options here are not used, the virtual machine will run or migrate according to its cluster's policy. » **`Allow manual and automatic migration`** - The virtual machine can be automatically migrated from one host to another in accordance with the status of the environment, or manually by an administrator. » **`Allow manual migration only`** - The virtual machine can only be migrated from one host to another manually by an administrator. » **`Do not allow migration`** - The virtual machine cannot be migrated, either automatically or manually.

Field Name	Sub-element	Description
	Use custom migration policy	Defines the migration convergence policy. If the check box is left unselected, the host determines the policy. » **Legacy** - Legacy behavior of 3.6 version. Overrides in **vdsm.conf** are still applied. The guest agent hook mechanism is disabled. » **Minimal downtime** - Allows the virtual machine to migrate in typical situations. Virtual machines should not experience any significant downtime. The migration will be aborted if virtual machine migration does not converge after a long time (dependent on QEMU iterations, with a maximum of 500 milliseconds). The guest agent hook mechanism is enabled. » **Suspend workload if needed** - Allows the virtual machine to migrate in most situations, including when the virtual machine is running a heavy workload. Virtual machines may experience a more significant downtime. The migration may still be aborted for extreme workloads. The guest agent hook mechanism is enabled.

Field Name	Sub-element	Description
	Use custom migration downtime	This check box allows you to specify the maximum number of milliseconds the virtual machine can be down during live migration. Configure different maximum downtimes for each virtual machine according to its workload and SLA requirements. Enter **0** to use the VDSM default value.
	Auto Converge migrations	Only activated with the **Legacy** migration policy. Allows you to set whether auto-convergence is used during live migration of the virtual machine. Large virtual machines with high workloads can dirty memory more quickly than the transfer rate achieved during live migration, and prevent the migration from converging. Auto-convergence capabilities in QEMU allow you to force convergence of virtual machine migrations. QEMU automatically detects a lack of convergence and triggers a throttle-down of the vCPUs on the virtual machine. Auto-convergence is disabled globally by default. ≫ Select **Inherit from cluster setting** to use the auto-convergence setting that is set at the cluster level. This option is selected by default. ≫ Select **Auto Converge** to override the cluster setting or global setting and allow auto-convergence for the virtual machine. ≫ Select **Don't Auto Converge** to override the cluster setting or global setting and prevent auto-convergence for the virtual machine.

Field Name	Sub-element	Description
	Enable migration compression	Only activated with the **Legacy** migration policy. The option allows you to set whether migration compression is used during live migration of the virtual machine. This feature uses Xor Binary Zero Run-Length-Encoding to reduce virtual machine downtime and total live migration time for virtual machines running memory write-intensive workloads or for any application with a sparse memory update pattern. Migration compression is disabled globally by default. » Select **Inherit from cluster setting** to use the compression setting that is set at the cluster level. This option is selected by default. » Select **Compress** to override the cluster setting or global setting and allow compression for the virtual machine. » Select **Don't compress** to override the cluster setting or global setting and prevent compression for the virtual machine.
	Pass-Through Host CPU	This check box allows virtual machines to take advantage of the features of the physical CPU of the host on which they are situated. This option can only be enabled when **Do not allow migration** is selected.
Configure NUMA	**NUMA Node Count**	The number of virtual NUMA nodes to assign to the virtual machine. If the **Tune Mode** is **Preferred**, this value must be set to **1**.

Field Name	Sub-element	Description
	`Tune Mode`	The method used to allocate memory. » **Strict**: Memory allocation will fail if the memory cannot be allocated on the target node. » **Preferred**: Memory is allocated from a single preferred node. If sufficient memory is not available, memory can be allocated from other nodes. » **Interleave**: Memory is allocated across nodes in a round-robin algorithm.
	`NUMA Pinning`	Opens the **NUMA Topology** window. This window shows the host's total CPUs, memory, and NUMA nodes, and the virtual machine's virtual NUMA nodes. Pin virtual NUMA nodes to host NUMA nodes by clicking and dragging each vNUMA from the box on the right to a NUMA node on the left.

A.1.6. Virtual Machine High Availability Settings Explained

The following table details the options available on the **High Availability** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows.

