Red Hat
Training and Certification

Student Workbook (ROLE)

Red Hat Virtualization 4.3 RH318

Red Hat Virtualization

Edition 1

Red Hat Virtualization

Red Hat Virtualization 4.3 RH318
Red Hat Virtualization
Edition 1 20201106
Publication date 20190918

Authors: Fiona Allen, Trey Feagle, Saumik Paul, Michael Phillips, Dallas Spohn
Editor: Philip Sweany, Nicole Muller, Dave Sacco

Contributors: Ashish Lingayat, Rob Locke, Rudolf Kastl, Latha Murphy, Sajith Eyamkuzhy Sugathan

Document Conventions

References
"References" describe where to find external documentation relevant to a subject.

Note
"Notes" are tips, shortcuts or alternative approaches to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.

Important
"Important" boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled "Important" will not cause data loss, but may cause irritation and frustration.

Warning
"Warnings" should not be ignored. Ignoring warnings will most likely cause data loss.

Introduction

Red Hat Virtualization

Red Hat Virtualization (RH318) enables IT professionals to acquire the skills needed to deploy, administer, and operate virtual machines in their organization using Red Hat Virtualization. Through numerous hands-on exercises, students will deploy and configure the Red Hat Virtualization infrastructure and use it to provision and manage virtual machines. This course also prepares candidates for the Red Hat Certified Virtualization Administrator (RHCVA) certification exam.

Course Objectives

- Install and use a Red Hat Virtualization Manager server
- Set up physical hosts with Red Hat Virtualization Host to run virtual machines
- Create and manage virtual machines
- Create and manage virtual machine storage on NFS and iSCSI storage servers

Audience

- Linux system administrators and virtualization administrators interested in deploying and managing large-scale virtualization solutions managing virtual servers in their data centers, based on the Red Hat Virtualization open virtualization management platform.

Prerequisites

- Red Hat Certified System Administrator (RHCSA) in Red Hat Enterprise Linux certification or equivalent Linux system administration skills.

Orientation to the Classroom Environment

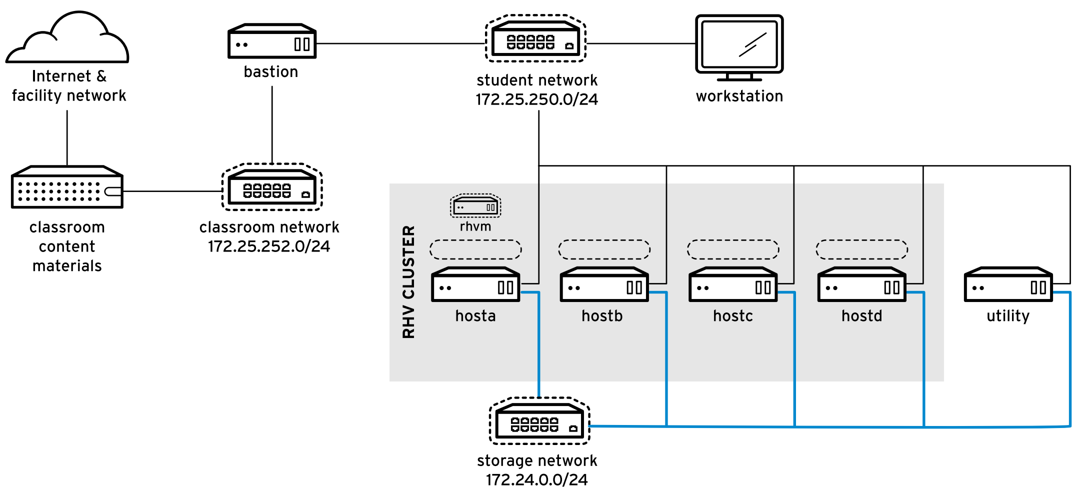

Figure 0.1: Classroom environment

In this course, the main computer system used for hands-on learning activities is **`workstation`**. The additional systems to be used by students for activities are **`hosta`**, **`hostb`**, **`hostc`**, **`rhvm`**, and **`utility`**. All systems are in the **`lab.example.com`** DNS domain.

All student computer systems have a standard user account, **`student`**, with the password **`student`**. The **`root`** password on all student systems is **`redhat`**.

Classroom Machines

Machine name	IP addresses	Role
utility.lab.example.com	172.25.250.8 172.24.0.8	A RHEL system running as a Network File System and a Red Hat Identity Management server
workstation.lab.example.com	172.25.250.9	Graphical workstation for system administration and as the Red Hat Ansible Engine control node
hosta.lab.example.com	172.25.250.10 172.24.0.10	First RHV-H hypervisor host
hostb.lab.example.com	172.25.250.11 172.24.0.11	Second RHV-H hypervisor host
hostc.lab.example.com	172.25.250.12 172.24.0.12	Third RHV-H hypervisor host
hostd.lab.example.com	172.25.250.13 172.24.0.13	Fourth RHV-H hypervisor host

Machine name	IP addresses	Role
rhvm.lab.example.com	172.25.250.14	Self-hosted RHV-M engine (management appliance)
bastion.lab.example.com	172.25.250.254 172.25.252.*X*	Gateway to classroom server
classroom.lab.example.com content.lab.example.com materials.lab.example.com	172.25.252.254 172.25.254.254	Classroom materials server, with multiple aliases

The **bastion** system acts as a router between the classroom and student networks. If **bastion** is down, student systems will only be able to access systems on their student network.

Several systems in the classroom provide supporting services. Two URLs provide software and lab materials used in hands-on activities, **content.example.com** and **materials.example.com**. These URLs are aliases of the **classroom.example.com** system. Information on how to use these materials is provided in the hands-on activities. The **utility** system provides course-specific services. In this course, **utility** is a RHEL Identity Management, Network File System and Red Hat Gluster Storage server.

Controlling Your Station

The top of the console describes the state of your machine.

Machine States

State	Description
none	Your machine has not yet been started. When started, your machine boots into a newly initialized state (the disk is reset).
starting	Your machine is in the process of booting.
running	Your machine is running and available (or, when booting, soon will be.)
stopping	Your machine is in the process of shutting down.
stopped	Your machine is completely shut down. Upon starting, your machine boots into the same state as when it was shut down (the disk is preserved).
impaired	A network connection to your machine cannot be made. Typically this state is reached when a student has corrupted networking or firewall rules. If the condition persists after a machine reset, or is intermittent, then please open a support case.

Depending on the state of your machine, a selection of the following actions is available to you.

Machine Actions

Action	Description
Start Station	Start ("power on") the machine.
Stop Station	Stop ("power off") the machine, preserving the contents of its disk.
Reset Station	Stop ("power off") the machine, resetting the disk to its initial state. **Caution: Any work generated on the disk is lost**.
Refresh	*Refresh* probes the machine state.
Increase Timer	Adds 15 minutes to the timer for each click.

The Station Timer

Your Red Hat Online Learning enrollment entitles you to a certain amount of computer time. In order to help you conserve your time, the machines have an associated timer, initialized to 60 minutes when your machine is started.

The timer operates as a "dead man's switch," which decrements as your machine runs. If the timer is winding down to 0, you may choose to increase the timer.

Internationalization

Language Support

Red Hat Enterprise Linux 7 officially supports 22 languages: English, Assamese, Bengali, Chinese (Simplified), Chinese (Traditional), French, German, Gujarati, Hindi, Italian, Japanese, Kannada, Korean, Malayalam, Marathi, Odia, Portuguese (Brazilian), Punjabi, Russian, Spanish, Tamil, and Telugu.

Per-user Language Selection

Users may prefer to use a different language for their desktop environment than the system-wide default. They may also want to set their account to use a different keyboard layout or input method.

Language Settings

In the GNOME desktop environment, the user may be prompted to set their preferred language and input method on first login. If not, then the easiest way for an individual user to adjust their preferred language and input method settings is to use the Region & Language application. Run the command **gnome-control-center region**, or from the top bar, select **(User)** → **Settings**. In the window that opens, select Region & Language. The user can click the **Language** box and select their preferred language from the list that appears. This also updates the **Formats** setting to the default for that language. The next time the user logs in, these changes take full effect.

These settings affect the GNOME desktop environment and any applications, including **gnome-terminal**, started inside it. However, they do not apply to that account if accessed through an **ssh** login from a remote system or a local text console (such as **tty2**).

Note

A user can make their shell environment use the same **LANG** setting as their graphical environment, even when they log in through a text console or over **ssh**. One way to do this is to place code similar to the following in the user's **~/.bashrc** file. This example code sets the language used on a text login to match the one currently set for the user's GNOME desktop environment:

```
i=$(grep 'Language=' /var/lib/AccountService/users/${USER} \
  | sed 's/Language=//')
if [ "$i" != "" ]; then
    export LANG=$i
fi
```

Japanese, Korean, Chinese, or other languages with a non-Latin character set may not display properly on local text consoles.

Individual commands can be made to use another language by setting the **LANG** variable on the command line:

```
[user@host ~]$ LANG=fr_FR.utf8 date
jeu. avril 24 17:55:01 CDT 2014
```

Subsequent commands revert to using the system's default language for output. The **locale** command can be used to check the current value of **LANG** and other related environment variables.

Input Method Settings

GNOME 3 in Red Hat Enterprise Linux 7 automatically uses the IBus input method selection system, which makes it easy to change keyboard layouts and input methods quickly.

The Region & Language application can also be used to enable alternative input methods. In the Region & Language application's window, the **Input Sources** box shows what input methods are currently available. By default, **English (US)** may be the only available method. Highlight **English (US)** and click the **keyboard** icon to see the current keyboard layout.

To add another input method, click the **+** button at the bottom left of the **Input Sources** window. An **Add an Input Source** window opens. Select your language, and then your preferred input method or keyboard layout.

Once more than one input method is configured, the user can switch between them quickly by typing **Super**+**Space** (sometimes called **Windows**+**Space**). A *status indicator* also appears in the GNOME top bar, which has two functions: it indicates which input method is active, and acts as a menu that can be used to switch between input methods or select advanced features of more complex input methods.

Some of the methods are marked with gears, which indicate that those methods have advanced configuration options and capabilities. For example, the Japanese **Japanese (Kana Kanji)** input method allows the user to pre-edit text in Latin and use **Down Arrow** and **Up Arrow** keys to select the correct characters to use.

US English speakers may find also this useful. For example, under **English (United States)** is the keyboard layout **English (international AltGr dead keys)**, which treats **AltGr** (or the right **Alt**) on a PC 104/105-key keyboard as a "secondary-shift" modifier key and dead key activation key for typing additional characters. There are also Dvorak and other alternative layouts available.

> **Note**
>
> Any Unicode character can be entered in the GNOME desktop environment if the user knows the character's Unicode code point, by typing **Ctrl**+**Shift**+**U**, followed by the code point. After **Ctrl**+**Shift**+**U** has been typed, an underlined **u** will be displayed to indicate that the system is waiting for Unicode code point entry.
>
> For example, the lowercase Greek letter lambda has the code point U+03BB, and can be entered by typing **Ctrl**+**Shift**+**U**, then **03bb**, then **Enter**.

System-wide Default Language Settings

The system's default language is set to US English, using the UTF-8 encoding of Unicode as its character set (**en_US.utf8**), but this can be changed during or after installation.

From the command line, *root* can change the system-wide locale settings with the **localectl** command. If **localectl** is run with no arguments, it displays the current system-wide locale settings.

To set the system-wide language, run the command **`localectl set-locale LANG=`*`locale`***, where *locale* is the appropriate **`$LANG`** from the "Language Codes Reference" table in this chapter. The change takes effect for users on their next login, and is stored in **`/etc/locale.conf`**.

```
[root@host ~]# localectl set-locale LANG=fr_FR.utf8
```

In GNOME, an administrative user can change this setting from Region & Language and clicking the **Login Screen** button at the upper-right corner of the window. Changing the **Language** of the login screen also adjusts the system-wide default language setting stored in the **`/etc/locale.conf`** configuration file.

> **Important**
>
> Local text consoles such as **`tty2`** are more limited in the fonts that they can display than **`gnome-terminal`** and **`ssh`** sessions. For example, Japanese, Korean, and Chinese characters may not display as expected on a local text console. For this reason, it may make sense to use English or another language with a Latin character set for the system's text console.
>
> Likewise, local text consoles are more limited in the input methods they support, and this is managed separately from the graphical desktop environment. The available global input settings can be configured through **`localectl`** for both local text virtual consoles and the X11 graphical environment. See the **`localectl`**(1), **`kbd`**(4), and **`vconsole.conf`**(5) man pages for more information.

Language Packs

When using non-English languages, you may want to install additional "language packs" to provide additional translations, dictionaries, and so forth. To view the list of available langpacks, run **`yum langavailable`**. To view the list of langpacks currently installed on the system, run **`yum langlist`**. To add an additional langpack to the system, run **`yum langinstall`** ***`code`***, where *code* is the code in square brackets after the language name in the output of **`yum langavailable`**.

> **References**
>
> **`locale`**(7), **`localectl`**(1), **`kbd`**(4), **`locale.conf`**(5), **`vconsole.conf`**(5), **`unicode`**(7), **`utf-8`**(7), and **`yum-langpacks`**(8) man pages
>
> Conversions between the names of the graphical desktop environment's X11 layouts and their names in **`localectl`** can be found in the file **`/usr/share/X11/xkb/rules/base.lst`**.

Language Codes Reference

Language Codes

Language	$LANG value
English (US)	en_US.utf8
Assamese	as_IN.utf8

Language	$LANG value
Bengali	bn_IN.utf8
Chinese (Simplified)	zh_CN.utf8
Chinese (Traditional)	zh_TW.utf8
French	fr_FR.utf8
German	de_DE.utf8
Gujarati	gu_IN.utf8
Hindi	hi_IN.utf8
Italian	it_IT.utf8
Japanese	ja_JP.utf8
Kannada	kn_IN.utf8
Korean	ko_KR.utf8
Malayalam	ml_IN.utf8
Marathi	mr_IN.utf8
Odia	or_IN.utf8
Portuguese (Brazilian)	pt_BR.utf8
Punjabi	pa_IN.utf8
Russian	ru_RU.utf8
Spanish	es_ES.utf8
Tamil	ta_IN.utf8
Telugu	te_IN.utf8

Chapter 1

Red Hat Virtualization (RHV) Overview

Goal	Explain the purpose and architecture of Red Hat Virtualization.
Objectives	• Describe the purpose of Red Hat Virtualization and its architectural design.
Sections	Introducing Red Hat Virtualization (and Quiz)

Introducing Red Hat Virtualization

Objectives

After completing this section, you should be able to describe the purpose of Red Hat Virtualization and its architectural design.

Describing Virtualization Concepts

System virtualization allows a single computer to be partitioned into multiple, concurrent virtual computers, each running its own operating system. These virtual machines are fully isolated from each other, as if running on private hardware. Virtual machines have their own network interfaces, IP addresses, file systems, and other peripherals. Each virtual machine can run any supported operating system or version.

Virtualization allows efficient use of physical computing resources. A large system can be partitioned into many virtual systems, allowing it to be used to full capacity. Virtualization allows quick provisioning of new virtual servers which is faster than deploying new physical hardware.

The *hypervisor* is the component that manages and supports the physical server virtualization. This hypervisor runs the virtual machines for each virtualized operating system, providing access to virtual CPUs, memory, disks, networking, and other peripherals, while restricting virtual machines from having direct access to hardware or other virtual machines.

The physical machine and operating system providing the hypervisor is called a *host*. A *guest* is an operating system running in a *virtual machine* on a hypervisor.

A *virtualization management* solution effectively manages virtual machines and hosts running in your infrastructure. Typically, virtualization management tools create an integrated virtualization environment to create, manage, and monitor virtual machines, hypervisors, storage, and networking resources. A virtualization manager provides centralized control for your virtualization infrastructure, rather than requiring you to individually manage and monitor hundreds of hypervisors and their virtual machines.

Cloud Computing

Virtualization provides a basic foundation for cloud computing. Cloud computing uses virtualization technology to provide shared computing resources as on-demand service offerings over the network. A managed virtualization environment is different from a public or private cloud. While both managed virtualization and cloud computing allow the administration of physical resources as multiple virtual resources, cloud computing shares a pool of automatically allocated resources as a service. Cloud resources are available through a self-service model and can also offer the benefits of automated management, scaling, and elasticity, whereas virtualization environments offer enterprise-class private data center management.

Use Cases of Virtualization and Cloud

Managed virtualization and cloud computing each lend themselves to different use cases. One primary determinant for the selection of virtualization over cloud is the expected workload. Virtualization is typically the better platform for workloads requiring finite resources for a specific application available to a defined set of users.

Virtualization and Cloud Comparison

Virtualization	Cloud
Creates multiple virtual environments for individual physical systems.	Shares a pool of automated virtual resources as an on-demand service.
Provides finite resources for a specific use to a defined set of users.	Provides variable resources to groups of users for different purposes.
Characterized by long-term allocation of resources.	Characterized by short-term allocation of resources.
Workloads are stateful.	Workloads are stateless.
High Availability (HA) is handled at the infrastructure level.	High Availability (HA) is handled at the application layer.
Scales up in response to increased workload demands.	Scales out in response to increased workload demands.

Describing Red Hat Virtualization

Red Hat Virtualization (RHV) is an open source virtualization platform for centralized management of hosts, virtual servers, and desktops across one or more enterprise data centers. RHV is based on Red Hat Enterprise Linux (RHEL), Kernel-based Virtual Machine (KVM) technology, and the oVirt virtualization management project. Red Hat Virtualization offers features for enterprise-grade virtualization, such as live migration, high availability, system scheduling, power management, image management, snapshots, thin provisioning, and monitoring.

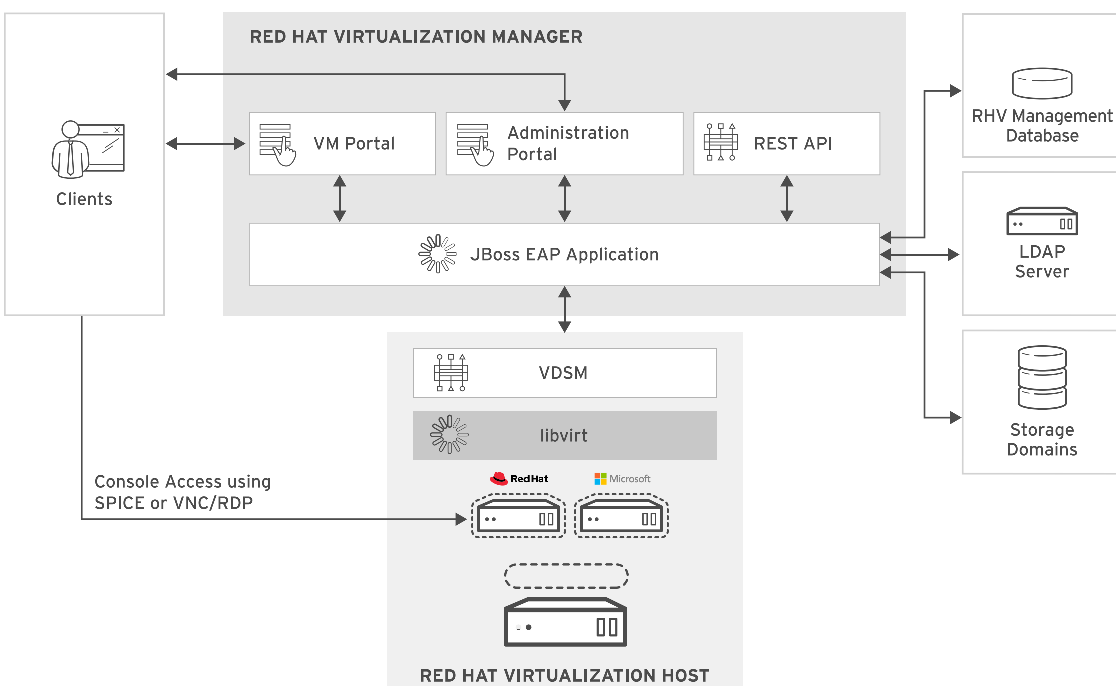

Figure 1.1: Overview of Red Hat Virtualization

Red Hat Virtualization Manager (RHV-M)

The core component of RHV is the *Red Hat Virtualization Manager (RHV-M)*, which provides a central management platform for both physical and virtual resources in a RHV environment. RHV-M installs either on a Red Hat Enterprise Linux server or as a self-hosted appliance, and stores its data in a local or external PostgreSQL database. Administrators can simplify the management of RHV-M user access through integration with supported directory servers, such as RHEL IdM, Active Directory or other supported LDAP server.

Through the use of a graphical user interface and a RESTful API, RHV-M offers the management of virtual machine disk images, installation ISOs, and other storage and network components used in the data center. It allows virtual machines to be started, stopped, created manually or from templates, migrated from one physical hypervisor node to another, and more. RHV-M provides access to the graphical consoles of virtual machines using the SPICE protocol, providing a desktop experience similar to running on a user's local physical system.

Red Hat Virtualization supports two types of deployment for RHV-M, as a *Standalone Manager* or as a *Self-Hosted Engine*.

In a Standalone Manager deployment, RHV-M is installed on either a physical system, or as a virtual machine hosted in another virtualization environment.

In a Self-Hosted Engine deployment, RHV-M is installed on a virtual machine within the RHV environment that it manages. Instead of manually installing RHV-M, you deploy and modify a virtual appliance that Red Hat provides. Its virtual machine is created as part of host configuration, and the RHV-M engine is installed and configured during that process.

Hosts

Red Hat Virtualization supports two types of hosts, *Red Hat Virtualization Host* (RHV-H) or Red Hat Enterprise Linux (RHEL) with virtualization:

- *Red Hat Virtualization Host* (RHV-H) is a standalone, minimal operating system based on Red Hat Enterprise Linux. RHV-H is co-engineered with Red Hat Enterprise Linux and supports the same hardware and software ecosystem. Available as an ISO file, RHV-H can be used to provision a bare-metal physical system as a hypervisor in a RHV environment. RHV-H can be installed using optical media, USB storage, PXE/TFTP distribution, or by cloning.

  A RHV-H system contains only the packages necessary for a system to serve as a host. This configuration simplifies its management, maintenance, and deployment of the RHV environment. Red Hat Virtualization 4 supports many enhancements, including a writable root file system, the ability to install additional RPM packages, and the Web Console graphical administration interface.

- Alternatively, you can configure Red Hat Enterprise Linux to provide a RHV hypervisor. With access to a wider selection of packages, a Red Hat Enterprise Linux host provides greater flexibility and customization than a host deployed with the RHV-H operating system.

Depending on your organization's requirements, you may choose to deploy RHV-H or RHEL hosts, or both, within a Red Hat Virtualization environment.

The *Virtual Desktop Server Manager (VDSM)* management agent runs on all RHV hosts and allows for communication between the Red Hat Virtualization Manager and hosts. VDSM allows RHV-M to manage virtual machines and storage, and retrieve statistics from hosts and guests. VDSM uses libvirt to perform basic virtual machine life-cycle commands, such as start, stop, and reboot.

Storage

In a Red Hat Virtualization environment, storage domains are configured to provide hypervisor hosts access to virtual machine disk images, templates, and ISO files. RHV supports the use of various file systems (NFS, GlusterFS, or other POSIX-compliant file systems), as well as block devices (iSCSI targets, locally attached storage, or Fibre Channel storage devices) for the creation of storage domains. The file systems or block devices used to back storage domains are typically provided by NAS or SAN hardware in the infrastructure.

There are three types of storage domains: *data domains*, *ISO domains*, and *export domains*:

- Data domains hold virtual machine disk images, as well as templates used for virtual machine creation. Data domains can be migrated between different data centers, to allow moving virtual machines and templates from one to another.
- ISO domains store media images used for the deployment of virtual machines, but are now deprecated. Recommended practice is to use a data domain. You can upload images to data domains. In current Red Hat Virtualization versions, some tasks may still require an ISO domain.
- Export domains connect to and disconnect from one data center at a time, as a transport mechanism for moving virtual machines between data centers. Recommended practice is to use a data domain. You can migrate data domains between data centers and import the virtual machines into the new data center. In current Red Hat Virtualization versions, some tasks may still require an export domain.

Data domains can be created from any of the supported file system or block device types. ISO and export storage domains can only be created using the NFS file system type.

Getting Started with Red Hat Virtualization

In addition to this course, there are a number of useful resources that can help you prepare to deploy and use Red Hat Virtualization. In particular, review the official documentation at https://access.redhat.com/documentation/en/red-hat-virtualization/. Some very useful documents at that site include, but are not limited to:

- *Product Guide* provides an overview of the architecture of Red Hat Virtualization.
- *Planning and Prerequisites Guide* specifies hardware and software requirements and design considerations of a Red Hat Virtualization environment; review this document when planning your deployment.
- *Red Hat Virtualization Installation Methods* is an overview of supported methods for installing Red Hat Virtualization. Each of the four supported methods has an additional document:
  - *Installing Red Hat Virtualization as a self-hosted engine using the Cockpit web interface* explains how to use the Web Console to install RHV-M as a virtual machine running on the same hosts it manages. *This is the method used in this course.*
  - *Installing Red Hat Virtualization as a self-hosted engine using the command line* explains using the command-line to install RHV-M as a virtual machine running on the same hosts it manages.
  - *Installing Red Hat Virtualization as a standalone Manager with local databases* explains installing RHV-M and its PostgresSQL databases on a standalone server.
  - *Installing Red Hat Virtualization as a standalone Manager with remote databases* explains installing RHV-M on one server, and its databases on a separate PostgresSQL server.

- The *Introduction to the Administration Portal* provides detailed information about using the Red Hat Virtualization Administration Portal.
- The *Administration Guide* provides detailed information about key administration tasks using Red Hat Virtualization.
- The *Virtual Machine Management Guide* provides detailed information about the installation, configuration, and administration of virtual machines using Red Hat Virtualization.
- The *Introduction to the VM Portal* provides detailed information about using the Red Hat Virtualization VM Portal.
- The *Technical Reference* provides in-depth information about the technical design of Red Hat Virtualization.

References

Red Hat Virtualization
https://www.redhat.com/rhv

Further information is available in the documentation for Red Hat Virtualization 4.3, which can be found at
https://access.redhat.com/documentation/en-US/index.html

What's the difference between cloud and virtualization?
https://www.redhat.com/en/topics/cloud-computing/cloud-vs-virtualization

Access to Red Hat Virtualization software for evaluation purposes is available at
https://access.redhat.com/products/red-hat-virtualization/evaluation

Quiz

Describing the Concepts of Red Hat Virtualization

Choose the correct answers to the following questions:

1. Which software partitions the hardware into multiple virtual computers and runs the virtual machines?

a. Host
b. Guest
c. Hypervisor
d. Storage domain

2. Which of the following statements about managed virtualization and cloud computing is the least true?

a. Virtualization and cloud computing both allow the administration of physical resources as multiple virtual resources.
b. Expected workload is one way to determine whether managed virtualization or cloud computing is the best platform for a use case.
c. Virtualization technology is not useful for cloud computing.
d. Typically, virtualization is characterized by long-term allocation of resources, while cloud computing is characterized by short-term allocation of resources.

3. Which two of the following statements correctly describe features or requirements of Red Hat Virtualization Manager? (Choose two)

a. Virtualization Manager provides an API, and a graphical user interface to manage resources used in the data center.
b. Must be installed on a virtual machine external to the Red Hat Virtualization environment it manages.
c. Integrates with various directory servers for simplified user access management.
d. Manages physical and virtual resources in a Red Hat Virtualization environment.

4. Which three of the following statements about Red Hat Virtualization Host are true? (Choose three)

a. It is a standalone, minimal operating system based on Red Hat Enterprise Linux.
b. It includes a graphical web administration interface.
c. It is provided as an ISO image, but may be installed from optical media, USB storage, PXE/TFTP distribution, or by cloning.
d. It is the only way that Red Hat Virtualization can support hypervisors on physical hosts.

5. **Which two storage domain types must be provided by an NFS share? (Choose two)**
    a. Export domains
    b. Data domains
    c. ISO domains

Solution

Describing the Concepts of Red Hat Virtualization

Choose the correct answers to the following questions:

1. **Which software partitions the hardware into multiple virtual computers and runs the virtual machines?**
   a. Host
   b. Guest
   c. Hypervisor
   d. Storage domain

2. **Which of the following statements about managed virtualization and cloud computing is the least true?**
   a. Virtualization and cloud computing both allow the administration of physical resources as multiple virtual resources.
   b. Expected workload is one way to determine whether managed virtualization or cloud computing is the best platform for a use case.
   c. Virtualization technology is not useful for cloud computing.
   d. Typically, virtualization is characterized by long-term allocation of resources, while cloud computing is characterized by short-term allocation of resources.

3. **Which two of the following statements correctly describe features or requirements of Red Hat Virtualization Manager? (Choose two)**
   a. Virtualization Manager provides an API, and a graphical user interface to manage resources used in the data center.
   b. Must be installed on a virtual machine external to the Red Hat Virtualization environment it manages.
   c. Integrates with various directory servers for simplified user access management.
   d. Manages physical and virtual resources in a Red Hat Virtualization environment.

4. **Which three of the following statements about Red Hat Virtualization Host are true? (Choose three)**
   a. It is a standalone, minimal operating system based on Red Hat Enterprise Linux.
   b. It includes a graphical web administration interface.
   c. It is provided as an ISO image, but may be installed from optical media, USB storage, PXE/TFTP distribution, or by cloning.
   d. It is the only way that Red Hat Virtualization can support hypervisors on physical hosts.

5. Which two storage domain types must be provided by an NFS share? (Choose two)

a. Export domains

b. Data domains

c. ISO domains

Summary

In this chapter, you learned:

- Virtualization allows a single computer to be divided into multiple virtual computers in order to more efficiently use physical computing resources.
- Red Hat Virtualization (RHV) is an open source virtualization platform that allows centralized management of hosts, virtual servers, and desktops across an enterprise data center. It consists of three major components: the Red Hat Virtualization Manager, physical hosts, and storage domains.
- Red Hat Virtualization Manager provides a central management platform for both physical and virtual resources in a Red Hat Virtualization environment, and can be deployed as a Standalone Manager or Self-Hosted Engine.
- Data domains store virtual machine disk images and templates. ISO domains store ISO files used for operating system and application deployments.

Chapter 2

Installing and Configuring Red Hat Virtualization

Goal	Install a minimal Red Hat Virtualization (RHV) environment and use it to create a virtual machine.
Objectives	Install Red Hat Virtualization Host (RHV-H) on a computer that will host virtual machines. • Install Red Hat Virtualization Manager (RHV-M) as a VM on the Red Hat Virtualization Host system. • Configure storage domains in Red Hat Virtualization that can be used to store virtual machine disks and installation media. • Use the Administration Portal to manually create a Linux virtual machine running in the Red Hat Virtualization environment.
Sections	• Installing a Red Hat Virtualization Host (and Guided Exercise) • Installing a Self-hosted Red Hat Virtualization Manager (and Guided Exercise) • Preparing Storage for Virtual Machines and Installation Media (and Guided Exercise) • Creating a Linux Virtual Machine (and Guided Exercise)
Lab	Installing and Configuring Red Hat Virtualization

Installing a Red Hat Virtualization Host

Objectives

After completing this section, students should be able to describe how to install Red Hat Virtualization Host (RHV-H).

Deploying Red Hat Virtualization

Red Hat Virtualization (RHV) can be deployed as a self-hosted engine, or as a standalone Red Hat Virtualization Manager (RHV-M). Red Hat Virtualization can also be deployed as hyper-converged with Red Hat Gluster Storage using the Red Hat Hyper-converged Infrastructure for Virtualization (RHHI-V) product, which is another self-hosted option. RHHI-V is discussed later in this course. Because the self-hosted engine is Red Hat's recommended deployment option, the self-hosted deployment is used in this course.

Choosing a deployment option is one of multiple planning and preparation steps that are required before installing RHV components. The *Planning and Prerequisites Guide*, introduced earlier in this course, covers the prerequisites in detail.

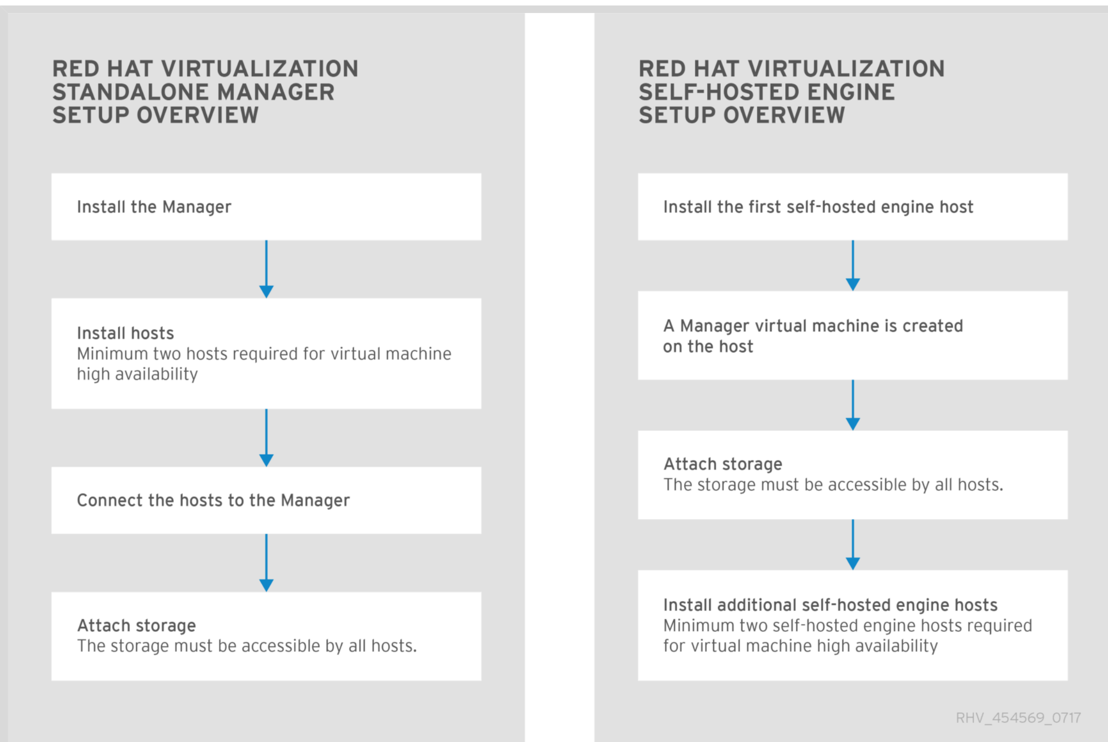

Figure 2.1: RHV Infrastructure Deployment Options

In a standalone deployment, the Manager runs on a physical server, or a virtual machine hosted in a separate virtualization environment. A standalone Manager is easier to deploy and manage, but requires an additional physical server. The Manager is only highly available when managed externally with a product such as Red Hat's High Availability Add-On.

In a self-hosted deployment, the Manager runs as a virtual machine on self-hosted engine nodes (specialized hosts) in the same environment it manages. A self-hosted engine environment requires one less physical server, but requires more administrative overhead to deploy and manage. The Manager is highly available without external HA management.

The Pre-built Classroom RHV Deployment

This course teaches the recommended self-hosted deployment, pre-installed on an engine host using the web console method. The classroom RHV deployment includes:

- A single engine host (**`hosta`**)and two guest hosts (**`hostb`** and **`hostc`**).
- Automatically configured local PostgreSQL relational configuration for the manager (**`engine`**) and data warehouse (**`ovirt-engine-history`**) databases.
- A data domain backed by NFS storage (**`hosted_storage`**).
- A management logical network (**`ovirtmgmt`**).
- An initial data center (**`default`**) and cluster (**`default`**).

Referring to the Documentation

Multiple RHV 4.3 installation guides are available on the Documentation page at the Red Hat Customer Portal, and are also provided with this course. These guides cover the deployment options, high availability requirements, local and remote database options, initial data storage service integration, and the calculated hardware requirements for RHV hosts and the workloads they will manage. Refer to the online documents for current procedures and recommendations.

Describing Red Hat Virtualization Hosts

Hosts, also known as hypervisors, are the physical servers on which virtual machines run. Red Hat Virtualization hosts use the Kernel-based Virtual Machine (KVM) technology. RHV supports two types of host: Red Hat Virtualization Host (RHV-H) and Red Hat Enterprise Linux (RHEL). The two host types can be mixed in your RHV environment, depending on your requirements.

RHV-H is a minimal version of the RHEL operating system, containing only the packages needed to act as a hypervisor and integrate with the RHV Manager. RHV-H includes a web console with extensions specific to Red Hat Virtualization, including virtual machine monitoring tools and a GUI installer for the self-hosted engine.

RHV hosts can also be installed using a normal RHEL distribution and adding the required Red Hat Virtualization repository packages and modules. RHEL hosts are highly customizable, so this host type may be preferable if, for example, your hosts require a specific file system layout.

Hosts that can run the Manager virtual machine are referred to as self-hosted engine nodes. At least two self-hosted engine nodes are required to support the high availability feature for the Manager engine. Hosts that service virtual machine guests are referred to as guest hosts or Virt Service hosts. Red Hat recommends that you install at least two guest hosts in your RHV environment, which is the minimum required to enable the migration and high availability features.

The Virtual Desktop and Server Manager (VDSM) is a RHV host service and API for communication between host nodes and the RHV Manager. The service is responsible for monitoring the memory, storage, and networks and to create, migrate, and destroy virtual machines.

System Requirements

Host hardware must meet the following requirements before installing Red Hat Virtualization Host (RHV-H). The requirements listed here are for Red Hat Virtualization 4.3.

Hardware certification for Red Hat Virtualization Host is essentially the same as the hardware certification for Red Hat Enterprise Linux. See the Knowledgebase article *"Does Red Hat Enterprise Virtualization also have hardware certification?"*.

RHV-H Hardware Requirements

RHV-H and RHEL use the kernel-based KVM hypervisor, which requires hardware virtualization extensions. Systems supporting the Intel 64 or AMD64 (x86-64) architecture must have 64-bit native CPUs and support the Intel VT-x or AMD-V virtualization extensions and the No eXecute (NX) flag. IBM POWER8 architecture systems are also supported.

Hosts must have at least 2 GB of RAM to be installed, and supports up to 4 TB of RAM. The amount of RAM required varies depending on the guest virtual machines operating system, applications and level of activity. KVM can over-commit physical RAM for virtual machines, allowing guests to be provision with greater memory allocations than is physically present. Sufficient swap space must be configured to provide sufficient virtual memory to permit such allocations. Memory over-commit is valid when guests do not operate at full capacity at once, but must be architected accurately.

RHV-H Storage Requirements

Hosts require storage to store configuration, logs, kernel dumps, and for use as swap space. The minimum storage requirements of RHV-H is 55 GB. However, Red Hat recommends using the default allocations, which use more storage space. If you are also installing the RHV-M appliance for self-hosted engine installation, then **`/var/tmp`** must be at least 5 GB.

RHV-H Network Requirements

Each host should have a minimum of one network interface card (NIC) with a bandwidth of at least 1 Gbps. Higher bandwidth NICs, such as 10 or 40 Gbps, are recommended. It is also recommended to use at two or more network interface cards, one dedicated to management traffic, and one or more additional NICs supporting network-intensive activities such as virtual machine migration and data domain storage access. Network operation performance will be affected by the bandwidth available.

RHV hosts must have a fully qualified domain name and correctly configured forward and reverse DNS name resolution that is shared with all other RHV host and the RHV Manager. DNS and NTP services for the RHV environment must be hosted outside the RHV environment, not on any host or guest of Red Hat Virtualization.

RHV-H automatically configures its local firewall to allow connections to required network services.

Installing Red Hat Virtualization Host

The RHV-H software is provided as an ISO image from the Red Hat Customer Portal at (https://access.redhat.com/). After locating the product using the **Downloads** link, the *Get Started* tab explains the installation requirements and provides the ISO file. The ISO file can be used to create bootable media to install the operating system.

RHV-H uses the same Anaconda installer as Red Hat Enterprise Linux. You can install RHV-H interactively, or you can perform a Kickstart-based automatic installation. This section focuses on the manual installation process.

The installer displays a graphical interface when started in manual interactive mode. The first screen asks to select the installation language, defaulting to **English (United States)**. Select the preferred language and click the **Continue** button.

The main screen displays several configuration options in a "hub-and-spoke" model. You can select options in any order to configure aspects of the installation, then return to the main screen to configure others. Options that are grayed may require other configurations to be completed first, such as configuring the network before an installation will allow access to an installation server to choose software to install, when installing by network.

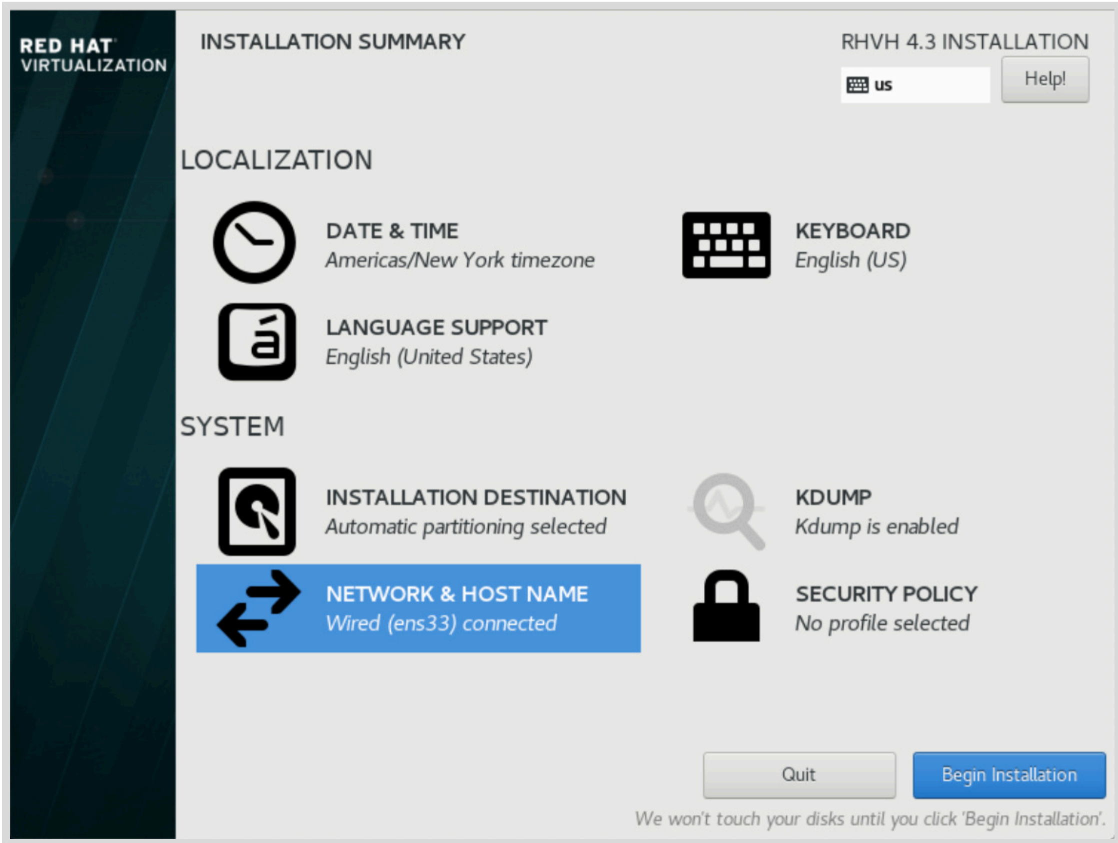

Figure 2.2: RHV-H Anaconda Installer Main Screen

- **DATE & TIME** configures the clock. Set the timezone using the graphical map or by selecting the **Region** and **City**. **Coordinated Universal Time** (UTC) is available from the **Etc** region.
- **KEYBOARD** is used to configure the physical host's keyboard type.
- **INSTALLATION DESTINATION** specifies how to partition and format the local RHV-H disk. There are two options for this section.

  1. Red Hat strongly recommends using **Automatically configure partitioning**. This option automatically determines the optimal storage configuration using the available storage devices. **/boot** is configured with a standard partition, but **/** and a number of other file systems are created on thinly-provisioned logical volumes.
  2. The other option, **I will configure partitioning**, allows customizing the file system sizes and layout. To configure this correctly requires advanced knowledge of required directories, sizing and layout. Use the automatic settings unless you are familiar with use case requiring manual configuration.

- **NETWORK & HOSTNAME** configures networking settings. By default, all detected network interfaces are disabled. Select the interfaces to enable from the list. By default, DHCP is used to configure the network interface. This screen also allows you to manually set the host name for the host. Ensure that you click the upper right button to **ON** after configuring this interface.

Select the **Configure** button to configure the network interface manually and to have it activated at boot. In the **General** tab, enable the checkbox for **Automatically connect to this network when it is available,**. Manually configure the IP address, netmask, and other information, create bonded interfaces or perform other advanced configuration tasks.

- The default settings for **KDUMP** and **SECURITY POLICY** are usually sufficient.

The installation begins when the **Begin Installation** button is selected, and progress bar screen displays. The screen displays two additional configuration items:

- **ROOT PASSWORD** allows you to set the host's root password to enable maintenance access.
- **USER CREATION** allows you to create an additional unprivileged user on the host. For security reasons, only create accounts required for software integration, such as for Red Hat Ansible or Red Hat Identity Management.

Click **Reboot** to complete the installation. Log into the console as **`root`** using the password set during installation. The node displays the current operating status and the web console URL.

```
Red Hat Virtualization Host 4.3.0 (el7.6)
Kernel 3.10.0-957.el7.x86_64 on an x86_64

rhvh login: root
Password:

  node status: OK
  See `nodectl check` for more information

Admin Console: https://172.25.250.10:9090

[root@rhvh ~]#
```

Managing Red Hat Virtualization Host

Red Hat Virtualization Host 4.3 includes a web console for host administration and for managing the host's VMs from a remote browser.

The RHV Administration Portal, discussed later in this course, and the RHV-H host's web console are two of the available interfaces for managing a RHV-H host. The web console can display a variety of information including the host's health status, self-hosted engine status, virtual machines, and virtual machine statistics.

Some other administrative operations that can be performed through the web console on a Red Hat Virtualization Host includes:

- Display a list of the virtual machines running on the host.
- Display a list of the virtual machines in the host's cluster.
- Comprehensive statistics for running virtual machines.
- Providing console access to running virtual machines.
- Starting virtual machines.
- Shutting down or forcing power off for virtual machines.
- Editing the **`vdsm.conf`** file.
- Management of the **`vdsmd`** service.
- Deployment of a self-hosted Red Hat Virtualization Manager.

The web console is accessed through a HTTPS connection to port 9090 on the Red Hat Virtualization Host. The web console URL is displayed when you log in to the host's physical console as **root**.

> **Note**
> Like RHV Manager's Administration Portal, the RHV-H host's web console service may present a TLS certificate for the HTTPS connection that is signed by an unrecognized Certificate Authority. Either add a security exception for that certificate in your web browser, or configure your web environment to use a trusted Certificate Authority.

To log in to web console, use the **root** user and password set when you installed the host.

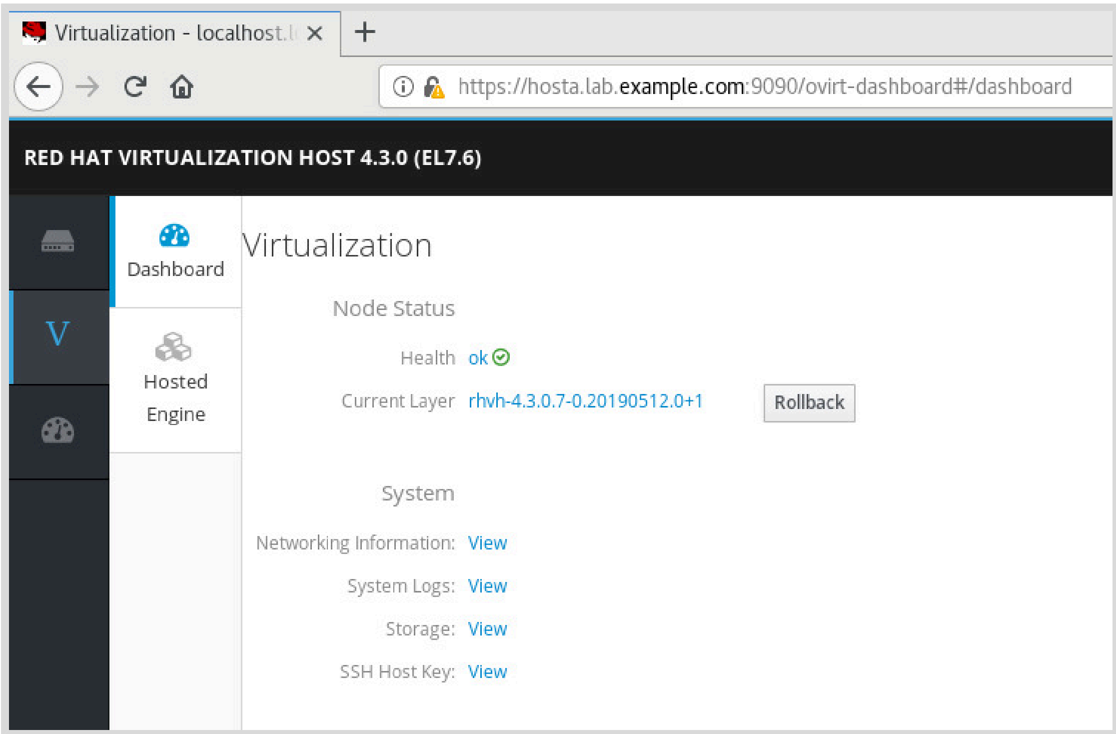

Figure 2.3: Virtualization Dashboard on a Red Hat Virtualization Host

Registering the Host

After a manual installation of a RHV-H host, the host must be register to have access to current software updates. Registration can be accomplished through the web console as the **root** user.

- Under the **Subscriptions** menu for this host, select **Register System**. Use the Customer Portal username and password associated with your support entitlements.
- Using a shell window started from the **Terminal** menu, enable the **rhel-8-server-rhvh-4-rpms** repository:

```
[root@rhvh ~]# subscription-manager repos --enable=rhel-8-server-rhvh-4-rpms
```

References

Further information about RHV hosts and installation is available in the following Red Hat Virtualization 4.3 documentation: *Planning and Prerequisites Guide* located at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3
Red Hat Virtualization 4.3 Installation Methods located at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3
Installing Red Hat Virtualization as a self-hosted engine using the Cockpit web interface located at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3
Installing Red Hat Virtualization as a self-hosted engine using the command line located at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3
Installing Red Hat Virtualization as a standalone Manager with local databases located at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3
Installing Red Hat Virtualization as a standalone Manager with remote databases located at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3

For more information on hardware certification, see the Knowledgebase article *Does Red Hat Virtualization also have hardware certification?* at
https://access.redhat.com/solutions/725243/
.

Guided Exercise

Viewing a Red Hat Virtualization Host

In this exercise, you will explore a prebuilt Red Hat Virtualization Host and its environment.

Outcomes

You should be able to:

- Familiarize yourself with a Red Hat Virtualization Host's web console and locate the various tools that manage the host.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab installation-install-rhvh start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab installation-install-rhvh start
```

1. Use your web browser to access **https://hosta.lab.example.com:9090**.

   Log in with username **root** with the password **redhat**.

   1.1. Open Firefox and go to the **https://hosta.lab.example.com:9090** URL. If necessary, add a security exception to permit Firefox to use the self-signed SSL certificate and bypass the **Your connection is not secure** warning by clicking the **Advanced** button and then **Add Exception** button. In the pop up window click **Confirm Security Exception**. The page will reload.

   1.2. Log in with username **root** with the password **redhat**.

   Click **Log In**.

2. Click **hosta.lab.example.com** -> **System** on the menu to access the system's statistics.

   This page shows basic operating system statistics, such as current load, disk usage, disk I/O, and network traffic.

3. Click on **Logs** on the menu to access the system's logs.

   This page shows the systemd system logs. You can choose the date to access the logs from as well as the severity of the log entries.

   3.1. Click the **Severity** drop-down menu, and choose Everything.

   3.2. Review the log entries for a chosen day. Click a day to view. A log entry detail page displays additional event information.

4. Click **Storage** on the menu to view local storage and logging. You can add additional **NFS** mount points.

In the upper right corner of the page, you can add **RAID Devices**, **Volume Groups**, **VDO Devices**, and **iSCSI Targets** as required.

5. On the menu, choose **Networking**.

   This page shows detailed information for the current network configuration.

   5.1. There are two NICs, one for management traffic and one for storage traffic.

   5.2. Click the **Firewall** link.

   5.3. Under **Allowed Services**, choose **Add Services** to view the available services.

   5.4. Click on **Cancel**.

6. Click on **Services** on the menu to view the enabled, disabled and static services within RHV-H.

7. Check to see if each host has forward and reverse DNS name resolution. Open a shell window, click on **Terminal** on the menu. Use the **dig** command to locate the DNS server and view the existing forward and reverse host entries. Repeat for all four hosts.

   7.1. Use **dig** to perform a forward DNS query for **hosta**.

```
[root@hosta ~]# dig hosta.lab.example.com
...output omitted...
ANSWER SECTION:
hosta.lab.example.com. 3600 IN A 172.25.250.10

Query time: 0 msec
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

   7.2. Use **dig** to perform a reverse DNS query using the **hosta** IP address.

```
[root@hosta ~]# dig -x 172.25.250.10
...output omitted...
ANSWER SECTION:
10.250.25.172.in-addr.arpa. 3600 IN PTR hosta.lab.example.com.

Query time: 0 msec
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

   7.3. Use **dig** to perform a forward DNS query for **hostb**.

```
[root@hosta ~]# dig hostb.lab.example.com
...output omitted...
ANSWER SECTION:
hostb.lab.example.com.  3600 IN A 172.25.250.11

Query time: 0 msec
```

```
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

7.4. Use **dig** to perform a reverse DNS query using the **hostb.lab.example.com** IP address.

```
[root@hosta ~]# dig -x 172.25.250.11
 ...output omitted...
ANSWER SECTION:
11.250.25.172.in-addr.arpa. 3600 IN PTR hostb.lab.example.com.

Query time: 0 msec
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

7.5. Use **dig** to perform a forward DNS query for **hostc.lab.example.com**.

```
[root@hosta ~]# dig hostc.lab.example.com
...output omitted...
ANSWER SECTION:
hostc.lab.example.com.  3600 IN A 172.25.250.12

Query time: 0 msec
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

7.6. Use **dig** to perform a reverse DNS query using the **hostc.lab.example.com** IP address.

```
[root@hosta ~]# dig -x 172.25.250.12
...output omitted...
ANSWER SECTION:
12.250.25.172.in-addr.arpa. 3600 IN PTR hostc.lab.example.com.

Query time: 0 msec
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

7.7. Use **dig** to perform a forward DNS query for **hostd.lab.example.com**.

```
[root@hosta ~]# dig hostd.lab.example.com
...output omitted...
ANSWER SECTION:
hostd.lab.example.com.  3600    IN      A       172.25.250.13

Query time: 0 msec
```

```
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

7.8. Use **dig** to perform a reverse DNS query using the **hostd.lab.example.com** IP address.

```
[root@hosta ~]# dig -x 172.25.250.13
...output omitted...
ANSWER SECTION:
10.250.25.172.in-addr.arpa. 3600 IN     PTR     hostd.lab.example.com.

Query time: 0 msec
SERVER: 172.25.250.254#53(172.25.250.254)
WHEN: Mon Jul 08 00:53:42 CDT 2019
MSG SIZE  rcvd: 56
```

Finish

On **workstation**, run the **lab installation-install-rhvh finish** script to complete this exercise.

```
[student@workstation ~]$ lab installation-install-rhvh finish
```

This concludes the guided exercise.

Installing Red Hat Virtualization Manager

Objectives

After completing this section, you should be able to describe how to install Red Hat Virtualization Manager (RHV-M) as a self-hosted engine on an existing Red Hat Virtualization host.

Red Hat Virtualization Manager

Red Hat Virtualization Manager (RHV-M) provides a central management platform for the physical and logical resources of a Red Hat Virtualization (RHV) environment. RHV Manager is built upon Red Hat Enterprise Linux and Red Hat JBoss Enterprise Application Platform (EAP). It uses PostgreSQL databases to store information and provides several management interfaces, including a REST API.

RHV-M Hardware Requirements
: At minimum, a dual core CPU, but a quad core CPU or multiple dual core CPUs are recommended. 16 GB or more of system RAM is recommended, but RHV-M can function with 4 GB of available system RAM if the data warehouse is not installed and if memory is not being consumed by existing processes.

RHV-M Storage Requirements
: At least 25 GB of locally accessible, writable disk space is required, but 50 GB or more is recommended.

RHV-M Network Requirements
: A single Network Interface Card (NIC) with a bandwidth of at least 1 Gbps.

Installation and Configuration Process

The process of installing and configuring a Red Hat Virtualization self-hosted engine can be broken down into three steps:

1. Install a Red Hat Virtualization Host.

   Subscribe the host to the entitlements for Red Hat Virtualization and enable the appropriate software repositories. Confirm that forward and reverse DNS queries succeed for this host.

2. Initiate the RHV-M installation from the RHV-H node.

   RH-M can be installed using either the graphical web console, or from the command-line using the **`hosted-engine --deploy`** command. Red Hat recommends the web console RHV-M installation.

3. Configure the initial data domain.

   Specify storage for use by the Red Hat Virtualization engine hosts. The back end storage for the requested storage type, such as NFS or iSCSI, can be previously configured and made accessible. The storage becomes the initial data domain for the **`Default`** data center and contains the system image for the Red Hat Virtualization Manager virtual machine.

Install a Red Hat Virtualization Host

In this course, you have already learned how to install a Red Hat Virtualization Host. Since the self-hosted engine installation runs on a Red Hat Virtualization host, at least one host must be installed first. Additional Red Hat Virtualization hosts can be added later.

Initiate the RHV-M Installation From the RHV-H Node

To perform the installation using the web console, open a browser from another system and connect to port 9090 on the your RHV-H node, using the node's fully qualified domain name in the URL, such as `https://hosta.lab.example.com:9090`.

Select **V** in the menu to access virtualization information for the host.

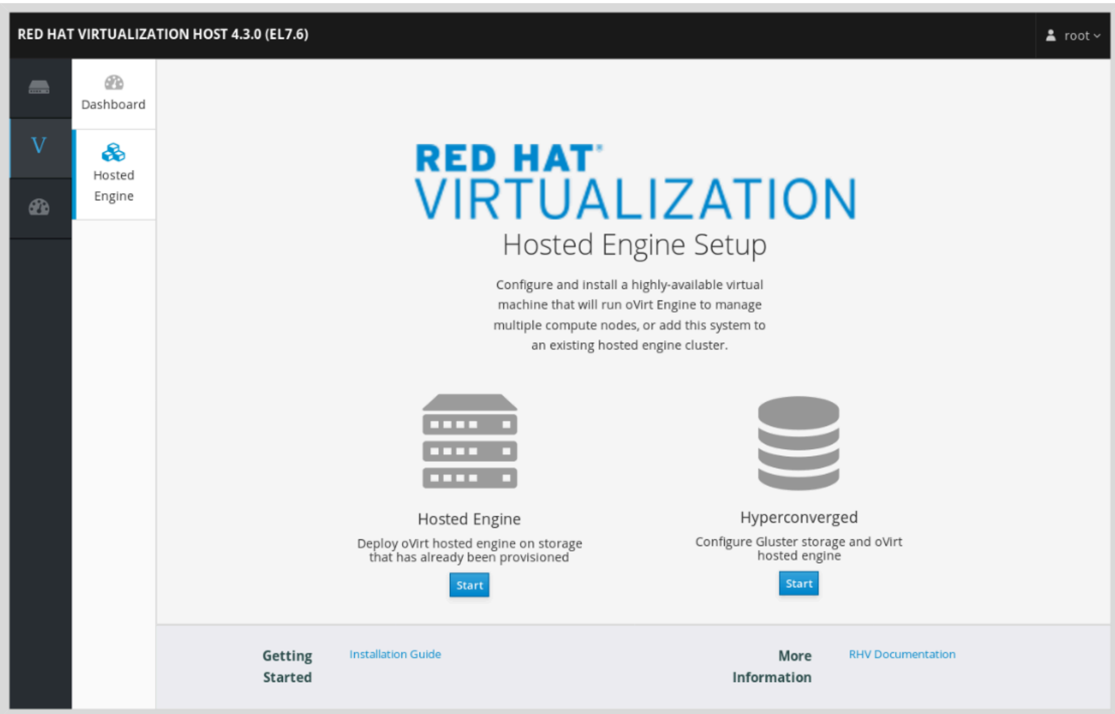

Figure 2.4: Red Hat Virtualization Hosted Engine Setup

The self-hosted engine can either be installed using external storage that has already been provisioned or it can be installed in a hyper-converged environment where Red Hat Gluster Storage is configured into the guest nodes as steps in the installation. This section describes a self-hosted engine installation using external storage that has already been provisioned.

The **Start** button begins an installation process. The Hosted Engine Deployment wizard prompts for configuration information. Alternatively, run the **`hosted-engine --deploy`** command from a RHV-H command-line, which will also prompt for the needed configuration information.

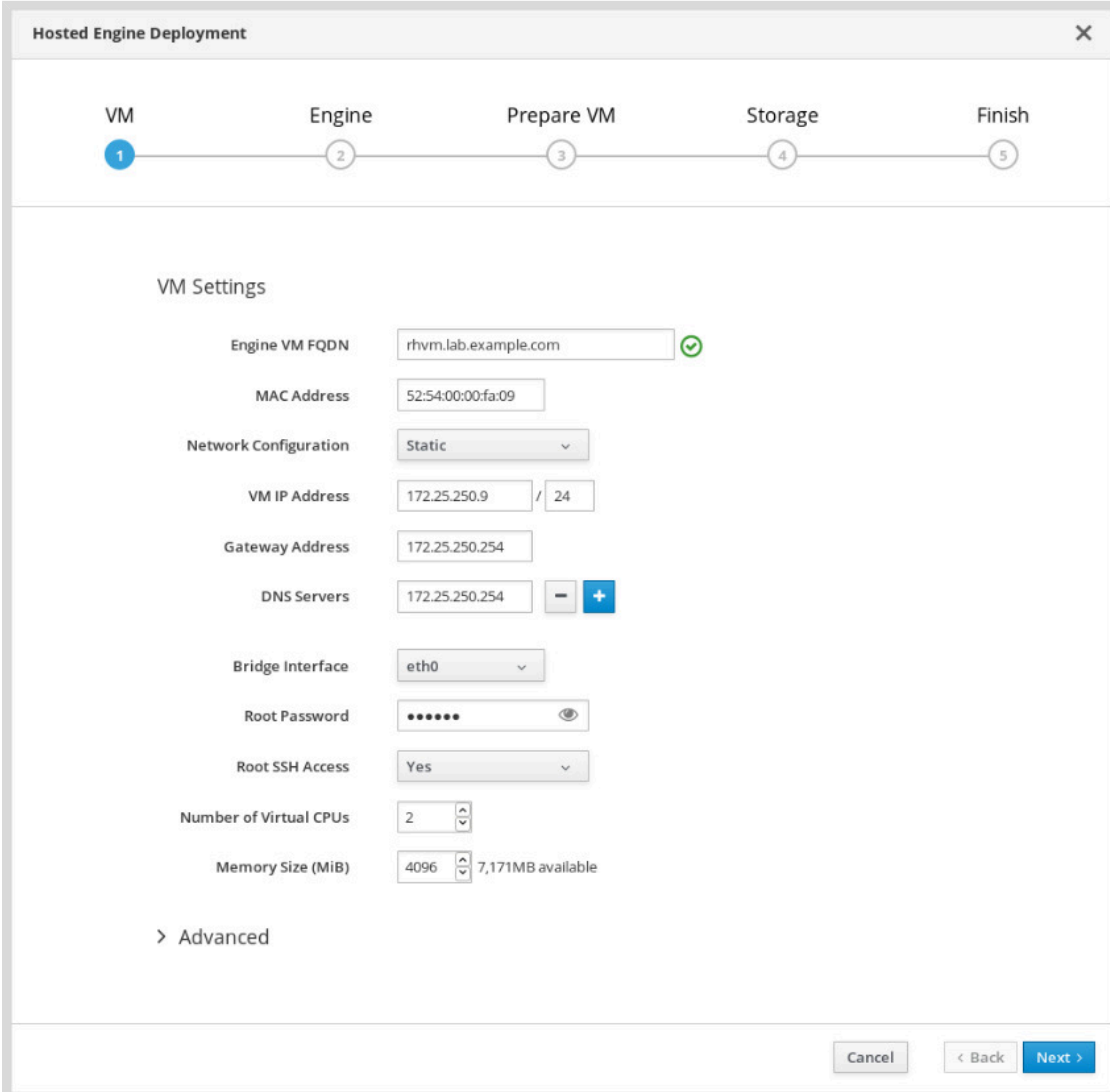

Figure 2.5: Hosted Engine Deployment Configuration

Most configuration fields are self-explanatory. DNS must be able to resolve the fully qualified domain name of the engine virtual machine. The wizard provides a **Validate** button to check for DNS resolution. In many environments, DHCP is configured with MAC address reservations to ensure that infrastructure nodes retain a consistent IP address. The RHV-M virtual machine's IP address (either by DHCP or manually entered) must match the DNS-resolved IP address.

During the installation process, the following actions will occur:

- The RHV-M appliance will be downloaded and started as a virtual machine on the host. The RHV-M appliance can be installed in advance before starting the installation:

```
[root@rhvh ~]# yum install rhvm-appliance
```

- The RHV-M virtual machine will run the **engine-setup** command using the configuration information you provided.

- The installer will add and activate the RHV-H node (where the installer is running) as the first host in the RHV environment. If successful, the installation wizard prompts for the storage configuration.

Configure the Initial Data Domain

A data domain must be created to complete the self-hosted engine installation. The data domain allow the Default data center to be activated and for RHV-M to run as a virtual machine. Connect to storage which has already been provisioned for use by the RHV deployment. This classroom environment uses NFS for the initial data domain. When storage is attached, the self-hosted installation completes. You can access the RHV Manager using the Administration Portal.

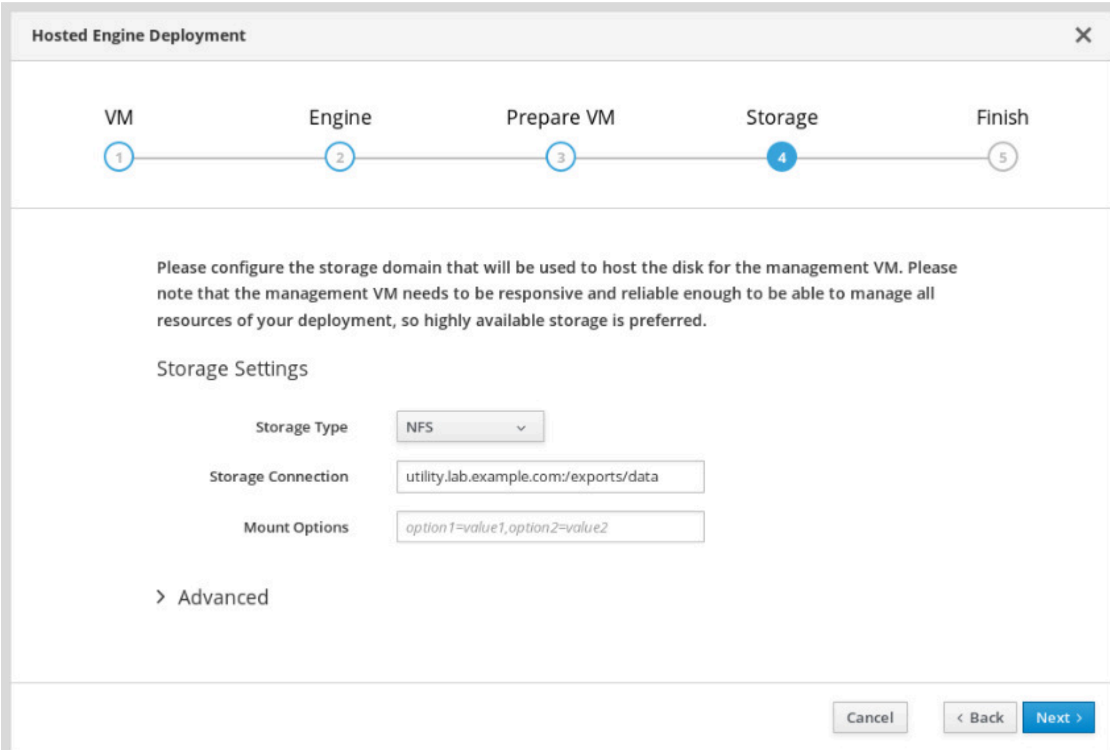

Figure 2.6: Storage Configuration

Accessing the Administration Portal

The RHV Manager has multiple interfaces, including web-based programming and RESTful APIs, that manage the RHV environment. The Administration Portal is a primary interface, available as a link on the RHV Manager's landing page. This cross-platform interface is accessible using any supported web browser.

Browser Support

A web browsers is used to access RHV Manager's Administration Portal and other resources. Red Hat browser support for RHV-M has three support tiers:

Tier 1: Mozilla Firefox Extended Support Release (ESR) version on RHEL.
Browser and operating system combinations that are fully tested and supported. Red Hat Engineering is committed to fixing issues.

Tier 2: Most recent version of Google Chrome, Mozilla Firefox, or Microsoft Edge on any OS.
Browser and operating system combinations that are partially tested and are likely to work. Red Hat Engineering will attempt to fix issues.

Tier 3: Older versions of Google Chrome, Mozilla Firefox, or other browsers on any OS.
Browser and operating system combinations that are not tested but may work. Minimal support is provided. Red Hat Engineering will attempt to fix only minor issues.

Reaching the Administration Portal

To access the Administration Portal, open a web browser and enter the landing page URL: `https://`***`your-rhvm-server-fqdn`***. In the **Portals** section, click on the **Administration Portal** link. Alternatively, open the Administration Portal directly by going to the URL `https://`***`your-rhvm-server-fqdn`***`/ovirt-engine/webadmin/`.

If you cannot see the **Administration Portal** link, you may need to increase your display resolution or you may need to use your web browser's *zoom out* feature.

Figure 2.7: Red Hat Virtualization Manager Landing Page

Configuring RHV-M Certificate Security

When you attempt to access the Red Hat Virtualization landing page, your web browser may complain because it does not recognize the certificate authority (CA) that signed the TLS certificate for RHV Manager's web server. There are three choices of methods to correct this:

- Download and install the local CA certificate in your web browser. The CA certificate is available at `http://`***`your-rhvm-server-fqdn`***`/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA` When installing this certificate in your browser, enable the option to **Trust this CA to identify websites**.

- Alternately, replace the TLS certificate used by RHV-M with one that is signed by a CA already trusted by your web browser. Currently, instructions for this method are found in Appendix D, "Red Hat Virtualization and Encrypted Communication", of the *Red Hat Virtualization 4.3 Administration Guide*.

- The simplest but least secure method is to add a security exception in your web browser so that it accepts the self-signed certificate as valid. The procedure from the Firefox error page is:
  1. Click the **Advanced** button.
  2. Click the **Add Exception** button.
  3. Click the **Confirm Security Exception** button in the **Add Security Exception** dialog.

Logging in to the Administration Portal

For your first login, authenticate using the **`admin`** user and the password set during the self-hosted engine installation. The **Profile** field should be set to **internal** to use the internal authentication domain.

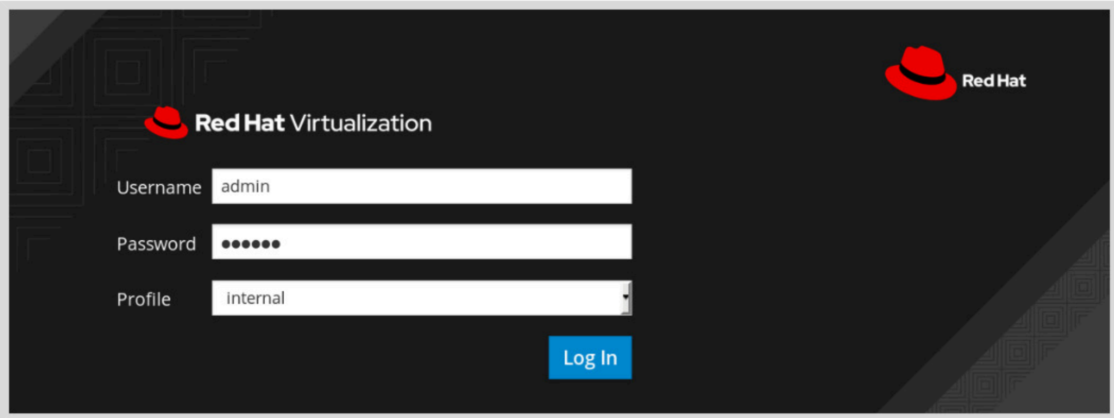

Figure 2.8: Administration Portal Login Screen

Upon successful login, you reach the Administration Portal dashboard.

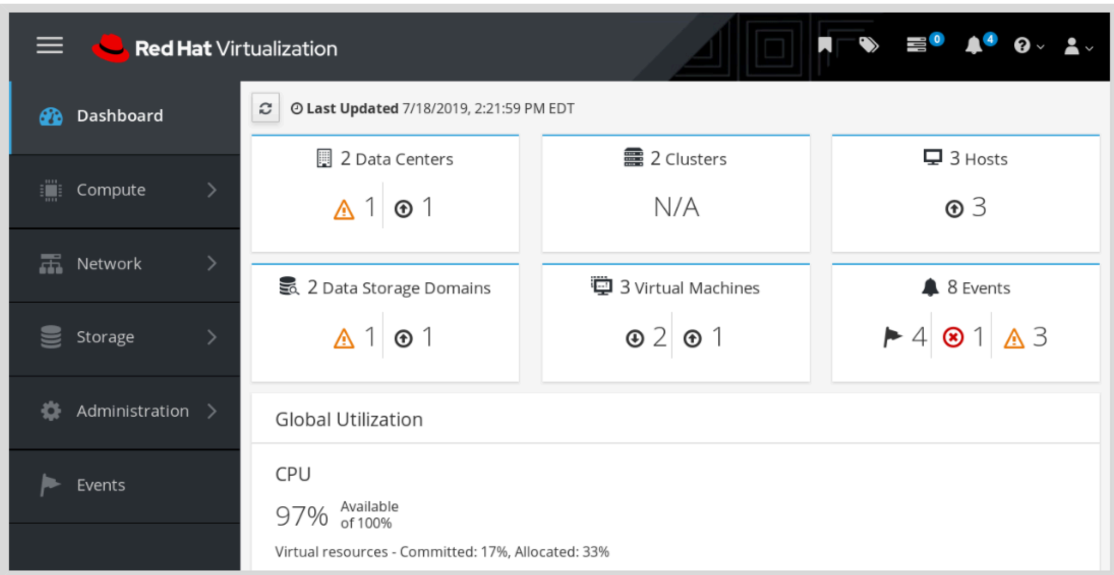

Figure 2.9: Administration Portal Dashboard

References

Further information is available in the following documents:

- The *Red Hat Virtualization 4.3 Installing Red Hat Virtualization as a self-hosted engine using the Cockpit web interface Guide*, especially the chapter "Installing the Red Hat Virtualization Manager".
  https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/installing_red_hat_virtualization_as_a_self-hosted_engine_using_the_cockpit_web_interface/index

- The *Red Hat Virtualization 4.3 Planning and Prerequisites Guide*.
  https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/planning_and_prerequisites_guide/index

- The *Red Hat Virtualization 4.3 Administration Guide*, especially the discussion in the appendices on replacing the TLS certificate for Red Hat Virtualization Manager's web server.
  https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index

Guided Exercise

Installing a Self-hosted Red Hat Virtualization Manager

In this exercise, you will explore the prebuilt *Red Hat Virtualization Manager* (*RHV-M*) and its Administration Portal.

Outcomes

You should be able to:

- Check prerequisites and some configuration settings on RHV-M.
- Download and install the Certificate Authority (CA) certificate for RHV-M.
- Check RHV-M resources for data centers, clusters, hosts, storage, and virtual machines.

Before You Begin

Log in to **`workstation`** as **`student`** using **`student`** as the password.

On **`workstation`**, run the **`lab install-rhvm start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab install-rhvm start
```

1. Confirm that the self-hosted RHV-M meets certain prerequisites and familiarize yourself with some of the configuration settings.

    1.1. From **`workstation`**, open a terminal and use **`ssh`** to log in to **`rhvm.lab.example.com`** using the username **`root`** and the password **`redhat`**.

```
[student@workstation ~]$ ssh root@rhvm.lab.example.com
```

    1.2. Verify that forward and reverse DNS records exist for **`rhvm.lab.example.com`** and that the **`rhvm.lab.example.com`** **`eth0`** interface is assigned that address.

```
[root@rhvm ~]# host rhvm.lab.example.com
rhvm.lab.example.com has address 172.25.250.14
[root@rhvm ~]# host -t PTR 172.25.250.14
9.250.25.172.in-addr.arpa domain name pointer rhvm.lab.example.com.
[root@rhvm ~]# ip add show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
 group default qlen 1000
    link/ether 52:54:00:00:fa:09 brd ff:ff:ff:ff:ff:ff
    inet 172.25.250.14/24 brd 172.25.250.255 scope global noprefixroute eth0
```

```
       valid_lft forever preferred_lft forever
    inet6 fe80::5054:ff:fe00:fa09/64 scope link
       valid_lft forever preferred_lft forever
```

1.3. Identify the amount of available memory and CPU. At a minimum, RHV-M should have a dual core CPU with 4 GB of available system RAM. Your output may be different.

```
[root@rhvm ~]# free
              total        used        free   ...
Mem:        3870976     2860780      122144   ...
Swap:       8388604        2824     8385780
[root@rhvm ~]# lscpu | grep 'CPU(s)'
CPU(s):                 2
On-line CPU(s) list:    0,1
NUMA node0 CPU(s):      0,1
```

1.4. Your self-hosted RHV-M installation created local databases for the engine and data warehouse. Verify that the databases exist on the localhost and that the localhost is listening on the specified port.

```
[root@rhvm ~]# grep -A1 localhost /usr/share/ovirt-engine-dwh/\
services/ovirt-engine-dwhd/ovirt-engine-dwhd.conf | \
grep -v '^#'
--
DWH_DB_HOST="localhost"
DWH_DB_PORT="5432"
--
ENGINE_DB_HOST="localhost"
ENGINE_DB_PORT="5432"
[root@rhvm ~]# ss -tln | grep -w 5432
LISTEN     0      128          *:5432                     *:*
LISTEN     0      128         :::5432                    :::*
```

1.5. The RHV-M REST API needs to be running to access the RHV-M Administration Portal. Verify that **`ovirt-engine.service`** is both running and enabled on **`RHV-M`**. Exit the SSH session when you are done.

```
[root@rhvm ~]# systemctl is-active ovirt-engine.service
active
[root@rhvm ~]# systemctl is-enabled ovirt-engine.service
enabled
[root@rhvm ~]# exit
```

2. Download and install the local CA certificate.

2.1. On **`workstation`**, open **`http://rhvm.lab.example.com/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA`** in a web browser to download and install the local CA certificate.

2.2. When prompted, select the option to **Trust this CA to identify websites** and click the **OK** button.

3. Using the Administration Portal, examine the resources created during the self-hosted engine installation.

3.1. On **`workstation`**, open **`https://rhvm.lab.example.com`** in a web browser to access the landing page for the RHV Manager.

3.2. On the landing page, click **Administration Portal** under the **Portals** section to be redirected to the Administration Portal login page.

3.3. Log in using the user name **`admin`**, the password **`redhat`**, and the **Profile** set to **internal**. Upon successful login, you reach the dashboard for the Administration Portal.

3.4. Identify the data centers that exist by going to: **Compute** → **Data Centers.** You should see a data center named **`Default`**. The **`Default`** data center should have a **Status** of **`Up`**.

3.5. Identify the clusters that exist by going to: **Compute** → **Clusters.** You should see a cluster named **`Default`**. The **`Default`** cluster belongs to the **`Default`** data center.

3.6. Identify the hosts that exist by going to: **Compute** → **Hosts.** Only one host is required for a self-hosted installation, but four hosts exist in your classroom environment to provide high availability. All four hosts belong to the **`Default`** data center and the **`Default`** cluster.

3.7. Identify the storage that exists by going to: **Storage** → **Domains.** You should see a storage domain with the name of **`hosted_storage`**. The domain type should be **`Data (Master)`** and the storage type should be **NFS**. Hovering over the crown icon indicates that the **`hosted_storage`** storage domain contains the hosted engine VM data.

3.8. Verify that the RHV-M guest is running as a virtual machine within the RHV environment by going to: **Compute** → **Virtual Machines.** You should see a virtual machine with the name of **`HostedEngine`**. Notice how the IP address matches the DNS entry for **`rhvm.lab.example.com`**. Hovering over the crown icon indicates that this machine is the Hosted Engine VM.

Finish

On **`workstation`**, run the **`lab install-rhvm finish`** script to complete this exercise.

```
[student@workstation ~]$ lab install-rhvm finish
```

This concludes the guided exercise.

Configuring Storage for Virtual Machines and Installation Media

Objective

After completing this section, students should be able to describe the use of storage domains in Red Hat Virtualization for storing virtual machine disks and installation media.

Storage Domains

A storage domain is a repository for virtual machine disk images used for system disks, data, or installation media. There are three types of storage domains (data, iso, and export), but only data domains are needed. Although still available, the iso and export storage domains have been deprecated.

A data domain stores the disk images that represent the virtual hard drives or other storage for virtual machines. These disk images can contain the operating system of the virtual machine or be purely used for data. ISO disk images, used to install virtual machine operating systems and applications, can be uploaded to a data domain. When creating a new virtual machine, an ISO disk image in the data storage domain can be attached to the virtual machine as if was inserted into a CD/DVD drive.

A storage domain can use one of a number of different storage technologies to provide its back end storage:

- Red Hat Gluster Storage native client (GlusterFS)
- Fiber Channel Protocol (FCP)
- Internet Small Computer System Interface (iSCSI)
- Network File System (NFS)
- Other POSIX-compliant file systems
- Local storage attached directly to a virtualization host. The host must be in a data center and cluster containing no other hosts.

For NFS-based storage domains, all virtual disks, templates, and snapshots are stored as files. For block-oriented storage, such as iSCSI and FCP, Red Hat Virtualization uses Logical Volume Management (LVM) to organize the block storage as a volume group, with individual virtual disks, images, and snapshots managed internally as logical volumes.

For a RHV data center to be considered active, it must have at least one cluster containing at least one host with a status of **`Up`**. Additionally, a data domain must exist with the active cluster host able to access the data domain's storage. These requirements must be met for a RHV self-hosted engine installation to succeed.

Your RHV self-hosted engine classroom environment contains:

- The **`Default`** data center contains the **`Default`** cluster.
- Four RHV hosts (**`hosta`**, **`hostb`**, **`hostc`**, and **`hostd`**) belong to the **`Default`** cluster and have a status of **`Up`**.
- The **`hosted_storage`** data domain exists and is backed by an NFS share available on **`utility.lab.example.com`**. The **`hosted_storage`** data domain has a status of **`active`**. The initial data domain can be any of the listed storage types. Using NFS is a classroom convenience.

Preparing NFS Exports for Storage Domains

NFS exports, for use as NFS-based storage domains, must be prepared for use with Red Hat Virtualization.

- A file system of sufficient size for the storage domain is exported from the NFS server.
- The NFS server exports the file system in read-write mode to all Red Hat Virtualization hosts in the cluster. Configure the export in either **`/etc/exports`** or a file ending with .exports in **`/etc/exports.d/`** (for example, **`/etc/exports.d/rhv.exports`**).
- The top-level directory on the exported file system must be owned by user **`vdsm`** (UID 36) and by group **`kvm`** (GID 36). The directory permissions are set so that user **`vdsm`** has read-write-execute access and the **`kvm`** group and all other users have read-execute access (octal permissions **`0755`**).
- Ensure that **`nfs-server.service`** is running and enabled.
- If a firewall is in use, verify that all Red Hat Virtualization hosts are allowed access to the NFS share.

> **Important**
> The NFS server must *not* be one of the physical Red Hat Virtualization hosts. The NFS server can be a single point of failure for any storage domains using it, the cluster, and the virtual machines using those storage domains. Ideally, the NFS server should be made resilient and highly available.

The hosted_storage Data Domain

This section shows details of the rebuilt **`hosted_storage`** data domain as viewed in the RHV-M Administration Portal.

From the Administration Portal, navigate to the **Storage** menu and select **Domains**. The **`hosted_storage`** data domain should exist. Details are viewed by right-clicking the **`hosted_storage`** link and selecting **Manage Domain**.

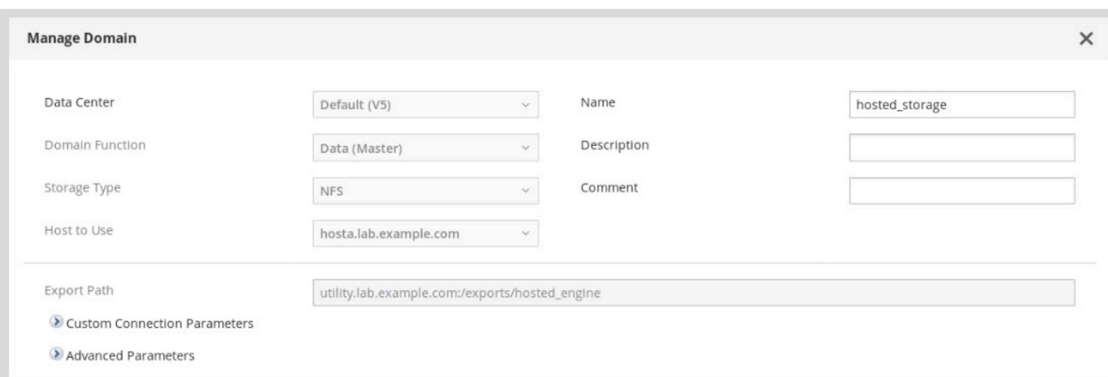

Figure 2.10: Details of the hosted_storage Data Domain.

Data Center
: The name of the data center associated with the storage domain. The classroom environment uses the **`Default`** data center.

Name

The name of the storage domain. A RHV self-hosted engine installation defaults to creating a data storage domain named **`hosted_engine`**. The classroom environment uses this default.

Domain Function

This can be Data, ISO, or Export. An active Data domain needs to exist before ISO and Export appear as options. The ISO and Export functions are deprecated. The **`hosted_engine`** storage domain is a data domain.

Storage Type

Available options are NFS, POSIX compliant FS, GlusterFS, iSCSI, and Fibre Channel. The classroom environment uses the NFS storage type.

Host to Use

This can be any active host associated with the selected data center. If this menu is blank, then no hosts are associated with the selected data center or the hosts are not active. For example, the hosts may be in maintenance mode.

Export Path

The information displayed below **Host to Use** will vary based on the selected storage type. For NFS, export path expects the name of the NFS server and the path to the share. The classroom environment uses **`utility.lab.example.com:/exports/hosted_engine`** for the export path.

Uploading ISO Files to a Data Domain

Upload an ISO file if you plan to perform either manual or automated installations of virtual machines. When a network server, such as HTTP or FTP, shares your installation media, uploading a boot ISO can be used as an alternative to uploading a DVD ISO.

Uploading ISO Images with the Administration Portal

Start by downloading one or more ISO files to the machine used to access the RHV Administration Portal. From the Administration Portal, navigate to **Storage** in the menu and select **Disks**. The **Upload** button start an upload, opening a dialog to locate the ISO file on the local system and select it.

For ISO files, the Alias and Description fields default to the name of the ISO file. Use the Data Center and Storage Domain fields to specify the destination for the upload. Use the Disk Profile and Host to Use fields to specify how it should be uploaded, and which will perform the task. Different disk profiles can be created to specify storage quality of service levels.

To upload, you must be able to connect to the **`ovirt-imageio-proxy`**. Use the **Test Connection** button to test that connection. A green success box indicates that the upload can succeed.

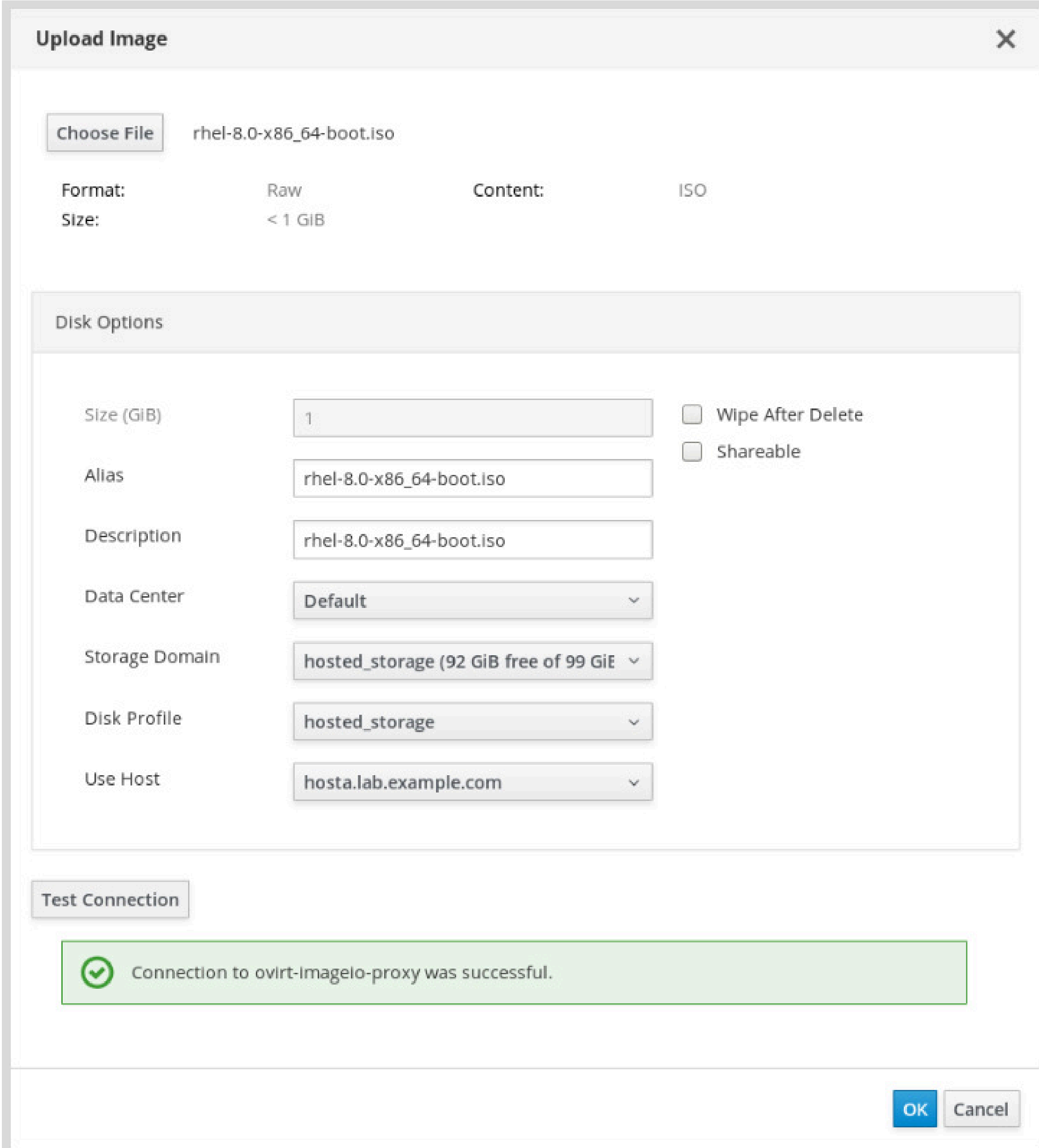

Figure 2.11: Upload Image Screen with Successful Connection Test.

If the **Test Connection** button returns an orange warning box, click the ovirt-engine certificate link.

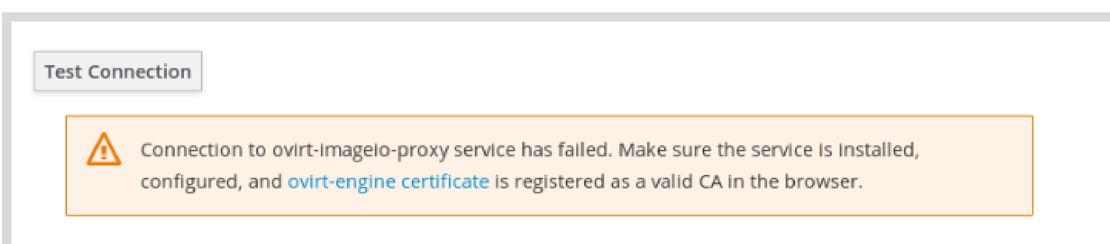

Figure 2.12: The ovirt-engine Certificate Needs to be Downloaded.

In the Downloading Certificate window, check the box next to Trust this CA to identify websites and then click the **OK** button. At this point, clicking the **Test Connection** button returns a green success box.

Figure 2.13: Download the ovirt-engine Certificate and Trust It to Identify Websites.

References

Further information is available in the Storage chapter of the *Administration Guide* for Red Hat Virtualization; at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#chap-Storage

Guided Exercise

Configuring Storage for Virtual Machines and Installation Media

In this exercise, you will explore the existing NFS configuration on **utility.lab.example.com** and upload an ISO in preparation for installing your first virtual machine.

Outcomes

You should be able to:

- Verify that NFS is configured correctly on **utility.lab.example.com**.
- Upload an ISO file to the existing **hosted_storage** data storage domain.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab install-storage start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab install-storage start
```

1. Connect to the **utility** server to verify its existing NFS configuration:

   1.1. Use ssh to connect to **utility.lab.example.com** as the **root** user:

```
[student@workstation ~]$ ssh root@utility
[root@utility ~]#
```

   1.2. Verify that **nfs-server.service** is running and enabled:

```
[root@utility ~]# systemctl is-active nfs-server.service
active
[root@utility ~]# systemctl is-enabled nfs-server.service
enabled
```

   1.3. Identify the active firewalld zones and verify that NFS is allowed. NFS access may be restricted to a specific internal network. In your environment, only the public zone is being used.

```
[root@utility ~]# firewall-cmd --get-active-zones
public
  interfaces: eth0 eth1 eth0.10 eth0.20
```

```
[root@utility ~]# firewall-cmd --list-all --zone=public
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: eth0 eth1 eth0.10 eth0.20
  sources:
  services: ssh dhcpv6-client tftp dhcp mountd rpc-bind nfs freeipa-ldap freeipa-ldaps dns
  ports: 9090/tcp 3260/tcp
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

1.4. Identify the currently exported shares:

```
[root@utility ~]# exportfs
/exports/hosted_engine
              <world>
/exports/data   <world>
/exports/data2  <world>
```

1.5. Confirm user ownership, group ownership, and permissions on the share. The share should be owned by user **vdsm** (UID 36), group **kvm** (GID 36), and should have octal permissions of 0755. When you are done, exit the ssh session and return to **workstation**.

```
[root@utility ~]# ls -ld /exports/hosted_engine/
drwxr-xr-x. 3 vdsm kvm 76 Jul 12 08:05 /exports/hosted_engine/
[root@utility ~]# exit
logout
Connection to utility closed.
[student@workstation ~]$
```

2. Download the RHEL 8 boot ISO file to the **/home/student/Downloads** directory on **workstation** in preparation for uploading the file to RHV-M.

```
[student@workstation ~]$ wget -P ~/Downloads \
http://materials.example.com/rhel-8.0-x86_64-boot.iso
...output omitted...
HTTP request sent, awaiting response... 200 OK
Length: 558891008 (533M) [application/octet-stream]
Saving to: '/home/student/Downloads/rhel-8.0-x86_64-boot.iso'
...output omitted...
```

3. Use the Administration Portal to verify RHV-M can access the shares previously confirmed on the **utility** server.

3.1. On **workstation**, open Firefox and navigate to **https://rhvm.lab.example.com**.

3.2. Click the **Administration Portal** link, and log in as **admin** using **redhat** as a password.

3.3. View existing storage domains by clicking **Storage** in the menu and selecting **Domains**. Locate the row where the **Domain Name** is **hosted_storage**, the **Domain Type** is Data (Master), and the **Storage Type** is NFS. The green triangle icon indicates that **hosted_storage** is active and the gold crown icon indicates that **hosted_storage** contains the hosted engine VM storage data.

3.4. View additional details by clicking the **hosted_storage** link in the **Domain Name** column. The **General** tab shows the size of the share and indicates that the path for this share comes from **utility.lab.example.com:/exports/hosted_engine**.

4. Use the RHV-M Administration Portal to view networking information for the RHV environment.

4.1. In the menu, click **Network** → **Networks**.

4.2. Select the **ovirtmgmt** link for the **Default** data center.

4.3. On the **ovirtmgmt (Default)** page, click the **Hosts** tab. Verify that **hosta.lab.example.com**, **hostb.lab.example.com**, **hostc.lab.example.com**, and **hostd.lab.example.com** are attached to the ovirtmgmt network.

4.4. On the **ovirtmgmt (Default)** page, click the **Clusters** tab. Verify that the **Default** cluster is listed. The icons in the **Network Role** column indicate that the **ovirtmgmt** network currently has the roles of Management, Display, Migration, and Default Route. In a later chapter, you will create an additional logical network to segregate migration traffic. Currently, all traffic (including storage traffic) uses the **ovirtmgmt** logical network.

5. Use the RHV-M Administration Portal to upload the RHEL 8 boot ISO file.

5.1. View existing disks by clicking **Storage** in the menu and then selecting **Disks**.

5.2. Open the Upload Image wizard by clicking **Upload** in the upper-right corner and then selecting **Start**

5.3. Select the RHEL 8 boot ISO by clicking the **Choose File** button. In the File Upload window, click **Downloads** in the menu and then double-click the **rhel-8.0-x86_64-boot.iso** file to choose that file. Locate the **rhel-8.0-x86_64-boot.iso** if it was downloaded to a different location.

5.4. In the Upload Image window, verify that Data Center is set to **Default** and Disk Profile is set to **hosted_storage** . Accept the default settings for everything else.

5.5. In a previous guided exercise, you should have downloaded the ovirt-engine certificate. Click the **Test Connection** button to verify this. If clicking the **Test Connection** button returns a green success box, then you are ready to upload. If clicking the **Test Connection** button returns an orange warning box, click the ovirt-engine certificate link within the warning box. Check the box next to Trust this CA to identify websites and then click the **OK** button. After you have done this, click the **Test Connection** button again. It should return a green success box.

> **Important**
>
> If you accidentally forget to check the box next to **Trust this CA to identify websites**, the following procedure will bring up that window again:
>
> 1. Open Preferences for Firefox and then select **Privacy & Security** in the left menu.
> 2. Scroll down to the Security section (at the bottom) and click the **View Certificates...** button.
> 3. In the Certificate Manager window, scroll down to lab.example.com, click rhvm.lab.example.com.34088 so that it is highlighted, and then click the **Delete or Distrust** button.
> 4. Back on the Preferences tab for **Privacy & Security**, scroll up to the Cookies and Site Data section and then click the **Clear Data...** button.
> 5. Accept the default selections and click the **Clear** button. Confirm your choice by clicking the **Clear Now** button in the new window that appears.

5.6. Click the **OK** button to start the upload process.

6. Verify that the ISO uploaded successfully.

   6.1. The **Disks** table should list a disk with an **Alias** of **`rhel-8.0-x86_64-boot.iso`**. Within a few minutes, the disk **Status** will update to OK.

Finish

On **`workstation`**, run the **`lab install-storage finish`** script to complete this exercise.

```
[student@workstation ~]$ lab install-storage finish
```

This concludes the guided exercise.

Creating a Linux Virtual Machine

Objective

After completing this section, students should be able to use the Administration Portal to manually create a RHEL virtual machine running in the Red Hat Virtualization environment.

Installing Console Components on Client Systems

Creating and managing a virtual machine requires occasional console access, especially during manual installations when console access is necessary to interact with the installation program's prompts. Virtual machines are configured to use either the SPICE or VNC protocols to manage remote connections. On the client systems from which you will access the virtual machines, you will install the Remote Viewer.

Remote Viewer for 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 opening 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 in the RHEL BaseOS repository. The *spice-xpi* Firefox plug-in is no longer required, since the external Remote Viewer application makes the connection.

Install the *virt-viewer* package on the system from which you access the Application Portal or the VM Portal. No browser restart is required, since no browser plugin is being installed.

```
[root@client ~]# yum -y install virt-viewer
```

Remote Viewer for Microsoft Windows

Remote Viewer for Microsoft Windows is available as a download from your RHV Manager system, with a 32-bit and a 64-bit version of the *virt-viewer* installer. Download the architecture of your choice, double-click the file, and answer the security warning and User Account Control prompts to complete the install.

- `https://`*`your-manager-fqdn`*`/ovirt-engine/services/files/spice/virt-viewer-x86.msi` (32-bit)
- (64-bit Windows) `https://`*`your-manager-fqdn`*`/ovirt-engine/services/files/spice/virt-viewer-x64.msi` (64-bit)

The usbdk driver is also available for installation on Windows clients. usbdk enables remote viewer exclusive access to USB devices on Windows. Installing usbdk requires Administrator privileges.

- `https://`*`your-manager-fqdn`*`/ovirt-engine/services/files/spice/usbdk-x86.msi` (32-bit)
- `https://`*`your-manager-fqdn`*`/ovirt-engine/services/files/spice/usbdk-x64.msi` (64-bit)

Creating a New Virtual Machine

To create a new virtual machine from the Administration Portal, click open the **Compute** → **Virtual Machines** menu and click the **New** button to provision a new virtual machine. The **New Virtual Machine** form defines the basic virtual machine parameters.

Virtual Machine Build Parameters

The top of the form defines that parameters necessary for RHV to provision resources. Choose the data center and cluster in which to deploy the new virtual machine. When deploying from prebuilt images, choose an existing template to use, which avoids performing an installation.

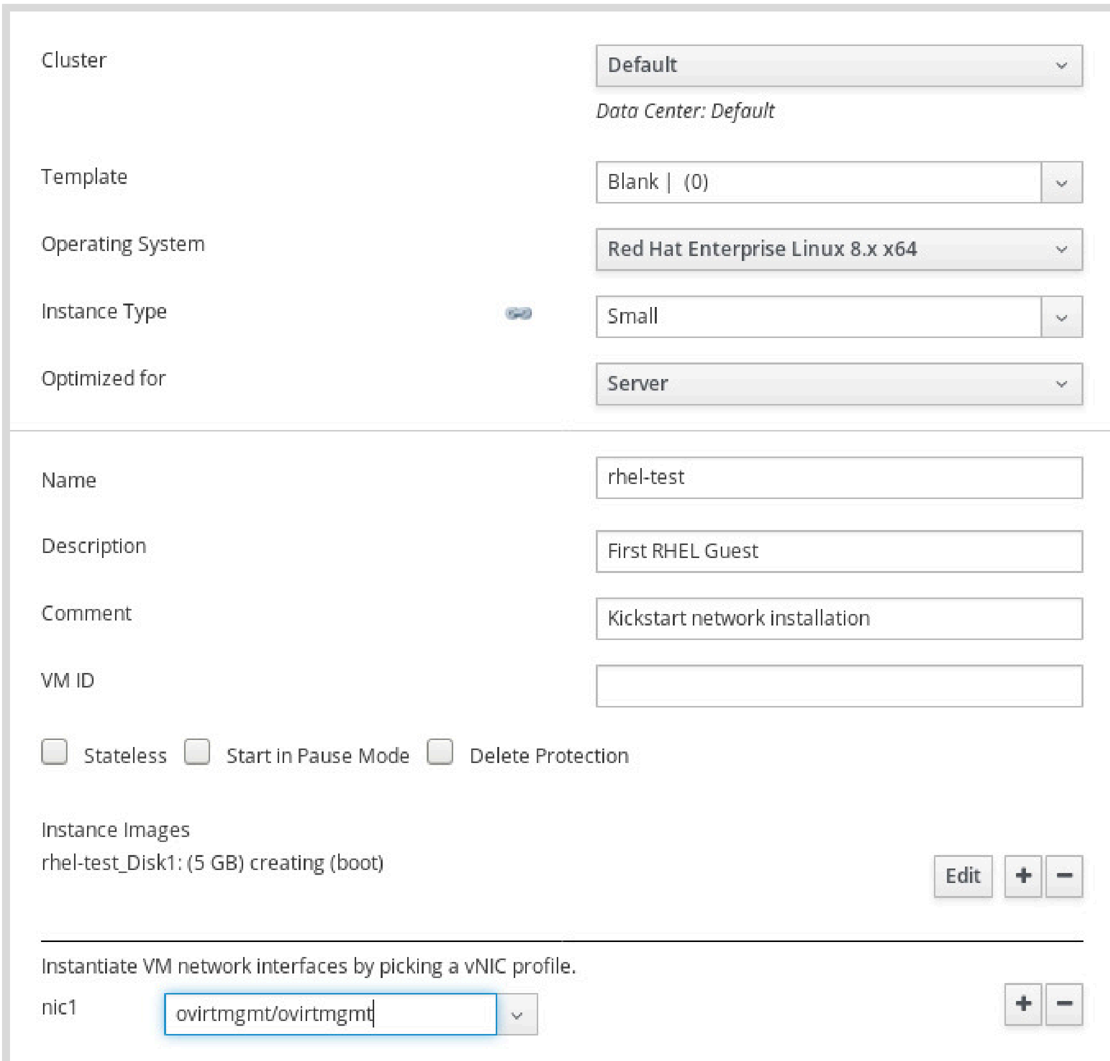

Figure 2.14: New Virtual Machine Window

Select the correct operating system and the sizing profile, which are predefined and customizable. You have three choices for optimizing the virtual machine build:

Desktop

- Enable all USB devices
- Enable the Smart Card device
- Use an Image with Thin Allocation
- Instance is Stateless

Server

- Disable the Sound Card device
- Use a Cloned Disk Image
- Instance is Not Stateless

High Performance

- Enable Headless Mode and enable Serial console
- Disable all USB devices
- Disable the Sound Card device
- Disable the Smart Card device
- Enable Pass-Through Host CPU
- Disable VM migration
- Enable IO Threads, Num Of IO Threads = 1
- Disable the Memory Balloon Device
- Enable High availability only for pinned hosts
- Disable the Watchdog device
- Enable paravirtualized Random Number Generator PCI (virtio-rng) device
- Enable Multi Queues per Virtual Interface
- Set the IO and Emulator threads pinning topology

In the middle of the **New Virtual Machine** window, specify a unique name, description and comment that will assist administrators in identifying this system among all those in production. The **Instance Images** options determine the system disk source, discussed below.

Near the bottom of the window, specify an existing logical network to connect to this virtual machine's NIC. Additional NICs can be created now or added after the machine has been built.

Virtual Machine Disk Selection

An instance can be deployed or cloned from an existing template or disk, booted from a stored disk image, or created on a new disk with an operating system installation. Using templates, and booting from stored disks, is covered later in this course. For this new installation example, the virtual machine will need a new, blank disk, selected by clicking the **Create** button.

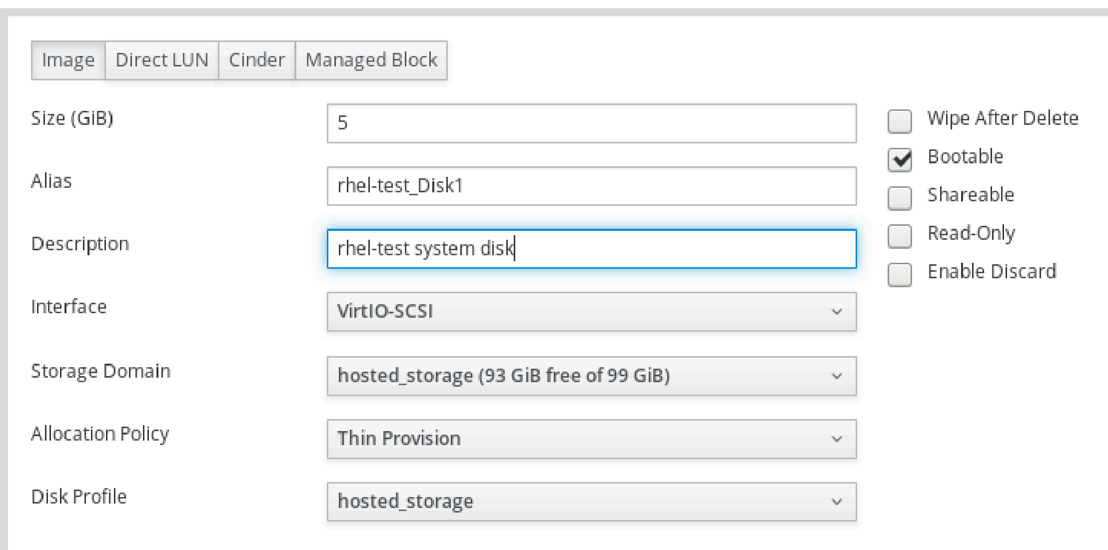

Figure 2.15: New Virtual Disk Window

Enter the **Size (GiB)** in gibibytes. The **Alias** is auto-generated by adding "_Disk" and a sequence number to your virtual machine name. A **Description** is not required, but is helpful when locating disks stored in a data domain when not currently attached to virtual machines.

You have a choice for **Interface**, indicating the disk controller protocol to use:

IDE
: The oldest and slowest, but also the most compatible with older operating systems and disks. This selection is not recommended for performance.

VirtIO
: Will be created as a **/dev/vdX** device. This driver is much faster than IDE, but has been surpassed in capabilities by the VirtIO-SCSI driver. VirtIO is recommended if advanced features are not needed, handling at most 25-30 devices due with one device per controller.

VirtIO-SCSI
: Will be created as a **/dev/sdX** device. The Virtio-SCSI driver significantly improves scalability compared to the virtio-blk driver it replaces. VirtIO-SCSI can connect directly to SCSI LUNs, and handles hundreds of devices on a single controller.

The **Storage Domain** stores this disk image and the virtual machine will boot from that stored image. The data domains listed are those attached to the cluster you specified earlier.

The **Allocation Policy** provides a choice of thick or thin provisioning.

Pre-allocated (thick)
: Provisions the full disk storage capacity when the disk is created. Space allocated to this disk is not available for any other use by the data store and other virtual machines. The performance of pre-allocated disks is higher, because write requests are not delayed by space extension requests.

Thin provisioned (sparse)
: Allocates only the initial required space, and adds space allocations as disk use grows. Thin-provisioned disks are created faster and use less storage space. Ongoing performance can be slower, because space allocations must be zeroed before data blocks can be used.

A **Disk Profile** defines the maximum level of I/O transactions and throughput for a virtual disk. Disk profiles are created based on storage profiles defined under data centers, and must be manually assigned to individual virtual disks for the profile to take effect. There is currently only one disk profile to select for the **hosted_storage** data domain.

Click the **OK** button to create this virtual disk and return to the **New Virtual Machine** window. Again click the **OK** button to close the **New Virtual Machine** dialog and create the virtual machine. The machine lists in the **Virtual Machines** table with a status of **Down**. Because this virtual machine's new system disk is empty, the next task is to perform an installation to the disk.

Booting the Virtual Machine

Now that the new virtual machine is configured, it is time to install an operating system. If the virtual machine started normally, it would default to booting from the empty disk. Instead, we will create a temporary (*Run Once*) set of parameters to start from a RHEL boot ISO.

Understanding Run Once

There are two options on the **Run** button: Run and Run Once. Run will start a virtual machine using the default boot configuration. Run Once presents a window for entering temporary boot parameters that will be used only once and discarded.

However, Run Once understands that installations typically require at least one reboot before finishing. To discard Run Once, the virtual machine must be shut down, not rebooted, and an operator must select Run to initiate a normal boot. This explains why installation failures that reboot without a shutdown will continue to boot using Run Once parameters. Run Once settings are discarded when the virtual machine is shut down.

Select the new virtual machine in the **Virtual Machines** window. In the **Run** drop down menu select **Run Once**. This opens the **Run Virtual Machine(s)** window. Click the the **+** button next to **Boot Options** to open the **Boot Options** screen.

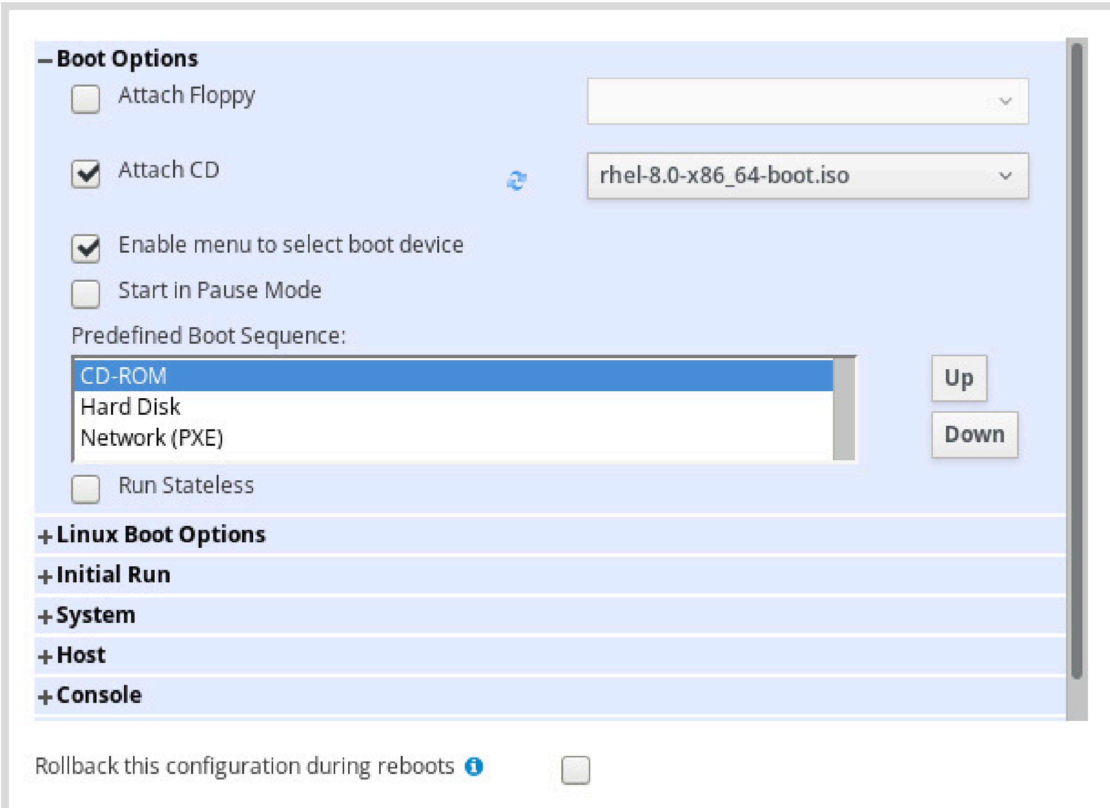

Figure 2.16: Run Virtual Machine(s) Boot Options

Virtual machines can be installed with the same methods used to install RHEL on physical systems:

Full RHEL DVD installation

The RHEL DVD ISO image is uploaded and stored in the data domain. The full ISO is attached as a virtual CD-ROM and the system boots to that CD-ROM to start a manual, interactive installation. The installation can be automated by specifying a Kickstart file location as a kernel boot parameter. The required BaseOS repository is on the Full DVD, and no network installation server is required.

RHEL Boot DVD ISO network installation

The RHEL Boot DVD ISO image is uploaded and stored in the data domain. The boot ISO is attached as a virtual CD-ROM and the system boots to that CD-ROM to start a manual, interactive installation. The installation can be automated by specifying a Kickstart file location as a kernel boot parameter. The Kickstart must specify an accessible network installation server URL to provide the BaseOS repository.

PXE network installation

Starting a network installation without a CD-ROM requires a PXE compliant network card. The NIC initiates a PXE-initiated installation from a network installation server URL, located using DHCP options. Network installations can also be interactive, but usually are automated with a Kickstart file store and found automatically on the network installation server.

Boot Options

In the **Boot Options** screen, enable the **Attach CD** checkbox. The drop-down field to the right lists all media images that have been loaded into this cluster's data domain. In this classroom, it is expected to default to the RHEL 8 boot ISO named **`rhel-8.0-x86_64-boot.iso`**.

In this classroom installation example, we chose the 550MB boot ISO option, to avoid downloading and storing the 8GB RHEL 8 DVD in the classroom data stores, and to provide an opportunity to practice a CDROM install with a manually entered Kickstart file boot parameter.

In the **Predefined Boot Sequence:** box, select the CD-ROM to highlight it, then use the **Up** button to move the CD-ROM to the top of the list. The virtual machine will now boot from that ISO.

> **Important**
>
> Clicking the **OK** button now would start the virtual machine and its installation. Because this installation method requires adding a Kickstart file kernel parameter, you need to access the virtual machine's console quickly, before the boot menu times out. Console access requires that you have installed the Remote Viewer. Verify the Remote Viewer installation before returning to the **Boot Options** screen.

Connecting to the Virtual Machine Console

To connect to the virtual machine's console, select the virtual machine row in the **Virtual Machines** table. When a virtual machine row is selected, the **Console** button will become available. Either click the **Console** button or right-click a virtual machine row and choose **Console** from the context menu. As observed below, the virtual machine is configure for SPICE functionality by default.

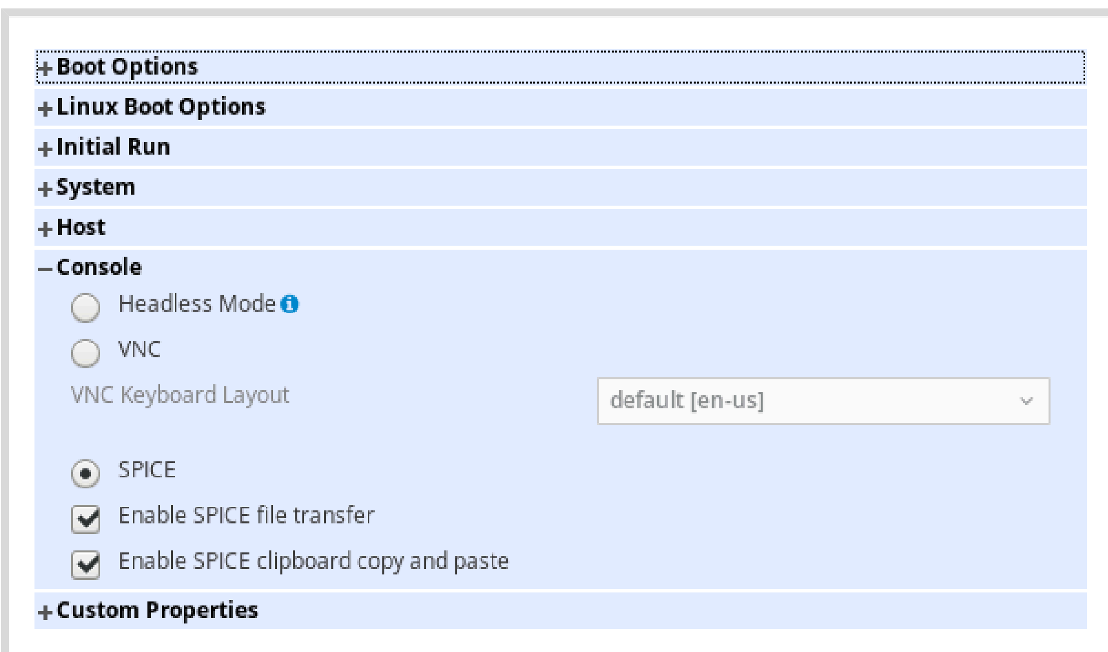

Figure 2.17: Run Virtual Machine(s) Console Settings

The Remote Viewer Connection File

When a virtual machine console is requested, a *virt-viewer* connection parameters file (**`console.vv`**) is sent to your browser. The first time you open a console connection, be sure to enable the **Do this automatically for files like this from now on.** so that your browser stores the application action for Remote Viewer files.

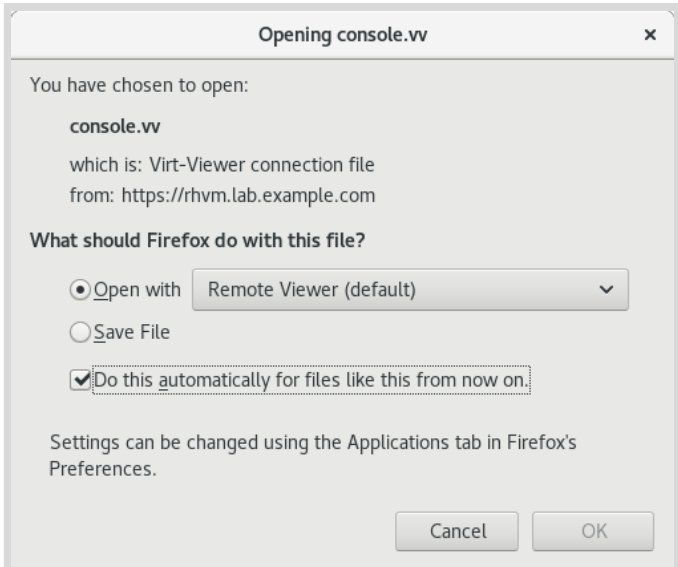

Figure 2.18: Opening a Remote Viewer File

The **`console.vv`** contains secure one-time connection information, including secure keys for connection encryption. An example file is displayed below.

```
[user@demo ~]$ cat console.vv
[virt-viewer]
type=spice
host=172.25.250.10
port=5900
password=onvRPyH3O9Hi
# Password is valid for 120 seconds.
delete-this-file=1
fullscreen=0
title=HostedEngine:%d
toggle-fullscreen=shift+f11
release-cursor=shift+f12
secure-attention=ctrl+alt+end
tls-port=5901
enable-smartcard=0
enable-usb-autoshare=1
usb-filter=-1,-1,-1,-1,0
tls-ciphers=DEFAULT
host-subject=O=lab.example.com,CN=hosta.lab.example.com
ca=-----BEGIN CERTIFICATE-----
...output omitted...
```

Adding the Kickstart File Kernel Parameter

When the Remote Viewer connection has been made, the virtual machine's console opens. The RHEL boot ISO will have booted, and presents an installation menu. Use your keyboard arrow keys to highlight **Install Red Hat Enterprise Linux 8.0.0**. When the line is selected, all that line's text will be white. With the line selected, hit your **`Tab`** key once, which causes the kernel booting parameters to display. Here is where you add the syntax (**`init.ks=`*`Kickstart_file_URL`***) to locate the Kickstart file, as in the example below.

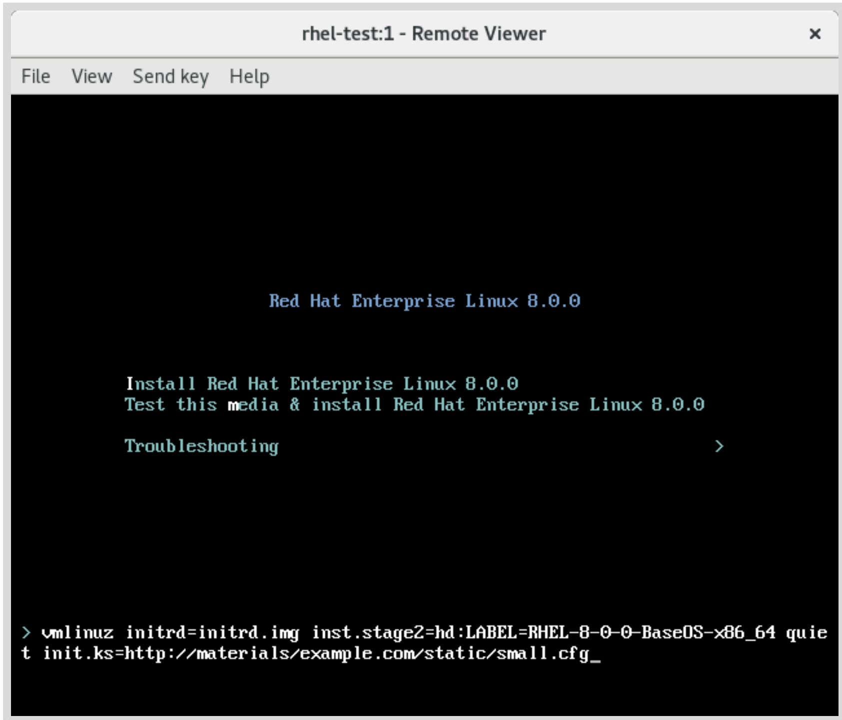

Figure 2.19: RHEL boot ISO menu with Kickstart Kernel Parameter

After entering the kernel parameters, press **`Enter`** to start the installation. Wait for the installation to complete. The virtual machine in the **Virtual Machines** table will change to an **UP** status..

Controlling Virtual Machines

Starting Virtual Machines

To start a virtual machine currently down, navigate to **Compute** → **Virtual Machines**. To start the virtual machine with its normal configuration, click the **Run** icon or right-click the virtual machine and select **Run** from the menu.

To start a virtual machine with non-default settings, select **Run Once** from the **Run** button pull-down to display the **Run Virtual Machine(s)** window.

Stopping Virtual Machines

There are three options to shut down a virtual machine:

- Shut down the virtual machine directly using operating system commands. For example, shut down a RHEL virtual machine by logging in and running **`poweroff`**.
- In the **Virtual Machines** table, right-click the virtual machine name and select **Shutdown** from the context menu. A virtual ACPI power button event is to the virtual machine. In some case,

the guest operating system may ignore this event (for example, when Microsoft Windows 7 is displaying a login screen).

- In the **Virtual Machines** table, right-click the virtual machine name and select **Power Off** from the context menu. This causes an ungraceful shutdown, similar to losing power. Use this method as a last resort, since pending file writes will be lost and data corruption could occur.

Suspending Virtual Machines

In the **Virtual Machines** table, select the virtual machine name and click the **Suspend** button (crescent moon icon.) A suspended virtual machine is put into **`Hibernate`** mode. The virtual machine's memory and CPU state are saved to disk and the machine stops processing.

To resume a suspended virtual machine, right-click the virtual machine and select **Run** from the menu.

Removing Virtual Machines

To remove a virtual machine, shut it down gracefully, then right-click the virtual machine and select **Remove** from the context menu. Click **OK** in the confirmation dialog box to confirm removing the virtual machine.

Warning

When removing a virtual machine, all resources associated with the virtual machine are deleted, including all virtual disks and virtual network cards. When a virtual disk is removed, the disk and its contents are erased permanently, as a security feature.

References

Further information is available in the Installing Linux Virtual Machines chapter of the *Virtual Machine Management Guide* for Red Hat Virtualization 4.3 at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index

Guided Exercise

Creating a Linux Virtual Machine

In this exercise, you will create a new Red Hat Enterprise Linux virtual machine in your Red Hat Virtualization environment.

Outcome

You should be able to create a new, functioning Red Hat Enterprise Linux virtual machine.

Before You Begin

Log in to **`workstation`** as **`student`** using **`student`** as the password.

On **`workstation`**, run the **`lab install-vm start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab install-vm start
```

1. On **`workstation`** confirm that you have access to the Kickstart file. Use Firefox to access `http://materials.example.com/static/small.cfg` to review the contents of the Kickstart file.

2. Confirm that virt-viewer is installed. On **`workstation`**, install the virt-viewer client. The password for the **sudo** is student.

```
[student@workstation ~]# sudo yum -y install virt-viewer
```

> **Important**
> On **`workstation`** increase the screen resolution to 1920x1080. Different dialog windows displayed in this chapter and in the later chapters require a higher resolution.

3. Using the URL `https://rhvm.lab.example.com`, log in to Red Hat Virtualization Administration Portal as user **`admin`** with the password **`redhat`** in the authentication domain **`internal`**.

4. Navigate to **Virtual Machines** page by clicking Compute in the menu bar and then selecting Virtual Machines.

5. To create a new Red Hat Enterprise Linux virtual machine click the **New** button. The **New Virtual Machine** dialog displays.

   5.1. In the **Cluster** section, choose the **Default** cluster.

   5.2. As the **Operating System**, select **Red Hat Enterprise Linux 8.x x64**.

5.3. Click the **Instance Type** drop-down list and choose the **Small** type.

5.4. Click the **Optimized for** drop-down list and choose the **Server** type.

5.5. In the **Name** field, type in the name for the virtual machine as **`rhel-test`**.

5.6. In the **Description** field, type in the description for the virtual machine as **`First RHEL Guest`**.

5.7. To create an image for the new VM, click the **Create** button on the **Instance Images** line. Specify the **Size** of the image as **5** GB. Leave all the other options with their default values and confirm by clicking the **OK** button.

5.8. In the bottom part of the dialog window, choose a network interface by clicking on the **Please select an item** list next to the nic1 network card. From the list, choose the **ovirtmgmt (ovirtmgmt)** network.

5.9. To confirm the creation of this virtual machine, click the **OK** button.

> **Note**
> Selecting an instance type populates memory and CPU values for the virtual machine. These values are visible from the System tab when viewing advanced options. Although choosing either Server or High Performance from the Optimized for menu will switch the Instance Type menu back to Custom, the memory and CPU values for the instance type that you selected will remain the same.

6. Notice that on the list of available virtual machines the new **`rhel-test`** virtual machine appeared.

7. Modify the configuration of the virtual machine so that it boots using the Red Hat Enterprise Linux installation image in the virtual **CD-ROM/DVD-ROM** device, and boot it.

7.1. Highlight the **`rhel-test`** virtual machine from the list of virtual machines. Click the drop down next to the **Run** button to display the **Run Once** option.

7.2. Click the **+** icon next to **Boot Options** to open the boot options dialog.

7.3. Click the check box next to the **Attach CD**. From the drop-down list of available ISO files, choose the Red Hat Enterprise Linux installation ISO **`rhel-8.0-x86_64-boot.iso`**. This “inserts” the ISO file into the virtual CD-ROM drive.

7.4. In the **Predefined Boot Sequence** list, choose the **CD-ROM** by clicking on it. With the **CD-ROM** highlighted, click the **Up** button once, to bring the **CD-ROM** drive to the top of the **Boot Sequence** list.

7.5. To confirm your changes and boot the virtual machine from the Red Hat Enterprise Linux installation ISO, click the **OK** button.

8. Open the virtual machine console and start a Kickstart installation of the virtual machine.

> **Warning**
> In case you open the console too late, at a stage where you can not modify the kernel argument line anymore, simply reboot the virtual machine by right-clicking the name and choosing **Reboot**. After the virtual machine reboots you will be able to access the console and follow along with the rest of the guided exercise.

8.1. Once the virtual machine has started and the console button becomes available, click the active console button to start the Red Hat Enterprise Linux installation. Accept the opening of the **console.vv** file using Remote Viewer by clicking the **OK** button.

> **Note**
> If Remote Viewer prompts for your confirmation to inhibit shortcuts, click Allow.

8.2. In the GRUB menu, edit the installer options by choosing **Install Red Hat Enterprise Linux 8.0.0** and press the **Tab** key to append the installer.

8.3. The editor should open with the cursor automatically positioned at the end of the existing kernel arguments. At the end of the kernel argument line, add a space and the argument **inst.ks=http://materials.example.com/static/small.cfg** to specify the location of your Kickstart file.

8.4. Press **Enter** to start the Kickstart installation of the virtual machine.

9. Watch for the installation to complete. After the installation completes, the virtual machine automatically reboots from the CD.

10. Close the console and power off the VM by right-clicking the VM entry and selecting **Power Off**. Confirm that you really want to power off the machine.

11. To test the Red Hat Enterprise Linux virtual machine, power on your new virtual server by right-clicking the machine name in the overview and selecting **Run**.

12. Open the console by right-clicking the machine name and selecting **Console**.

13. After the virtual machine starts up, log in with the **root** user account with a password of **redhat**.

14. Log out of the virtual machine.

Finish

On **workstation**, run the **lab install-vm finish** script to complete this exercise.

```
[student@workstation ~]$ lab install-vm finish
```

This concludes the guided exercise.

Lab

Installing and Configuring Red Hat Virtualization

Performance Checklist

In this lab, you will create a new virtual machine in your Red Hat Virtualization environment using a qcow2 image.

Outcomes

You should be able to:

- Upload a qcow2 image to an existing data storage domain.
- Create a new virtual machine using an unattached disk in the data storage domain.
- Verify that your virtual machine starts correctly.

Before You Begin

Log in to **`workstation`** as **`student`** using **`student`** as the password.

On **`workstation`**, run the **`lab install-review start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab install-review start
```

1. On **`workstation`**, download the qcow2 image located at `http://materials.example.com/rhel-8.0-lab.qcow2` to the **`/home/student/Downloads/`** directory.
2. Use Firefox to access the RHV-M Administration Portal located at `https://rhvm.lab.example.com/ovirt-engine/webadmin/`. Log in as user **`admin`** with a password of **`redhat`**. The profile should be set to **`internal`**.
3. Upload the qcow2 file to the **`hosted_storage`** data storage domain as **`rhel8-lab`**.
4. Create a new virtual machine according to the following requirements:
   - Cluster: Default
   - Template: Blank | (0)
   - Operating System: Red Hat Enterprise Linux 8.x x64
   - Instance Type: Custom
   - Optimized for: Server
   - Name: rhel-lab-test
   - Instance Images: Attach the rhel8-lab disk (ensure that the check box under the OS column is checked)
   - nic1: ovirtmgmt/ovirtmgmt
5. Verify that the **`rhel-lab-test`** virtual machine starts correctly.

Evaluation

On **workstation**, run the **lab install-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab install-review grade
```

Finish

On **workstation**, run the **lab install-review finish** script to complete this exercise.

```
[student@workstation ~]$ lab install-review finish
```

This concludes the guided exercise.

Solution

Installing and Configuring Red Hat Virtualization

Performance Checklist

In this lab, you will create a new virtual machine in your Red Hat Virtualization environment using a qcow2 image.

Outcomes

You should be able to:

- Upload a qcow2 image to an existing data storage domain.
- Create a new virtual machine using an unattached disk in the data storage domain.
- Verify that your virtual machine starts correctly.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab install-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab install-review start
```

1. On **workstation**, download the qcow2 image located at `http://materials.example.com/rhel-8.0-lab.qcow2` to the **/home/student/Downloads/** directory.

   1.1. Use the wget command to download the qcow2 file:

```
[student@workstation ~]$ wget -P ~/Downloads/ \
http://materials.example.com/rhel-8.0-lab.qcow2
...output omitted...
HTTP request sent, awaiting response... 200 OK
Length: 1864302592 (1.15G)
Saving to: '/home/student/Downloads/rhel-8.0-lab.qcow2'
...output omitted...
```

2. Use Firefox to access the RHV-M Administration Portal located at `https://rhvm.lab.example.com/ovirt-engine/webadmin/`. Log in as user **admin** with a password of **redhat**. The profile should be set to **internal**.

3. Upload the qcow2 file to the **hosted_storage** data storage domain as **rhel8-lab**.

   3.1. In the menu bar, click **Storage** and then select **Disks**.

3.2. In the upper-right corner, click **Upload** and then select **Start**.

3.3. Use the **Choose File** button to select the previously downloaded qcow2 file. In the **File Upload window**, click **Downloads** in the left navigation and then double-click the **`rhel-8.0-lab.qcow2`** file to select it. If you downloaded **`rhel-8.0-lab.qcow2`** to a different location, you will need to find it.

3.4. In the **Upload Image window**, use **`rhel8-lab`** in both the **Alias** and **Description** fields. Verify that **Data Center** is set to **`Default`** and that **Disk Profile** is set to **`hosted_storage`**. Accept the default settings for everything else.

3.5. In a previous guided exercise, you should have downloaded the **`ovirt-engine`** certificate. Click the **Test Connection** button to verify this. If clicking the **Test Connection** button returns a green success box, then you are ready to upload. If clicking the **Test Connection** button returns an orange warning box, click the **ovirt-engine certificate link** within the warning box. Check the box next to **Trust this CA to identify websites** and then click the **OK** button. After you have done this, click the **Test Connection** button again. It should return a green success box.

Important

If you accidentally forget to check the box next to **Trust this CA to identify websites**, the following procedure will allow you to bring up that window again:

1. Open Preferences for Firefox and then select **Privacy & Security** in the left navigation.
2. Scroll down to the **Security** section (at the bottom) and click the **View Certificates...** button.
3. In the **Certificate Manager** window, scroll down to **`lab.example.com`**, click **`rhvm.lab.example.com.34088`** so that it is highlighted, and then click the **Delete or Distrust** button.
4. Back on the **Preferences** tab for **Privacy & Security**, scroll up to the **Cookies and Site Data** section and then click the **Clear Data...** button.
5. Accept the default selections and click the **Clear** button. Confirm your choice by clicking the **Clear Now** button in the new window that appears.

3.6. Click the **OK** button to start the upload process.

4. Create a new virtual machine according to the following requirements:

- Cluster: Default
- Template: Blank | (0)
- Operating System: Red Hat Enterprise Linux 8.x x64
- Instance Type: Custom
- Optimized for: Server
- Name: rhel-lab-test
- Instance Images: Attach the rhel8-lab disk (ensure that the check box under the OS column is checked)
- nic1: ovirtmgmt/ovirtmgmt

4.1. Within the RHV-M Administration Portal, click **Compute** in the menu bar and then select **Virtual Machines**.

4.2. Click the **New** button.

4.3. Ensure Cluster is set to Default.

4.4. Ensure Template is set to Blank | (0).

4.5. Ensure Operating System is set to Red Hat Enterprise Linux 8.x x86.

4.6. Leave Instance Type set as Custom.

4.7. Ensure Optimized for is set to Server.

4.8. Use rhel-lab-test as the name of the virtual machine.

4.9. For Instance Images, click the **Attach** button. In the **Attach Virtual Disks** window, click the radio button for **`rhel8-lab`** and ensure the check box in the **OS** column is checked. Click the **OK** button to attach the disk.

4.10. For **`nic1`**, ensure **`ovirtmgmt/ovirtmgmt`** is selected.

4.11. Click the **OK** button to create the virtual machine.

5. Verify that the **`rhel-lab-test`** virtual machine starts correctly.

5.1. Within the RHV-M Administration Portal, click **Compute** in the menu bar and then select **Virtual Machines**.

5.2. Click the blank **Comment** column for the row containing the **`rhel-lab-test`** virtual machine. The entire row should now be highlighted in blue.

5.3. Click the **Run** button to start the **`rhel-lab-test`** virtual machine.

5.4. Once it becomes accessible, click the **Console** button. If prompted, open the virtual machine using Remote Viewer.

5.5. Verify that you can log in as user **`root`** with a password of **`redhat`**.

5.6. Shut down the virtual machine using the **`systemctl poweroff`** command.

Evaluation

On **`workstation`**, run the **`lab install-review grade`** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab install-review grade
```

Finish

On **`workstation`**, run the **`lab install-review finish`** script to complete this exercise.

```
[student@workstation ~]$ lab install-review finish
```

This concludes the guided exercise.

Summary

In this chapter, you learned:

- RHV-H leverages Anaconda for installation, LVM for image management, and web console for administration and monitoring.
- The **`Administration Portal`** provides various controls for the management of physical and virtual resources in a RHV environment. RHV-M also exposes REST APIs and SDKs for various programming languages.
- A data domain is a centrally accessed repository for virtual machines disk and images files, ISO files, and other data accessible to all hosts in a RHV data center. NFS, iSCSI, as well as other protocols, can be used for storage domains.
- The Remote Viewer application is used to access the consoles of RHV virtual machines from client systems. On a **`Red Hat Enterprise Linux`** client system, the *spice-xpi* package installs Remote Viewer and all required plugins.

Chapter 3

Creating and Managing Data Centers and Clusters

Goal	Organize hypervisors into groups using data centers and clusters.
Objectives	• Explain the purpose of a data center for organizing hosts, and create a new data center. • Explain how clusters are used to group hosts in a data center, and create a new cluster.
Sections	• Creating and Managing Data Centers (and Guided Exercise) • Creating and Managing Clusters (and Guided Exercise)
Lab	Creating and Managing Data Centers and Clusters

Creating and Managing Data Centers

Objectives

After completing this section, you should be able to explain the purpose of a data center for organizing hosts, and create a new data center.

Introduction to Data Centers

The top-level organizational object in Red Hat Virtualization is the *data center*. A data center contains all the physical and logical resources in a single, managed, virtual environment; it is a collection of resources that includes clusters, hosts, logical networks, and storage domains.

A single Red Hat Virtualization data center is a self-contained virtualization environment. It may consist of:

- Resources that are all in a particular physical data center at a particular location.
- A set of systems and storage belonging to a particular business unit of the organization.
- Some other arbitrary division or organization selected by the administrator.

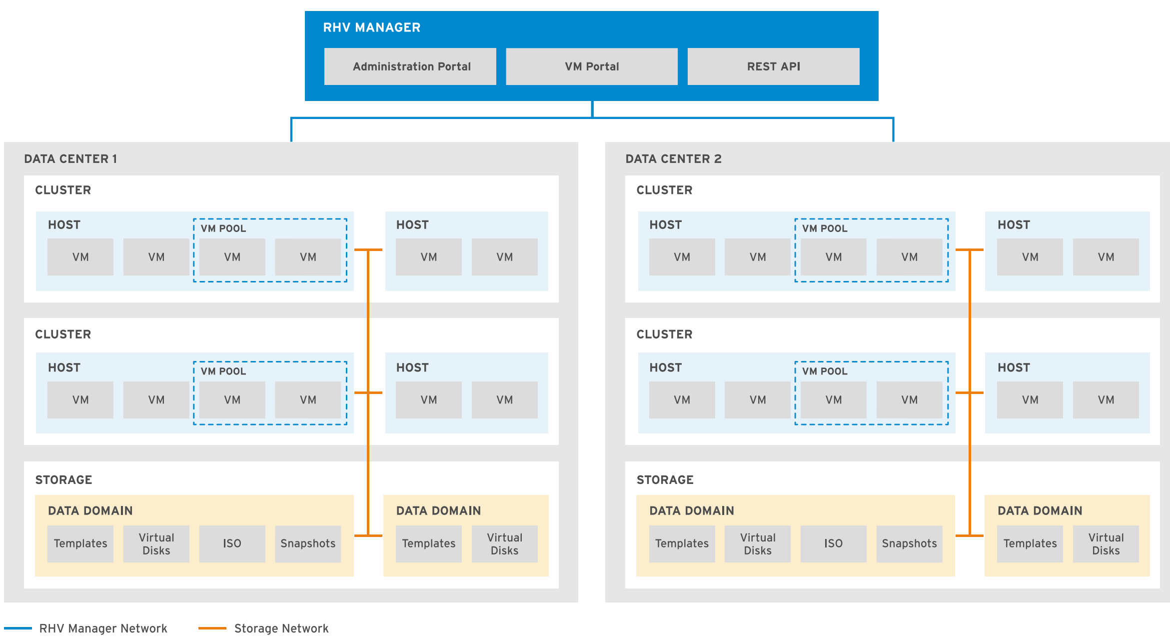

Figure 3.1: Data centers

A data center can be used to isolate resources belonging to an organization or group from other organizations or groups that normally would not have access to those resources. This feature allows you to restrict access to data and servers to a specific user group.

One characteristic of a Red Hat Virtualization data center is that all hosts and clusters in that data center share the same storage and must be able to access that storage. Therefore, if some hosts cannot or should not be configured to access certain storage resources, then those hosts need to be in clusters in a separate Red Hat Virtualization data center.

A data center named **`Default`** is created automatically. Additional data centers can be created using the Administration Portal.

Creating a New Data Center

The following procedure details how to create a new data center using the Administration Portal while logged in as the **`admin`** user. This procedure creates a data center that does not yet have any associated resources, such as storage domains or clusters. Resources can be associated with the data center later.

Click the **Compute** tab, then the **Data Centers** tab to display the options available to manage data centers. Initially, the **`Default`** data center is empty and has no resources assigned. Resources like storage and hosts can be assigned to the data center after you create it.

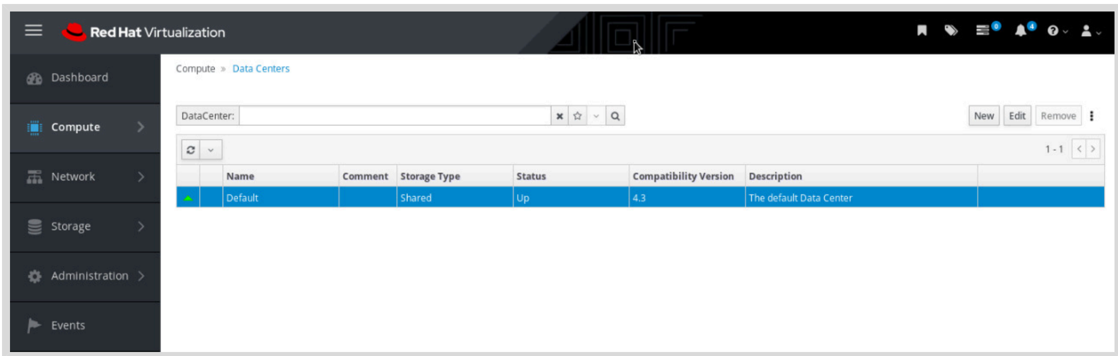

Figure 3.2: Available data centers

Click **New** to create a new data center. In the pop-up window **New Data Center**, enter the name you want to use for the data center in the **Name** field.

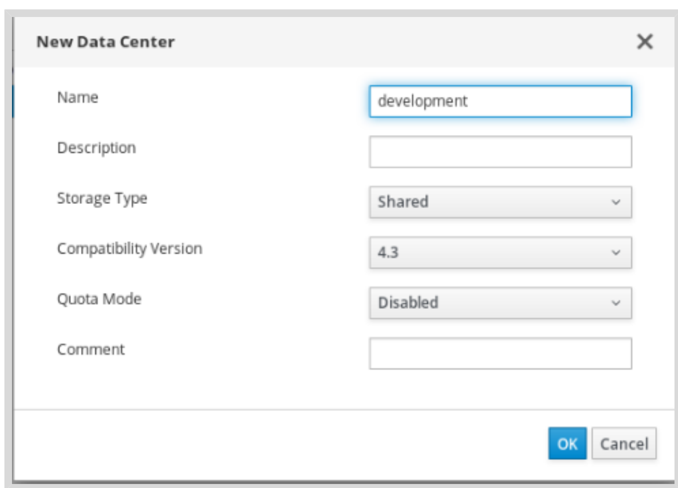

Figure 3.3: Data center's name

Use **Storage Type** menu to select the storage type to use in the new data center.

There are two options available: **Shared** and **Local**. In most cases, you should select **Shared**, which allows the data center to contain multiple clusters and hosts that can run virtual machines. If you select **Local**, the data center is restricted to having a single cluster consisting of exactly one host, but the data center's storage may be provided by that host's local file system.

Select the Red Hat Virtualization **Compatibility Version** supported by the data center. For a new data center, select the latest version available.

When Red Hat Virtualization is upgraded to a new version, existing data centers, clusters, and hosts can still be configured to use an older product protocol. Existing hosts and clusters assigned to the data center must be able to support the selected compatibility version. This ensures that all data center clusters support a particular set of Red Hat Virtualization features.

Red Hat Virtualization supports quotas that you may use to limit usage of memory, CPU, and storage resources. A data center can be configured to use these quota settings using the **Quota Mode** menu. There are three options available:

- **Disabled** - turns off quota-based restrictions.
- **Audit** - set quota limits but does not enforce them.
- **Enforced** - restrict resource use based on quota settings.

Like most data center settings, this selection can be changed after you create the data center. When done, click the **OK** button. A new pop-up window, titled **Data Center - Guide Me** displays.

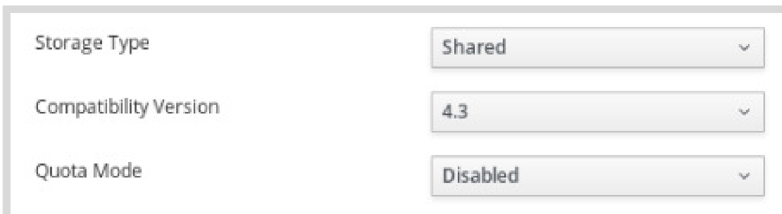

Figure 3.4: Data center's quota mode

The **Data Center - Guide Me** wizard provides an easy way to configure the other resources that must be assigned to the data center in order for it to be useful. This resources include hosts that run virtual machines, clusters to organize those hosts, and storage domains for virtual machine disks and installation media. Each button opens up a new wizard to configure each resource.

If you are setting up a new data center, then those resources might not yet be configured. In that case, click the **Configure Later** button to complete configuration of the data center later.

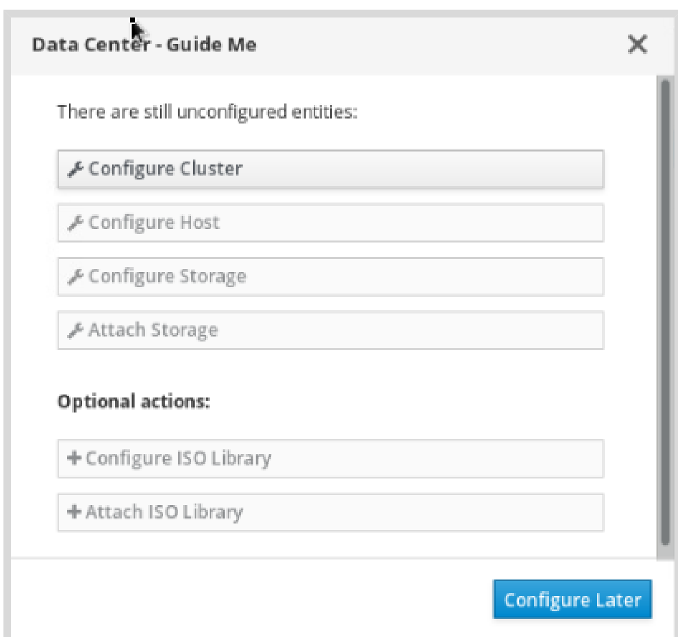

Figure 3.5: Data center resource configuration wizard

When first created, your new data center's status is **`Uninitialized`**. This status changes to **`Up`** when the resources are assigned to the data center and Red Hat Virtualization confirms that the data center can use them.

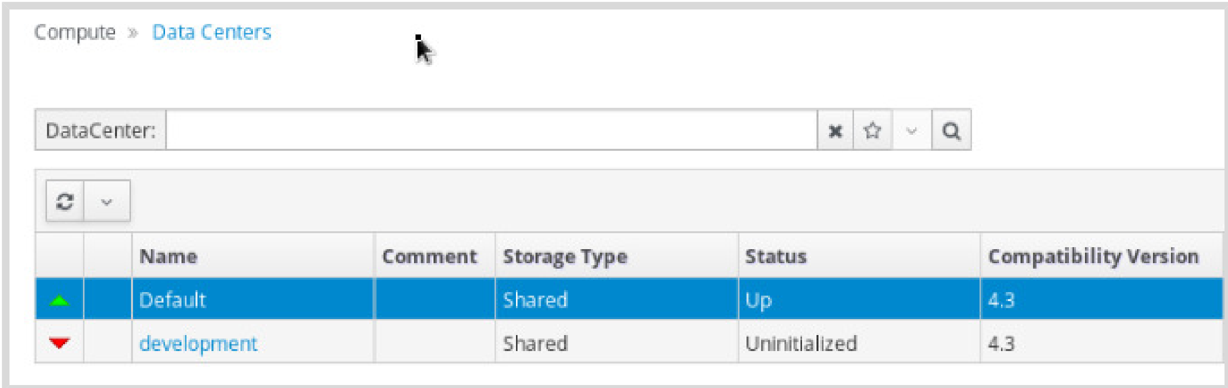

Figure 3.6: Uninitialized status

Important

Information on how to configure resources and assign them to an existing data center is covered in upcoming sections of this course.

References

Further information is available in the Data Centers chapter of the *Administration Guide* for Red Hat Virtualization; at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#chap-Data_Centers

Guided Exercise

Creating and Managing Data Centers

In this exercise, you will create a new data center in your Red Hat Virtualization environment.

Outcomes

You should be able to create a new data center.

Before You Begin

Log in to the **workstation** virtual machine (VM) as the **student** user, with **student** as the password.

On the **workstation** VM, run the **lab datacenters-createdc start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab datacenters-createdc start
```

1. On **workstation**, open the Firefox web browser. Log in to the **Red Hat Virtualization Manager** Administration Portal using the `https://rhvm.lab.example.com` URL. The username is **admin** and the password is **redhat**.

2. Create a new data center, named **development**.

   2.1. On the menu, click **Compute**, then choose **Data Centers** from the menu.

   2.2. At the top right hand side of the window, find and click the **New** button.

   2.3. In the **New Data Center** window, enter **development** in the **Name** field. Keep the default values for the other fields. Click the **OK** button to create the data center.

   2.4. In the pop-up window titled **Data Center - Guide Me**, click **Configure Later**.

3. In the **Compute >> Data Centers** window, verify that the value of the **Status** field for the **development** data center is **Uninitialized**.

Finish

On **workstation**, run the **lab datacenters-createdc finish** script to complete this exercise.

```
[student@workstation ~]$ lab datacenters-createdc finish
```

This concludes the guided exercise.

Creating and Managing Clusters

Objective

After completing this section, you should be able to explain how clusters group hosts in a data center, and the information required to create a new cluster.

Introduction to Clusters

A *cluster* is a group of hosts in a single data center with the same architecture and CPU model. A cluster is a *migration domain*, such that the cluster's virtual machines may only live-migrate to other hosts defined within that same cluster. All cluster hosts must be configured with the same resources, including logical networks, storage domains, and sufficient computing capacity.

When preparing to build a cluster, or when adding physical hosts to a cluster, it is recommended that all hosts use the same CPU model. CPU features are detected by an initializing application and expected to remain for the duration of that application's runtime. Live-migrating applications expect those same CPU features to exist and function after moving to another host, and would fail if the destination host does not support the required features. This limitation does not apply to stopped virtual machines that are cold-migrated or exported to other hosts or clusters, since those virtual machines will re-detect available CPU features when restarted in a new host.

Matching CPU Features in Cluster Hosts

Clusters with a mix of CPU models must restrict hosts to a CPU feature set (family) shared by all cluster host CPUs, determined by the oldest physical CPU family in the cluster. The most efficient performance is achieved by populating all cluster hosts with the same physical CPU model. In mixed-CPU clusters, applications may not be able to take advantage of newer CPU features since they are not shared by all cluster CPUs, thus reducing potential performance. The cluster CPU type (family) is set when the cluster is created. Hosts that do not meet that minimum CPU requirement can not be added to the cluster, unless the cluster's CPU type is reconfigured.

> **Important**
> Red Hat recommends standardizing the make, model, hardware, and firmware of all hosts assigned to the same cluster, thus assuring predictable virtual machine performance characteristics on any host in the cluster.

Multiple Clusters per Data Center

A data center may have multiple clusters, to support different application use case requirements. Use clusters to segregate hardware and workloads into classes or groups, as needed. One use case is to segregate application components, such as running front-end web and back-end database applications or middleware in different clusters. Another example is to group workloads according to performance tuning, data isolation and security requirements.

All clusters in a same data center share access to the same storage domains and logical networks. Data centers configured as **`Local`** use only local disks and are restricted to the single cluster and host that manage the local storage. Virtual machines in local data centers cannot live-migrate and are unsuitable for resilient production use.

Creating a New Cluster

During initial installation, RHV creates an initial cluster named **`Default`** in the initial **`Default`** data center. Additional clusters can be created, by users with sufficient privileges, using the Administration Portal, the REST API, or from the command line. The data center administrator must prepare to make multiple decisions when creating a new cluster. After the cluster is created, hosts that meet the cluster's specifications can be added to it, or removed from it, as needed.

Before creating the cluster, learn if all hosts have the same CPU type, or determine the oldest CPU family in use, by view the host hardware information found under the **General** tab's Hardware link. Record the CPU model and CPU Type for each host expected to be in this cluster, to verify that all CPUs are identical, or can share the same CPU Type family.

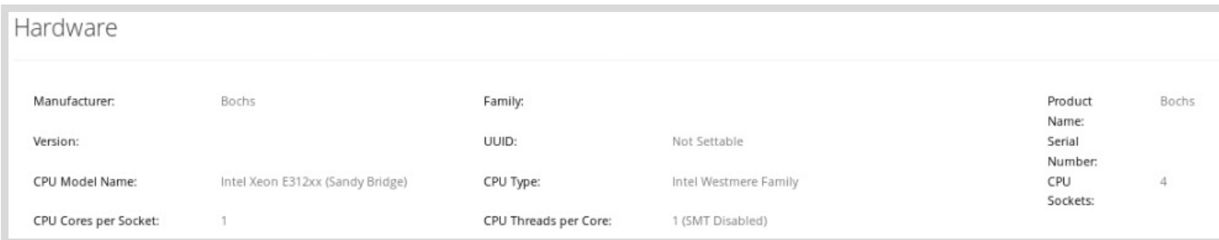

Figure 3.7: Host's CPU type

Using the Form in the Administration Portal

The form for creating a new cluster is found on the Administration Portal's **Clusters** tab, and includes multiple required fields for successfully creating a new cluster. In addition to setting a useful cluster name, description, and comment, an administrator will select the data center to which this cluster belongs. Previous configuration choices for the selected data center, such as the data center's storage type, will determine the available choices on this form.

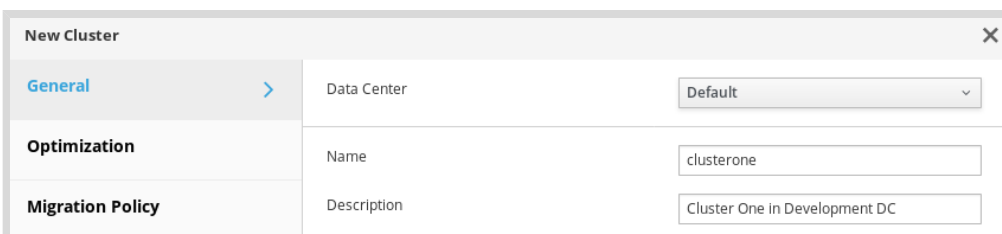

Figure 3.8: New cluster form

Selecting a Host-shared Management Network

The **Management Network** menu displays all currently available logical networks on this host. At minimum, the default logical network created during the RHV installation is available. If a different logical network is required, but not yet available, you must first create the logical network and assign it to all hosts that will be in this cluster, before returning here to select it. Alternatively, choose any available logical network now, and change the configuration to a another logical network later. Additional logical networks are recommended to segregate between management traffic, and virtual machine workload or migration traffic.

Choosing the Best Performing, Common CPU Type

As previously discussed, choose the **CPU Architecture** and the best shared **CPU Type** from your collected host information. This important setting forces a CPU type feature set upon all hosts in the cluster. Hosts that do not meet this minimum will not be allowed to join. Hosts with a better CPU will be feature-restricted down to this CPU type's feature set.

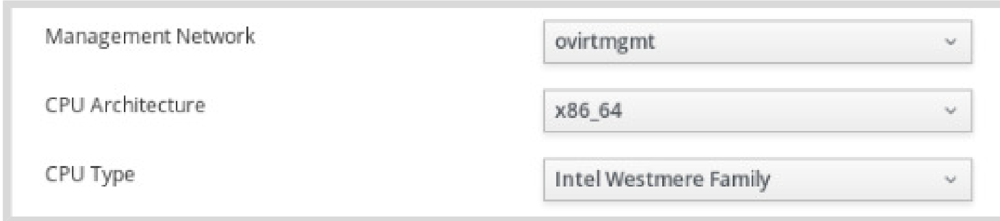

Figure 3.9: Cluster's CPU type

Setting the API Compatibility Version

The compatibility setting specifies the common API protocol version supported among all RHV hosts and management engines that will communicate together as a single RHV infrastructure. When building a new infrastructure, choose the most recent version setting. When adding new clusters to an existing infrastructure, or when adding systems during RHV upgrades or expansion, choose the same compatibility version currently configured throughout the existing infrastructure. During upgrades, newer components can be installed to temporarily function using older protocols to avoid disrupting the virtualization environment.

Networking and Service Infrastructure Options

Two network switch types are available, but only one can be configured at a time. Legacy **Linux Bridge** configuration is common, but Open vSwitch (**OVS (experimental)**) has become popular due to the flexibility and capabilities of software-defined networking. The OVS option is a Technology Preview, and is not yet supported for production use. When supported in a future release, you would choose OVS if you expect to integrate your Red Hat Virtualization infrastructure with OVS networks provided by Red Hat OpenStack Platform (RHOSP) or Red Hat OpenShift Container Platform (RHOCP).

Set when hosts in this cluster will be used to run virtual machines. The engine host cluster that manages the RHV-M self-hosted VM can deselect this option to avoid contention on egnine hosts between the RHV-M and production workload VMs. When building a normal workload cluster, enable the **`virt`** service.

Hosts in this cluster will be used as Red Hat Gluster Storage server nodes, and not for running virtual machines. You cannot add a Red Hat Enterprise Virtualization Hypervisor host to a cluster with this option enabled.

It is common for hosted virtual machines to be brought down safely, and hosts placed in maintenance mode to perform diagnostics or scheduled, or unexpected, maintenance. To help track this activity, the cluster has two settings, one each for virtual machines and hosts, to require that a reason be entered when either shutting down or placing hosts in maintenance mode. When this feature is enabled, the reason entered is logged with the event.

Using an External Random Number Generator

Some application workloads require significant amounts of entropy to operate properly. To assist this need, the cluster can be configured to utilize the **`/dev/hwrng`** hardware-based random number generator instead of the default **`/dev/urandom`** device. To select and use this hardware-based entropy source requires that every host in the cluster have that hardware device available and functioning. With this option selected, new hosts added to cluster must also support that hardware device.

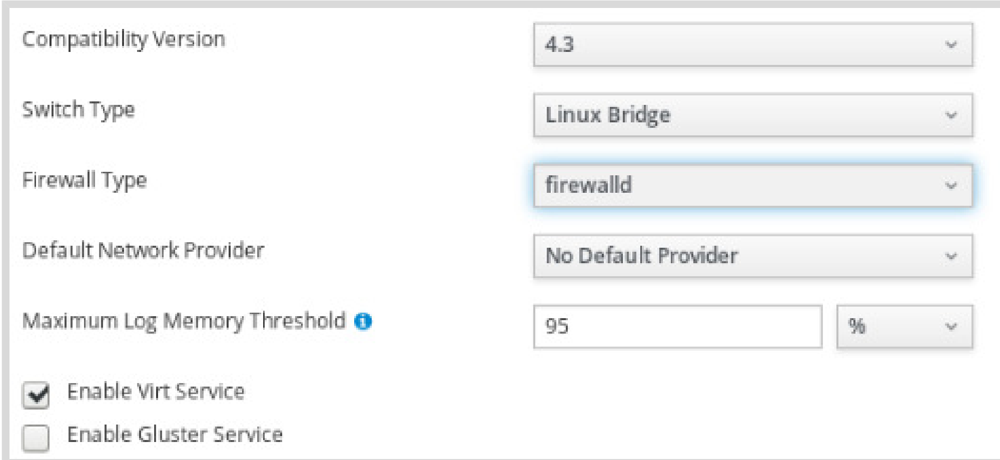

Figure 3.10: Cluster network and service options

Site-specific Configuration Options

Other cluster configuration choices are available in additional tabs on the **New Cluster** form, but other settings have defaults that do not require modification during initial cluster creation. If you are familiar with the use those settings in your RHV environment, you can modify them, now or at a later time. Some options are discussed later in this course.

Optimization
: Memory page sharing threshold, CPU thread handling, and memory ballooning.

Migration Policy
: Rules for determining when existing VMs migrate automatically between hosts. For load balancing, as an example.

Scheduling Policy
: Rules for selecting on which host to new virtual machine will start.

Console
: Choosing console connection protocols and proxies, such as SPICE.

Fencing
: Actions taken when managed hosts fail, to ensure that attached storage is not corrupted.

MAC Address Pool
: Defining the MAC address range to be used for NICs on VMs in this cluster, other than using the data center default pool.

When creating a cluster using the Administration Portal's form, selecting the final **OK** button creates the cluster using the entered settings. The Portal's workflow expects you to add more infrastructure resources immediately, and opens a window to guide you through the process. Unless you are ready to add additional hosts now, select the **Configure Later** button to close the window without taking further action.

MAC Address Pools

A MAC address pool defines a range of MAC addresses allocated for each cluster. Each cluster is configured with only one MAC address pool, but that pool may contain multiple address ranges. The same MAC address pool can be shared by multiple clusters. The default installation MAC address pool created by the RHV installation is used for each cluster, unless another MAC address pool is created and assigned. However, when clusters share physical networks, it is not recommended to also share an address pool, because each cluster will be unaware about addresses assigned by the other clusters, which can result in conflicting MAC address assignments. Instead, create a unique MAC address range for each cluster sharing a physical network.

Red Hat Virtualization automatically generates and assigns MAC addresses to new virtual network devices, which helps to prevent MAC address duplication. The MAC address pool assigns the next available MAC address following the last address that was returned to the pool. If there are no further addresses left in the range, the search starts again from the beginning of the range. If there are multiple MAC address ranges, with available MAC addresses, defined in a single MAC address pool, the ranges take turns serving incoming requests in the same way available MAC addresses are selected.

A pool can be configured to allow manually assigning duplicates. A MAC address pool will not *automatically* use a duplicate MAC address, but enabling the duplicates option means a user can *manually* use a duplicate MAC address when required.

Using Pools for the Infrastructure Migration Solution

Understanding MAC address pool behavior is important when using the Infrastructure Migration Solution (IMS), which automates the migration of VMware virtual machines to your Red Hat Virtualization environment. IMS is covered later in this course. Migrated virtual machines can retain their original MAC addresses, because the migration process preserves the source virtual machine's MAC addresses.

You can create a MAC address pool that includes the existing MAC addresses of the source VMware virtual machines to be migrated. If the RHV MAC address pool range overlaps the VMware MAC address range, migrating virtual machine MAC addresses must not duplicate addresses on existing virtual machines, or else the migration will fail.

To ensure that migrated virtual machines obtain MAC addresses in the same range as normal virtual machines created in Red Hat Virtualization, create a MAC address pool to provide new MAC addresses to the VMware virtual machines during migration.

References

Further information is available in the Clusters chapter of the *Administration Guide* for Red Hat Virtualization; at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#chap-Clusters

Guided Exercise

Creating and Managing Clusters

In this exercise, you will create a new cluster in an existing data center in your Red Hat Virtualization environment.

Outcomes

You should be able to create a new cluster.

Before You Begin

Log in to **workstation** as the **student** user with **student** as the password.

On **workstation**, run the **lab datacenters-createcluster start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab datacenters-createcluster start
```

1. On **workstation**, open the Firefox web browser. Log in to the **Red Hat Virtualization Manager** Administration Portal using the `https://rhvm.lab.example.com` URL. The username is **admin** and the password is **redhat**.

2. In the menu, click **Administration**, and then click on **Configure**.

   - In the **Configure** window, select **MAC Address Pools** from the left menu list.
   - Click the **Add** button to bring up the **New MAC Address Pool** window.
   - Enter **mac-pool-clusterone** in the name field.
   - In the **MAC Address Ranges** fields add **56:6f:07:a7:0a:01** in the **From** field.
   - Add **56:6f:07:a7:0a:0a** to the **To** field.
   - Click the **OK** button to close the **New MAC Address Pool** window.

   Click the **Close** button to close out the **Configure** window.

3. In the menu, click **Compute**, and then click on **Hosts**. In the **Hosts** tab select **hosta.lab.example.com** from the host list.

4. In the **Hardware** section of the **General** tab determine the value of the **CPU Type** field. In this classroom, the same CPU type is used by **hostb**, **hostc**, and **hostd**.

5. In the menu, click **Compute**, and then click on **Clusters**. At the top right hand side of the window, find and click the **New** button.

6. In the **New Cluster** window, make sure the **General** section is displayed, and then enter the following configuration settings:

   - Choose **development** in the Data Center drop down menu.
   - Enter **clusterone** in the **Name** field.
   - Select **ovirtmgmt** in the **Management Network** menu.

- Select **x86_64** in the **CPU Architecture** menu.
- Select the CPU type for your hardware in the **CPU Type** menu.
- Select **4.3** in the **Compatibility Version** menu.
- Select **Linux Bridge** in the **Switch Type** menu.
- Check the **Enable Virt Service** check box to allow hosts in this cluster to run virtual machines.
- Keep the default values for the other fields.
- Click the **MAC Address Pool** tab, then change **Default** value to **mac-pool-clusterone** value in the **MAC Address Pool** dropdown.

Click the **OK** button to create the **`clusterone`** cluster.

7. Click **Configure Later** in the pop-up window titled **Cluster - Guide Me**.

8. From the **Clusters** page under **Compute**, verify that the **`clusterone`** cluster appears in the list of available clusters.

Finish

On **`workstation`**, run the **`lab datacenters-createcluster finish`** script to complete this exercise.

```
[student@workstation ~]$ lab datacenters-createcluster finish
```

This concludes the guided exercise.

Lab

Creating and Managing Data Centers and Clusters

Performance Checklist

In this lab, you will create an additional data center and a new cluster in your Red Hat Virtualization environment.

Outcomes

You should be able to:

- Create a new data center.
- Create a new cluster.

Before You Begin

Log in to the **workstation** VM as the **student** user, with **student** as the password.

On **workstation**, run the **lab datacenters-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab datacenters-review start
```

1. On **workstation**, open the Firefox web browser. Log in to the **Red Hat Virtualization Manager** Administration Portal using the `https://rhvm.lab.example.com` URL. The username is **admin** and the password is **redhat**.
2. Create a new data center named **production** using the default values.
3. Create a pool of MAC addresses named **mac-pool-clustertwo** with **56:6f:07:a7:14:01** through **56:6f:07:a7:14:0a** as the MAC address range.

   Create a new cluster named **clustertwo** in the **production** data center. Use **ovirtmgmt** as the management network, use an appropriate CPU Type that could support **hosta**, the **Linux Bridge** switch type, ensure that the cluster can run virtual machines, and use **mac-pool-clustertwo** for the MAC addresses pool. Do not add any hosts to the cluster yet.
4. Verify that the **clustertwo** cluster exists and belongs to the **production** data center.

Evaluation

On the **workstation** VM, run the **lab datacenters-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab datacenters-review grade
```

Finish

On the **workstation** VM, run the **lab datacenters-review finish** command to complete this exercise.

```
[student@workstation ~]$ lab datacenters-review finish
```

This concludes the lab.

Solution

Creating and Managing Data Centers and Clusters

Performance Checklist

In this lab, you will create an additional data center and a new cluster in your Red Hat Virtualization environment.

Outcomes

You should be able to:

- Create a new data center.
- Create a new cluster.

Before You Begin

Log in to the **`workstation`** VM as the **`student`** user, with **`student`** as the password.