Table A.6. Virtual Machine: High Availability Settings

Field Name	Description

Field Name	Description
Highly Available	Select this check box if the virtual machine is to be highly available. For example, in cases of host maintenance, all virtual machines are automatically live migrated to another host. If the host crashed and is in a non-responsive state, only virtual machines with high availability are restarted on another host. If the host is manually shut down by the system administrator, the virtual machine is not automatically live migrated to another host. Note that this option is unavailable if the **Migration Options** setting in the **Hosts** tab is set to either **Allow manual migration only** or **Do not allow migration**. For a virtual machine to be highly available, it must be possible for the Manager to migrate the virtual machine to other available hosts as necessary.
Priority for Run/Migration queue	Sets the priority level for the virtual machine to be migrated or restarted on another host.

Field Name	Description
`Watchdog`	Allows users to attach a watchdog card to a virtual machine. A watchdog is a timer that is used to automatically detect and recover from failures. Once set, a watchdog timer continually counts down to zero while the system is in operation, and is periodically restarted by the system to prevent it from reaching zero. If the timer reaches zero, it signifies that the system has been unable to reset the timer and is therefore experiencing a failure. Corrective actions are then taken to address the failure. This functionality is especially useful for servers that demand high availability. `Watchdog Model`: The model of watchdog card to assign to the virtual machine. At current, the only supported model is `i6300esb`. `Watchdog Action`: The action to take if the watchdog timer reaches zero. The following actions are available: » **none** - No action is taken. However, the watchdog event is recorded in the audit log. » **reset** - The virtual machine is reset and the Manager is notified of the reset action. » **poweroff** - The virtual machine is immediately shut down. » **dump** - A dump is performed and the virtual machine is paused. » **pause** - The virtual machine is paused, and can be resumed by users.

A.1.7. Virtual Machine Resource Allocation Settings Explained

The following table details the options available on the `Resource Allocation` tab of the `New Virtual Machine` and `Edit Virtual Machine` windows.

Table A.7. Virtual Machine: Resource Allocation Settings

Field Name	Sub-element	Description

Field Name	Sub-element	Description
CPU Allocation	CPU Profile	The CPU profile assigned to the virtual machine. CPU profiles define the maximum amount of processing capability a virtual machine can access on the host on which it runs, expressed as a percent of the total processing capability available to that host. CPU profiles are defined on the cluster level based on quality of service entries created for data centers.
	CPU Shares	Allows users to set the level of CPU resources a virtual machine can demand relative to other virtual machines. » **Low** - 512 » **Medium** - 1024 » **High** - 2048 » **Custom** - A custom level of CPU shares defined by the user.
	CPU Pinning topology	Enables the virtual machine's virtual CPU (vCPU) to run on a specific physical CPU (pCPU) in a specific host. The syntax of CPU pinning is **v#p[_v#p]**, for example: » **0#0** - Pins vCPU 0 to pCPU 0. » **0#0_1#3** - Pins vCPU 0 to pCPU 0, and pins vCPU 1 to pCPU 3. » **1#1-4,^2** - Pins vCPU 1 to one of the pCPUs in the range of 1 to 4, excluding pCPU 2. In order to pin a virtual machine to a host, you must also select the following on the **Host** tab: » **Start Running On: Specific** » **Migration Options: Do not allow migration** » **Pass-Through Host CPU**

Field Name	Sub-element	Description
Memory Allocation		The amount of physical memory guaranteed for this virtual machine.
IO Threads	**IO Threads Enabled**	Enables virtio-blk data plane. Select this check box to improve the speed of disks that have a VirtIO interface by pinning them to a thread separate from the virtual machine's other functions. Improved disk performance increases a virtual machine's overall performance. Disks with VirtIO interfaces are pinned to an IO thread using a round-robin algorithm.
	Num Of IO Threads	Optionally enter a number value to create multiple IO threads, up to a maximum value of 127. The default value is 1.
Storage Allocation		The **Storage Allocation** option is only available when the virtual machine is created from a template.
	Thin	Provides optimized usage of storage capacity. Disk space is allocated only as it is required. When selected, the format of the disks will be marked as QCOW2 and you will not be able to change it.
	Clone	Optimized for the speed of guest read and write operations. All disk space requested in the template is allocated at the time of the clone operation. When selected, you can select either QCOW2 or RAW as the disk format.
	VirtIO-SCSI Enabled	Allows users to enable or disable the use of VirtIO-SCSI on the virtual machines.
Disk Allocation		The **Disk Allocation** option is only available when the virtual machine is created from a template.
	Alias	An alias for the virtual machine disk. By default, the alias is set to the same value as that of the template.