On **`workstation`**, run the **`lab datacenters-review start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab datacenters-review start
```

1. On **`workstation`**, open the Firefox web browser. Log in to the **`Red Hat Virtualization Manager`** Administration Portal using the `https://rhvm.lab.example.com` URL. The username is **`admin`** and the password is **`redhat`**.
2. Create a new data center named **`production`** using the default values.
   - 2.1. In the menu, click **Data Centers** in the **Compute** section.
   - 2.2. Click the **New** button on the **Data Centers** tab.
   - 2.3. In the **New Data Center** window, enter **`production`** in the **Name** field. Keep the default values for the other fields. Click the **OK** button to create the data center.
   - 2.4. A pop-up window titled **Data Center - Guide Me** opens. Click the **Configure Later** button to create the empty data center.
3. Create a pool of MAC addresses named **`mac-pool-clustertwo`** with **`56:6f:07:a7:14:01`** through **`56:6f:07:a7:14:0a`** as the MAC address range.

   Create a new cluster named **`clustertwo`** in the **`production`** data center. Use **`ovirtmgmt`** as the management network, use an appropriate CPU Type that could support **`hosta`**, the **`Linux Bridge`** switch type, ensure that the cluster can run virtual machines, and use **`mac-pool-clustertwo`** for the MAC addresses pool. Do not add any hosts to the cluster yet.
   - 3.1. In the menu, click **Administration**, and then click on **Configure**.

Create a pool of MAC addresses with **`mac-pool-clustertwo`** as the name and **`56:6f:07:a7:14:01`** through **`56:6f:07:a7:14:0a`** as the MAC address range.

- In the **Configure** window, select **`MAC Address Pools`** from the left menu list.
- Click the **Add** button to bring up the **New MAC Address Pool** window.
- Enter **`mac-pool-clustertwo`** in the name field.
- In the **MAC Address Ranges** fields add **`56:6f:07:a7:14:01`** in the **From** field.
- Add **`56:6f:07:a7:14:0a`** to the **To** field.

Click the **OK** button to close the **New MAC Address Pool** window.

3.2. In the menu, click **Compute**, and then click on **Hosts**. In the **Hosts** tab select **`hosta.lab.example.com`** on the host list.

3.3. In the Hardware section of the **General** tab, determine the value for the **CPU Type** field. In this classroom, the same CPU type is used by **`hostb`**, **`hostc`**, and **`hostd`**.

3.4. In the menu, click **Compute**, then click on **Clusters**. At the top right hand side of the window, find and click the **New** button.

3.5. In the **New Cluster** window, make sure the **General** section is being displayed, and enter the following configuration settings:

- Select **`production`** in the **Data Center** menu.
- Enter **`clustertwo`** in the **Name** field.
- Select **`ovirtmgmt`** in the **Management Network** menu.
- Select **`x86_64`** in the **CPU Architecture** menu.
- Select the CPU type for your hardware in the **CPU Type** menu.
- Select **`4.3`** in the **Compatibility Version** menu.
- Select **`Linux Bridge`** in the **Switch Type** menu.
- Check the **Enable Virt Service** checkbox to allow hosts in this cluster to run virtual machines.
- Keep the default values for the other fields.
- Click the **MAC Address Pool** tab, then change **Default** value to **mac-pool-clustertwo** value in the **MAC Address Pool** dropdown.

Click **OK** to create the **`clustertwo`** cluster.

3.6. Click **Configure Later** in the pop-up window titled **Cluster - Guide Me**.

4. Verify that the **`clustertwo`** cluster exists and belongs to the **`production`** data center.

4.1. While still being in the **Clusters** page under **Compute**, verify that the **`clustertwo`** cluster appears in the list of available clusters.

4.2. You should see **`production`** as the data center for **`clustertwo`** in the **Data Center** column of the same page. The fact that **`production`** appears in the **Data Center** column for **`clustertwo`** confirms that the **`production`** data center exists.

Evaluation

On the **`workstation`** VM, run the **`lab datacenters-review grade`** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab datacenters-review grade
```

Finish

On the **`workstation`** VM, run the **`lab datacenters-review finish`** command to complete this exercise.

```
[student@workstation ~]$ lab datacenters-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- A *data center* consists of a collection of logical resources, including clusters, hosts, logical networks, and storage domains.
- A *cluster* is a group of hosts in a single data center that act as a migration domain for virtual machines. The hosts do not need to have the same CPU physically, but their hardware does have to match the features that the cluster's CPU type provides.
- The CPU type of the cluster specifies which CPU features are supported by the hardware of every host in the cluster.
- The installation process automatically creates a data center named **`Default`** containing an empty cluster named **`Default`**.
- Additional data centers and clusters can be created in the RHVM Administration Portal.

Chapter 4

Managing User Accounts and Roles

Goal	Configure user accounts using a central directory service. Assign access to resources based on job responsibilities using roles.
Objectives	• Configure Red Hat Virtualization to authenticate users based on information in a central directory service. • Control resource access and management in Red Hat Virtualization using Roles. • Explain the purposes of the User Portal and the Administration Portal, and provide a high-level overview of these user interfaces.
Sections	• Integrating Users from an External Directory Service (and Guided Exercise) • Controlling User Access with Roles (and Guided Exercise) • Navigating the VM Portal and the Administration Portal (and Guided Exercise)
Lab	Managing User Accounts and Roles

Integrating Users from an External LDAP Server

Objectives

After completing this section, you should be able to configure Red Hat Virtualization to authenticate users based on information from an external LDAP server.

Describing Users in Red Hat Virtualization

To interact with the Red Hat Virtualization management system, user accounts must be configured and granted access rights. These accounts come from various sources called *user domains*. Users are identified by their *User Principal Name* (UPN), which has the form *username@domain*.

By default, the initial installation of Red Hat Virtualization creates a *local domain* called **`internal`**, which can contain local user accounts. An initial *local user* is created in this domain, with the UPN **`admin@internal`**, that has full administrative control of the Red Hat Virtualization environment.

While the **`ovirt-aaa-jdbc-tool`** command can create additional local users, it is recommended to configure an *external domain* that gets information about users from an external directory service such as Red Hat Enterprise Linux (RHEL) Identity Manager (IdM), Microsoft Active Directory, OpenLDAP, or one of the many other supported options. These users are referred to as *directory users*. This approach allows simplified user and group management from an operational standpoint by using the same single source of truth for Red Hat Virtualization user information that the IT organization uses for other account management.

In a real-world situation, a centralized directory server handles the directory users. A dedicated security team is responsible for managing the user accounts in the directory server. When there is a requirement of new users, the authorized security team should be contacted with a request to create the new user accounts in the centralized directory server. Because authorization to perform administrative operations on the directory server is restricted, administrative access to the database of user accounts is unavailable.

Administratively, users and groups are created in the directory service. After the directory service is attached to Red Hat Virtualization as an external domain, the users from that service must be configured in Red Hat Virtualization with roles that grant them appropriate levels of access to the Red Hat Virtualization environment.

Directory users can be granted administrative rights. The **`admin@internal`** account is generally better used as an emergency administration account if there is a problem with the connection to the directory service.

Attachment of more than one directory server to the Red Hat Virtualization environment is also possible and supported. If there is more than one directory server attached, then administrators can choose which one they want to authenticate against by selecting the correct domain from the login menu.

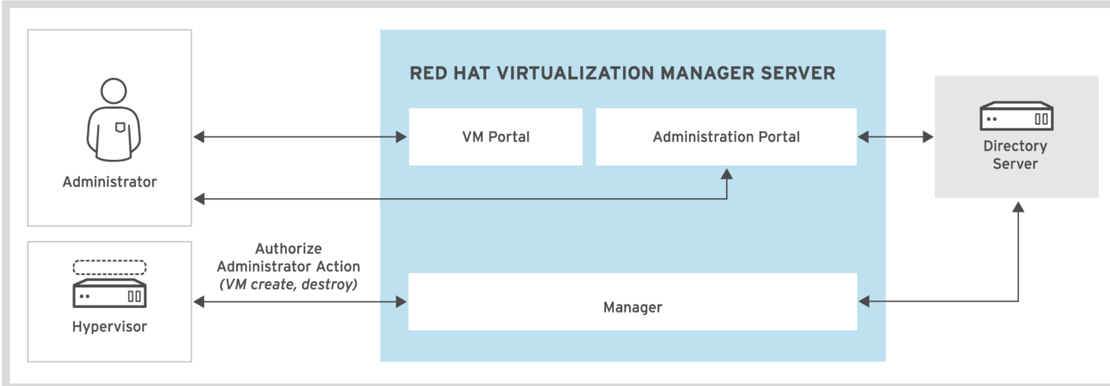

Figure 4.1: Red Hat Virtualization Authentication Provider

Configuring an External LDAP Provider

The *ovirt-engine-extension-aaa-ldap* software package provides support for the integration of generic **`Lightweight Directory Access Protocol (LDAP)`** directory services with Red Hat Virtualization Manager. This includes Red Hat Enterprise Linux Identity Manager, Microsoft Active Directory, OpenLDAP, and a number of other LDAP servers.

Configuring Red Hat Virtualization Manager to use an LDAP-based directory service as an external domain is straightforward. A helper package named *ovirt-engine-extension-aaa-ldap-setup* provides a configuration script, and must be installed on your Red Hat Virtualization Manager server. This helper package also installs *ovirt-engine-extension-aaa-ldap* as a dependency. Finally, the **`ovirt-engine-extension-aaa-ldap-setup`** script is used to configure LDAP integration with Red Hat Virtualization Manager.

The following discussion explores how this configuration is accomplished for two use cases. The first example discusses how to configure Red Hat Enterprise Linux Identity Manager as a directory source. The second example does the same with Microsoft Active Directory.

Attaching Red Hat Enterprise Linux Identity Manager

Red Hat Enterprise Linux Identity Manager is an open source centralized identity, policy, and authorization service included with Red Hat Enterprise Linux that provides an LDAP integration interface. This service is based on the upstream FreeIPA project. Use a Red Hat Enterprise Linux Identity Manager as an authentication source for your Red Hat Virtualization environment.

Gather the following prerequisite information before starting the configuration process:

- The fully qualified DNS domain name of the LDAP server or servers.
- The public TLS/SSL CA certificate that validates the LDAP server's TLS certificate, in PEM format, for a secure connection.
- An existing LDAP account configured so that Red Hat Virtualization Manager can perform search and login queries on the LDAP server. Obtain the base distinguished name (DN) and password from your directory administrator.

After all the prerequisites are met, ensure that the *ovirt-engine-extension-aaa-ldap-setup* package is installed on the Red Hat Virtualization Manager server. The *ovirt-engine-extension-aaa-ldap-setup* package provides the **`ovirt-engine-extension-aaa-ldap-setup`** script, which allows you to interactively configure the Red Hat Virtualization Manager server to use the Red Hat Enterprise Linux Identity Manager as an external domain that serves information about the users.

Running the **ovirt-engine-extension-aaa-ldap-setup** command results in an interactive prompt that is used to provide answers to the questions presented. The first required input is the LDAP implementation to use. Every LDAP implementation has a number to identify it. For example, the **Active Directory** option is identified by the number **3**, and the **IPA** option is identified by the number **6**. To select Red Hat Enterprise Linux Identity Manager, type **6** (**IPA**).

```
[root@rhvm-demo ~]# ovirt-engine-extension-aaa-ldap-setup
[ INFO  ] Stage: Initializing
[ INFO  ] Stage: Environment setup
          Configuration files: ['/etc/ovirt-engine-extension-aaa-ldap-
setup.conf.d/10-packaging.conf']
          Log file: /tmp/ovirt-engine-extension-aaa-ldap-setup-20190715235438-
zjoext.log
          Version: otopi-1.8.2 (otopi-1.8.2-1.el7ev)
[ INFO  ] Stage: Environment packages setup
[ INFO  ] Stage: Programs detection
[ INFO  ] Stage: Environment customization
          Welcome to LDAP extension configuration program
          Available LDAP implementations:
           1 - 389ds
           2 - 389ds RFC-2307 Schema
           3 - Active Directory
           4 - IBM Security Directory Server
           5 - IBM Security Directory Server RFC-2307 Schema
           6 - IPA
           7 - Novell eDirectory RFC-2307 Schema
           8 - OpenLDAP RFC-2307 Schema
           9 - OpenLDAP Standard Schema
          10 - Oracle Unified Directory RFC-2307 Schema
          11 - RFC-2307 Schema (Generic)
          12 - RHDS
          13 - RHDS RFC-2307 Schema
          14 - iPlanet
          Please select: 6
```

In the same interactive prompt, displayed by the **ovirt-engine-extension-aaa-ldap-setup** command, specify whether to use DNS to resolve the name of your Red Hat Enterprise Linux Identity Manager server. Normally, the correct answer is **Yes**.

```
NOTE:
It is highly recommended to use DNS resolution for LDAP server.
If for some reason you intend to use hosts or plain address disable DNS usage.

Use DNS (Yes, No) [Yes]: Enter
```

Note

The default answer is enclosed within square brackets in the prompt. If you press **Enter**, this default answer is selected.

Specify how the Red Hat Virtualization Manager server should find the Red Hat Enterprise Linux Identity Manager server. Select from the following four options:

- The **`Single server`** option, which uses the fully qualified domain name of the server.
- The **`DNS domain LDAP SRV record`** option, which uses a DNS SRV record to locate the server.
- The **`Round-robin between multiple hosts`** option requires a space-separated list of Red Hat Enterprise Linux Identity Manager servers; the Red Hat Virtualization Manager load balances LDAP requests among these servers.
- The **`Failover between multiple hosts`** option requires a space-separated list of Red Hat Enterprise Linux Identity Manager servers; the Red Hat Virtualization Manager sends all requests to the first server in the list, and failover to subsequent servers occurs only if preceding servers do not respond.

To select the **`Single server`** option, type **`1`**. If you select the **`Single server`** option, specify the fully qualified domain name of the Red Hat Enterprise Linux Identity Manager server.

```
Available policy method:
 1 - Single server
 2 - DNS domain LDAP SRV record
 3 - Round-robin between multiple hosts
 4 - Failover between multiple hosts
Please select: 1
Please enter host address: identity-server.example.net
```

At the interactive prompt displayed by the **`ovirt-engine-extension-aaa-ldap-setup`** command, specify the protocol to use when communicating with the directory server. To protect the LDAP connection with the directory server, it is recommended to use the **`StartTLS`** protocol.

```
...output omitted...
Please select protocol to use (startTLS, ldaps, plain) [startTLS]: Enter
```

The **`StartTLS`** protocol requires the PEM-encoded CA certificate that validates the Red Hat Enterprise Linux Identity Manager server's TLS server certificate. You can provide this in a number of different ways: as a URL, a local file, inline through the tool, or from the system-wide CA configuration. Red Hat does not recommend selecting **`Insecure`** because that selection disables validation of the server's TLS certificate, which may lead to the integration of a compromised Red Hat Enterprise Linux Identity Manager server. To download the certificate from a remote location, type **`URL`**, and then specify the URL of the certificate at the subsequent prompt.

```
...output omitted...
Please select method to obtain PEM encoded CA certificate (File, URL, Inline,
 System, Insecure):: URL
URL: http://identity-server.example.net/ipa/config/ca.crt
...output omitted...
```

While using the Red Hat Enterprise Linux Identity Manager, you must specify the distinguished name (DN) of the LDAP user to the **`ovirt-engine-extension-aaa-ldap-setup`** script. The Red Hat Virtualization Manager uses this LDAP user to authenticate itself to the directory server, and to search the directory for information. After specifying the DN of the LDAP user, enter the password of the LDAP user (or if anonymous search is allowed, leave the password blank).

```
Enter search user DN (for example, uid=username,dc=example,dc=com or leave empty
 for anonymous): uid=rhvadmin,cn=users,cn=accounts,dc=example,dc=net
Enter search user password: redhat
...output omitted...
```

While using the Red Hat Enterprise Linux Identity Manager, you must also specify the base DN to the **ovirt-engine-extension-aaa-ldap-setup** script that Red Hat Virtualization Manager uses when searching the LDAP directory.

```
Please enter base DN (dc=example,dc=net) [dc=example,dc=net]: Enter
```

The **ovirt-engine-extension-aaa-ldap-setup** script also prompts you to configure single sign-on on virtual machines using the users provided by the directory servers configured for this external domain. If prompted, enter **Yes** and look at the *Additional Configuration* instructions from the *Red Hat Virtualization Virtual Machine Management Guide* at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/.

Specify a meaningful profile name to **ovirt-engine-extension-aaa-ldap-setup** for the new external domain. A meaningful profile name helps you to easily identify and categorize the external domain when you have multiple external domains configured. The profile name displays in the menu on the login page for the **Administration Portal**. This profile name is the **domain** part of *username@domain*.

```
Please specify profile name that will be visible to users [identity-
server.example.net]: example.net
```

The **ovirt-engine-extension-aaa-ldap-setup** script prompts you for the username and password of a valid user, the Red Hat Enterprise Linux Identity Manager provides. This username and password are used to attempt authentication as that user in order to test whether the connection to the server is working. When you are finished testing, type **Done** and the script exits.

```
[ INFO  ] Stage: Setup validation

NOTE:
It is highly recommended to test drive the configuration before applying it into
 engine.
Login sequence is executed automatically, but it is recommended to also execute
 Search sequence manually after successful Login sequence.

Please provide credentials to test login flow:
Enter user name: idmuser1
Enter user password: redhat
[ INFO  ] Executing login sequence...
...output omitted...
[ INFO  ] Login sequence executed successfully
Please make sure that user details are correct and group membership meets
 expectations (search for PrincipalRecord and GroupRecord titles).
Abort if output is incorrect.
Select test sequence to execute (Done, Abort, Login, Search) [Done]: Done
[ INFO  ] Stage: Transaction setup
```

```
[ INFO  ] Stage: Misc configuration (early)
[ INFO  ] Stage: Package installation
[ INFO  ] Stage: Misc configuration
[ INFO  ] Stage: Transaction commit
[ INFO  ] Stage: Closing up
          CONFIGURATION SUMMARY
          Profile name is: lab.example.com
          The following files were created:
              /etc/ovirt-engine/aaa/lab.example.com.jks
              /etc/ovirt-engine/aaa/lab.example.com.properties
              /etc/ovirt-engine/extensions.d/lab.example.com-authz.properties
              /etc/ovirt-engine/extensions.d/lab.example.com-authn.properties
[ INFO  ] Stage: Clean up
          Log file is available at /tmp/ovirt-engine-extension-aaa-ldap-
setup-20190702121518-kusgin.log:
[ INFO  ] Stage: Pre-termination
[ INFO  ] Stage: Termination
```

After making the configuration changes, restart the **ovirt-engine** service on the Red Hat Virtualization Manager server.

```
[root@rhvm-demo ~]# systemctl restart ovirt-engine
```

Attaching Microsoft Active Directory

RHV also supports using Microsoft Active Directory as an authentication source. Gather the following prerequisite information before starting the configuration process:

- The Active Directory root domain name (the forest name).
- The DNS servers that can resolve the Active Directory forest name.
- The public TLS/SSL CA certificate that validates the LDAP server's TLS certificate, in PEM format, for a secure connection.
- An existing Active Directory account configured so that Red Hat Virtualization Manager can perform search and login queries on the LDAP server. Obtain the account information from your Active Directory administrator. Alternatively, the directory can enable anonymous search.

When all the prerequisites are met, start the integration process. On your Red Hat Virtualization Manager server, use the same **ovirt-engine-extension-aaa-ldap-setup** script to integrate the Microsoft Active Directory service in the Red Hat Virtualization environment. For Microsoft Active Directory, type **3** when the **ovirt-engine-extension-aaa-ldap-setup** script prompts you to select an LDAP implementation from those available.

You must specify your Microsoft Active Directory forest name to the **ovirt-engine-extension-aaa-ldap-setup** script.

You must also select the secure protocol for accessing your Microsoft Active Directory server, and specify the method used to obtain a CA certificate. Again, Red Hat recommends that you use the **StartTLS** protocol and provide a PEM-encoded CA certificate that can validate the Active Directory server certificate. Red Hat recommends that you do not select the **Insecure** option.

While using the Microsoft Active Directory, you must specify the distinguished name (DN) of the Microsoft Active Directory user to the **ovirt-engine-extension-aaa-ldap-setup** script. This user should have permission to browse all users and groups on the Microsoft

Active Directory server. The Red Hat Virtualization Manager uses this Microsoft Active Directory user to authenticate itself to the directory server and search the directory for information. Enter the user's password (or if anonymous search is allowed, leave the password blank).

Specify a meaningful profile name to **`ovirt-engine-extension-aaa-ldap-setup`** for the new external domain. This profile name displays in the menu on the login page for the **Administration Portal**.

The **`ovirt-engine-extension-aaa-ldap-setup`** script prompts you for the username and password of a valid user that the Microsoft Active Directory server provides, in order to test whether the connection to the server is working. After testing completes, enter **Done** and the script exits.

After making the configuration changes, restart the **`ovirt-engine`** service on the Red Hat Virtualization Manager server.

```
[root@rhvm-demo ~]# systemctl restart ovirt-engine
```

Important

The preceding examples configure Red Hat Virtualization Manager to authenticate users based on information in an external directory service. However, those users must still be assigned roles to authorize them to use Red Hat Virtualization Manager and work with resources in the Red Hat Virtualization environment.

The next section of this chapter covers how users are assigned roles.

References

Further information is available in the Users and Roles chapter of the *Administration Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/chap-users_and_roles#chap-Users_and_Roles

Guided Exercise

Integrating Users from an External Directory Service

In this exercise, you will integrate users from a Red Hat Enterprise Linux (RHEL) Identity Manager (IdM) server with your Red Hat Virtualization environment.

Outcomes

You should be able to configure the Red Hat Virtualization Manager to use the Red Hat Enterprise Linux Identity Manager server, preconfigured in the classroom as a source for user information.

Before You Begin

You have the Red Hat Enterprise Linux Identity Manager server installed, preconfigured, and running on the **utility** system.

Log in as the **student** user on **workstation** and run the **lab users-directory start** command. This command verifies the operational state of the Red Hat Virtualization environment.

```
[student@workstation ~]$ lab users-directory start
```

1. From **workstation**, open an SSH session to **rhvm** as the **root** user.

```
[student@workstation ~]$ ssh root@rhvm
...output omitted...
[root@rhvm ~]#
```

2. Configure the Red Hat Virtualization Manager running on **rhvm** to use the Red Hat Enterprise Linux Identity Manager server as a source for user information.

   2.1. Use the **rpm** command to verify that the *ovirt-engine-extension-aaa-ldap-setup* package is installed on **rhvm**.

```
[root@rhvm ~]# rpm -q ovirt-engine-extension-aaa-ldap-setup
ovirt-engine-extension-aaa-ldap-setup-1.3.9-1.el7ev.noarch
```

   The *ovirt-engine-extension-aaa-ldap-setup* package is already installed because it is automatically included in a self-hosted engine installation, like the one used in this class.

   2.2. To start the interactive setup, run the **ovirt-engine-extension-aaa-ldap-setup** command.

```
[root@rhvm ~]# ovirt-engine-extension-aaa-ldap-setup
[ INFO  ] Stage: Initializing
[ INFO  ] Stage: Environment setup
          Configuration files: ['/etc/ovirt-engine-extension-aaa-ldap-
setup.conf.d/10-packaging.conf']
          Log file: /tmp/ovirt-engine-extension-aaa-ldap-setup-20190702112955-
wwd3ln.log
          Version: otopi-1.8.2 (otopi-1.8.2-1.el7ev)
[ INFO  ] Stage: Environment packages setup
[ INFO  ] Stage: Programs detection
[ INFO  ] Stage: Environment customization
...output omitted...
```

2.3. Type **6** to select **IPA** from the **Available LDAP implementations** list.

```
...output omitted...
           6 - IPA
...output omitted...
          Please select: 6
```

2.4. Press **Enter** to accept the default setting of using DNS to resolve the host name of the Red Hat Enterprise Linux Identity Manager server.

```
...output omitted...
          NOTE:
          It is highly recommended to use DNS resolution for LDAP server.
          If for some reason you intend to use hosts or plain address disable DNS
 usage.

          Use DNS (Yes, No) [Yes]: Enter
```

2.5. Type **1** to select **Single server** from the **Available policy method** list.

```
...output omitted...
          Available policy method:
           1 - Single server
           2 - DNS domain LDAP SRV record
           3 - Round-robin between multiple hosts
           4 - Failover between multiple hosts
          Please select: 1
```

2.6. Type **utility.lab.example.com** to specify the host address of the Red Hat Enterprise Linux Identity Manager server.

```
Please enter host address: utility.lab.example.com
```

2.7. Press **Enter** to accept the default secure connection method (**StartTLS**) for the Red Hat Enterprise Linux Identity Manager server.

```
...output omitted...
          Please select protocol to use (startTLS, ldaps, plain) [startTLS]: Enter
```

2.8. Select the **URL** method to obtain the CA certificate.

```
Please select method to obtain PEM encoded CA certificate (File, URL, Inline,
 System, Insecure): URL
```

2.9. Specify `http://utility.lab.example.com/ipa/config/ca.crt` as the URL to the CA certificate.

```
URL: http://utility.lab.example.com/ipa/config/ca.crt
[ INFO  ] Connecting to LDAP using 'ldap://utility.lab.example.com:389'
[ INFO  ] Executing startTLS
[ INFO  ] Connection succeeded
```

2.10. The Red Hat Enterprise Linux Identity Manager server in the classroom has been configured with a user that the RHV Manager can use to search the LDAP directory for user information. The user DN is **`uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com`**. The password for this DN is **`redhat`**.

```
Enter search user DN (for example uid=username,dc=example,dc=com or leave empty
 for anonymous): uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com
Enter search user password: redhat
[ INFO  ] Attempting to bind using
 'uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com'
```

2.11. Accept **`dc=lab,dc=example,dc=com`** as the proposed base DN by pressing **`Enter`**.

```
Please enter base DN (dc=lab,dc=example,dc=com) [dc=lab,dc=example,dc=com]: Enter
```

2.12. Type **`No`** to indicate that you will not use single sign-on for virtual machines.

```
Are you going to use Single Sign-On for Virtual Machines (Yes, No) [Yes]: No
```

2.13. Specify **`lab.example.com`** as the name of the profile for the external domain.

```
Please specify profile name that will be visible to users
 [utility.lab.example.com]: lab.example.com
[ INFO  ] Stage: Setup validation
```

2.14. Test the login function to ensure that the Red Hat Enterprise Linux Identity Manager server is connected to the Red Hat Virtualization Manager.

```
NOTE:
It is highly recommended to test drive the configuration before applying it into
 engine.
Login sequence is executed automatically, but it is recommended to also execute
 Search sequence manually after successful Login sequence.

Please provide credentials to test login flow:
Enter user name: rhvadmin
Enter user password: redhat
[ INFO  ] Executing login sequence...
...output omitted...
[ INFO  ] Login sequence executed successfully
```

2.15. Press **Enter** to use **Done** as the default selection. This completes the configuration.

```
Please make sure that user details are correct and group membership meets
 expectations (search for PrincipalRecord and GroupRecord titles).
Abort if output is incorrect.
Select test sequence to execute (Done, Abort, Login, Search) [Done]: Enter
[ INFO  ] Stage: Transaction setup
[ INFO  ] Stage: Misc configuration (early)
[ INFO  ] Stage: Package installation
[ INFO  ] Stage: Misc configuration
[ INFO  ] Stage: Transaction commit
[ INFO  ] Stage: Closing up
          CONFIGURATION SUMMARY
          Profile name is: lab.example.com
          The following files were created:
              /etc/ovirt-engine/aaa/lab.example.com.jks
              /etc/ovirt-engine/aaa/lab.example.com.properties
              /etc/ovirt-engine/extensions.d/lab.example.com-authz.properties
              /etc/ovirt-engine/extensions.d/lab.example.com-authn.properties
[ INFO  ] Stage: Clean up
          Log file is available at /tmp/ovirt-engine-extension-aaa-ldap-
setup-20190702121518-kusgin.log:
[ INFO  ] Stage: Pre-termination
[ INFO  ] Stage: Termination
```

3. Use **systemctl** to restart the **ovirt-engine** service. Wait for the service to finish activating components before accessing the RHV Manager Administration Portal. Log out of **rhvm**.

```
[root@rhvm ~]# systemctl restart ovirt-engine
[root@rhvm ~]# logout
Connection to rhvm closed.
[student@workstation ~]$
```

4. On **workstation**, open Firefox and navigate to `https://rhvm.lab.example.com/ovirt-engine`. From the page that displays, click on the **Administration Portal** link.

   If the RHEL IdM server was successfully integrated with the RHV Manager, you will see **lab.example.com** as one of the listed profiles. To login using the **lab.example.com** profile, you must first map the domain user accounts to appropriate roles. You will map the domain user accounts in the next guided exercise.

Finish

On **workstation**, run the **lab users-directory finish** script to complete this exercise.

```
[student@workstation ~]$ lab users-directory finish
```

This concludes the guided exercise.

Controlling User Access with Roles

Objectives

After completing this section, you should be able to use roles to control resource access and management in Red Hat Virtualization.

Managing User Access

New users are typically created in a directory service that is configured in Red Hat Virtualization as an external domain, using native administration mechanisms for that directory service. This configuration was discussed in the preceding section of this chapter.

New users created in a directory service are not initially authorized to access to the Red Hat Virtualization environment. User accounts must be granted permission to perform actions in the Red Hat Virtualization environment before they can be used. In this section, you learn how to manage user access using preconfigured settings called *roles*.

The Red Hat Virtualization authorization model is based around users, actions, and objects. Actions are tasks that can be performed, such as starting or stopping a virtual machine, creating a new template, or migrating a virtual machine to a different host.

Each type of action corresponds to a *permission*. Users have permissions that allow them to perform actions on objects. Objects are things like data centers, clusters, hosts, networks, or virtual machines.

To simplify maintenance, multiple permissions can be combined into a *role*. A role, in Red Hat Virtualization environment, is a set of privileges permitting access to physical and virtual resources at various levels. The system comes with multiple predefined roles, such as **`SuperUser`** and **`PowerUserRole`**. These roles make it easier to provide a specific level of access to a user.

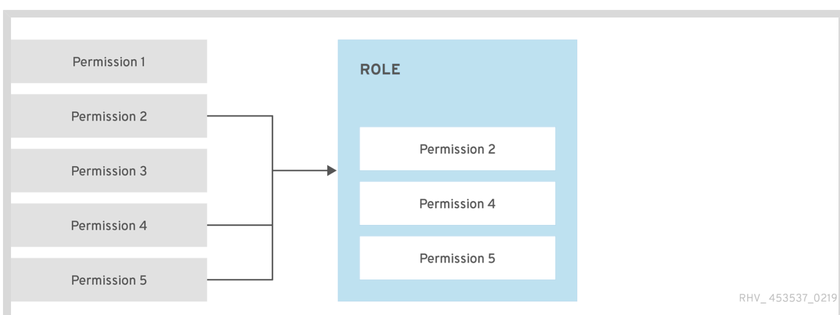

Figure 4.2: Roles and permissions

Users can be assigned roles that apply to the entire Red Hat Virtualization environment, or only to a specific object (such as a virtual machine or a data center). If a user is assigned a role on an object that contains other objects, then the user gets the same role on all objects in the container. For example, if a user is assigned the **`HostAdmin`** role on a cluster, then the user gets the **`HostAdmin`** role on all hosts in that cluster.

When a user performs an action on an object, the user is identified as an entity. This entity is a combination of the user account, and its permissions on the target object. If both the user account information and permissions are correct, the user is allowed to perform the requested action on the target object; otherwise, the request is denied.

> **Important**
> To perform certain actions, a user may need to have permissions (or roles) on multiple objects. For example, copying a template between storage domains requires the user to have relevant permissions on both storage domains.

The following graphic shows how permissions are inherited between objects.

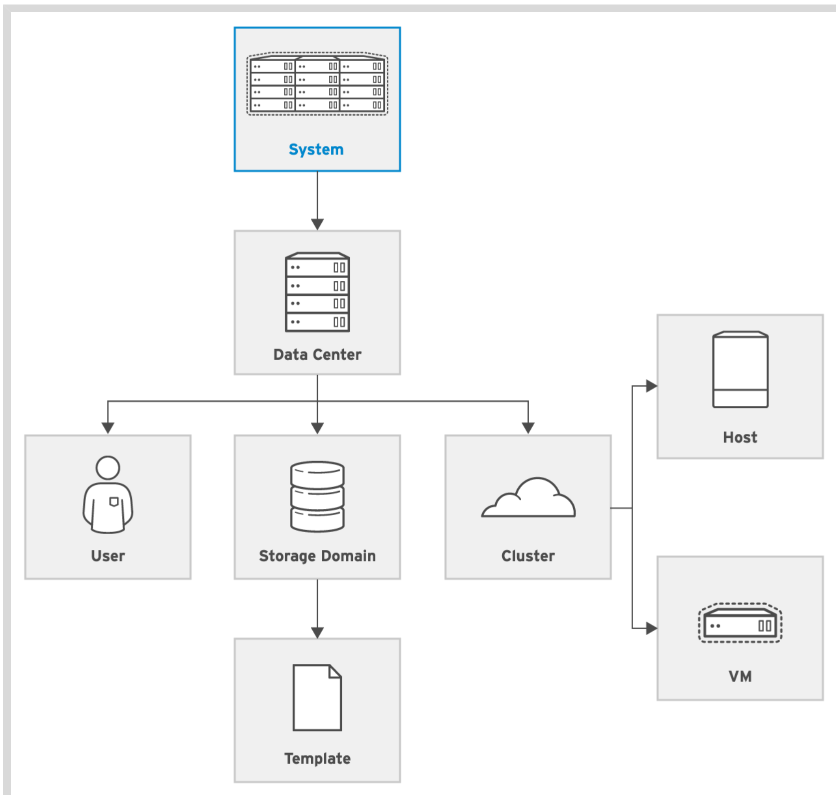

Figure 4.3: The hierarchical layout of objects in Red Hat Virtualization

Role Types

Red Hat Virtualization comes with a variety of preconfigured roles. Two types of roles that exist in the Red Hat Virtualization environment are:

Administrator role
: Users assigned an administrator role can access the **Administration Portal**. Using these roles, users can manage physical and virtual resources.

User role
: Users assigned a user role can access the **VM Portal**. The assigned role determines what a user can view and is allowed to do in the **VM Portal**.

The following roles come predefined in a standard Red Hat Virtualization environment. Users can be assigned multiple roles at multiple levels in the RHV hierarchy. The complete set of actions a user can accomplish is defined by the union of all roles assigned at a particular level.

Administrator Roles (Basic)

SuperUser

This role gives the user full permissions across all objects and levels in your Red Hat Virtualization environment. The **`admin@internal`** user has this role. This role is for architects and engineers that create and manage the RHV environment, and for external resources that support it. For security, this role should be used sparingly after the data centers and clusters have been initially created.

ClusterAdmin

This role gives the user administrative permissions for all resources in a specific cluster. This role is for cluster administrators for specific clusters. Users with this role assigned to one or more clusters can administer those clusters and their resources, but cannot create new clusters.

DataCenterAdmin

This role gives the user administrative permissions across all objects in a specific data center, except for storage. This role is for data center administrators for specific data centers. Users with this role assigned to one or more data centers can administer all objects in a those data centers and their resources except for storage, which is manage by a StorageAdmin.

Administrator Roles (Advanced)

TemplateAdmin

This role represents the permissions required by a virtual machine template administrator. Users with this role assigned can create, delete, and configure the storage domains and network details of templates.

StorageAdmin

This role represents the permissions required by a storage administrator. Users with this role assigned can create, delete, and manage assigned storage domains.

HostAdmin

This role represents the permissions required by a host administrator. Users with this role assigned can attach, remove, configure, and manage a host.

NetworkAdmin

This role represents the permissions required by a network administrator. Users with this role assigned can create, remove, and edit the networks of an assigned data center or cluster.

GlusterAdmin

This role represents the permissions required for a Red Hat Gluster Storage administrator. Users with this role can create, remove, and manage Gluster storage volumes.

VmImporterExporter

This role represents the permissions of an import and export administrator. Users with this role can import and export virtual machines.

User Roles (Basic)

UserRole

This role allows users to log in to the **VM Portal**. This role allows the user to access and use assigned virtual machines, including checking their state, and viewing virtual machine details. This role does not allow the user to administer their assigned virtual machines.

PowerUserRole

This role gives the user permission to manage and create virtual machines and templates at their assigned level. Users with this role assigned at a data center level can create virtual machines and templates in the data center. This role allows users to self-service their own virtual machines.

UserVmManager
: This role allows users to manage virtual machines, and to create and use snapshots for the VMs they are assigned. If a user creates a virtual machine using the **VM Portal**, that user is automatically assigned this role on the new virtual machine.

User Roles (Advanced)

UserTemplateBasedVm
: This role gives the user limited privileges to use only the virtual machine templates. Users with this role assigned can create virtual machines based on templates.

DiskOperator
: This role gives the user privileges to manage virtual disks. Users with this role assigned can use, view, and edit virtual disks.

VmCreator
: This role gives the user permission to create virtual machines using the **User Portal**. Users with this role assigned can create virtual machines using **VM Portal**.

TemplateCreator
: This role gives the user privileges to create, edit, manage, and remove templates. Users with this role assigned can create, remove, and edit templates.

DiskCreator
: This role gives the user permission to create, edit, manage, and remove virtual disks. Users with this role can create, remove, manage, and edit virtual disks within the assigned part of the environment.

TemplateOwner
: This role gives the user privileges to edit and remove templates, as well as assign user permissions for templates. It is automatically assigned to the user who creates a template.

VnicProfileUser
: This role gives the user permission to attach or detach network interfaces. Users with this role can attach or detach network interfaces from logical networks.

Use these roles to better manage user access and to delegate administrative authority. For example, assign **`SystemAdmin`** to specific users without giving them access to the **`admin@internal`** account. Users with roles can be properly tracked and managed for compliance.

Assign less comprehensive roles to appropriate users in order to offload administrative tasks. The **`DataCenterAdmin`**, **`ClusterAdmin`**, and **`PowerUserRole`** roles are useful for this purpose.

> **Important**
>
> If you have **`UserRole`** on a virtual machine, then you can see the virtual machine in **VM Portal** and can start or stop that machine. You cannot create new virtual machines, or edit or delete existing ones. Also, if you only have **`UserRole`**, then you can only see the basic mode of the **VM Portal**.
>
> If you have **`UserVmManager`** on a virtual machine, then you have full control of the virtual machine in **VM Portal**, and you can edit its configuration or even delete it.
>
> If you have only **`PowerUserRole`**, you can create virtual machines in the **VM Portal**, and you can see your own virtual machines because you automatically get **`UserVmManager`** on the virtual machines you create. You are not able to see the virtual machines, other users created, unless you also have at least **`UserRole`** on those machines. If an administrator removes your **`UserVmManager`** role on the virtual machines you created, and you only have **`PowerUserRole`**, but not **`UserRole`** on those virtual machines, then you are no longer able to see your machine in the **VM Portal**.

> **Note**
>
> The default roles cannot be changed or removed.
>
> It is possible to clone the default roles for customization, or to create entirely new roles. How to do so is beyond the scope of this course, but more information is available in the *Red Hat Virtualization Administration Guide* at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/chap-global_configuration#sect-Roles.

Assigning Roles to Users

Confirm that the user exists using the administrative tools supported by your directory service domain. After confirming the user, grant any desired permissions or roles using the **Administration Portal**.

Assigning System-wide Roles to Users

To assign a role to a user applicable to all objects in the Red Hat Virtualization environment, log in to the **Administration Portal** as a user that has the **`SuperUser`** role, for example as **`admin@internal`** user.

In the web interface, assign the system-wide roles to the users using the **Configure** dialog box. To access this dialog box, navigate to **Administration** → **Configure**. You can find the **Administration** menu on the navigation pane displayed on the left.

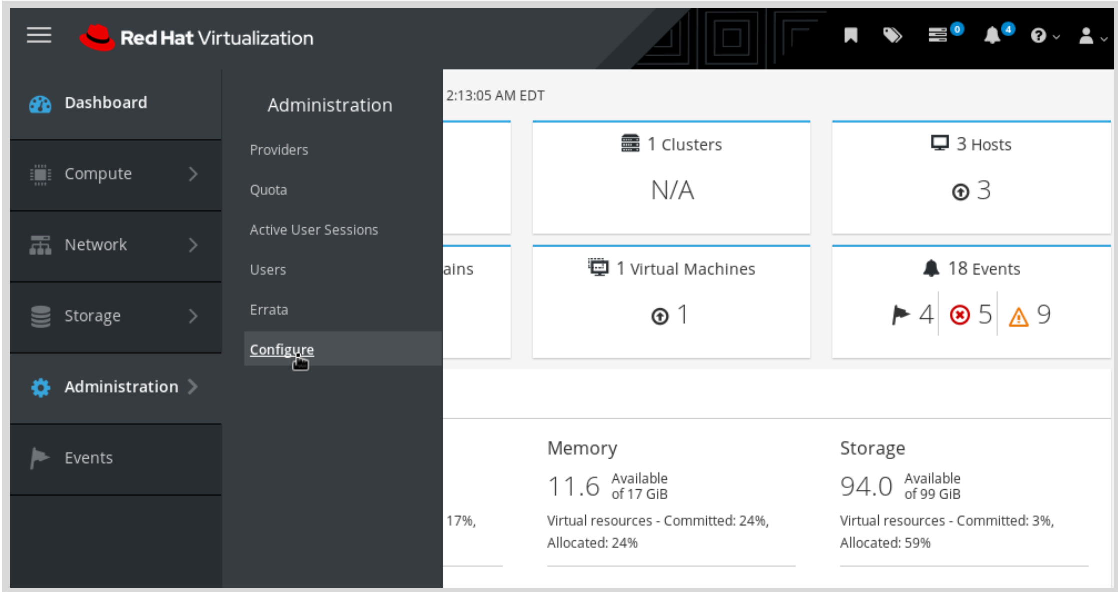

Figure 4.4: Accessing configure dialog box

In the **Configure** dialog box, under **System Permissions**, the **Add** button is used to assign roles to the users.

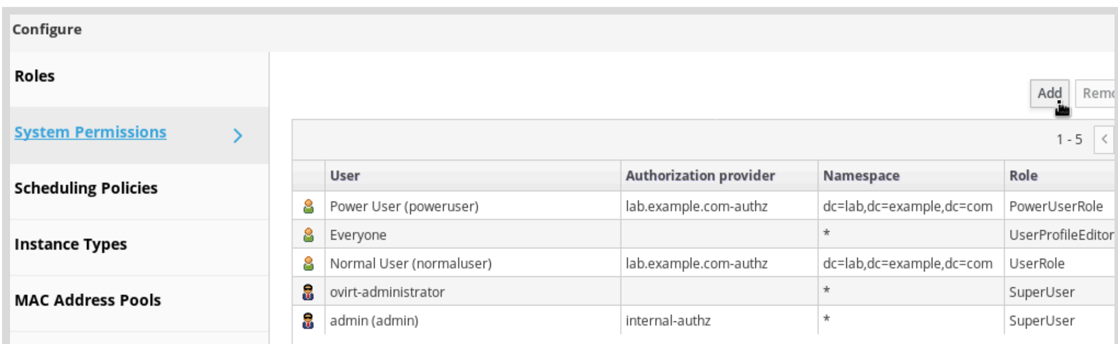

Figure 4.5: Adding users

The **Add** button in the **Configure** dialog box opens the **Add System Permission to User** dialog box. The **Add System Permission to User** dialog box allows you to select the appropriate profile to use, under the **Search** field. In the **Add System Permission to User** dialog box the **GO** button retrieves the list of users and groups from the appropriate source, based on the selected profile.

From the returned list of users, enable the check box for the specific user and select the appropriate role from the drop-down menu under **Role to Assign**. Confirm the role assigned to the selected user and then click the **OK** button.

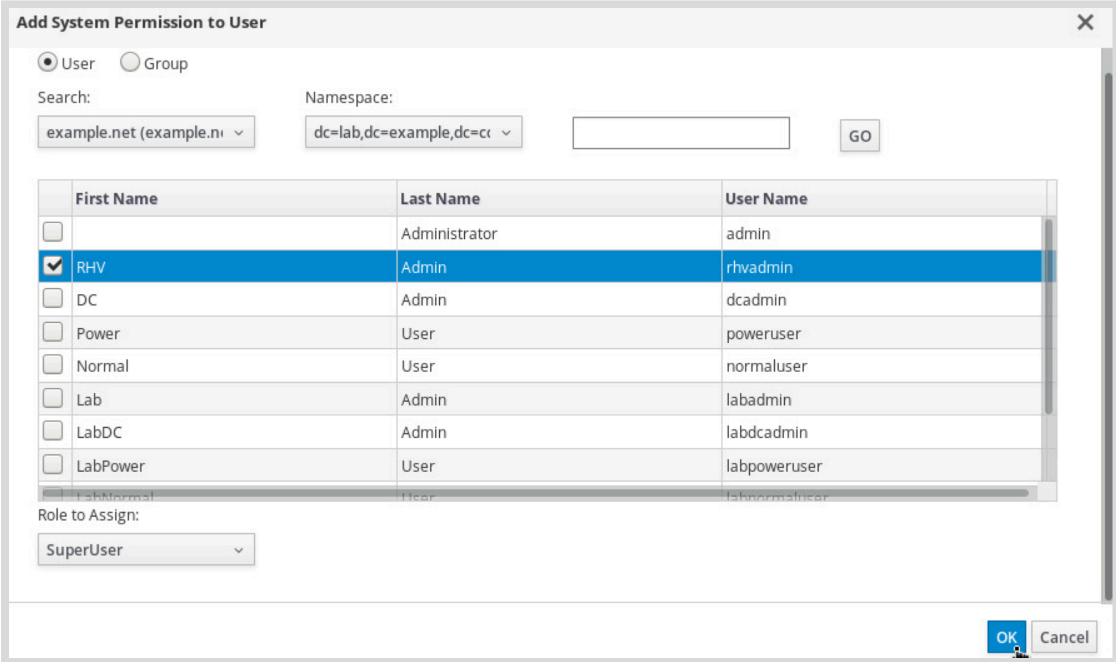

Figure 4.6: Adding permissions

To verify that the user has been granted the correct permissions, log in to the appropriate portal using the user's credentials.

Assigning Resource-specific Roles to Users

Sometimes users should be assigned a role that only applies to a subset of resources in the Red Hat Virtualization environment. Depending on the role assigned, users are able to access and use resources as described earlier in this chapter.

To assign roles to users at the resource level, determine the resource and then add permissions specific to that particular resource. The following screenshot considers **`Data Centers`** as the determined resource.

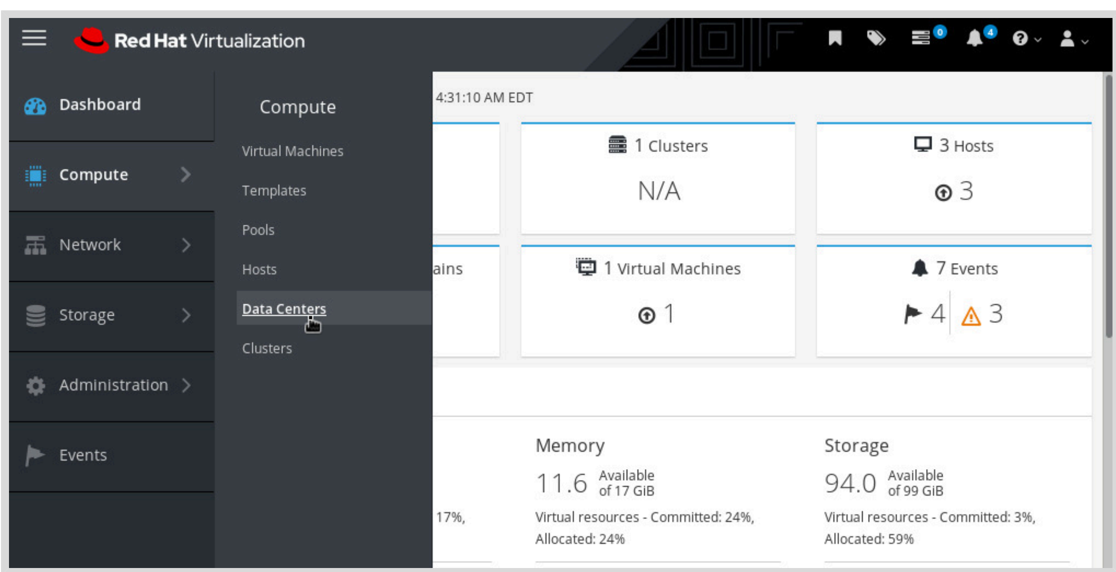

Figure 4.7: Accessing Resources

Each type of resources has its own page. This page lists the available resources of the specific type. For example, all the available data centers are listed on a page that is different from the page that lists all the networks.

Click on the name of the resource to access the objects associated with the resource. Among the associated objects are **`Permissions`**, which enables adding users with specific roles at the resource level. These permissions define the rights of a user with a specific role on the resource and its children resources, if any. The user does not inherit the same permissions on other resources at the same level as the resource to which permissions were applied.

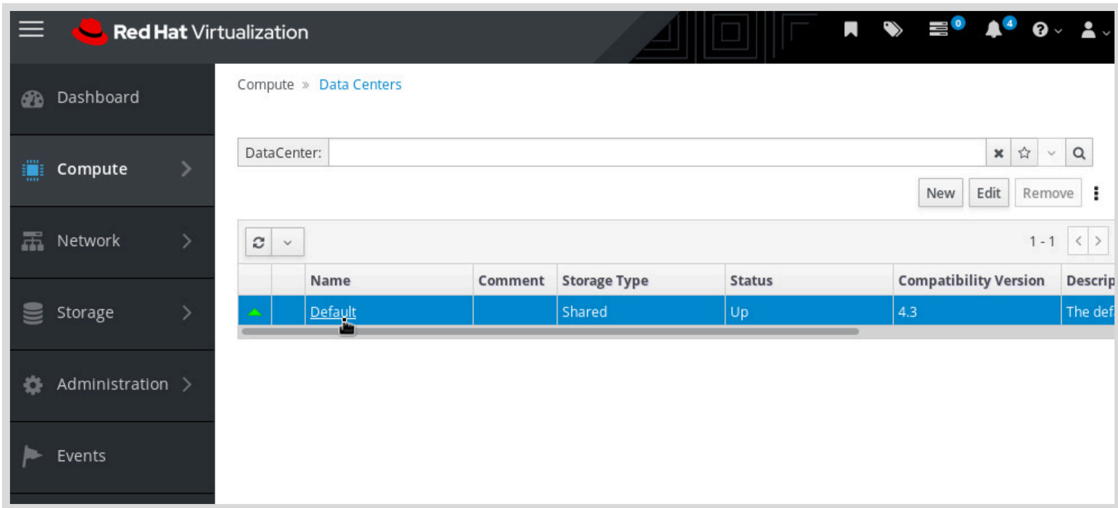

Figure 4.8: Accessing the resource objects

For the selected resource, the **Permissions** tab returns a list of assigned users, user roles, and inherited permissions. The **Add** button on the **Permissions** tab leads you to the **Add Permission to User** dialog box where you can assign roles to the users for the resource.

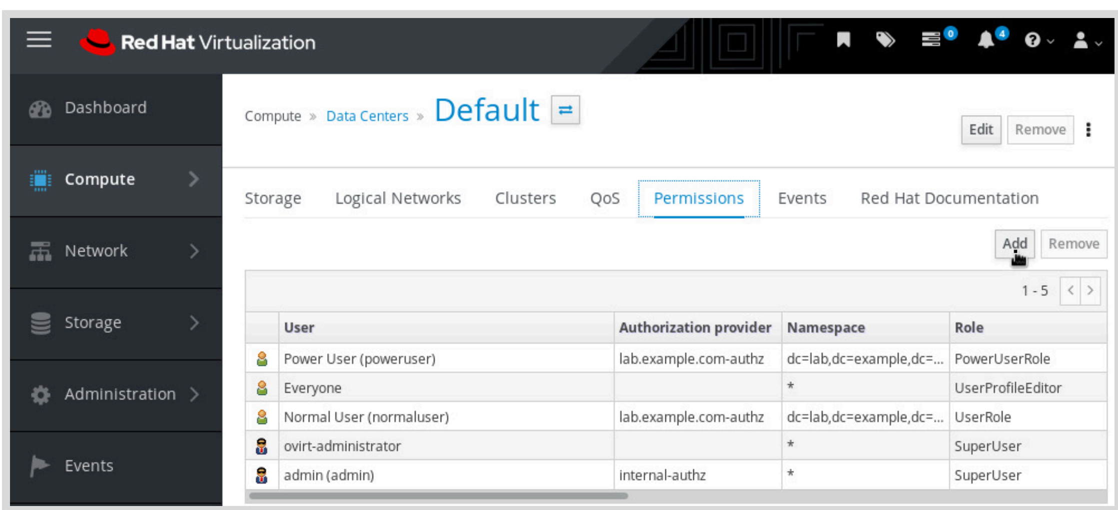

Figure 4.9: Adding Permissions

The **Add Permission to User** dialog box allows you to select the appropriate profile to use from the **Search** field. In the **Add Permission to User** dialog box, the **GO** button retrieves the list of users and groups from the appropriate source, based on the selected profile.

From the list of returned users, enable the check box for the specific user and select an appropriate role from the drop-down menu under **Role to Assign**. The **OK** button in this dialog box is used to confirm the role assignment to the selected user for the resource.

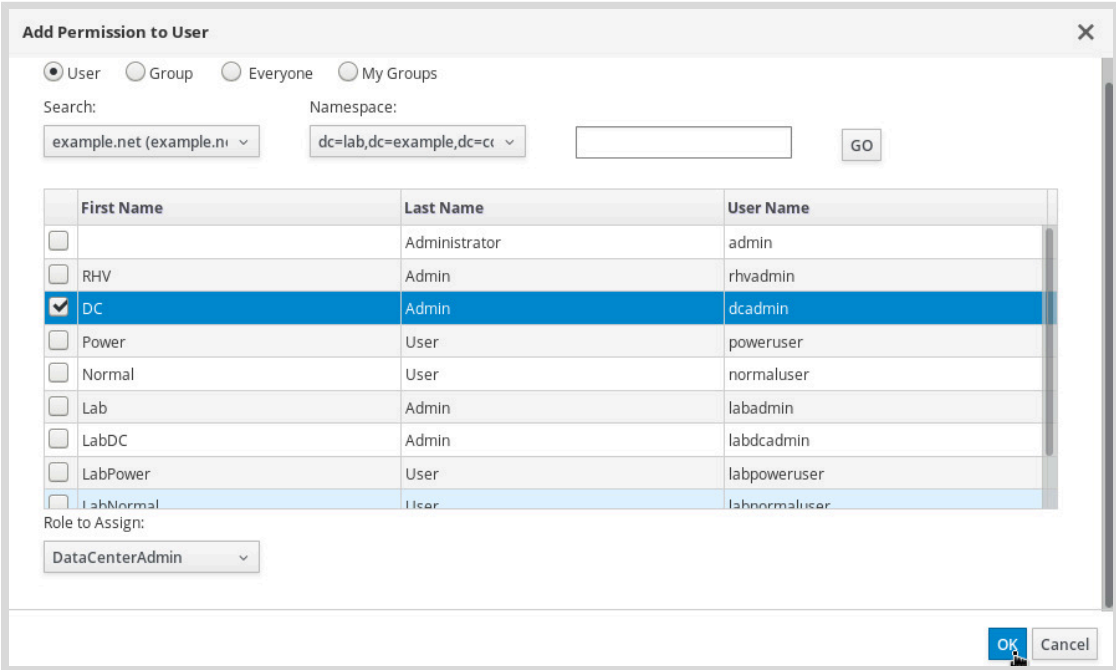

Figure 4.10: Assigning Role to users at resource level

To verify that the user has been added with the correct permissions, log in to the appropriate portal with the user's credentials and access the resource for which you added the permissions.

Removing Roles from Users

To remove a role from a user in the Red Hat Virtualization environment, clear the box for the user in the **Add System Permission to User** or **Add Permission to User** dialog box, and then click **OK** to confirm. The **`SuperUser`** role is required to perform this action.

Resetting the Internal Administration User's Password

The **`admin@internal`** account is created at installation time as a default user that has the system-wide **`SuperUser`** role. Like **`root`** on a Red Hat Enterprise Linux system, this can be useful as an emergency administration account if your external directory service is down.

From time to time, you may need to change or reset the password for this account. Use the **`ovirt-aaa-jdbc-tool`** command to perform the reset. There is no need to restart the **`ovirt-engine`** service in your RHV environment for this change to take effect.

The following command changes the password of the **`admin@internal`** user to **`redhat123`**, and sets the password validity to **`2025-08-01 12:00:00Z`**.

```
[root@rhvm-demo ~]# ovirt-aaa-jdbc-tool user password-reset admin \
--password-valid-to="2025-08-01 12:00:00Z"
Password: redhat123
Reenter password: redhat123
updating user admin...
user updated successfully
```

User accounts in the **`internal`** local domain use the following complexity criteria for the password, by default.

- Passwords must be six characters long.
- The last three passwords cannot be used again.

You can list or change the default policy using **`ovirt-aaa-jdbc-tool`** with the **`settings`** subcommand. Detailed information on how to do so is beyond the scope of this course.

Important

If you attempt to log in to the **Administration Portal** of the Red Hat Virtualization Manager as **`admin`** too many times with the wrong password, then the account may be locked. The following command unlocks the account. You must execute the command as the **`root`** user on the Red Hat Virtualization Manager server.

```
[root@rhvm-demo ~]# ovirt-aaa-jdbc-tool user unlock admin
updating user admin...
user updated successfully
```

References

Further information is available in the Global Configuration chapter of the *Administration Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/chap-global_configuration

Guided Exercise

Controlling User Access with Roles

In this exercise, you will configure users in your Red Hat Virtualization environment using both newly configured and existing roles, as needed to perform common job responsibilities.

Outcomes

You should be able to configure four different users from the **`lab.example.com`** profile with various roles:

- **`rhvadmin`** with the **`SuperUser`** role system-wide.
- **`normaluser`** with the **`UserRole`** role system-wide.
- **`poweruser`** with the **`PowerUserRole`** role system-wide.
- **`dcadmin`** with the **`DataCenterAdmin`** role for the **`development`** data center.

  Although we configure users and roles in this chapter all subsequent chapters will use the default username, password and profile.

Before You Begin

Log in as the **`student`** user on **`workstation`**, and then run the **`lab users-roles start`** command. This command ensures that the domain users of the Red Hat Enterprise Linux Identity Management server in the classroom are mapped to the appropriate roles in the Red Hat Virtualization environment.

```
[student@workstation ~]$ lab users-roles start
```

1. Assign the **`SuperUser`** role, system-wide, to the **`rhvadmin`** user in the **`lab.example.com`** profile.

   1.1. On **`workstation`**, open Firefox and navigate to `https://rhvm.lab.example.com/ovirt-engine`. Click **Administration Portal** to log in to the web interface as the internal user called **`admin`** with **`redhat`** as the password. Select the **`internal`** profile.

   1.2. In the menu, click **Administration**, and then click **Configure**.

   1.3. In the **Configure** dialog box, click **System Permissions**.

   1.4. Click the **Add** button to add a role to a user.

   1.5. In the **Add System Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item. This item represents the **`lab.example.com`** profile you configured in the preceding exercise to allow Red Hat Virtualization Manager to use the Red Hat Enterprise Linux Identity Manager as a source for the users.

   1.6. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

1.7. In the list of users that displays, click the check box for the **rhvadmin** user.

1.8. Click the drop-down list under **Role to Assign**. From the list of available roles, select **SuperUser** role for **rhvadmin**.

1.9. Click **OK** to assign the specified role to the selected user. Notice that the **rhvadmin** user displays in the **System Permissions** list. This list confirms that the **rhvadmin** user has been assigned a role granting administrative access to Red Hat Virtualization.

1.10. In the **Configure** dialog box, click **Close**.

2. Verify that you can log in to the **Administration Portal** as the **rhvadmin** user in the **lab.example.com** profile.

2.1. Sign out as **admin** from the **Administration Portal**.

2.2. Log back in to the **Administration Portal** as the **rhvadmin** user that you have just added. Use **rhvadmin** as the user name and **redhat** as the password. In the **Profile** field, click the drop-down list and select the **lab.example.com** profile.

Click the **Log In** button to log in as the **rhvadmin** user.

3. As **rhvadmin**, assign the **UserRole** role, system-wide, to the **normaluser** user from the **lab.example.com** profile.

3.1. In the web interface click **Administration** on the navigation pane in the left side and click **Configure** from the options that appear.

3.2. In the **Configure** dialog box, click **System Permissions**.

3.3. Click the **Add** button to add a role to a user.

3.4. In the **Add System Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item.

3.5. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

3.6. In the list of users that displays, click the check box for the **normaluser** user.

3.7. In the list of available roles, leave the default **UserRole** role selected for the **normaluser** user.

3.8. Click **OK** to assign the specified role to the selected user. Notice that the **normaluser** user displays in the **System Permissions** list. This list confirms that the **normaluser** user has been assigned a role granting access to Red Hat Virtualization.

4. Assign the **PowerUserRole** role, system-wide, to the user **poweruser** in the **lab.example.com** profile.

4.1. Click the **Add** button to add a role to another user.

4.2. In the **Add System Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item.

4.3. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

4.4. In the list of users that displays, click the check box for the **poweruser** user.

4.5. Click the drop-down list under **Role to Assign**. Choose **PowerUserRole** role from the list of available roles.

4.6. Click **OK** to assign the specified role to the selected user. Notice that the **poweruser** user displays in the **System Permissions** list. This list confirms that the **poweruser** user has been assigned a role granting access to Red Hat Virtualization.

4.7. In the **Configure** dialog box, click **Close**.

5. Assign the **DataCenterAdmin** role, for only the **development** data center, to the **dcadmin** user in the **lab.example.com** profile.

5.1. In the menu click **Compute**, then choose **Data Centers**.

5.2. Click the **development** data center among the available data centers.

5.3. Navigate to the **Permissions** tab and click **Add** to add a role to a user for the **development** data center.

5.4. In the **Add Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item.

5.5. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

5.6. In the list of users that displays, click the check box for the **dcadmin** user.

5.7. Click the drop-down list under **Role to Assign**. From the list of available roles, select **DataCenterAdmin**.

5.8. Click **OK** to assign the specified role to the selected user. Notice that the **dcadmin** user displays in the **Permissions** list. This list confirms that the **dcadmin** user has been assigned a role granting access to Red Hat Virtualization for the specific data center.

6. Sign out as **rhvadmin** from the **Administration Portal**.

Finish

On **workstation**, run the **lab users-roles finish** script to complete this exercise.

```
[student@workstation ~]$ lab users-roles finish
```

This concludes the guided exercise.

Navigating the VM Portal and the Administration Portal

Objectives

After completing this section, you should be able to describe the purposes of the **VM Portal** and the **Administration Portal**, and provide a high-level overview of each user interface.

Accessing and Using the VM Portal

The **VM Portal** is dedicated primarily to users of particular virtual machines. Usually, those users are assigned with the **`UserRole`** role. This role allows for easy access to the console of a virtual machines, as well as the ability to start, stop, restart, or shut them down. It is designed for users who do not need to access any of the Red Hat Virtualization environment resources. Users with the **`PowerUserRole`** role have access to more advanced options in the **VM Portal** than users with the **`UserRole`** role. Users with the **`PowerUserRole`** role can manage virtual machines, virtual disks, templates, and network interfaces to the virtual machines, and can also start, stop, restart, or shutdown the virtual machines.

To access the **VM Portal**, open your web browser and go to the server with Red Hat Virtualization Manager installed. Click the **VM Portal** link to open the login page. Enter your user name and password. Select the appropriate profile for your domain and click **Log In**.

The screen in the **VM Portal** is divided into three sections: the header bar, the toolbar, and the virtual machines pane. The refresh button, user drop-down menu, and messages drop-down menu are presented on the header bar.

Buttons that allow you to perform additional actions are presented on the toolbar. For example, users with the **`PowerUserRole`** role will see the **Create Virtual Machine** button on the toolbar. However, users with the **`UserRole`** role will not see any buttons on the toolbar.

Virtual machines are presented in the virtual machines pane. This pane displays the operating system, name, state, and management menu for the virtual machines.

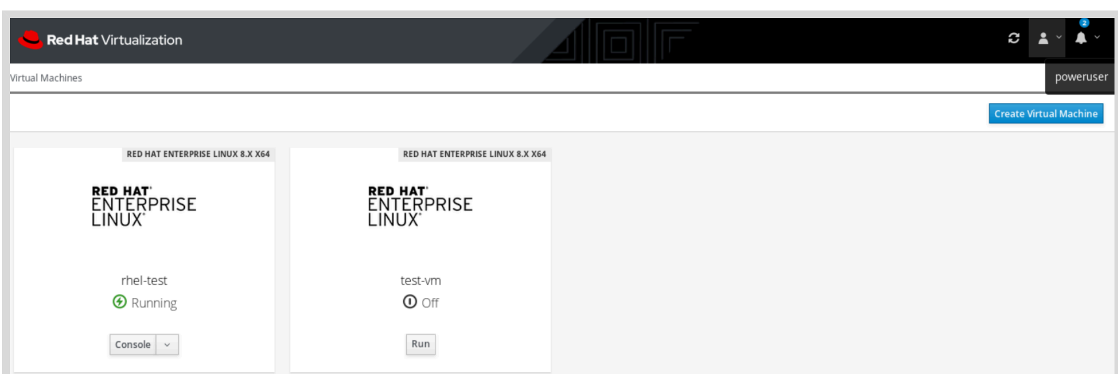

Figure 4.11: VM portal

Hover over the user drop-down menu in the header bar to get the current user. This drop-down menu has a **Log out** option for logging out of the **VM Portal**. The user menu also has **Options** (to add save SSH key) and **About** (to get portal and REST API versions).

The virtual machines allows you to access the console of a virtual machine. Every instance displaying the the operating system logo represents a virtual machine in an individual box called a *card*.

Each virtual machine card has a **Run** button, used to start the virtual machine. To access the console of the virtual machine, click the **Console** button. Use the drop-down menu on the virtual machine card to suspend, stop, and restart the virtual machine.

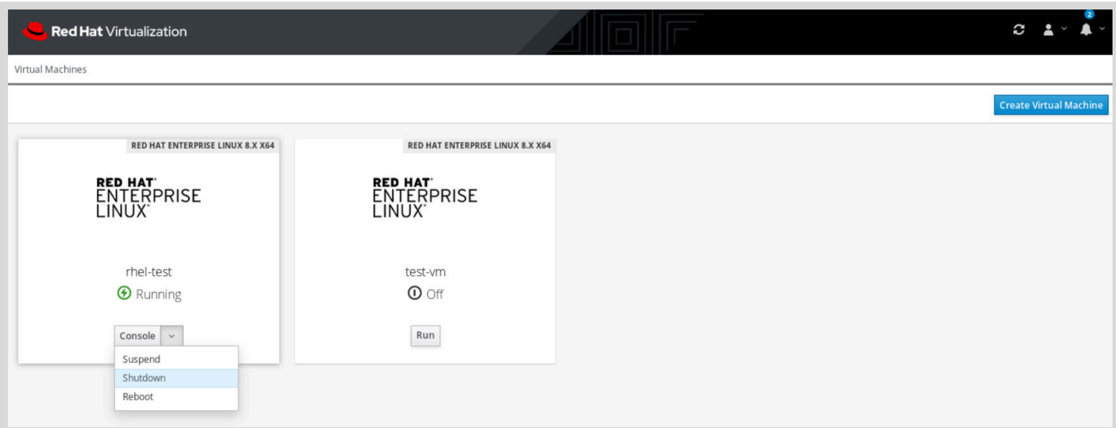

Figure 4.12: VM portal

To display the statistics of a particular virtual machine, click the name of the virtual machine.

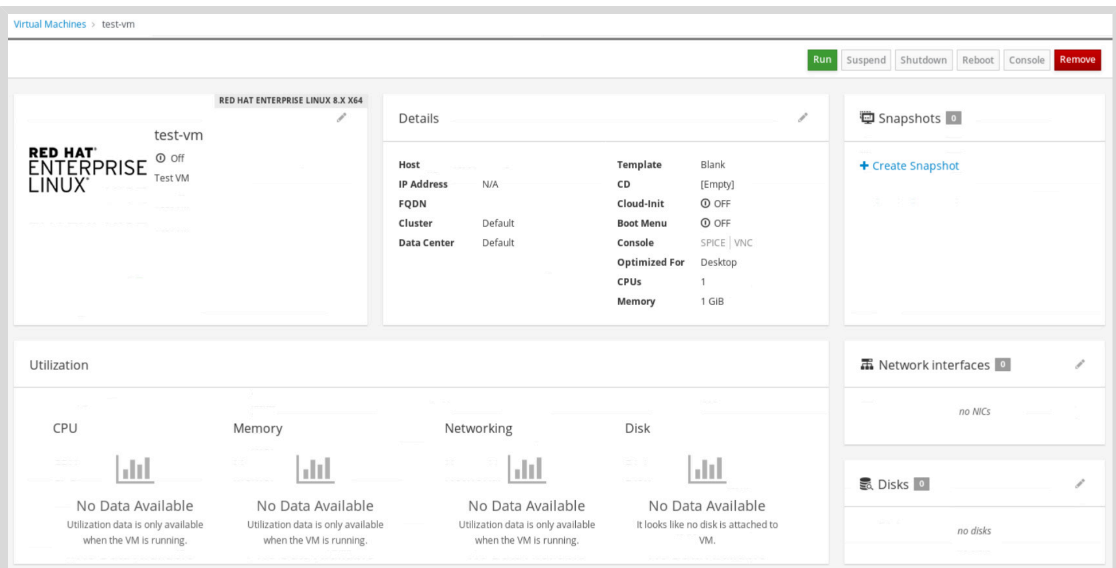

Figure 4.13: Virtual machine statistics

The **`PowerUserRole`** extends the functionality of the **VM Portal** for the users with this role because these users have additional permissions within the Red Hat Virtualization environment.

The following table outlines additional actions available and unavailable in the **VM Portal** to users with the **`PowerUserRole`** and **`UserRole`** roles respectively, .

Action	PowerUserRole	UserRole
Create, edit, and remove virtual machines	Yes	No

Action	PowerUserRole	UserRole
Manage virtual disks and network interfaces	Yes	No
Create and use virtual machine snapshots	Yes	No

Accessing and Using the Administration Portal

The **Administration Portal** is dedicated to administrators of the Red Hat Virtualization environment that are assigned at least one administrative role, such as the **`SuperUser`** or **`DataCenterAdmin`** role. This portal is designed for users who need to manage Red Hat Virtualization environment resources.

To access the **Administration Portal**, open your web browser and go to the server with Red Hat Virtualization Manager installed. Click the **Administration Portal** link to open the login page. Enter the administrative user name and password. Select the **`internal`** profile and click **Log In**.

The Red Hat Virtualization **Administration Portal** is divided into menus that appear in the navigation pane on the left. Navigate to resources using the options presented on these menus.

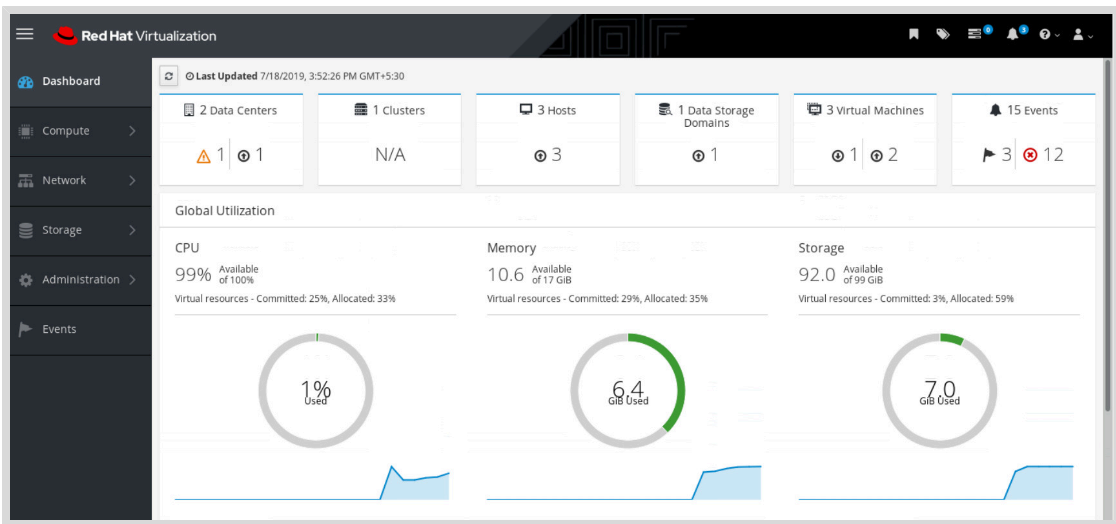

Figure 4.14: Administration portal

The following table outlines the actions available and unavailable in the **Administration Portal** to users with the **`SuperUser`** and **`DataCenterAdmin`** roles respectively.

Action	SuperUser	DataCenterAdmin
Create, and remove data centers	Yes	No
Edit data center	Yes	Yes
Create, and remove clusters within a data center	Yes	No
Edit cluster within a data center	Yes	Yes
Manage virtual machines within a cluster in the permitted data center	Yes	Yes

Searching Resources

The **Administration Portal** has a powerful search bar that you can use to effectively find resources in the Red Hat Virtualization infrastructure. Search supports both free-text searches and syntax-based searches. An autocomplete feature makes it easy for administrators who are unfamiliar with the search syntax to quickly construct effective searches. Frequently used searches can be bookmarked.

The formal search syntax has one required field: the type of resource to be returned in the search results. If no other parameters are provided, all resources are returned. Search results can be narrowed to match a rich set of search criteria, which is useful when looking for a single virtual machine or host on a site managing thousands of guest virtual machines on hundreds of hosts.

The following is the syntax for search bar expressions.

```
Result-type: [Criteria...] [sortby Sort_spec]
```

The *Result-type* part of the syntax is a Red Hat Virtualization resource such **`Vms`**, **`DataCenter`**, **`Host`**, **`Events`**, **`Cluster`**, **`Storage`**, or **`Template`**.

The following example shows a search for virtual machines that are configured with 2048 MB RAM or more.

```
Vms: memory >= 2028
```

Optional *Criteria* is an expression of the form: *Property Operator Value*.

- *Property* is the thing being matched, like machine name.
- *Operator* is how things are compared (= != > < >= <=).
- *Value* is the value to match.

An optional **`sortby`** directive can be associated with a sorting specification to sort the results based on various criteria in ascending or descending order.

References

Further information on the **Administration Portal** interface is available in the *Using the Administration Portal* chapter of *Introduction to the Administration Portal* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/introduction_to_the_administration_portal/chap-using_the_administration_portal

Further information on the **VM Portal** interface is available in the *Accessing the VM Portal* chapter of *Introduction to the VM Portal* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/introduction_to_the_vm_portal/index

Guided Exercise

Navigating the VM Portal and the Administration Portal

In this exercise, you will log in to the **VM Portal** and **Administration Portal** to observe how user roles determine the level of access in the Red Hat Virtualization environment.

Outcomes

You should be able to use the **VM Portal** and the **Administration Portal** to manage the Red Hat Virtualization environment.

Before You Begin

Four different users are mapped from the **`lab.example.com`** profile to the appropriate roles in the Red Hat Virtualization environment, as described below:

- **`rhvadmin`** to **`SuperUser`** role system-wide.
- **`normaluser`** to **`UserRole`** role system-wide.
- **`poweruser`** to **`PowerUserRole`** role system-wide.
- **`dcadmin`** to **`DataCenterAdmin`** role for the **`development`** data center.

Log in as the **`student`** user on **`workstation`** and run the **`lab users-portals start`** command. This command ensures that the domain users of the Red Hat Enterprise Linux Identity Management server in the classroom are mapped to the appropriate roles in the Red Hat Virtualization environment. This command also ensures that the **`development`** data center does not have any description.

```
[student@workstation ~]$ lab users-portals start
```

1. Log in to the **Administration Portal** as **`normaluser`** from the **`lab.example.com`** profile.

    1.1. On **`workstation`**, open Firefox and navigate to `https://rhvm.lab.example.com/ovirt-engine`. Click on the **Administration Portal** link. Log in as **`normaluser`** with the password **`redhat`**. Select the **`lab.example.com`** profile.

    1.2. Notice that the role assigned to **`normaluser`** prevents it from accessing the **Administration Portal**. In the upper right corner of the **Red Hat Virtualization** page, you can see that **`normaluser`** is successfully logged in. This page also indicates that **`normaluser`** is not authorized to access the **Administration Portal**.

2. Log in to the **VM Portal** as **`normaluser`** from the **`lab.example.com`** profile.

    2.1. The **`UserRole`** role is assigned to the **`normaluser`** user. This role permits the user to access only the **VM Portal**. Within the **VM Portal**, the user can only view the virtual machines. The **`normaluser`** user has access to the console of the virtual machine, and is able to start, stop, suspend, or reboot virtual machines.

2.2. While still logged in as **normaluser**, click on the **VM Portal** link in the **Red Hat Virtualization** page, located at `http://rhvm.lab.example.com/ovirt-engine`. You have successfully logged in to the **VM Portal** of Red Hat Virtualization.

You should see a running virtual machine called **rhel-vm1**. Click on the **Console** button of **rhel-vm1** to open its console. The **Console** dialog box opens with an error related to single sign-on. Ignore the error and click **Connect**. The **Opening console.vv** dialog box opens and prompts for your confirmation to open the connection file (**console.vv**). Click the **OK** button to open the **console.vv** file using Remote Viewer. Optionally, log in to the virtual machine as **root** with the password **redhat**.

> **Note**
> If Remote Viewer prompts for your confirmation to inhibit shortcuts, click **Allow**.

2.3. Click **View** → **Full screen** on the menu bar of the **rhel-vm1:1 - Remote Viewer** window to open the **rhel-vm1** virtual machine console in full-screen mode.

2.4. Hover your mouse over the upper part of the console and click the **X** button to close the virtual machine console. Click **OK** to confirm and close the console.

2.5. If the **rhel-vm1** virtual machine is running, click the drop-down button displayed near the **Console** button. Click **Shutdown**. The **Shutdown the VM** dialog box displays, prompting for confirmation to stop the virtual machine either forcefully or gracefully. Click **Yes** and wait for few seconds to gracefully stop the virtual machine. Notice that the **rhel-vm1** virtual machine is stopped.

> **Note**
> If you click the **Force** button, the virtual machine is immediately stopped without gracefully terminating the processes in the virtual machine, which is not a recommended practice.

2.6. Click the **Run** button to start the **rhel-vm1** virtual machine. Wait for a minute and notice that the virtual machine, **rhel-vm1**, is started.

2.7. Stop **rhel-vm1**.

2.8. Log out of the **VM Portal** as **normaluser**.

3. Log in to the **Administration Portal** as **poweruser** from the **lab.example.com** profile.

3.1. Click on the **Administration Portal** link.

3.2. Log in as **poweruser** with the password **redhat**. Select the **lab.example.com** profile.

3.3. Notice that the **PowerUserRole** role, assigned to **poweruser**, does not allow **poweruser** to access the **Administration Portal**. In the upper right corner of the **Red Hat Virtualization** page, you can see that **poweruser** is successfully logged in, but not authorized to access the **Administration Portal**.

4. Log in to the **VM Portal** as **poweruser** from the **lab.example.com** profile.

4.1. As **poweruser**, click the **VM Portal** link on the **Red Hat Virtualization** web page. You have successfully logged in to VM Portal of the Red Hat Virtualization.

4.2. The **PowerUserRole**, assigned to **poweruser**, allows this user to access the **VM Portal** with more privileges than are granted to users with the **UserRole** role. From within the **VM Portal**, the **poweruser** user has access to the **Create Virtual Machine** button to create a new virtual machine. The **poweruser** user can create new virtual machines based on existing templates.

4.3. Click the **Create Virtual Machine** button in the upper right corner of the page. The **Create a New Virtual Machine** page displays, which allows **poweruser** to create a virtual machine.

4.4. Click **Close** to close the **Create a New Virtual Machine** page.

4.5. Log out of the **VM Portal** as **poweruser**.

5. Log in to the **Administration Portal** as **dcadmin** from the **lab.example.com** profile.

5.1. Return to the **Red Hat Virtualization** web page. Click on the **Administration Portal** link.

5.2. Log in as **dcadmin** with the password **redhat**. Select the **lab.example.com** profile.

5.3. Notice that the **DataCenterAdmin** role assigned to **dcadmin** allows the user to access the **Administration Portal**.

6. Try to create a new storage domain called **nfs-data** as **dcadmin**. This operation fails because the **dcadmin** user does not have the required permissions to create a new storage domain.

6.1. While logged in as **dcadmin** in the **Administration Portal**, click **Storage** on the left navigation pane, and then click **Domains**.

6.2. Click **New Domain** to open the **New Domain** dialog box.

6.3. In the the **New Domain** dialog box, specify **nfs-data** as the name of the new storage domain in the **Name** field. Leave all the other fields except **Export Path** with their default values.

6.4. In the **Export Path** field, specify **utility.lab.example.com:/dcstorage** as the **Network File System** path for this new storage domain.

6.5. Click **OK** to create this new storage domain. As you can see, the **dcadmin** user is not allowed to create new storage domains in the data center.

6.6. Click **Close** to close the **Operation Canceled** dialog box. Click **Cancel** to close the **New Domain** dialog box.

Note that the **dcadmin** user has various privileges within the **Administration Portal**. For example, the **dcadmin** user can create, delete, start, and stop virtual machines, as well as hypervisor hosts.

7. Try to create a new data center called **operations** as **dcadmin**. This operation fails because the **dcadmin** user does not have the required permissions to create a new data center.

7.1. While logged in as **dcadmin** in the **Administration Portal**, click **Compute** on the left navigation pane, and then click **Data Centers**.

7.2. Click the **New** button to open the **New Data Center** dialog box.

7.3. In the the **New Data Center** dialog box, specify **operations** as the name of the new data center in the **Name** field. Leave all the other fields with their default values.

7.4. Click the **OK** button to create this new data center. As you can see, the **dcadmin** user is not allowed to create new data centers.

7.5. Click **Close** to close the **Operation Canceled** dialog box.

8. Edit the existing **development** data center as the **dcadmin** user to set **dcadmin example** to the description of the data center. This operation succeeds because the **dcadmin** user has permission to modify the existing data center called **development**.

8.1. While logged in as **dcadmin** in the **Administration Portal**, click **Compute** on the left navigation pane, and then click **Data Centers**.

8.2. Select the **development** data center from the list of available data centers.

8.3. Click the **Edit** button in the upper right part of the page. The **Edit Data Center** dialog box displays.

8.4. In the **Edit Data Center** dialog box, modify the **Description** field to have **dcadmin example** display the description of the **development** data center.

8.5. Click **OK** to confirm the change.

8.6. Notice that **dcadmin** was allowed to make changes to the specific data center because it is assigned the **DataCenterAdmin** role. The **dcadmin** user has the **DataCenterAdmin** role assigned only for the **development** data center.

8.7. Log out from the **Administration Portal** as **dcadmin**.

Finish

On **workstation**, run the **lab users-portals finish** script to complete this exercise.

```
[student@workstation ~]$ lab users-portals finish
```

This concludes the guided exercise.

Lab

Managing User Accounts and Roles

Performance Checklist

In this lab, you will assign additional roles to users in your Red Hat Virtualization environment and confirm that the roles have the expected effect.

Outcomes

You should be able to configure additional users with additional roles.

Before You Begin

The Red Hat Enterprise Linux Identity Manager is installed, preconfigured, and running on the **utility** system.

The Red Hat Virtualization environment is configured to use the external directory service running on the **utility** system.

Log in as the **student** user on **workstation** and run the **lab users-review start** command. This command ensures that the Red Hat Virtualization environment is configured to use the Red Hat Enterprise Linux Identity Management server as a source for user information.

```
[student@workstation ~]$ lab users-review start
```

1. Assign the system-wide **SuperUser** role for the entire Red Hat Virtualization environment to the **labadmin** user from the **lab.example.com** profile.
2. For the **Default** data center only, assign **DataCenterAdmin** to **labdcadmin** from the **lab.example.com** profile. Use the **labadmin** user from the **lab.example.com** profile to perform this step. The password for **labadmin** is **redhat**.
3. For only the **Default** data center, assign **PowerUserRole** to **labpoweruser** from the **lab.example.com** profile.
4. For only the **Default** data center, assign **UserRole** to **labnormaluser** from the **lab.example.com** profile.
5. Verify that the proper roles have been assigned to the selected users by checking the level of access to the **VM Portal** for both **labnormaluser** and **labpoweruser**.

Evaluation

On **workstation**, run the **lab users-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab users-review grade
```

Finish

On **workstation**, run the **lab users-review finish** script to complete this exercise.

```
[student@workstation ~]$ lab users-review finish
```

This concludes the lab.

Solution

Managing User Accounts and Roles

Performance Checklist

In this lab, you will assign additional roles to users in your Red Hat Virtualization environment and confirm that the roles have the expected effect.

Outcomes

You should be able to configure additional users with additional roles.

Before You Begin

The Red Hat Enterprise Linux Identity Manager is installed, preconfigured, and running on the **utility** system.

The Red Hat Virtualization environment is configured to use the external directory service running on the **utility** system.

Log in as the **student** user on **workstation** and run the **lab users-review start** command. This command ensures that the Red Hat Virtualization environment is configured to use the Red Hat Enterprise Linux Identity Management server as a source for user information.

```
[student@workstation ~]$ lab users-review start
```

1. Assign the system-wide **SuperUser** role for the entire Red Hat Virtualization environment to the **labadmin** user from the **lab.example.com** profile.

    1.1. On **workstation**, open Firefox and navigate to `https://rhvm.lab.example.com/ovirt-engine`. Click **Administration Portal** to log in to the web interface as the **rhvadmin** user, using the **lab.example.com** profile. The password is **redhat**.

    1.2. In the menu click **Administration**, then click **Configure**.

    1.3. In the **Configure** dialog box, click **System Permissions**.

    1.4. Click the **Add** button to add a role to a user.

    1.5. In the **Add System Permission to User** dialog box, select the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** option to use the user accounts from the Red Hat Enterprise Linux Identity Manager.

    1.6. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

    1.7. In the list of users that displays, select the check box for the **labadmin** user.

    1.8. Click the drop-down list under **Role to Assign**. From the list of available roles, select the **SuperUser** role for **labadmin**.

1.9. Click **OK** to assign the specified role to the selected user. Notice that the **`labadmin`** user displays in the **System Permissions** list. This list confirms that the **`labadmin`** user has been assigned a role granting administrative access to Red Hat Virtualization.

1.10. In the **Configure** dialog box, click **Close**.

1.11. Sign out as **`admin`** from the **Administration Portal**.

2. For the **`Default`** data center only, assign **`DataCenterAdmin`** to **`labdcadmin`** from the **`lab.example.com`** profile. Use the **`labadmin`** user from the **`lab.example.com`** profile to perform this step. The password for **`labadmin`** is **`redhat`**.

2.1. Click **Administration Portal** to log in to the web interface as the domain user called **`labadmin`** with **`redhat`** as the password. Select the **`lab.example.com`** profile while logging in.

2.2. In the web interface, click **Compute** on the navigation pane on the left, and then click **Data Centers** from the options that display.

2.3. Click on the name of the **`Default`** data center.

2.4. In the page that displays, click on the **Permissions** tab.

2.5. Click the **Add** button to add a role to a user from the Red Hat Enterprise Linux Identity Manager.

2.6. In the **Add Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item so that you can use the user accounts from the Red Hat Enterprise Linux Identity Manager.

2.7. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

2.8. In the list of users that displays, click the check box for the **`labdcadmin`** user.

2.9. Click the drop-down list under **Role to Assign**. From the list of available roles, select the **`DataCenterAdmin`** role for **`labdcadmin`**.

2.10. Click **OK** to assign the specified role to the selected user. Notice that the **`labdcadmin`** user displays in the **Permissions** list. This list confirms that the **`labdcadmin`** user has been assigned a role granting access to Red Hat Virtualization for the specific data center.

3. For only the **`Default`** data center, assign **`PowerUserRole`** to **`labpoweruser`** from the **`lab.example.com`** profile.

3.1. From the same **Permissions** tab for the **`Default`** data center, click the **Add** button.

3.2. In the **Add Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item so that you can use the user accounts from the Red Hat Enterprise Linux Identity Manager.

3.3. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

3.4. In the list of users that displays, select the check box for the **`labpoweruser`** user.

3.5. Click the drop-down list under **Role to Assign**. From the list of available roles, select the **`PowerUserRole`** role for **`labpoweruser`**.

3.6. Click **OK** to assign the specified role to the selected user. Notice that the **`labpoweruser`** user displays in the **Permissions** list. This list confirms that the **`labpoweruser`** user has been assigned a role granting access to Red Hat Virtualization for the specific data center.

4. For only the **`Default`** data center, assign **`UserRole`** to **`labnormaluser`** from the **`lab.example.com`** profile.

4.1. From the same **Permissions** tab for the **`Default`** data center, click the **Add** button.

4.2. In the **Add Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item to use the user accounts from the Red Hat Enterprise Linux Identity Manager.

4.3. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

4.4. In the list of users that displays, select the check box for the **`labnormaluser`** user.

4.5. Click the drop-down list under **Role to Assign**. From the list of available roles, select the **`UserRole`** role for **`labnormaluser`**.

4.6. Click **OK** to assign the specified role to the selected user. Notice that the **`labnormaluser`** user displays in the **Permissions** list. This list confirms that the **`labnormaluser`** user has been assigned a role granting access to Red Hat Virtualization for the specific data center.

4.7. Sign out as **`labadmin`** from the **Administration Portal**.

5. Verify that the proper roles have been assigned to the selected users by checking the level of access to the **VM Portal** for both **`labnormaluser`** and **`labpoweruser`**.

5.1. On the **Red Hat Virtualization** page located at `http://rhvm.lab.example.com/ovirt-engine`, click the **VM Portal** link.

5.2. Log in as **`labnormaluser`** with the password **`redhat`**. Select the **`lab.example.com`** profile while logging in.

The virtual machines that exist in the Red Hat Virtualization classroom environment are displayed. The **`labnormaluser`** user is allowed to start, stop, suspend, or reboot virtual machines, and can also access the console for those virtual machines. This confirms that the proper role has been assigned to **`labnormaluser`**.

5.3. Log out from the **VM Portal**.

5.4. Log back in to the **VM Portal**, this time as **`labpoweruser`** with the password **`redhat`**. Select the **`lab.example.com`** profile while logging in.

The **Create Virtual Machine** button displays. The presence of this button confirms that **`labpoweruser`** has the proper role assigned.

5.5. Sign out as **`labpoweruser`** from the **VM Portal**.

Evaluation

On **`workstation`**, run the **`lab users-review grade`** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab users-review grade
```

Finish

On **workstation**, run the **lab users-review finish** script to complete this exercise.

```
[student@workstation ~]$ lab users-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- Using the Red Hat Enterprise Linux Identity Manager as a source for your Red Hat Virtualization environment, simplifies user and group management.
- Users can be assigned roles which apply to the entire Red Hat Virtualization environment, or only to a specific object (such as a virtual machine or a data center).
- The **`admin@internal`** account is created at installation time as a default user that has the system-wide **`SuperUser`** role.
- The **VM Portal** allows for easy access to the console of a virtual machine, as well as the ability to start, stop, restart, or shutdown.
- The **Administration Portal** is dedicated to administrators of your Red Hat Virtualization environment. This portal is designed for users who need to manage the Red Hat Virtualization environment resources.

Chapter 5

Scaling RHV Infrastructure

Goal	Add and remove Red Hat Virtualization Hosts, both manually and with automated provisioning.
Objectives	• Remove hosts from an existing cluster for maintenance or reassignment, and add hosts to increase the capability of a cluster in a data center. • Configure a network installation and kickstart server as provisioning tools in the Red Hat Virtualization environment.
Sections	• Adding and Removing Hosts in a Cluster (and Guided Exercise) • Automating Host Provisioning (and Guided Exercise)
Quiz	Scaling RHV Infrastructure

Adding and Removing Hosts in a Cluster

Objectives

After completing this section, you should be able to remove a host from an existing data center for maintenance or reassigning, and add a host to increase the capability of a cluster in a data center.

Scaling the Infrastructure

Enterprise virtualization workloads can be diverse and complex; Red Hat Virtualization is designed to be flexible enough to support even the most demanding requirements. RHV data centers and clusters can be reconfigured to match changing requirements, providing an ability to scale both smaller and larger, to reconfigure existing clusters for new workloads, and to reprovision hypervisor hosts to handle those workloads.

Hypervisor hosts may be added or removed from clusters for a number of reasons, including:

- Increasing the capacity of a cluster to run more or larger virtual machines.
- Replacing a failed or damaged host.
- Increasing the redundant capacity of a host to handle incoming failover workloads.
- Reducing the capacity of an over-provisioned cluster.
- Retiring a cluster and repurposing the hosts to other clusters.
- Temporarily removing hosts to perform maintenance tasks on host software or hardware.
- Modifying or replacing the connect data storage domain or related storage infrastructure on a host.
- Modifying or replacing the host networking infrastructure.

Using Maintenance Mode

Many common maintenance tasks, including network configuration and deployment of software updates, require that hosts be placed into *maintenance mode*. Hosts should be placed into maintenance mode before any event that might cause **`VDSM`** to stop working properly, such as a reboot, or issues with networking or storage. Because **`VDSM`** provides the communication between the RHV Manager and hosts, placing a host in maintenance mode signals **`VDSM`**, which temporarily disables engine health checking. **`VDSM`** does not stop while the host is in maintenance mode.

When a host is placed in maintenance mode, the RHV Manager attempts to migrate all running virtual machines to alternative hosts. The standard prerequisites for live migration apply, in particular there must be at least one active host in the cluster with capacity to run the migrated virtual machines.

Virtual machines that are pinned to the host and cannot be migrated are shut down. Check which virtual machines are pinned to the host by clicking **Pinned to Host** in the **Virtual Machines** tab of the host **Details** screen.

If the host is the **`Storage Pool Manager (SPM)`**, the SPM role is migrated to another host. The **Status** field of the host changes to **`Preparing for Maintenance`**, and finally **`Maintenance`** when the operation completes successfully. Advanced host editing options become available in the **Edit Host** window.

Placing a Data Center into Maintenance Mode

If a RHV host is the last active host in a data center, then the data center must also be placed into maintenance mode because a data center with no hosts cannot function. Placing the data center in maintenance mode will place all of the data center storage domains into maintenance mode. The **`Storage Pool Manager`**, which was migrated to the last active host as other hosts were placed in maintenance mode or shut down, will change the data center storage configuration as storage domains become inactive.

When all storage domains in a data center are in maintenance mode, the data center automatically transitions to maintenance mode and the **`SPM`** no longer has storage tasks to manage. The data center stops producing log output. Log output restarts only when the master storage domain becomes active again. The last active host may now be placed in maintenance mode.

Removing a Host from Red Hat Virtualization

To remove a host from the RHV infrastructure, use the **`Administration Portal`** as a user with a sufficient role. Select the host in the **Hosts** table, then select **Maintenance** from the drop-down list on the **Management** button. Existing virtual machines on the host may need to be manually migrated or shut down before the host can enter maintenance mode. When the host displays a **`Maintenance`** status, the **Remove** button becomes active. Verify that the correct host is selected in the table, then click the **Remove** button, and then click the **OK** button to confirm the removal.

Removing the host only removes its cluster and data center associations in the RHV engine database, but does not physically remove or wipe the host system, nor alter the host operating system configuration. Unless maintenance tasks or additional storage or network configuration is required, the host can be added to any existing cluster where it meets the joining criteria, such as a sufficient CPU family architecture.

Adding a Host to Red Hat Virtualization

After a RHV host has been installed, it is added to a Red Hat Virtualization environment by joining a data center and, on the same screen, a cluster of the selected data center. Logical networks and storage domains created in a data center then become available to the host. You must configure a host to connect to the storage domains and logical networks required by the cluster it has joined.

To function properly, all hosts in a single cluster must share access to the same storage domains and logical networks, which requires that host be configured to access the shared storage back ends and network segments. Hosts in the same cluster should standardize on the same CPU vendor and model, storage adapter and network interface hardware, and firmware versions, for consistent behavior and performance.

To add a host to the RHV infrastructure, use the **`Administration Portal`** as a user with a sufficient role. In the **Hosts** screen, click the **New** button to open the **New Host** window. In addition to entering a unique name and a helpful comment, you must provide the correct information for functioning SSH access. For stronger security, SSH public key access is preferred.

The **Hostname** field can be entered as either a fully qualified domain name (FQDN) or as the host IP address. To function properly, the host FQDN must be DNS resolvable, both forward and reverse.

By default, the RHV Manager automatically updates the host firewall rules to allow the required network protocols. If you wish to manually manage the host firewall rules, open the **Advanced Parameters** accordion icon to disable this behavior by clearing the **Automatically configure host firewall** checkbox.

Moving a Host Between Clusters or Data Centers

To move a host from one cluster to another, use the **`Administration Portal`** as user with a sufficient role. Select the host in the **Hosts** table, and place it in maintenance mode as previously described. When the host transitions to the **`Maintenance`** status, click the **Edit** button to open the **Edit Host** window.

Using the **Host Cluster** drop-down list, choose a new cluster in the same or a different data center. Click the **OK** button twice, to exit the editing window and confirm the move. The host remains in maintenance mode until you activate it using the **Activate** option on the **Management** button drop-down list. When the host successfully activates, its status icon changes from red to green to indicate that the host is now available in the new cluster and data center.

References

Further information is available in the "Data Centers" chapter of the *Administration Guide* for Red Hat Virtualization
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/chap-users_and_roles#chap-Data_Centers

Guided Exercise

Adding and Removing Hosts in a Cluster

In this exercise, you will remove **hostb.lab.example.com** from the **Default** cluster, and then add it to **clusterone**.

Outcomes

You should be able to remove a host from a cluster, add it to a new cluster, and locate system events in the **Events** log.

Before You Begin

Log in to **workstation** as the **student** user, using **student** as the password.

On **workstation**, run the **lab hypervisor-remove start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab hypervisor-remove start
```

1. Before moving **hostb.lab.example.com** into **Maintenance** mode, ensure that all virtual machines hosted by **hostb.lab.example.com** are powered off.

   1.1. Log in to **workstation** as **student** using **student** as the password. Open Firefox and log in to the **Administration Portal** using the https://rhvm.lab.example.com URL. The username is **admin**, the domain is **internal**, and the password is **redhat**.

   1.2. Click on **Compute**, then click on **Virtual Machines**.

   1.3. Review the **Host** column for each virtual machine. If the column has a value of **hostb.lab.example.com**, right-click the name of the virtual machine and select **Shutdown**. Click on **Confirm** to shut down the virtual machine.

> **Important**
> Do not attempt to switch off the **Hosted Engine** virtual machine. This virtual machine is the self-hosted RHV-M installation.

2. Put **hostb.lab.example.com** into **Maintenance** mode.

   2.1. Click on **Compute**, and then click on **Hosts**.

   2.2. Click on **hostb.lab.example.com**.

   2.3. From the **Management** drop-down menu, select **Maintenance** to put **hostb.lab.example.com** into maintenance mode. This opens the **Maintenace Host(s)** window. In the **Reason** box type, **removing hostb.lab.example.com from the cluster**. Click on **OK**.

2.4. Confirm that the **Status** of **`hostb.lab.example.com`** is **Maintenance**.

3. Remove **`hostb.lab.example.com`** from the **`Default`** cluster. Locate the system event messages in **`Events`**.

3.1. Click on **`hostb.lab.example.com`**. Click on the **Remove** button. This opens the **Remove Host(s)** window. In the **Reason** box, type **`removing hostb.lab.example.com from the default cluster`**. Click on **OK**.

3.2. Click on **Events**. Notice the events for **`hostb.lab.example.com`**.

The events below are listed in the order in which they occur. In the **`Administration Portal`**, the event order may be selected by clicking the **Time** column heading in the **Events** table.

- Ansible host-remove playbook execution started on host hostb.lab.example.com.
- Ansible host-remove playbook execution has successfully finished on host hostb.lab.example.com. For more details, check log /var/log/ovirt-engine/ansible/ansible-20190722023326-ovirt-host-remove_yml.log
- Host hostb.lab.example.com was removed by admin@internal-authz.

> **Note**
> It can take several minutes for all the system event messages to appear.

4. Add **`hostb.lab.example.com`** to **`clusterone`**.

4.1. Click on **Compute**, and then click on **Hosts**.

4.2. Click on the **New** button. This opens the **New Host** window.

4.3. In the **New Host** window enter the following configuration settings:

- Choose **`clusterone`** as the Host Cluster.
- Enter **`hostb.lab.example.com`** in the **Name** field.
- Enter **`hostb.lab.example.com`** in the **Hostname** field.
- Enter **`redhat`** in the **Password** field.

Click the **OK** button to add **`hostb.lab.example.com`** to the cluster.

4.4. Click **OK** in the **Power Management Configuration** window.

The **Status** of **`hostb.lab.example.com`** is **`Installing`**. It takes several minutes for the **Status** to change to **Up**.

5. Click on **Events** on the menu bar to locate the system event messages. Notice the events for **`hostb.lab.example.com`**.

The events below are listed in the order in which they occur. In the **`Administration Portal`**, the event order may be selected by clicking the **Time** column heading in the **Events** table.

- Ansible host-deploy playbook execution has successfully finished on host hostb.lab.example.com.
- Host hostb.lab.example.com installed.
- No faulty multipath paths on host hostb.lab.example.com.
- Status of host hostb.lab.example.com was set to Up.

Finish

On **workstation**, run the **lab hypervisor-remove finish** script to complete this exercise.

```
[student@workstation ~]$ lab hypervisor-remove finish
```

This concludes the guided exercise.

Automating Host Provisioning

Objectives

After completing this section, you should be able to configure a network installation and kickstart server as provisioning tools in the Red Hat Virtualization environment.

Configuring a Network Boot Server

Red Hat Virtualization Host (RHV-H) can be installed manually, which will not scale in a large enterprise environment with many hosts. Since RHV-H is built from RHEL, RHV-H uses the same Anaconda installer, which you can automate using a Kickstart configuration.

To create a fully automated, unattended installation, use a network installation server configured with PXE, TFTP, and a shared Kickstart file to start an installation by booting from the network.

Booting from the Network Using PXE

Preboot eXecution Environment (PXE) is a computer bootstrap mechanism using a network server. The client network interface must support PXE, and the system firmware must enable PXE support.

The network boot infrastructure must provide the following services:

- A DHCP server to handle the initial client communication, the network configuration, and the TFTP server location with a usable boot image.
- A TFTP server to provide boot images with command-line options, to start the installer.
- An HTTP, FTP, or NFS server to provide the installation media and the Kickstart file for this installation.

At boot, the client's network interface card broadcasts a DHCPDISCOVER packet extended with PXE-specific options. A DHCP server on the network replies with a DHCPOFFER, giving the client information about the PXE server and offering an IP address. When the client responds with a DHCPREQUEST, the server sends a DHCPACK with the Trivial FTP (TFTP) server URL of a file that can boot the client into an installer program.

The client downloads the file from the TFTP server (frequently the same system as the DHCP server), verifies the file using a checksum, and loads the file. Typically, the file is a network boot loader called **`pxelinux.0`**. This boot loader accesses a configuration file on the TFTP server that tells it how to download and start the RHV-H installer, and how to locate the Kickstart file on an HTTP, FTP, or NFS server. After verification, the files are used to boot the client.

Configuring a PXE Boot Service

To configure an automated network installation of RHV-H, perform the following tasks:

- Configure a DHCP server to use PXE, and then point your RHV-H clients to the TFTP server and the **`pxelinux.0`** file.
- Configure a TFTP server to provide the **`pxelinux.0`** file, a configuration file that points to the RHV-H installer kernel and initial ram disk image, and to the location of the Kickstart file.
- Share the RHV-H installation media and the Kickstart file using a supported network service such as NFS, HTTPS, HTTP, or FTP.

Configuring the DHCP and TFTP Server

This example configuration uses a Red Hat Enterprise Linux network installation server for booting BIOS-based AMD64 and Intel 64 systems. In this example, the DHCP and TFTP servers are on the same system, with the IPv4 **`172.25.250.8`** address.

> **Important**
>
> This example procedure will not boot hosts using UEFI-based boot firmware. UEFI-based systems require extra files from the *shim* and *grub2-efi* packages, and a different configuration file.
>
> For more information, see the section "Configuring a TFTP Server for UEFI-based AMD64 and Intel 64 Clients" in the *Installation Guide* for Red Hat Enterprise Linux 8 at https://access.redhat.com/documentation/.

Install a Red Hat Enterprise Linux server with the *syslinux*, *tftp-server*, and *dhcp* packages. In the **`/var/lib/tftpboot`** directory, create a **`pxelinux`** directory. Copy the **`/usr/share/syslinux/pxelinux.0`** file into **`/var/lib/tftpboot/pxelinux`**. In the **`/var/lib/tftpboot/pxelinux`** directory, create a **`pxelinux.cfg`** directory. This directory contains the **`default`** configuration file that is used by any system PXE-booting from this TFTP service. The following example shows the **`default`** configuration file:

```
default vesamenuc32
prompt 1
timeout 60

display boot.msg

label rhvh-host
  menu label ^Install RHV-H host
  menu default
  kernel vmlinuz
  append initrd=initrd.img ip=dhcp inst.stage2=http://install-server/RHVH-
installation-media-directory inst.ks=http://install-server/kickstart-file-
directory/kickstart-file.cfg
```

The important parts of this configuration file are:

- **`label rhvh-host`** is the boot loader configuration for the RHV-H installation, which appears in the menu as **`Install RHV-H Host`**
- The **`vmlinuz`** and **`initrd.img`** files are provided by TFTP from **`/var/lib/tftpboot/pxelinux`**.
- The **`inst.stage2`** directive is an HTTP server (*install-server*) URL for the installation media. The **`inst.ks`** directive is an HTTP server URL for the Kickstart file.

Copy the boot image from inside the RHV-H ISO file to the **`/var/lib/tftpboot`** directory. Set up the DHCP server configuration file, **`/etc/dhcp/dhcpd.conf`**.

The following example provides basic network information for a 172.25.250.0/24 subnet, and specifies a TFTP server at 172.25.250.8, which is the same server. Clients download and boot the **`pxelinux/pxelinux.0`** file from the TFTP server.

```
option space pxelinux;
option pxelinux.magic code 208 = string;
option pxelinux.configfile code 209 = text;
option pxelinux.pathprefix code 210 = text;
option pxelinux.reboottime code 211 = unsigned integer 32;
option architecture-type code 93 = unsigned integer 16;

subnet 172.25.250.0 netmask 255.255.255.0 {
  option routers              172.25.250.254;
  option subnet-mask          255.255.255.0;
  option domain-search        "lab.example.com";
  option domain-name-servers  172.25.250.254;

  range 172.25.250.21 172.25.250.30;

  class "pxeclients" {
      match if substring (option vendor-class-identifier, 0, 9) = "PXEClient";
      next-server 172.25.250.8;

      if option architecture-type = 00:07 {
        filename "uefi/shim.efi";
      } else {
        filename "pxelinux/pxelinux.0";
      }
  }
}
```

Configure the firewall to allow the necessary services. If **`firewalld`** is configured, use the predefined TFTP and DHCP services. Start and enable the **`dhcpd`** and **`tftp`** services. When finished, the PXE boot server is ready.

Configuring an Installation Media and Kickstart Server

The PXE server **`default`** configuration file specified two files required to start an installation, a live image file containing the RHV-H operating system, and a Kickstart file. These files can be shared using HTTP, NFS or FTP. This example uses HTTP.

Providing the RHV-H Live Image

Prepare your web server for an unattended network installation of new RHV-H hosts by extracting the RHV-H live image from the RHV-H installation ISO. Download the RHV-H ISO image file to a local directory and mount it. Locate the large **`redhat-virtualization-host-image-`*`version`*`.rpm`** and unarchive it. Locate the **`squashfs`** configuration file in the unarchived rpm directory.

Copy the **`redhat-virtualization-host-`*`version`*`.squashfs.img`** to your web server directory. Rename the file to make it easier to reference inside the Kickstart file. In this example, the file was renamed to **`squashfs.img`**. The **`squashfs.img`** file contains the RHV-H operating system with all required packages, and it automatically deploys the new RHV-H host.

Creating a RHV-H Kickstart File

An unattended network image Kickstart file is simpler than a typical Kickstart file because you do not need to specify any packages to install, as they are all contained in **squashfs.img**. Even the partition layout is specified and automatically created using an LVM Thin Provisioning mechanism.

Client systems can be configured to always first boot to the network, to allow systems to be reinstalled remotely without manual intervention. PXE servers can be configured to enable or disable the ability to respond to individual clients, thus providing installation service only to designated clients. PXE servers can also be configured to respond to PXE booting requests, in a manner that instructs the client to boot from a local disk instead.

> **Warning**
> Configuring clients to automatically boot into an unattended network installation can result in accidental host re installation. Red Hat recommends that you enable PXE and DHCP configurations only on demand, and disable individual host configuration as soon as the RHV-H host installation and deployment is complete.

Create the Kickstart file using any text editor, using a descriptive name that makes it easy to remember and select the correct file among those stored on the Kickstart server. The following example shows a Kickstart file for an automatic RHV-H installation:

```
liveimg --url=http://install-server/RHVH-installation-media-directory/squashfs.img
clearpart --all
autopart --type=thinp
zerombr
rootpw --plaintext root_password_in_clear_text
timezone Etc/UTC --isUtc
text
reboot
%post --erroronfail
nodectl init
%end
```

The **liveimg** parameter specifies the HTTP URL for the RHV-H installation tree on the network.

The **clearpart --all** parameter removes any existing partitions. The **autopart --type=thinp** parameter creates a new LVM-based layout, and the **zerombr** parameter clears the MBR.

The **rootpw --plaintext** parameter sets the root password. The **timezone Etc/UTC --isUtc** parameter sets the time zone. The **text** parameter sets the UI mode for the installer, which is text mode in this example. The **reboot** parameter reboots the host after the installation completes.

The **%post** section starts the configuration process of the newly installed RHV-H host.

This example can be extended with additional commands and options, but it is sufficient to automatically install a new RHV-H host.

Share the new Kickstart file using HTTP, NFS, or FTP. The earlier example **pxelinux.cfg** file specified the web server directory (**/var/www/html/*kickstart-file-directory*/*kickstart-file.cfg***).

Starting the Automated Installation

The example presented in this section configures an automated RHV-H Kickstart installation whenever a recognized node PXE-boots. The behavior is controlled by the **`/var/lib/tftpboot/pxelinux/pxelinux.cfg`** file on the example server. When using UEFI-based systems, the configuration will be slightly different.

The boot loader specified in the Kickstart file downloads and starts the **`vmlinuz`** kernel from the RHV-H installation ISO file. That kernel is invoked with four command-line kernel parameters:

- **`initrd=initrd.img`** to download the initial RAM disk image (**`initrd.img`**) that was collocated with the **`vmlinuz`** file on the RHV-H ISO.
- **`ip=dhcp`** to obtain an IP address using DHCP.
- An **`inst.stage2`** directive pointing to a shared URL containing the **`squash.img`** file.
- An **`inst.ks`** directive pointing to a shared URL for the Kickstart file.

Satellite as a Provisioning Server

Using Red Hat Satellite Server, an administrator can provision many hosts concurrently. Red Hat Satellite Server has multiple methods for provisioning hosts.

The Bare Metal Provisioning method provisions hosts using PXE boot, although environments without PXE can use other methods. The Cloud Providers method connects to private and public cloud providers to provision hosts. The Cloud Providers method uses images stored in the cloud environment, whereas Bare Metal Provisioning requires the installation files to be available on the local network. Satellite Server can also connect to a VMware ESX, or other virtualization infrastructure, to provision hosts. Satellite Server can also use Linux containers to provision hosts.

To avoid configuring DHCP or TFTP servers on the network, *cloud-init* can be used to provision hosts. You can download a template from the Red Hat Customer portal and upload it to an RHV-H host. The *cloud-init* template can also be created using a standard RHEL 8 image. The image, either downloaded or created from the RHEL 8 image, is then uploaded to Satellite Server. A provisioning template is then created in Satellite, with a specific configuration for *cloud-init*.

> **Note**
> Customizing deployed virtual machines using *cloud-init* is covered in a later chapter.

Adding a Satellite host provider in RHV-M is almost the same as adding a new RHEL host. Selecting **Use Foreman/Satellite** displays the options for adding a Satellite host provider. The default option, **Discovered Hosts**, allows the administrator to choose **Discovered Hosts**, **Host Groups**, and **Compute Resources** from a drop-down menu.

Satellite Server has many advantages over Kickstart using Anaconda. For example, Satellite Server allows administrators to provision grouped machines, using HostGroups. Target hosts can be customized easily using Puppet support in Satellite Server. Enterprise grade provisioning should always be done using Satellite Server. A simple Kickstart provisioning uses sub-management from Red Hat, for example, pulling down packages from Red Hat servers. Satellite Server uses local package installation from servers on the company side of the firewall. Satellite Server also allows the administrator to have complete control. For example, different classes of virtual machines can have fine-grained customization. In addition to provisioning, Satellite also provides ongoing support for upgrades.

References

Further information on automated installation of RHV-H is available in the "Advanced Installation" section of the *Installation Guide* for Red Hat Virtualization 4.3 at
https://access.redhat.com/documentation/

Further information on how to set up a Red Hat Enterprise Linux-based PXE service using **dhcpd** and **tftpd** is available in the chapter "Preparing for a Network Installation" in the *Installation Guide* for Red Hat Enterprise Linux 8 at
https://access.redhat.com/documentation/

Further information on DHCP server configuration for Red Hat Enterprise Linux is available in the chapter "DHCP Servers" in the *Networking Guide* for Red Hat Enterprise Linux 8 at
https://access.redhat.com/documentation/

Guided Exercise

Automating Host Provisioning

In this exercise, you will explore the network installation server and the deployment environment.

Outcomes

You should be able to explore the network installation and Kickstart server.

Before You Begin

Log in to **`workstation`** as the **`student`** user, using **`student`** as the password.

On **`workstation`**, run the **`lab hypervisor-kickstart start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab hypervisor-kickstart start
```

1. Download the **`RHVH-4.3-20190512.3-RHVH-x86_64-dvd1.iso`** file.

   1.1. From **`workstation`**, open a terminal and log into the **`utility.lab.example.com`** server as the **`root`** user using **`ssh`**.

```
[student@workstation ~]$ ssh root@utility.lab.example.com
[root@utility ~]#
```

   1.2. Change into the **`/tmp`** directory.

```
[root@utility ~]# cd /tmp
```

   1.3. Use the **`wget`** command to download the **`RHVH-4.3-20190512.3-RHVH-x86_64-dvd1.iso`** file.

```
[root@utility tmp]# wget \
http://content.example.com/rhv4.3/x86_64/isos/\
RHVH-4.3-20190512.3-RHVH-x86_64-dvd1.iso
...output omitted...
Saving to: ‘RHVH-4.3-20190512.3-RHVH-x86_64-dvd1.iso.1’

RHVH-4.3-20190512.3-RHVH-x86_64-dvd1.
 100%[======================================================================>]
 1.05G   450MB/s    in 2.4s

2019-07-26 12:14:06 (450 MB/s) - ‘RHVH-4.3-20190512.3-RHVH-x86_64-dvd1.iso.1’
 saved [1131413504/1131413504]
```

2. Mount the **ISO** image and find the files used during a **PXE** boot.

   2.1. Using the **mount** command, mount the **ISO** image to the **/mnt** mount point.

```
[root@utility tmp]# mount RHVH-4.3-20190512.3-RHVH-x86_64-dvd1.iso /mnt
mount: /dev/loop0 is write-protected, mounting read-only
```

   2.2. List the contents of the **/mnt/images** directory.

   The files in this directory are used for PXE booting.

```
[root@utility tmp]# ls -al /mnt/images
total 8717
dr-xr-xr-x. 3 root root    2048 May 12 13:45 .
dr-xr-xr-x. 8 root root    2048 May 12 13:45 ..
-r--r--r--. 1 root root 8919040 May 12 13:44 efiboot.img
-r--r--r--. 1 root root     280 May 12 13:43 product.img
dr-xr-xr-x. 2 root root    2048 May 12 13:45 pxeboot
-r--r--r--. 1 root root     446 May 12 13:45 TRANS.TBL
```

3. Find the **squash** file that is required for **Kickstart** configuration.

   3.1. List the contents of the **/mnt/Packages** directory.

```
[root@utility tmp]# ls -al /mnt/Packages
total 606494
dr-xr-xr-x.  2 root root      2048 May 12 13:45 .
dr-xr-xr-x.  8 root root      2048 May 12 13:45 ..
-r--r--r--.  1 root root 621026735 May 12 13:28 redhat-virtualization-host-image-
update-4.3-20190512.0.el7_6.noarch.rpm
-r--r--r--. 22 root root     17140 Nov 22  2018 redhat-virtualization-host-
productimg-4.3-0.0.el7.x86_64.rpm
-r--r--r--.  1 root root       555 May 12 13:45 TRANS.TBL
```

   3.2. Use **rpm2cpio** to extract the **redhat-virtualization-host-image-update-4.3-20190512.0.el7_6.noarch.rpm** file.

```
[root@utility tmp]# rpm2cpio \
/mnt/Packages/\
redhat-virtualization-host-image-update-4.3-20190512.0.el7_6.noarch.rpm \
| cpio -idmv
./usr/share/redhat-virtualization-host/image
./usr/share/redhat-virtualization-host/image/redhat-virtualization-
host-4.3-20190512.0.el7_6.squashfs.img
./usr/share/redhat-virtualization-host/image/redhat-virtualization-
host-4.3-20190512.0.el7_6.squashfs.img.meta
```

   The **redhat-virtualization-host-4.3-20190512.0.el7_6.squashfs.img** file is used in the Kickstart configuration file. To use the file for Kickstart it must first be copied to the installation tree directory. Rename the file to **squashfs.img**.

4. Create a new NFS export on the **utility** server to provide your Kickstart file.

4.1. Return to the **root** user's home directory. Create a new directory on **utility** called **/ks**.

```
[root@utility tmp]# cd
[root@utility ~]# mkdir /ks
```

4.2. Edit the **/etc/exports** file to export the new **/ks** directory with read permission to ***.lab.example.com** using NFS.

The **utility** server is already an NFS server for other exported file systems; if any configuration already exists in this file, then you must not change it. Add the configuration to the bottom of the file. The file should display as follows:

```
...output omitted...
/ks            *.lab.example.com(ro)
```

Save the file and exit the editor.

4.3. Export the directories shared by the NFS daemon by issuing the **exportfs -r** command.

```
[root@utility ~]# exportfs -r
```

5. Inside the **/ks** directory, create a new Kickstart file to automate the installation of your Red Hat Virtualization Hosts.

5.1. On **utility**, create a new file called **/ks/rhvh.cfg**. Open this file using a text editor, and then type in the following lines:

```
liveimg --url=http://content.example.com/rhv4.3/x86_64/squashfs.img
clearpart --all
autopart --type=thinp
rootpw --plaintext redhat
timezone Etc/UTC --isUtc
zerombr
text

reboot

%post --erroronfail
nodectl init
%end
```

- **liveimg** is the shared location of the **squashfs.img** image; **http** dictates that the file is shared using **HTTP**.
- **clearpart --all** clears all partitions on the server.
- **autopart --type=thinp** automatically creates new partitions using thin provisioning.
- **rootpw --plaintext redhat** passes the root password in plain text.
- **timezone Etc/UTC --isUtc** sets the timezone.
- **zerombr** clears the MBR.
- **text** indicates that the installation will be performed in text mode.
- **reboot** will reboot the server once the **squashfs.img** is extracted and the installation is complete.

- **nodectl init** initializes the configuration of the server.

5.2. Save the file and close your text editor.

5.3. Log out of the **utility.lab.example.com** server.

```
[root@utility ~]# logout
Connection to utility closed.
[student@workstation ~]$
```

6. The **Kickstart** configuration file can now be used to install a new **RHV** host by adding the **inst.ks** command during the boot process.

However, we are not actually doing an installation during this guided exercise.

```
vmlinuz initrd=initrd.img ip=dhcp inst.stage2=http://content.example.com/rhv4.3/
x86_64/rhvh/dvd inst.ks=nfs://utility.lab.example.com:/ks/rhvh.cfg
```

Finish

On **workstation**, run the **lab hypervisor-kickstart finish** script to complete this exercise.

```
[student@workstation ~]$ lab hypervisor-kickstart finish
```

This concludes the guided exercise.

Lab

Scaling RHV Infrastructure

Performance Checklist

In this lab, you will remove **hostc.lab.example.com** from the **Default** cluster, and then add it to **clustertwo**.

Outcomes

You should be able to remove a host from a cluster, add it to a new cluster, and locate system events in the **Events** log.

Before You Begin

Log in to **workstation** as the **student** user, using **student** as the password.

On **workstation**, run the **lab hypervisor-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab hypervisor-review start
```

1. Ensure that all virtual machines hosted by **hostc.lab.example.com** are powered off.
2. Put **hostc.lab.example.com** into **Maintenance** mode.
3. Remove **hostc.lab.example.com** from the **Default** cluster. Locate the system event messages in **Events**.
4. Add **hostc.lab.example.com** into **clustertwo**.
5. Click **Events** on the menu bar to locate the system event messages. Notice the events for **hostc.lab.example.com**.

Evaluation

On **workstation**, run the **lab hypervisor-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab hypervisor-review grade
```

Finish

On **workstation**, run the **lab hypervisor-review finish** script to complete this exercise.

```
[student@workstation ~]$ lab hypervisor-review finish
```

This concludes the lab.

Solution

Scaling RHV Infrastructure

Performance Checklist

In this lab, you will remove **`hostc.lab.example.com`** from the **`Default`** cluster, and then add it to **`clustertwo`**.

Outcomes

You should be able to remove a host from a cluster, add it to a new cluster, and locate system events in the **`Events`** log.

Before You Begin

Log in to **`workstation`** as the **`student`** user, using **`student`** as the password.

On **`workstation`**, run the **`lab hypervisor-review start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab hypervisor-review start
```

1. Ensure that all virtual machines hosted by **`hostc.lab.example.com`** are powered off.

1.1. Log in to **`workstation`** as **`student`**, using **`student`** as the password.
Open Firefox and log in to the **Administration Portal** using the `https://rhvm.lab.example.com` URL. The username is **`admin`**, the domain is **`internal`**, and the password is **`redhat`**.

1.2. Click on **Compute**, and then click on **Virtual Machines**.

1.3. Look at the **Host** column for each virtual machine. If the column has a value of **`hostc.lab.example.com`**, right-click the name of the virtual machine and select **`Shutdown`**. Click on **Confirm** to shut down the virtual machine.

> **Important**
> Do not attempt to switch off the **`HostedEngine`** virtual machine. This virtual machine is the self-hosted RHV-M installation.

2. Put **`hostc.lab.example.com`** into **`Maintenance`** mode.

2.1. Click on **Compute**, and then click on **Hosts**.

2.2. Click on **`hostc.lab.example.com`**.

2.3. From the **Management** drop-down menu, select **Maintenance** to put **`hostc.lab.example.com`** into maintenance mode. This opens the **Maintenace Host(s)** window. In the **Reason** box, type **`removing hostc.lab.example.com from the cluster`**. Click on **OK**.

2.4. Confirm that the **Status** of **`hostc.lab.example.com`** is **Maintenance**.

3. Remove **`hostc.lab.example.com`** from the **`Default`** cluster. Locate the system event messages in **`Events`**.

3.1. Click on **`hostc.lab.example.com`**. Click on the **Remove** button. This opens the **Remove Host(s)** window. In the **Reason** box, type **`removing hostc.lab.example.com from the default cluster`**. Click on **OK**.

3.2. Click on **Events**. Notice the events for **`hostc.lab.example.com`**.

The events below are listed in the order in which they occur. In the **`Administration Portal`**, the event order may be selected by clicking the **Time** column heading in the **Events** table.

- Ansible host-remove playbook execution started on host hostc.lab.example.com.
- Ansible host-remove playbook execution has successfully finished on host hostc.lab.example.com. For more details check log /var/log/ovirt-engine/ansible/ansible-20190722023326-ovirt-host-remove_yml.log
- Host hostc.lab.example.com was removed by admin@internal-authz.

> **Note**
> It can take several minutes for all the system event messages to appear.

4. Add **`hostc.lab.example.com`** into **`clustertwo`**.

4.1. Click on **Compute**, and then click on **Hosts**.

4.2. Click on the **New** button. This opens the **New Host** window.

4.3. In the **New Host** window, enter the following configuration settings:

- Choose **`clustertwo`** as the Host Cluster.
- Enter **`hostc.lab.example.com`** in the **Name** field.
- Enter **`hostc.lab.example.com`** in the **Hostname** field.
- Enter **`redhat`** in the **Password** field.

Click the **OK** button to add **`hostc.lab.example.com`** to the cluster.

4.4. Click **OK** in the **Power Management Configuration** window.

The **Status** of **`hostc.lab.example.com`** is **`Installing`**. It takes several minutes for the **Status** to change to **Up**.

5. Click **Events** on the menu bar to locate the system event messages. Notice the events for **`hostc.lab.example.com`**.

The events below are listed in the order in which they occur. In the **`Administration Portal`**, the event order may be selected by clicking the **Time** column heading in the **Events** table.

- Ansible host-deploy playbook execution has successfully finished on host hostc.lab.example.com.
- Host hostc.lab.example.com installed.
- No faulty multipath paths on host hostc.lab.example.com.
- Status of host hostc.lab.example.com was set to Up.

Evaluation

On **workstation**, run the **lab hypervisor-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab hypervisor-review grade
```

Finish

On **workstation**, run the **lab hypervisor-review finish** script to complete this exercise.

```
[student@workstation ~]$ lab hypervisor-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- Red Hat Virtualization Hosts can be installed automatically using Kickstart.
- Kickstart can be used, in conjunction with PXE and TFTP, to start the installation by booting from the network, which allows quick and fully unattended, automatic installation of new RHV-H hosts.
- Many major changes to RHV hosts and data centers requires the hosts or data centers to be placed into maintenance mode. This mode allows you to make permanent changes to any resource.
- When the RHV-H host is the last host left in a data center, the data center must be placed in maintenance mode before the host can be placed into maintenance mode.
- Satellite Server can be used as a provisioning server. Using Satellite Server allows administrators to provision grouped machines, using HostGroups. Satellite Server also provides ongoing support for upgrades. Enterprise grade provisioning and management should always be done using Satellite Server.

Chapter 6

Managing RHV Networks

Goal	Separate network traffic into multiple networks on one or more interfaces to improve the performance and security of Red Hat Virtualization.
Objectives	• Create logical networks to segregate traffic in a data center. • Configure hosts to use available logical networks. • Configure RHV to use networks provided by an external OpenStack provider.
Sections	• Managing Logical Networks (and Guided Exercise) • Configuring Hosts to Use Logical Networks (and Guided Exercise) • Configuring External Network Providers (and Quiz)
Lab	Managing RHV Networks

Managing Logical Networks

Objectives

After completing this section, you should be able to create logical networks to segregate traffic in a data center.

Networking in Red Hat Virtualization

Network configuration is one of the most important factors influencing the performance of your virtualization environment. Networking in Red Hat Virtualization is defined in several layers. The underlying physical networking infrastructure must exist and be configured for connectivity between hardware and the logical components of the RHV environment.

You create *logical networks* to segregate different types of network traffic onto separate VLANs or physical networks for improved security and performance. For example, separate VLANs can be used for management traffic, storage traffic, or to isolate the traffic of a set of virtual machines.

Logical Networks

Logical networks are defined in a data center and assigned to one or more clusters. A logical network can be assigned to multiple clusters in a data center, to provide communication between VMs in different clusters.

New logical networks are defined with a unique name, the data center in which it resides, and whether it will be used for workload (VM network) traffic. Logical networks require setting a unique VLAN tag (VLAN ID) if this virtual network will share access with any other virtual network on a RHV host physical NIC. If needed, you can also configure Quality of Service (QoS) and bandwidth limiting settings for the logical network.

Designating VM Networks

Logical networks are either designated as *VM network* or else they are regarded as *infrastructure networks*. A VM network is a logical network that will connect to virtual network interface cards (vNICs) to carry virtual machine application traffic. A software-defined Linux bridge is created, per logical network, on the RHV host to provide connectivity between the host's physical NIC and the virtual machine vNICs configured to use that logical network. Linux bridge performance is proportional to that of the host's physical NIC, and limited by any QoS settings.

A data center and its clusters can have multiple VM networks, for traffic management and security reasons. When virtual machines are created, each vNIC is assigned to a VM network.

All other logical networks are regarded as infrastructure networks for communication between RHV-M and RHV host only, and are not connected to virtual machines. Because infrastructure networks do not connect to VMs they do not need an associated Linux bridge on RHV hosts.

Infrastructure networks must be configured at the cluster level to indicate what type of traffic it carries. Each host in the cluster also must have the host's correct physical network interface configured for that network. The logical network types are discussed next.

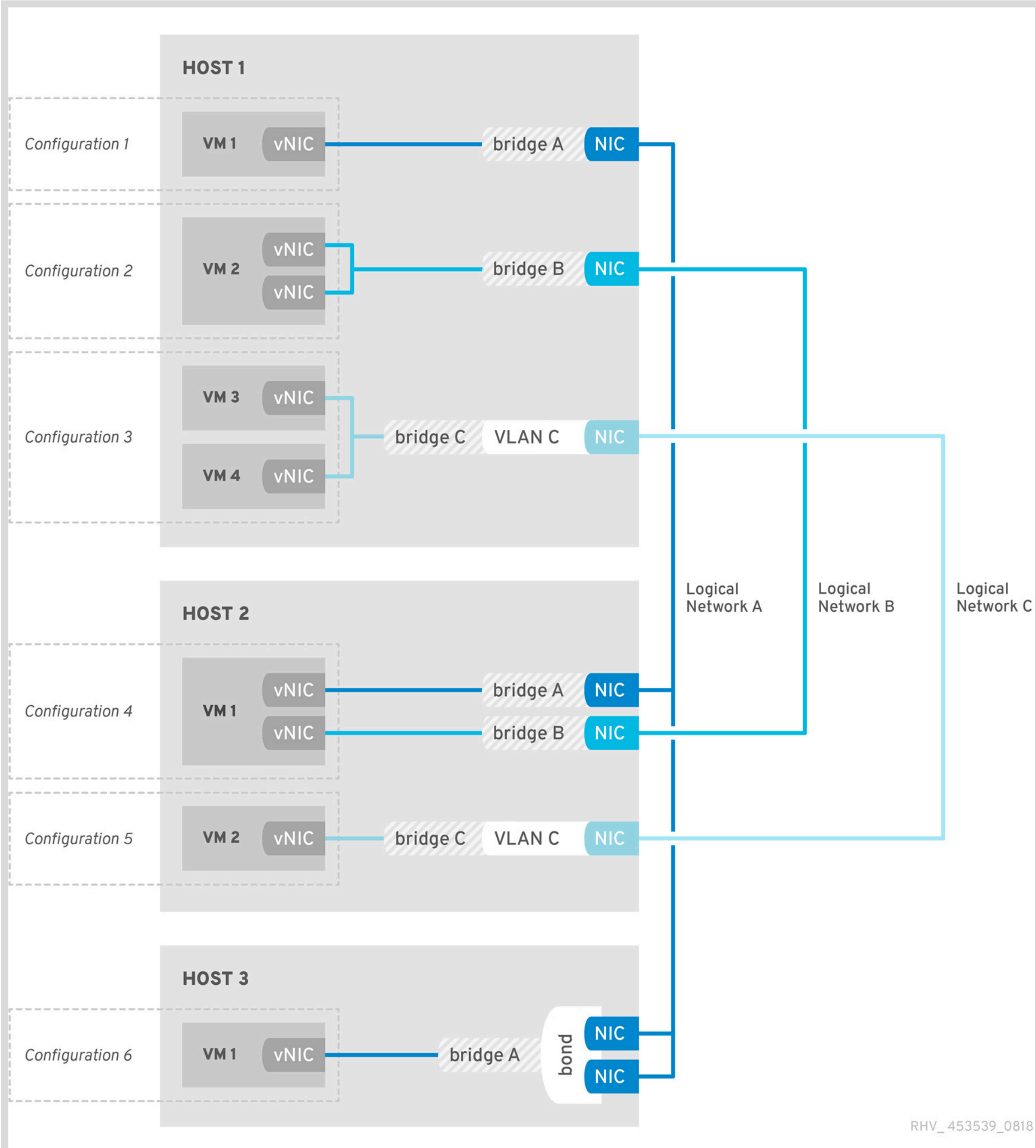

Figure 6.1: RHV logical VM networking

Types of Logical Networks

Logical networks can be configured to segregate different traffic types on different logical networks. The initial, single default logical network, called **`ovirtmgmt`**, is configured as both a VM network and also to handle all infrastructure traffic. By default, **`ovirtmgmt`** is used for management, display and migration network traffic, in addition to VM traffic. Although this a functional configuration, it provides no boundaries between different network traffic types.

Red Hat recommends that you plan and create additional logical networks to segregate traffic. The following lists provides examples of practical traffic segregation.

Segregating Network Traffic by Types

Management
: This network role facilitates VDSM communication between the RHV-M and the RHV hosts. By default, it is created during the RHV-M engine deployment and named **`ovirtmgmt`**. It is the only logical network created automatically; all others are created according to environment requirements.

Display
: This network role is assigned to a network to carry the virtual machine display (SPICE or VNC) traffic from the Administration or VM Portal to the host running the VM. The RHV host then accesses the VM console using internal services. Display networks are not connected to virtual machine vNICs.

VM network
: Any logical network designated as a VM network carries network traffic relevant to the virtual machine network. This network is used for traffic created by VM applications and connects to VM vNICs. If applications require public access, this network must be configured to access appropriate routing and the public gateway.

Storage
: A storage network provides private access for storage traffic from RHV hosts to storage servers. Multiple storage networks can be created to further segregate file system based (NFS or POSIX) from block based (iSCSI or FCoE) traffic, to allow different performance tuning for each type. Jumbo Frames are commonly configured on storage networks. Storage networks are not a network role, but are configured to isolated storage traffic to separate VLANs or physical NICs for performance tuning and QoS. Storage networks are not connected to virtual machine vNICs.

Migration
: This network role is assigned to handle virtual machines migration traffic between RHV hosts. Assigning a dedicated non-routed migration network ensures that the management network does not lose connection to hypervisors during network-saturating VM migrations.

Gluster
: This network role is assigned to provide traffic from Red Hat Gluster Servers to GlusterFS storage clusters.

Fencing
: Although not a network role, creating a network for isolating fencing requests ensure that this critical requests are not missed. RHV-M does not perform host fencing itself but sends fence requests to the appropriate host to execute the fencing command.

Required and Optional Networks

When created, logical networks may be designated as *Required* at the cluster level. By default, new logical networks are added to clusters as required networks. Required networks must be connected to every host in the cluster, and are expected to always be operational.

When a required network becomes nonoperational for a host, that host's virtual machines are migrated to another cluster host, as specified by the current cluster migration policy. Mission-critical workloads should be configured to use required networks.

Logical networks that are not designated as required are regarded as optional. Optional networks may be implemented only on the hosts that will use them. The presence or absence of optional networks does not affect the host's operational status.

When an optional network becomes nonoperational for a host, that host's virtual machines that were using that network are *not* migrated to another host. This prevents unnecessary overhead caused by multiple, simultaneous migrations for noncritical network outages. However, a virtual machine with a vNIC configured for an optional VM network will not start on a host that does not have that network available.

Logical Network Configuration at RHV Logical Layers

Logical network configuration occurs at each layer of the RHV environment.

Data Center Layer
: Logical networks are defined at the data center level. Each data center has the ovirtmgmt management network by default. Additional logical networks are optional but recommended. VM network designation and a custom MTU are set at the data center level. A logical network defined for a data center must be added to the clusters that use the logical network.

Cluster Layer
: Logical networks are available from the data center, and added to clusters that will use them. Each cluster is connected to the management network by default. You can add any logical networks to a cluster if they are defined for the parent data center. When a required logical network is added to a cluster, it must be implemented on each cluster host. Optional logical networks can be added to hosts as needed.

Host Layer
: Virtual machine logical networks are connected to each host in a cluster and implemented as a Linux bridge device associated with a physical network interface. Infrastructure networks do not implement Linux bridges but are directly associated with host physical NICs. When first added to a cluster, each host has a management network automatically implemented as a bridge on one of its NICs. All required networks in a cluster must be associated with a NIC on each cluster host to become operational for the cluster.

Virtual Machine Layer
: Logical networks that are available for a host are available to attach to a virtual machine NIC on that host. The virtual machine then gains access to other systems and destinations available on the logical network through the connected vNIC.

Performance Considerations

Gigabit Ethernet is sufficient for the management network, and is typically sufficient for the display network. Any migration and storage networks you add will perform better as dedicated high-bandwidth networks or VLANs. Use 10 GbE or 40 GbE infrastructure when available. Smaller networks can be aggregated larger throughput by using network bonding or teaming. Bandwidth requirements for VM Networks must be calculated from your application requirements.

Red Hat recommends using larger networks integrated with VLANs and advanced QoS features to more easily manage host infrastructure and networking performance of the RHV environment.

Creating Logical Networks

Create logical networks in the **Networks** page under the **Compute** menu. Click **New** to open a **New Logical Network** window. Select the data data center for the network and assign a unique name. You need to know if the network will using a VLAN tag, which should already have been planned across the RHV environment. Click **Enable VLAN tagging** and enter the VLAN ID number assigned. If this network is to be use for virtual machine traffic, select **VM network**. Click on **OK**, to attach the new network to all data center clusters as a required network.

To select a cluster, click on **Compute** then **Clusters**. Highlight a cluster in the list and click on it. In the upper pane that displays, select the **Logical Networks** tab.

Using the **Manage Networks** button, access the **Manage Networks** window. Using the available check boxes, specify which network will carry each type of infrastructure traffic. You can also assign or unassign networks to the cluster, and designate whether each is required or optional.

Finally, the logical network must be attached to a physical network interface on one or more hosts in the cluster. This topic is covered in the next section of this chapter.

References

Further information is available in the Logical Networks chapter of the *Administration Guide* for Red Hat Virtualization; at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#chap-Logical_Networks

Further information is available in the Network chapter of the *Technical Reference* for Red Hat Virtualization; at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/technical_reference/index#chap-Network

Guided Exercise

Managing Logical Networks

In this exercise, you will create logical networks for one of your clusters that will separate RHV management traffic from storage and VM traffic.

Outcomes

You should be able to:

- Create a logical network named **vm-net** and tag it as VLAN 10, for VM traffic.
- Create an untagged logical network for storage traffic named **storage-net**.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab manage-networks start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab manage-networks start
```

1. Log in to **workstation** as **student** user with **student** as password. Open Firefox and log in to the Administration Portal, `https://rhvm.lab.example.com`, as the **admin** user with the **internal** profile, using **redhat** as password.

2. Separate the management network traffic from virtual machine network traffic by creating a new logical network called **vm-net** for the virtual machine traffic. This network will be on VLAN 10.

   2.1. In the menu click on **Network**, then select **Networks**. Click the **New** button to create a new network. In the **New Logical Network** dialog window, fill in the fields using the following information:

Field	Value
Data Center	`development`
Name	`vm-net`
Description	`VM Network`
Comment	`Network for VM Traffic`
Enable VLAN Tagging	`Enabled`
In the text field next to the **Enable VLAN tagging** check box, enter the VLAN number:	`10`
VM network	`Enabled`

2.2. Leave the other options with their default values and click the **OK** button to create the network.

2.3. Notice that the new network appeared on the list of available networks. This confirms the creation of the new logical network.

3. Create another logical network, called **`storage-net`**, to separate the storage traffic from other networks. This logical network does not use VLAN tagging, but will eventually be assigned its own dedicated physical interface on each host in the cluster.

3.1. Click the **New** button to create the new logical network.

3.2. In the **New Logical Network** dialog window, complete the fields as follows:

Field	Value
Data Center	`development`
Name	`storage-net`
Description	`Storage Network`
Comment	`Network for storage traffic`
Enable VLAN Tagging	`Disabled`
VM network	`Disabled`

3.3. Leave the other options with their default values and click the **OK** button to create the network.

3.4. Notice that the new network displays on the list of available networks. This confirms the creation of the new logical network.

Finish

On **`workstation`**, run the **`lab manage-networks finish`** script to complete this exercise.

```
[student@workstation ~]$ lab manage-networks finish
```

This concludes the guided exercise.

Configuring Hosts to Use Logical Networks

Objectives

After completing this section, you should be able to configure hosts to use available logical networks.

Adding Logical Networks to RHV-H Hosts

In the previous section, you learned how to create logical networks to separate different types of network traffic. This section describes the procedures needed to implement the logical networks on cluster hosts. Use caution when performing this type of reconfiguration of the RHV environment, as a misstep might make your RHV-H host nonoperational.

Assigning Logical Networks

When a logical network is created, it is automatically attached to all clusters in the data center, unless you specify otherwise. If a cluster has at least one associated RHV-H host, the network's state is marked **Non Operational** until the logical network is attached to a physical interface on each cluster host. This is because the RHV-H host network configuration must be modified to attach the logical network to a physical network interface. When the attachment is complete, the logical network state becomes **Operational**.

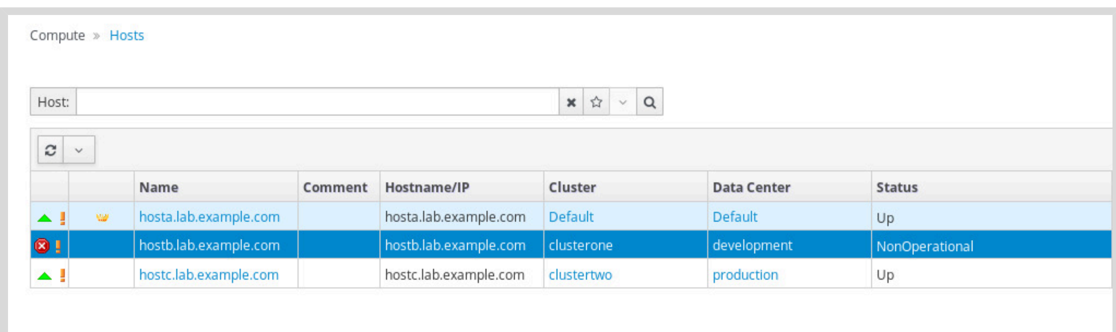

Figure 6.2: Non Operational network state

After creating a new data center logical network, assign it to network interfaces on cluster hosts. For a required logical network to become active for a cluster, you must assign it to every RHV-H host in the cluster.

Optional networks, however, become operational immediately. A cluster host may use an optional logical network only when that network is associated with a network interface on that host.

To assign a logical network to an RHV-H host:

From the **Hosts** page, click the host name to which you will attach a network. On the upper menu bar, click the **Network Interfaces** tab to list the host's NICs. Click the **Setup Host Networks** button to open the **Setup Host Networks** window. Dragging a logical network listed under **Unassigned Logical Networks** and dropping it on a specific interface row will assign that network to the chosen interface.

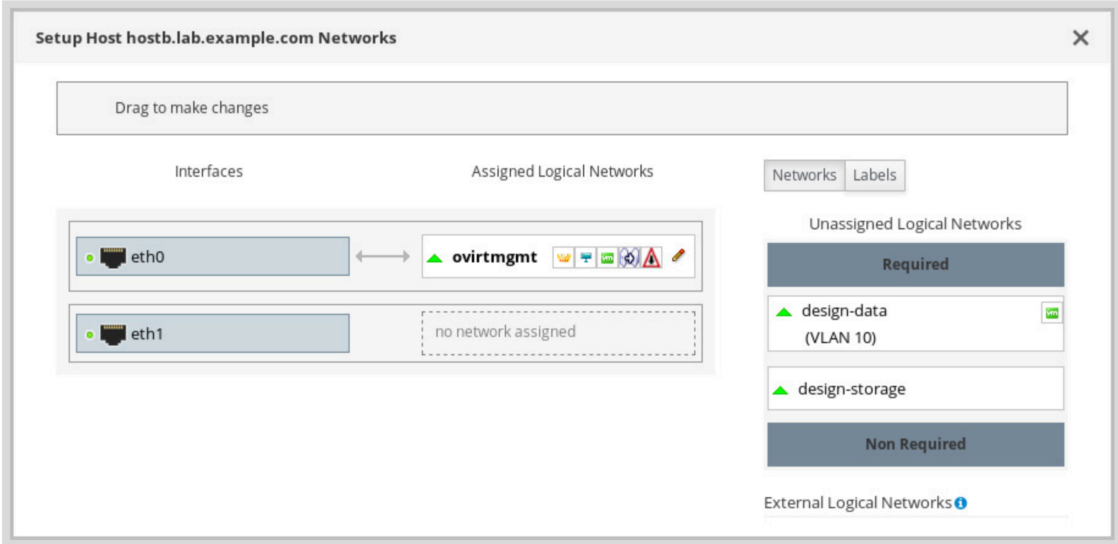

Figure 6.3: Assigning a network to a host NIC

While still in the **Setup Host Networks** window, click the **pencil** icon at the right of a row of icons in a logical network box to open the **Edit Management Network** window for setting network parameters.

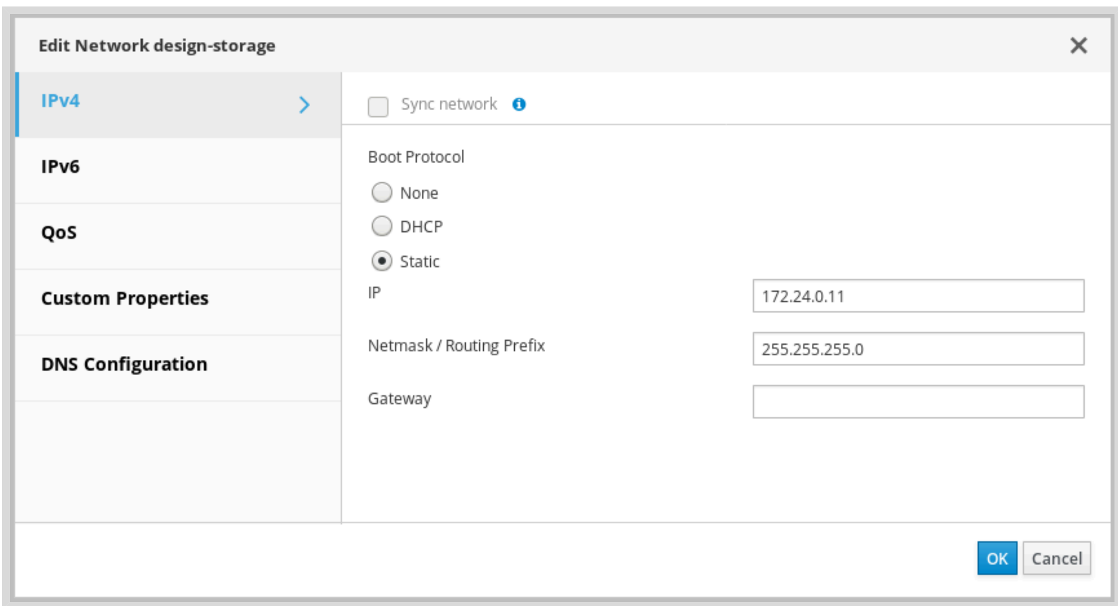

Figure 6.4: Viewing or Editing a Logical Network's Parameters

The window displays the network name. You can modify the boot protocol, and the IP address, netmask, and gateway when using static addressing. View the optional tabs for this network's IPv6, QoS, and DNS configuration.

Important

Occasionally, altered network configuration might not be saved or honored on the hosts. This occurs when the RHV-M configuration and the host configuration have become out of sync. To correct this, force the synchronization of the current network configuration onto the host. To perform the host configuration synchronization, click the **Sync All Networks** button.

Automating Attachment Using Labels

In a large RHV environment with multiple RHV-H hosts, it can be difficult to manually assign logical networks to host interfaces when the network design is changed. To help, *Network labels* can make assignments of logical networks to host vNICs simpler, by having RHV automatically assign new logical networks to hosts, or existing logical networks to new hosts, by matching up the labels assigned to each.

A network label is an arbitrary text string (using only upper and lowercase, underscores, or hyphens) that is set on a logical network, or a host physical interface. If cluster hosts have assigned labels on their interfaces, and a new logical network with a same label is added to the cluster, the logical network is automatically assigned to the matching network interfaces. Setting a specific label on a new logical network, for example, causes an automatic environment-wide attachment of that network on all hosts. Conversely, removing a label from a logical network removes that logical network from all hosts with that label.

Multiple Logical Networks on a Labeled Host NIC

Two or more logical networks can have the same label, if they are to be assigned to the same host physical NICs. Attaching two or more logical networks to the same NIC requires the use of unique VLAN IDs on each logical network. For example, you could add the **`internal`** label to one network interface on each host. Then, create two logical networks (VLAN tag 10 one and VLAN tag 20 on the other) and use the **`internal`** label on both. When you click the **OK** button to complete each logical network, each will automatically be associated with the same labeled network interface on every host.

Use the **Setup Host Networks** page to assign labels. Above the **Unassigned Logical Networks** is the **Labels** button. Dragging the ***`New_Label`*** to a desired interface adds the label name.

> **Note**
>
> This section covered how to create attach logical networks to a host network interface. Attaching VM networks to virtual machines is covered in an upcoming chapter that discusses virtual machine deployment.

> **References**
>
> Further information is available in the Hosts and Networking chapter of the *Administration Guide* for Red Hat Virtualization at
> https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#sect-Virtual_Network_Interface_Cards

Guided Exercise

Configuring Hosts to Use Logical Networks

In this exercise, you will configure hosts to use logical networks to separate different types of traffic in a cluster.

Outcomes

You should be able to connect the new logical networks from the preceding exercise to RHV-H hosts in the **clusterone** cluster:

- **vm-net** (on VLAN 10) is on the host's **eth0** network interface for each host.
- **storage-net** is on the host's **eth1** network interface for each host.

Before You Begin

Log in to **workstation** as **student**, using **student** as the password.

On **workstation**, run the **lab networks-host start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab networks-host start
```

1. Log in to **workstation** and open Firefox. Using the https://rhvm.lab.example.com URL, go to the RHV-M web interface. Click on the **Administration Portal** link and log in to the web interface as the **admin** user with the **internal** profile, using **redhat** as password.

2. Attach the existing **storage-net** and **vm-net** logical networks to the **hostb** host. The logical network **vm-net** should be on **eth0**, and using DHCP to get an address. The logical network **storage-net** should be on **eth1**, and should statically set its IPv4 address on this host to 172.24.0.11 with a 255.255.255.0 netmask.

   2.1. Click the **Compute** on the menu and select **Hosts**.

   2.2. From the list of available hosts, click on **hostb.lab.example.com** to select it.

   2.3. In the upper part of the page, click on the **Network Interfaces** tab.

   2.4. Click the **Setup Host Networks** button to change the host network configuration.

   2.5. In the **Setup Host hostb.lab.example.com Networks** dialog window, click and drag the **storage-net** box from the right side to the left side of the window. Drop that box onto the **no network assigned** field, next to the **eth1** network interface.

   2.6. Click and drag the **vm-net (VLAN 10)** box from the right side to the left side of the window. Drop that box onto the **eth0** interface field. After dropping, you should see two logical networks assigned to **eth0** interface.

2.7. Click on the **pencil** icon inside the **storage-net** box. In the **Edit Network storage-net** dialog window, under **Boot Protocol**, click the radio button next to **Static** label.

2.8. In the **IP** field, type **172.24.0.11** as the IP address of **hostb** on the network.

2.9. In the **Netmask/Routing Prefix** field, type **255.255.255.0** as the netmask. Leave the **Gateway** field empty.

2.10. Click **OK** to save the settings.

2.11. Ensure that **Verify connectivity between Host and Engine** and **Save network configuration** options are selected.

2.12. Click the **OK** button to confirm and save the new network configuration for that host.

Finish

On **workstation**, run the **lab networks-host finish** script to complete this exercise.

```
[student@workstation ~]$ lab networks-host finish
```

This concludes the guided exercise.

Configuring External Network Providers

Objectives

After completing this section, you should be able to configure RHV to use networks provided by an external OpenStack provider.

Configuring External Network Providers

RHV-M can utilize networks from any external network provider that implements the OpenStack Neutron REST API, including the Red Hat OpenStack Platform (RHOSP) OpenStack Networking service. The OpenStack Networking component provides an API for software-defined networking (SDN) capabilities, including dynamic creation and management of switches, routers, firewalls, and external connections to physical networks.

Neutron includes an expanding list of plugins, drivers, and agents that enable coupling with many commercial and open-source network technologies, including Cisco virtual and physical switches, NEC OpenFlow, Open vSwitch and Open Virtual Networking, Linux bridging, VMware NSX, MidoNet, OpenContrail, Open Daylight, Brocade, Juniper, and IBM. RHV 4.3 supports RHOSP versions 10, 13, and 14 as external network providers, when deployed with the original Open vSwitch driver up to and including version 13, and the Open Virtual Network and Open Daylight drivers starting in version 13.

Reviewing Software Defined Networking

Software-defined networking is more than deploying virtual networking components in a virtualization or cloud environment. An SDN controller is the control plane component that manages network devices in the data (forwarding) plane. These network devices, such as switches, routers, and firewalls, are programmatically configured for network routes, security, subnets and bandwidth in cooperation with the cloud-native application requiring dynamic services and allocation. An SDN controller centralizes the network global view, and presents the perception of a massively scalable, logical network switch to those applications.

Open vSwitch (OVS) can plug and unplugs port, create networks or subnets, and provide IP addressing. An Open vSwitch bridge allocates virtual ports to instances, and can span across to the physical network for incoming and outgoing traffic. Implementation is provided by OpenFlow (OF), which defines the communication protocol that enables the SDN Controller to act as the middle manager with both physical and virtual networking components, passing information to switches and routers below, and the applications and business logic above.

Enhanced features of OVN

Open Virtual Networking (OVN) enhanced OVS significantly to add native support for virtual network abstractions, such as virtual L2 and L3 overlays and security groups. Some high level features of OVN include:

- Provides virtual networking abstraction for OVS, implemented using L2 and L3 overlays, but can also manage connectivity to physical networks.
- Supports flexible security policies implemented using flows that use OVS connection tracking.
- Native support for distributed L3 routing using OVS flows, with support for both IPv4 and IPv6.
- Native support for NAT and load balancing using OVS connection tracking.

- Native fully distributed support for DHCP.
- Works with any OVS datapath (such as the default Linux kernel datapath, DPDK, or Hyper-V) that support Geneve tunnels and OVS connection tracking.
- Supports L3 gateways from logical to physical networks.
- Supports software-based L2 gateways.
- Can provide networking for both VMs and containers running inside of those VMs, without a second layer of overlay networking.

Inclusion of OVN significantly enhances RHV networking, by providing a native overlay networking solution, and shifting networking away from being handled only by Linux host networking. OVN enhances networking and provides management capabilities to address diverse or complex networking requirements for advanced use cases.

Importing External Networks

To use neutron networks, first register the external network provider with RHV-M. After the network provider is validated, networks offered by that provider are automatically discovered and listed for selection as logical networks that are available to import into a data center. Imported networks are treated as RHV logical networks and can be attached to virtual machines in the data center. Certain limitations apply to using logical networks imported from an external provider:

- External provider networks are for use as virtual machine networks, and cannot be used as display networks.
- The same logical network can be imported more than once, but only to different data centers.
- External provider networks may not be edited in RHV, but must be managed in the source external provider who implemented the drivers, agents and interfaces of the network.
- External provides cannot be deleted from from RHV-M while any of their logical networks remain in use by a virtual machine.
- External provider networks are not required. Although virtual machine scheduling does not check availability for imported networks during host selection, imported networks are acceptable for all normal deployment use cases.

Configuring Hosts to use External Networks

RHV hosts must communicate with the neutron API to use imported networks, requiring the installation and configuration of a neutron agent (the provider's virtual interface driver) using one of two methods. The second method, using Red Hat OpenStack Platform director, is recommended. For the first method, install the neutron agents manually. Register each host to the following repositories:

- rhel-7-server-rpms
- rhel-7-server-rhv-4-mgmt-agent-rpms
- rhel-7-server-ansible-2-rpms

Then install the OpenStack Networking hook, by which VDSM invokes the correct plugin type (such as OVS) to pass the correct information to libvirt to treat vNICs that are handled by the OpenStack Networking provider. Also, remove the firewall rule that rejects ICMP traffic:

```
[root@host ~]# yum update
[root@host ~]# yum install vdsm-hook-openstacknet
[root@host ~]# iptables -D INPUT -j REJECT --reject-with icmp-host-prohibited
```

The second method is to use Red Hat OpenStack Platform director to deploy the Networker role to a node, then add the network node to the RHV Manager as a new host. Using the RHOSP director is recommended.

Configuring Subnets on External Networks

External network providers assign IP addresses through subnets configured on the logical networks. On networks with more than one available subnet, virtual machines can be assigned an IP address from any subnet. The external network provider's DHCP service assigns these IP addresses. Although RHV-M can discover predefined subnets on imported logical networks, subnets can be removed in RHV, without changing the external provider's configuration, to avoid using that subnet in RHV. RHV can also add new subnets for RHV use. To add or remove subnets from the networks on the external provider, use that provider's configuration tools.

Integrating Virtualization and Cloud

Neutron network integration between RHV and RHOSP provides the hybrid platform to solve many rightsizing and scaling problems. In early cloud migration plans, many companies attempted to force fit enterprise workloads into the cloud without adequate redesign to handle scale out, stateless, and application-resilient HA (as opposed to platform resilient) requirements. Lessons were learned, in the first decade of the cloud rush, that not all workloads are conducive to redesign as cloud workloads. Hybrid infrastructure is increasingly seen as not just useful, but requisite, because most applications are comprised of a mix of enterprise and microservice workloads.

For example, significant applications remain as virtualized enterprise workloads, including:

- Private cloud control plane, monitoring, and configuration management servers.
- Authentication and authorization, and identity management servers.
- Highly specialized, mission-critical applications. including high performance computing.
- Database, data warehouse and big data business intelligence applications.
- Highly available support servers, including DNS, DHCP, file servers, and email servers.
- Any application requiring service level agreements addressing performance, backups, and stateful workloads.

Benefit of integrating environments

These applications remain on Red Hat Virtualization while middleware and front end interfaces move to the cloud. Organizations now require these hybrid abilities:

- To support applications requiring both scale up and scale out technologies simultaneously for different application components.
- To support software-defined networking in enterprise virtualization and including network overlay, encapsulation, and network security groups.
- To manage a complex network topology from a single management platform for both the virtualization and the private cloud environments.

Neutron integration allows Red Hat Virtualization and Red Hat OpenStack Platform to be deployed in such a manner that applications can be designed to utilize both environments simultaneously. The front end, middleware, and business intelligence are deployed to OpenStack for the ability to scale out and back on demand, while a back end database continues as a highly available workload in Red Hat Virtualization, where it is better suited.

Traditional virtualization has become efficient at deploying and provisioning, minimizing delivery time using templates, snapshots, and images. Until now, enterprise virtualization still required physical switches and cabling for complex network infrastructure between clusters and data centers, thus limiting the increases in deployment speed. With RHOSP deployed alongside RHV in a data center, and utilizing Neutron integration, SDN features and management are shared, allowing automated, on-demand network configuration and scaling. By incorporating SDN into RHV, the legacy virtualization infrastructure now has access to the advanced networking capabilities of VXLAN, OpenFlow, Geneve, and Open vSwitch to supplement traditional VLANs and Linux bridges.

References

For more information, refer to the *External Providers* chapter in *Adding an OpenStack Networking (Neutron) Instance for Network Provisioning* at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#Adding_an_OpenStack_Network_Service_Neutron_for_Network_Provisioning

Quiz

Configuring External Network Providers

Choose the correct answers to the following questions:

1. **How does RHV support Red Hat OpenStack the Platform?**
   a. By showing logging in real time for networks hosted on OpenStack.
   b. By reducing workloads during peak hours.
   c. By utilizing OpenStack Neutron network to integrate both OpenStack and RHV together.
   d. By managing all OpenStack deployments and scaling out when needed.

2. **What are two limitations to using External Provider Networks? (Choose two.)**
   a. The ability to support applications that require both scale up and scale out technologies simultaneously for different segments of the application.
   b. Port mirroring is not available for virtual network interface cards connected to logical networks.
   c. Software defined networking (SDN) in virtualization, along with overlay, encapsulation, and security groups is not supported.
   d. It is not possible to customize the name of the network that you are importing.
   e. The same logical network can be imported more than once, but only to different data centers.

3. **What feature does Skydive not support?**
   a. Captures network topology, interface, bridge, namespace attributes, and keeps the history of all the modifications.
   b. Provides only one static binary, and is distributed, scalable, and easy to deploy.
   c. Reviews all your applications and reports on the processes that are running.
   d. Real-time monitoring of the network infrastructure.

Solution

Configuring External Network Providers

Choose the correct answers to the following questions:

1. How does RHV support Red Hat OpenStack the Platform?

a. By showing logging in real time for networks hosted on OpenStack.
b. By reducing workloads during peak hours.
c. By utilizing OpenStack Neutron network to integrate both OpenStack and RHV together.
d. By managing all OpenStack deployments and scaling out when needed.

2. What are two limitations to using External Provider Networks? (Choose two.)

a. The ability to support applications that require both scale up and scale out technologies simultaneously for different segments of the application.
b. Port mirroring is not available for virtual network interface cards connected to logical networks.
c. Software defined networking (SDN) in virtualization, along with overlay, encapsulation, and security groups is not supported.
d. It is not possible to customize the name of the network that you are importing.
e. The same logical network can be imported more than once, but only to different data centers.

3. What feature does Skydive not support?

a. Captures network topology, interface, bridge, namespace attributes, and keeps the history of all the modifications.
b. Provides only one static binary, and is distributed, scalable, and easy to deploy.
c. Reviews all your applications and reports on the processes that are running.
d. Real-time monitoring of the network infrastructure.

Lab

Managing RHV Networks

Performance Checklist

In this lab, you will create additional logical networks for a new cluster, and configure hosts in the cluster to use the new networks.

Outcomes

You should be able to create additional logical networks and associate those networks with appropriate hosts and clusters.

Before You Begin

Log in to **workstation** as **student**, using **student** as the password.

On **workstation**, run the **lab networks-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab networks-review start
```

1. You have a working **production** data center, which has a cluster named **clustertwo**. That cluster consists of the host **hostc.lab.example.com**.

   In the **production** data center, create a new logical network called **prod-vm-net** on VLAN 20 as a VM Network to separate virtual machine traffic from RHV management traffic. Assign this logical network to **clustertwo**.

2. In the same data center, create another logical network called **prod-storage**. This network should not be VLAN tagged. The new logical network will be used to separate storage traffic from other traffic. Assign the logical network to **clustertwo**.

3. Attach the logical networks to network interfaces on **hostc**.

   Attach the **prod-vm-net** logical network to the **eth0** network interface. The network configuration comes from the Dynamic Host Configuration Protocol (DHCP) server.

   Attach the **prod-storage** logical network to the **eth1** network interface. Set the IPv4 address to 172.24.0.12, with the 255.255.255.0 netmask. Do not set a gateway for this logical network.

Evaluation

On **workstation**, run the **lab networks-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab networks-review grade
```

Finish

On **`workstation`**, run the **`lab networks-review finish`** script to complete this exercise.

```
[student@workstation ~]$ lab networks-review finish
```

This concludes the lab.

Solution

Managing RHV Networks

Performance Checklist

In this lab, you will create additional logical networks for a new cluster, and configure hosts in the cluster to use the new networks.

Outcomes

You should be able to create additional logical networks and associate those networks with appropriate hosts and clusters.

Before You Begin

Log in to **workstation** as **student**, using **student** as the password.

On **workstation**, run the **lab networks-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab networks-review start
```

1. You have a working **production** data center, which has a cluster named **clustertwo**. That cluster consists of the host **hostc.lab.example.com**.

   In the **production** data center, create a new logical network called **prod-vm-net** on VLAN 20 as a VM Network to separate virtual machine traffic from RHV management traffic. Assign this logical network to **clustertwo**.

   1.1. Log in to **workstation** as **student** user, using **student** as password. Open Firefox and log in to the **Administration Portal**, using the `https://rhvm.lab.example.com` URL. Use the **admin** user with the **internal** profile, and **redhat** as the password.

   1.2. In the **Administration Portal** dashboard, click **Network**, and then click **Networks** to access the network configuration for the data centers. Click the **New** button to create a new network. In the **New Logical Network** window, complete in the fields using the following information:

Field	Value
Data Center	**`production`**
Name	**`prod-vm-net`**
Description	**`VM LAB network`**
Comment	**`Network for VM LAB traffic`**
Enable VLAN Tagging	**`Enabled`**
In the text field next to **Enable VLAN tagging**, specify the VLAN number:	**`20`**
VM Network	**`Enabled`**

1.3. Leave the other options with their default values and click the **OK** button to create the network.

1.4. Notice that the new network displays on the list of available networks. This confirms the creation of the new logical network.

2. In the same data center, create another logical network called **`prod-storage`**. This network should not be VLAN tagged. The new logical network will be used to separate storage traffic from other traffic. Assign the logical network to **`clustertwo`**.

2.1. Click the **New** button to create the new logical network.

2.2. In the **New Logical Network** window, fill in the fields using the following information:

Field	Value
Data Center	**`production`**
Name	**`prod-storage`**
Description	**`Storage Network LAB`**
Comment	**`Network for LAB storage traffic`**
Enable VLAN Tagging	**`Disabled`**
VM network	**`Disabled`**

2.3. Leave the other options with their default values and click the **OK** button to create the network.

2.4. Notice that the new network appeared on the list of available networks. This confirms the creation of the new logical network.

3. Attach the logical networks to network interfaces on **`hostc`**.

Attach the **`prod-vm-net`** logical network to the **`eth0`** network interface. The network configuration comes from the Dynamic Host Configuration Protocol (DHCP) server.

Attach the **prod-storage** logical network to the **eth1** network interface. Set the IPv4 address to 172.24.0.12, with the 255.255.255.0 netmask. Do not set a gateway for this logical network.

3.1. Click the **Compute** then Hosts menu.

3.2. From the list of available hosts, choose **hostc.lab.example.com** by clicking on it.

3.3. Click on the **Network Interfaces** tab.

3.4. Click the **Setup Host Networks** button to change that host network configuration.

3.5. In the **Setup Host hostc.lab.example.com Networks** window, click and drag the **prod-storage** box from the right side to the left side of the window. Drop this box onto the **no network assigned** field, next to the **eth1** network interface.

3.6. Click and drag the **prod-vm-net (VLAN 20)** box from the right side to the left side of the window. Drop this box onto the **eth0** interface field. After dropping, you should see two logical networks assigned to **eth0** interface.

3.7. Click the **pencil** icon inside the **prod-storage** box. In the **Edit Network prod-storage** dialog window, under **Boot Protocol**, select **Static** as the **Boot Protocol**.

3.8. In the **IP** field, type in **172.24.0.12** as the IP address of **hostc** in that network.

3.9. In the **Netmask/Routing Prefix** field, type **255.255.255.0** as the netmask.

3.10. Click the **OK** button to save the settings.

3.11. Ensure that the check boxes for the **Verify connectivity between Host and Engine** and **Save network configuration** options are selected.

3.12. Click the **OK** button to confirm and save the new network configuration for that host.

3.13. Both new logical networks in the **clustertwo** cluster are now accessible by the **hostc** host.

Evaluation

On **workstation**, run the **lab networks-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab networks-review grade
```

Finish

On **workstation**, run the **lab networks-review finish** script to complete this exercise.

```
[student@workstation ~]$ lab networks-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- Logical networks allow different types of network traffic to be separated onto different VLANs or physical networks.
- Logical networks are defined in a particular data center, and assigned to one or more clusters for specific uses.
- VM Networks can be used by virtual machines directly, and a Linux bridge is used by the host to connect virtual machines on a VM Network to one of its network interfaces.
- Once a logical network is assigned to a cluster, it must also be associated with a specific network interface on a host so that it can be used.
- Network labels can be used to automatically assign logical networks to network interfaces on hosts.

Chapter 7

Managing RHV Storage

Goal	Create and manage data storage domains.
Objectives	• Explain how data storage domains and the Storage Pool Manager work. • Create and manage data storage domains from NFS, iSCSI, and GlusterFS sources. • Explain how to configure volume and image storage from an external OpenStack provider.
Sections	• Creating and Managing Storage Domains (and Guided Exercise) • Configuring External Storage Providers (and Quiz)
Lab	Managing RHV Storage

Creating and Managing Storage Domains

Objectives

After completing this section, you should be able to:

- Describe how data storage domains and the Storage Pool Manager work.
- Create and manage data storage domains from NFS, iSCSI and GlusterFS sources.

Storage Domain Overview

A storage domain is a collection of images with a common storage interface. A storage domain contains images of templates, virtual machines, snapshots, and ISO files. A storage domain is made of either block devices (iSCSI, FC) or a file system (NFS, GlusterFS).

On file system backed storage, all virtual disks, templates, and snapshots are files. On block device backed storage, each virtual disk, template, or snapshot is a logical volume. Block devices are aggregated into a logical entity called a volume group, and then divided by LVM (Logical Volume Manager) into logical volumes for use as virtual hard disks.

Virtual disks use either the QCOW2 or raw format. Storage can be either sparse or preallocated. Snapshots are always sparse, but can be taken for disks of either format. Virtual machines that share the same storage domain can migrate between hosts in the same cluster.

Types of Storage Domain Back Ends

Storage domains may use file-based or block-based storage. The supported file-based storage types are NFS, GlusterFS, other POSIX compliant file system, or local host storage devices. The supported block-based storage types are iSCSI and Fibre Channel (FC).

NFS

Red Hat Virtualization (RHV) supports NFS exports to create a storage domain. NFS exports are easy to manage, and work seamlessly with RHV. RHV recognizes NFS export resizing immediately, without requiring any additional manual intervention. NFS is supported for data, ISO, and export storage domains. When enterprise NFS is deployed over 10GbE, segregated with VLANs, and individual services are configured to use specific ports, it is both fast and secure.

GlusterFS

RHV supports the native GlusterFS driver for creating a Red Hat Gluster Storage backed data storage domain. Three or more servers are configured as a Red Hat Gluster Storage server cluster, instead of using a SAN array. Red Hat Storage should be used over 10GbE and segregated with VLANs. Red Hat Gluster Storage is only supported for data domains.

> **Important**
>
> Not all versions of RHV work with all versions of Red Hat Gluster Storage. Refer to the *Red Hat Gluster Storage Version Compatibility and Support* document at https://access.redhat.com/articles/2356261 to verify.

iSCSI

An iSCSI-based storage domain enables RHV to use existing Ethernet networks to connect to the block devices presented as LUNs in an iSCSI target. iSCSI-based storage is only supported for data domains. In a production environment, also consider booting hosts from the enterprise grade iSCSI server Enterprise iSCSI servers have native cloning features for easy deployment of new hosts using host templates. For optimum performance, hosts should use hardware-based iSCSI initiators and deploy over 10 GbE or faster networks.

FC SAN

RHV also supports fast and secure Fibre Channel based SANs to create data storage domains. If you already have FC SAN in your environment, then you should take advantage of it for RHV. However, FC SANs require specialized network devices and skills to operate. Like iSCSI, a FC SAN also supports booting hosts directly from storage. FC SAN has the native cloning features to support easy deployment of new hosts using host templates.

Local storage

Local storage should only be considered for small lab environments. Do not use local storage for production workloads. Local storage precludes the use of live migration, snapshots, and the flexibility that virtualization supports.

Describing the Storage Pool Manager

The host that can make changes to the structure of the data domain is known as the *Storage Pool Manager* (SPM). The SPM coordinates all metadata changes in the data center, such as creating and deleting disk images, creating and merging snapshots, copying images between storage domains, creating templates, and storage allocation for block devices. There is one SPM for every data center. All other hosts can only read storage domain structural metadata.

All hosts can read and write to stored images within a storage domain, but only the SPM can apply changes to the configuration of storage domains. Either manually enable a host as the SPM, or let RHV-M select one automatically.

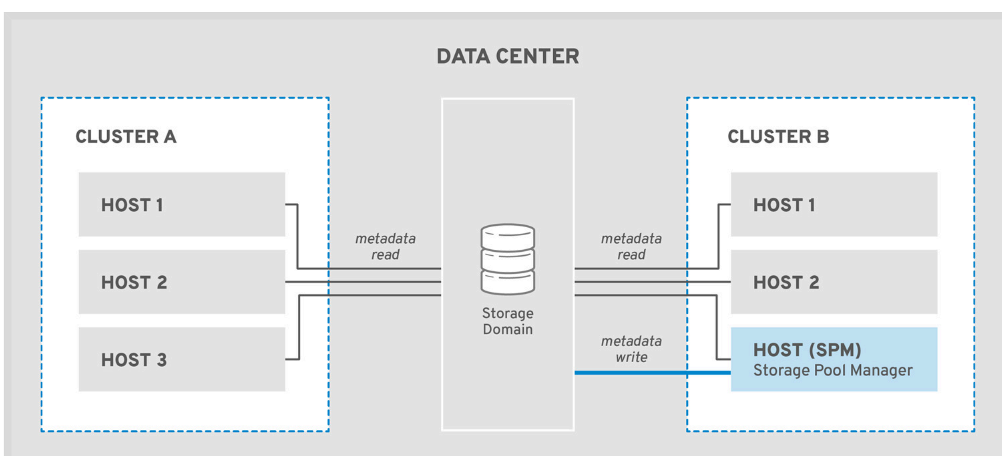

Figure 7.1: Storage Pool Manager writing metadata

RHV-M identifies the SPM based on the SPM assignment history for the host. If the host was the last SPM used, RHV-M selects the host as the SPM. If the selected SPM is unresponsive, RHV-M randomly selects another potential SPM. This selection process requires the host to assume and retain a storage-centric lease, which allows the host to modify storage metadata. Storage-centric leases are saved in the storage domain, rather than in the RHV-M or hosts.

The SPM must be running on a data center host to add and configure storage domains. An administrator must register a host (hypervisor) before setting up a new data center. Once a host is part of the data center, it is possible to configure the data center storage domains.

In an NFS data domain, the SPM creates a virtual machine disk as a file in the file system, either as a QCOW2 file for thin provisioning (sparse), or as a normal file for preallocated storage space (RAW).

In an iSCSI or FC data domain, the SPM creates a volume group (VG) on the storage domain's LUN, and creates a virtual machine disk as a logical volume (LV) in that volume group. For a preallocated virtual disk, the SPM creates a logical volume of the specified size (in GB). For a thin provisioned virtual disk, the SPM initially creates a 512 MB logical volume. The host on which the virtual machine is running continuously monitors the logical volume. If the host determines that more space is needed, then the host notifies the SPM, and the SPM extends the logical volume by another 512 MB.

From a performance standpoint, a preallocated (RAW) virtual disk is significantly faster than a thin provisioned (QCOW2) virtual disk. The recommended practice is to use thin provisioning for non-I/O intensive virtual desktops, and preallocation for virtual servers.

Storage Domain Types

Red Hat Virtualization supports three types of storage domains. Beginning in RHV 4, the ISO and export storage domains are deprecated, but remain available and supported. Functionality for ISO and export storage domains is now handled by data storage domains.

Data Storage Domain
: A data storage domain stores hard disk images for virtual machines. These disk images can contain the operating system or other virtual machine data. ISO disk images, used to install operating systems and applications, are uploaded to a data domain. When creating a new virtual machine, an ISO image from a data storage domain is attached as a virtual CD/DVD drive. Data storage domains support NFS, iSCSI, FC, GlusterFS and POSIX compliant storage backing. Every RHV data center must have at least one data storage domain. A data domain can be associated with only one RHV data center at a time.

ISO Storage Domain
: An ISO storage domain stores disk images used to install virtual machine operating systems and applications. These are ISO 9660-formatted CD or DVD images. Unlike the data storage domain, the ISO storage domain can store only ISO images and not hard disk images. Only one ISO domain is needed in a data center, and a single ISO domain can be shared by multiple data centers in a deployed RHV environment. An ISO storage domain supports the NFS, GlusterFS, and POSIX compliant storage backing.

Export Storage Domain
: An export storage domain holds hard disk images and virtual machine templates to transfer between data centers. An export storage domain can be attached to only one data center at a time. To share the domain contents across data centers, detach the export storage domain from the current data center and attach it to the intended data center. Export storage domains support the NFS, GlusterFS, and POSIX compliant storage backing.

> **Note**
> The ISO and export storage domains are deprecated. Instead, upload ISO images to any data storage domain. Also, use a data storage domain to transfer data between data centers. To move images and templates, detach the data storage domain from the source data center and attach it to the destination data center.

Configuring an NFS backed Storage Domain

To configure an NFS backed storage domain, navigate to **Storage** → **Domains** from the **Administration Portal** menu. From the **Storage Domains** page, click the **New Domain** button to open the **New Domain** window. Specify appropriate values for the required fields.

Enter a meaningful name and optional description for the new storage domain. Select the data center from the drop-down list in the **Data Center** field that the storage domain will serve. New virtual disks created for virtual machines assigned to this data center are stored in the storage domain for this data center. The **Domain Function** field selects one of the three storage domain types described previously.