Field Name	Sub-element	Description
	Virtual Size	The total amount of disk space that the virtual machine based on the template can use. This value cannot be edited, and is provided for reference only.
	Format	The format of the virtual machine disk. The available options are QCOW2 and Raw. If **Thin** is selected in the **Storage Allocation** section, then QCOW2 will be automatically selected and cannot be changed.
	Target	The storage domain on which the virtual disk is stored. By default, the storage domain is set to the same value as that of the template.
	Disk Profile	The disk profile to assign to the virtual machine disk. Disk profiles are created based on storage profiles defined in the data centers.

A.1.8. Virtual Machine Boot Options Settings Explained

The following table details the options available on the **Boot Options** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows

Table A.8. Virtual Machine: Boot Options Settings

Field Name	Description
First Device	After installing a new virtual machine, the new virtual machine must go into Boot mode before powering up. Select the first device that the virtual machine must try to boot: » **Hard Disk** » **CD-ROM** » **Network (PXE)**
Second Device	Select the second device for the virtual machine to use to boot if the first device is not available. The first device selected in the previous option does not appear in the options.
Attach CD	If you have selected **CD-ROM** as a boot device, tick this check box and select a CD-ROM image from the drop-down menu. The images must be available in the ISO domain.

A.1.9. Virtual Machine Random Generator Settings Explained

The following table details the options available on the **Random Generator** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows.

Table A.9. Virtual Machine: Random Generator Settings

Field Name	Description
Random Generator enabled	Selecting this check box enables a paravirtualized Random Number Generator PCI device (virtio-rng). This device allows entropy to be passed from the host to the virtual machine in order to generate a more sophisticated random number. Note that this check box can only be selected if the RNG device exists on the host and is enabled in the host's cluster.
Period duration (ms)	Specifies the duration of a period in milliseconds. If omitted, the libvirt default of 1000 milliseconds (1 second) is used. If this field is filled, **Bytes per period** must be filled also.
Bytes per period	Specifies how many bytes are permitted to be consumed per period.
Device source:	The source of the random number generator. This is automatically selected depending on the source supported by the host's cluster. ≫ **/dev/random source** - The Linux-provided random number generator. ≫ **/dev/hwrng source** - An external hardware generator.

> ⭐ **Important**
>
> This feature is only supported with a host running Red Hat Enterprise Linux 6.6 and later or Red Hat Enterprise Linux 7.0 and later.

A.1.10. Virtual Machine Custom Properties Settings Explained

The following table details the options available on the **Custom Properties** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows.

Table A.10. Virtual Machine: Custom Properties Settings

Field Name	Description	Recommendations and Limitations
sap_agent	Enables SAP monitoring on the virtual machine. Set to **true** or **false**.	-

Field Name	Description	Recommendations and Limitations
sndbuf	Enter the size of the buffer for sending the virtual machine's outgoing data over the socket. Default value is 0.	-
vhost	Disables vhost-net, which is the kernel-based virtio network driver on virtual network interface cards attached to the virtual machine. To disable vhost, the format for this property is: *LogicalNetworkName*: false This will explicitly start the virtual machine without the vhost-net setting on the virtual NIC attached to *LogicalNetworkName*.	vhost-net provides better performance than virtio-net, and if it is present, it is enabled on all virtual machine NICs by default. Disabling this property makes it easier to isolate and diagnose performance issues, or to debug vhost-net errors; for example, if migration fails for virtual machines on which vhost does not exist.
viodiskcache	Caching mode for the virtio disk. **writethrough** writes data to the cache and the disk in parallel, **writeback** does not copy modifications from the cache to the disk, and **none** disables caching. See https://access.redhat.com/solutions/2361311 for more information about the limitations of the **viodiskcache** custom property.	If viodiskcache is enabled, the virtual machine cannot be live migrated.

> ⚠️ **Warning**
>
> Increasing the value of the sndbuf custom property results in increased occurrences of communication failure between hosts and unresponsive virtual machines.

A.1.11. Virtual Machine Icon Settings Explained

You can add custom icons to virtual machines and templates. Custom icons can help to differentiate virtual machines in the User Portal. The following table details the options available on the **Icon** tab of the **New Virtual Machine** and **Edit Virtual Machine** windows.