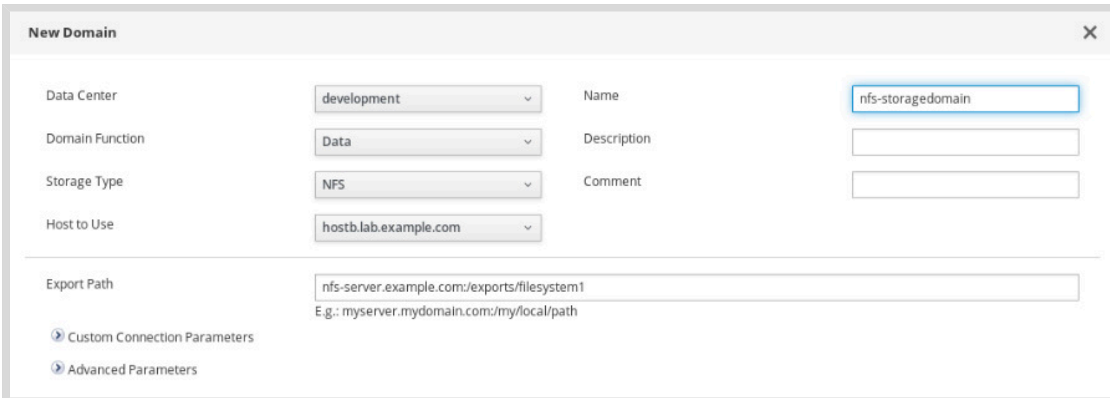

Figure 7.2: Creating an NFS-based storage domain

When configuring an NFS backed storage domain, select **NFS** from the drop-down list for the **Storage Type** field to enable additional NFS related fields. In the **Export Path** field, enter a complete NFS server and share name, such as ***nfs-server*:/*filesystem_path***. The NFS shares must be previously created, and configured with correct ownership and permissions, to be available for connection.

RHV will attempt to access and create initial metadata structures on the NFS share. In the **Host to Use** field, you can specify any data center host that is correctly configured to access the share URL for the NFS server. The designated host will NFS mount the share and perform the initial storage domain configuration tasks.

> **Note**
> When specifying the **Host to Use**, any correctly configured data center host is sufficient. You do not need to select the SPM. Also, this selection does not make that host preferred for subsequent, ongoing storage domain activity.

Click the **OK** button to create the NFS storage domain. If successfully created, then the storage domain will list as active in the **Storage Domains** table.

Configuring an iSCSI backed Storage Domain

Red Hat Virtualization supports iSCSI backed storage to create a data storage domain. iSCSI storage domains are configured as iSCSI server targets containing configured LUNs. Hosts act as iSCSI initiators, and log in to the iSCSI target specified by the data domain to locate the LUNs. Only one storage domain at a time can connect to and use each iSCSI LUN.

To configure an iSCSI backed data storage domain, navigate to **Storage** → **Domains** from the **Administration Portal** menu. From the **Storage Domains** page, click the **New Domain** button to open the **New Domain** window. Specify appropriate values for the required fields.

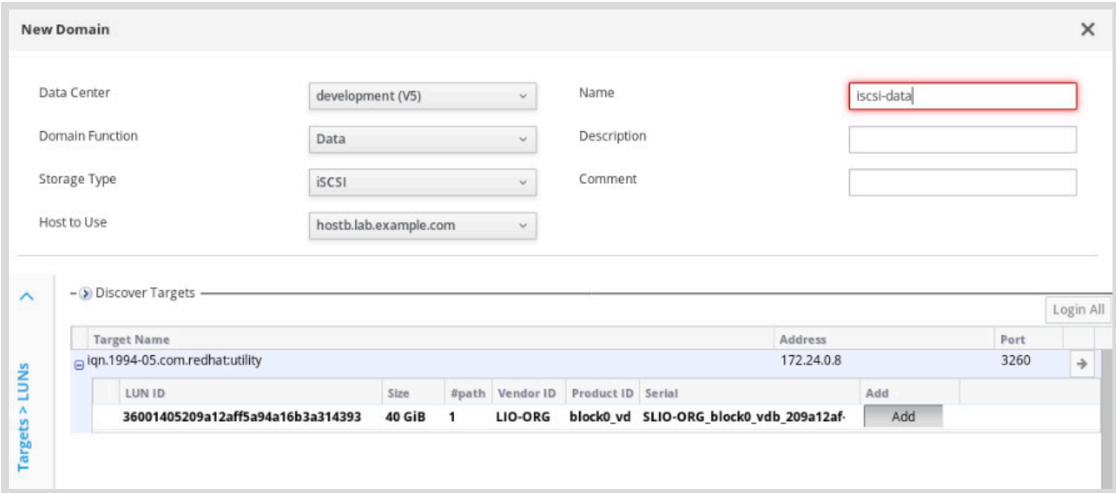

Figure 7.3: Creating an iSCSI backed storage domain

When configuring an iSCSI backed storage domain, selecting **iSCSI** from the drop-down list for the **Storage Type** field enables additional iSCSI related fields. Set the IP address and port for the iSCSI target server, and then click **Discover** to discover the available targets. Select the right arrow button to log in to the presented target.

After successfully logging in to the target, a **+** button appears next to the iSCSI target. Clicking on the button expands and displays a list of available **`iSCSI`** LUNs for the target. An **Add** button displays next to each LUN, to include the LUN in the storage domain. After adding one or more LUNs, click the **OK** button to confirm creating the iSCSI backed storage domain.

> **Note**
>
> Previously used LUNs contain metadata that prevents the LUN from being accidentally selected for another domain. To reuse a LUN, you must clear the partition table for that LUN. To determine the LUN ID, use the **`multipath -l`** command on any RHV Host supporting the iSCSI storage domain.
>
> ```
> host# dd if=/dev/zero of=/dev/mapper/LUN_ID bs=1M count=200 oflag=direct
> ```
>
> After clearing the partition table for a LUN, use the **`systemctl`** command to stop all of the **`multipathd.service`** instances on every domain host. When all **`multipathd.service`** instances are stopped, start them all again.

Configuring a GlusterFS backed Storage Domain

Red Hat Virtualization supports using Red Hat Gluster Storage to create block device backed storage domains. Red Hat Gluster Storage presents a GlusterFS *volume* made of combined XFS file system *bricks*, which are redundantly distributed across multiple storage nodes for performance and resiliency. GlusterFS provides flexible and scalable storage to handle the dynamic storage growth of data centers in Red Hat Virtualization.

Install the *glusterfs-fuse* and *glusterfs* packages on your data center hosts to enable native GlusterFS client support.

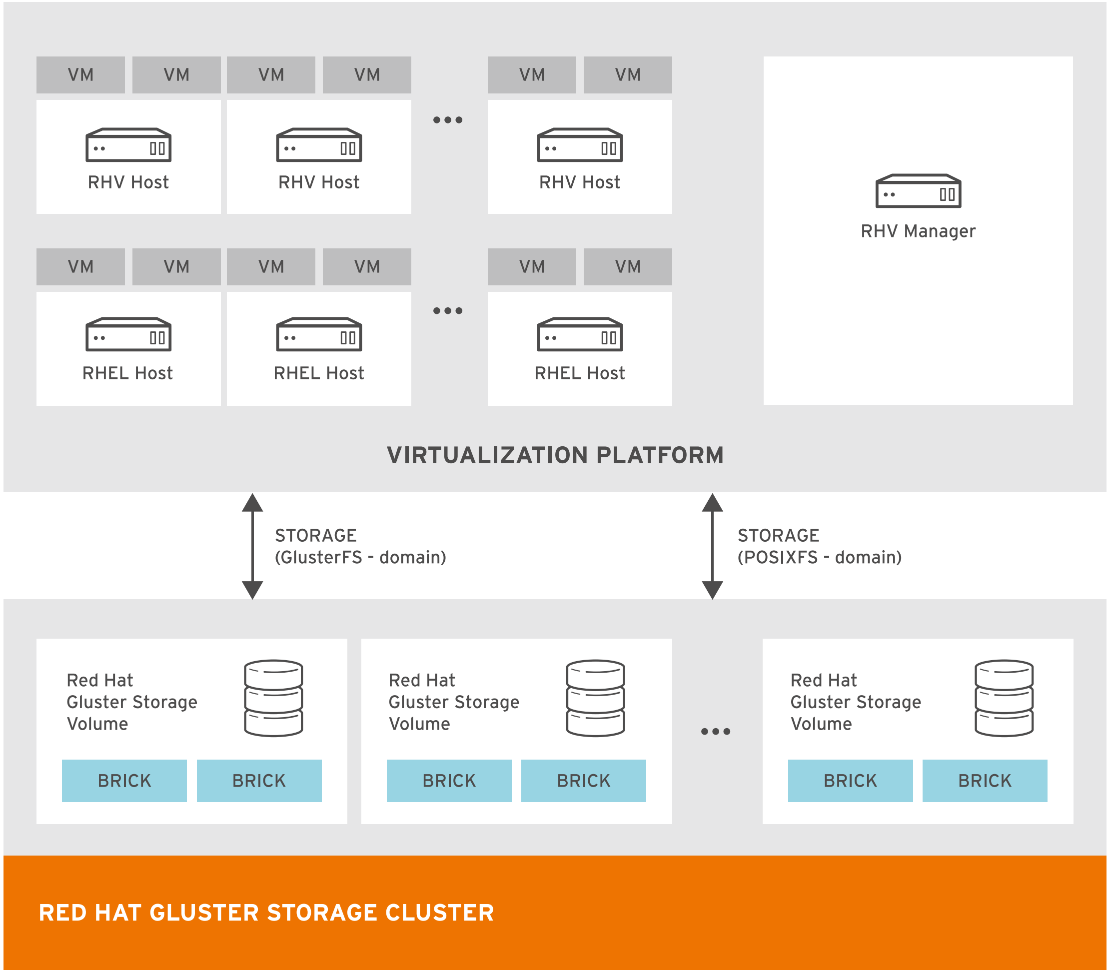

Figure 7.4: Red Hat Virtualization Integrated with Red Hat Gluster Storage

To configure a GlusterFS backed data storage domain, navigate to **Storage** → **Domains** from the **Administration Portal** menu. Specify appropriate values for the required fields.

When configuring an iSCSI backed storage domain, selecting **GlusterFS** from the drop-down list for the **Storage Type** field enables additional GlusterFS related fields. In the **Path** field, specify the IP address and name for the GlusterFS volume using a colon (`:`) as the delimiter. Be sure to specify any required **Mount Options** that you generally use on your GlusterFS **`mount`** command. Click the **OK** button to confirm the creation of the storage domain.

References

For more information, refer to the *Storage* chapter in the *Red Hat Virtualization Administration Guide* at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/chap-storage

Guided Exercise

Creating and Managing Storage Domains

In this exercise, you will create a data storage domain using iSCSI, and make it available to the data center.

Outcomes

You should be able to create a new data storage domain for the **development** data center using iSCSI.

Before You Begin

This exercise uses the **development** data center, the **clusterone** cluster, and the **hosta** and **hostb** hosts. The iSCSI traffic uses the **storage-net** logical network. The host, **utility.lab.example.com**, has been configured as an iSCSI target.

Log in as the **student** user on **workstation** and run the **lab storage-domains start** command. This command ensures that the users, hosts, clusters, data centers, and networks of the Red Hat Virtualization environment are configured appropriately.

```
[student@workstation ~]$ lab storage-domains start
```

1. Create an iSCSI-based storage domain called **iscsi-data** to function as the data domain in the **development** data center. Use a LUN from the iSCSI target on the **172.24.0.8** address of **utility**. The **172.24.0.8** IP address belongs to the storage network.

   1.1. Using the `https://rhvm.lab.example.com` URL, log in to the **Administration Portal** of the Red Hat Virtualization Manager. The user is **admin**, the profile is **internal**. Use **redhat** as the password.

   1.2. In the web interface, click **Storage** on the menu, and then click **Domains**.

   1.3. Click **New Domain**.

   1.4. In the **New Domain** window, set the values of the fields according to the following table.

Field	Value
Data Center	**development**
Domain Function	**Data**
Storage Type	**iSCSI**
Hosts to Use	**hostb.lab.example.com**
Name	**iscsi-data**

1.5. In the **Discover Targets** section, specify **`172.24.0.8`** in the **Address** field. Set the **Port** field to 3260, if not already set. Click **Discover** to display the available iSCSI target LUNs.

The **`utility`** system uses the **`172.24.0.8`** IP address in the storage network.

1.6. Verify that the **Targets > LUNs** section includes the **`iqn.2019-07.com.example.lab:utility`** target name. Click the right arrow button for the **`iqn.2019-07.com.example.lab:utility`** target name to log in to it.

1.7. Click **+** next to the **`iqn.2019-07.com.example.lab:utility`** target name to expand and display the list of available iSCSI target LUNs. Click **Add** for the displayed iSCSI target LUN. Click **OK** to create the **`iscsi-data`** storage domain.

1.8. From the **Storage Domains** page under **Storage**, verify that the **`iscsi-data`** storage domain exists with an **Active** status in the **Cross Data Center Status** column. It may take a couple of minutes for the **`iscsi-data`** storage domain status to transition from **Locked** to **Active**.

2. Sign out as **`admin`** from the **Administration Portal**.

Finish

On **`workstation`**, run the **`lab storage-domains finish`** script to complete this exercise.

```
[student@workstation ~]$ lab storage-domains finish
```

This concludes the guided exercise.

Configuring External Storage Providers

Objectives

After completing this section, you should be able to describe how to configure volume and image storage from an external OpenStack provider.

Describing External Providers

Red Hat Virtualization can be configured to use storage resources from certain providers external to RHV. Like RHV, Red Hat OpenStack Platform (RHOSP) also provisions virtual machines and containers, but in cloud environments for self-service users. RHOSP uses ISOs and images from its OpenStack image service (Glance) to build and deploy virtual machine instances, and stores instance hard disk and snapshot volumes in its OpenStack block storage (Cinder) service.

Adding the OpenStack Glance and Cinder services to RHV as external storage providers has multiple advantages. ISOs, images, and hard disk volumes already used in OpenStack can be imported for use in RHV, saving the effort of duplicating images for both environments. Other than the import process, RHV administrators have no increased management tasks, since the original OpenStack images are created and managed by OpenStack users. Similar, RHV image files can be exported for use in OpenStack to create and deploy virtual machine instances.

Note

Concerning terminology: An OpenStack image is equivalent to a RHV template. Both are used to deploy virtual machines. The Red Hat Ceph Storage in OpenStack and the Red Hat Gluster Storage in RHV both create VM hard disks as volumes in storage. OpenStack refers to these disks as volumes. Also, OpenStack refers to VMs as instances, since many stateless cloud VMs are also ephemeral, meaning that they do not attach persistent virtual disks.

OpenStack Glance for Image Management

The OpenStack image service provides a catalog of virtual machine images. In RHV, these images can be imported and used as floating disks, or attached to virtual machines and converted into templates. The Glance service is used as a storage domain that is not attached to any data center. Virtual disks in RHV can also be exported to Glance as virtual disks. Imported OpenStack images must be converted to templates to be used for deploying new virtual machines in RHV.

OpenStack Cinder for Storage Management

The OpenStack block storage service provides persistent block storage management for virtual hard disks. Cinder volumes are provisioned by Ceph Storage. In RHV, you can create disks on Cinder storage that can be used as floating disks or attached to virtual machines. After you add Cinder to the Manager, you can create a disk on the storage provided by Cinder.

Configuring External Providers

To add an external provider, navigate to **Administration** → **Providers** from the **Administration Portal** menu. From the **Providers** page, click the **Add** button to open the **Add Provider** window. Specify appropriate values for the required fields.

Adding an OpenStack Glance External Provider

In the **Add Provider** window, select **OpenStack Image** from the drop-down list for the **Type** field to create the Glance external provider. Enter a meaningful name and description for the new OpenStack image external provider. Use the **Provider URL** field to specify the URL to your OpenStack image service, such as **`http://`*`server`*`:`*`port`***. The default OpenStack image service port is 9292.

The OpenStack image service uses a service account, registered in the OpenStack identity service, for authentication and authorization to the rest of the OpenStack infrastructure. Using credentials for that account, the OpenStack image service obtains the required OpenStack identity service tokens to communicate with other OpenStack services. Select the **Requires Authentication** checkbox to specify the authentication credentials for RHV-M to successfully request the required tokens on behalf of the OpenStack image service.

Adding an OpenStack Cinder External Provider

In the **Add Provider** window, select **OpenStack Block Storage** from the drop-down list for the **Type** field to create the Cinder external provider. Enter a meaningful name and description for the new OpenStack image external provider. and then select the data center. Use the **Provider URL** field to specify the URL to your OpenStack block storage service, such as **`http://`*`server`*`:`*`port`***. The default OpenStack block storage service port is 8776.

The OpenStack block storage service also uses a service account, registered in the OpenStack identity service, for authentication and authorization to the rest of the OpenStack infrastructure. Select the **Requires Authentication** checkbox, and then specify the authentication credentials for RHV-M to successfully request the required tokens on behalf of the OpenStack block storage service.

The authentication credentials include the following fields.

Username
: The user name of the OpenStack service user account.

Password
: The password of the OpenStack service user account.

Protocol
: The service connection protocol, either insecure (**`HTTP`**) or secure(**`HTTPS`**).

Host Name
: The fully qualified domain name of the OpenStack identity (Keystone) server.

API Port
: The Keystone service port that listens for incoming authentication requests. The default value is port 35357.

Tenant Name
: Every user account in the OpenStack environment is associated with a tenant. The tenant is a top-level container for organizing resources in OpenStack. OpenStack service accounts are associated with the **`services`** tenant. Use an appropriate value obtained from your OpenStack environment.

References

For more information, refer to the *External Providers* chapter in the *Red Hat Virtualization Administration Guide* at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#chap-External_Providers

Quiz

Configuring External Storage Providers

Choose the correct answers to the following questions:

1. **Which external provider type allows Red Hat Virtualization to import virtual machine images from an external OpenStack image service?**
   a. **`KVM`**
   b. **`Foreman/Satellite`**
   c. **`OpenStack Image`**
   d. **`OpenStack Block Storage`**

2. **Which external provider type allows Red Hat Virtualization to create disks on the storage device of the external OpenStack block storage service?**
   a. **`External Network Provider`**
   b. **`OpenStack Networking`**
   c. **`OpenStack Image`**
   d. **`OpenStack Block Storage`**

3. **Which statement is correct regarding the authentication credentials you set when creating the OpenStack image service external provider?**
   a. The authentication credentials set for the OpenStack image service external provider enables Red Hat Virtualization Manager to authenticate to the OpenStack identity service on behalf of the OpenStack image service.
   b. The authentication credentials set for the OpenStack image service external provider enables the **`admin`** user of the Red Hat Virtualization Manager to log in to the OpenStack dashboard using single sign-on.
   c. The authentication credentials set for the OpenStack image service external provider authorizes the Red Hat Virtualization Manager to modify the OpenStack service catalog.
   d. The authentication credentials set for the OpenStack image service external provider authorizes the Red Hat Virtualization Manager to modify the resources in the OpenStack environment.

4. Which statement is correct regarding the external provider in the Red Hat Virtualization environment?

a. The external providers are responsible for synchronizing the status of each Red Hat Virtualization Host with the other hosts in the cluster.

b. The external providers provide resources, such as virtual machine images, Red Hat Virtualization Hosts, or networks.

c. The external providers are responsible for the power management of the Red Hat Virtualization Hosts in the cluster.

d. The external providers control the life cycle of the virtual machines in the Red Hat Virtualization environment.

Solution

Configuring External Storage Providers

Choose the correct answers to the following questions:

1. **Which external provider type allows Red Hat Virtualization to import virtual machine images from an external OpenStack image service?**
   a. `KVM`
   b. `Foreman/Satellite`
   c. **`OpenStack Image`**
   d. `OpenStack Block Storage`

2. **Which external provider type allows Red Hat Virtualization to create disks on the storage device of the external OpenStack block storage service?**
   a. `External Network Provider`
   b. `OpenStack Networking`
   c. `OpenStack Image`
   d. **`OpenStack Block Storage`**

3. **Which statement is correct regarding the authentication credentials you set when creating the OpenStack image service external provider?**
   a. The authentication credentials set for the OpenStack image service external provider enables Red Hat Virtualization Manager to authenticate to the OpenStack identity service on behalf of the OpenStack image service.
   b. The authentication credentials set for the OpenStack image service external provider enables the `admin` user of the Red Hat Virtualization Manager to log in to the OpenStack dashboard using single sign-on.
   c. The authentication credentials set for the OpenStack image service external provider authorizes the Red Hat Virtualization Manager to modify the OpenStack service catalog.
   d. The authentication credentials set for the OpenStack image service external provider authorizes the Red Hat Virtualization Manager to modify the resources in the OpenStack environment.

4. Which statement is correct regarding the external provider in the Red Hat Virtualization environment?

a. The external providers are responsible for synchronizing the status of each Red Hat Virtualization Host with the other hosts in the cluster.

b. The external providers provide resources, such as virtual machine images, Red Hat Virtualization Hosts, or networks.

c. The external providers are responsible for the power management of the Red Hat Virtualization Hosts in the cluster.

d. The external providers control the life cycle of the virtual machines in the Red Hat Virtualization environment.

Lab

Managing RHV Storage

Performance Checklist

In this lab, you will create a storage domain using **NFS** and make it available to the data center and cluster.

Outcomes

You should be able to create an **NFS**-based data storage domain.

Before You Begin

This exercise uses the **production** data center, the **clustertwo** cluster, and the **hostc** host. The NFS traffic uses the **storage-net** logical network. The host **utility.lab.example.com** has been configured as an NFS server.

Log in as the **student** user on **workstation** and run the **lab storage-review start** command. This command ensures that the users, hosts, clusters, data centers, and networks of the Red Hat Virtualization environment are configured appropriately. This command also verifies that the **NFS** share is available.

```
[student@workstation ~]$ lab storage-review start
```

1. Create an **NFS**-based storage domain called **nfs-data** to function as the data domain in the **production** data center. This storage domain should use **172.24.0.8:/exports/data** as the **NFS** export path in the back end for the **nfs-data** storage domain in the **production** data center. The **172.24.0.8** IP address belongs to the storage network.
2. Sign out as **admin** from the **Administration Portal**.

Evaluation

On **workstation**, run the **lab storage-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab storage-review grade
```

Finish

On **workstation**, run the **lab storage-review finish** script to complete this lab.

```
[student@workstation ~]$ lab storage-review finish
```

This concludes the lab.

Solution

Managing RHV Storage

Performance Checklist

In this lab, you will create a storage domain using **NFS** and make it available to the data center and cluster.

Outcomes

You should be able to create an **NFS**-based data storage domain.

Before You Begin

This exercise uses the **production** data center, the **clustertwo** cluster, and the **hostc** host. The NFS traffic uses the **storage-net** logical network. The host **utility.lab.example.com** has been configured as an NFS server.

Log in as the **student** user on **workstation** and run the **lab storage-review start** command. This command ensures that the users, hosts, clusters, data centers, and networks of the Red Hat Virtualization environment are configured appropriately. This command also verifies that the **NFS** share is available.

```
[student@workstation ~]$ lab storage-review start
```

1. Create an **NFS**-based storage domain called **nfs-data** to function as the data domain in the **production** data center. This storage domain should use **172.24.0.8:/exports/data** as the **NFS** export path in the back end for the **nfs-data** storage domain in the **production** data center. The **172.24.0.8** IP address belongs to the storage network.

    1.1. Using the `https://rhvm.lab.example.com` URL, log in to the **Administration Portal** of the Red Hat Virtualization Manager. The username is **admin**, the profile is **internal**, and the password is **redhat**.

    1.2. In the web interface, click **Storage** on the menu bar, and then click **Domains** from the options that display.

    1.3. Click **New Domain**.

    1.4. In the **New Domain** window, set the values of the fields according to the following table.

Field	Value
Data Center	**production**
Domain Function	**Data**
Storage Type	**NFS**
Hosts to Use	**hostc.lab.example.com**
Name	`nfs-data`
Export Path	`172.24.0.8:/exports/data`

1.5. Click **OK** to create the **`nfs-data`** storage domain.

1.6. From the **Storage Domains** page under **Storage**, verify that the **`nfs-data`** storage domain exists, and displays the **Active** status in the **Cross Data Center Status** column. It may take a couple of minutes for the **`nfs-data`** storage domain status to transition from **Locked** to **Active**.

2. Sign out as **`admin`** from the **Administration Portal**.

Evaluation

On **`workstation`**, run the **`lab storage-review grade`** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab storage-review grade
```

Finish

On **`workstation`**, run the **`lab storage-review finish`** script to complete this lab.

```
[student@workstation ~]$ lab storage-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- The Storage Pool Manager (SPM) updates the storage domain configuration and metadata requested by Red Hat Virtualization Manager.
- When creating a storage domain, the host that can access the new storage is specified in the **Host to Use** field.
- When using iSCSI backed data domains, the SPM uses logical volume management (LVM) to configure each virtual machine disk.
- The external providers help Red Hat Virtualization to use the preexisting resources from an external source.

Chapter 8

Deploying and Managing Virtual Machines

Goal	Operate virtual machines in the Red Hat Virtualization environment.
Objectives	• Install virtual machines that are optimized and configured for the Red Hat Virtualization environment. • Change configuration or virtual hardware of an existing virtual machine. • Create a template of a Red Hat Enterprise Linux virtual machine, and use it to deploy a new virtual machine. • Deploy a virtual machine using a template configured to use cloud-init.
Sections	• Installing Virtual Machines (and Guided Exercise). • Editing Virtual Machine Hardware (and Guided Exercise). • Creating and Deploying Virtual Machines with Templates (and Guided Exercise). • Automating Virtual Machine Configuration (and Guided Exercise).
Lab	Deploying and Managing Virtual Machines.

Installing Virtual Machines

Objectives

After completing this section, you should be able to install virtual machines that are optimized and configured for the Red Hat Virtualization environment.

Describing Virtual Machines

Red Hat Virtualization allows you to create virtual machines (guests) running different operating systems. Various Red Hat and third-party operating systems are certified and supported as guest operating systems.

A list of supported guest operating systems for a 32-bit (x86) or 64-bit (x86-64) Red Hat Virtualization environment is available. This list may be updated from time to time; a current list can be found on the Red Hat Customer Portal at https://access.redhat.com/articles/973163/.

Operating System	Version
Red Hat Enterprise Linux	3, 4, 5, 6, 7, 8*
Microsoft Windows Server	2008, 2008 (R2), 2012, 2012 R2, 2016, 2019**
Microsoft Windows	7, 8, 8.1, 10
SUSE Linux Enterprise Server	10, 11, 12, 15

* Red Hat Enterprise Linux 8 is supported by Red Hat Virtualization 4.3 with the following limitations: SSO in gnome environments is not supported, only the qemu-guest-agent is supported, virt-sysprep (template sealing) and virt-sparsify are not supported, and v2v conversion of RHEL 8 is not supported.

** Microsoft Windows Server 2019 is only certified for compatibility with Red Hat Enterprise Linux 8.

Installing a New Virtual Machine

A new virtual machine can be installed using either the **Administration Portal** or the **VM Portal**. This is a basic outline of the steps required to install a new virtual machine:

- Create a blank virtual machine for the new operating system. Choose hardware that meets the requirements of your new operating system, and create a new virtual disk for storage. Create one or more network interfaces to connect the virtual machine to the required logical networks.

- Boot the virtual machine from the installation CD and install the operating system.

- Start the virtual machine, and then install guest agents and drivers to extend the functionality.

Creating the Virtual Machine

The first step is to create a blank virtual machine. There are many ways to do this, but one method uses the Administration Portal. From **Compute**, select **Virtual Machines** and then click the **New** button. This opens the **New Virtual Machine** window.

When advanced options are not displayed, there are two tabs available. The **General** tab provides key configuration settings for your virtual machine. The **Console** tab provides settings that affect the console of your virtual machine (the virtualized "physical" monitor for the system).

On the **General** tab, **Cluster** is the name of the cluster in which the new virtual machine will be created. Since a cluster belongs to a data center, the virtual machine will belong to the same data center as the cluster.

The **Template** menu lists templates available in the selected data center and cluster. Templates enable the rapid creation of new virtual machines. If no templates are available, the **Template** menu defaults to **Blank**.

Operating System configures the virtual machine with virtualized devices that are likely to be supported by a particular operating system.

Instance Type configures the default hardware settings, including the number of CPUs and the amount of memory, for the new virtual machine. Instance types can be customized by clicking **Administration** in the menu bar, selecting **Configure**, and then clicking the **Instance Types** tab from the **Configure** window.

The five default instance types are:

- **`Tiny`**: 1 vCPU, 512 MB RAM
- **`Small`**: 1 vCPU, 2048 MB RAM
- **`Medium`**: 2 vCPUs, 4096 MB RAM
- **`Large`**: 2 vCPUs, 8192 MB RAM
- **`XLarge`**: 4 vCPUs, 16384 MB RAM

The **Optimized for** option affects some advanced settings for persistence and configuration. Select **Server** for most virtual machines.

The fields for **Name**, **Description**, and **Comment** are self-explanatory.

The **Instance Images** section is used to configure local storage for the virtual machine. Click the **Create** button to create a new disk. The **New Virtual Disk** window displays. In this window, you are presented with options to select a size, alias, and data domain for the disk. Two other options are particularly important:

- **Interface** specifies the hardware interface for storage on the virtual machine. **VirtIO-SCSI** and **VirtIO** are faster, but require that you have paravirtualized guest drivers for the virtual machine operating system. Red Hat Enterprise Linux is installed with these drivers by default. **IDE** emulates a basic IDE interface, which is supported by most operating systems.
- **Allocation Policy** specifies whether the whole disk is created immediately (**Preallocated**), or whether the size of the disk is increased as needed. A disk specified as **Thin Provision** only allocates what the virtual machine requires.

  Thin Provision is generally faster for deployment, backup, replication, and recovery. However, as storage grows, the underlying storage needs to be allocated on the fly, resulting in wait time and lower performance. Preallocated is faster from a performance standpoint, but takes up more space.

If the storage array backing your data domain supports deduplication, you can configure the hardware storage array to thin provision, enable deduplication at that level, and then preallocate the virtual machine storage in Red Hat Virtualization.

A recommended practice is to give your virtual machine a local boot disk managed by RHV that contains the operating system and application binaries. Any additional disks for application data are probably best managed externally and accessed directly by the virtual machine.

The advantage of this method is that virtual machines can be managed and backed up separately from application data. A virtual machine can be rebuilt quickly from a snapshot or template, and then the application data can be remounted from the SAN or NAS. Performance may also be better for the application by not having the Red Hat Virtualization host translate storage traffic for the virtual machine.

To configure the network interface, use the **Instantiate VM network interfaces by picking a vNIC profile** pane. One network interface (`nic1`) is defined by default, and the drop-down menu next to it may be used to attach the interface to a virtual machine logical network. After configuring one vNIC, you can click the **+** button to add additional network interfaces to the virtual machine.

The **Show Advanced Options** button causes the **New Virtual Machine** window to display a number of advanced options. These can be used to customize vCPU and memory configuration, adjust the boot order, or enable advanced features. For more information on all the options available, look in the appendix "Reference: Settings in Administration Portal and User Portal Windows" in the *Virtual Machine Management Guide* at https://access.redhat.com/documentation/.

Installing a Virtual Machine Using an ISO Image

To install a virtual machine using an ISO image, select the row containing the virtual machine, click the **Run** menu, and select **Run Once**. This opens a **Run Virtual Machine(s)** window, which lets you define boot options. For example, to boot a virtual machine from the available ISO using the ISO library, click the **+** button next to the **Boot Options** label. This opens a new section of the window, where you can specify the way this virtual machine will boot. Changes made with the **Run Once** option are used only once.

To attach the installation ISO to the virtual CD-ROM drive of your virtual machine, select the check box next to the **Attach CD** label. This activates the virtual CD-ROM drive for this boot, and lets you choose the appropriate ISO file from a list of all available ISOs in your library.

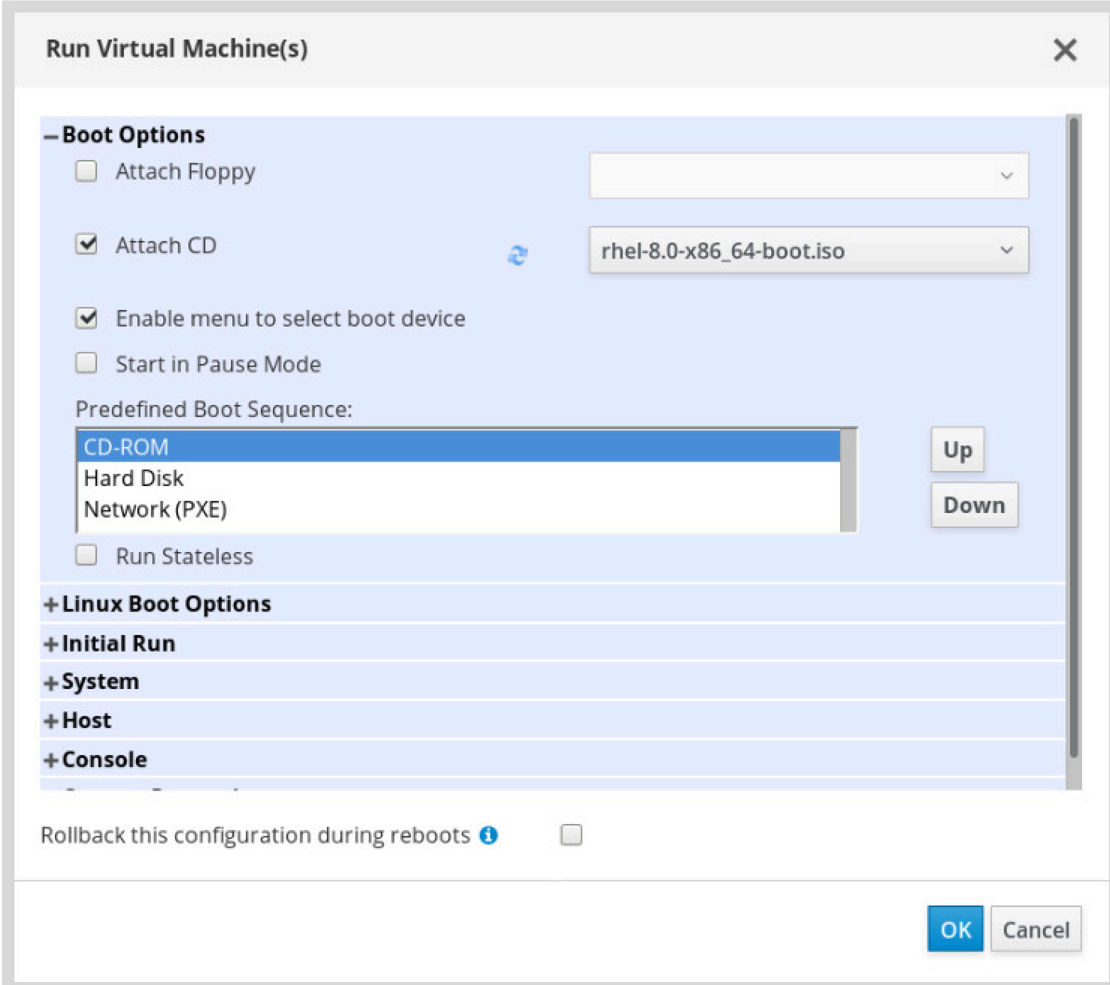

Figure 8.1: Run Virtual Machine(s) window

With the correct ISO file chosen and attached, specify the desired boot order for the virtual machine. In this example, highlight the **CD-ROM** drive by clicking on its name in the **Predefined Boot Sequence** list. This chooses the virtual CD-ROM drive and activates two additional buttons: **Up** and **Down**. Because you want to boot from the ISO file mounted in your virtual CD-ROM drive, use the **Up** button to bring the CD-ROM to the top of the **Predefined Boot Sequence** list. These steps enable the virtual machine to start the installation process. Clicking the **OK** button will run the virtual machine with the specified settings.

> **Important**
> Remember that **Run Once** settings are used until the virtual machine is completely shut down. If the virtual machine is rebooted, it will continue to use the **Run Once** settings.

Manual Installation of Guest Drivers and Agents

Guest drivers and agents are tools that are installed in a guest operating system to improve management and performance of virtual machines. A *guest driver* provides a paravirtualized device driver that uses interfaces presented by the hypervisor, rather than standard device drivers, to improve performance. A *guest agent* typically provides information about the guest, and notifications about its status, from the guest operating system to the Red Hat Virtualization environment. One of the key features of the guest agent is its ability to monitor the usage of resources and to gracefully shut down or reboot virtual machines, using the **VM Portal** or the **Administration Portal**.

It is a good practice to install Red Hat Virtualization guest agents and drivers for Red Hat Enterprise Linux and Microsoft Windows virtual machines. Guest agents and drivers need to be installed on each virtual machine that requires this functionality.

The table below describes the different guest drivers available for Red Hat Virtualization guests. Not all drivers are available for all supported operating systems.

Driver	Description
virtio-net	Paravirtualized network driver for enhancing performance of network interfaces.
virtio-block	Paravirtualized HDD driver for increased I/O performance. Optimizes communication and coordination between the guest and hypervisor.
virtio-scsi	Paravirtualized iSCSI HDD driver provides support for adding hundreds of devices, and uses the standard SCSI device naming scheme.
virtio-serial	Provides support for multiple serial ports to improve performance, for faster communication between the guest and host.
virtio-balloon	Controls the amount of memory a guest can actually access. Optimizes memory overcommitment.
qxl	This paravirtualized display driver reduces CPU usage on the host and provides better performance.

When a virtual machine is started by RHV-M, it uses the guest agent on the virtual machine to gather information, such as the IP address for the virtual machine. RHV-M also tries to use the guest agent to communicate with the virtual machine to shut it down gracefully.

On Red Hat Enterprise Linux, this communication is done by the **`qemu-guest-agent`** service. This service is provided by the *qemu-guest-agent* package, and should be installed by default on a Red Hat Enterprise Linux system. If the package is not installed by default, it can always be installed later.

On Windows, install the RHV Agent as part of the RHEV-Tools installation. These are the available guest agents and tools:

Name	Description
ovirt-guest-agent-common	Allows Red Hat Virtualization Manager to execute specific commands, and to receive guest internal events or information.
spice-agent	Supports multiple monitors, and reduces the bandwidth usage over wide area networks. It also enables cut and paste operations for text and images between the guest and client.

Name	Description
rhev-sso	Desktop agent that enables users to automatically log in to their virtual machines.

Installing the Guest Agents on Red Hat Enterprise Linux

On Red Hat Enterprise Linux virtual machines, the Red Hat Virtualization guest agents are installed using the *qemu-guest-agent* package. This package is installed by default, even in a minimal installation of Red Hat Enterprise Linux 7 or Red Hat Enterprise Linux 8. Virtualization drivers are also installed by default in RHEL.

After the virtual machine is installed, use the **`subscription-manager`** command to register the system, grant it entitlements by attaching it to a subscription, and enable the desired repositories. Use either the **`yum`** command or the **`rpm`** command to verify the *qemu-guest-agent* package is installed, and install it if necessary. Once the *qemu-guest-agent* package is installed, use the **`systemctl`** command to start and enable the **`qemu-guest-agent`** service.

Red Hat Virtualization Manager will start receiving additional usage information from the guest agent running on the virtual machine. You can check this by logging into the Administration Portal, clicking **Compute**, and then selecting **Virtual Machines**. After a few minutes, the main virtual machines page should display the IP addresses and FQDN for the virtual machine. Additional information collected by the agent can be seen by clicking on the name of the virtual machine, and then exploring tabs such as **General**, **Network Interfaces**, and **Guest Info**.

> **Note**
> In the past, both the *ovirt-guest-agent-common* package and the *qemu-guest-agent* package were required on a Red Hat Enterprise Linux virtual machine. This is no longer the case.

Installing Guest Agents and Drivers on Windows

One of the best ways to improve the performance of Microsoft Windows guests is to use paravirtualized devices and drivers for KVM in the guests. This provides close to bare-metal performance (up to 95%).

On Windows virtual machines, the Red Hat Virtualization guest agents and drivers are installed using the **`rhv-tools-setup.iso`** ISO file. The ISO is available in the *rhv-tools-setup-iso* package provided by the **`rhel-7-server-rhv-4.3-manager-rpms`** repository.

Once the *rhv-tools-setup-iso* package is installed, the **`rhv-tools-setup.iso`** file can be uploaded to a data storage domain from the **`/usr/share/rhv-guest-tools-iso/`** directory. New versions of the **`rhv-tools-setup.iso`** file must be manually attached to Windows virtual machines to update the tools and drivers.

The **`rhv-tools-setup.iso`** file can be attached to a Windows virtual machine by following the same basic procedure used to install a Red Hat Enterprise Linux guest from an ISO file. However, the Windows virtual machine should already be installed. In addition, after you attach the **`rhv-tools-setup.iso`** ISO in the CD drive, you will not adjust the predefined boot sequence. The Windows virtual machine should boot from its hard drive, rather than attempting to boot from the CD drive.

After logging in to the Windows virtual machine, the **`rhv-tools-setup.iso`** ISO will be accessible from the CD-ROM drive. Running the **`RHEV-toolsSetup`** file launches the **RHEV-**

Tools InstallShield Wizard window. Decide which components should be installed, and then start the installation. When the installation completes, restart the virtual machine in order to apply the changes.

Cloning a Virtual Machine

Another way to create a virtual machine is to *clone* an existing one. A clone is a copy of a virtual machine created on new virtual hardware. The clone is an exact copy of the disk image from the original virtual machine, which includes configuration settings, logs, and other data on that image.

To clone a virtual machine from an existing virtual machine, start by selecting the virtual machine that you wish to clone. Since a clone can only be made when a virtual machine is in a stopped state, shut down the virtual machine if it is running. With the row containing the virtual machine highlighted, select the **Clone VM** menu entry from the three vertical dot menu displayed to the right of the **Create Snapshot** button.

After providing a name for the clone, the process to create the clone begins. While the clone is being created, the disk images for both the original virtual machine and the clone will be in a locked state. You will not be able to start either the original virtual machine or the clone until the status of the disk images changes to **OK**. Once the clone is complete, and the status of the images updates, both of the virtual machines are ready to run.

Be aware that attempting to run both the original virtual machine and the cloned virtual machine at the same time may cause problems. Remember, a cloned virtual machine starts with all of the data from the original virtual machine, including logs, SSH keys, and other unique information. This may not be desirable if you are trying to create a new virtual machine with a *similar configuration*, rather than an *exact copy*.

As an alternative, use the virtual machine to create a sealed *template* that has been cleared of unique data, and then create virtual machines from that template. Creating and using templates is covered later in this chapter.

References

Certified Guest Operating Systems in Red Hat OpenStack Platform and Red Hat Enterprise Virtualization
https://access.redhat.com/articles/973163

More information is available in the "Installing Red Hat Enterprise Linux Virtual Machines" chapter in the *Virtual Machine Management Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#Installing_Red_Hat_Enterprise_Linux_Virtual_Machines

Additional information is available in the "Installing Windows Virtual Machines" chapter in the *Virtual Machine Management Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#chap-Installing_Windows_Virtual_Machines

Detailed information on **New Virtual Machine** window settings are documented in the "Reference: Settings in Administration Portal and VM Portal Windows" appendix to the *Virtual Machine Management Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#appe-Reference_Settings_in_Administration_Portal_and_User_Portal_Windows

Guided Exercise

Installing Virtual Machines

In this lab, you will install a virtual machine in your Red Hat Virtualization environment.

Outcomes

You should be able to install a new virtual machine, associate that machine with available logical networks, and install the qemu-guest-agent package.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab vms-install start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab vms-install start
```

1. Download the RHEL 7.6 boot ISO file to the **/home/student/Downloads** directory on **workstation**, in preparation for uploading the file to RHV-M. You will use this boot ISO to initiate the installation of a RHEL 7.6 virtual machine.

```
[student@workstation ~]$ wget -P ~/Downloads \
http://materials.example.com/rhel-server-7.6-x86_64-boot.iso
...output omitted...
HTTP request sent, awaiting response... 200 OK
Length: 571473920 (545M) [application/octet-stream]
Saving to: '/home/student/Downloads/rhel-server-7.6-x86_64-boot.iso'
...output omitted...
```

2. Use the RHV-M Administration Portal to upload the RHEL 7.6 boot ISO file.

   2.1. On **workstation**, open Firefox and navigate to `https://rhvm.lab.example.com`.

   2.2. Click the **Administration Portal** link, and log in using **admin** as the username, **redhat** as the password, and **internal** as the profile.

   2.3. View existing disks by clicking **Storage** in the menu, and then selecting **Disks**.

   2.4. Open the Upload Image wizard by clicking **Upload** in the upper-right corner, and then selecting **Start**.

   2.5. Select the RHEL 7.6 boot ISO by clicking the **Choose File** button. In the **File Upload** window, click **Downloads** in the menu and then double-click the **rhel-server-7.6-x86_64-boot.iso** file to choose that file. Locate the **rhel-server-7.6-x86_64-boot.iso** if it was downloaded to a different location.

2.6. In the **Upload Image** window, verify that Data Center is set to **`development`** and Storage Domain is set to **`iscsi-data`** . Accept the default settings for everything else.

2.7. In a previous guided exercise, you should have downloaded the ovirt-engine certificate. Click the **Test Connection** button to verify this. If clicking the **Test Connection** button returns a green success box, then you are ready to upload. If clicking the **Test Connection** button returns an orange warning box, then click the ovirt-engine certificate link within the warning box. Select the check box next to **Trust this CA to identify websites**, and then click the **OK** button. Next, click the **Test Connection** button again. It should return a green success box.

> **Important**
> If you forget to enable the check box next to **Trust this CA to identify websites**, the following procedure will return you to the applicable window:
>
> 1. Open **Preferences for Firefox**, and then select **Privacy & Security** in the left menu.
> 2. Scroll down to the **Security** section and click the **View Certificates...** button.
> 3. In the **Certificate Manager** window, scroll down to lab.example.com, click rhvm.lab.example.com.34088 to highlight it, and then click the **Delete or Distrust** button.
> 4. Back on the **Preferences** tab for **Privacy & Security**, scroll up to the **Cookies and Site Data** section, and then click the **Clear Data...** button.
> 5. Accept the default selections and click the **Clear** button. Confirm your choice by clicking the **Clear Now** button in the new window that displays.

2.8. Click the **OK** button to start the upload process.

3. Verify that the ISO uploaded successfully.

3.1. The **Disks** table should list a disk with an **Alias** of **`rhel-server-7.6-x86_64-boot.iso`**. Within a few minutes, the disk **Status** will update to OK.

4. Navigate to **Compute** in the menu bar, and then select **Virtual Machines** to see a list of existing virtual machines.

5. Create a new virtual machine named **`rhel-vm1`**, in the **`clusterone`** cluster of the **`development`** data center, according to the following specifications:

Install the virtual machine with Red Hat Enterprise Linux 7. Create a **`Small`** instance customized to use two CPUs. Optimize the virtual machine for the **`Server`** type. Create a disk image that is 3 GB in size. Associate the network interface for this virtual machine with the **`vm-net`** logical network. Use the Advanced option to set the memory balloon device enabled.

5.1. On the **Compute >> Virtual Machines** page, click the **New** button. In the **New Virtual Machine** window that opens, select the following settings:

- In the **Cluster** section, choose the **clusterone** cluster.

- As the **Operating System**, select **Red Hat Enterprise Linux 7.x x64**.
- Click the **Instance Type** drop-down list and choose **Small**.
- Click the **Optimized for** drop-down list and choose **Server**. Note that the **Instance Type** previously selected changes from **Small** to **Custom**.
- In the **Name** field, type **`rhel-vm1`** as the name for the virtual machine.
- In the **Description** field, type **`RHEL Guest using development data center`**.

> **Note**
> Selecting an instance type populates memory and CPU values for the virtual machine. These values are visible from the System tab when viewing advanced options. Although choosing either Server or High Performance from the Optimized for menu will switch the Instance Type menu back to Custom, the memory and CPU values for the instance type that you selected will remain the same.

5.2. In the **New Virtual Machine** window, configure a disk for the new VM. On the **Instance Images** line, click the **Create** button. Specify the **Size** of the image as **3** GB. Leave all other options with their default values and confirm by clicking the **OK** button.

Notice that there is a new entry under the **Instance Images** line. This entry confirms that the new image, defined in the previous step, will be created for the new virtual machine, once accepted.

5.3. Assign the logical network **vm-net** to **`nic1`**.

In the bottom part of the **New Virtual Machine** window, choose a network interface by clicking the **Please select an item** list next to the **`nic1`** network card. From the list, choose the **vm-net/vm-net** logical network.

5.4. Click the **Show Advanced Options** button.

Click the **System** tab to access the CPU and memory settings for this virtual machine.

Change the value in the **Total Virtual CPUs** line to **2**.

Click the **Resource Allocation** tab to access settings for memory allocation.

Make sure that the check box for **Memory Balloon Device Enabled** option is enabled.

5.5. To confirm the creation of this virtual machine, click the **OK** button.

Notice that on the list of available virtual machines, the new **`rhel-vm1`** virtual machine is displayed.

6. Temporarily modify the configuration of the virtual machine so that it boots using the Red Hat Enterprise Linux installation image in the virtual CD-ROM/DVD-ROM device, and then boot it.

6.1. Click the name of the **`rhel-vm1`** virtual machine. In the **Run** drop-down list, select **Run Once**.

6.2. Click the **+** icon next to **Boot Options** to open the **Boot Options** window.

6.3. Select the check box next to **Attach CD**. From the drop-down list of available ISO files, choose the Red Hat Enterprise Linux boot ISO **`rhel-server-7.6-x86_64-boot.iso`**. This "inserts" the ISO file into the virtual CD-ROM/DVD-ROM drive.

6.4. In the **Predefined Boot Sequence** list, choose the **CD-ROM** by clicking on it. With the **CD-ROM** highlighted, click the **Up** button once, to bring the **CD-ROM** drive to the top of the **Boot Sequence** list.

6.5. Click the **OK** button to confirm your changes and to boot the virtual machine from the Red Hat Enterprise Linux boot ISO.

7. Open the virtual machine console, and begin a Kickstart installation of the virtual machine.

7.1. Once the virtual machine has started, and the console button becomes available, click the **Console** button to start the Red Hat Enterprise Linux installation. Open the **`console.vv`** file using Remote Viewer by clicking the **OK** button.

7.2. When you hover over the console window, a prompt displays to either deny or allow the Remote Viewer to inhibit shortcuts. Click the **Allow** button.

7.3. It may take a minute to get to the installation screen. When it displays, highlight **Install Red Hat Enterprise Linux 7.6**, and then press the **Tab** key to edit the installer options.

7.4. The editor should open with the cursor automatically positioned at the end of the existing kernel arguments. At the end of the kernel command line, add a space and the argument **`inst.ks=http://materials.example.com/static/small-7.6.cfg`** to specify the location of your Kickstart file.

7.5. Press **`Enter`** to start the Kickstart installation of the virtual machine.

8. Watch for the installation to complete. Please be patient as the installation may take up to 10 minutes to complete. After the installation completes, the virtual machine automatically reboots from the CD. Do not worry if it looks like the virtual machine wants to proceed with another installation.

9. Close the console and then power off the virtual machine by clicking the **Shutdown** drop-down menu and then selecting **Power Off**. Click the **OK** button to confirm that you want to power off the machine.

10. To test the **`rhel-vm1`** virtual machine, power it on by clicking the **Run** button. In some cases the **Run** button may be accessible before the virtual machines has fully shut down. If clicking the **Run** button results in an error, simply wait a few seconds and then try again.

11. Once available, open the console by clicking the **Console** button.

12. When you hover over the console window, you are prompted to either deny or allow the Remote Viewer to inhibit shortcuts. Click the **Allow** button.

13. After the virtual machine starts, log in as the **`root`** user with a password of **`redhat`**.

14. In the following steps, you install the **`qemu-guest-agent`** package inside the virtual machine. The host name displayed in the example prompts is likely different than what you see on your screen.

14.1. To install the necessary software packages on your **rhel-vm1** virtual machine, ensure that the system is registered with Red Hat Subscription Manager and has the correct entitlements and Yum repositories enabled.

In the classroom environment, this step is different because the classroom does not have access to the Content Distribution Network or a Red Hat Satellite server. Instead, local Yum repositories have been provided, which contain the correct packages.

Download the **rhel_dvd.repo** file from `http://materials.example.com/yum.repos.d/rhel_dvd.repo`, and place it in the **/etc/yum.repos.d/** directory to enable those repositories.

```
[root@rhel-vm1 ~]# curl http://materials.example.com/yum.repos.d/rhel_dvd.repo \
 -o /etc/yum.repos.d/rhel_dvd.repo
```

14.2. Check to see if the *qemu-guest-agent* package is already installed.

```
[root@rhel-vm1 ~]# rpm -q qemu-guest-agent
qemu-guest-agent-2.12.0-2.el7.x86_64
```

14.3. If the *qemu-guest-agent* package is not already installed, install it, and its dependencies, using the **yum** command.

```
[root@rhel-vm1 ~]# yum -y install qemu-guest-agent
```

14.4. Start and enable the **qemu** guest agent by issuing the **systemctl enable --now qemu-guest-agent** command.

```
[root@rhel-vm1 ~]# systemctl enable --now qemu-guest-agent
```

15. Close the console, but leave the virtual machine running. Return to the **Compute >> Virtual Machines** page and wait until the information from the agent reaches RHV-M. After some time, the columns for **IP Address** and **FQDN** will display information for the **rhel-vm1** virtual machine.

Finish

On **workstation**, run the **lab vms-install finish** script to complete this exercise.

```
[student@workstation ~]$ lab vms-install finish
```

This concludes the guided exercise.

Editing Virtual Machine Hardware

Objectives

After completing this section, you should be able to change the configuration of virtual hardware for an existing virtual machine.

Editing a Virtual Machine

To persistently change the settings of a virtual machine, navigate to the list of virtual machines by clicking **Compute** in the menu bar, and then selecting **Virtual Machines**. Right-click the virtual machine and select **Edit** from the menu. This brings up the **Edit Virtual Machine** window.

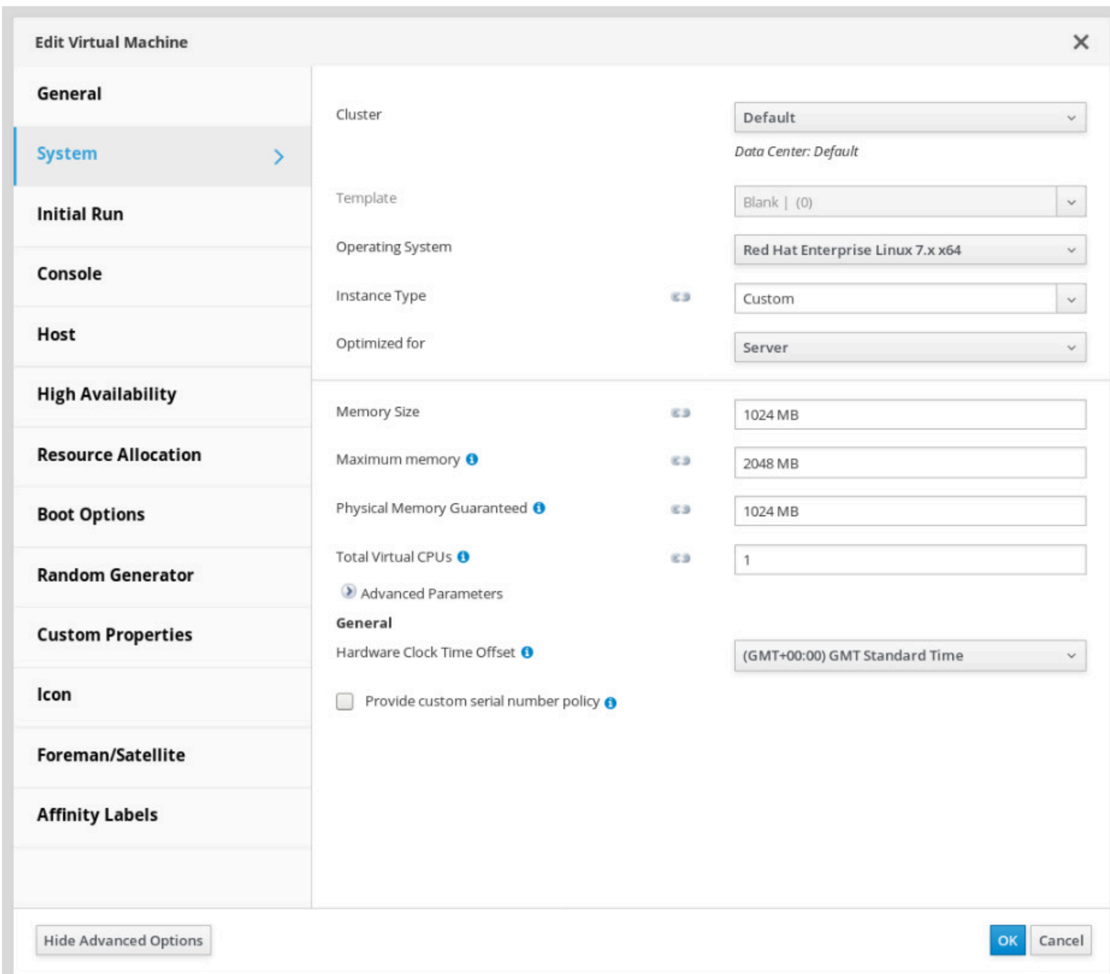

Figure 8.2: Edit Virtual Machine window

The **Edit Virtual Machine** window is nearly identical to the **New Virtual Machine** window, discussed earlier. In order to see all of the settings, you will need to click the **Show Advanced Options** button. Please note that some changes require that the virtual machine is shut down and restarted. You will be alerted if a shutdown is required; when viewing the list of virtual machines, the impacted virtual machine will be marked with an orange pending changes icon.

The following section describes some of the main tabs in the **New Virtual Machine** window, and the types of changes you might make within them.

System

Settings found in the **System** tab allow you to change memory, virtual CPUs, and time zone. The **Instance Type** menu populates the memory and CPU fields every time you change the selection, but you can always customize from there.

- **Memory Size**: Increasing the **Memory Size** or the **Physical Memory Guaranteed** can be done on a running virtual machine. Reducing the **Memory Size**, reducing or increasing the **Maximum Memory**, and reducing the **Physical Memory Guaranteed** requires restarting the virtual machine.

- **Virtual Sockets**: You can hot-plug vCPUs, but the operating system must support this feature. You can also unplug vCPUs, but only if they were hot-plugged. You cannot unplug more vCPUs than the virtual machine had when it was created.

Initial Run

These settings can adjust how the machine boots initially, using either **`Cloud-Init`** or **`Sysprep`**. The **Initial Run** tab will be discussed later in this chapter.

Console

These settings affect how the virtual machine is accessed, either through the Administration Portal or the VM Portal. You can also specify the keyboard layout, whether or not USB support is enabled, whether or not to enable single sign-on, and more from the **Console** tab.

Host

There may be cases where you need a virtual machine to run on a specific host. If you run a virtual machine on a specific host, you can disable automatic migration, but still leave manual migration as an available option. For example, suppose two out of five hosts in a cluster have extra capabilities. You may need a virtual machine to run on one of them. You could start the virtual machine on one of the hosts, and then migrate it to the other host, if necessary. Specifying that a virtual machine should run on a particular host, and deciding if a virtual machine should be migrated, are some of the types of changes that can be made from the **Hosts** tab.

High Availability

You can flag a virtual machine as being highly available on the **High Availability** tab. Additionally, you can specify a priority for the run/migration queue on the machine. More information regarding high availability is presented later in this course.

Resource Allocation

You can make adjustments to CPU and memory allocation from the **Resource Allocation** tab. When creating a new virtual machine from a template, you can specify if the the virtual machine will be a clone, or if the virtual machine will have a thin storage overlay. A cloned virtual machine can exist independently of the template used to create it, but a virtual machine with a thin storage overlay continues to use the template from which it was created. More information about templates is presented later in this chapter.

Boot Options

These settings allow you to persistently specify a boot order. For example, you can persistently attach an ISO to the CD drive of the virtual machine, or pass additional information to GRUB. The **Boot Options** tab also allows you to pass additional information to GRUB.

Configuring Virtual NUMA

In the Administration Portal, you can configure virtual non-uniform memory access (NUMA) nodes on a virtual machine, and pin them to physical NUMA nodes on one or more hosts. This can be done from the **Host** tab when viewing **Show Advanced Options** in either the **New Virtual Machine** window, or the **Edit Virtual Machine** window. The default policy is to schedule and run virtual machines on any available resources on the host. As a result, resources backing a large virtual machine that cannot fit within a single host socket could be spread out across multiple NUMA nodes. Over time these resources may be moved around, leading to poor and unpredictable performance. Configure and pin virtual NUMA nodes to avoid this outcome and improve performance.

Hot-Plugging Network Interfaces and Disks

RHV allow adding network interfaces and disks to running virtual machines. This is referred to as hot-plugging. For network interfaces, it is possible to add a network interface to a virtual machine and then plug it in, or unplug it, as necessary. A plugged in network interface is accessible to the virtual machine, while an unplugged one is not. One way to view network interfaces from a Linux virtual machine is using the **`ip link`** command.

Additional hard disks can be added in a similar way. A new hard disk is accessible to the virtual machine when it is activated. A deactivated hard disk is still associated with a virtual machine, but it is not accessible by the virtual machine. One way to view hard disks from a Linux virtual machine is using the **`lsblk`** command.

Care must be taken when removing network interfaces or disks from running virtual machines to ensure that they are not being used.

Adding Network Interfaces

Virtual machines connect network interfaces to logical networks in your RHV environment. You can create multiple network interfaces on your virtual machines, each of which can be associated with a specific logical network.

> **Important**
> For a logical network to be available to a virtual machine, it must be configured as a **VM Network** by the cluster.

A quick way of adding a new network interface is to use the same basic procedure that you used when creating the virtual machine. To add a new network interface, edit the virtual machine and go to the **General** tab. The **Instantiate VM network interfaces by picking a vNIC profile** section allows you to add new network interfaces or remove existing ones, using the **+** and **-** buttons. You cannot specifically plug or unplug a network interface from the **General** tab. New network interfaces added from the **General** tab are automatically plugged in and accessible to the virtual machine.

More options are available when you view virtual machine details by clicking the name of an individual virtual machine listed on the **Compute >> Virtual Machines** page. From the details page, you can click the **Network Interfaces** tab. Here you can either add a new network interface or select an existing network interface to edit. When adding a new network interface, in addition to selecting a VM Network profile, you can select a driver type, and you can customize the MAC address. When adding or editing a network interface, you can specify the link state as being **Up** or **Down**, and the card status as being either **Plugged** or **Unplugged**. You can edit a network interface on a running machine, but depending on the changes you make, the virtual machine might need to be restarted.

Adding Disks

New disks can be added to a virtual machine. As with network interfaces, there is a quick method and a more advanced method. The quick method uses the same basic procedure that you used when creating the virtual machine. To add a new disk, edit the virtual machine and go to the **General** tab. The **Instance Images** section allows you to add new disks or remove existing disks, using the **+** and **-** buttons. Existing disks can be modified using the **Edit** button.

More options are available when you view virtual machine details by clicking the name of an individual virtual machine listed on the **Compute >> Virtual Machines** page. From the details page, click the **Disks** tab. Additional configuration fields are available when you choose to add or edit a disk. To deactivate an individual disk, highlight the row containing the disk by clicking an empty cell in the row. With the row highlighted, click the three vertical dot menu, located to the right of the **Remove** button, and select **Deactivate**. A deactivated disk can be removed. When you attempt to remove a disk, you will be prompted to confirm you action. You must also decide if the disk should be removed permanently. A disk which is not removed permanently can be attached to a different virtual machine.

> **Warning**
> A bootable disk may contain the OS icon, or it may have **`(boot)`** as descriptive text. Deleting the boot disk for a virtual machine will render it non-bootable.

> **References**
> Further information is available in the "Editing Virtual Machines" chapter of the *Virtual Machine Management Guide* for Red Hat Virtualization at
> https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#chap-Editing_Virtual_Machines
>
> Details on **Edit Virtual Machine** window settings are documented in the "Reference: Settings in Administration Portal and VM Portal Windows" appendix to the *Virtual Machine Management Guide* for Red Hat Virtualization at
> https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#appe-Reference_Settings_in_Administration_Portal_and_User_Portal_Windows

Guided Exercise

Editing Virtual Machine Hardware

In this exercise, you will make changes to the configuration of an existing Red Hat Enterprise Linux-based virtual machine.

Outcomes

You should be able to change the amount of RAM and the number of CPUs on one of your virtual machines.

Before You Begin

Log in to **`workstation`** as **`student`** using **`student`** as the password.

On **`workstation`**, run the **`lab vms-edit start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working. Wait for the start script to finish before editing the virtual machine, as the start script will undo changes made before it completes.

```
[student@workstation ~]$ lab vms-edit start
```

1. On **`workstation`**, open Firefox and access the RHV-M Administration Portal, using the `http://rhvm.lab.example.com` URL. Log in using **`admin`** as the username, **`redhat`** as the password, and **`internal`** as the profile.

2. Navigate to **Virtual Machines** by clicking on **Compute** in the menu bar, and then selecting **Virtual Machines**.

3. View details for the **`rhel-vm1`** virtual machine by clicking its name. In the **General** tab, observe how **Defined Memory** is set to 2048 MB and **Number of CPU Cores** is set to 2. The extra information of (2:1:1) indicates that the virtual machine has 2 virtual sockets, with 1 core per virtual socket, and 1 thread per core.

4. If your **`rhel-vm1`** virtual machine is still running, click the **Shutdown** button to shut down the virtual machine. Confirm the shutdown by clicking the **OK** button.

5. Wait until the virtual machine status changes to **`Down`**. Click the **Edit** button to edit the properties of the virtual machine.

6. In the **Edit Virtual Machine** window, click the **Show Advanced Options** button and verify that you can access the advanced options. Click the **System** tab.

7. Change the amount of RAM available to the machine by modifying the **Memory Size** text field to a value of **`512 MB`**. Change the maximum amount of RAM available to the machine by modifying the **Maximum memory** text field to a value of **`1024 MB`**.

8. Change the number of CPUs available to the machine by modifying the **Total Virtual CPUs** text field to a value of **`1`**.

9. Accept the changes by clicking **OK**.

10. Observe the changed values that you specified in the **General** tab. **Defined Memory** should have a value of 512 MB and **Number of CPU Cores** should have a value of 1.

11. Start the **rhel-vm1** virtual machine by clicking the **Run** button.

12. Once it is active, click the console button to access the **rhel-vm1** virtual machine console. Click the **OK** button to open the **console.vv** file using Remote Viewer.

13. Log in to the virtual machine using the **root** user account and the password **redhat**.

14. Issue the **dmidecode** command to examine the hardware components, and confirm that the amount of RAM available to the system has changed to **512** MB.

```
[root@rhel-vm1 ~]# dmidecode | grep -A10 '^Memory Device'
Memory Device
 Array Handle: 0x1000
 Error Information Handle: Not Provided
 Total Width: Unknown
 Data Width: Unknown
 Size: 512 MB
 Form Factor: DIMM
 Set: None
 Locator: DIMM 0
 Bank Locator: Not Specified
 Type: RAM
```

15. Issue the **lscpu** command to see that the number of CPUs has changed to **1**.

```
[root@rhel-vm1 ~]# lscpu | grep '^CPU(s)'
CPU(s):                 1
```

16. Close the virtual machine console and log out from the **Administration Portal**.

Finish

On **workstation**, run the **lab vms-edit finish** script to complete this exercise.

```
[student@workstation ~]$ lab vms-edit finish
```

This concludes the guided exercise.

Creating and Deploying Virtual Machines with Templates

Objective

After completing this section, you should be able to create a Linux virtual machine template and deploy new virtual machines using that template.

Rapid Deployments Using Templates

A *template* is a copy of a preconfigured virtual machine, used to simplify the subsequent, repeated creation of similar virtual machines. Templates capture the installed software, software configuration, and hardware configuration of the original virtual machine. When administrators need to deploy multiple, mostly identical machines, it can be beneficial to use templates instead of doing multiple installations.

A template can be seen as a *bit-for-bit* copy of the original disk image, and bears many resemblances to imaging a machine with tools like Clonezilla or Ghost. Just like an automated installation, such as Kickstart, this helps maintain consistency across machines, making management and troubleshooting easier.

When working with virtual machines, basing machines on a template can also help reduce memory usage. Since machines based on the same template will be very similar, Kernel Same-page Merging (KSM) has a higher chance of finding duplicate memory pages on those machines that can be merged in physical memory.

> **Warning**
> One of the challenges involved in using templates is making sure that no identifying information, such as references to MAC addresses or SSL certificates, are left behind, because issues may occur when multiple machines are deployed from the same template. This process is called *sealing* the image.

Creating a Template from a Virtual Machine

The first step to create a template is to install a new virtual machine. This can be done manually, or you may automate the installation using a tool such as Kickstart. After the virtual machine finishes installing, log into the virtual machine and perform any additional customization that is needed. Again, you can add customizations manually, or use a tool such as Ansible to do the configuration for you. Your goal is to get the virtual machine to a state where everything works as expected for the desired function when the VM is started.

Once the virtual machine has been configured, remove all information unique to the virtual machine. This is referred to as sealing the image. Unique information includes hardware information specific to the virtual machine, such as MAC addresses; unique system configurations, such as the host name and static IP addresses; and possibly logs and other data. Depending on the operating system of the virtual machine, you may need to perform these steps manually, or there may be tools available to seal the image for you. For example, Linux virtual machines can typically use the **`virt-sysprep`** utility on a stopped virtual machine. A running Windows virtual machine typically allows using **`sysprep.exe`** to initiate a System Out-of-Box-Experience (OOBE). The process for sealing a Windows virtual machine concludes when you shut down the virtual machine. The RHV-M Administration Portal also provides an option where it will seal the virtual machine image prior to creating a template. Be aware that this option does not work for all operating systems.

One important thing to keep in mind is that many items that were stripped out of the virtual machine during the sealing process are recreated when the virtual machine boots up for the first time. As such, once a virtual machine has been sealed, do not start it until after you have made a template. If you accidentally start the virtual machine before you create the template, you will have to go through the process of sealing the image again.

A new template can be created from the RHV-M Administration Portal by clicking **Compute** in the menu bar, and then selecting **Virtual Machines**. Highlight the entire row of the virtual machine that will be used to create the template by clicking an empty cell in that row. With the virtual machine row highlighted, click the three vertical dot menu located to the right of the **Create Snapshot** button, and select **Make Template**. You will be prompted to specify details for the template in the **New Template** window, and then the template is created.

After a template has been created, it is always a good idea to test the template by creating a new virtual machine based on the template.

The following window is presented when creating a new template.

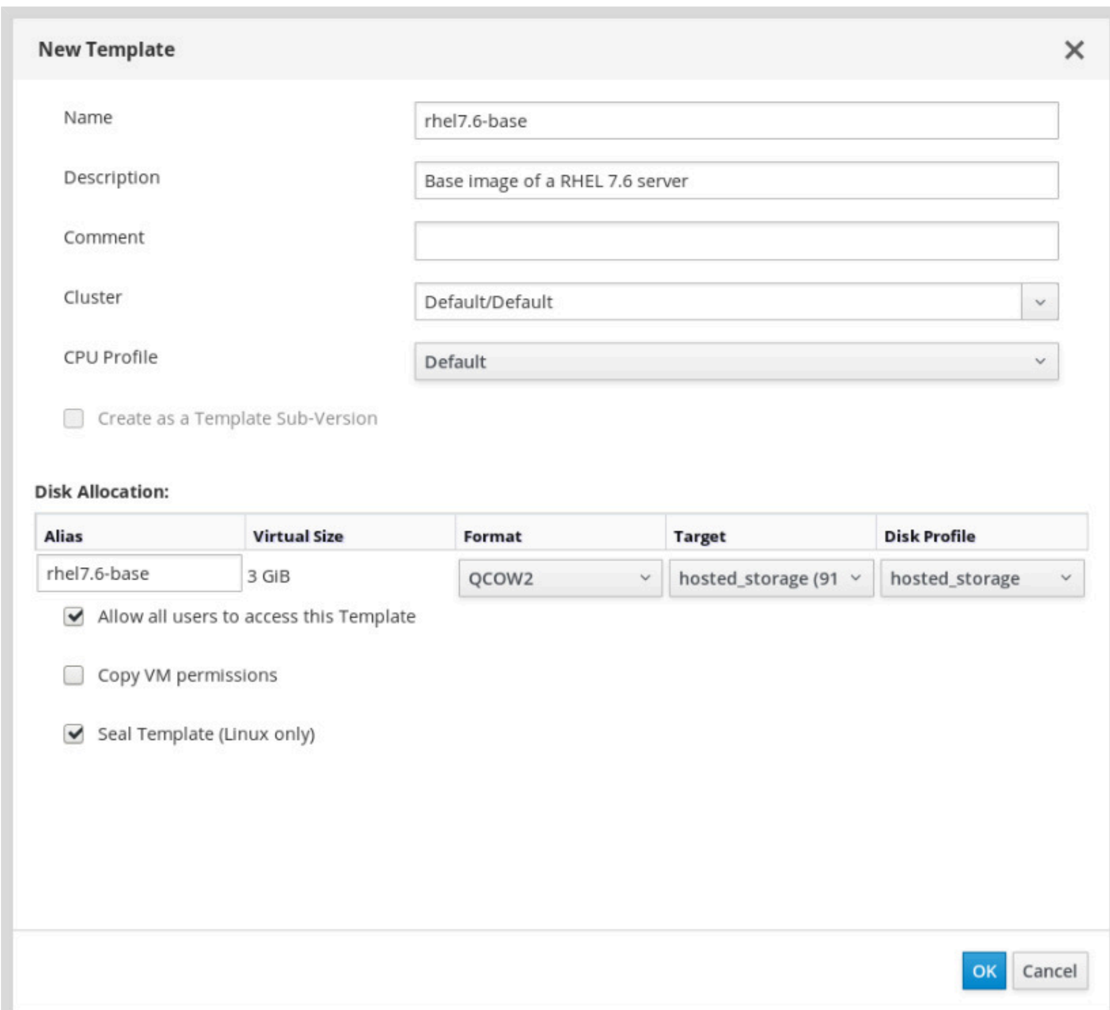

Figure 8.3: New Template window

The settings for the **New Template** window are described below:

Name
The name of the template. This name is listed on the **Templates** tab in the Administration Portal. It must be a unique name, and can include any combination of uppercase and lowercase letters, numbers, hyphens, or underscores.

Description
A limited description of the template. This field is recommended but not mandatory.

Comment
A field for adding plain text, human-readable comments about the template. For example, you might use this to elaborate on the purpose or even the usage of a template.

Cluster
The cluster with which the template is associated. By default, this is the same as the original virtual machine cluster. However, any cluster within the same data center can be selected.

CPU Profile
An advanced feature used to limit how much CPU capacity a virtual machine can use on a host. The default setting imposes no limit. Use **CPU Profile** to define the maximum amount of processing capability a virtual machine can access on its host. This is expressed as a percent of the total processing capability available on a host. See the *Administration Guide* for more information.

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 current actual size of the virtual disk used by the template. This value cannot be edited, and is provided for reference only.
- **Format**: The format of the disks that the template should use. A QCOW2 format always implies that a disk is thin provisioned. The RAW format in file storage implies thin provisioned, while RAW on block storage implies preallocated virtual disks.
- **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. Any storage domain in the cluster can be selected.

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 permissions set on the source virtual machine to the template.

Seal Template
Seals the template using the **`virt-sysprep`** command. Information on the actions completed by this command are in the "virt-sysprep Operations" appendix of the Red Hat Virtualization *Virtual Machine Management Guide* on the Red Hat Customer Portal. This option only works for Linux virtual machines.

If you are not able to use this option, you must manually seal the virtual machine image before using this window to create the template.

Using a Template to Create a New Virtual Machine

Once a template has been created from a virtual machine, administrators can create new virtual machines using that template. Consider this a test to validate that virtual machines created from a template do what you expect them to do. If the virtual machine used to create the template was not sealed, or was not sealed correctly, virtual machines created from the template could experience problems. If the first virtual machine created from the template works as expected, you know that additional virtual machines will also work.

You can create a new virtual machine from a template using the RHV-M Administration Portal. The process for creating a new virtual machine from a template is very similar to the process used for creating any other virtual machine.

There are a few significant differences:

- From the **New Virtual Machine** window, select a template from the **Template** menu. Only templates associated with the current data center, based on the selected cluster, are listed.
- When you select a template, many fields within the **New Virtual Machine** window are populated using the template. On the **General** tab, this includes **Operating System**, **Instance Type**, **Optimized for**, and network interfaces. Additionally, you cannot attach or create a hard disk, as this come from the template.
- Clicking the **Show Advanced Options** button reveals additional options. Although much of this information is populated based on the selected template, you can make adjustments as needed.
- The **Resource Allocation** tab has a section specifically related to templates. In the **Storage Allocation** section, you can specify how to provision the virtual disks for the new virtual machine.

  When choosing **Thin**, an overlay is used on top of the original template. This choice is more efficient for disk space, and the virtual machine is created more quickly, but the virtual machine cannot exist apart from the template. Editing a virtual machine which is still associated with a template will show the name of the template in the **Template** menu.

  When choosing **Clone**, the original template is cloned; the **Format** option becomes available to select either **Raw** or **QCOW2** for each disk. Once created, this type of virtual machine no longer needs the template, and an edit of the virtual machine will show **Blank** in the **Template** menu.
- Finally, after you create the new virtual machine, there is no need to perform an installation using **Run Once**. The virtual machine is already installed and should run normally. Be aware that depending on the operating system of the virtual machine, the initial boot may take longer than normal as items that were stripped out during the sealing process are recreated.

References

`virt-sysprep`(1) man page

Further information is available in the "Templates" chapter and the "**`virt-sysprep`** Operations" appendix of the *Virtual Machine Management Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#chap-Templates

Guided Exercise

Creating and Deploying Virtual Machines with Templates

In this exercise, you will create a template of a Red Hat Enterprise Linux virtual machine and use it to deploy a new virtual machine.

Outcomes

You should be able to create a template from an existing virtual machine (**rhel-vm1**) and use it to deploy a new virtual machine.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab vms-template start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab vms-template start
```

1. On **workstation**, open Firefox and using the `https://rhvm.lab.example.com` URL go to your RHV-M web interface. Click on the **Administration Portal** link and log in to the web interface as the **admin** user, with the **internal** profile, using **redhat** as the password.

2. Make sure that the **rhel-vm1** virtual machine is powered down.

    2.1. Navigate to **Virtual Machines** by clicking on **Compute** in the menu bar, and then selecting **Virtual Machines**.

    2.2. Highlight the row containing the **rhel-vm1** virtual machine by clicking an empty cell in that row. Confirm that the **rhel-vm1** virtual machine has a **Status** of Down. To the left of **rhel-vm1**, you should see a single red triangle pointing downwards. If the **rhel-vm1** virtual machine is not down, click the **Shudown** button and wait for the status to change to Down.

3. Create a template named **rhel-template** from the **rhel-vm1** virtual machine.

    3.1. With the **rhel-vm1** virtual machine row highlighted, click the three vertical dots to the right of the **Create Snapshot** button, and then select **Make Template**. The **New Template** window opens.

    3.2. In the **Name** text field, type the name **rhel-template** for the template.

    3.3. In the **Description** text field, type the description **RHEL 7.6 server template** for the template.

    3.4. In the **Alias** text field, type the name **rhel7-server** for the alias.

3.5. If it is not already selected, select the **Seal Template (Linux only)** check box.

3.6. Leave the other options with their default settings. Click the **OK** button to create the template.

3.7. Notice that the **`rhel-vm1`** virtual machine and all of its disks are locked. You should see an hour glass icon in the row for the **`rhel-vm1`** virtual machine. The template can take a couple of minutes to be prepared. Wait until the process of creating the template finishes. Template generation is complete when the virtual machine is released from its locked state.

4. Create a new virtual machine named **`rhel-vm2`**, based on the template that you created.

4.1. On the **Compute >> Virtual Machines** page, click the **New** button. The **New Virtual Machine** window displays.

4.2. In the **Cluster** menu, choose **clusterone**.

4.3. In the **Template** menu, choose **rhel-template**.

4.4. In the **Name** field, type the name for the virtual machine as **`rhel-vm2`**.

4.5. In the **Description** field, type the description for the virtual machine as **`RHEL Guest from a template`**.

4.6. Notice that the network configuration matches the one from the original **`rhel-vm1`** virtual machine, and that you are unable to create any new disk images.

4.7. Leave all other options as they are. Click the **OK** button to deploy the new template-based virtual machine.

4.8. Wait while the virtual machine is created. Once it becomes available, confirm that it is functional by starting the virtual machine and logging into it as the **`root`** user using **`redhat`** as the password.

Finish

On **`workstation`**, run the **`lab vms-template finish`** script to complete this exercise.

```
[student@workstation ~]$ lab vms-template finish
```

This concludes the guided exercise.

Automating Virtual Machine Configuration with cloud-init

Objectives

After completing this section, you should be able to prepare a template that includes **`cloud-init`**, and use that template to create a virtual machine.

cloud-init Overview

The **`cloud-init`** tool is used to automate the initial setup of virtual machines, such as configuring the host name, network interfaces, and authorized keys. It can be used to avoid conflicts on the network when provisioning virtual machines that have been deployed based on a template.

To use this tool, the *cloud-init* package must first be installed on the virtual machine. On a Red Hat Enterprise Linux 8 system, the *cloud-init* package is available in the **`rhel-8-for-x86_64-appstream-rpms`** repository. On a Red Hat Enterprise Linux 7 system, the *cloud-init* package is available in the **`rhel-7-server-rpms`** repository. Once installed, the **`cloud-init`** service starts during the boot process to search for configuration instructions.

Use the options in the **Run Once** window to provide instructions for the immediate boot process. You can persistently configure **`cloud-init`** to run every time the virtual machine boots by editing the virtual machine, and making changes to the **Initial Run** tab in the advanced options view. The same changes can be made to a template, so that any virtual machine created from the template will always run **`cloud-init`** at boot.

`cloud-init` can be used to automate the configuration of virtual machines in a variety of scenarios. Some uses include:

- *Customizing virtual machines using a standard template*: 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 that are based on this template.

- *Customizing virtual machines using "Initial Run"*: Administrators can use the **`cloud-init`** options in the **Initial Run** section of the **Run Once** window to initialize a virtual machine. This could be used to override settings set by a template.

Preparing the Template

As soon as the *cloud-init* package is installed on your Linux virtual machine, you can use that machine to create a template with **`cloud-init`** enabled.

Configure the template so that the advanced option **Initial Run** has the setting **Use Cloud-Init/Sysprep** selected. This enables additional configuration options for setting host name, time zone, authentication and network properties, and for running customer **`cloud-init`** scripts.

There are two easy ways to apply **Initial Run** settings to the template.

- The template inherits any settings from the **Initial Run** configuration of the original virtual machine, just like it inherits other characteristics of the virtual machine. However, this means you have to change the base virtual machine's settings and then create the template.

- You can create the template normally, and then use **Edit Template** to change the **Initial Run** settings for **`cloud-init`**. The original virtual machine will not have these settings applied, but machines created from the template will.

If the **`cloud-init`** settings are in the template, then when you create a new virtual machine, those **Initial Run** settings are applied to the virtual machine by default. You also have the option of overriding those settings from the **New Virtual Machine** window when you create the VM from the template.

The following graphic shows the **Initial Run** tab from the **Edit Template** window.

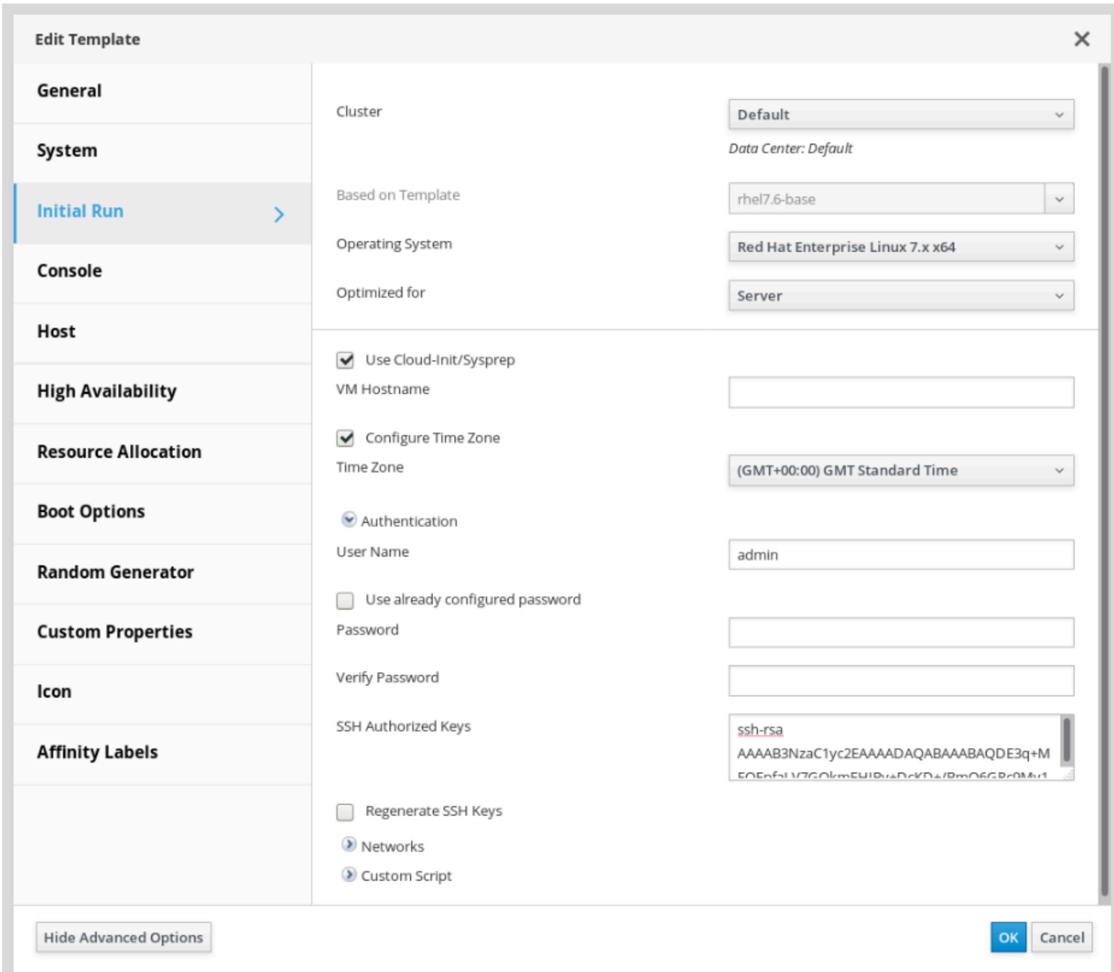

Figure 8.4: Edit Template window

The **Initial Run `cloud-init`** settings, displayed in the **Edit Template** window, are the same as those in the **Edit Virtual Machine** window, and the **Run Virtual Machine(s)** window, when you choose to make one-time changes with the **Run Once** menu option.

VM Hostname
: A static host name for the virtual machine is set when the machines boots. Set the initial run for an individual virtual machine, rather than as part of the template itself. Often, the host name is set in combination with configuring a static IP address, where the host name and IP address match DNS records.

Authentication
: Authentication allows you to configure a user with a password and public keys, authorized to allow SSH logins. This could create a new user, or reconfigure an existing one, in the virtual

machine. Be aware that this user (whether new or existing) is given full sudo access without needing to enter a sudo password. To create a regular user, use the **Custom Script** section.

Networks

Networks allows you to make networking changes. This includes specifying DNS information to be used in **/etc/resolv.conf** file, as well as configuring network interfaces to have either static or dynamic addresses.

Custom Script

Custom Script allows you to run a custom **cloud-init** script in cloud-config YAML format. While **cloud-init** uses YAML, and the syntax looks similar to Ansible, **cloud-init** is not Ansible. For example, the following custom script will create a user named **developer2** with a password of **redhat**. If you forget to add the **lock_passwd: false** line, or you accidentally use **lock_password** instead of **lock_passwd**, you cannot simply add the line and use the script again. You must either log in to the machine, delete the user, and then run the custom script again, or you must unlock the user manually.

The following portion of a **cloud-init** custom script creates a regular user, named **developer2**, with a password of **redhat**:

```
users:
  - name: developer2
    passwd: $6$l.uq5YSZ/aebb.SN$S/KjOZQFn.3bZcmlgBRGF7fIEefBPCHD.k46IW0dKx/
XK.I0DmZQBKGgCIxg7mykIIzzmW02JyZwXgORfHWBE.
    lock_passwd: false
```

The following portion of a **cloud-init** custom script creates a **.vimrc** file for user **developer2**:

```
write_files:
  - path: /home/developer2/.vimrc
    content: |
      set ai et ts=2 sts=2 sw=2
    owner: developer2:developer2
    mode: '0664'
```

Be aware that spacing and indentation is very important to YAML, and that incorrect indentation can cause the script to fail. Therefore, it may be easier to create the script in a separate text editor and then copy and paste the script into the **Custom Script** field. See http://cloudinit.readthedocs.io for cloud-config examples.

> **Important**
>
> Some changes made by **cloud-init** are persistent, even when applied by starting the virtual machine using **Run Once**. This includes setting the host name, configuring a time zone, adding users, setting or changing user passwords, and actions performed by custom scripts.
>
> Network settings are not persistent. Even if you persistently specify **cloud-init** settings by editing a virtual machine, network settings are not applied unless you start the virtual machine using **Run Once**, and enable **Use Cloud-Init** in the **Initial Run** settings. To persistently configure network settings, use the **write_files** module as part of a custom script.

Using cloud-init to Configure a New Virtual Machine

New virtual machines can be deployed using a *cloud-init* enabled template. You could just create and run a new virtual machine from the template, and the **`cloud-init`** directives included in the template would be used to customize the resulting virtual machine. Alternatively, you can override the template's **`cloud-init`** settings to further customize the resulting virtual machine. For example, use **`cloud-init`** to configure the virtual machine, including host name, creating additional users, changing the network configuration, and specifying a custom **`cloud-init`** script to run at boot.

Unless you are making a very simple change, it is often a better idea to edit the settings for the virtual machine, rather than specifying the settings through **Run Once**. When advanced options display while editing a virtual machine, select the **Initial Run** tab. Select the box for **Use Cloud-Init/Sysprep**, and specify the **`cloud-init`** settings.

Apply the settings made when editing a virtual machine by starting the virtual machine using **Run Once**. From the **Run Virtual Machine(s)** dialog, click the **+** icon next to **Initial Run**, and then make sure that the check box in the **Use Cloud-Init** line is selected. You will see that the **`cloud-init`** settings are populated based on the changes that you made when you edited the virtual machine. Making additional changes here will not modify the **Initial Run** settings, which are accessible when you edit a virtual machine. If you find that you made a mistake with your **`cloud-init`** settings, rather than having to type everything in again, you can edit the virtual machine to make slight adjustments to the **Initial Run** settings. Then restart the machine again, using **Run Once**.

References

Further information is available in the "Using Cloud-Init to Automate the Configuration of Virtual Machines" chapter of the *Virtual Machine Management Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#Using_Cloud-Init_to_Automate_the_Configuration_of_Virtual_Machines

cloud-init: Cloud config examples
http://cloudinit.readthedocs.io/en/latest/topics/examples.html

Guided Exercise

Automating Virtual Machine Configuration

In this exercise, you will deploy a virtual machine using a template configured to use **`cloud-init`**.

Outcomes

You should be able to prepare a template that is installed with **`cloud-init`**, and create a virtual machine with a customized configuration using **`cloud-init`** to apply your customization.

Before You Begin

Log in to **`workstation`** as **`student`** using **`student`** as the password.

On **`workstation`**, run the **`lab vms-cloud-init start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab vms-cloud-init start
```

1. On **`workstation`**, open Firefox and using the `https://rhvm.lab.example.com` URL go to the RHV-M web interface. Click on the **Administration Portal** link and log in to the web interface as the **`admin`** user, with the **`internal`** profile, using **`redhat`** as the password.

2. Install *cloud-init* on **`rhel-vm2`**.

   2.1. Navigate to **Virtual Machines** by clicking on **Compute** in the menu bar, and then selecting **Virtual Machines**.

   2.2. Highlight the row containing the **`rhel-vm2`** virtual machine by clicking an empty cell in that row. Ensure that the **`rhel-vm2`** virtual machine is running. If it is not, click the **Run** button to start it.

   2.3. When available, open the console and log in to **`rhel-vm2`** as user **`root`** with **`redhat`** as the password.

   2.4. To install the *cloud-init* software package on your **`rhel-vm2`** virtual machine, you normally would ensure the system is registered with Red Hat Subscription Manager and has the correct entitlements and Yum repositories enabled.

   In this classroom environment, this step has been modified because the classroom might not have access to the Content Distribution Network or a Red Hat Satellite server. Instead, local Yum repositories have been provided, which contain the correct packages.

   In a previous lab, you downloaded a file to enable the local Yum repositories. Verify that the rhel_dvd repository is accessible using the **`yum repolist`** command.

```
[root@rhel-vm2 ~]# yum repolist
...output omitted...
repo id          repo name                          status
rhel_dvd         Remove classroom copy of dvd       5,152
...output omitted...
```

If you do not see the rhel_dvd repository, download the **rhel_dvd.repo** file from `http://materials.example.com/yum.repos.d/rhel_dvd.repo` and place it in the **/etc/yum.repos.d/** directory.

```
[root@rhel-vm2 ~]# curl http://materials.example.com/yum.repos.d/rhel_dvd.repo \
-o /etc/yum.repos.d/rhel_dvd.repo
...output omitted...
```

Install the *cloud-init* package with all dependencies using the **yum** command.

```
[root@rhel-vm2 ~]# yum -y install cloud-init
...output omitted...
```

2.5. Shut down the **rhel-vm2** virtual machine using the **systemctl poweroff** command.

```
[root@rhel-vm2 ~]# systemctl poweroff
```

3. Create a new template named **rhel-cloud-template** from the **rhel-vm2** virtual machine.

3.1. With the **rhel-vm2** virtual machine row highlighted, click the three vertical dots to the right of the **Create Snapshot** button, and then select **Make Template**.

3.2. Use the **New Template** window to specify all the settings needed to create the template.

3.3. In the **Name** text field, type the name **rhel-cloud-template** for the template.

3.4. In the **Description** text field, type the description **cloud-init based template** for the template.

3.5. In the **Alias** text field, type the name **rhel7-cloud-init** for the alias.

3.6. If it is not already selected, select the **Seal Template** check box.

3.7. Leave the other options as their default settings. Click the **OK** button to create the template.

3.8. Notice that the **rhel-vm2** virtual machine and all of its disks are locked. This step can take a couple of minutes to finish. When the process of creating the template is complete, the virtual machine is released from its locked state.

4. Configure **rhel-cloud-template** to use **cloud-init** for its **Initial Run**. These settings will be inherited by any virtual machine created from this template.

4.1. View existing templates by clicking **Compute** in the menu bar, and then selecting **Templates**.

4.2. Ensure the row containing **`rhel-cloud-template`** is highlighted by clicking an empty cell in that row.

4.3. Click the **Edit** button to edit the properties of **`rhel-cloud-template`**.

4.4. In the **Edit Template** window, click the **Show Advanced Options** button, if it is visible, to access the advanced options. Click the **Initial Run** tab.

4.5. Ensure that the check box for **Use Cloud-Init/Sysprep** is selected.

4.6. Make sure the **VM Hostname** text field is empty.

4.7. Leave the other options with their default values. Click the **OK** button to accept the changes.

5. Create a new virtual machine named **`rhel-cloud-vm`**, based on the template **`rhel-cloud-template`**.

5.1. On the **Compute >> Virtual Machines** page, click the **New** button. The **New Virtual Machine** window displays.

5.2. In the **Cluster** menu, choose **clusterone**.

5.3. In the **Template** menu, choose **rhel-cloud-template**.

5.4. In the **Name** field, type the name **`rhel-cloud-vm`** for the virtual machine.

5.5. In the **Description** field, type the description **`RHEL Guest for cloud-init test`** for the virtual machine.

5.6. Notice that the network configuration matches the one from the original **`rhel-vm2`** virtual machine, and that you are unable to create any new disk images.

5.7. Leave all the other options as they are and click the **OK** button to create the new template-based virtual machine.

5.8. Wait until the virtual machine is created.

6. Customize the **`cloud-init`** configuration for the new **`rhel-cloud-vm`** virtual machine, and then run it for the first time.

6.1. A new virtual machine created from a template has an initial state of stopped. Ensure the row containing the **`rhel-cloud-vm`** virtual machine is highlighted by clicking an empty cell in that row.

6.2. With the row containing **`rhel-cloud-vm`** highlighted, click the **Run** drop-down menu and select **Run Once** to display the **Run Virtual Machine(s)** dialog.

6.3. Click the **+** icon next to **Initial Run**.

6.4. Make sure that the check box in the **Use Cloud-Init** line is selected.

6.5. In the **VM Hostname** text field, ensure that the **`rhel-cloud-vm`** name is present. This will change the hostname of the new virtual machine.

6.6. Click **Authentication** to reveal the Authentication section. Enter a **User Name** of **`developer`**, and a **Password** of **`redhat`**. Enter the password again in the **Verify Password** field.

6.7. Click **OK** to initiate the creation of the virtual machine, using **`cloud-init`** as the provisioning method.

6.8. Once the virtual machine is running, click the **Console** button and verify the changes made by **`cloud-init`**. Note that it may take a minute before the virtual machine starts booting from its hard drive. Log in as the **`developer`** user with **`redhat`** as the password. Notice that the hostname of the virtual machine is **`rhel-cloud-vm`**. When you have finished reviewing the changes to the **`rhel-cloud-vm`** virtual machine, close the virtual machine console.

7. The exercise is complete, but you must clean up your classroom environment in order to preserve resources for upcoming exercises. Delete the **`rhel-cloud-vm`** virtual machine and the **`rhel-cloud-template`** template from your RHV environment.

7.1. View available virtual machines by clicking on **Compute** in the menu bar, and then selecting **Virtual Machines**. From the list of available virtual machines, highlight the row for the **`rhel-cloud-vm`** virtual machine by clicking an empty cell in that row. Click the **Shutdown** drop-down menu and then select **Power Off**. When the **Power Off Virtual Machine(s)** window appears, confirm powering off the machine by clicking the **OK** button.

7.2. With the row containing the **`rhel-cloud-vm`** virtual machine highlighted, click the **Remove** button to remove the virtual machine. When the **Remove Virtual Machine(s)** window displays, make sure the check box next to **Remove Disk(s)** is selected. Confirm the removal of the machine by clicking the **OK** button.

7.3. Notice that the **`rhel-cloud-vm`** has been removed and is no longer present on the list of available virtual machines.

7.4. View available templates by clicking on **Compute** in the menu bar, and then selecting **Templates**. From the list of available templates, highlight the row containing **`rhel-cloud-template`** by clicking an empty cell in that row.

7.5. With the row containing the **`rhel-cloud-template`** template highlighted, click the **Remove** button to remove the template. When the **Remove Template(s)** window displays, click the **OK** button to confirm the removal of the template.

7.6. Notice that the template, **`rhel-cloud-template`**, has been removed and is no longer present on the list of available templates. Log out from the **Administration Portal**.

Finish

On **`workstation`**, run the **`lab vms-cloud-init finish`** script to complete this exercise.

```
[student@workstation ~]$ lab vms-cloud-init finish
```

This concludes the guided exercise.

Lab

Deploying and Managing Virtual Machines

Performance Checklist

In this lab, you will create templates and use them to deploy virtual machines.

Outcome

You should be able to create a template and deploy a virtual machine.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab vms-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working. Wait for the start script to finish before starting the lab. The start script creates a new virtual machine named **rhel-vm1**, which does not initially exist in the **production** data center.

```
[student@workstation ~]$ lab vms-review start
```

The start script created the **rhel-vm1** virtual machine in the **production** data center.

1. Install *cloud-init* on the **rhel-vm1** virtual machine in the **production** data center.
2. Create a new template named **cloud-lab-template** in the **production** data center, using the **rhel-vm1** virtual machine. Give the template a description of **cloud-init based template**, and give the template disk an alias of **rhel7-cloud-lab**.
3. Ensure that virtual machines created from the **cloud-lab-template** template will use **cloud-init**.
4. Use the **cloud-lab-template** template to deploy a new virtual machine called **rhel-lab-vm** to the **production** data center. Configure **cloud-init** to:
   - Make sure that the new virtual machine has the correct host name (**rhel-lab-vm**).
   - Configure the **/etc/motd** file to display the message: **This VM has been provisioned using cloud-init.**
   - Create a new user called **labuser** with **redhat** as the password.
5. Once the new virtual machine is ready, log in to the **rhel-lab-vm** virtual machine as **labuser** to ensure that all the changes were successfully applied by **cloud-init**.

Evaluation

On **workstation**, run the **lab vms-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab vms-review grade
```

> **Note**
>
> The grading script uses the IP address for the virtual machine reported by the qemu guest agent. If the grading script fails the check for evaluating the **labuser** user and the **/etc/motd** file, verify that the IP address for the **rhel-vm1** virtual machine is visible from the **Compute >> Virtual Machines** page. If the IP address is not visible, try restarting the **qemu-guest-agent** service on the **rhel-vm1** virtual machine, or shut down and then restart the virtual machine. Run the grading script again once the IP address is displayed.

Cleanup

Using the Administration Portal, manually clean up after completing the lab.

1. Remove the **rhel-lab-vm** virtual machine.
2. Remove the **cloud-lab-template** template.

Finish

On **workstation**, run the **lab vms-review finish** script to complete this exercise.

```
[student@workstation ~]$ lab vms-review finish
```

This concludes the lab.

Solution

Deploying and Managing Virtual Machines

Performance Checklist

In this lab, you will create templates and use them to deploy virtual machines.

Outcome

You should be able to create a template and deploy a virtual machine.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab vms-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working. Wait for the start script to finish before starting the lab. The start script creates a new virtual machine named **rhel-vm1**, which does not initially exist in the **production** data center.

```
[student@workstation ~]$ lab vms-review start
```

The start script created the **rhel-vm1** virtual machine in the **production** data center.

1. Install *cloud-init* on the **rhel-vm1** virtual machine in the **production** data center.

    1.1. Open Firefox and, using the `https://rhvm.lab.example.com` URL, log in to the **Administration Portal** as the **admin** user, with the **internal** profile, using **redhat** as the password.

    1.2. Navigate to **Virtual Machines** by clicking on **Compute** in the menu bar and then selecting **Virtual Machines**.

    1.3. Highlight the row containing the **rhel-vm1** virtual machine in the **clustertwo** cluster by clicking on an empty cell in that row. Ensure that the **rhel-vm1** virtual machine is running. If it is not, click the **Run** button to start it.

    1.4. When available, open the console and log in to **rhel-vm1** as user **root** with **redhat** as the password.

    1.5. To install the *cloud-init* software package on your **rhel-vm1** virtual machine, you normally would ensure the system is registered with Red Hat Subscription Manager and has the correct entitlements and Yum repositories enabled.

    In this classroom environment, this step has been modified because the classroom might not have access to the Content Distribution Network or a Red Hat Satellite server. Instead, local Yum repositories have been provided that contain the correct packages.

    Install the *cloud-init* package with all dependencies using the **yum** command.

```
[root@rhel-vm1 ~]# yum -y install cloud-init
...output omitted...
```

1.6. Shut down the **rhel-vm1** virtual machine using the **systemctl poweroff** command.

```
[root@rhel-vm1 ~]# systemctl poweroff
```

2. Create a new template named **cloud-lab-template** in the **production** data center, using the **rhel-vm1** virtual machine. Give the template a description of **cloud-init based template**, and give the template disk an alias of **rhel7-cloud-lab**.

   2.1. Ensure that the row containing the **rhel-vm1** virtual machine in the **clustertwo** cluster is highlighted by clicking an empty cell in that row.

   2.2. With the **rhel-vm1** virtual machine row highlighted, click the three vertical dots to the right of the **Create Snapshot** button, and then select **Make Template**.

   2.3. Specify all the settings needed to create the template using the **New Template** window.

   2.4. In the **Name** text field, type **cloud-lab-template** as the template name.

   2.5. In the **Description** text field, type **cloud-init based template** as the template description.

   2.6. In the **Alias** text field, type **rhel7-cloud-lab** as the template alias.

   2.7. If it is not already selected, then select the **Seal Template (Linux only)** check box.

   2.8. Leave all other options as they are.

   2.9. Click the **OK** button to begin the template creation. This step can take a couple of minutes to finish. When the process of creating the template is complete, the virtual machine is released from its locked state.

3. Ensure that virtual machines created from the **cloud-lab-template** template will use **cloud-init**.

   3.1. On the **Compute >> Templates** page, click an empty cell in the row containing **cloud-lab-template**. With the row containing **cloud-lab-template** highlighted, click the **Edit** button. The **Edit Template** window displays.

   3.2. In the **Edit Template** window, click the **Show Advanced Options** button to access the advanced options. Click the **Initial Run** tab.

   3.3. Make sure the check box for **Use Cloud-Init/Sysprep** is selected.

   3.4. Make sure the **VM Hostname** text field is empty.

   3.5. Leave the other options with their default values. Click the **OK** button to accept the changes.

4. Use the **cloud-lab-template** template to deploy a new virtual machine called **rhel-lab-vm** to the **production** data center. Configure **cloud-init** to:

- Make sure that the new virtual machine has the correct host name (**`rhel-lab-vm`**).
- Configure the **`/etc/motd`** file to display the message: **`This VM has been provisioned using cloud-init.`**
- Create a new user called **`labuser`** with **`redhat`** as the password.

4.1. On the **Compute >> Virtual Machines** page, click the **New** button. The **New Virtual Machine** window displays.

4.2. In the **Cluster** menu, choose **clustertwo**.

4.3. In the **Template** menu, choose **cloud-lab-template**.

4.4. In the **Name** field, type **`rhel-lab-vm`** as the virtual machine name.

4.5. In the **Description** field, type **`RHEL Guest for cloud-init lab`** as the virtual machine description.

4.6. Leave all the other options as they are and click the **OK** button to create the new template-based virtual machine.

4.7. Wait while the virtual machine is created.

4.8. After the **`rhel-lab-vm`** virtual machine becomes ready to use, highlight the row containing the virtual machine by clicking an empty cell in that row.

4.9. With the row containing **`rhel-lab-vm`** highlighted, click the **Run** drop-down menu and select **Run Once** to display the **Run Virtual Machine(s)** window.

4.10. Click the **+** icon next to **Initial Run**.

4.11. Make sure that the check box on the **Use Cloud-init** line is selected.

4.12. In the **VM Hostname** text field, ensure that the **`rhel-lab-vm`** name is present. This will change the host name of the new virtual machine.

4.13. Click **Authentication** to reveal the Authentication section. Enter a **User Name** of **`labuser`**, and a **Password** of **`redhat`**. Enter the password again in the **Verify Password** field.

4.14. In the **Custom Script** text area, enter the following lines:

```
write_files:
  - path: /etc/motd
    content: |
      This VM has been provisioned using cloud-init.
```

> **Note**
> YAML used by **`cloud-init`** is sensitive to white space indentation. Indent your script as shown in the example for this step, using two spaces for each indentation level. The blank line at the end (with one or more spaces) ensures a new line character is added to the end of the file.

4.15. Click **OK** to initiate the creation of the virtual machine, using **`cloud-init`** as the provisioning method.

5. Once the new virtual machine is ready, log in to the **`rhel-lab-vm`** virtual machine as **`labuser`** to ensure that all the changes were successfully applied by **`cloud-init`**.

   5.1. Once the virtual machine is running, click the **Console** button and verify the changes made by **`cloud-init`**. Log in as the **`labuser`** user with **`redhat`** as the password. Notice that the host name of the virtual machine is **`rhel-lab-vm`**. The message displayed after you log in matches that which **`cloud-init`** placed into the **`/etc/motd`** file. When you have finished reviewing the changes to the **`rhel-lab-vm`** virtual machine, close the virtual machine console.

Evaluation

On **`workstation`**, run the **`lab vms-review grade`** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab vms-review grade
```

> **Note**
> The grading script uses the IP address for the virtual machine reported by the qemu guest agent. If the grading script fails the check for evaluating the **`labuser`** user and the **`/etc/motd`** file, verify that the IP address for the **`rhel-vm1`** virtual machine is visible from the **`Compute >> Virtual Machines`** page. If the IP address is not visible, try restarting the **`qemu-guest-agent`** service on the **`rhel-vm1`** virtual machine, or shut down and then restart the virtual machine. Run the grading script again once the IP address is displayed.

Cleanup

Using the Administration Portal, manually clean up after completing the lab.

1. Remove the **`rhel-lab-vm`** virtual machine.
2. Remove the **`cloud-lab-template`** template.

Finish

On **`workstation`**, run the **`lab vms-review finish`** script to complete this exercise.

```
[student@workstation ~]$ lab vms-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- Installing guest agents for Red Hat Enterprise Linux and Microsoft Windows allows RHV-M to see additional virtual machine information, such as the host name, IP address, and logged in users.
- **Advanced Options** allow you to refine the hardware on your virtual machine, including specifying the number of virtual sockets, cores, threads per core, and a CPU type for the virtual machine.
- A template is a copy of a virtual machine used to simplify subsequent creation of similar virtual machines.
- A virtual machine that will be used as a template must be "sealed" by removing any configuration information, authentication keys, and data unique to the original machine.
- The **`cloud-init`** tool can automate provisioning of new virtual machines to help prevent configuration conflicts.

Chapter 9

Migrating Virtual Machines Between Hosts

Goal	Migrate and control automatic migration of virtual machines between hosts in a single cluster.
Objectives	• Migrate virtual machines from one host to another within a cluster in the Red Hat Virtualization environment. • Configure virtual machines to automatically schedule and migrate VMs using cluster policies.
Sections	• Migrating a Virtual Machine in a Cluster (and Guided Exercise) • Configuring Automated Migration and Scheduling Policies (and Guided Exercise)
Lab	Migrating Virtual Machines Between Hosts

Migrating a Virtual Machine in a Cluster

Objectives

After completing this section, you should be able to migrate virtual machines from one host to another within a cluster in the Red Hat Virtualization environment.

Virtual Machine Migration

Live migration refers to the process of moving a virtual machine from one physical host to another while it is running. The contents of memory, network configuration, and access to storage are moved from the original virtual machine on one physical host to a new virtual machine on a different physical host, which then replaces the old virtual machine. Live migration is useful to support maintenance tasks on hosts without disrupting running virtual machines.

Live migration is transparent to the end-user. The virtual machine remains powered on, and user applications continue to run, while the virtual machine is migrated to a new physical host. Clients communicating with the virtual machine should notice no more than a network pause of a few milliseconds as the transfer completes.

For live migration to work properly, the new host must have a CPU with the same architecture and features as the original host. Red Hat Virtualization helps manage this by organizing hosts into clusters. A virtual machine may only migrate to hosts that are members of its cluster. This helps ensure that virtual machines do not migrate between machines that support different sets of processor features.

Administrators must ensure that their Red Hat Virtualization environment is correctly configured to support live migration before starting a migration. Live migration of virtual machines requires the following configuration prerequisites:

- The virtual machine must be migrated to a host in the same cluster as the host where the virtual machine is running. The status of both hosts must display **Up**.
- Both hosts must have access to the same virtual networks, VLANs, and data storage domains.
- The destination host must have enough CPU capacity and RAM to support the requirements of the virtual machine.
- The virtual machine must not have the **cache!=none** custom property set. The **cache** parameter configures the different cache modes for a virtual machine. Live migration requires a disabled virtual machine cache to ensure a coherent virtual machine migration.

Live migration is performed using the migration network. The default configuration uses the **ovirtmgmt** network as both the management network and the migration network. Although each live migration is limited to a maximum transfer speed, and there are a maximum number of migrations that may run concurrently, concurrent live migrations can saturate a network shared by management and migration traffic. For best performance, the storage, migration, and management networks should be separated to avoid network saturation.

Manually Migrating Virtual Machines

Red Hat Virtualization supports manual migration of a virtual machine between hosts associated with the same cluster. Migration can be triggered using the RHV-M **Administration Portal**. A virtual machine can be migrated to a specific host, or to a host that RHV-M automatically selects. Manual migration of a virtual machine requires administrative privileges.

The **Compute** → **Virtual Machines** menu option displays the **Compute >> Virtual Machines** page in the RHV-M **Administration Portal**. The **Compute >> Virtual Machines** page lists all the virtual machines. Each virtual machine is represented in a specific row. These rows use different columns to display properties of the virtual machines. One of these columns, the **Host** column, shows the host that contains the virtual machine.

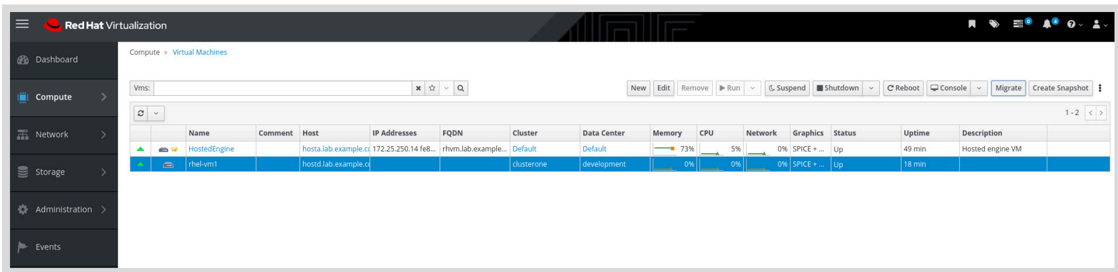

Figure 9.1: Viewing the virtual machines

The **Compute >> Virtual Machines** page includes a **Migrate** button that triggers migration of the selected virtual machine. Clicking on the **Migrate** button opens the **Migrate VM(s)** window. The **Migrate VM(s)** window has a **Destination Host** field that allows you to select a specific destination host for the virtual machine. The default value of the **Destination Host** field is **Automatically Choose Host**, which causes the RHV-M to automatically decide the destination host for the virtual machine. The **Migrate** button in the **Migrate VM(s)** window is used to confirm the intention of migrating the selected virtual machine. After confirming the migration, the **Status** field of the virtual machine transitions through a series of values, including **Migrating From: 0%** and **Migrating To**, before finishing with a status of **Up**.

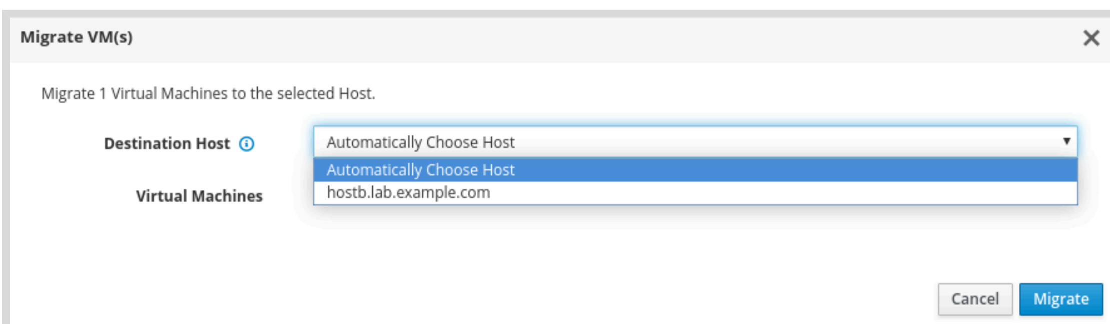

Figure 9.2: Selecting the destination host

References

For more information, refer to the *Manually Migrating Virtual Machines* section of the *Administrative Tasks* chapter in the *Red Hat Virtualization Virtual Machine Management Guide* at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/sect-migrating_virtual_machines_between_hosts#Manually_migrating_virtual_machines

Guided Exercise

Migrating a Virtual Machine in a Cluster

In this exercise, you will manually migrate a running virtual machine from one host to another within a cluster.

Outcomes

You should be able to manually migrate a virtual machine between two hosts.

Before You Begin

This exercise uses the **`development`** data center, which contains the **`clusterone`** cluster, and the **`hostb`** and **`hostd`** hosts. This exercise also uses the **`rhel-vm1`** virtual machine, which runs on either the **`hostb`** or the **`hostd`** host.

Log in as the **`student`** user on **`workstation`** and run the **`lab migration-vm start`** command. This command ensures that the hosts, clusters, data centers, and networks of the Red Hat Virtualization environment are configured appropriately.

```
[student@workstation ~]$ lab migration-vm start
```

1. Attach the **`hostd.lab.example.com`** host to the **`development`** data center.

   1.1. Using the `https://rhvm.lab.example.com` URL log in to the **Administration Portal** of the RHV Manager as **`admin`** in the **`internal`** profile. Use **`redhat`** as the password.

   1.2. From the menu, click **Compute** → **Hosts** to view the list of available hosts.

   1.3. From the **Compute >> Hosts** page, select the **`hostd.lab.example.com`** host.

   1.4. Click **Management**, and then select **Maintenance** to mark **`hostd.lab.example.com`** for housekeeping operations. The **Maintenance Host(s)** window displays.

   1.5. Click **OK** to confirm.

   1.6. From the **Compute >> Hosts** page, select the **`hostd.lab.example.com`** host, and then click **Edit**. The **Edit Host** window displays.

   1.7. Select **clusterone** from the drop-down menu of the **Host Cluster** field. Click **OK** to set the cluster of **`hostd.lab.example.com`** to **`clusterone`**. In the **Power Management Configuration** window, click **OK** to ignore configuring the power management settings and continue setting the cluster for **`hostd.lab.example.com`** to **`clusterone`**.

2. Attach the existing **`storage-net`** and **`vm-net`** logical networks to **`hostd.lab.example.com`**. The logical network **`vm-net`** should be on **`eth0`** and use DHCP to get an address. The logical network **`storage-net`** should be on **`eth1`**, with the static IPv4 address **`172.24.0.13`**, and the **`255.255.255.0`** netmask.

2.1. From the **Compute >> Hosts** page, click the name of the `hostd.lab.example.com` host.

2.2. In the **Compute >> Hosts >> hostd.lab.example.com** page, click the **Network Interfaces** tab.

2.3. Click the **Setup Host Networks** button to change the network configuration of `hostd.lab.example.com`.

2.4. In the **Setup Host hostd.lab.example.com Networks** window, click and drag the **storage-net** box from the right side to the left side of the window. Drop the box onto the `no network assigned` field, next to the `eth1` network interface.

2.5. Click and drag the **vm-net (VLAN 10)** box from the right side to the left side of the window. Drop the box next to the `eth0` network interface. After dropping, you should see two logical networks assigned to `eth0` interface.

2.6. Click on the pencil icon inside the `storage-net` box. The **Edit Network storage-net** window displays. In the **Edit Network storage-net** window, under **Boot Protocol**, select the `Static` radio button.

2.7. In the **IP** field, type `172.24.0.13` as the IP address of `hostd.lab.example.com` in the `storage-net` network.

2.8. In the **Netmask/Routing Prefix** field, type `255.255.255.0` as the netmask. Leave the **Gateway** field empty.

2.9. Click **OK** to save the settings.

2.10. Ensure that the check boxes for the **Verify connectivity between Host and Engine** and **Save network configuration** options are selected.

2.11. Click **OK** to save the new network configuration for `hostd.lab.example.com`.

2.12. In the **Compute >> Hosts** page, click **Management** and select **Activate** to make `hostd.lab.example.com` operational in the `clusterone` cluster.

3. Identify the host that is running the `rhel-vm1` virtual machine. If there is more than one `rhel-vm1` virtual machine, use the virtual machine located in the `clusterone` cluster.

3.1. From the menu bar of the RHV-M **Administration Portal**, click **Compute** → **Virtual Machines** to view the list of virtual machines.

3.2. In the **Status** field for the `rhel-vm1` virtual machine, verify that the status of the `rhel-vm1` virtual machine is `Up`.

3.3. View the **Host** field for the `rhel-vm1` virtual machine to determine the host for this virtual machine.

3.4. Right-click the row for the `rhel-vm1` virtual machine, and click **Console**.

In the **Opening console.vv** window, which opens in Firefox, select the **Open with** radio button, and then select **Remote Viewer (default)** from the drop-down menu, if not already selected. Click **OK** to display the console for `rhel-vm1` with Remote Viewer.

4. With the **rhel-vm1:1 - Remote Viewer** window open and displaying the console, migrate the `rhel-vm1` virtual machine to a different host.

4.1. In the **Compute >> Virtual Machines** page of the **Administration Portal** of RHV Manager, select the **`rhel-vm1`** virtual machine.

4.2. Click the **Migrate** button. The **Migrate VM(s)** window displays.

4.3. In the **Destination Host** field, use the drop-down menu to select the host to which the **`rhel-vm1`** virtual machine should be migrated. The destination host may display as either **`hostb.lab.example.com`** or **`hostd.lab.example.com`**. Click the **Migrate** button to start the migration.

4.4. Verify that the value of the **Status** field for the **`rhel-vm1`** virtual machine transitions from **Migrating From** to **Up**. The console for **`rhel-vm1`** should show that the virtual machine is running during the migration.

5. Verify that the **Host** field for the **`rhel-vm1`** virtual machine displays the new host to confirm the successful migration of the virtual machine.

Finish

On **`workstation`**, run the **`lab migration-vm finish`** script to complete this exercise.

```
[student@workstation ~]$ lab migration-vm finish
```

This concludes the guided exercise.

Configuring Automated Migration and Scheduling Policies

Objectives

After completing this section, you should be able to configure virtual machines to automatically schedule and migrate VMs using cluster policies.

Describing Automated Migration

RHV-M checks the resources of the clustered hosts either when a virtual machine starts, or when migrating a virtual machine, to determine an appropriate host for the virtual machine in the cluster. RHV-M also checks the resources periodically to detect any noncompliance between the current load on the individual host and the cluster policies. If the current load on the individual host is not compliant with the cluster policies, RHV-M migrates virtual machines from one host to another host in the cluster without requiring manual administrative intervention. This method of having RHV-M automatically migrate a virtual machine from one host to another is called automated migration.

RHV environment requires the hosts to be offline for upgrades and maintenance. When you mark a host for maintenance, RHV-M automatically migrates virtual machines from the host in maintenance to another available host. The migration policy states the criteria for the live migration of the virtual machines.

Describing Load Balancing and Scheduling

In RHV, *load balancing* refers to the distribution of the virtual machines among the hosts in the cluster to ensure efficient utilization of the clustered resources. Each cluster uses a specific policy for load balancing that has tunable properties. RHV-M uses these properties to decide when to move the virtual machines from one host to another. Load balancing runs once every minute for a certain cluster to ensure that the load on the cluster is balanced.

RHV-M uses the process called *scheduling* to determine the host on which a virtual machine starts. RHV-M performs scheduling based on the scheduling policies of the cluster. A scheduling policy is a combination of filters, weights, and load balancing logic. The filters apply the hard constraints that a host must satisfy to run a virtual machine, such as the minimum RAM, or CPU. The weights apply the soft constraints that control the relative preference for a host to run a virtual machine. Lower weights are considered better for the scheduler preference. The load balancing logic determines whether a specific host is underutilized or overutilized. After identifying the underutilized and overutilized hosts, the load balancing logic calls the scheduler to migrate the virtual machine from the overutilized host to the underutilized host.

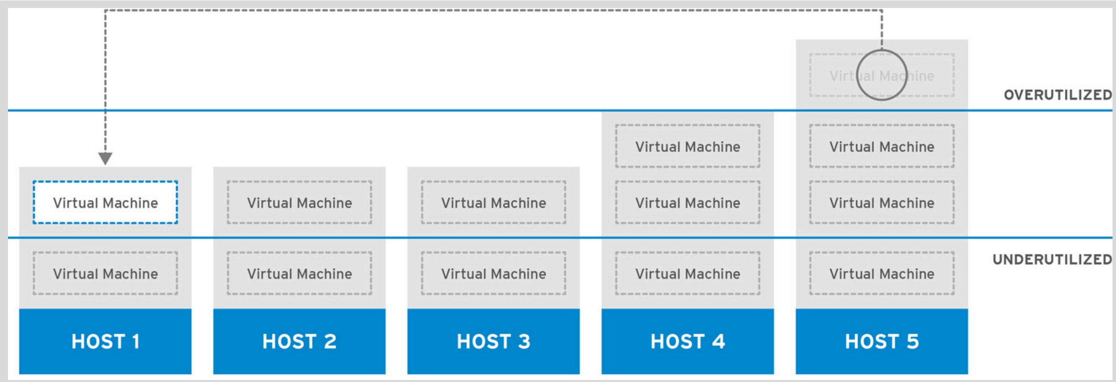

Figure 9.3: Virtual machine migrating off of overutilized host

Automated migration and scheduling are different actions that RHV-M performs to make sure that load in the cluster is balanced. Load balancing in RHV is unrelated to the high availability of the virtual machines. The high availability of a virtual machine is enabled separately on the virtual machine. A high availability virtual machine is automatically restarted in the event of an interruption, either on its original host or another appropriate host. The interruption could be due to host failure or due to host being put in maintenance mode. More details on the high availability virtual machine are given in a later chapter.

Configuring a Migration Policy

Virtual machine migration is a network-intensive operation. RHV-M copies the memory state of the virtual machine over the network to the new host. When a host contains ten or more virtual machines, migrating all of them can be a long and resource-consuming process. Therefore, administrators must be sure to select the migration policy that best suits their environment.

> **Important**
>
> For live migration to work, RHV-M copies the virtual machine state to the new host in real time. As the migration completes, the state change in place while the migration was running may need to be retransmitted. Eventually, the migration converges and allows RHV-M to pause the virtual machine for a fraction of a second to transmit the last few changes to the new host. Finally, the virtual machine is resumed on the new host.
>
> A system that is very busy may take longer to converge. Migration policies also determine how RHV handles this situation.

RHV-M also automatically initiates live migration of virtual machines to maintain load balancing or power-saving levels, according to the current policy. RHV-M allows administrators to disable the automated migration of virtual machines. It is also possible to disable manual migration of virtual machines by setting the virtual machine to run only on a specific host. The configuration of a migration policy also includes the configuration of a resilience policy, which determines the virtual machine migration policy when a host fails.

In the **Administration Portal** of RHV-M, selecting **Compute** → **Clusters** from the menu displays the **Compute >> Clusters** page. The **Compute >> Clusters** page lists the available clusters. To configure the migration policy for a specific cluster, select the cluster, and then click **Edit** to open the **Edit Cluster** window.

The **Edit Cluster** window contains the migration configuration for the cluster under the **Migration Policy** section. The **Migration Policy** section allows you to select the migration policy to apply in the **Migration Policy** menu. The following describes the available migration policies.

- The **Minimal downtime** policy is the default migration policy. This migration policy optimizes for the shortest pause of the virtual machine during migration, but may abort the migration if it is taking an excessive time to converge.

- The **Post-copy migration** policy also optimizes for the shortest pause, if possible. If the migration fails to converge after an extended time, then this policy is applied. Post-copy starts the virtual machine in the destination host as soon as possible. To achieve this, a subset of the virtual machine memory moves to the destination hosts. If the virtual machine tries to access a memory page that is not in the destination host, then it issues a page fault, and the source host transfers that page.

- The **Suspend workload if needed** migration policy supports migration under most load conditions, but a longer pause of the virtual machine may occur if it has a heavy load.

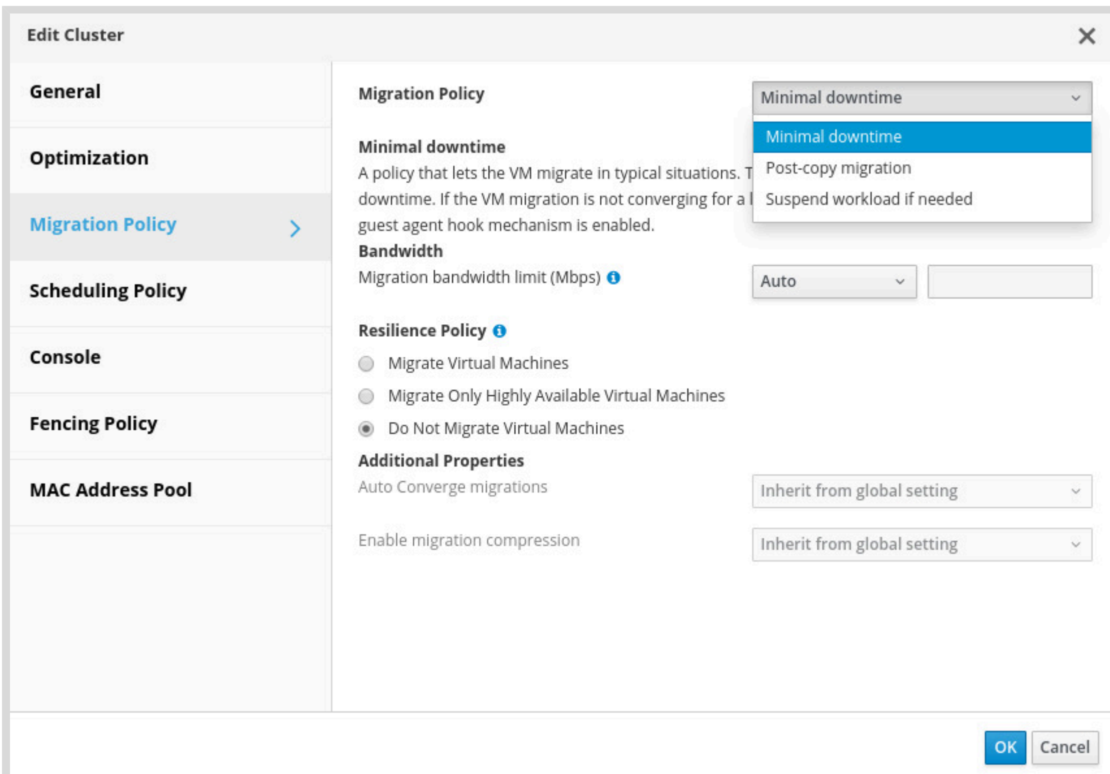

Figure 9.4: Configuring migration policy

The **Bandwidth** menu allows you to limit maximum bandwidth in Mbps per host for migrations, both outgoing and incoming. The three available options for **Bandwidth** are described below.

- The **Auto** mode uses the **`Rate Limit [Mbps]`** setting in the data center **`host network QoS`**. If there is no **`Rate Limit [Mbps]`** setting defined, then the minimum speed for the NICs of the source and destination hosts is applied.

- The **Hypervisor default** mode uses the VDSM setting on the source host.

- The **Custom** mode uses the bandwidth defined by the user in Mbps.

The **Resilience Policy** sets the virtual machine migration policy in the event of host failure. RHV-M migrates virtual machines to other hosts in the cluster if the host unexpectedly shuts down or moves into maintenance mode.

RHV supports the migration of all virtual machines using the **Migrate Virtual Machines** policy, migration of only the highly available virtual machines using the **Migrate only Highly Available Virtual Machines** policy, and disabling the virtual machine migration using the **Do Not Migrate Virtual Machines** option.

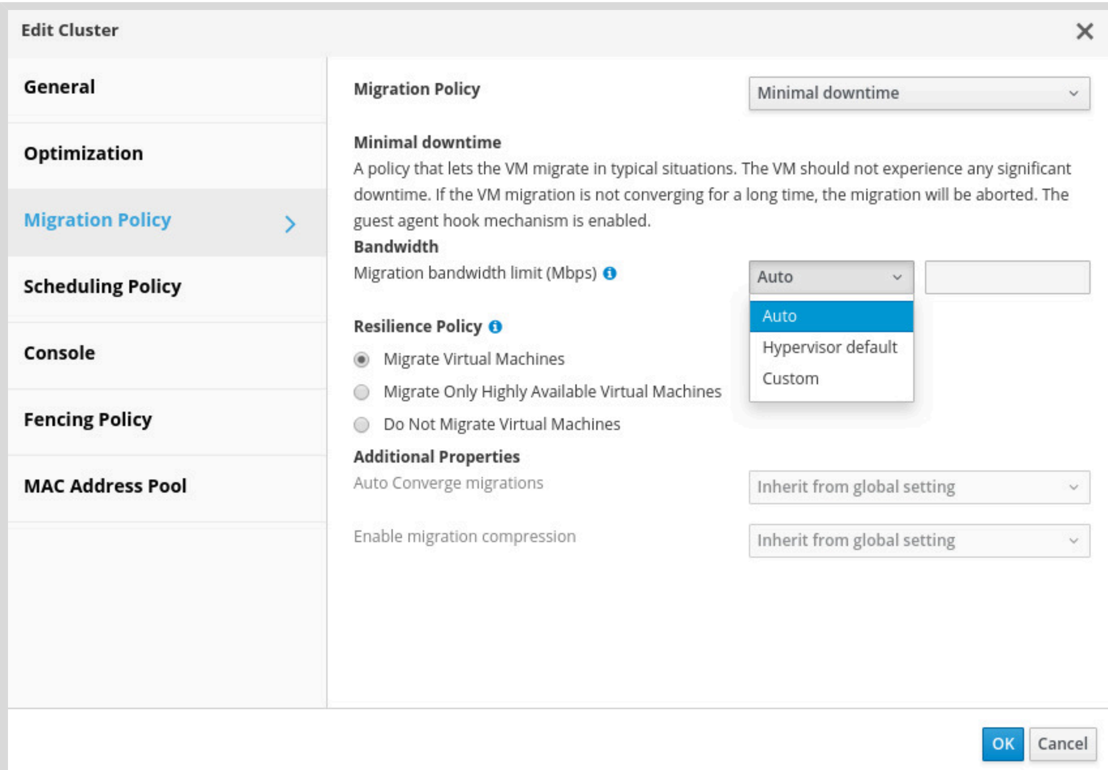

Figure 9.5: Configuring bandwidth and resilience policy

Configuring a Scheduling Policy

The **Edit Cluster** window allows you to configure the scheduling policy for a cluster. The **Scheduling Policy** section contains the current scheduling policy. The default configuration does not allow deployment of a virtual machine on an overutilized host. A host is overutilized when its CPU load is higher than 80% for more than 2 minutes.

The **Select Policy** drop-down menu allows you select the scheduling policy for the cluster. The following discusses the five scheduling policies that RHV supports by default.

- The **power_saving** policy improves the efficiency of electrical power consumption on the hosts. The hosts that remain underutilized, in terms of CPU load, for longer than the defined time interval are marked to be powered down. Before powering down the underutilized host, all of its running virtual machines are migrated to other appropriate hosts.

- The **none** policy disables load balancing in the cluster. This policy spreads the compute (memory and CPU) load evenly across all the hosts in the cluster. Hosts that reach the defined values of any of the policy properties, such as **CpuOverCommitDurationMinutes**, **HighUtilization**, or **MaxFreeMemoryForOverUtilized**, do not run additional attached virtual machines. These policy properties are described later in this section.

- The **cluster_maintenance** policy disables starting new virtual machines during maintenance tasks. Only the highly available virtual machines are scheduled to start on appropriate hosts. In the event of a host failure, any of the highly available or regular virtual machines can migrate to a healthy host.

- The **evenly_distributed** policy spreads the compute (memory and CPU) load evenly across all the hosts in the cluster. Hosts that reach the defined values of any of the policy properties, such as **CpuOverCommitDurationMinutes**, **HighUtilization**, or **MaxFreeMemoryForOverUtilized**, do not run additional attached virtual machines. These policy properties are described later in this section.

- The **vm_evenly_distributed** policy schedules virtual machines evenly between the hosts in the cluster, based on a count of the virtual machines. To keep the cluster balanced, all the hosts should have a virtual machine count below the defined **HighVmCount**, and no host in the cluster should have a virtual machine count beyond the defined **MigrationThreshold**. These policy properties are described later in this section.

Describing Scheduling Policy Properties

Each scheduling policy has an associated set of properties to customize its behavior. The following are among the scheduling policy properties.

- The **HighVmCount** property represents the minimum number of running virtual machines per host required to initiate load balancing. The default value is 10. An overutilized host runs more virtual machines than this number.

- The **MigrationThreshold** property configures a buffer before virtual machines migrate from the host. This value is the maximum inclusive difference in virtual machine count between the highly utilized hosts and least utilized hosts. The default value is 5. To keep the cluster balanced, the virtual machine count of every host should be lower than the value of this property.

- The **SpmVmGrace** property represents the number of slots reserved for virtual machines on SPM hosts. In a cluster, the SPM hosts have relatively lower loads. This property defines how many virtual machines run on the SPM host, in comparison to other hosts. The default value is 5.

- The **CpuOverCommitDurationMinutes** property represents the time in minutes that the host can run a CPU load beyond the utilization values that are defined. After the specified time elapses, the scheduling policy takes action to implement the necessary virtual machine migration. The default value is 2. This value is limited to a maximum of two characters.

- The **HighUtilization** property is the percentage of CPU usage on hosts that causes the virtual machine migration, if the CPU usage continues at this level for the defined time interval. The default value is 80.

Click the **OK** button in the **Edit Cluster** window to apply the selected policy with the defined properties.

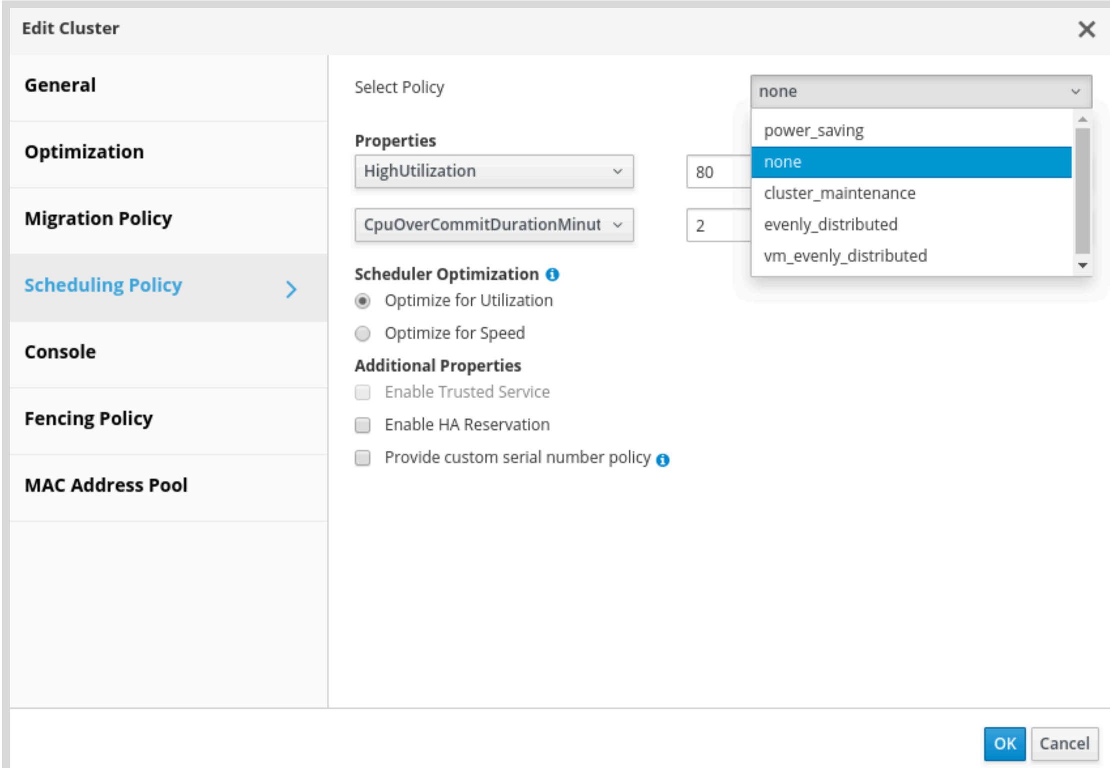

Figure 9.6: Configuring scheduling policy

References

For more information, refer to the *Migration Policy Settings Explained* section of the *Clusters* chapter in the *Red Hat Virtualization Administration Guide* at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/sect-cluster_tasks#Cluster_Migration_Policy_Settings_Explained

For more information, refer to the *Scheduling Policy Settings Explained* section of the *Clusters* chapter in the *Red Hat Virtualization Administration Guide* at https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html/administration_guide/sect-cluster_tasks#Cluster_Scheduling_Policy_Settings

Guided Exercise

Configuring Automated Migration and Scheduling Policies

In this exercise, you will configure a virtual machine to automatically migrate to another host in the cluster when that host enters maintenance mode.

Outcomes

You should be able to:

- Configure a cluster policy to automatically migrate virtual machines when a host moves into maintenance mode.
- Move a host running a VM into maintenance mode.

Before You Begin

This exercise uses the **`development`** data center, which contains the **`clusterone`** cluster, and the **`hostb`** and **`hostd`** hosts. This exercise also uses the **`rhel-vm1`** virtual machine, which runs on either the **`hostb`** or the **`hostd`** host.

Log in as the **`student`** user on **`workstation`** and run the **`lab migration-auto start`** command. This command ensures that the hosts, clusters, data centers, and networks of the Red Hat Virtualization environment are configured appropriately.

```
[student@workstation ~]$ lab migration-auto start
```

1. Verify that the migration policy for the **`clusterone`** cluster is **Minimal downtime**, and the resilience policy is **Migrate Virtual Machines**.

   1.1. Using the `https://rhvm.lab.example.com` URL log into the **Administration Portal** of the RHV Manager as **`admin`** in the **`internal`** profile. Use **`redhat`** as the password.

   1.2. From the menu bar, navigate to **Compute** → **Clusters**.

   1.3. Right-click the row for **`clusterone`**, and then select **Edit**. The **Edit Cluster** window displays.

   1.4. In the **Migration Policy** section, verify that **Migration Policy** is **Minimal downtime**, and the **Resilience Policy** is **Migrate Virtual Machines**. When done, click **OK** to apply the configuration.

2. Move the host running the **`rhel-vm1`** virtual machine into maintenance mode.

   2.1. From the menu bar, click **Compute** → **Virtual Machines**.

   2.2. Determine which host is running the **`rhel-vm1`** virtual machine by checking the **Host** field for this virtual machine.

2.3. From the menu bar, click **Compute** → **Hosts**.

2.4. Select the host that contains the **`rhel-vm1`** virtual machine. Click **Management**, and select **Maintenance**. The **Maintenance Host(s)** window displays.

2.5. Click **OK** to enable maintenance mode for the host.

2.6. In the **Compute >> Hosts** page, verify that the **Status** for the host transitions from **Preparing for Maintenance** to **Maintenance**. This process may take up to a minute.

3. Verify that the **`rhel-vm1`** virtual machine is running on the only host in the **`clusterone`** cluster that is **`Up`** .

3.1. From the menu bar, click **Compute** → **Virtual Machines**.

3.2. Verify that the **Host** for the **`rhel-vm1`** virtual machine is the available host in the **`clusterone`** cluster.

4. Activate the host currently under maintenance in the **`clusterone`** cluster.

4.1. From the menu bar, click **Compute** → **Hosts**.

4.2. Select the host with **Maintenance** as the value for the **Status** field. Click **Management**, and then select **Activate** to activate the host in the **`clusterone`** cluster.

4.3. Verify that the **Status** field for that host transitions from **Activating** to **Up**.

Finish

On **`workstation`**, run the **`lab migration-auto finish`** script to complete this exercise.

```
[student@workstation ~]$ lab migration-auto finish
```

This concludes the guided exercise.

Lab

Migrating Virtual Machines Between Hosts

Performance Checklist

In this lab, you will configure a virtual machine to migrate to another host in the cluster when certain conditions are met.

Outcomes

You should be able to configure a scheduling policy that automatically migrates virtual machines when certain conditions are met.

Before You Begin

This exercise uses the **development** data center, which contains the **clusterone** cluster, and the **hostb** and **hostd** hosts. This exercise also uses the **rhel-vm1**, **rhel-vm2**, and **rhel-vm3** virtual machines, running on either the **hostb** or the **hostd** host.

Log in as the **student** user on **workstation** and run the **lab migration-review start** command. This command ensures that the hosts, clusters, data centers, and networks of the Red Hat Virtualization environment are configured appropriately.

```
[student@workstation ~]$ lab migration-review start
```

1. Clone the **rhel-vm1** virtual machine, and create a new virtual machine named **rhel-vm3**. Run **rhel-vm1** and **rhel-vm3** in the same host as **rhel-vm2**. All the virtual machines are located in the **development** data center.
2. Modify the scheduling policy for the **clusterone** cluster to migrate virtual machines to other hosts in the cluster when more than two virtual machines are running on a host. Configure this scheduling policy to avoid slot reservation on the Storage Pool Manager (SPM) host, and configure a migration threshold of **2**.
3. Verify that one of the three virtual machines running on the same host has migrated to the other available host in the **clusterone** cluster. It may take up to a minute for this migration to be triggered.

Evaluation

On **workstation**, run the **lab migration-review grade** command to confirm success of this exercise.

```
[student@workstation ~]$ lab migration-review grade
```

Finish

On **workstation**, run the **lab migration-review finish** script to complete this lab.

```
[student@workstation ~]$ lab migration-review finish
```

This concludes the lab.

Solution

Migrating Virtual Machines Between Hosts

Performance Checklist

In this lab, you will configure a virtual machine to migrate to another host in the cluster when certain conditions are met.

Outcomes

You should be able to configure a scheduling policy that automatically migrates virtual machines when certain conditions are met.

Before You Begin

This exercise uses the **development** data center, which contains the **clusterone** cluster, and the **hostb** and **hostd** hosts. This exercise also uses the **rhel-vm1**, **rhel-vm2**, and **rhel-vm3** virtual machines, running on either the **hostb** or the **hostd** host.

Log in as the **student** user on **workstation** and run the **lab migration-review start** command. This command ensures that the hosts, clusters, data centers, and networks of the Red Hat Virtualization environment are configured appropriately.

```
[student@workstation ~]$ lab migration-review start
```

1. Clone the **rhel-vm1** virtual machine, and create a new virtual machine named **rhel-vm3**. Run **rhel-vm1** and **rhel-vm3** in the same host as **rhel-vm2**. All the virtual machines are located in the **development** data center.

   1.1. Using the `https://rhvm.lab.example.com` URL log in to the **Administration Portal** of the Red Hat Virtualization (RHV) Manager as **admin** in the **internal** profile. Use **redhat** as the password.

   1.2. From the menu bar, click **Compute**, and then select **Virtual Machines**.

   1.3. Confirm that the value of the **Status** field of the **rhel-vm1** virtual machine is **Down**.

   If its status is **Up**, right-click the row for the **rhel-vm1** virtual machine, and then select **Power Off** to stop the **rhel-vm1** virtual machine. A window titled **Power Off Virtual Machine(s)** displays. Click **OK** to stop the **rhel-vm1** virtual machine. Wait until the value of the **Status** field of the **rhel-vm1** virtual machine transitions to **Down**.

   1.4. On the **Compute >> Virtual Machines** page, select the **rhel-vm1** virtual machine. Click on the three vertical dots next to the **Create Snapshot** button. From the menu that displays, click **Clone VM**. The **Clone Virtual Machine** window displays.

   1.5. Enter **rhel-vm3** in the **Clone Name** field as the name for the new virtual machine. Click **OK** to confirm your intention of cloning the **rhel-vm1** virtual machine as **rhel-vm3**.

It may take up to a couple of minutes to clone the **rhel-vm1** virtual machine and create the **rhel-vm3** virtual machine.

1.6. From the menu bar, click **Events**. Wait until the **VM rhel-vm3 creation has been completed** log message displays. This message displays when the cloning operation is finished.

1.7. From the menu bar, click **Compute**, and then select **Virtual Machines**. View the value of the **Host** field for the **rhel-vm2** virtual machine to determine the RHV Host containing the virtual machine. If **rhel-vm2** is not running, start it.

1.8. On the **Compute >> Virtual Machines** page, select the **rhel-vm1** virtual machine, and then click the drop-down menu next to the **Run** button. From the menu that displays, click **Run Once**.

The **Run Virtual Machine(s)** window displays. Click the + sign to expand the **Host** section. Select the **Specific Host(s)** radio button. From the menu, select the host on which **rhel-vm2** is running. Click **OK** to start the **rhel-vm1** virtual machine.

Apply these instructions for both **rhel-vm1** and **rhel-vm3**.

1.9. Verify that the value of the **Status** field of both **rhel-vm1** and **rhel-vm3** is **Up**. It may take up to a couple of minutes for them to boot.

1.10. Verify that the value of the **Host** field is the same for **rhel-vm1**, **rhel-vm2**, and **rhel-vm3**.

2. Modify the scheduling policy for the **clusterone** cluster to migrate virtual machines to other hosts in the cluster when more than two virtual machines are running on a host. Configure this scheduling policy to avoid slot reservation on the Storage Pool Manager (SPM) host, and configure a migration threshold of **2**.

2.1. From the menu bar, click **Compute**, and then select **Clusters**.

2.2. Right-click the row for the **clusterone** cluster, and then select **Edit**. The **Edit Cluster** window displays.

2.3. In the **Scheduling Policy** section, click the drop-down menu for **Select Policy**. From the menu that displays, select **vm_evenly_distributed**.

2.4. In the **Properties** section, modify the **HighVmCount** property value to **2**, the **SpmVmGrace** property value to **0**, and the **MigrationThreshold** property value to **2**. Leave the other properties and configuration parameters with the default values. Click **OK** to update the scheduling policy for the **clusterone** cluster.

3. Verify that one of the three virtual machines running on the same host has migrated to the other available host in the **clusterone** cluster. It may take up to a minute for this migration to be triggered.

3.1. From the menu bar, click **Compute**, and then select **Virtual Machines**.

3.2. Verify that the RHV Manager migrates either **rhel-vm1**, **rhel-vm2**, or **rhel-vm3** to the other available host in the **clusterone** cluster. It may take up to a minute for the migration to complete.

3.3. After the virtual machine is migrated to the other available host in the **clusterone** cluster, click **Events** in the menu bar.

3.4. Verify that the message associated with the virtual machine migration includes the string **`Migration initiated by system (...)`**.

Evaluation

On **`workstation`**, run the **`lab migration-review grade`** command to confirm success of this exercise.

```
[student@workstation ~]$ lab migration-review grade
```

Finish

On **`workstation`**, run the **`lab migration-review finish`** script to complete this lab.

```
[student@workstation ~]$ lab migration-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- Live migration is the process of moving a virtual machine from one host to another while it is running.
- Red Hat Virtualization supports manual migration of virtual machines between hosts in the same cluster.
- Red Hat Virtualization Manager automatically initiates live migration of all virtual machines running on a host when the host is moved into maintenance mode.
- Red Hat Virtualization supports the configuration of scheduling policies to automatically distribute virtual machines among hosts.

Chapter 10

Managing Virtual Machine Images

Goal	Manage virtual machine snapshots and disk images.
Objectives	• Create, restore, and delete snapshots of virtual machine images. • Import and export virtual machine images between data centers in a Red Hat Virtualization environment.
Sections	• Creating and Using Image Snapshots (and Guided Exercise) • Importing and Exporting Virtual Machine Images (and Guided Exercise)
Lab	Managing Virtual Machine Images

Creating and Using Image Snapshots

Objectives

After completing this section, you should be able to create, restore, and delete snapshots of virtual machine images.

Snapshots of Virtual Machines

A *snapshot* is a view of a virtual machine that includes the operating system and applications on any or all available disks at a given point in time. An administrator may take a snapshot of a virtual machine before making changes to it. This can protect against errors that might have unintended consequences. If there is a problem, the administrator can revert the state of the virtual machine to one recorded by the snapshot.

Red Hat Virtualization allows you to take live snapshots of running virtual machines, as well as snapshots of stopped virtual machines. RHV-M supports several snapshots of a virtual machine, but it can only use a single snapshot at a time.

You can shut down the virtual machine and permanently roll the disk image back to an earlier snapshot. When you do, RHV-M discards all snapshots taken at later points in time. Before committing to the rollback, you can temporarily preview a particular snapshot by booting it to confirm that it is the one you want to use.

Snapshots can also be used to create new virtual machines. You can clone a virtual machine from an older snapshot rather than cloning directly from a current virtual machine. A clone is a copy of that virtual machine on new hardware. Be aware that a clone is likely to have machine specific data and configuration settings from the original virtual machine. This may or may not be desired. As an alternative, you can use a snapshot to create a "sealed template" that can be used to create virtual machines from an image that has had this machine specific information cleared.

Creating a Snapshot of a Virtual Machine

Red Hat Virtualization allows you to manually create snapshots for virtual machines with the Administration Portal. A virtual machine needs the RHV guest agent to create a consistent snapshot.

Snapshots are created in the **Compute** → **Virtual Machines** menu of the Administration console. Right-click the virtual machine to use as the source of the snapshot, and then select **Create Snapshot** from the menu. The **Create Snapshot** window displays. The virtual machines disks are automatically selected. However, one or more disks can be deselected when creating the snapshot. Enter a description for the snapshot in the **Description** field.

If the virtual machine is running, you may select the **Save Memory** check box to save the memory state of the virtual machine in the snapshot. The resulting snapshot looks like a suspended version of the virtual machine. Click **OK** to create the snapshot.

After clicking on **OK**, click on the name of the virtual machine. Then, click on **Snapshots**. Watch the snapshot display in the bottom window. Click on **General**. The status should display **OK**. It may take a minute or so for this value to transition from **Locked** to **OK**.

Rolling Back to an Earlier Snapshot

Once you have a snapshot, you can shut down the original virtual machine and relaunch it using an earlier snapshot. You have the option to *preview* the snapshot before committing to the rollback. In this mode, the virtual machine runs using the virtual image of the snapshot. This allows you to confirm that you have the right snapshot selected.

When you are ready, you can permanently commit to the rollback. The active image for the virtual machine reverts to the snapshot, and all snapshots newer than the one selected for rollback are permanently discarded.

To preview and roll back a virtual machine state using a snapshot, click the **Compute** → **Virtual Machines** menu item in the Administration Portal. Select a virtual machine from the list, and make sure that it has been shut down. Select the **Snapshots** tab for the virtual machine by clicking on the name of the virtual machine, and then clicking on the **Snapshots** tab. Find the snapshot that you want to restore and select it from the list.

To see information about the snapshot, click the **Preview** button. If the snapshot has a saved memory state, then a window opens with a prompt to restore memory. The snapshot moves to the state **IN_PREVIEW**, which indicates that it is ready to run. In **Preview** mode, you are simply trying to decide if you have selected the correct snapshot. If you **Undo** the preview, then any changes you make are discarded. If you **Commit** the snapshot, then changes made in the preview are retained.

To run the preview on a temporary basis, click on **Run** from the **Snapshots** tab. When you have decided whether or not to roll back to that snapshot, you can shut down the virtual machine.

If you have decided to roll back to that snapshot permanently, then click the **Commit** option on the **Snapshots** tab. This rolls the state of your virtual machine back to that snapshot on a permanent basis, and discards any snapshots that were taken more recently. Then, **Run** the virtual machine normally to restart it.

Alternatively, if you decide not to roll back to that snapshot, click the **Undo** option on the **Snapshots** tab. The snapshot changes state from **In Preview** to **OK**, and your original image changes state from **Locked** to **OK**. You can now run the virtual machine normally, or you can try rolling back to a different snapshot.

> **Warning**
> The decision to commit to a particular snapshot is irreversible. The formerly current image state, any snapshot newer than the snapshot you committed, and all unique associated data, is permanently lost.

Cloning a Virtual Machine from a Snapshot

Any existing snapshot can be used to clone a virtual machine. Remember, a clone is a copy of a virtual machine created on new hardware. It can be useful to create a clone from a snapshot instead of a current virtual machine, for example, to make a copy of some older state of that virtual machine. Alternatively, you may want to permanently revert to an earlier snapshot, and you may want to clone later snapshots before they are deleted.

To clone a virtual machine from an existing snapshot, click on **Compute**, then click on **Virtual Machines** in the Administration Portal and select a virtual machine from the list. Select the **Snapshots** tab for the virtual machine by clicking on the name of the virtual machine, then clicking on the **Snapshots** tab. Find the snapshot you want to clone and select it from the list, and then click **Clone**. This opens a new window, **Clone VM from Snapshot**, which is very similar to the **New Virtual Machine** window.

At a minimum, set a **Name** for the cloned virtual machine. You can customize other details as well. Then, click **OK** to create the cloned virtual machine.

Watch the status of the virtual machine on the **Compute** → **Virtual Machines** tab. Once the status switches to **Down**, you may run the new machine.

> **Note**
>
> A cloned virtual machine may still have data from its source image. You might not want this if you are trying to create a new virtual machine with a similar configuration rather than an exact copy.
>
> As an alternative, you can use the snapshot to create a sealed *template* that has been cleared of unique data, and then create virtual machines from that template. A template can be created from a snapshot by using the **Make Template** option instead of **Clone**.
>
> *Chapter 8, Deploying and Managing Virtual Machines* discusses how to create and use templates.

Deleting a Snapshot

To delete a snapshot using the Administration Portal, click on **Compute**, and then click on **Virtual Machines**. Select the row for the virtual machine associated with the snapshot. Click on the name of the virtual machine then select the **Snapshots** tab. Select the snapshot to delete, and then click **Delete**.

The **Delete Snapshot** window opens to confirm that you want to permanently remove that snapshot. If you do, click **OK** to delete the snapshot. In the **Snapshots** tab, verify that the snapshot is no longer displayed. It may take a minute or so to delete the snapshot.

> **References**
>
> Further information is available in the Administrative Tasks chapter of the *Virtual Machine Management Guide* for Red Hat Virtualization; at
> https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#chap-Administrative_Tasks

Guided Exercise

Creating and Using Image Snapshots

In this exercise, you will create and restore virtual machine snapshots.

Outcomes

You should be able to:

- Create a snapshot of a virtual machine.
- Revert a virtual machine to an earlier state using a snapshot.

Before You Begin

Log in to **`workstation`** as the **`student`** user with **`student`** as the password.

On **`workstation`**, run the **`lab snapimex-snapshots start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab snapimex-snapshots start
```

1. Create a snapshot of the **`rhel-vm1`** virtual machine. If there is more than one **`rhel-vm1`** virtual machines, use the one located in **`clusterone`**.

   1.1. Using the `https://rhvm.lab.example.com` URL log into the Administration Portal as **`admin`** with the **`internal`** profile. Use **`redhat`** as the password.

   1.2. In the menu, click **Compute**, then **Virtual Machines**.

   1.3. Right-click the row for the **`rhel-vm1`** virtual machine in the **`development`** data center, and then click **Create Snapshot**. The **Create Snapshot** window displays.

   1.4. Enter **`rhel-vm1-snapshot`** as a description for the snapshot in the **Description** field. Verify that the **`rhel-vm1_Disk1`** disk is selected in the **Disks to include** section. Click **OK** to create the snapshot.

   1.5. In **Compute >> Virtual Machines**, select the row for the **`rhel-vm1`** virtual machine and click on the machine name. Select **Snapshots**. Click on the **`rhel-vm1-snapshot`** snapshot. Click on **General**. Verify that the value of the **Status** field is **OK**. It may take up to a minute to create this snapshot.

2. Modify the Message of the Day (MOTD) on the **`rhel-vm1`** virtual machine.

   2.1. In the menu, click **Compute**, then click on **Virtual Machines**.

   2.2. Highlight the row containing the **`rhel-vm1`** virtual machine by clicking an empty cell in that row. Click on **Run** to start the virtual machine.

   2.3. Right-click the row for the **`rhel-vm1`** virtual machine, and then select **Console**. Accept the opening of the **`console.vv`** file using Remote Viewer by clicking the **OK** button. When the warning box displays, click on **Allow**.

2.4. Log into the **rhel-vm1** virtual machine as **root** using **redhat** as the password.

2.5. Verify that the MOTD file for the **rhel-vm1** virtual machine is empty.

```
[root@localhost ~]# cat /etc/motd
[root@localhost ~]#
```

2.6. Modify **/etc/motd** to contain the string **RHV managed virtual machine**.

```
[root@localhost ~]# vi /etc/motd
RHV managed virtual machine
```

2.7. Log out of **rhel-vm1**.

```
[root@localhost ~]# logout
```

3. Revert **rhel-vm1** back to the **rhel-vm1-snapshot** snapshot.

3.1. In the menu, click **Compute**, and then click on **Virtual Machines**.

3.2. Highlight the row for the **rhel-vm1** virtual machine by clicking on an empty cell, and then select **Shutdown**. The **Shut down Virtual Machine(s)** window displays. Click **OK** to shut down **rhel-vm1**.

3.3. Verify that the value of the **Status** field for the **rhel-vm1** virtual machine is **Down**. You may need to scroll the virtual machine list window to the right. It may take a few seconds for RHV-M to mark **rhel-vm1** as **Down**.

3.4. Click on the name of the **rhel-vm1** virtual machine. A new window opens with the virtual machine configuration.

3.5. Click on the **Snapshots** tab.

3.6. Select the snapshot with the description **rhel-vm1-snapshot**. Click the **Preview** drop-down menu and select **Custom**. The **Custom Preview Snapshot** window displays.

3.7. Select the radio button for the snapshot with the description **rhel-vm1-snapshot**. Verify that the check box for the **rhel-vm1_Disk1** disk is selected. Click **OK** to revert **rhel-vm** to the **rhel-vm-snapshot** snapshot.

3.8. Click on **Compute >> Virtual Machines** at the top of the page.

3.9. In the row for the **rhel-vm1**, verify that the value of the **Status** field transitions from **Image Locked** to **Down**.

3.10. Click on the name of the **rhel-vm1** virtual machine, and then click on **Snapshots**. Click on the **General** menu for the **rhel-vm1-snapshot**. Verify that the value of the **Status** field for the **rhel-vm1-snapshot** snapshot is **IN_PREVIEW**. Click the **Commit** button. When the **Commit Snapshot** window displays, click on **OK**.

3.11. In the menu, click on **Events**. Verify that the message **Commiting a Snapshot-Preview for VM rhel-vm1 has been completed** displays. This confirms that RHV-M successfully restored the **rhel-vm-snapshot** snapshot in the **rhel-vm** virtual machine.

4. Verify that **`rhel-vm1`** properly rolled back to the state in the **`rhel-vm1-snapshot`** snapshot.

4.1. In the menu, click **Compute**, and then select **Virtual Machines**.

4.2. Right-click the row for the **`rhel-vm1`** virtual machine, and then select **Run**. Verify that the value of the **Status** field for the **`rhel-vm1`** virtual machine is **Up**. It may take up to a minute for the **`rhel-vm`** virtual machine to start.

4.3. Right-click the row for the **`rhel-vm1`** virtual machine, and then select **Console**. Log in using **`root`** as the user and **`redhat`** as the password.

4.4. Using the **`cat`** command, verify that the MOTD for the **`rhel-vm1`** virtual machine is empty.

```
[root@localhost ~]# cat /etc/motd
[root@localhost ~]#
```

4.5. Log out of the **`rhel-vm1`** virtual machine.

```
[root@localhost ~]# logout
```

5. Delete the **`rhel-vm1-snapshot`** snapshot.

5.1. In **Virtual Machines**, select the **`rhel-vm1`** virtual machine by clicking on the name. The **`rhel-vm1`** virtual machine is in **`clusterone`**.

5.2. Go to **Snapshots**. Select the **`rhel-vm1-snapshot`** snapshot. Click **Delete**. The **Delete Snapshot** window displays. Click **OK**.

5.3. Confirm that RHV-M removes the **`rhel-vm1-snapshot`** snapshot from the snapshot listing. It may take up to a minute.

Finish

On **`workstation`**, run the **`lab snapimex-snapshots finish`** script to complete this exercise.

```
[student@workstation ~]$ lab snapimex-snapshots finish
```

This concludes the guided exercise.

Importing and Exporting Virtual Machine Images

Objective

After completing this section, you should be able to import and export virtual machine images between data centers in a Red Hat Virtualization environment.

Managing Virtual Machine Images

RHV-M stores virtual machine disk images in data domains. A data domain can be attached to only one data center at a time. However, a single data center can have multiple data domains attached simultaneously.

A disk image is stored in a single data center, but it can be relocated to another. There are multiple relocation methods available, including:

- Moving a disk image for a virtual machine from one data domain to another data domain.
- Exporting virtual machines from one data center and importing them into another data center.
- Importing an existing QCOW2 image from outside RHV into a data domain, and then attaching it to a virtual machine.

The current RHV version can import images directly into data domains, and move data domains from one data center to another. Previous versions of RHV used an *export domain* to export and import images between data domains. Export domains are being deprecated, but are still available for use when needed.

Importing Virtual Machine Images into RHV

The latest RHV version uses the Administration Portal, or the API, to directly import QCOW2 formatted disk images into a data domain. Disk images can then be attached to existing virtual machines in the data center.

To import images with this method, RHV-M must be configured to provide the Image I/O Proxy. When importing, the Administration Portal, or the API, must be able to validate the request. Your browser must import the RHV-M CA certificate, which must be configured to trust its usage for web sites, and also support the necessary HTML5 APIs. The browsers that support specific RHV versions are found in the Red Hat documentation.

To import a virtual machine image using the Administration Portal, click **Storage**, and then click on **Disks** . Select **Upload** → **Start**.

The **Upload Image** window displays. In the window, click **Choose File**, and then select the image to upload from your local system. Specify a size to make the image, an **Alias** for its name, and the **Data Center** and **Storage Domain** to store it in. Click **OK** to start the import.

The image appears on the list in the **Disks** tab. A progress bar displays underneath it as it uploads. When RHV-M finishes the image upload, its status changes to **OK**. Note that it is not yet attached to any virtual machine.

To attach a disk image to an existing virtual machine using the Administration Portal, select **Compute** in the menu bar, and then click on **Virtual Machines** tab. Click on the name of the virtual machine to which you want to attach the image. Select the **Disks** tab for that virtual machine at the bottom of the interface. This lists all disk images attached to the virtual machine.

If the virtual machine has any disks already attached that you want to delete, click **Remove**. A window opens asking you to confirm the disk removal. If you want the disk deleted from the data domain entirely, select the **Remove permanently** check box.

To add the imported disk image, click **Attach**. A window opens, listing all disk images. Select the check box next to the disk image you want to attach. Adjust its **Interface** to use the desired connection protocol. Finally, if this is the boot disk for the system, select the check box under the circled letters **OS** (Operating System). Click **OK** to attach the disk to your virtual machine.

Importing VM Images using Export Domains

An older method to import virtual machine images used *export domains* to import and export virtual machines in Open Virtualization Format (OVF). Export domains were also used to transfer images from one data domain to another in different data centers.

> **Note**
> Export domains have been deprecated in Red Hat Virtualization 4.1, but are currently still available and supported, as they may have practical use cases. For example, the guided exercise following this lecture uses an export domain.

First, an export domain must be created and attached to the data center for the new virtual machine. To create an Export domain, click on the **Storage** menu, and then click on **Domains**. Select **Export** as the domain function.

To upload an image to the Export domain, click on the domain name, and then click on **VM Import**. A list of all available virtual machines to import displays. Click on the virtual machine to import, and then click on **Import**.

The **Import Virtual Machine(s)** window opens. Name the new virtual machine, and then click **OK** to import it into the data domain and the data center.

Exporting VM Images using Export Domains

When RHV-M exports a virtual machine into an export domain, it puts the OVF Package for the virtual machine in a directory structure in that export domain. This directory structure includes two subdirectories: **`images`** and **`master`**. The directories which comprise the OVF Package include an "OVF file" which is named with the **`.ovf`** file extension. This is a descriptor file that specifies the virtual hardware configuration for the virtual machine. The directories also include virtual disk image files for that virtual machine. An OVA file or package is just a TAR archive of the OVF Package directory structure.

If you can directly access the storage for the export domain, this provides an unsupported way to extract virtual machines from Red Hat Virtualization.

The official way to extract images from Red Hat Virtualization is to use its API, which is beyond the scope of this course. For more information, see the Red Hat Virtualization *REST API Guide*.

Virtual-2-Virtual Migration

Virtual-2-Virtual migration, or *V2V*, is a tool to move virtual machines from one data center to another. The *virt-v2v* tool is included in Red Hat Enterprise Linux 7 and Red Hat Enterprise Linux

8. This tool can convert virtual machines for use with Red Hat Virtualization Platform, as well as Red Hat OpenStack Platform and KVM.

Virtual-2-Virtual migration is discussed in more detail in a later chapter.

Moving VM Disks to a New Data Domain

If a particular data domain is getting full or its usage is high, you may want to move some virtual machine disks to another data domain in the data center. You can also export virtual machines to a new data center by moving them into a new data domain, and then moving the data domain to another data center.

Red Hat Virtualization supports the manual migration of virtual machine disks from one data domain to another.

To move virtual machine disks to a new data domain using the Administration Portal, click on the **Storage** menu item, then click on **Disks**. Select the disk(s) associated with the virtual machine and click **Move**.

The **Move Disk(s)** window opens. For each disk, select the destination data domain in the **Target** and the **Disk profile** fields. Click **OK** to move the virtual machine disks to the destination data domain. It may take up to a minute for the virtual machine disks to move.

In the **Disks** window, click on each disk entry. A new window displays providing information about the disk. Click on the **Storage** tab, and then verify that the destination data domain is listed.

Exporting Virtual Machines to a Different Data Center

RHV-M supports the usage of data domains to move virtual machine images between data centers. A data domain must store the virtual machine images. To export a virtual machine between data centers using a data domain in the Administration Portal, click on **Storage**, and then click on **Domains**. Select the row for the data domain. A new window opens with the data domain configuration.

In the new window, go to the **Data Center** tab, and then click **Maintenance** to move the data domain into maintenance mode in the source data center. All virtual machines using the data domain should be powered off to move a data domain into maintenance mode.

A **Storage Domain maintenance** window opens, asking you to confirm that you want to move the data domain into maintenance mode. Click **OK**. When the value of the **Domain status in Data Center** field for the source data center displays **`Maintenance`**, click **Detach** to detach the data domain from the source data center.

A window titled **Detach Storage** opens. Click **OK** to detach the data domain from the source data center. Once detached, the source data center no longer appears in the **Data Center** tab for the data domain configuration details section. In the same tab, click **Attach**.

The **Attach to Data Center** window opens. Select the radio button for the destination data center. Click **OK** to attach the data domain to the destination data center.

In the data domain configuration details section, go to the **VM Import** tab. This tab includes a list of virtual machine images stored in the data domain. Select a virtual machine and click **Import**.

The **Import Virtual Machine(s)** window opens. Select the cluster of the destination data center. Click **OK** to import the virtual machine.

Click **Compute** in the left navigation pane, and then select **Virtual Machines**. Verify that the imported virtual machine is listed, and that its status is **Down**.

References

Further information is available in the "Administrative Tasks" chapter of the *Virtual Machine Management Guide* for Red Hat Virtualization; at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#chap-Administrative_Tasks

Further information is available in the "Virtual Disks" chapter of the *Administration Guide* for Red Hat Virtualization; at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#sect-Virtual_Disks

Guided Exercise

Importing and Exporting Virtual Machine Images

In this exercise, you will export and import a virtual machine.

Outcome

You should be able to export and import a virtual machine using an Export Domain.

Before You Begin

Log in to **workstation** as the **student** user with **student** as the password.

On **workstation**, run the **lab snapimex-virtual start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab snapimex-virtual start
```

1. Add a new export domain to the **development** datacenter using the **172.24.0.8:/exports/data2** NFS share as a back-end.

   1.1. On **workstation**, open Firefox, and then navigate to `https://rhvm.lab.example.com`.

   Click **Administration Portal**. Log into the RHV-M Administration Portal as **admin** in the **internal** profile. Use **redhat** as the password.

   1.2. In the menu, click **Storage**, and then click **Domains**.

   1.3. Click **New Domain** to create a new storage domain. In the **New Domain** window, enter the following configuration settings:

   - Choose **development** for **Data Center**.
   - Choose **Export** for **Domain Function**.
   - Choose **NFS** for **Storage Type**.
   - Use the default value of **Host to Use**.
   - Enter **nfs-move** in the **Name** box.
   - Enter **172.24.0.8:/exports/data2** in the **Export Path** box.

   Click **OK** to create the **nfs-move** storage domain.

   1.4. Verify that the **Cross Data Center Status** is **Active** for the **nfs-move** storage domain. It may take up to a minute to become active.

2. Export the **rhel-vm3** virtual machine to the **nfs-move** export domain.

   2.1. In the menu, click **Compute**, and then click **Virtual Machines**.

   2.2. Verify that the value of the **Status** field for **rhel-vm3** is **Down**.

2.3. Click on the line containing the **rhel-vm3** virtual machine.

2.4. Click on the three vertical dots next to the **Create Snapshot** button. A menu displays. In that menu click on **Export to Export Domain**. The **Export Virtual Machine** window opens. Ignore the two check boxes and click on **OK**.

2.5. Click on **Events** in the menu bar. Note the latest entry in **Events** window. The export process may take some time.

```
Vm rhel-vm3 was exported successfully to nfs-move
```

3. Detach the **nfs-move** export domain from the **development** data center and attach it to the **Default** data center.

3.1. In the menu click on **Compute**, then click on **Data Centers**.

3.2. In the **Storage** tab for the **development** data center, select **nfs-move** and click **Maintenance**. The window titled **Storage Domain maintenance** displays. Click **OK** to confirm your intention of marking **nfs-move** for maintenance. Wait for the status of **nfs-move** to transition to **Maintenance**. It may take a minute for the status to transition from **Active** to **Maintenance**.

3.3. In the **Compute >> Data Centers >> development** page, select **nfs-move** and click **Detach** to detach it from the **development** data center. Click on **OK**.

3.4. From the menu, navigate to **Compute** → **Data Centers** to access the **Compute >> Data Centers** page.

3.5. In the **Compute >> Data Centers** page, click on the name of the **Default** data center.

3.6. In the **Storage** tab for the **Default** data center, click **Attach Export**. The window titled **Attach Export Domain** displays. In the **Attach Export Domain** window, click the radio button next to **nfs-move**. Click **OK** to confirm attaching the export domain. Wait for the activation of **nfs-move** to be complete. It may take a minute for the status to transition from **Locked** to **Active**.

4. Import the **rhel-vm3** virtual machine to the **Default** data center as a new virtual machine.

4.1. In the menu, click **Storage**, and then click on **Domains**.

4.2. Click on the **nfs-move** name. Then click on **VM Import**.

4.3. Click on the **rhel-vm3** virtual machine. Click on the **Import** button.

4.4. The **Import Virtual Machine(s)** window displays. Click on **OK**. Note that next to the name of the virtual machine there is a **Warning**. Using the mouse, hover over the warning. The following message displays: **Invalid name**.

4.5. Click on the name of the virtual machine. A new window displays. Change the name of the virtual machine to **rhel-vm4**. Click on **OK**. The **Import Virtual Machine(s)** pop-up box opens with a message. Click on **Close**.

4.6. Click on **Compute**, and then click on **Virtual Machines**. The **rhel-vm4** virtual machine is imported. Once the import is complete, the virtual machine is in a status of **Down**.

5. The network configuration of an exported virtual machine must be updated. Add a logical network to **rhel-vm4**.

   5.1. In the **Compute >> Virtual Machines** window click on **rhel-vm4**. Click on **Network Interfaces**. Click on **Edit**. The **Edit Network Interface** window displays.