Table A.11. Virtual Machine: Icon Settings

Button Name	Description
Upload	Click this button to select a custom image to use as the virtual machine's icon. The following limitations apply: » Supported formats: jpg, png, gif » Maximum size: 24 KB » Maximum dimensions: 150px width, 120px height
Use default	Click this button to set the operating system's default image as the virtual machine's icon.

A.2. Explanation of Settings in the New Network Interface and Edit Network Interface Windows

These settings apply when you are adding or editing a virtual machine network interface. If you have more than one network interface attached to a virtual machine, you can put the virtual machine on more than one logical network.

Table A.12. Network Interface Settings

Field Name	Description
Name	The name of the network interface. This text field has a 21-character limit and must be a unique name with any combination of uppercase and lowercase letters, numbers, hyphens, and underscores.
Profile	Logical network that the network interface is placed on. By default, all network interfaces are put on the **ovirtmgmt** management network.
Type	The virtual interface the network interface presents to virtual machines. VirtIO is faster but requires VirtIO drivers. Red Hat Enterprise Linux 5 and higher includes VirtIO drivers. Windows does not include VirtIO drivers, but they can be installed from the guest tools ISO or virtual floppy disk. rtl8139 and e1000 device drivers are included in most operating systems.
Custom MAC address	Choose this option to set a custom MAC address. The Red Hat Virtualization Manager automatically generates a MAC address that is unique to the environment to identify the network interface. Having two devices with the same MAC address online in the same network causes networking conflicts.

Field Name	Description

`Link State`	Whether or not the network interface is connected to the logical network.
	
	≫ **Up**: The network interface is located on its slot.
	  ▫ When the **Card Status** is **Plugged**, it means the network interface is connected to a network cable, and is active.
	  ▫ When the **Card Status** is **Unplugged**, the network interface will be automatically connected to the network and become active.
	≫ **Down**: The network interface is located on its slot, but it is not connected to any network. Virtual machines will not be able to run in this state.

`Card Status`	Whether or not the network interface is defined on the virtual machine.
	
	≫ **Plugged**: The network interface has been defined on the virtual machine.
	  ▫ If its **Link State** is **Up**, it means the network interface is connected to a network cable, and is active.
	  ▫ If its **Link State** is **Down**, the network interface is not connected to a network cable.
	≫ **Unplugged**: The network interface is only defined on the Manager, and is not associated with a virtual machine.
	  ▫ If its **Link State** is **Up**, when the network interface is plugged it will automatically be connected to a network and become active.
	  ▫ If its **Link State** is **Down**, the network interface is not connected to any network until it is defined on a virtual machine.

A.3. Explanation of Settings in the New Virtual Disk and Edit Virtual Disk Windows

Table A.13. New Virtual Disk and Edit Virtual Disk Settings: `Image`

Field Name	Description
`Size(GB)`	The size of the new virtual disk in GB.
`Alias`	The name of the virtual disk, limited to 40 characters.

Field Name	Description
`Description`	A description of the virtual disk. This field is recommended but not mandatory.
`Interface`	The virtual interface the disk presents to virtual machines. **VirtIO** is faster, but requires drivers. Red Hat Enterprise Linux 5 and higher include these drivers. Windows does not include these drivers, but they can be installed from the guest tools ISO or virtual floppy disk. **IDE** devices do not require special drivers. The interface type can be updated after stopping all virtual machines that the disk is attached to.
`Data Center`	The data center in which the virtual disk will be available.
`Storage Domain`	The storage domain in which the virtual disk will be stored. The drop-down list shows all storage domains available in the given data center, and also shows the total space and currently available space in the storage domain.
`Allocation Policy`	The provisioning policy for the new virtual disk. » **Preallocated** allocates the entire size of the disk on the storage domain at the time the virtual disk is created. The virtual size and the actual size of a preallocated disk are the same. Preallocated virtual disks take more time to create than thinly provisioned virtual disks, but have better read and write performance. Preallocated virtual disks are recommended for servers and other I/O intensive virtual machines. If a virtual machine is able to write more than 1 GB every four seconds, use preallocated disks where possible. » **Thin Provision** allocates 1 GB at the time the virtual disk is created and sets a maximum limit on the size to which the disk can grow. The virtual size of the disk is the maximum limit; the actual size of the disk is the space that has been allocated so far. Thinly provisioned disks are faster to create than preallocated disks and allow for storage over-commitment. Thinly provisioned virtual disks are recommended for desktops.

Field Name	Description
Disk Profile	The disk profile assigned to the virtual disk. Disk profiles define the maximum amount of throughput and the maximum level of input and output operations for a virtual disk in a storage domain. Disk profiles are defined on the storage domain level based on storage quality of service entries created for data centers.
Activate Disk(s)	Activate the virtual disk immediately after creation. This option is not available when creating a floating disk.
Wipe After Delete	Allows you to enable enhanced security for deletion of sensitive material when the virtual disk is deleted.
Bootable	Allows you to enable the bootable flag on the virtual disk.
Shareable	Allows you to attach the virtual disk to more than one virtual machine at a time.
Read Only	Allows you to set the disk as read-only. The same disk can be attached as read-only to one virtual machine, and as rewritable to another. This option is not available when creating a floating disk.

The **Direct LUN** settings can be displayed in either **Targets > LUNs** or **LUNs > Targets**. **Targets > LUNs** sorts available LUNs according to the host on which they are discovered, whereas **LUNs > Targets** displays a single list of LUNs.

Table A.14. New Virtual Disk and Edit Virtual Disk Settings: Direct LUN

Field Name	Description
Alias	The name of the virtual disk, limited to 40 characters.
Description	A description of the virtual disk. This field is recommended but not mandatory. By default the last 4 characters of the LUN ID is inserted into the field. The default behavior can be configured by setting the **PopulateDirectLUNDiskDescriptionWithLUNId** configuration key to the appropriate value using the **engine-config** command. The configuration key can be set to **-1** for the full LUN ID to be used, or **0** for this feature to be ignored. A positive integer populates the description with the corresponding number of characters of the LUN ID.

Field Name	Description
Interface	The virtual interface the disk presents to virtual machines. **VirtIO** is faster, but requires drivers. Red Hat Enterprise Linux 5 and higher include these drivers. Windows does not include these drivers, but they can be installed from the guest tools ISO or virtual floppy disk. **IDE** devices do not require special drivers. The interface type can be updated after stopping all virtual machines that the disk is attached to.
Data Center	The data center in which the virtual disk will be available.
Use Host	The host on which the LUN will be mounted. You can select any host in the data center.
Storage Type	The type of external LUN to add. You can select from either **iSCSI** or **Fibre Channel**.
Discover Targets	This section can be expanded when you are using iSCSI external LUNs and **Targets > LUNs** is selected. **Address** - The host name or IP address of the target server. **Port** - The port by which to attempt a connection to the target server. The default port is 3260. **User Authentication** - The iSCSI server requires User Authentication. The **User Authentication** field is visible when you are using iSCSI external LUNs. **CHAP user name** - The user name of a user with permission to log in to LUNs. This field is accessible when the **User Authentication** check box is selected. **CHAP password** - The password of a user with permission to log in to LUNs. This field is accessible when the **User Authentication** check box is selected.
Activate Disk(s)	Activate the virtual disk immediately after creation. This option is not available when creating a floating disk.
Bootable	Allows you to enable the bootable flag on the virtual disk.
Shareable	Allows you to attach the virtual disk to more than one virtual machine at a time.

Field Name	Description
`Read Only`	Allows you to set the disk as read-only. The same disk can be attached as read-only to one virtual machine, and as rewritable to another. This option is not available when creating a floating disk.
`Enable SCSI Pass-Through`	Available when the **Interface** is set to **VirtIO-SCSI**. Selecting this check box enables passthrough of a physical SCSI device to the virtual disk. A VirtIO-SCSI interface with SCSI passthrough enabled automatically includes SCSI discard support. **Read Only** is not supported when this check box is selected. When this check box is not selected, the virtual disk uses an emulated SCSI device. **Read Only** is supported on emulated VirtIO-SCSI disks.
`Allow Privileged SCSI I/O`	Available when the **Enable SCSI Pass-Through** check box is selected. Selecting this check box enables unfiltered SCSI Generic I/O (SG_IO) access, allowing privileged SG_IO commands on the disk. This is required for persistent reservations.
`Using SCSI Reservation`	Available when the **Enable SCSI Pass-Through** and **Allow Privileged SCSI I/O** check boxes are selected. Selecting this check box disables migration for any virtual machine using this disk, to prevent virtual machines that are using SCSI reservation from losing access to the disk.