   5.2. In **Profile**, select **ovirtmgmt(ovirtmgmt)**. Click on **OK**. Click on **Run** to start the virtual machine.

   5.3. In the **Compute >> Virtual Machines** window verify that the value of the **Status** field for the **rhel-vm4** virtual machine is **Up**. It may take up to a minute for the **rhel-vm4** virtual machine to start.

   5.4. To confirm that **rhel-vm4** has an IP address. Click on **console**. Log in to the virtual machine using **root** as the username and **redhat** as the password. Run the **ip a** command to confirm the IP address of **eth0**.

```
rhel-test login: root
Password: redhat
[root@rhel-test ~]# ip a
...output omitted...
  inet 172.25.250.21/24 brd 172.25.250.255 scope global noprefixroute dynamic eth0
...output omitted...
```

6. Delete the **rhel-vm4** virtual machine to cleanup resources.

   6.1. Click on **Compute**, and then click on **Virtual Machines**. Right-click the row for the **rhel-vm4** virtual machine, and then select **Power Off**. The **Power Off Virtual Machine(s)** window appears. Click **OK** to power off **rhel-vm4**.

   6.2. Verify that the value of the **Status** field for the **rhel-vm4** virtual machine is **Down**. It may take some time for RHV-M to mark the **rhel-vm4** virtual machine as **Down**.

   6.3. Right-click the row for the **rhel-vm4** virtual machine, and then select **Remove**. The **Remove Virtual Machine(s)** window displays. Verify that the **Remove Disk(s)** checkbox is selected. Click **OK** to delete **rhel-vm4**.

   6.4. Verify that RHV-M removes the **rhel-vm4** virtual machine from the virtual machine listing.

Finish

On **workstation**, run the **lab snapimex-virtual finish** script to complete this exercise.

```
[student@workstation ~]$ lab snapimex-virtual finish
```

This concludes the guided exercise.

Lab

Managing Virtual Machine Images

Performance Checklist

In this lab, you will import a virtual machine image, take a snapshot of a virtual machine, and export a virtual machine.

Outcomes

You should be able to:

- Upload a virtual machine image into Red Hat Virtualization, using the Administration Portal.
- Take a snapshot of an existing virtual machine.
- Use the Administration Portal to export a virtual machine as an OVA file onto an RHV host.

Before You Begin

Log in to **workstation** as the **student** user with **student** as the password.

On **workstation**, run the **lab snapimex-review start** command. This command runs a start script that determines if the Red Hat Virtualization environment is configured and working.

```
[student@workstation ~]$ lab snapimex-review start
```

1. On **workstation**, download the **http://materials.example.com/rhel7-minimal-sealed.qcow2** virtual machine image to the **~/Downloads/** directory, if it does not already exist. Upload **rhel7-minimal-sealed.qcow2** to the **production** data center. Use **rhel7-image** as the alias and **nfs-data** as the storage domain. The size of **3 GB** will be automatically assigned.
2. Create a new virtual machine named **rhel-lab-vm**. Use **clustertwo**. The operating system should use **Red Hat Enterprise Linux 7.x x64**, the instance type should use **Small**, and Optimized for should use **Server**. Attach the **rhel7-image** virtual machine image as a bootable disk. Associate the first network interface with the **prod-vm-net** logical network.
3. Create a snapshot of the **rhel-lab-vm** virtual machine. Use **rhel-lab-vm-snapshot** for the snapshot description.
4. Modify **/etc/motd** on the **rhel-lab-vm** virtual machine to contain the string **RHV managed virtual machine**.
5. Revert the **rhel-lab-vm** virtual machine back to the **rhel-lab-vm-snapshot** snapshot.
6. Verify that the **/etc/motd** file is empty to determine if the **rhel-lab-vm** virtual machine is using the **rhel-lab-vm-snapshot** snapshot.
7. Export the **rhel-lab-vm** virtual machine as an **OVA** file to the **/var/lib/libvirt/images/** directory, on **hostc.lab.example.com**.

Evaluation

On **`workstation`**, run the **`lab snapimex-review grade`** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab snapimex-review grade
```

Finish

On **`workstation`**, run the **`lab snapimex-review finish`** script to complete this lab.

```
[student@workstation ~]$ lab snapimex-review finish
```

This concludes the lab.

Solution

Managing Virtual Machine Images

Performance Checklist

In this lab, you will import a virtual machine image, take a snapshot of a virtual machine, and export a virtual machine.

Outcomes

You should be able to:

- Upload a virtual machine image into Red Hat Virtualization, using the Administration Portal.
- Take a snapshot of an existing virtual machine.
- Use the Administration Portal to export a virtual machine as an OVA file onto an RHV host.

Before You Begin

Log in to **`workstation`** as the **`student`** user with **`student`** as the password.

On **`workstation`**, run the **`lab snapimex-review start`** command. This command runs a start script that determines if the Red Hat Virtualization environment is configured and working.

```
[student@workstation ~]$ lab snapimex-review start
```

1. On **`workstation`**, download the **`http://materials.example.com/rhel7-minimal-sealed.qcow2`** virtual machine image to the **`~/Downloads/`** directory, if it does not already exist. Upload **`rhel7-minimal-sealed.qcow2`** to the **`production`** data center. Use **`rhel7-image`** as the alias and **`nfs-data`** as the storage domain. The size of **`3 GB`** will be automatically assigned.

    1.1. Check to see if **`rhel7-minimal-sealed.qcow2`** already exists in the **`~/Downloads/`** directory.

```
[student@workstation ~]$ ls ~/Downloads/
rhel7-minimal-sealed.qcow2
```

Download the **`http://materials.example.com/rhel7-minimal-sealed.qcow2`** virtual machine image, if necessary.

```
[student@workstation ~]$ wget -nc -P ~/Downloads/ \
http://materials.example.com/rhel7-minimal-sealed.qcow2
...output omitted...
HTTP request sent, awaiting response... 200 OK
Length: 1210712064 (1.1G)
Saving to: '/home/student/Downloads/rhel7-minimal-sealed.qcow2'
...output omitted...
```

1.2. Open the Administration Portal at `https://rhvm.lab.example.com`. If necessary, log in as **`admin`** in the **`internal`** profile. Use **`redhat`** as the password.

1.3. In the menu, click **Storage**, and then click **Disks**.

1.4. Select **Upload** → **Start**. The **Upload Image** window displays.

1.5. In the **Upload Image** window, click **Choose File**, and then select the **`/home/student/Downloads/rhel7-minimal-sealed.qcow2`** virtual machine image file. The value of **3** GB will be automatically assigned in the **Size (GiB)** field. Enter **`rhel7-image`** in the **Alias** text field. Choose **production** for the **Data Center**. Verify that the value of the **Storage Domain** field is **`nfs-data`**. Use the default values for the other fields.

1.6. In a previous guided exercise, you should have downloaded the ovirt-engine certificate. Click the **Test Connection** button to verify this. If clicking the **Test Connection** button returns a green success box, then you are ready to upload. If clicking the **Test Connection** button returns an orange warning box, click the ovirt-engine certificate link within the warning box. Check the box next to **Trust this CA to identify websites** and then click the **OK** button. Once complete, click the **Test Connection** button again. It should return a green success box.

Important

If you accidentally forget to check the box next to **Trust this CA to identify websites**, the following procedure will bring up that window again:

1. Open **Preferences** for Firefox, and then select **Privacy & Security** in the left menu.
2. Scroll down to the **Security** section, and then click the **View Certificates...** button.
3. In the **Certificate Manager** window, scroll down to lab.example.com, click rhvm.lab.example.com.34088 so that it is highlighted, and then click the **Delete or Distrust** button.
4. Back on the **Preferences** tab for **Privacy & Security**, scroll up to the **Cookies and Site Data** section, and then click the **Clear Data...** button.
5. Accept the default selections and click the **Clear** button. Confirm your choice by clicking the **Clear Now** button in the new window that displays.

1.7. Click the **OK** button to start the upload process.

1.8. Verify that the value of the **Status** field for the **`rhel7-image`** virtual machine image is **OK**. It may take up to a minute to upload this image.

2. Create a new virtual machine named **`rhel-lab-vm`**. Use **`clustertwo`**. The operating system should use **`Red Hat Enterprise Linux 7.x x64`**, the instance type should use **`Small`**, and Optimized for should use **`Server`**. Attach the **`rhel7-image`** virtual machine image as a bootable disk. Associate the first network interface with the **`prod-vm-net`** logical network.

2.1. In the menu, click **Compute**, and then click **Virtual Machines**.

2.2. To create a new virtual machine, click **New**. The **New Virtual Machine** window displays.

2.3. In the **New Virtual Machine** window, set the values of the fields according to the following table.

Field	Value
Cluster	**clustertwo**
Operating System Menu	**Red Hat Enterprise Linux 7.x x64**
Instance Type	**Small**
Optimized for	**Server**
Name	**rhel-lab-vm**
Attach Virtual Disks	**rhel7-image** (check the box under OS)
nic1	**prod-vm-net**

To attach the virtual disk, click on the **Attach** button. Select the **`rhel7-image`** image. Make sure to check the box under OS to indicate that this is a bootable disk. Click **OK**.

Click **OK** to create the virtual machine.

2.4. Verify that the virtual machine list includes the **`rhel-lab-vm`** virtual machine.

3. Create a snapshot of the **`rhel-lab-vm`** virtual machine. Use **`rhel-lab-vm-snapshot`** for the snapshot description.

3.1. Right-click the row for the **`rhel-lab-vm`** virtual machine, and then click **Create Snapshot**. The **Create Snapshot** window displays.

3.2. Enter **`rhel-lab-vm-snapshot`** as a description of the snapshot in the **Description** field. Select the virtual disk by clicking on the check box in the **Disks to include** section. Click **OK** to create the snapshot.

3.3. In the **Virtual Machines** tab, click on the name of the **`rhel-lab-vm`** virtual machine. Select the **Snapshots** tab. Click on the **General** link in the **`rhel-lab-vm-snapshot`** row. Verify that the value of the **Status** field is **OK**. It may take up to a minute to create this snapshot.

4. Modify **`/etc/motd`** on the **`rhel-lab-vm`** virtual machine to contain the string **`RHV managed virtual machine`**.

4.1. In the menu, click **Compute**, and then select **Virtual Machines**.

4.2. Right-click the row for the **rhel-lab-vm** virtual machine, and then select **Run**. Verify that the value of the **Status** field for the **rhel-lab-vm** virtual machine is **Up**. It may take up to a minute for the **rhel-lab-vm** virtual machine to start.

4.3. Right-click the row for the **rhel-lab-vm** virtual machine, and then select **Console**. The **Opening console.vv** window appears. Choose **Open with**, and then select **Remote Viewer**. Click **OK** to open the console for **rhel-lab-vm**. Click on **Allow**.

4.4. Log in to the **rhel-lab-vm** virtual machine as **root** using **redhat** as the password.

4.5. Verify that the **/etc/motd** file for the **rhel-lab-vm** virtual machine is empty.

```
[root@localhost ~]# cat /etc/motd
```

4.6. Modify the **/etc/motd** file to contain the string **RHV managed virtual machine**.

```
[root@localhost ~]# echo 'RHV managed virtual machine' > /etc/motd
```

4.7. Power off the **rhel-lab-vm** virtual machine.

```
[root@localhost ~]# systemctl poweroff
```

5. Revert the **rhel-lab-vm** virtual machine back to the **rhel-lab-vm-snapshot** snapshot.

5.1. In the Administration Portal, click **Compute**, and then select **Virtual Machines**.

5.2. Verify that the value of the **Status** field for the **rhel-lab-vm** virtual machine is **Down**. If the **rhel-lab-vm** virtual machine is still running, right-click on the name, and then select **Shutdown**. You may need to scroll the virtual machine list window to the right in order to see the **Status** field. It may take a minute for the virtual machine to shut down.

5.3. Click on the name of the **rhel-lab-vm** virtual machine.

5.4. Click on the **Snapshots** tab.

5.5. Select the **rhel-lab-vm-snapshot** snapshot. Click the **Preview** drop-down menu, and then select **Custom...**. The **Custom Preview Snapshot** window displays.

5.6. Select the radio button for the **rhel-lab-vm-snapshot** snapshot. Select the check box for **rhel7-image**. Click **OK** to preview the **rhel-lab-vm-snapshot** snapshot.

5.7. In the **Snapshots** tab, click on **General** for the **rhel-lab-vm-snapshot** snapshot. Verify that the value of the **Status** field is **IN_PREVIEW**. Click **Commit**. The **Commit Snapshot** window displays. Click on **OK**.

5.8. In the menu, click **Events**. Verify that the message **Committing a Snapshot-Preview for VM rhel-lab-vm has been completed** displays. This confirms that RHV-M successfully restored the **rhel-lab-vm-snapshot** snapshot in the **rhel-lab-vm** virtual machine.

6. Verify that the **/etc/motd** file is empty to determine if the **rhel-lab-vm** virtual machine is using the **rhel-lab-vm-snapshot** snapshot.

6.1. In the menu, click **Compute**, and then select **Virtual Machines**.

6.2. Right-click the row for the **rhel-lab-vm** virtual machine, and then select **Run**. Verify that the value of the **Status** field for the **rhel-lab-vm** virtual machine is **Up**. It may take up to a minute for the **rhel-lab-vm** virtual machine to start.

6.3. Right-click the row for the **rhel-lab-vm** virtual machine, and then select **Console**. The **Opening console.vv** window displays. Choose **Open with**, and then select **Remote Viewer**. Click **OK** to open the console for **rhel-lab-vm**. Click on **Allow**.

6.4. Log in to the **rhel-lab-vm** virtual machine as **root** using **redhat** as the password.

6.5. Verify that the **/etc/motd** file for the **rhel-lab-vm** virtual machine is empty.

```
[root@rhel-lab-vm ~]$ cat /etc/motd
```

6.6. Log out from **rhel-lab-vm**.

```
[root@rhel-lab-vm ~]$ logout
```

7. Export the **rhel-lab-vm** virtual machine as an **OVA** file to the **/var/lib/libvirt/images/** directory, on **hostc.lab.example.com**.

7.1. In the menu, click **Compute** and then click **Virtual Machines**.

7.2. Click the row for the **rhel-lab-vm** virtual machine. Click on the three vertical dots next to **Create Snapshot**, and then select **Export as OVA**.

7.3. The **Export Virtual Machines(s) As Virtual Appliance** window displays. Verify that the **Host** is **hostc.lab.example.com**. In the **Directory** field, enter **/var/lib/libvirt/images/**. Confirm that the **Name** is **rhel-lab-vm.ova**. Click on **OK**. A message displays to confirm that the OVA has been created.

7.4. From **workstation** log in to **hostc.lab.example.com** as the **root** user.

```
[student@workstation ~]$ ssh root@hostc.lab.example.com
 Last login: Tue Aug 13 06:26:29 2019 from 172.25.250.14

 node status: OK
 See `nodectl check` for more information

 Admin Console: https://172.24.0.12:9090/ or https://172.25.250.12:9090/
[root@hostc ~]#
```

7.5. Locate the **rhel-lab-vm.ova** on **hostc.lab.example.com**.

```
[root@hostc ~]# ll /var/lib/libvirt/images
total 1182364
-rw-------. 1 root root 1210857984 Aug 13 06:26 rhel-lab-vm.ova
```

7.6. Log out from **hostc.lab.example.com**.

```
[root@hostc ~]$ logout
```

Evaluation

On **workstation**, run the **lab snapimex-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab snapimex-review grade
```

Finish

On **workstation**, run the **lab snapimex-review finish** script to complete this lab.

```
[student@workstation ~]$ lab snapimex-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- A snapshot saves the state of a virtual machine at a given point in time.
- Snapshots allow administrators to preserve the state of a virtual machine before making changes, and they can revert the virtual machine to that state if the changes are bad.
- A new virtual machine can be cloned from any snapshot.
- RHV-M can import virtual machine images using the Administration Portal.
- RHV-M exports virtual machines to another data center by moving them to a separate data domain, and then reassigning the entire data domain to the new data center.
- Export domains are an obsolete mechanism that can be used to import virtual machines and move them from one data center to another, and can also be used to export virtual machines from the RHV environment.

Chapter 11

Introducing the Infrastructure Migration Solution

Goal	Describe the Infrastructure Migration Solution for migrating production virtual machines and networks from a VMware vSphere environment to Red Hat Virtualization.
Objectives	• Describe how virtual machines and network are migrated from a VMware ESX environment to KVM hypervisors and RHV Manager.
Sections	• Converting Instances with the Infrastructure Migration Solution (and Quiz)

Converting Instances with the Infrastructure Migration Solution

Objectives

After completing this section, you should be able to describe how virtual machines and networking are migrated from a VMware ESX environment to KVM hypervisors and RHV Manager.

Introducing the Infrastructure Migration Solution

Over the last decade, cloud migration rates have increased as organizations recognize the value of re-engineered applications and advanced hybrid cloud architectures. Red Hat customers working toward an infrastructure where workloads and resources span physical, virtual, and cloud-based environments, must migrate away from legacy virtualization solutions.

To help address this need, Red Hat created the Red Hat infrastructure migration solution (IMS), an enterprise-ready utility for migrating VMware ESX-based enterprise workloads to advanced Red Hat technology destinations, including Red Hat Virtualization, Red Hat OpenStack Platform, Red Hat Hyperconverged Infrastructure for Virtualization, and Red Hat Virtualization for Cloud. Workload migration provides the pathway to cloud-native application development via Linux containers, Kubernetes, automation, and other open source technologies.

The Phases of Migration

IMS is designed to safely manage and migrate VMware ESX-based workloads to an open source infrastructure platform. A standard migration consists of three phases:

Discovery Session
: A complementary Discovery Session is scheduled to better understand and document the scope of the migration.

Migration pilot
: One or more chosen open source platforms are deployed and made operational using Red Hat Hybrid Cloud Management infrastructure tooling. Pilot and practice migrations demonstrate typical approaches, establish initial migration capability, and define the resource requirements for a larger scale migration.

Migration at scale
: IT teams perform workload migration at scale. Red Hat Consulting offers design and implementation assistance to build and optimize production infrastructure, unify and streamline operations across virtualization pools, and navigate complex migration cases.

Open Source Platform Destinations

After the Discovery Session, recommendations are provided for a more flexible open source virtualization platform based on Red Hat technologies, including:

Red Hat Virtualization (RHV)
: A software-defined infrastructure and centralized management platform for traditional virtualized Linux and Windows workloads. RHV is also the platform for integrating future cloud-native and container-based applications.

Red Hat OpenStack Platform (RHOSP)
RHOSP builds an on-premise cloud architecture that provides resource elasticity, scalability, and efficient user self-service and management.

Red Hat Hyperconverged Infrastructure (RHHI)
There are two choices: RHHI-V using RHV and Red Hat Gluster Storage, and RHHI-C for cloud using RHOSP and Red Hat Ceph Storage. Each provides consolidated compute, network, and storage resources in a form-factor designed for back office, remote office, and edge computing.

IMS is based on Red Hat management technologies, including Red Hat Ansible Automation and the Red Hat CloudForms management platform, built on top of Red Hat Enterprise Linux and Kernel-based Virtual Machine (KVM) technology. Existing workloads are analyzed and migrated in a controlled manner designed to successfully preserve workload-specific business requirements.

Container Native Virtualization (CNV)

During an infrastructure migration, some applications may run more efficiently in containers, which in some cases can help to eliminate the need for legacy virtual machines. For these scenarios, the Red Hat infrastructure migration solution provides the path to adopt Linux containers and Kubernetes orchestration via Red Hat OpenShift Container Platform as the common hybrid cloud platform. Red Hat OpenShift simplifies the migration of existing applications to Linux containers, enabling organizations to benefit from cloud-native workloads as a final migration destination.

> **Important**
> Container-native Virtualization (CNV) allows legacy virtual workloads to run in container environments, further enabling migration directly from legacy workloads to cloud-native platforms. Learn about CNV and hybrid cloud developments by viewing technical breakout sessions recorded at Red Hat Summit 2019 in Boston.

Red Hat is developing platforms to merge distinctions between virtual machines and containers. Red Hat CNV, based on the open source KubeVirt community project, enables developers to work with VMs in the same way that they work with Linux container-based applications, thus eliminating the current legacy and microservice application development silos.

Preparing For Infrastructure Migration

This section overviews an IMS deployment and virtual machine migration steps. It is not intended to cover all use cases. Typical large migrations require preparation, such as mapping networks and resources in the source environment to the RHV destination. Only a migration to Red Hat Virtualization is discussed, as is appropriate for this course.

The minimum product versions required to run the solution are the following:

- CloudForms 4.7.0+
- Red Hat Virtualization 4.2.7+
- Red Hat Enterprise Linux (Hypervisor) 7.6+
- Red Hat OpenStack Platform 13+
- VMware vSphere 5.5+

The software to install on the machines is obtained from the following Red Hat repositories:

- rhel-7-server-rpms (RHEL 7.6 virt-v2v updates needed)
- jb-eap-7-for-rhel-7-server-rpms (JBoss EAP 7 rpm packages for RHV and for JBoss VMs)
- rhel-7-server-optional-rpms (RHEL 7.6 optional packages)

- rhel-7-server-extras-rpms (RHEL 7.6 extras packages)
- rhel-7-server-supplementary-rpms (RHEL 7.6 supplementary packages)
- rhel-7-server-rhv-4.2-manager-rpms (RHV Manager 4.2 packages)
- rhel-7-server-rhv-4-manager-tools-rpms (RHV Manager 4.2 tools packages)
- rhel-7-server-rhv-4-mgmt-agent-rpms (RHV 4 agents for RHEL)
- rhel-7-server-ansible-2-rpms (Ansible 2.x packages)
- rhel-7-server-rh-common-rpms (RH common packages agents)

Enabling Red Hat Virtualization as a Migration Target

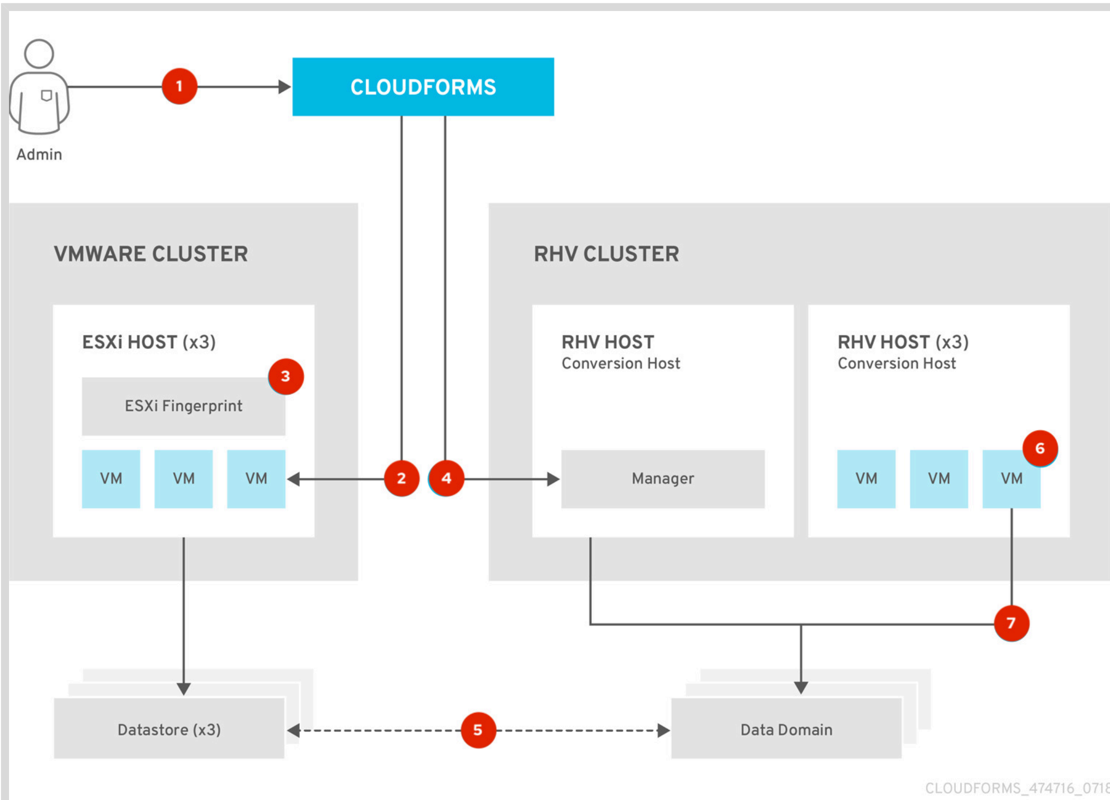

Figure 11.0: Red Hat Virtualization VM migration workflow

1. The Infrastructure Admin creates an infrastructure mapping and a virtual machine migration plan in CloudForms, and runs the migration plan.

2. CloudForms locates the virtual machines to be migrated, based on the infrastructure mapping.

3. The ESXi host fingerprint is captured for authentication during the conversion process if the VDDK transport method is used. If SSH is used, then a shared SSH key is used to connect to the ESX host on which the virtual machine resides.

4. Using the RHV attributes for the target environment, CloudForms initiates communication with the RHV conversion host.

5. The RHV conversion host connects to the source data store through the ESX host, using **`virt-v2v-wrapper.py`**, and streams the disk to be converted to the target data domain that was selected in the infrastructure mapping using**`virt-v2v`**.

6. After the disk is converted, the target virtual machine is created in RHV. During creation, the target virtual machine uses the source metadata from the virtual machine to maintain virtual

machine attributes, such as tags, power state, MAC address, CPU count, memory, disks, and virtual machine name, after migration.

7. After the virtual machine is created, the disk is attached to the target virtual machine. The VM migration is complete. The status is displayed in CloudForms throughout the process.

RHV Conversion Host Requirements

To perform the VM conversion tasks during migration, a conversion host is required. For Red Hat Virtualization, the architectural choice is to use RHEL Hypervisors as conversion hosts.

VDDK SDK
: VMware-vix-disklib-6.5.2-6195444.x86_64.tar.gz (Virtual Disk Development Kit)

nbdkit SRPMS
: rhel-7-server-rhv-4-mgmt-agent-source-rpms (nbdkit Source RPMS)

Add a vSphere Virtualization Provider

In CloudForms, locate the infrastructure providers window to add a new provider. The provider type is **VMware vCenter**, and requires you to enter the vCenter host name and an administrators username and password. Before completing the addition, use the **Validate** button to check for successful access. Once validated, the **Credential validation was successful** message displays. Finally, click the **Add** button finish the addition.

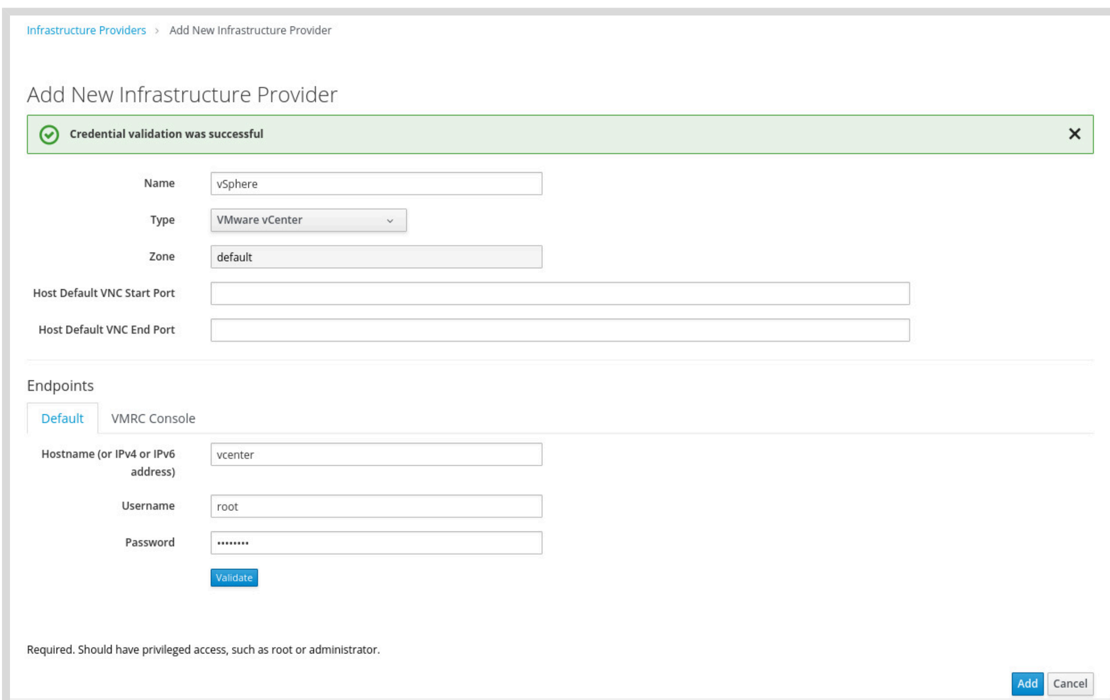

Figure 11.1: Add vSphere to CloudForms

Add an RHV Virtualization Provider

Add a the RHV environment as an infrastructure provider. The provider type is **Red Hat Virtualization**, and requires you to enter the RHV-M host name, and an administrators username and password. Additionally, deactivate the **Verify TLS Certificates** and use the **Validate** button to check for successful access. Once validated, the **Credential validation was successful** message displays. Finally, click the **Add** button to complete the addition.

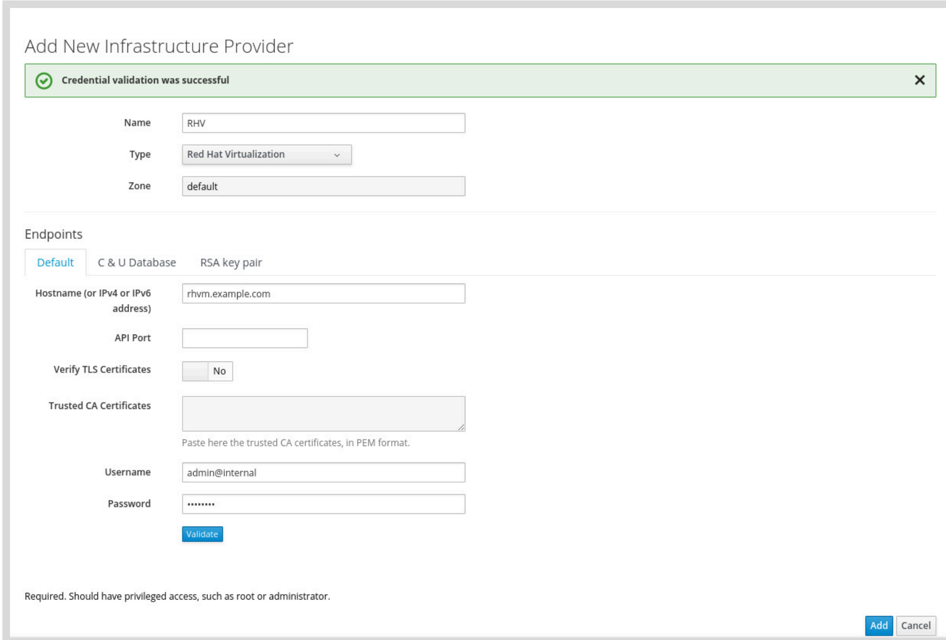

Figure 11.2: Add RHV to CloudForms

Adding Credentials to a Conversion Host in RHV

In CloudForms, existing RHV hosts were discovered by adding the provider. Selected hosts have significant physical resources to perform migration disk conversions. Locate the **Hosts** window, and supply a privileged username and password. Validate successful access, and then save the configuration. Perform this step for each RHV host and ESX host that is involved in the migration.

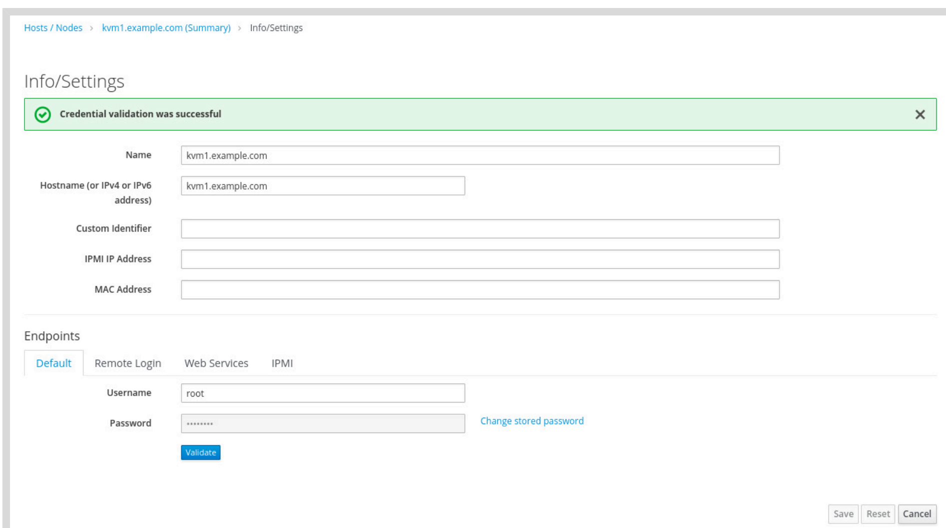

Figure 11.3: Adding credentials to conversion host in RHV

Installing Tools in Conversion Host in RHV

The conversion hosts require additional software installations. An Ansible playbook is provided, which requires an inventory file, similar to the following:

```
[root@workstation ~]# ssh rhvm
[root@rhvm ~]# cd /usr/share/ovirt-ansible-v2v-conversion-host/playbooks
[root@rhvm ~]# cat conversion_hosts_inventory.yml
all:
  vars:
    ansible_ssh_private_key_file: /etc/pki/ovirt-engine/keys/engine_id_rsa
    v2v_repo_rpms_name: "rhel-7-server-rhv-4-mgmt-agent-rpms"
    v2v_repo_rpms_url: "http://storage.example.com/repos/rhel-7-server-rhv-4-mgmt-
agent-rpms"
    v2v_repo_srpms_name: "rhel-7-server-rhv-4-mgmt-agent-source-rpms"
    v2v_repo_srpms_url: "http://storage.example.com/repos/rhel-7-server-rhv-4-
mgmt-agent-source-rpms"
    v2v_vddk_package_name: "VMware-vix-disklib-6.5.2-6195444.x86_64.tar.gz"
    v2v_vddk_package_url: "http://storage.example.com/repos/VMware-vix-
disklib-6.5.2-6195444.x86_64.tar.gz"
    manageiq_url: "https://cf.example.com"
    manageiq_username: "admin"
    manageiq_password: "redhat"
    manageiq_zone_id: "1"
    manageiq_providers:
      - name: "RHV"
        connection_configurations:
          - endpoint:
              role: "default"
              verify_ssl: false
  hosts:
    kvm1.example.com:
    kvm2.example.com:
      v2v_host_type: rhv
      v2v_transport_methods:
        - vddk
      manageiq_provider_name: "RHV"
```

A **conversion_host_check.yml** playbook is provided to verify proper installation. Use the **conversion_host_enable.yml** playbook to install the necessary tools, and then use the host check playbook to verify the installation.

```
[root@rhvm ~]# cd /usr/share/ovirt-ansible-v2v-conversion-host/playbooks
[root@rhvm ~]# ansible-playbook --inventory-file=conversion_hosts_inventory.yml
 conversion_host_enable.yml
[root@rhvm ~]# ansible-playbook --inventory-file=conversion_hosts_inventory.yml
 conversion_host_check.yml
```

When the conversion host software is properly installed, the conversion host is found in the **vmdb** on the CloudForms host. SSH into the CloudForms host and query the database.

```
[root@cf ~]# vmdb
[root@cf vmdb]# rails c
** CFME 5.10.0.29, codename: Hammer
Loading production environment (Rails 5.0.7.1)
irb(main):002:0> pp ConversionHost.all
[#<ConversionHost:0x0000000000a23e88
  id: 1,
```

```
    name: "kvm2.example.com",
    address: nil,
    type: nil,
    resource_type: "Host",
    resource_id: 4,
    version: nil,
    max_concurrent_tasks: nil,
    vddk_transport_supported: true,
    ssh_transport_supported: false,
    created_at: Tue, 15 Jan 2019 14:44:53 UTC +00:00,
    updated_at: Tue, 15 Jan 2019 14:44:53 UTC +00:00,
    concurrent_transformation_limit: nil,
    cpu_limit: nil,
    memory_limit: nil,
    network_limit: nil,
    blockio_limit: nil>]
  => #<ActiveRecord::Relation [#<ConversionHost id: 1, name: "kvm2.example.com",
   address: nil, type: nil, resource_type: "Host", resource_id: 4,
   version: nil, max_concurrent_tasks: nil, vddk_transport_supported: true,
   ssh_transport_supported: false, created_at: "2019-01-15 14:44:53", updated_at:
   "2019-01-15 14:44:53", concurrent_transformation_limit: nil, cpu_limit: nil,
   memory_limit: nil, network_limit: nil, blockio_limit: nil>]>
```

Notice the transport methods and concurrent task settings at the bottom of the output. There are two choices for supported transport methods:

VDDK
VMware Virtual Disk Development Kit is recommended, and uses the ESXi server Network File Copy (NFC) service for the transformation.

SSH
SSH transformation is the fallback option. Configure the VMware hypervisors for passwordless access by sharing a public SSH key for the conversion hosts with the hypervisors.

Adding Ansible Playbooks

The final preparation step is to install the IMS migration and conversion Ansible playbooks onto the CloudForms system. The CloudForms system must be configured to enable the **Embedded Ansible** role on the EVM Configuration screen. It may take up to ten minutes to enable this role, configure the feature, and start the **`Embedded Ansible Worker`** service.

The Ansible playbooks perform the migration tasks by copying virtual machine disks for processing. Configure the credential username and password used by the Ansible playbooks.

Configure CloudForms to have access to the Ansible repository for the playbooks. The repository should be set to update playbooks to the latest stored version each time one is launched. If the repository is correctly configured, then the Ansible Playbooks screen will list the available playbooks.

To make the playbooks available in CloudForms, add a catalog item for the Ansible playbooks. Configure the provisioning by selecting the repository, the playbook, and the machine credentials of the systems to be migrated. The environment is now ready to perform a migration.

Performing an Infrastructure Migration

Infrastructure Mapping

A migration requires a mapping between the source infrastructure resources and the RHV destination. CloudForms provides a wizard to walk through each resource to map. The first resource is the compute hosts, where source and destination clusters are mapped to each other.

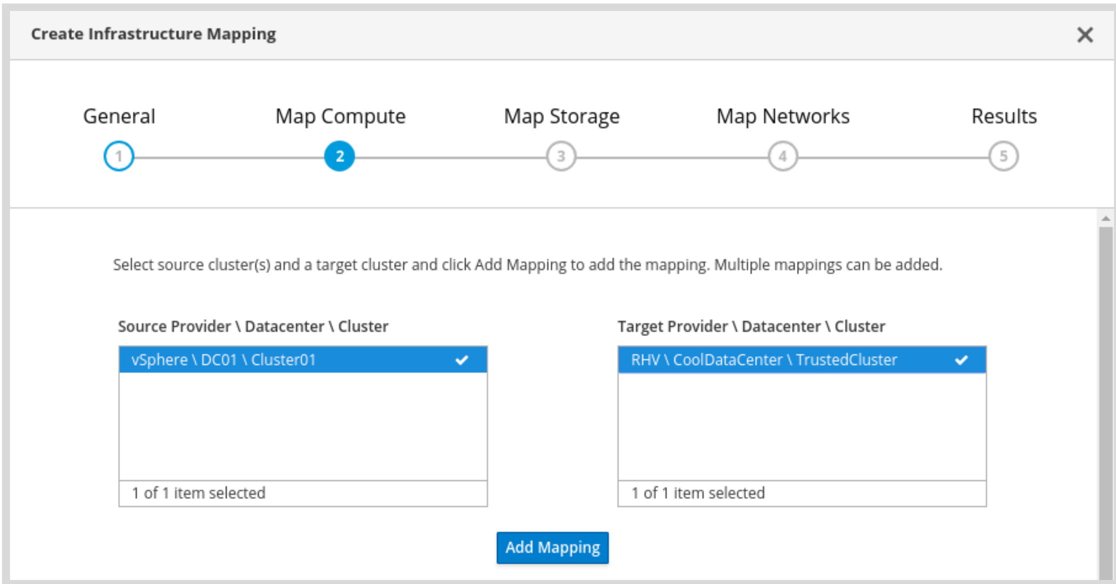

Figure 11.4: Mapping compute hosts by selecting clusters

The next resource is storage, for mapping data stores to RHV data domains.

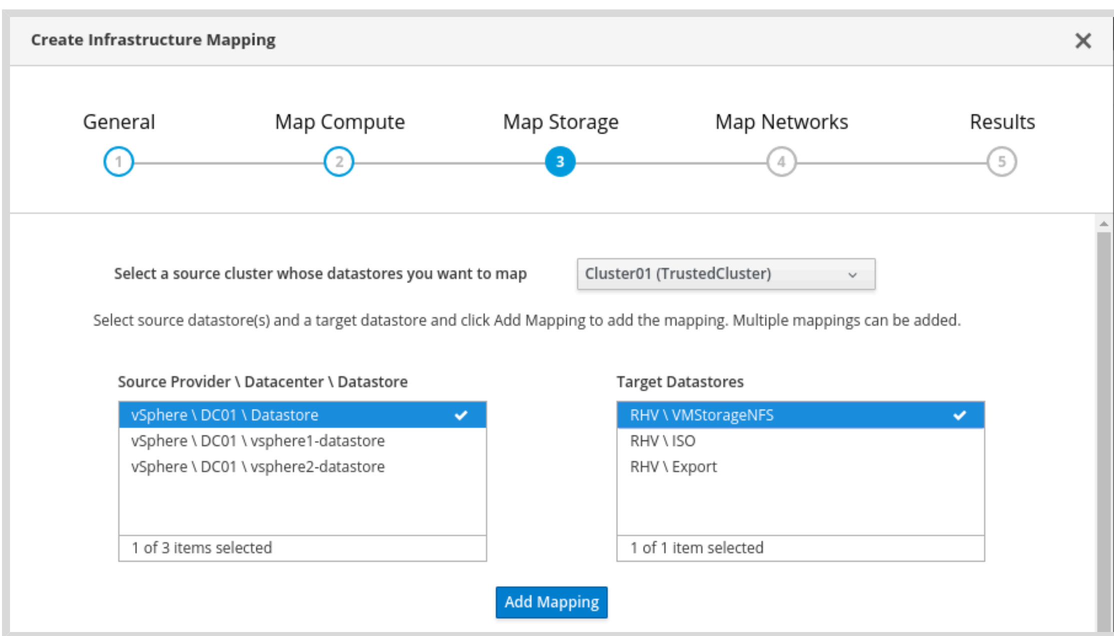

Figure 11.5: Mapping storage by selecting data stores

There will be a map for each network, including VM networks, storage networks, the management network, and any others required by VMs being migrated.

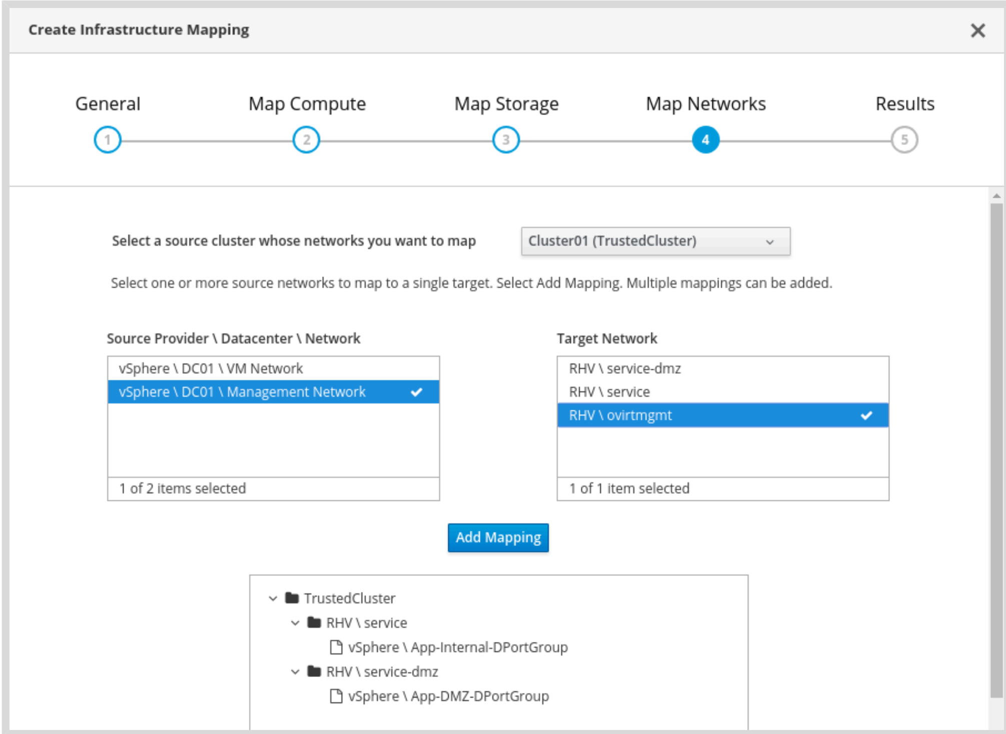

Figure 11.6: Mapping networks by selecting logical networks

Migration Plan

A migration plan is used to select the virtual machines to be migrated. CloudForms provides a selector wizard in the **Migration Plans** screen, with manual section and filters available for locating the correct VMs. Larger lists can be uploaded using a CSV file.

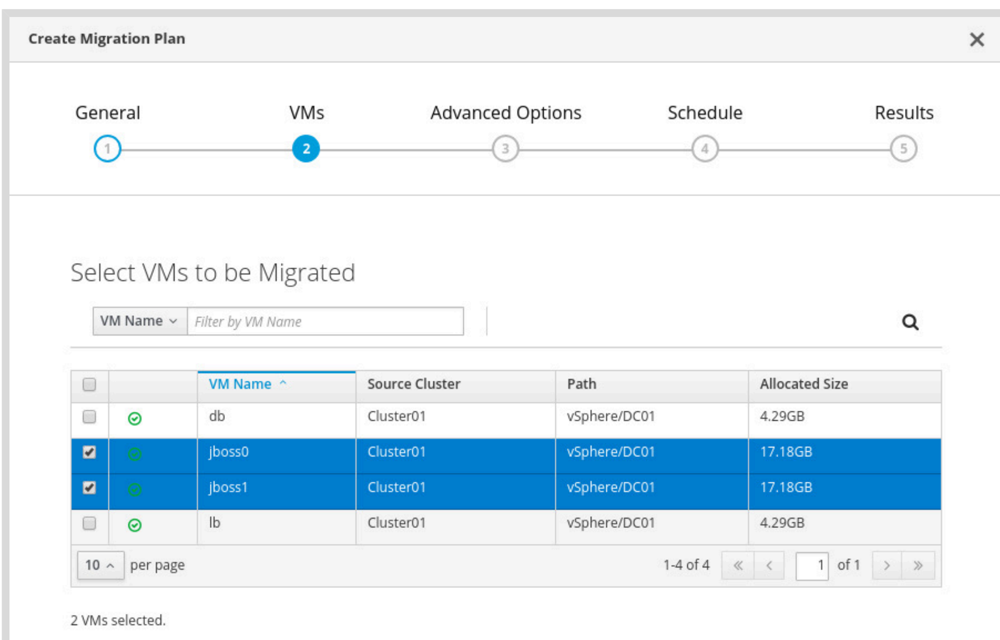

Figure 11.7: Selecting virtual machines

Selected virtual machines can be assigned Ansible playbooks for pre- and post-processing tasks.

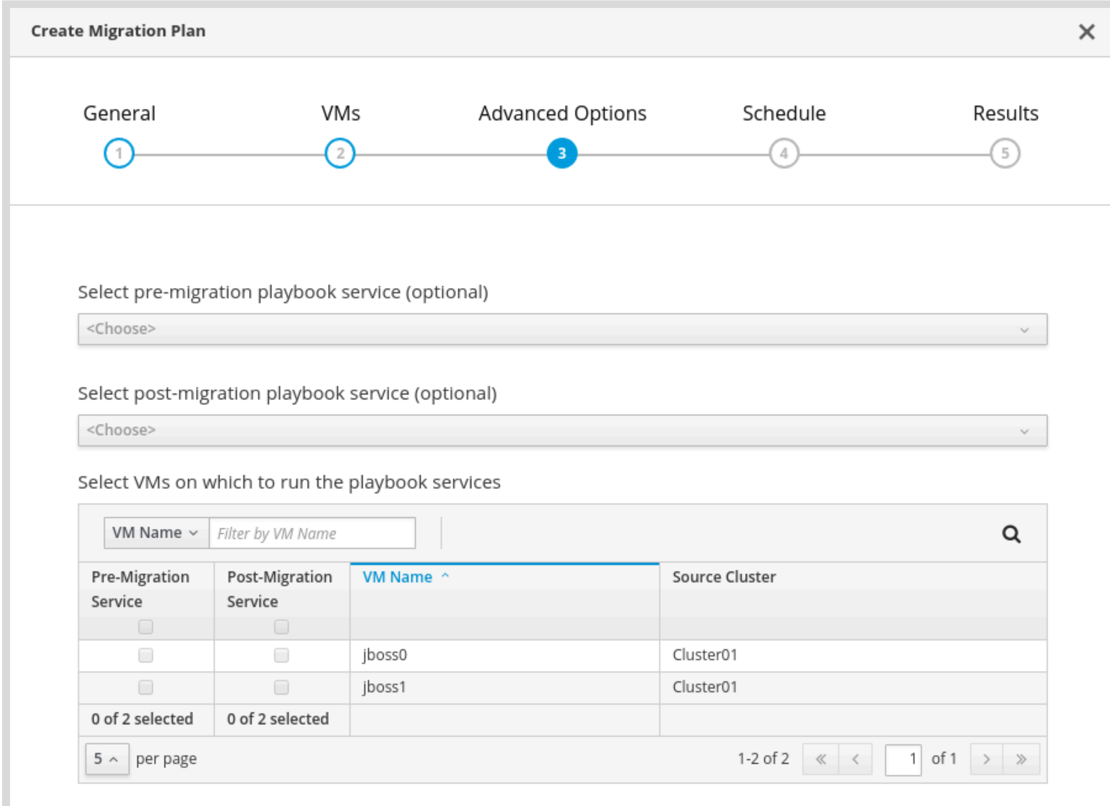

Figure 11.8: Configuring migration options

The migration plan can be run immediately, or saved to launch later. When initiated, the plan gathers data and performs pre-migration checks. Once approved, the migration starts.

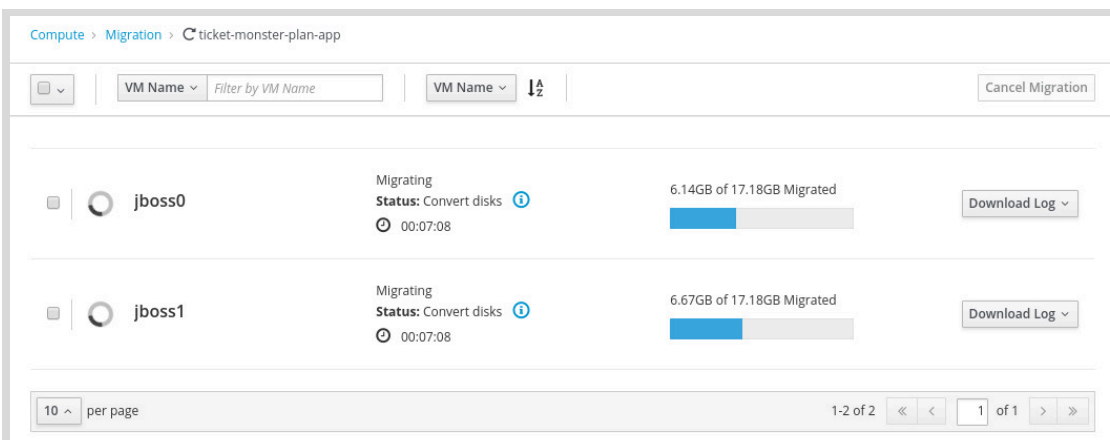

Figure 11.9: Observing the virtual machines in CloudForms

The CloudForms logs contain a record of the orchestration process. See **`/var/www/miq/vmdb/log/automation.log`** on the CloudForms server.

The conversion process for each VM can be tracked in the conversion host logs. See **`/var/log/vdsm/import/v2v-import-*`** on each conversion host.

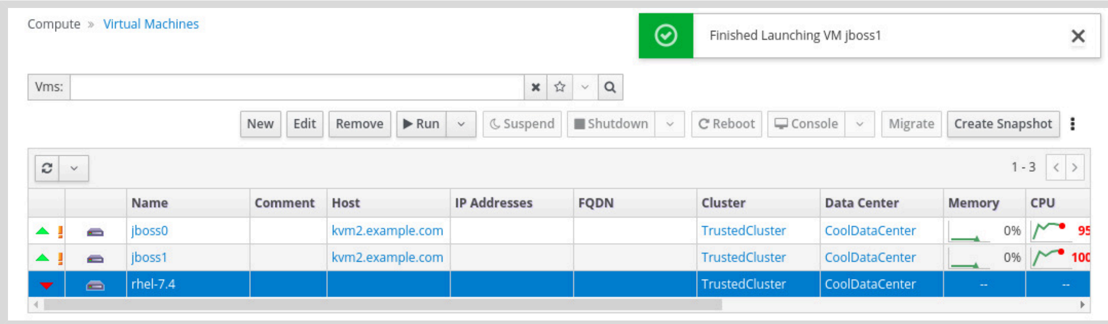

Figure 11.10: Observing the virtual machines in RHV

Progress can also be monitored on the **Migration Plan** screen. Each VM to migrate will be powered off in vSphere, migrated, and then powered on in RHV. As they arrive, VMs can be observed from the RHV Virtual Machines screen. When all VMs have been converted, the migration is complete.

References

Red Hat Summit 2019 session catalog
https://summit.redhat.com/conference/sessions

Why migrate your virtual infrastructure to Red Hat?
https://www.redhat.com/en/topics/virtualization/why-migrate-your-virtual-infrastructure-to-red-hat

Virtualization and Infrastructure Migration Technical Overview
https://www.redhat.com/en/services/training/rh018-virtualization-and-infrastructure-migration-technical-overview

References

For more information, refer to the *Infrastructure Migration Solution Guide* at https://access.redhat.com/documentation/en-us/red_hat_infrastructure_migration_solution/1.2/html-single/infrastructure_migration_solution_guide/index

Quiz

Converting Instances with the Infrastructure Migration Solution

The steps for Red Hat Virtualization VM Migration Workflow are shown below. Indicate the order in which the steps should be taken.

☐ 1. CloudForms locates the virtual machines to be migrated.

☐ 2. The target virtual machine is created in RHV.

☐ 3. CloudForms initiates communication with the RHV conversion host.

☐ 4. The ESXi host is authenticated during the conversion process.

☐ 5. Create an infrastructure mapping and a virtual machine migration plan in CloudForms.

☐ 6. The disk is attached to the target virtual machine.

☐ 7. The RHV conversion host connects to the source data store, and streams the disk to be converted to the target data domain.

Solution

Converting Instances with the Infrastructure Migration Solution

The steps for Red Hat Virtualization VM Migration Workflow are shown below. Indicate the order in which the steps should be taken.

2	1.	CloudForms locates the virtual machines to be migrated.
6	2.	The target virtual machine is created in RHV.
4	3.	CloudForms initiates communication with the RHV conversion host.
3	4.	The ESXi host is authenticated during the conversion process.
1	5.	Create an infrastructure mapping and a virtual machine migration plan in CloudForms.
7	6.	The disk is attached to the target virtual machine.
5	7.	The RHV conversion host connects to the source data store, and streams the disk to be converted to the target data domain.

Summary

In this chapter, you learned:

- Red Hat Infrastructure Migration Solution (IMS) enables you to migrate virtual machines from VMware 5.5 (and later) to Red Hat Virtualization or Red Hat OpenStack Platform, using Red Hat CloudForms.
- Environment resources, such as compute hosts, storage domains, and networks, are mapped from the source infrastructure to the Red Hat Infrastructure.
- The virtual disks are converted with the VMware Virtual Disk Development Kit (VDDK), or by using SSH transformation.

Chapter 12

Managing Red Hat Virtualization Infrastructure

Goal	Back up, restore, and upgrade the software in a Red Hat Virtualization environment.
Objectives	• Backup and restore a Red Hat Virtualization Manager (RHV-M) server. • Perform upgrades and minor updates to Red Hat Virtualization Manager and Red Hat Virtualization Hosts.
Sections	• Backing Up and Restoring Red Hat Virtualization Manager (and Guided Exercise) • Updating and Upgrading Red Hat Virtualization (and Guided Exercise)
Lab	Managing Red Hat Virtualization Infrastructure

Backing Up and Restoring Red Hat Virtualization Manager

Objectives

After completing this section, you should be able to backup and restore a Red Hat Virtualization Manager (RHV-M) server.

Backing Up Red Hat Virtualization Manager

It is important to maintain complete backups of the machine running Red Hat Virtualization Manager, especially when making changes to the configuration of that machine. As part of a backup strategy, the **`engine-backup`** utility can be used to back up the RHV-M database and configuration files into a single archive file that can be easily stored.

> **Warning**
>
> The **`engine-backup`** command only backs up key configuration files, the engine database, and the Data Warehouse database of your RHV-M installation. It does not back up the operating system or installed software. The restore process requires that the RHV-M server has been reinstalled with an operating system and the RHV-M software packages, but that **`engine-setup`** has not yet been run.
>
> In addition, **`engine-backup`** does not backup virtual machines. There are several possible approaches to backing up virtual machines, including using the Backup and Restore API discussed in the Red Hat Virtualization *Administration Guide*.

The **`engine-backup`** command will either perform a backup or restore your backup from archive. Further detailing is available with extended parameters.

Consider the following when adding options:

--mode=mode
: Specifies the operating mode of the command. Two modes are available: backup, which creates a backup, and restore, which restores a backup. This option is required.

--file=backup-file
: Specifies the location of the archive file containing the backup. This option is required.

--log=log-file
: Specifies the location of a file used to record log messages from the backup or restore operation. This option is required.

--scope=scope
: Specifies the scope of the backup or restore operation. There are four scopes:

- **`all`** - backup or restore the engine database, Data Warehouse, and RHV-M configuration files (this is the default option).
- **`db`** - backup or restore only the engine database.
- **`files`** - backup or restore only RHV-M configuration files.
- **`dwhdb`** - backup or restore only the Data Warehouse database.

This course uses a standard RHV-M installation running the engine, engine database, and Data Warehouse on the same server. In this case, **`--scope=all`** may be the easiest way to create backups. The other scopes are most useful for backing up more complex or advanced RHV-M configurations. Note that you can run **`engine-backup`** while RHV-M is running.

When using **`engine-backup`** to restore the database from a backup, there are options that may needed:

`--provision-db`
: Creates a PostgreSQL database for the RHV-M engine on the server being restored. Used when restoring to a fresh installation that has not been setup.

`--provision-dwh-db`
: Creates a database for the Data Warehouse on the server being restored. Used when restoring to a fresh installation that has not been setup.

`--restore-permissions`
: Restores database permissions stored in the backup. Used when restoring to a fresh installation, or when overwriting an installation that was previously set up.

By using the **`engine-backup`** with **`--mode=backup`** option, you will create a .tgz (TAR Archive file). The file can be created in a directory using the **`--file`** option.

The **`tar`** file contains a backup of RHV-M configuration files, the engine database, and the Data Warehouse database. This backup archive should be copied from the RHV-M server to secure storage for later use.

Restoring Red Hat Virtualization Manager

While the process for restoring a backup using the **`engine-backup`** command is straightforward, there are several additional steps when compared to the process for creating a backup. The steps required depend on the destination to which the backup will be restored. For example, the **`engine-backup`** command can be used to restore backups to fresh installations of Red Hat Virtualization, on top of existing installations of Red Hat Virtualization, and using local or remote databases.

> ⚠ **Warning**
>
> Backups can only be restored to environments of the same major release as the backup. For example, a backup of a Red Hat Virtualization version 4.1 environment can only be restored to another Red Hat Virtualization version 4.1 environment. To view the version of Red Hat Virtualization contained in a backup file, administrators can unpack the backup file and read the value in the **`version`** file, located in the root directory of the unpacked files.

The **`engine-backup`** command can be used to restore a backup to a fresh installation of the Red Hat Virtualization Manager. The following procedure must be performed on a machine with the base operating system and required packages for the Red Hat Virtualization Manager have been installed, but the **`engine-setup`** command has not yet been run. This procedure assumes that the backup file can be accessed from the machine on which the backup is to be restored.

Consider the following steps in order to restore a backup:

1. Log in to the machine on which the Red Hat Enterprise Virtualization Manager is installed.
2. Restore the full backup using the **`engine-backup`** command. Since the machine has not been set up, use the **`--provision-db`** option to provision the engine database, the

--provision-dwh-db option to provision the Data Warehouse database, and the **--restore-permissions** option to restore the permissions for the databases.

```
[root@demo ~]# engine-backup --mode=restore --file=backup-file.tgz \
--log=log-file --provision-db --provision-dwh-db --restore-permissions
```

If successful, the following output displays:

```
You should now run engine-setup.
Done.
```

3. Run the command and follow the prompts to configure the engine:

```
[root@demo ~]# engine-setup
```

4. The engine database and configuration files for the Red Hat Virtualization Manager have been restored to the version in the backup.

Overwriting a RHV-M Installation

If you have made environment changes to the RHV-M installation since the last backup, you can discard those changes by running an **engine-cleanup** command. The engine-cleanup prompts for removal of components, stopping the engine service, and removing all installed ovirt data. If you do not want to remove the data, then **engine-cleanup** will abort.

After cleaning the RHV-M setup from the host server, you can then run the **engine-backup** command to restore a full backup, or a database only backup. The tables and credentials are already created, so you do not need to create new ones.

You can omit specific databases by leaving out the **--scope="database"** option when running the **engine-backup** command. After restoring the database, you must run the **engine-setup** command again to reconfigure the RHV-M.

References

Further information is available in the "Backups and Migration" chapter of the *Administration Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#chap-Backups_and_Migration

Guided Exercise

Backing Up and Restoring Red Hat Virtualization Manager

In this exercise, you will back up a RHV Manager installation and restore the RHV Manager from the backup.

Outcome

You should be able to back up and restore the RHV Manager server.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab upgrade-backup start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab upgrade-backup start
```

1. From **workstation**, open a terminal and use **ssh** to log in to **hosta.lab.example.com** as **root**. The **student** user on the **workstation** system is configured with the SSH keys needed to log in to **hosta.lab.example.com** as the **root** user.

```
[student@workstation ~]$ ssh root@hosta.lab.example.com
...output omitted...
```

2. Set the maintenance mode to **global**.

```
[root@hosta ~]# hosted-engine --set-maintenance --mode=global
```

Exit from hosta.lab.example.com.

```
[root@hosta ~]# exit
...output omitted...
```

3. Use **ssh** to log in to **rhvm.lab.example.com** as **root**.

```
[student@workstation ~]$ ssh root@rhvm.lab.example.com
...output omitted...
[root@rhvm ~]#
```

4. Perform a full backup of the RHV environment, including the databases. This backup will be used later to restore the environment.

The following **engine-backup** command operates in backup mode. It performs a full backup, and then saves the backup archive file to **rhvm-backup.tgz** and the backup log file as **backup.log**, both in the current working directory (**/root**).

```
[root@rhvm ~]# engine-backup --scope=all --mode=backup --file=rhvm-backup.tgz \
--log=backup.log
Start of engine-backup with mode 'backup'
scope: all
archive file: rhvm-backup.tgz
log file: backup.log
Backing up:
Notifying engine
- Files
- Engine database 'engine'
- DWH database 'ovirt_engine_history'
Packing into file 'rhvm-backup.tgz'
Notifying engine
Done.
```

5. To restore the backup archive to your RHV Manager server, first you must remove the RHV Manager configuration files and the databases.

   Issue the **engine-cleanup** command to completely clean up the environment. This command executes an interactive environment, taking you through a series of questions with default settings displayed in square brackets.

```
[root@rhvm ~]# engine-cleanup
[ INFO ] Stage: Initializing
...output omitted...
[ INFO ] Stage: Environment customization
Do you want to remove all components? (Yes, No) [Yes]:<ENTER>
...output omitted...
During execution engine service will be stopped (OK, Cancel) [OK]:<ENTER>
All the installed ovirt components are about to be removed, data will be lost (OK,
 Cancel) [Cancel]:OK
...output omitted...

--== END OF SUMMARY ==--

[ INFO ] Stage: Clean up Log file is located at /var/log/ovirt-engine/setup/ovirt-
engine-remove-20171027063123-w605h6.log
[ INFO ] Generating answer file '/var/lib/ovirt-engine/setup/
answers/20171027063653-cleanup.conf'
[ INFO ] Stage: Pre-termination
[ INFO ] Stage: Termination
[ INFO ] Execution of cleanup completed successfully
```

6. You have successfully removed the RHV Manager setup from the **rhvm.lab.example.com** server. To confirm the complete cleanup of RHV environment, open Firefox on **workstation**, and verify that you can no longer access the RHV environment at **rhvm.lab.example.com**.

7. Restore the RHV environment on **rhvm.lab.example.com** using the full backup that you previously created.

7.1. Use the **restore** mode, and specify names for the backup and log files along with the **all** scope.

```
[root@rhvm ~]# engine-backup --scope=all --mode=restore \
--file=rhvm-backup.tgz --log=restore.log --restore-permissions
Preparing to restore:
 - Unpacking file 'rhvm-backup.tgz'
Restoring:
- Files
- Engine database 'engine'
  - Cleaning up temporary tables in engine database 'engine'
  - Updating DbJustRestored VdcOption in engine database
  - Resetting DwhCurrentlyRunning in dwh_history_timekeeping in engine database
  - Resetting HA VM status
------------------------------------------------------------------------------
Please note:

The engine database was backed up at 2019-08-07 20:44:36.000000000 -0700 .

Objects that were added, removed or changed after this date, such as virtual
 machines, disks, etc., are missing in the engine, and will probably require
 recovery or recreation.
------------------------------------------------------------------------------
- DWH database 'ovirt_engine_history'
You should now run engine-setup.
Done.
```

7.2. Run the **engine-setup** command with the **--accept-defaults** option to ensure that the **ovirt-engine** service is correctly configured:

```
[root@rhvm ~]# engine-setup --accept-defaults --offline
[ INFO ] Stage: Initializing
...output omitted...
--== END OF SUMMARY ==--
[ INFO ] Stage: Clean up Log file is located at /var/log/ovirt-engine/setup/ovirt-
engine-setup-20171027064927-pq0980.log
[ INFO ] Generating answer file '/var/lib/ovirt-engine/setup/
answers/20171027065453-setup.conf'
[ INFO ] Stage: Pre-termination
[ INFO ] Stage: Termination
[ INFO ] Execution of setup completed successfully
```

The **--accept-defaults** option automatically uses default answers in questions that have them. Since we are in a closed network, use the **--offline** option.

8. Exit from the RHV Manager instance, and then set the maintenance mode back to normal on **hosta**.

```
[root@rhvm ~]# exit
...output omitted...
[student@workstation ~]$ ssh root@hosta.lab.example.com
```

Using **hosted-engine**, set the maintenance mode back to **none**.

```
[root@hosta ~]# hosted-engine --set-maintenance --mode=none
[root@hosta ~]# exit
...output omitted...
```

9. Confirm that everything is working again, and that the restoration from backup was successful. On **workstation**, open Firefox and log in to the **Administration Portal** as the **admin** user with the **internal** profile. Use **redhat** as the password.

Finish

On **workstation**, run the **lab upgrade-backup finish** script to complete this exercise.

```
[student@workstation ~]$ lab upgrade-backup finish
```

This concludes the guided exercise.

Updating and Upgrading Red Hat Virtualization

Objectives

After completing this section, you should be able to perform upgrades and minor updates to Red Hat Virtualization Manager and Red Hat Virtualization Hosts.

Updating RHV-M

Keeping your Red Hat Virtualization environment updated is a recommended practice. Updates for all Red Hat products are released using the Content Delivery Network. Ensure that all RHV environment components are registered and attached to software entitlements for Red Hat Enterprise Linux and Red Hat Virtualization. This ensures that you can access the updates from the Red Hat Content Distribution Network or from a Red Hat Satellite server.

Distinguishing Between Updates and Upgrades

This section focuses on updates between "minor releases" of Red Hat Virtualization 4.3. In this context, that means updates between different releases of Red Hat Virtualization *4.3*, for example from 4.3.4 to 4.3.7.

When the documentation discusses upgrades between "major releases," it generally considers an update from 4.2 to 4.3 as a "major" release. This is because changes between those versions may involve updating cluster and data center compatibility versions, and may add or change features, among other things.

The Red Hat Virtualization *Upgrade Guide* on the Customer Portal discusses special considerations when upgrading from RHV 4.2 and earlier to RHV 4.3, but this section will not go into detail on upgrading the system.

The Yum Versionlock Method of Software Maintenance

Many major complex Red Hat software products use version locking. RHV-M keeps track of packages that make up the application, and knows which packages need to update together. If package X uses version 2.1, package Y uses 4.2, and updating package X by itself will break the application, then Versionlock will lock out the updating process for these particular packages, which must stay in sync. The **`engine-setup`** program knows the packages and versions that sync together, and will only unlock the packages that can be safely updated. To update the setup packages, run the **`yum update ovirt\*setup\*`** command. When you subsequently run the **`engine-setup`** command, the **`ovirt-engine`** service is stopped, and then downloads and installs the updated packages.

When installing software updates for Red Hat Virtualization Manager, you cannot only run **`yum update`**. To ensure RHV-M is correctly updated without inadvertently installing incompatible versions of the RHV-M packages, many RHV-related packages are protected from updates and are skipped by a normal **`yum update`** command.

Important

A normal **yum update** does not update RHV-M, because the RHV installation locked the RHV-M packages from updates by using the *yum-plugin-versionlock* package. The list of locked packages is in the file **/etc/yum/pluginconf.d/versionlock.list**.

Checking for updates for RHV-M when using a self-hosted engine requires some extra steps. Prior to the update, the RHV environment needs to be placed into global maintenance mode. When the update completes, the RHV environment needs to be taken out of global maintenance mode. These actions can be taken from a terminal on the RHV-H machine that is running the self-hosted engine. You can identify this host from the Administration Portal. Click **Compute** in the menu bar and then select **Hosts**. One of the hosts should have a gold crown icon in its row. Hovering over the crown icon reveals the message, **Running the Hosted Engine VM**. Connect to a terminal on that host and place the RHV environment into global maintenance mode.

```
[root@host ~]# hosted-engine --set-maintenance --mode=global
```

You can check the availability of RHV Manager updates by using the **engine-upgrade-check** command on the RHV-M machine.

If no new updates are available, the **engine-upgrade-check** command outputs this information:

```
[root@demo ~]# engine-upgrade-check
VERB: queue package ovirt-engine-setup for update
VERB: Building transaction
VERB: Empty transaction
VERB: Transaction Summary:
No upgrade is available for the setup package.
Please note that system may not be up to date if engine-setup wasn't executed
 after yum update.
```

If there are packages to update, the command lists them all.

To update the setup packages, issue the **yum update ovirt*setup*** command.

```
[root@host ~]# yum update ovirt\*setup\*
```

With the setup packages updated to the most current version, execute the **engine-setup** command as **root** without arguments. This script updates the Red Hat Virtualization Manager. It stops the **ovirt-engine** service, and then downloads and installs all the updates. During this process, it also creates a backup of the database, performs the update of the database, applies post-installation configuration, and starts the **ovirt-engine** service.

```
[root@host ~]# engine-setup
```

Important

The update process takes time. It must download all the necessary packages and review updates required by the underlying software. Allow time for the process to complete and do not stop the update once initiated.

At the end of this process, update the operating system and any other installed packages by issuing a normal **yum update** command. This ensures that all the latest RHEL OS packages are supporting the latest RHV-M engine packages.

```
[root@host ~]# yum update
```

> **Important**
> If the kernel package was updated during the process, a reboot of the RHV-M server is required.

The RHV environment must be placed back into normal mode. Using the **hosted-engine** command, set the maintenance mode back to **none**.

```
[root@host ~]# hosted-engine --set-maintenance --mode=none
```

Updating Red Hat Virtualization Hosts

Administrators can use the host upgrade manager to update RHV-H hosts directly from the **Administration Portal**. In a large environment with many hosts, using the upgrade manager to update hosts instead of manually updating each host can save time by automating the necessary steps.

For this procedure to work, all the RHV-H hosts must be registered, and attached to the software entitlements for Red Hat Enterprise Linux and Red Hat Virtualization. This enables access to the updates from the Red Hat Content Distribution Network or a Red Hat Satellite server. When the hosts are registered and entitled, the upgrade manager checks for updates automatically and notifies of any available host updates.

> **Important**
> Only RHV-H hosts that are **Up** or **Non-Operational** are checked by the upgrade manager. Hosts in **Maintenance** mode are skipped.

The update manager uses the **yum check-update** command on RHV-H hosts to automatically check for available updates to the RHV-H image. For these automation checks to work, you need to enable the **Red Hat Virtualization Host 7** (**rhel-7-server-rhvh-4-rpms**) repository on the RHV-H hosts. You can do this by logging into each RHV-H host using the **Web Console**. In the **Subscriptions** tab of the **Web Console**, click **Register** to register with your Customer Portal account information. Then, open the **Terminal** tab and run the following command:

```
[root@host ~]# subscription-manager repos --enable=rhel-7-server-rhvh-4-rpms
```

By default, the upgrade manager checks for updates every 24 hours. You can change that setting on the RHV-M server by using the **engine-config** command with the **HostPackagesUpdateTimeInHours** configuration value. Changes made with the **engine-config** command are not applied until you restart the **ovirt-engine** service.

```
[root@host ~]# engine-config -s HostPackagesUpdateTimeInHours=48
```

On RHV-H hosts, the whole image is updated. Only the content of the **`/etc`** and **`/var`** directories are preserved during the update. Any other data is replaced during an update.

During a host update, if migration is enabled at the cluster level, then RHV automatically triggers a migration of running virtual machines to other hosts in the cluster. Before starting the update, ensure that there is more than one host in the cluster. There must be one host available in the cluster to perform Storage Pool Manager (SPM) tasks. Since the RHV-H host that is updated will be put temporarily into **Maintenance** mode, the cluster must have enough memory and other resources on the remaining hosts to support the migrated virtual machines. Otherwise, the virtual machine migration will fail.

Although whole clusters of hosts can be upgraded automatically using the Ansible Playbook method, single host minor upgrades can be manually initiated by a user with sufficient Administration Portal privileges when an upgrade is available.

Locate an intended cluster host in the **Hosts** tab. To determine if an update is available, select the row for that host in the table and choose the **Check for Upgrade** option from the **Installation** drop-down menu. If an upgrade has been made available, the **Upgrade** option on the same **Installation** drop-down menu will become available to select.

After starting an upgrade, the selected host will transition through a series of states, including **`Preparing for Maintenance`**, **`Maintenance`**, **`Installing`**, **`Reboot`**, and **`Unresponsive`**, before finishing with an **Up** state. After virtual machines are migrated away from the host, a minor upgrade is expected to take only minutes to install.

References

Further information is available in the *Updating a Self-Hosted Engine* section of the *Administration Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#Updating_a_self-hosted_engine_SHE_admin

Guided Exercise

Updating and Upgrading Red Hat Virtualization

In this exercise, you will verify that your RHV environment has been updated to the latest version of the software.

Outcome

You should be able to update the **hostb** RHV-H host in your environment.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab upgrade-update start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab upgrade-update start
```

Make sure that the RHV-M environment configured in the previous labs is still working, including the engine, hosts, and resources.

1. From **workstation**, open a terminal and use **ssh** to log in to **hostb.lab.example.com** using the user name **root** and **redhat** as password.

```
[student@workstation ~]$ ssh root@hostb.lab.example.com
...output omitted...
```

2. To update the RHV-H host on your **hostb.lab.example.com** system, you normally would ensure the system is registered with Red Hat Subscription Manager and has the correct entitlements and Yum repositories enabled.

   In this classroom environment, this step has been modified because the classroom might not have access to the Content Distribution Network or to a Red Hat Satellite server. Instead, local Yum repositories have been provided that contain the correct updates.

   Download the **rhvh_updates.repo** file from `http://materials.example.com/yum.repos.d/rhvh_updates.repo`, and then place it in the **/etc/yum.repos.d/** directory to enable those repositories.

```
[root@hostb ~]# curl http://materials.example.com/yum.repos.d/rhvh_updates.repo \
 -o /etc/yum.repos.d/rhvh_updates.repo
```

3. Exit from hostb.

4. On **workstation**, open Firefox and log in to the **Administration Portal** as the **admin** user with the **internal** profile. Use **redhat** as the password.

5. Navigate to the **Compute**+**Hosts** menu option.

6. On the list of available RHV-H hosts, highlight the **hostb.lab.example.com** host. Click the **Installation** button, and then click **Check for Upgrade**.

7. When the **Upgrade Host** window displays, click **OK** to confirm the upgrade check.

   Notice that after a brief pause, a new **Action Item** appears next to the RHV-H host. This new icon is a reminder that an upgrade for this host is available.

8. Highlight the **hostb.lab.example.com** host. Click **Installation** button, and then click **Upgrade**.

9. In the **Upgrade Host** window, click the **OK** button to start the upgrade.

   Wait and watch the as the upgrade procedure takes place. The upgrade process can take some time to complete. Notice that the **hostb.lab.example.com** server status changes to **Preparing for Maintenance**, **Maintenance**, **Installing**, **Reboot**, **Unresponsive**, and finally, to **Up**.

10. When the upgrade process finishes, log out from the **Administration Portal**. This completes the guided exercise.

Finish

On **workstation**, run the **lab upgrade-update finish** script to complete this exercise.

```
[student@workstation ~]$ lab upgrade-update finish
```

This concludes the guided exercise.

Lab

Backing Up and Upgrading Red Hat Virtualization

Performance Checklist

In this lab, you will perform a backup of the Red Hat Virtualization environment and update RHV environment hosts.

Outcome

You should be able to update all RHV-H hosts existing in your environment.

Before You Begin

Log in to **`workstation`** as **`student`** using **`student`** as the password.

On **`workstation`**, run the **`lab upgrade-review start`** command. This command runs a start script that determines if the **`Red Hat Virtualization`** environment is configured and working.

```
[student@workstation ~]$ lab upgrade-review start
```

1. Upgrade the **`hostc.lab.example.com`** RHV-H host in the **`production`** data center.

   Download the `http://materials.example.com/yum.repos.d/rhvh_updates.repo` file to **`hostc`**, and then copy it into **`/etc/yum.repos.d/rhvh_updates.repo`**. This enables the RHV-H repository.

Evaluation

On **`workstation`**, run the **`lab upgrade-review grade`** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab upgrade-review grade
```

Finish

On **`workstation`**, run the **`lab upgrade-review finish`** script to complete this lab.

```
[student@workstation ~]$ lab upgrade-review finish
```

This concludes the lab.

Solution

Backing Up and Upgrading Red Hat Virtualization

Performance Checklist

In this lab, you will perform a backup of the Red Hat Virtualization environment and update RHV environment hosts.

Outcome

You should be able to update all RHV-H hosts existing in your environment.

Before You Begin

Log in to **workstation** as **student** using **student** as the password.

On **workstation**, run the **lab upgrade-review start** command. This command runs a start script that determines if the **Red Hat Virtualization** environment is configured and working.

```
[student@workstation ~]$ lab upgrade-review start
```

1. Upgrade the **hostc.lab.example.com** RHV-H host in the **production** data center.

   Download the http://materials.example.com/yum.repos.d/rhvh_updates.repo file to **hostc**, and then copy it into **/etc/yum.repos.d/rhvh_updates.repo**. This enables the RHV-H repository.

   1.1. From **workstation**, open a terminal and use **ssh** to log in to **hostc.lab.example.com** using the user name **root**.

```
[student@workstation ~]$ ssh root@hostc.lab.example.com
...output omitted...
```

   1.2. To update your RHV-H operating system on your **hostc.lab.example.com** host, you normally would ensure the system is registered with Red Hat Subscription Manager and has the correct entitlements and Yum repositories enabled.

   In this classroom environment, this step has been modified because the classroom might not have access to the Content Distribution Network or to a Red Hat Satellite server. Instead, local Yum repositories have been provided, and contain the correct updates.

   Download the **rhvh_updates.repo** file from **http://materials.example.com/yum.repos.d/rhvh_updates.repo** and place it in the **/etc/yum.repos.d/** directory to enable those repositories.

```
[root@hostc ~]# curl http://materials.example.com/yum.repos.d/rhvh_updates.repo \
 -o /etc/yum.repos.d/rhvh_updates.repo
```

1.3. Exit from hostc.

1.4. On **workstation**, open Firefox and using the `https://rhvm.lab.example.com` URL go to the RHV-M web interface. Click on the **Administration Portal** link and log in to the web interface as the **admin** user with the **internal** profile. Use **redhat** as the password.

1.5. Navigate to **Hosts** by clicking on the **Compute** → **Hosts** menu.

1.6. On the list of available RHV-H hosts, click on the **hostc.lab.example.com** host. In the window that displays, click **Installation**, and then click **Check for Upgrade**.

1.7. When the **Upgrade Host** window displays, click **OK** to confirm the upgrade check.

Notice that after a few moments, a new **Action Item** displays.

1.8. Click on **Upgrade** next to the **Action Item** labelled **A new version is available**.

1.9. In the **Upgrade Host** window, click the **OK** button to start the upgrade.

Return to the **Compute >> Hosts** page.

Wait and watch as the upgrade procedure taking place. Notice that the **hostc.lab.example.com** server status changes to **Preparing for Maintenance**, **Maintenance**, **Installing**, **Reboot**, **Unresponsive**, and finally to **Up**. The upgrade process can take some time to complete.

> **Note**
> If **hostc.lab.example.com** has running virtual machines they will be migrated to another host during the upgrade process. To ensure a smooth upgrade in the classroom environment shut down any running virtual machines.

Evaluation

On **workstation**, run the **lab upgrade-review grade** command to confirm that you have completed this exercise successfully.

```
[student@workstation ~]$ lab upgrade-review grade
```

Finish

On **workstation**, run the **lab upgrade-review finish** script to complete this lab.

```
[student@workstation ~]$ lab upgrade-review finish
```

This concludes the lab.

Summary

In this chapter, you learned:

- As part of a backup strategy, the **`engine-backup`** utility can be used to back up the RHV-M database and configuration files into a single archive file that can be easily stored.
- Administrators can use the host upgrade manager to update RHV-H hosts directly from the **Administration Portal**.
- To ensure RHV-M is correctly updated without inadvertently installing incompatible versions of the RHV-M packages, a number of RHV-related packages are protected from updates and are skipped by a normal **`yum update`** command.

Chapter 13

Implementing High Availability

Goal

Explain procedures to improve the resiliency and reliability of Red Hat Virtualization by removing single points of failure and implementing high availability features.

Objectives

- Describe the methodologies used to implement a resilient RHV infrastructure, including redundant networks, storage, and power management.
- Configure hosts and virtual machines to enable high-availability features and failover in the event of a host failure.

Sections

- Implementing Resilient RHV Infrastructure (and Quiz)
- Configuring Highly Available Virtual Machines (and Quiz)

Implementing Resilient RHV Infrastructure

Objectives

After completing this section, you should be able to describe the methodologies used to implement a resilient RHV infrastructure, including redundant networks, storage, and power management.

Improving Resilience of Red Hat Virtualization

In a mission-critical environment, planning requires the right choice of underlying hardware and infrastructure. Hardware choices for both RHV-M and RHV hosts, and the configuration of all supporting infrastructure including storage and networking, must account for fault tolerance and high availability to limit possible single points of failure.

Fault tolerant hardware is designed to provide redundancy and allow continued operation when individual parts fail. For example, a server with multiple power supplies or network cards, when configured properly, will continue to function even if there is a hardware failure, allowing time to plan replacement.

High availability refers to multiple devices operating as a single entity. In the event of a failure, the alternate device takes over control and continues standard operation, with little or no impact of failure to the end user.

Environment Requirements

A physical data center, supporting a mission-critical environment like Red Hat Virtualization, should be configured in a mission-critical way. This might include:

- Redundant power to all components.
- Redundant power in the data center.
- Redundant network providers to the data center.

Using a Standard Hardware Platform for Hosts in a Cluster

Clusters must support a consistent CPU family, since they are migration domains. To ease troubleshooting of a RHV environment, it is a good practice to use the same vendor and model of server, with the same configuration, for all hosts in a cluster. This limits inconsistency and misconfiguration by making all hosts in the cluster as identical as possible. Homogeneous hardware at the cluster level also provides consistent performance in the environment, especially when virtual machines migrate from one cluster host to another. Make sure that hardware, such as CPUs (family and number), network interfaces, host bus interfaces (HBA), and RAID cards, are the same across all hosts in the cluster. Firmware and BIOS versions should be up-to-date, and running the same version on all hosts in the cluster.

RHV Supporting Infrastructure

Due to the many parts required to provision a resilient RHV environment, it is imperative to account for other systems and services required to maintain a functioning environment.

DNS is critical for RHV to operate correctly. Ensure that forward and reverse name resolution is functioning correctly for hosts, and for the Red Hat Virtualization Manager, and that fully-qualified domain names are used.

A number of services, especially related to authentication and TLS/SSL certificates, are sensitive to time skew issues. Use NTP service to ensure that system clocks are synchronized.

If you are using an external authentication provider for RHV-M users, such as Red Hat Identity Management (FreeIPA) or Microsoft Active Directory, then ensure that the provider is highly available so that users can access the web portals. You can use local **`internal`** authentication profile users, such as the built-in **`admin`** superuser, to provide emergency access in case of an authentication outage.

You must properly test your RHV deployment prior to going into production, identify any single points of failure, and plan accordingly.

Storage Requirements

Selecting the right storage platform will help avoid any unnecessary issues. Good storage performance is critical to the overall performance of the RHV environment. If the storage infrastructure choices are not capable of handling its requirements, then no amount of memory or CPU at the host or VM level can make up for it. Storage infrastructure design should consider the need for data backup, data replication, application workload, and fault tolerance/high availability.

A storage environment for RHV should include the following configuration:

- Redundant Ethernet or Fibre channel (FC) switches for your storage networks.
- If using iSCSI or NFS, then multiple NICs should be used and bonded; 10/40GbE NICs are recommended to improve performance.
- If using a SAN, then multiple HBAs (FC) or initiators (iSCSI) should be used to provide multiple paths to the SAN. Use the same make, model, firmware version, and driver versions in the same systems and clusters, to ensure consistent performance and ease troubleshooting.

  Consider using SAN-based boot if there is already a SAN available to store VMs. This configuration avoids issues related to storage on the host, and improves performance on tasks like hypervisor image cloning, thus speeding up virtual machine deployment times.
- GlusterFS is a scalable network based file system that relies heavily on network performance, requiring high throughput NICs/network devices.

Networking Requirements

A networking infrastructure for RHV should include the following configuration:

- Use redundant network switches.
- Use bonded network interfaces, preferably in LACP mode.

> **Important**
> While Red Hat Enterprise Linux and Red Hat Virtualization support all bonding modes (0-6), logical networks that support VM traffic can only use modes 1, 2, 3, or 4. Modes 0, 5, and 6 do not support the Linux bridge needed for VM networks.