Fill in the fields in the **Discover Targets** section and click **Discover** to discover the target server. You can then click the **Login All** button to list the available LUNs on the target server and, using the radio buttons next to each LUN, select the LUN to add.

Using LUNs directly as virtual machine hard disk images removes a layer of abstraction between your virtual machines and their data.

The following considerations must be made when using a direct LUN as a virtual machine hard disk image:

 ≫ Live storage migration of direct LUN hard disk images is not supported.

 ≫ Direct LUN disks are not included in virtual machine exports.

 ≫ Direct LUN disks are not included in virtual machine snapshots.

The **Cinder** settings form will be disabled if there are no available OpenStack Volume storage domains on which you have permissions to create a disk in the relevant Data Center. **Cinder** disks require access to an instance of OpenStack Volume that has been added to the Red Hat Virtualization environment using the **External Providers** window; see Adding an OpenStack Volume (Cinder) Instance for Storage Management for more information.

Table A.15. New Virtual Disk and Edit Virtual Disk Settings: Cinder

Field Name	Description
Size(GB)	The size of the new virtual disk in GB.
Alias	The name of the virtual disk, limited to 40 characters.
Description	A description of the virtual disk. This field is recommended but not mandatory.
Interface	The virtual interface the disk presents to virtual machines. **VirtIO** is faster, but requires drivers. Red Hat Enterprise Linux 5 and higher include these drivers. Windows does not include these drivers, but they can be installed from the guest tools ISO or virtual floppy disk. **IDE** devices do not require special drivers. The interface type can be updated after stopping all virtual machines that the disk is attached to.
Data Center	The data center in which the virtual disk will be available.
Storage Domain	The storage domain in which the virtual disk will be stored. The drop-down list shows all storage domains available in the given data center, and also shows the total space and currently available space in the storage domain.
Volume Type	The volume type of the virtual disk. The drop-down list shows all available volume types. The volume type will be managed and configured on OpenStack Cinder.
Activate Disk(s)	Activate the virtual disk immediately after creation. This option is not available when creating a floating disk.
Bootable	Allows you to enable the bootable flag on the virtual disk.
Shareable	Allows you to attach the virtual disk to more than one virtual machine at a time.
Read Only	Allows you to set the disk as read-only. The same disk can be attached as read-only to one virtual machine, and as rewritable to another. This option is not available when creating a floating disk.

> **Important**
>
> Mounting a journaled file system requires read-write access. Using the **Read Only** option is not appropriate for virtual machine disks that contain such file systems (e.g. **EXT3**, **EXT4**, or **XFS**).

A.4. Explanation of Settings in the New Template and Edit Template Windows

The following table details the settings for the **New Template** and **Edit Template** windows.

Table A.16. New Template and Edit Template Settings

Field	Description/Action
Name	The name of the template. This is the name by which the template is listed in the **Templates** tab in the Administration Portal and is accessed via the REST API. This text field has a 40-character limit and must be a unique name within the data center with any combination of uppercase and lowercase letters, numbers, hyphens, and underscores. The name can be re-used in different data centers in the environment.
Description	A description of the template. This field is recommended but not mandatory.
Comment	A field for adding plain text, human-readable comments regarding the template.
Cluster	The cluster with which the template is associated. This is the same as the original virtual machines by default. You can select any cluster in the data center.
CPU Profile	The CPU profile assigned to the template. CPU profiles define the maximum amount of processing capability a virtual machine can access on the host on which it runs, expressed as a percent of the total processing capability available to that host. CPU profiles are defined on the cluster level based on quality of service entries created for data centers.
Create as a Sub Template version	Specifies whether the template is created as a new version of an existing template. Select this check box to access the settings for configuring this option. ≫ **Root Template**: The template under which the sub template is added. ≫ **Sub Version Name**: The name of the template. This is the name by which the template is accessed when creating a new virtual machine based on the template.

Field	Description/Action
`Disks Allocation`	`Alias` - An alias for the virtual machine disk used by the template. By default, the alias is set to the same value as that of the source virtual machine. `Virtual Size` - The total amount of disk space that a virtual machine based on the template can use. This value cannot be edited, and is provided for reference only. This value corresponds with the size, in GB, that was specified when the disk was created or edited. `Format` - The format of the virtual machine disk used by the template. The available options are QCOW2 and Raw. By default, the format is set to Raw. `Target` - The storage domain on which the virtual disk used by the template is stored. By default, the storage domain is set to the same value as that of the source virtual machine. You can select any storage domain in the cluster. `Disk Profile` - The disk profile to assign to the virtual machine disk used by the template. Disk profiles are created based on storage profiles defined in the data centers.
`Allow all users to access this Template`	Specifies whether a template is public or private. A public template can be accessed by all users, whereas a private template can only be accessed by users with the `TemplateAdmin` or `SuperUser` roles.
`Copy VM permissions`	Copies explicit permissions that have been set on the source virtual machine to the template.

A.5. Explanation of Settings in the Run Once Window

The `Run Once` window defines one-off boot options for a virtual machine. For persistent boot options, use the `Boot Options` tab in the `New Virtual Machine` window. The `Run Once` window contains multiple sections that can be configured.

The `Boot Options` section defines the virtual machine's boot sequence, running options, and source images for installing the operating system and required drivers.

Table A.17. Boot Options Section

Field Name	Description

Field Name	Description
Attach Floppy	Attaches a diskette image to the virtual machine. Use this option to install Windows drivers. The diskette image must reside in the ISO domain.
Attach CD	Attaches an ISO image to the virtual machine. Use this option to install the virtual machine's operating system and applications. The CD image must reside in the ISO domain.
Boot Sequence	Determines the order in which the boot devices are used to boot the virtual machine. Select either **Hard Disk**, **CD-ROM** or **Network**, and use **Up** and **Down** to move the option up or down in the list.
Run Stateless	Deletes all changes to the virtual machine upon shutdown. This option is only available if a virtual disk is attached to the virtual machine.
Start in Pause Mode	Starts then pauses the virtual machine to enable connection to the console, suitable for virtual machines in remote locations.

The **Linux Boot Options** section contains fields to boot a Linux kernel directly instead of through the BIOS bootloader.

Table A.18. Linux Boot Options Section

Field Name	Description
kernel path	A fully qualified path to a kernel image to boot the virtual machine. The kernel image must be stored on either the ISO domain (path name in the format of **iso://path-to-image**) or on the host's local storage domain (path name in the format of **/data/images**).
initrd path	A fully qualified path to a ramdisk image to be used with the previously specified kernel. The ramdisk image must be stored on the ISO domain (path name in the format of **iso://path-to-image**) or on the host's local storage domain (path name in the format of **/data/images**).
kernel parameters	Kernel command line parameter strings to be used with the defined kernel on boot.

The **Initial Run** section is used to specify whether to use Cloud-Init or Sysprep to initialize the virtual machine. For Linux-based virtual machines, you must select the **Use Cloud-Init** check box in the **Initial Run** tab to view the available options. For Windows-based virtual machines, you must attach the **[sysprep]** floppy by selecting the **Attach Floppy** check box in the **Boot Options** tab and selecting the floppy from the list.

The options that are available in the **Initial Run** section differ depending on the operating system that the virtual machine is based on.

Table A.19. Initial Run Section (Linux-based Virtual Machines)

Field Name	Description
`VM Hostname`	The host name of the virtual machine.
`Configure Time Zone`	The time zone for the virtual machine. Select this check box and select a time zone from the **Time Zone** list.
`Authentication`	The authentication details for the virtual machine. Click the disclosure arrow to display the settings for this option.
`Authentication > User Name`	Creates a new user account on the virtual machine. If this field is not filled in, the default user is **root**.
`Authentication > Use already configured password`	This check box is automatically selected after you specify an initial root password. You must clear this check box to enable the **Password** and **Verify Password** fields and specify a new password.
`Authentication > Password`	The root password for the virtual machine. Enter the password in this text field and the **Verify Password** text field to verify the password.
`Authentication > SSH Authorized Keys`	SSH keys to be added to the authorized keys file of the virtual machine.
`Authentication > Regenerate SSH Keys`	Regenerates SSH keys for the virtual machine.
`Networks`	Network-related settings for the virtual machine. Click the disclosure arrow to display the settings for this option.
`Networks > DNS Servers`	The DNS servers to be used by the virtual machine.
`Networks > DNS Search Domains`	The DNS search domains to be used by the virtual machine.
`Networks > Network`	Configures network interfaces for the virtual machine. Select this check box and click **+** or **-** to add or remove network interfaces to or from the virtual machine. When you click **+**, a set of fields becomes visible that can specify whether to use DHCP, and configure an IP address, netmask, and gateway, and specify whether the network interface will start on boot.
`Custom Script`	Custom scripts that will be run on the virtual machine when it starts. The scripts entered in this field are custom YAML sections that are added to those produced by the Manager, and allow you to automate tasks such as creating users and files, configuring **yum** repositories and running commands. For more information on the format of scripts that can be entered in this field, see the Custom Script documentation.

Table A.20. Initial Run Section (Windows-based Virtual Machines)

Field Name	Description
VM Hostname	The host name of the virtual machine.
Domain	The Active Directory domain to which the virtual machine belongs.
Organization Name	The name of the organization to which the virtual machine belongs. This option corresponds to the text field for setting the organization name displayed when a machine running Windows is started for the first time.
Active Directory OU	The organizational unit in the Active Directory domain to which the virtual machine belongs. The distinguished name must be provided. For example **CN=Users,DC=lab,DC=local**
Configure Time Zone	The time zone for the virtual machine. Select this check box and select a time zone from the **Time Zone** list.
Admin Password	The administrative user password for the virtual machine. Click the disclosure arrow to display the settings for this option.
Admin Password > Use already configured password	This check box is automatically selected after you specify an initial administrative user password. You must clear this check box to enable the **Admin Password** and **Verify Admin Password** fields and specify a new password.
Admin Password > Admin Password	The administrative user password for the virtual machine. Enter the password in this text field and the **Verify Admin Password** text field to verify the password.
Custom Locale	Locales must be in a format such as **en-US**. Click the disclosure arrow to display the settings for this option.
Custom Locale > Input Locale	The locale for user input.
Custom Locale > UI Language	The language used for user interface elements such as buttons and menus.
Custom Locale > System Locale	The locale for the overall system.
Custom Locale > User Locale	The locale for users.
Sysprep	A custom Sysprep definition. The definition must be in the format of a complete unattended installation answer file. You can copy and paste the default answer files in the **/usr/share/ovirt-engine/conf/sysprep/** directory on the machine on which the Red Hat Virtualization Manager is installed and alter the fields as required. The definition will overwrite any values entered in the **Initial Run** fields.
Domain	The Active Directory domain to which the virtual machine belongs. If left blank, the value of the previous **Domain** field is used.
Alternate Credentials	Selecting this check box allows you to set a **User Name** and **Password** as alternative credentials.

The **System** section enables you to define the supported machine type or CPU type.

Table A.21. System Section

Field Name	Description
`Custom Emulated Machine`	This option allows you to specify the machine type. If changed, the virtual machine will only run on hosts that support this machine type. Defaults to the cluster's default machine type.
`Custom CPU Type`	This option allows you to specify a CPU type. If changed, the virtual machine will only run on hosts that support this CPU type. Defaults to the cluster's default CPU type.

The **Host** section is used to define the virtual machine's host.

Table A.22. Host Section

Field Name	Description
`Any host in cluster`	Allocates the virtual machine to any available host.
`Specific`	Specifies a user-defined host for the virtual machine.

The **Console** section defines the protocol to connect to virtual machines.

Table A.23. Console Section

Field Name	Description
`VNC`	Requires a VNC client to connect to a virtual machine using VNC. Optionally, specify **VNC Keyboard Layout** from the drop-down list.
`SPICE`	Recommended protocol for Linux and Windows virtual machines. Using SPICE protocol without QXL drivers is supported for Windows 8 and Server 2012 virtual machines; however, support for multiple monitors and graphics acceleration is not available for this configuration.

The **Custom Properties** section contains additional VDSM options for running virtual machines.

Table A.24. Custom Properties Section

Field Name	Description
`sap_agent`	Enables SAP monitoring on the virtual machine. Set to **true** or **false**.
`sndbuf`	Enter the size of the buffer for sending the virtual machine's outgoing data over the socket.

Field Name	Description
vhost	Enter the name of the virtual host on which this virtual machine should run. The name can contain any combination of letters and numbers.
viodiskcache	Caching mode for the virtio disk. **writethrough** writes data to the cache and the disk in parallel, **writeback** does not copy modifications from the cache to the disk, and **none** disables caching. See https://access.redhat.com/solutions/2361311 for more information about the limitations of the **viodiskcache** custom property.

Field Name	Description