- If using Ethernet, plan at least 10GbE links for VM traffic and any Ethernet storage traffic, to avoid network traffic congestion issues. Use 40GbE links if available, potentially partitioning them using VLANs, as needed.

- Segregate different traffic types, like VM traffic, using Virtual LANs (VLANs). Grant different VLANs priority and available bandwidth based on their traffic, like VM live migration, user-to-VM communication, or communication with the engine.

  Networks for storage and VM live migration generally require high bandwidth, and may also require dedicated networks for performance and security. Bandwidth needed for virtual machine traffic varies depending on your applications. RHV-M management traffic, and console display traffic, is relatively low bandwidth and can use slower networks.

VLANs, 40GbE networking, and advanced quality of service settings in RHV can be used together to efficiently and flexibly manage physical network configuration, while segregating types of traffic and controlling bandwidth appropriately.

Configuring Network Bonds on RHV-H

Configuring two NICs as a bonded interface on RHV-H is simple, and can be configured in the Administration Portal after hosts have been added to Red Hat Virtualization Manager.

Click the **Compute** tab, and then the **Hosts** tab to select a host from the list. Then, select the **Network Interfaces** tab for the host. Click **Setup Host Networks**, just as you would to configure logical networks.

In the window, drag the icon for one physical interface to another interface that you want to bond. This opens the **Create New Bond** window. Select a **Bond Name** and **Bonding Mode**, and then click **OK**. You can treat the new bonded interface just like any other interface, adding and removing logical networks as desired.

> **Important**
>
> Remember to configure your networking hardware as needed to support your bonding mode. For example, the default mode used by RHV, IEEE 802.3ad/LACP (mode 4), requires bonding in that mode to be enabled for the switch ports connected to the participating NICs.
>
> You must also remember to configure your switch ports to permit the correct VLANs to be passed to the interfaces on your hosts.

Host Requirements

RHV supports hosts based on Red Hat Enterprise Linux, as well as Red Hat Virtualization Host. Red Hat Enterprise Linux-based hosts can be useful for environments requiring customization at the OS level, because of hardware support, for example. However, because of the manual configuration and updates performed on those hosts, Red Hat Enterprise Linux based hosts can cause unexpected issues in an RHV environment.

Red Hat recommends Red Hat Virtualization Host as the preferred operating system for hosts, because of the following features:

- Only the required packages and services supporting VMs and the hypervisor are part of RHV-H. This approach reduces operating system overhead. As an additional benefit, it also reduces the overall security "attack surface" by restricting the default configuration.

- The latest version of RHV-H allows you to install additional RPM packages, if you need them, reducing the need for "thick" Red Hat Enterprise Linux-based hosts.
- RHV-H includes the recommended configuration for a RHV host, so it does not require any manual configuration. This approach eliminates issues related to manual configuration of a system.
- RHV-H includes the Cockpit web administration tool, which is pre-installed. This tool improves the troubleshooting of issues related to a host and its VMs.

> **Note**
> Use RHV-H whenever possible to minimize the number of services installed in a host, and to ease troubleshooting. Only use a full host installation to support hardware that requires a certain software feature that is not supported in RHV-H.

A RHV host should also include:

- Available out-of-band (OOB) management to enable features like remote console and power control.
- Up-to-date hardware, firmware, and BIOS.
- Memory scaled to avoid memory swapping, which significantly degrades VM performance.
- RAID configuration of the local boot disks on the host, to reduce the chance of VMs going down due to host failure.

> **Note**
> Configuration of host power management and fencing will be discussed in the next section.

RHV-M Considerations

You should perform backups of RHV-M on a regular basis.

For this course, you used an installation of Red Hat Virtualization Manager, which was installed using the self-hosted prebuilt appliance. This is conceptually the simplest configuration, and provides high availability for the RHV-M services when provided with additional hosts and configured with matching resources.

Although an all-in-one (default) RHV-M installation is the preferred approach for deployment of Red Hat Virtualization Manager, for certain scenarios you may want to run some RHV-M components on separate hosts for higher performance. It is possible to deploy RHV-M components, like the PostgreSQL database, the data warehouse, and the WebSocket proxy, to other hosts. This complicates RHV-M deployment and requires careful thought about redundancy, availability, and backup scenarios.

Another strategy is to deploy the RHV-M engine running as a stand alone server, or as a VM in a separate environment. The remainder of this chapter considers these topics.

References

Further information is available to help with planning: *Best Practices for Red Hat Virtualization 4 Technology Detail* at
https://www.redhat.com/en/resources/best-practice-rhv-technology-detail

Administration Guide - 7.6. Host Resilience at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/administration_guide/index#sect-Host_Resilience

Disaster Recovery Guide at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html/disaster_recovery_guide/index

Quiz

Implementing Resilient RHV Infrastructure

Choose the correct answer(s) to the following questions:

1. **Which three of the following items are recommended when configuring a mission-critical physical data center? (Choose three.)**
   a. Redundant power to all components.
   b. Using hardware from different vendors in the same cluster for diversity.
   c. Redundant power in the data center.
   d. Redundant network providers in the data center.

2. **Which two services are key for avoiding issues in an RHV environment? (Choose two.)**
   a. SMTP
   b. NTP
   c. DNS
   d. PXE

3. **Which three features does RHV-H support? (Choose three.)**
   a. RHV-H supports teaming for aggregated NICs.
   b. RHV-H delivers a minimal number of packages and services to reduce the OS footprint.
   c. RHV-H supports the recommended configuration for RHV hosts.
   d. RHV-H includes the Cockpit web administration tool pre-installed, which eases troubleshooting.

4. **Which three of the following are recommended system configurations for hosts in an RHV environment? (Choose three.)**
   a. Running RHV-H as a highly available virtual machine.
   b. Remote management/out-of-band (OOB)/power management interfaces.
   c. RAID configuration for the host's boot disks.
   d. Memory scaled to avoid memory swapping.

Solution

Implementing Resilient RHV Infrastructure

Choose the correct answer(s) to the following questions:

1. **Which three of the following items are recommended when configuring a mission-critical physical data center? (Choose three.)**
   a. Redundant power to all components.
   b. Using hardware from different vendors in the same cluster for diversity.
   c. Redundant power in the data center.
   d. Redundant network providers in the data center.

2. **Which two services are key for avoiding issues in an RHV environment? (Choose two.)**
   a. SMTP
   b. NTP
   c. DNS
   d. PXE

3. **Which three features does RHV-H support? (Choose three.)**
   a. RHV-H supports teaming for aggregated NICs.
   b. RHV-H delivers a minimal number of packages and services to reduce the OS footprint.
   c. RHV-H supports the recommended configuration for RHV hosts.
   d. RHV-H includes the Cockpit web administration tool pre-installed, which eases troubleshooting.

4. **Which three of the following are recommended system configurations for hosts in an RHV environment? (Choose three.)**
   a. Running RHV-H as a highly available virtual machine.
   b. Remote management/out-of-band (OOB)/power management interfaces.
   c. RAID configuration for the host's boot disks.
   d. Memory scaled to avoid memory swapping.

Configuring Highly Available Virtual Machines

Objectives

After completing this section, you should be able to configure hosts and virtual machines to enable high availability features and failover in the event of host failure.

High Availability for Virtual Machines

A high availability virtual machine is automatically restarted if it crashes, or if its host becomes nonresponsive. When these events occur, RHV-M automatically restarts the high availability virtual machine, either on its original host or on another host in the cluster.

Red Hat Virtualization Manager constantly monitors hosts and storage to detect hardware failures. With high availability, interruption to service is kept to a minimum because RHV-M restarts virtual machines that are configured to be highly available within seconds, with no user intervention required.

Configuring high availability is a recommended practice for virtual machines running critical workloads.

Virtual machines may be configured to automatically restart if the host becomes nonresponse, or if the virtual machine unexpectedly crashes. To use this feature, all hosts in the cluster must support an out-of-band power management system, such as iLO, DRAC, RSA, or a network-attached remote power switch that is configured to act as a fencing device.

RHV-M can also automatically restart high priority virtual machines first. Multiple levels of priority give the highest restart priority to the most important virtual machines.

> **Note**
> An alternate method of handling high availability is with a cluster configured using Pacemaker. RHV-M high availability should not be enabled with Pacemaker as the fencing methods will conflict.

Fencing Hosts for VM Integrity

A virtual machine must never be running on two hosts at the same time, or its disk image is likely to become corrupt, leading to data loss. To avoid this issue, Red Hat Virtualization uses an out-of-band management agent to *fence* a nonresponsive host. The agent forces a power off, ensuring that the host and its virtual machines are truly down. Only then will it reboot the virtual machine on a new host.

A host is nonresponsive when RHV-M cannot communicate with it. RHV-M uses fencing to ensure that highly available virtual machines, running on a nonresponsive host, are stopped. Then, RHV-M restarts them on a different host in the cluster.

Red Hat Virtualization 4 and later also support the usage of a special storage volume as a lease, to control whether virtual machines boot on another host when the original host goes down unexpectedly. This feature also prevents two instances of the same virtual machine from running concurrently on different hosts.

> **Important**
>
> There is an important distinction between a **`Non-Operational`** host and a **`Non-Responsive`** host.
>
> A nonoperational host has encountered a problem, but RHV-M can still communicate with it. RHV-M works with the host to migrate any virtual machines running on that host to operational hosts in the cluster. Likewise, a host that is moved to **`Maintenance`** mode automatically migrates all its virtual machines to other operational hosts in the cluster.
>
> A nonresponsive host is one that is not communicating with RHV-M. After about 30 seconds, RHV-M fences that host and restarts any highly available virtual machines on operational hosts in the cluster.

Configuring a Fence Agent in a Host

RHV-M uses a *fence agent* to fence nonresponsive hosts. It does not do this directly, but uses VDSM to send power management requests to a *fencing proxy*, which is one of the other hosts in the same cluster or data center as the nonresponsive host. That host communicates with the fence agent to execute the power management request.

The **Power Management** tab in the **Edit Host** and **New Host** windows includes the power management configuration options for a host.

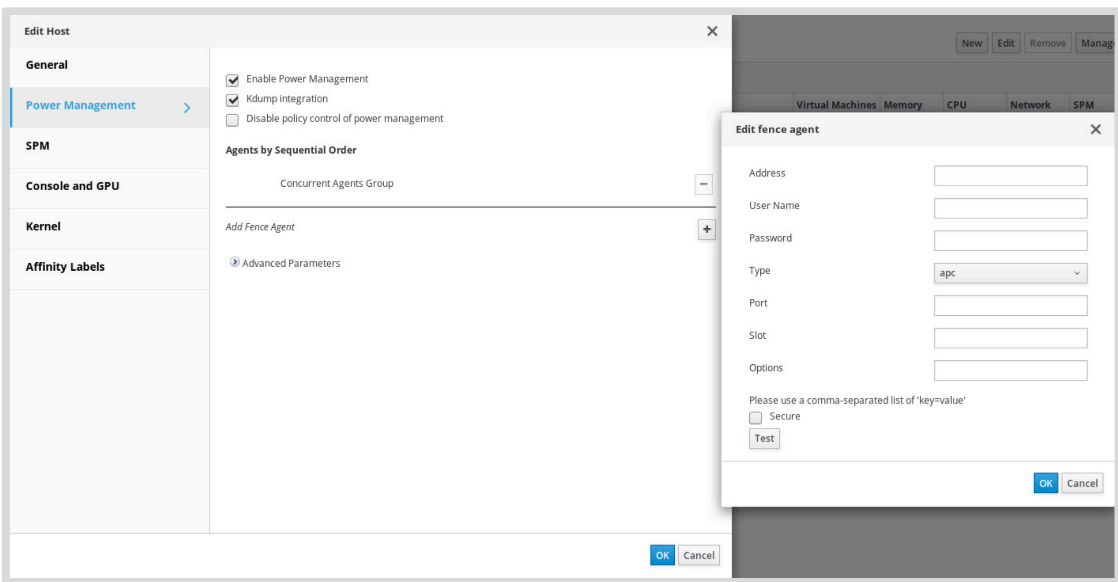

Figure 13.1: Configuration for host high availability

The configuration options included in the **Power Management** tab include:

- The **Enable Power Management** check box enables power management for the host.
- The **Kdump integration** check box disables host fencing while a kernel crash dump completes.
- The **Disable policy control of power management** check box disables the cluster scheduling policy for the host.

- The plus (**+**) button opens the **Edit fence agent** window, to configure a new fence agent for a host. This configuration includes parameters like the IP address of the Remote Access Card (RAC), and the username and password used to log in.
- The **Advanced Parameters** section specifies the search order for a proxy in the cluster and data center for the host.

Configuring a Highly Available Virtual Machine

Virtual machines are configured to be highly available on an individual basis. This configuration can be done when creating the virtual machine, or you can edit an existing VM to enable high availability.

The **High Availability** tab in the **Edit Virtual Machine** window includes the high availability configuration options for a virtual machine. To open the **Edit Virtual Machine** window, right-click on the virtual machine list item, and then click **Edit**.

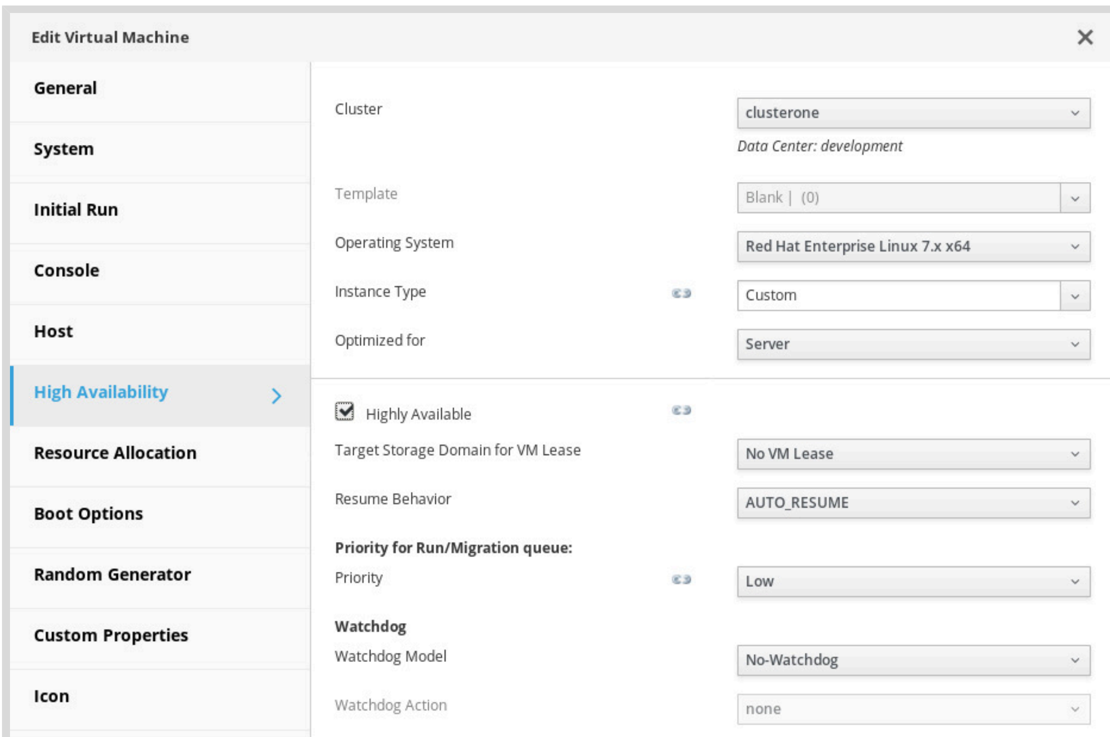

Figure 13.2: Configuration for virtual machine high availability

The configuration options included in the **High Availability** tab include:

- The **Highly Available** check box enables high availability for the virtual machine.
- The **Target Storage Domain for VM Lease** drop-down menu specifies whether or not to use a storage lease to control whether a virtual machine boots on another host when the original host goes down unexpectedly. To use a storage lease, you must configure a storage domain for the lease.
- The **Priority** drop-down menu sets the priority of the virtual machine in the migration queue.

If the following conditions are met, then highly available virtual machines successfully restart when their host becomes nonresponsive:

- Power management is available for the hosts running the highly available virtual machines.

- The host running the highly available virtual machine is part of a cluster that has other available hosts.
- The destination host is running.
- The source and destination hosts have access to the data domain on which the virtual machine resides.
- The source and destination hosts have access to the same virtual networks and VLANs.
- There are enough CPUs on the destination host that are not in use to support the virtual machine requirements.
- There is enough RAM on the destination host that is not in use to support the virtual machine requirements.

References

Further information on highly available virtual machine configuration is available in the "Improving Uptime with Virtual Machine High Availability" section of the *Administration Guide* for Red Hat Virtualization at
https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html-single/virtual_machine_management_guide/index#sect-Improving_Uptime_with_Virtual_Machine_High_Availability

Quiz

Configuring Highly Available Virtual Machines

Choose the correct answer(s) to the following questions:

1. **Which of the following conditions automatically trigger a *restart* (not a migration) of a highly available virtual machine?**
   a. Moving the host on which the virtual machine runs into maintenance mode.
   b. Powering off from within the virtual machine.
   c. Sending the shutdown command from RHV-M.
   d. Moving the data domain containing the virtual machine disks into maintenance mode.
   e. The virtual machine host becoming nonresponsive because of a hardware failure.

2. **When does RHV-M consider a host nonresponsive?**
   a. RHV-M can communicate with the host.
   b. The host has a problem or misconfiguration but RHV-M can communicate with it.
   c. RHV-M cannot communicate with the host.
   d. The host is down.

3. **What must be configured by all hosts to support highly available virtual machines?**
   a. Power management.
   b. RHV-H as an operating system.
   c. Pacemaker.
   d. Highly available RHV-M.

4. **Which new feature is supported by RHV 4 and later to enforce VM high availability?**
   a. Snapshots.
   b. Storage leases.
   c. Migration policy.
   d. Resilience policy.

5. **Which two parameters are included in a fence agent configuration for a host? (Choose two.)**
   a. Host Remote Access Card (RAC), user name, and password.
   b. Host IP address.
   c. Host Remote Access Card (RAC) IP address.
   d. Fencing policy.

Solution

Configuring Highly Available Virtual Machines

Choose the correct answer(s) to the following questions:

1. **Which of the following conditions automatically trigger a *restart* (not a migration) of a highly available virtual machine?**
   a. Moving the host on which the virtual machine runs into maintenance mode.
   b. Powering off from within the virtual machine.
   c. Sending the shutdown command from RHV-M.
   d. Moving the data domain containing the virtual machine disks into maintenance mode.
   e. The virtual machine host becoming nonresponsive because of a hardware failure.

2. **When does RHV-M consider a host nonresponsive?**
   a. RHV-M can communicate with the host.
   b. The host has a problem or misconfiguration but RHV-M can communicate with it.
   c. RHV-M cannot communicate with the host.
   d. The host is down.

3. **What must be configured by all hosts to support highly available virtual machines?**
   a. Power management.
   b. RHV-H as an operating system.
   c. Pacemaker.
   d. Highly available RHV-M.

4. **Which new feature is supported by RHV 4 and later to enforce VM high availability?**
   a. Snapshots.
   b. Storage leases.
   c. Migration policy.
   d. Resilience policy.

5. **Which two parameters are included in a fence agent configuration for a host? (Choose two.)**
   a. Host Remote Access Card (RAC), user name, and password.
   b. Host IP address.
   c. Host Remote Access Card (RAC) IP address.
   d. Fencing policy.

Summary

In this chapter, you learned:

- In a mission critical RHV deployment, the environment in the physical data center should be configured in a mission critical way.
- Storage, networking, and hardware configuration choices can affect the resilience and reliability of a Red Hat Virtualization environment.
- The environment must be correctly configured for installation before deploying and configuring Red Hat Virtualization. The associated documentation provides guidance on required configuration settings.
- Highly available virtual machines are configured to be automatically restarted when they crash, or when their host becomes nonresponsive.
- A nonresponsive host is one with which RHV-M cannot communicate.

Chapter 14

Introducing Red Hat Hyperconverged Infrastructure for Virtualization

Goal	Describe how to install and maintain a minimal Red Hat Hyperconverged Infrastructure (RHHI-V) environment.
Objectives	• Describe how to install Red Hat Virtualization Manager and Red Hat Gluster Storage as a self-hosted, hyperconverged deployment on Red Hat Virtualization Hosts. • Perform maintenance tasks to manage RHHI-V storage, high availability, and geo-replication.
Sections	• Deploying Red Hat Hyperconverged Infrastructure for Virtualization (and Quiz) • Maintaining Red Hat Hyperconverged Infrastructure for Virtualization (and Quiz)

Deploying Red Hat Hyperconverged Infrastructure for Virtualization

Objectives

After completing this section, you should be able to describe how to install Red Hat Virtualization Manager (RHV-M) and Red Hat Gluster Storage as a self-hosted, hyperconverged deployment on Red Hat Virtualization Hosts.

Introducing Red Hat Hyperconverged Infrastructure for Virtualization (RHHI-V)

Hyperconverged infrastructure has become increasingly popular for a broad range of use cases. Hyperconverged combines both compute and storage resources, and makes them simultaneously available among all hosts in a single, scalable, virtualization installation known as a *pod*. Storage is pooled across all hosts in a pod, eliminating the need for a storage area network, and is managed by the same Red Hat Virtualization management software used for standard RHV deployments.

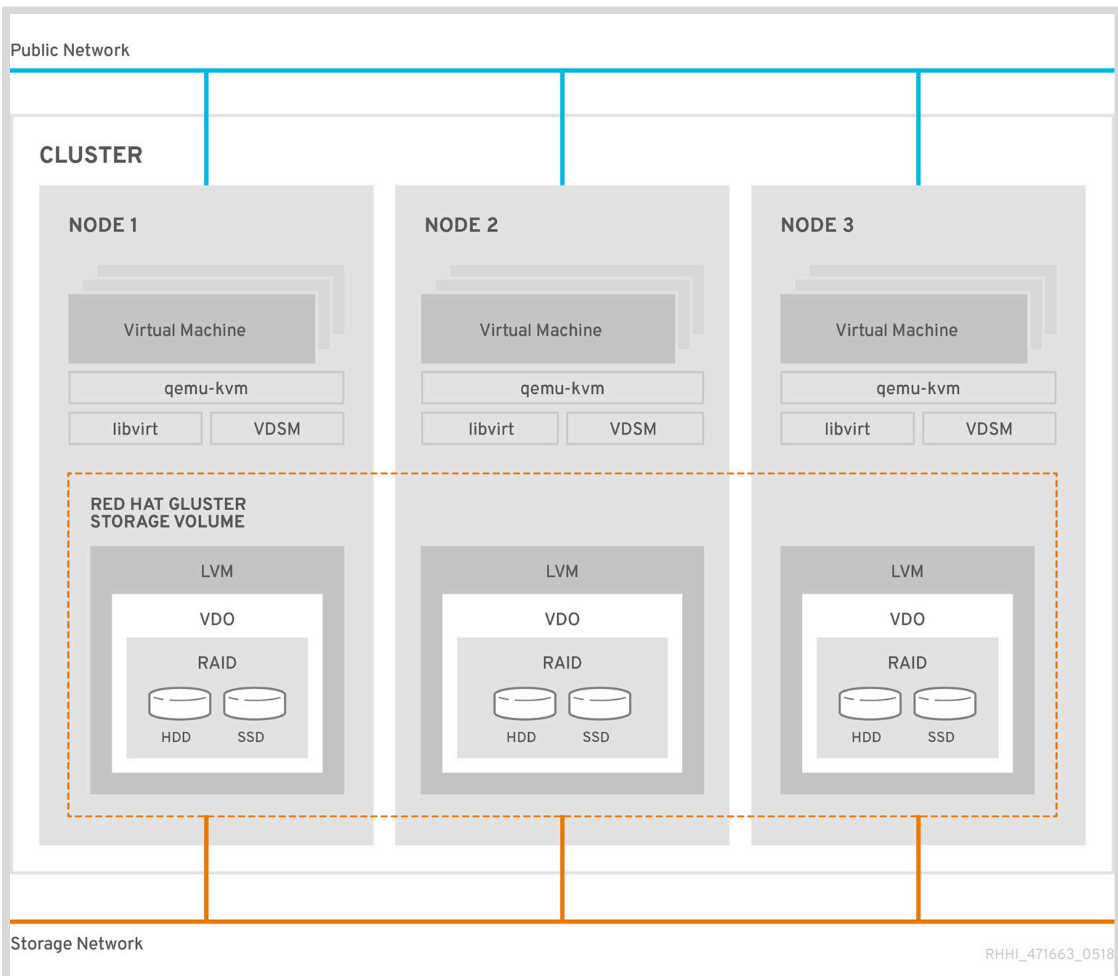

Figure 14.1: The RHHI-V architecture of a single pod

To scale, add more hosts to increase both the compute capacity and the storage space. Because RHHI-V is a software solution instead of a hardware appliance, resource scaling can be configured to match application requirements. Increase only disk space by expanding the storage pools on each host, without adding unnecessary compute capacity. Add more hosts to increase compute capacity, but include only the storage required to implement pool redundancy per cluster.

RHHI-V is a tuned combination of RHEL, Red Hat Virtualization, and Red Hat Gluster Storage, with an Open Virtual Network (OVN) software-defined networking stack, and Red Hat Ansible Automation for provisioning. With OVN, RHHI-V integrates with Red Hat OpenStack Platform and Red Hat OpenShift Container Platform infrastructures for a single hybrid cloud platform. RHHI-V also includes the Virtual Data Optimizer (VDO), which includes these storage enhancements:

- **zero-block elimination** - records zero-filled blocks as metadata only
- **deduplication** - redundant blocks are eliminated by using a pointer to the original block
- **LZ4 compression** - applied dynamically to individual data blocks

> **Note**
> VDO is supported only when enabled on new installations at deployment time, and cannot be enabled on deployments upgraded from earlier versions of RHHI for Virtualization. Additionally, thin provisioning is not currently compatible with VDO. These two technologies are not supported on the same device.

RHHI-V is offered as a solution for remote office, back office, and edge computing. However, it is also enjoying rapid acceptance in all data center sizes, as an easily managed and scaled mission-critical workload platform, primarily because RHHI-V deployments can be right-sized.

Installing a RHHI-V Pod

The workflow for deploying Red Hat Hyperconverged Infrastructure for Virtualization is as follows:

- **Install Red Hat Virtualization hosts** - install the physical machines as hyperconverged hosts
- **Configure SSH access** - configure passwordless key-based, SSH authentication between hosts
- **Configure Red Hat Gluster Storage** - setup Gluster storage on the physical hosts using the Web Console
- **Deploy the hosted engine** - deploy the RHV-M virtual machine using the Web Console
- **Configure the Gluster storage domain** - setup the Gluster storage domain using the Administration Portal

Install Red Hat Virtualization Hosts

The RHV-H images are the same as for a standard installation of RHV 4.3. Download the **Hypervisor Image for RHV 4.3** ISO image from the customer portal, and prepare bootable media using the image. Boot each physical system, and then perform a normal installation. Default values are acceptable, but adjust parameters as required for your hardware. Red Hat recommends these configuration choices:

- Use the **Automatically configure partitioning** option.
- Size **`/var/log`** to 15GB or more, as required for Red Hat Gluster Storage logging.
- Configure networks as required for your physical connections.
- Select the **Automatically connect to this network when it is available** option.

RHV provides a script for performing health checks on RHV hosts, including information from the bootloader, host mount points, storage logical volumes, and the hypervisor thin pool. For additional information, add the **`--debug`** option.

```
[root@hosta ~]# nodectl check
Status: OK
Bootloader ... OK
  Layer boot entries ... OK
  Valid boot entries ... OK
Mount points ... OK
  Separate /var ... OK
  Discard is used ... OK
Basic storage ... OK
  Initialized VG ... OK
  Initialized Thin Pool ... OK
  Initialized LVs ... OK
Thin storage ... OK
  Checking available space in thinpool ... OK
  Checking thinpool auto-extend ... OK
vdsmd ... OK
```

Enable the Red Hat Virtualization Host software repository on each physical host machine. Register the system first, then enable the RHEL 7 Red Hat Virtualization Host repository for packages and updates.

```
[root@hosta ~]# subscription-manager repos --enable=rhel-7-server-rhvh-4-rpms
```

Configure SSH Access

The first host will be the hypervisor for the self-hosted engine running the RHV Manager. The system SSH keys will be used to access all hypervisor hosts, including itself, with passwordless SSH. Repeat this procedure for each network interface, both management and storage, using both IP address and resolved FQDNs to prepopulate the **known_hosts** public key store file on the engine host. Give the password and accept the public key from the first host:

```
[root@hosta ~]# ssh root@hosta.example.com
[root@hosta ~]# exit
[root@hosta ~]# ssh root@hostb.example.com
...output omitted...
[root@hostb ~]# exit
[root@hosta ~]# ssh root@hostc.example.com
...output omitted...
[root@hostc ~]# exit
```

Generate a public/private key pair for the first host root account, so that internal RHV management access will use passwordless, key-based, SSH authentication. Do not use a pass phrase on the key pair, so that it is simple for Ansible to automate deployment and configuration processes using this key.

```
[root@hosta ~]# ssh-keygen -t rsa
...output omitted...
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
The private key is saved in /root/.ssh/id_rsa. The public key is saved in /
root/.ssh/id_rsa.pub.
...output omitted...
```

Copy the newly created SSH key pair to each host, using both IP addresses and FQDNs. Enter the `root` password for each host, if prompted.

```
[root@hosta ~]# ssh-copy-id -i /root/.ssh/id_rsa.pub root@hosta.example.com
...output omitted...
[root@hosta ~]# ssh-copy-id -i /root/.ssh/id_rsa.pub root@hostb.example.com
...output omitted...
[root@hosta ~]# ssh-copy-id -i /root/.ssh/id_rsa.pub root@hostc.example.com
...output omitted...
```

Configure Red Hat Gluster Storage

Before the hosted engine can be installed, the attached storage devices on all hosts must be configured for Red Hat Gluster Storage use. All host-attached disks, other than the just installed RHV-H system disk, must be empty and have no existing partitions or labels. Log in to the **`Web Console`** on the first host, choose the **Virtualization Hosted Engine** option, and then click the **Start** button for the **Hyperconverged** installation.

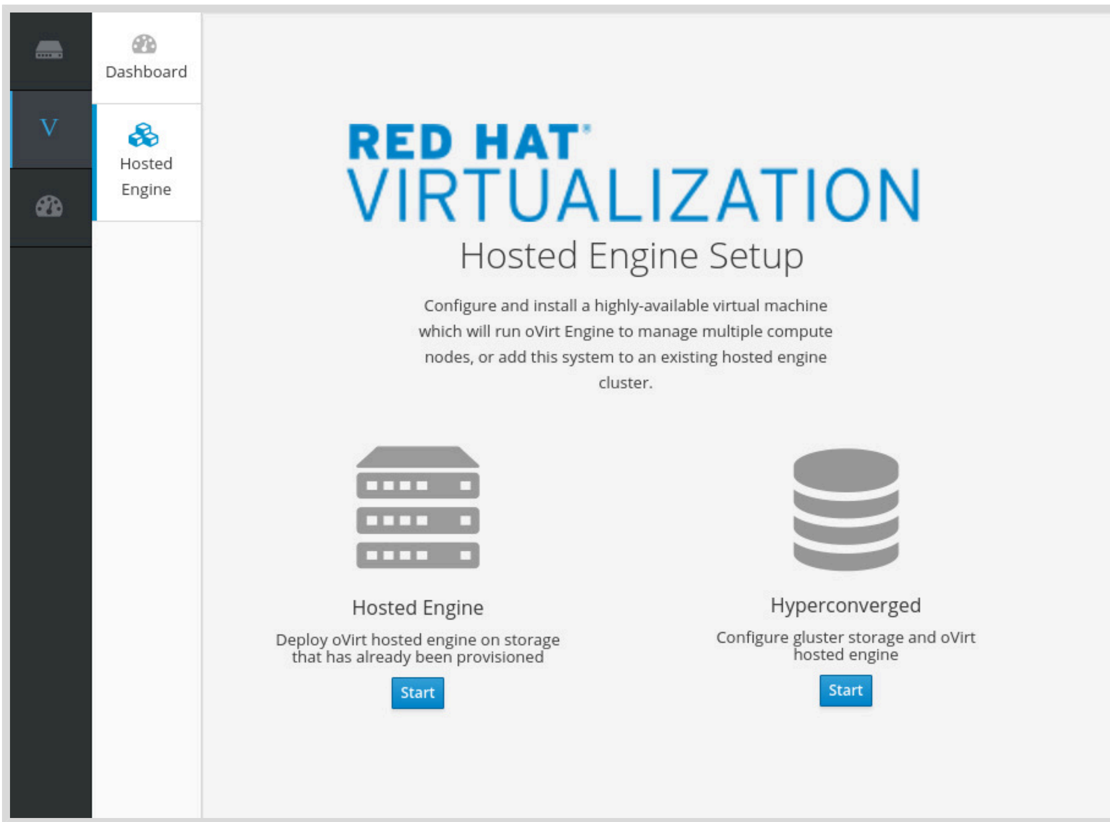

Figure 14.2: The Gluster Configuration window opens

Red Hat Gluster Storage is deployed using a wizard. RHHI-V pods are always installed in multiples of three hypervisor hosts. The initial installation is always accomplished by installing the first three hosts, and then adding hyperconverged hosts later in additional groups of three hosts.

Specify the storage network FQDNs for each of the three hyperconverged hosts. The first hyperconverged host, from which the SSH key pair was generated and distributed, must always be entered as **Host1**, since that host is the engine host and will run deployment tasks. If you are using arbitrated replicated volumes instead of standard replication, as recommended, list the host that will have the arbiter brick as **Host3**.

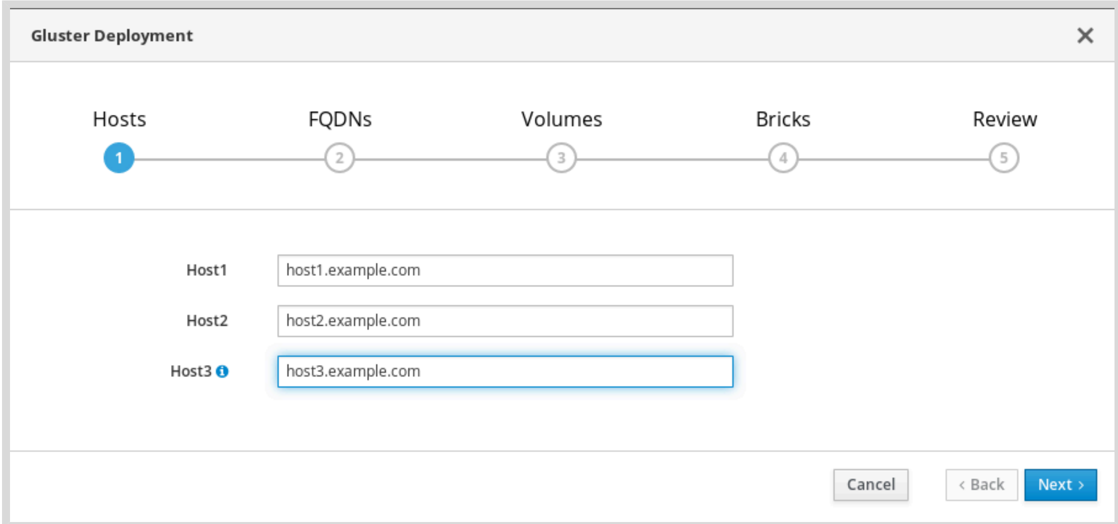

Figure 14.3: Specify hyperconverged hosts

The storage FQDNs were added in the previous step. On the next page, add the management network IP address or FQDN for each of the additional hosts in the RHHI-V pod. These addresses are required for these hosts to be automatically added to RHV-M as part of this deployment.

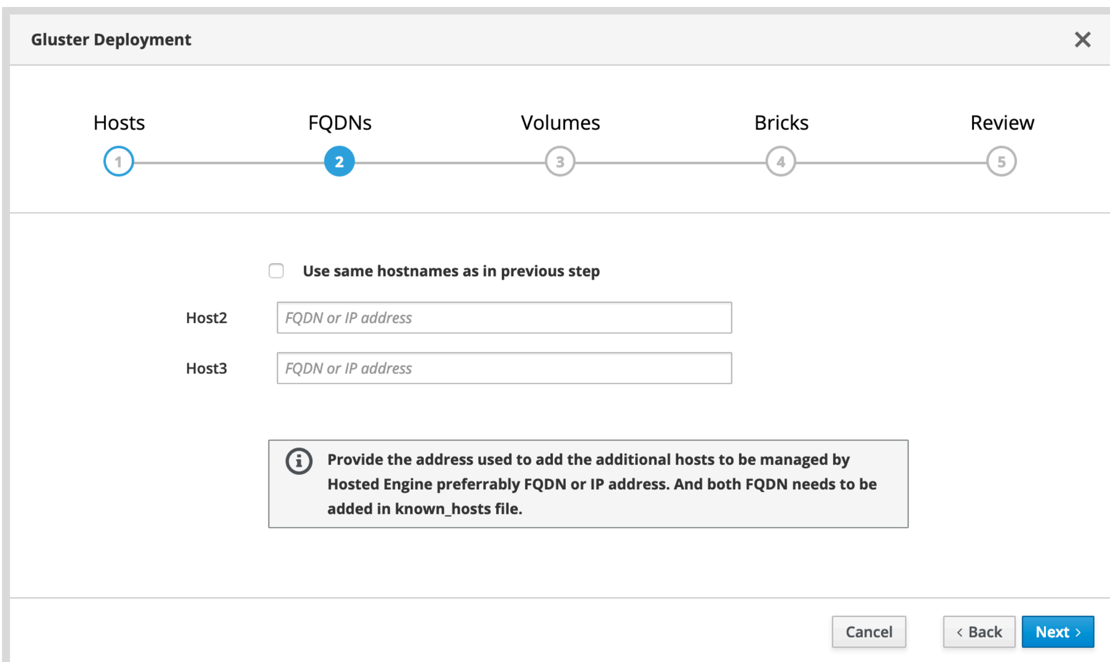

Figure 14.4: Specify management network FQDNs

The non-system storage disks will now be assigned to Gluster. Specify the names of the volumes to create. There must be at least 3 volumes. As needed, you can create multiple vmstore and data volumes, but always keep virtual machine system disk images separate from all other data disks, to facilitate more efficient back up and disaster recovery procedures.

- **engine** - for use by RHV-M to track RHV objects and activities.
- **vmstore** - holds the systems disks for all deployed virtual machines.
- **data** - stores all non-system data disks for deployed virtual machines.

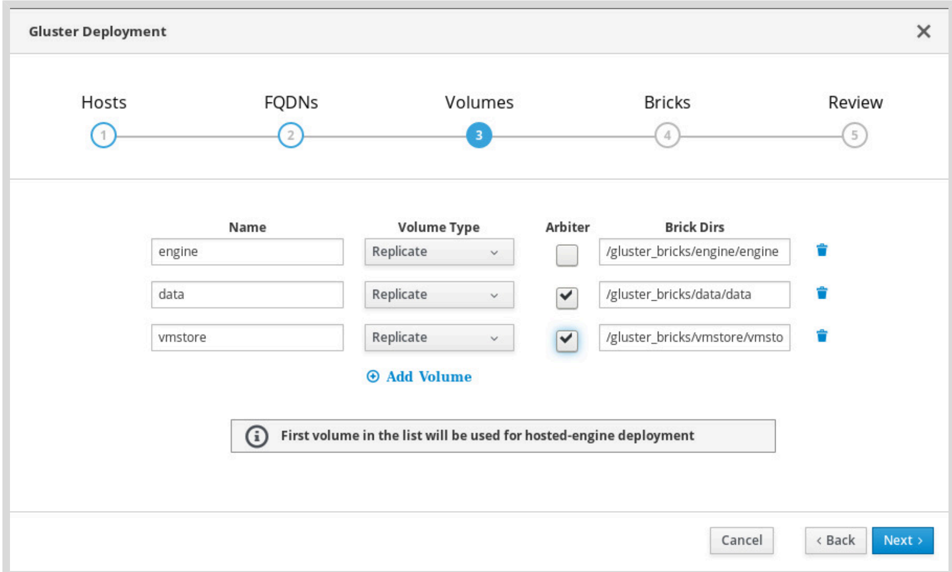

Figure 14.5: Specify the volumes needed

Each storage disk becomes a brick. You previously mapped the volume type, brick and arbiter layout, raid levels, and naming, as preparation for entering the required fields on this screen.

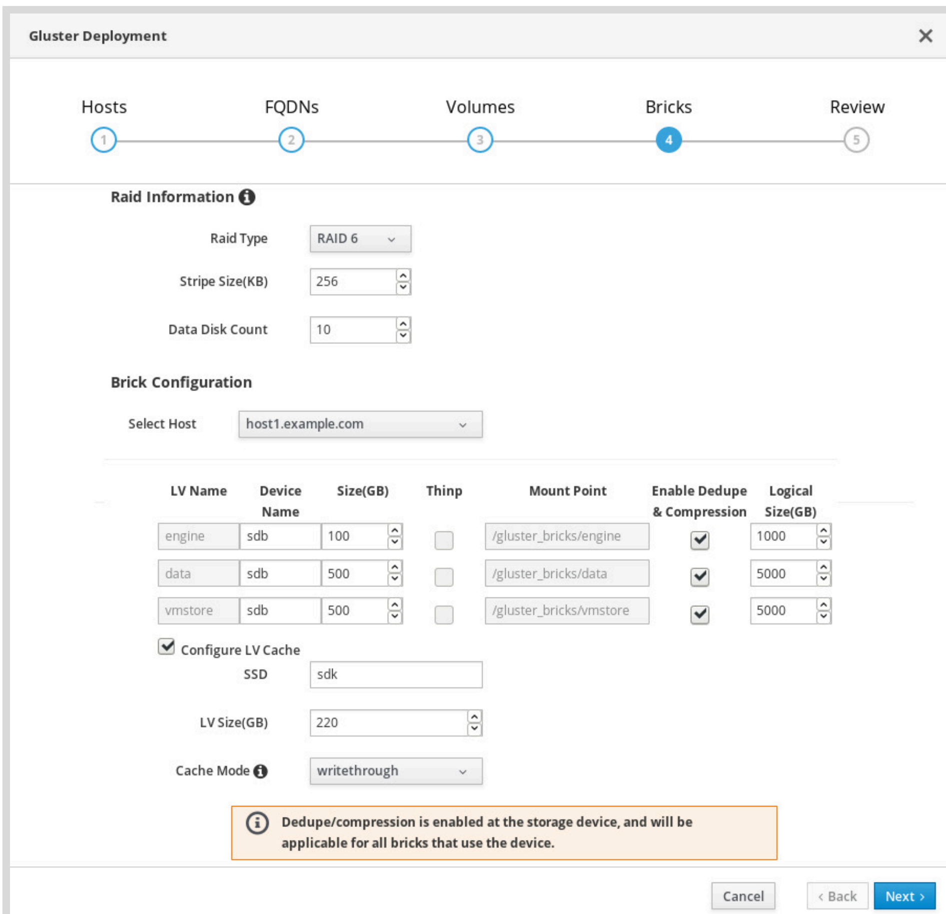

Figure 14.6: Specify the brick layout

After reviewing and approving the **Review**, click the **Deploy** button to begin the Gluster deployment. When the window displays the **`Successfully deployed Gluster`** message, you can install the hosted engine. Click the **Continue to Hosted Engine Deployment** button to proceed with the installation.

Deploy the Hosted Engine

When the RHV Manager is installed as self-hosted, the hosted engine deployment will deploy the RHV-M appliance image as a virtual machine on the first hypervisor host. The embedded RHV-M setup creates a **`Default`** data center, and a **`Default`** cluster with your three physical hosts as members, and then enables Red Hat Gluster Storage functionality on each. All cluster hosts are configured to use the **`virtual-host tuned`** profile.

The RHV-M hosted engine is deployed using a wizard. First, specify the configuration information to use as the deployment creates the RHV-M virtual machine. You must replace the prepopulated MAC address with a valid and unique value. The password entered here will be set for the **`root`** account of this virtual machine.

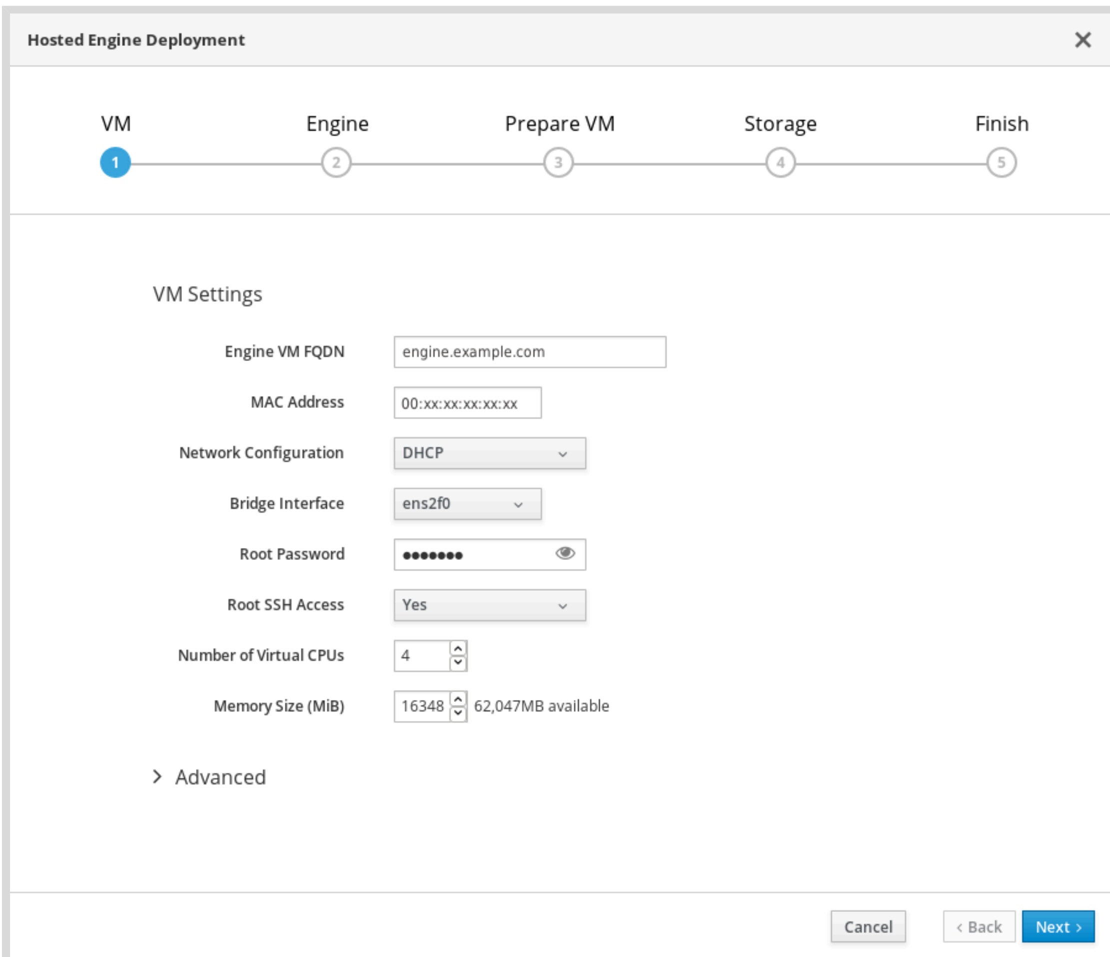

Figure 14.7: Specify the hosted engine virtual machine

Enter the password for the **`admin`** account in the Administration Portal. You can also specify notification behavior here. After reviewing and approving the **Prepare VM** page, click the **Prepare VM** button to deploy the hosted engine virtual machine. When the window displays the **`Execution completed successfully`** message, you can continue.

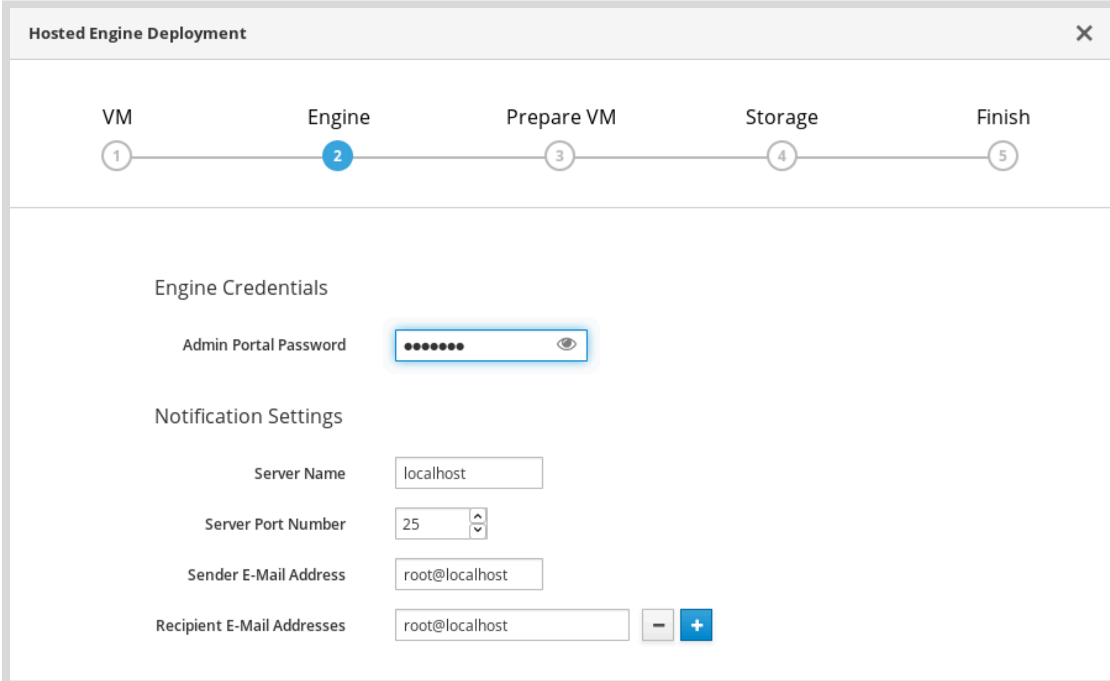

Figure 14.8: Specify the management parameters

Specify the primary host and the location of the engine volume created during the Gluster deployment. It is critical that the storage location and mount options are properly configured in order for the engine volume to be used efficiently. After reviewing and approving the **Finish** page, click the **Finish Deployment** button to complete the hosted engine configuration. This configuration takes approximately 30 minutes. When the window displays the **`Hosted Engine deployment complete`** message, verify the deployment.

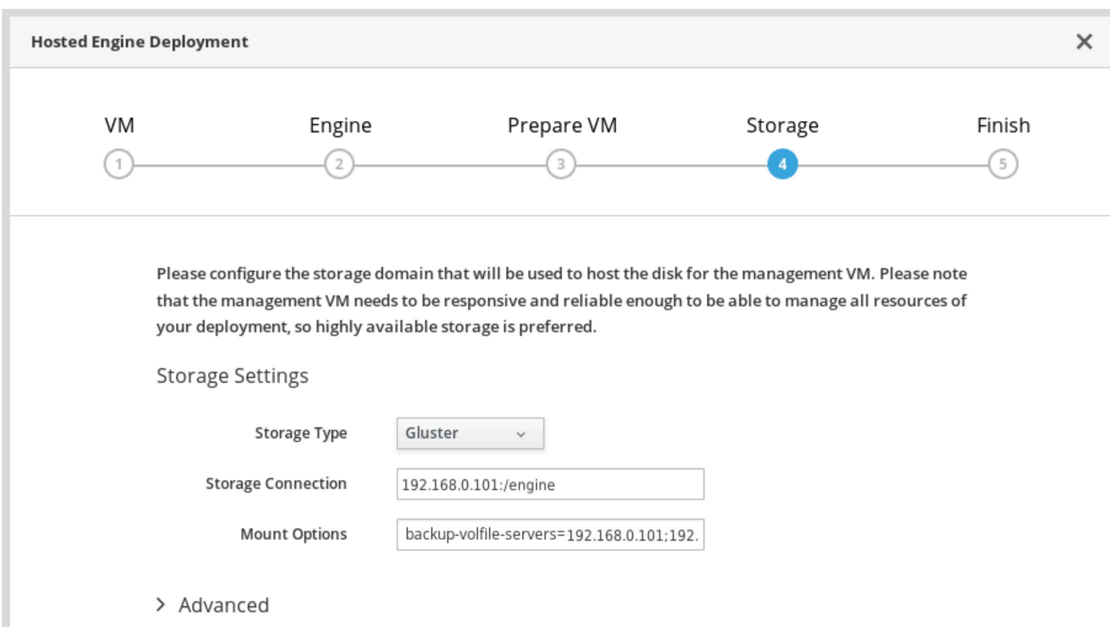

Figure 14.9: Specify the hosted engine storage

Log in to the Administration Portal at `http://engine.example.com/ovirt-engine`, using the **`admin`** account, and then inspect the dashboard for hosts, storage domains, and virtual machines.

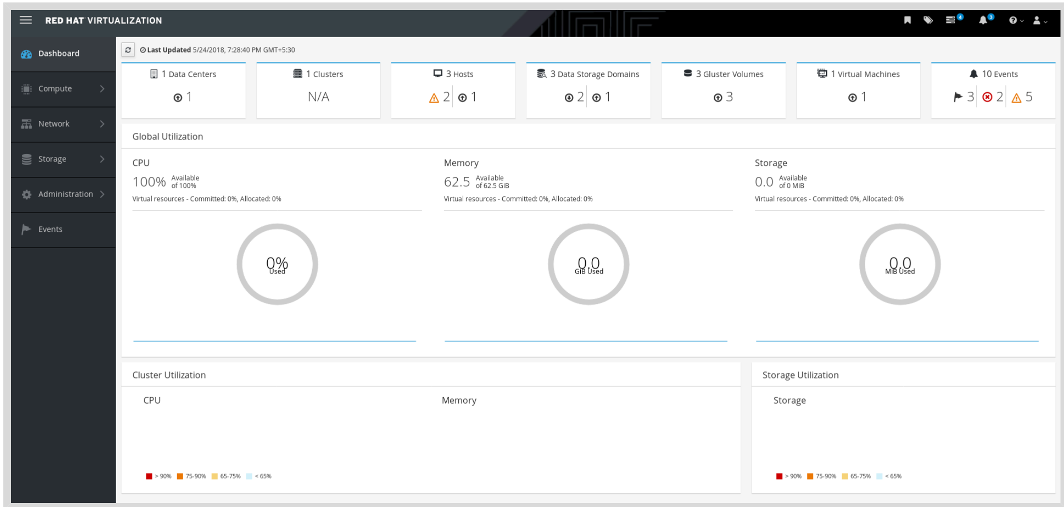

Figure 14.10: Verify the hosted engine deployment in the Dashboard

Configure the Gluster Storage Domain

The final task is to add the previously configured Gluster storage as a RHV storage domain. A storage domain requires a dedicated storage logical network attached to each cluster host.

Create a logical network for Gluster traffic

On the **Networks** page, create a new logical network. Ensure that both the **VM Network** check box, and the **Required** check box, are cleared. In the **Clusters** subtab, on the **Manage Network** window, ensure that the **Migration Network** and **Gluster Network** check boxes are selected.

Attach the Gluster network to each host

On the **Hosts** page, locate the **Network Interfaces** subtab to find and open the **Setup Host Networks** window. Drag and drop the newly created network to the correct interface. Ensure that both the **Verify connectivity**, and the **Save network configuration** check boxes, are selected.

When the network is attached to all hosts, verify the network health. In the **Network Interfaces** tab, inspect the state of the host network. If any network interface displays an **Out of sync** state, or is missing an IP Address, click the **Refresh Capabilities** button to synchronize the configuration.

The standard RHHI-V pod deployment is complete. If additional hosts beyond the first three are needed, they can be added now through the Administration Portal. Each host must have already been installed with the RHV-H operating system, configured for passwordless SSH access, and configured for Gluster storage, the same as the original three hosts. On the **Hosts** page, open the **New Host** window, clear the automatic firewall configuration, and select the hosted engine **Deploy** action. Finally, attach the **`gluster`** network to complete the new host deployment tasks.

Use the Administration Portal to verify the full RHHI-V pod installation. Inspect the hosts, storage domains, clusters, logical networks, and virtual machines.

References

Further information is available in the *Deploying Red Hat Hyperconverged Infrastructure for Virtualization* guide for Red Hat Hyperconverged Infrastructure for Virtualization 1.6 at
https://access.redhat.com/documentation/en-us/red_hat_hyperconverged_infrastructure_for_virtualization/1.6/html-single/deploying_red_hat_hyperconverged_infrastructure_for_virtualization/index

Quiz

Deploying Red Hat Hyperconverged Infrastructure for Virtualization

The steps to deploy Red Hat Hyperconverged Infrastructure for Virtualization follow. Indicate the order in which the steps should be taken.

- [ ] 1. Setup the Gluster storage domain using the Administration Portal.
- [ ] 2. Configure passwordless key-based SSH authentication between hosts.
- [ ] 3. Deploy the RHV-M virtual machine using the Web Console.
- [ ] 4. Setup Gluster storage on the physical hosts using the Web Console.
- [ ] 5. Install the physical machines as hyperconverged hosts.

Solution

Deploying Red Hat Hyperconverged Infrastructure for Virtualization

The steps to deploy Red Hat Hyperconverged Infrastructure for Virtualization follow. Indicate the order in which the steps should be taken.

5	1.	Setup the Gluster storage domain using the Administration Portal.
2	2.	Configure passwordless key-based SSH authentication between hosts.
4	3.	Deploy the RHV-M virtual machine using the Web Console.
3	4.	Setup Gluster storage on the physical hosts using the Web Console.
1	5.	Install the physical machines as hyperconverged hosts.

Maintaining Red Hat Hyperconverged Infrastructure for Virtualization

Objectives

After completing this section, you should be able to perform maintenance tasks to manage RHHI-V pods and storage.

Maintaining a RHHI-V Pod

Operational maintenance of RHHI-V pods includes many of the same tasks required for maintaining RHV clusters, including:

- Configuring high availability using fencing policies.
- Configuring backup and recovery options, including geo-replication, failover, and failback.
- Configuring encryption with Transport Layer Security (TLS/SSL) and certificates.
- Performing upgrades for RHV-H hosts and the RHV-M management engine.
- Monitoring the cluster and managing notification events.
- Adding and removing hypervisor hosts.

Some operational tasks are unique to RHHI-V because of the inclusion of Red Hat Gluster Storage on the hyperconverged hosts. Storage administrators who are already familiar with Gluster will recognize the storage management procedures. This section discusses a few unique RHHI-V tasks, including pod startup and shutdown, and scaling storage using bricks and volumes.

Performing an Ordered Hyperconverged Pod Shutdown

A hyperconverged environment must be shut down in a particular order. The simplest way to do this is to create a shutdown playbook that can be run from the hosted engine virtual machine.

The **`ovirt.shutdown_env`** role enables the global maintenance mode, and initiates shutdown for all virtual machines and hosts in the cluster. Host shutdown is asynchronous. The playbook terminates before hyperconverged hosts are actually shut down.

```
[root@hosta ~]# yum install ovirt-ansible-shutdown-env -y
```

To create a shutdown playbook for your environment, create a file similar to the following example:

```
[root@hosta ~]# cat shutdown_rhhi-v.yml
---
- name: oVirt shutdown environment
  hosts: localhost
  connection: local
  gather_facts: false

  vars:
    engine_url: https://ovirt-engine.example.com/ovirt-engine/api
    engine_user: admin@internal
    engine_password: redhat
    engine_cafile: /etc/pki/ovirt-engine/ca.pem
```

```
  roles:
    - ovirt.shutdown_env
```

Run the shutdown playbook against the hosted engine virtual machine.

```
[root@hosta ~]# ansible-playbook -i localhost shutdown_rhhi-v.yml
```

Performing an Ordered Hyperconverged Pod Startup

Starting up a hyperconverged cluster is more complex than starting up a traditional compute or storage cluster. Follow these instructions to start your hyperconverged cluster safely.

Power on all hosts in the cluster. Ensure that the required services are available. Verify that the **glusterd** service started correctly on all hosts. If **glusterd** is not started, start it.

```
[root@hosta ~]# systemctl status glusterd
...output ommitted...
[root@hosta ~]# systemctl start glusterd
...output ommitted...
```

Verify that host networks are available and have IP addresses assigned to required interfaces.

```
[root@hosta ~]# ip -br addr show
...output ommitted...
```

Verify that all hosts are in the storage cluster with state **Peer in Cluster (Connected)**.

```
[root@hosta ~]# gluster peer status
Number of Peers: 2

Hostname: 172.25.250.11
Uuid: 6e1cfcc6-6c21-48f3-882f-3c10f8c73ff3
State: Peer in Cluster (Connected)

Hostname: 172.25.250.11
Uuid: a79122c1-326f-4816-a7cb-082552bf1621
State: Peer in Cluster (Connected)
```

Verify that all bricks are online..

```
[root@hosta ~]# gluster volume status engine
Status of volume: engine
Gluster process                                    TCP       RDMA Online  Pid
-------------------------------------------------------------------------------
Brick 172.25.250.10:/gluster_bricks/engine/engine  49153     0      Y      23160
Brick 172.25.250.11:/gluster_bricks/engine/engine  49160     0      Y      12392
Brick 172.25.250.12:/gluster_bricks/engine/engine  49157     0      Y      15200
Self-heal Daemon on localhost                      N/A       N/A    Y      23008
Self-heal Daemon on 172.25.250.11                  N/A       N/A    Y      10905
Self-heal Daemon on 172.25.250.12                  N/A       N/A    Y      13568
```

Start the hosted engine virtual machine. Run the following command on the host that you want to be the hosted engine node. Verify that the hosted engine virtual machine has started correctly.

```
[root@hosta ~]# hosted-engine --vm-start
[root@hosta ~]# hosted-engine --vm-status
```

Take the hosted engine virtual machine out of global maintenance mode. Using the Administration Portal, locate the hosted engine node on the **Hosts** page. Click **Disable Global HA Maintenance** to return to normal operation. You may now start other virtual machines using the Web Console.

Managing and Scaling Storage

Managing and scaling available storage is a necessary skill in hyperconverged infrastructures. The embedded Red Hat Gluster Storage is implemented in volumes that span all hosts in the RHHI-V pod. Volumes are built using physical disks configured as bricks, which are evenly distributed across the hyperconverged hosts. Managing storage requires creating and maintaining bricks, and building or restructuring RAID volumes, to be used as RHHI-V data storage domains.

Managing Bricks

Bricks can be created and maintained using the host Web Console, the Administration Portal, or from the command line. Using the Administration Portal is typically the simplest method. The default result is to create a thinly provisioned logical volume on a storage disk device, to be presented to Gluster for use as a brick. On the **Hosts** page, in the **Storage Devices** window, click the **Create a Brick** button, and then give the brick a name and mount point. You can specify if the device presented is a RAID device, and assign a cache device for this disk, which is only useful for legacy hard drives and should not be used for solid state drives.

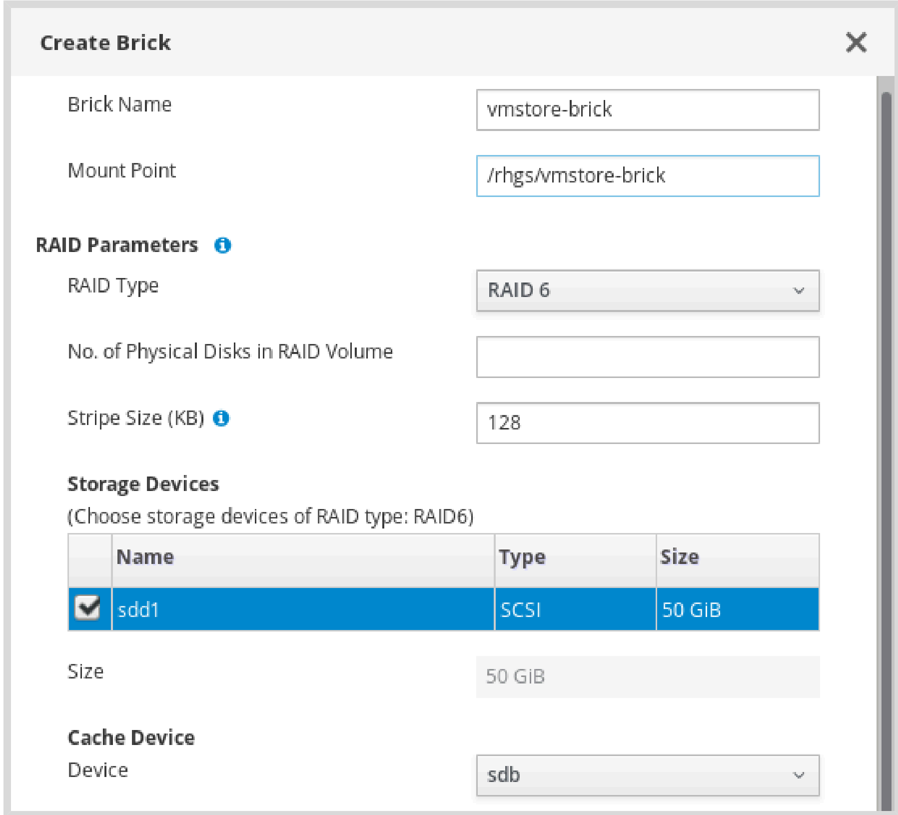

Figure 14.11: Creating a brick

Bricks must be free of partitions and labels before being configured. You can reconfigure a brick by resetting it, allowing the brick to be used as if the brick is being added to the cluster for the first time. Resetting a brick uses the same UUID, host name, and path as before.

Bricks can be replaced when they fail or for other maintenance reasons. Simply select the volume that contains the brick, and then choose the **Replace Brick** button to specify the host and directory of the replacement brick. Bricks can also be removed from a volume, but the volume must be stopped, and the brick unused, before the brick can be removed.

Managing Volumes

A volume is a logical set of bricks, where each brick is an export directory on a hyperconverged host in the pod storage pool. To create a volume in the Administration Portal, navigate to the **Storage** page. In the **Volumes** window, click the **New Volume** button, and then select the volume cluster. Give the volume a name and type, such as as **`Replicated`** or **`Distributed`**, and then select at least 3 bricks to use in the volume. When using arbitrated volumes, select the **Arbiter** check box to configure one or more arbiter bricks for the volume.

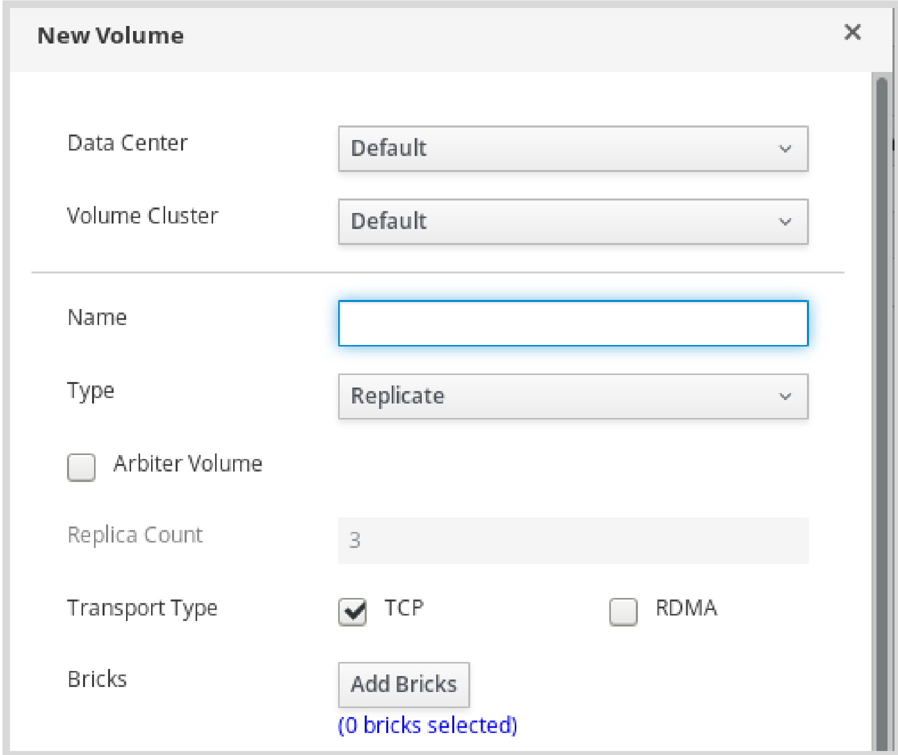

Figure 14.12: Creating a brick

After volumes have been created, they must be started. In the **Volumes** window, select the volume and click the **Start** button. Stopping a volume is done from the same **Volumes** window using the **Stop** button.

Scaling Volumes

Volumes can be expanded by adding bricks. Bricks must be distributed evenly across the hyperconverged hosts in the RHHI-V pod. Prepare bricks on each host, and then add them in multiples of the replica count of the volume. After increasing the bricks in a volume, rebalance the volume to distribute data more equally across all bricks, which will enhance performance of the volume.

Volumes can also be made smaller by removing bricks. Stop the volume before attempting to remove bricks. The volume must be stopped to keep the storage routines from attempting to balance data to a brick while you are trying to remove data from that brick. Bricks must also be removed in multiples of the replica count of the volume. After removing bricks, the volume should be rebalanced for best performance.

Rebalancing Volumes

After expanding or shrinking a volume, rebalance the data among the hosts. In non-replicated volumes, all bricks must be online to perform the rebalance operation. In a replicated volume, at least one of the bricks in the replica must be online. Start the rebalance process from the **Volumes** window. The time needed to rebalance a volume depends on the volume size and the current activity on the volume.

Volumes can be deleted from the **Volumes** page. Stop the volume first, then click the **Remove** button, and then confirm that the volume should be removed. Volume data remains on the disk devices until the volumes are wiped or repurposed.

References

Further information is available in the *Maintaining Red Hat Hyperconverged Infrastructure for Virtualization* guide for Red Hat Hyperconverged Infrastructure for Virtualization 1.6 at
https://access.redhat.com/documentation/en-us/red_hat_hyperconverged_infrastructure_for_virtualization/1.6/html-single/maintaining_red_hat_hyperconverged_infrastructure_for_virtualization/index

Further information is available in the *Managing Red Hat Gluster Storage using RHV Administration Portal* guide for Red Hat Hyperconverged Infrastructure for Virtualization 1.6 at
https://access.redhat.com/documentation/en-us/red_hat_hyperconverged_infrastructure_for_virtualization/1.6/html-single/managing_red_hat_gluster_storage_using_rhv_administration_portal/index

Quiz

Maintaining Red Hat Hyperconverged Infrastructure for Virtualization

Match the items below to their counterparts in the table.

a discrete installation in multiples of 3 hosts

a failover group of hypervisor hosts

a group of storage devices across all hosts in a discrete installation

a logical structure used to create storage domains

a prepared physical device used to build logical storage

a structure of physical devices made in logical devices

the RHV infrastructure manager

the storage management driver for increasing storage efficiency

Term	Meaning
pod	
VDO	
engine	
brick	
volume	
cluster	
RAID	
pool	

Solution

Maintaining Red Hat Hyperconverged Infrastructure for Virtualization

Match the items below to their counterparts in the table.

Term	Meaning
pod	a discrete installation in multiples of 3 hosts
VDO	the storage management driver for increasing storage efficiency
engine	the RHV infrastructure manager
brick	a prepared physical device used to build logical storage
volume	a logical structure used to create storage domains
cluster	a failover group of hypervisor hosts
RAID	a structure of physical devices made in logical devices
pool	a group of storage devices across all hosts in a discrete installation

Summary

In this chapter, you learned:

- RHHI-V is a tuned combination of RHEL, RHV, and Red Hat Gluster Storage, with an OVN SDN stack, VDO, and Red Hat Ansible Automation.
- RHHI-V pods are installed as a self-hosted cluster, with hypervisor hosts in multiples of 3.
- Managing a RHHI-V pod is almost identical to managing standard RHV clusters and standalone Gluster storage.
- Installation and maintenance of RHHI-V pods primarily uses the Web Console and Administration Portal browser utilities.

Chapter 15

Comprehensive Review: Red Hat Virtualization

Goal	Review tasks from *Red Hat Virtualization*
Objectives	• Review tasks from *Red Hat Virtualization*
Sections	• Comprehensive Review
Lab	• Lab: Configuring a Red Hat Virtualization Environment • Lab: Creating Virtual Machines • Lab: Managing Virtual Machines • Lab: Backing Up and Upgrading Red Hat Virtualization

Comprehensive Review

Objectives

After completing this section, students should be able to review and refresh knowledge and skills learned in *Red Hat Virtualization*.

Reviewing Red Hat Virtualization

Before beginning the comprehensive review for this course, students should be comfortable with the topics covered in each chapter.

Students can refer to earlier sections in the textbook for extra study.

Chapter 1, Red Hat Virtualization (RHV) Overview

Explain the purpose and architecture of Red Hat Virtualization.

- Describe the purpose of Red Hat Virtualization and its architectural design.

Chapter 2, Installing and Configuring Red Hat Virtualization

Install a minimal Red Hat Virtualization (RHV) environment and use it to create a virtual machine.

Install Red Hat Virtualization Host (RHV-H) on a computer that will host virtual machines.

- Install Red Hat Virtualization Manager (RHV-M) as a VM on the Red Hat Virtualization Host system.
- Configure storage domains in Red Hat Virtualization that can be used to store virtual machine disks and installation media.
- Use the Administration Portal to manually create a Linux virtual machine running in the Red Hat Virtualization environment.

Chapter 3, Creating and Managing Data Centers and Clusters

Organize hypervisors into groups using data centers and clusters.

- Explain the purpose of a data center for organizing hosts, and create a new data center.
- Explain how clusters are used to group hosts in a data center, and create a new cluster.

Chapter 4, Managing User Accounts and Roles

Configure user accounts using a central directory service. Assign access to resources based on job responsibilities using roles.

- Configure Red Hat Virtualization to authenticate users based on information in a central directory service.
- Control resource access and management in Red Hat Virtualization using Roles.

- Explain the purposes of the User Portal and the Administration Portal, and provide a high-level overview of these user interfaces.

Chapter 5, Scaling RHV Infrastructure

Add and remove Red Hat Virtualization Hosts, both manually and with automated provisioning.

- Remove hosts from an existing cluster for maintenance or reassignment, and add hosts to increase the capability of a cluster in a data center.
- Configure a network installation and kickstart server as provisioning tools in the Red Hat Virtualization environment.

Chapter 6, Managing RHV Networks

Separate network traffic into multiple networks on one or more interfaces to improve the performance and security of Red Hat Virtualization.

- Create logical networks to segregate traffic in a data center.
- Configure hosts to use available logical networks.
- Configure RHV to use networks provided by an external OpenStack provider.

Chapter 7, Managing RHV Storage

Create and manage data storage domains.

- Explain how data storage domains and the Storage Pool Manager work.
- Create and manage data storage domains from NFS, iSCSI, and GlusterFS sources.
- Explain how to configure volume and image storage from an external OpenStack provider.

Chapter 8, Deploying and Managing Virtual Machines

Operate virtual machines in the Red Hat Virtualization environment.

- Install virtual machines that are optimized and configured for the Red Hat Virtualization environment.
- Change configuration or virtual hardware of an existing virtual machine.
- Create a template of a Red Hat Enterprise Linux virtual machine, and use it to deploy a new virtual machine.
- Deploy a virtual machine using a template configured to use cloud-init.

Chapter 9, Migrating Virtual Machines Between Hosts

Migrate and control automatic migration of virtual machines between hosts in a single cluster.

- Migrate virtual machines from one host to another within a cluster in the Red Hat Virtualization environment.
- Configure virtual machines to automatically schedule and migrate VMs using cluster policies.

Chapter 10, Managing Virtual Machine Images

Manage virtual machine snapshots and disk images.

- Create, restore, and delete snapshots of virtual machine images.
- Import and export virtual machine images between data centers in a Red Hat Virtualization environment.

Chapter 11, Introducing the Infrastructure Migration Solution

Describe the Infrastructure Migration Solution for migrating production virtual machines and networks from a VMware vSphere environment to Red Hat Virtualization.

- Describe how virtual machines and network are migrated from a VMware ESX environment to KVM hypervisors and RHV Manager.

Chapter 12, Managing Red Hat Virtualization Infrastructure

Back up, restore, and upgrade the software in a Red Hat Virtualization environment.

- Backup and restore a Red Hat Virtualization Manager (RHV-M) server.
- Perform upgrades and minor updates to Red Hat Virtualization Manager and Red Hat Virtualization Hosts.

Chapter 13, Implementing High Availability

Explain procedures to improve the resiliency and reliability of Red Hat Virtualization by removing single points of failure and implementing high availability features.

- Describe the methodologies used to implement a resilient RHV infrastructure, including redundant networks, storage, and power management.
- Configure hosts and virtual machines to enable high-availability features and failover in the event of a host failure.

Lab

Configuring a Red Hat Virtualization Environment

In this review, you will configure your new Red Hat Virtualization environment.

Outcomes

You should be able to:

- Integrate users in an LDAP directory with your Red Hat Virtualization environment
- Create a new data center
- Create a new cluster
- Register and activate RHV-H hosts
- Create a new storage domain
- Create additional logical networks
- Upload the Red Hat Enterprise Linux installation boot ISO to the data storage domain

Before You Begin

Delete your lab environment and provision a new lab for the comprehensive review. The start script will fail if the environment is not new.

If you wish to go back and do a Lab or GE after running a start script for the review, then you *must* delete the environment again and re-provision a new one.

Log in as the **student** user on **workstation** and run the **lab deploy-cr start** command. This command ensures that the RHV environment is setup correctly.

```
[student@workstation ~]$ lab deploy-cr start
```

Instructions

Configure your Red Hat Virtualization environment according to the following specifications.

- Integrate **utility.lab.example.com**, the Red Hat Enterprise Linux (RHEL) Identity Manager (IdM) server, with your Red Hat Virtualization environment to provide users in a new profile named **lab.example.com**. You should use the StartTLS protocol to connect. The PEM-encoded CA certificate to validate the connection is available at `http://utility.lab.example.com/ipa/config/ca.crt`. Use **uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com** as the search user DN. The password to authenticate using that DN is **redhat**. Continue with default settings otherwise while integrating the RHEL IdM server with your RHV environment.
- Set **rhvadmin** as the system-wide administrative user in your RHV environment.

- Create two new data centers named **`datacenter1`** and **`datacenter2`**. In the **`datacenter1`** data center, create a new cluster named **`cluster1`**. In the **`datacenter2`** data center, create a new cluster named **`cluster2`**.

- Register the three RHV-H hosts as mentioned in the following table with your RHV environment. These RHV-H hosts use **`redhat`** as the password of the **`root`** user. The following table mentions the expected clusters and IP addresses of each host. Ensure that these RHV-H hosts are active in the cluster.

Host	Cluster
`hostb.lab.example.com`	**`cluster1`**
`hostc.lab.example.com`	**`cluster1`**
`hostd.lab.example.com`	**`cluster2`**

- Configure logical networks to help separate network traffic.

  In **`datacenter1`**, create a new logical network named **`virtual`** for virtual machine traffic. It should be tagged as VLAN 10. It should be usable by virtual machines. It should not be used for any RHV infrastructure traffic. It should be associated with the **`eth0`** interface of all hosts in **`cluster1`**. The hosts should use DHCP to get IPv4 settings for that network.

  In **`datacenter1`**, create a new logical network named **`storage`** for storage traffic. It should not use VLAN tagging. It should not be usable by virtual machines. It should not be used for any RHV infrastructure traffic. It should be associated with the **`eth1`** interface of all hosts in **`cluster1`**. The hosts should statically configure IPv4 settings for that network as indicated in the following table.

  In **`datacenter2`**, create a new logical network named **`storage`** for storage traffic. It should not use VLAN tagging. It should not be usable by virtual machines. It should not be used for any RHV infrastructure traffic. It should be associated with the **`eth1`** interface of all hosts in **`cluster2`**. The hosts should statically configure IPv4 settings for that network as indicated in the following table.

  The following two tables summarize the logical network configuration for hosts in **`cluster1`** and **`cluster2`**.

Logical Networks of `cluster1`

Host	Logical network	VLAN tag	Host interface	IPv4 configuration
`hostb`	**`ovirtmgmt`**	*untagged*	**`eth0`**	DHCP (**`172.25.250.11/255.255.255.0`**)
	`virtual`	10	**`eth0`**	DHCP
	`storage`	*untagged*	**`eth1`**	Static (**`172.24.0.11/255.255.255.0`**)
`hostc`	**`ovirtmgmt`**	*untagged*	**`eth0`**	DHCP (**`172.25.250.12/255.255.255.0`**)
	`virtual`	10	**`eth0`**	DHCP

Host	Logical network	VLAN tag	Host interface	IPv4 configuration
	storage	*untagged*	**eth1**	Static (**172.24.0.12/255.255.255.0**)

Logical Networks of cluster2

Host	Logical network	VLAN tag	Host interface	IPv4 configuration
hostd	**ovirtmgmt**	*untagged*	**eth0**	DHCP (**172.25.250.13/255.255.255.0**)
	storage	*untagged*	**eth1**	Static (**172.24.0.13/255.255.255.0**)

- Create a new data domain named **datastorage1** in the **datacenter1** data center using the NFS export **172.24.0.8:/exports/data**. Use the **hostc.lab.example.com** RHV-H hosts in **datacenter1** to mount the NFS export.
- Create a new data domain named **datastorage2** in the **datacenter2** data center using the iSCSI LUN available from the **172.24.0.8** iSCSI portal.
- Upload the boot image, available at **http://materials.example.com/rhel-server-7.6-x86_64-boot.iso**, to the **datastorage1** data domain. This boot image acts as the installation media for the Red Hat Enterprise Linux 7.6 operating system.

Evaluation

On **workstation**, run the **lab deploy-cr grade** command to confirm success of this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab deploy-cr grade
```

Finish

On **workstation**, run the **lab deploy-cr finish** script to complete this lab.

```
[student@workstation ~]$ lab deploy-cr finish
```

This concludes the lab.

Solution

Configuring a Red Hat Virtualization Environment

In this review, you will configure your new Red Hat Virtualization environment.

Outcomes

You should be able to:

- Integrate users in an LDAP directory with your Red Hat Virtualization environment
- Create a new data center
- Create a new cluster
- Register and activate RHV-H hosts
- Create a new storage domain
- Create additional logical networks
- Upload the Red Hat Enterprise Linux installation boot ISO to the data storage domain

Before You Begin

Delete your lab environment and provision a new lab for the comprehensive review. The start script will fail if the environment is not new.

If you wish to go back and do a Lab or GE after running a start script for the review, then you *must* delete the environment again and re-provision a new one.

Log in as the **student** user on **workstation** and run the **lab deploy-cr start** command. This command ensures that the RHV environment is setup correctly.

```
[student@workstation ~]$ lab deploy-cr start
```

Instructions

Configure your Red Hat Virtualization environment according to the following specifications.

- Integrate **utility.lab.example.com**, the Red Hat Enterprise Linux (RHEL) Identity Manager (IdM) server, with your Red Hat Virtualization environment to provide users in a new profile named **lab.example.com**. You should use the StartTLS protocol to connect. The PEM-encoded CA certificate to validate the connection is available at `http://utility.lab.example.com/ipa/config/ca.crt`. Use **uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com** as the search user DN. The password to authenticate using that DN is **redhat**. Continue with default settings otherwise while integrating the RHEL IdM server with your RHV environment.
- Set **rhvadmin** as the system-wide administrative user in your RHV environment.

- Create two new data centers named **datacenter1** and **datacenter2**. In the **datacenter1** data center, create a new cluster named **cluster1**. In the **datacenter2** data center, create a new cluster named **cluster2**.

- Register the three RHV-H hosts as mentioned in the following table with your RHV environment. These RHV-H hosts use **redhat** as the password of the **root** user. The following table mentions the expected clusters and IP addresses of each host. Ensure that these RHV-H hosts are active in the cluster.

Host	Cluster
hostb.lab.example.com	**cluster1**
hostc.lab.example.com	**cluster1**
hostd.lab.example.com	**cluster2**

- Configure logical networks to help separate network traffic.

  In **datacenter1**, create a new logical network named **virtual** for virtual machine traffic. It should be tagged as VLAN 10. It should be usable by virtual machines. It should not be used for any RHV infrastructure traffic. It should be associated with the **eth0** interface of all hosts in **cluster1**. The hosts should use DHCP to get IPv4 settings for that network.

  In **datacenter1**, create a new logical network named **storage** for storage traffic. It should not use VLAN tagging. It should not be usable by virtual machines. It should not be used for any RHV infrastructure traffic. It should be associated with the **eth1** interface of all hosts in **cluster1**. The hosts should statically configure IPv4 settings for that network as indicated in the following table.

  In **datacenter2**, create a new logical network named **storage** for storage traffic. It should not use VLAN tagging. It should not be usable by virtual machines. It should not be used for any RHV infrastructure traffic. It should be associated with the **eth1** interface of all hosts in **cluster2**. The hosts should statically configure IPv4 settings for that network as indicated in the following table.

  The following two tables summarize the logical network configuration for hosts in **cluster1** and **cluster2**.

Logical Networks of cluster1

Host	Logical network	VLAN tag	Host interface	IPv4 configuration
hostb	**ovirtmgmt**	*untagged*	**eth0**	DHCP (**172.25.250.11/255.255.255.0**)
	virtual	10	**eth0**	DHCP
	storage	*untagged*	**eth1**	Static (**172.24.0.11/255.255.255.0**)
hostc	**ovirtmgmt**	*untagged*	**eth0**	DHCP (**172.25.250.12/255.255.255.0**)
	virtual	10	**eth0**	DHCP

Host	Logical network	VLAN tag	Host interface	IPv4 configuration
	storage	*untagged*	**eth1**	Static (**172.24.0.12/255.255.255.0**)

Logical Networks of cluster2

Host	Logical network	VLAN tag	Host interface	IPv4 configuration
hostd	**ovirtmgmt**	*untagged*	**eth0**	DHCP (**172.25.250.13/255.255.255.0**)
	storage	*untagged*	**eth1**	Static (**172.24.0.13/255.255.255.0**)

- Create a new data domain named **datastorage1** in the **datacenter1** data center using the NFS export **172.24.0.8:/exports/data**. Use the **hostc.lab.example.com** RHV-H hosts in **datacenter1** to mount the NFS export.
- Create a new data domain named **datastorage2** in the **datacenter2** data center using the iSCSI LUN available from the **172.24.0.8** iSCSI portal.
- Upload the boot image, available at **http://materials.example.com/rhel-server-7.6-x86_64-boot.iso**, to the **datastorage1** data domain. This boot image acts as the installation media for the Red Hat Enterprise Linux 7.6 operating system.

1. Configure RHV-M to use the **utility.lab.example.com** RHEL IdM server to provide users in a new profile named **lab.example.com**. Enable the StartTLS protocol to establish a secure LDAP connection between RHV-M and the RHEL IdM server. Download the PEM-encoded CA certificate, required to validate the secure connection, from `http://utility.lab.example.com/ipa/config/ca.crt`. Use **uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com** as the search user DN. The password to authenticate using that DN is **redhat**. Continue with default settings otherwise while integrating the RHEL IdM server with your RHV environment.

   1.1. From **workstation**, open an SSH connection to **rhvm** as **root**.

```
[student@workstation ~]$ ssh root@rhvm
...output omitted...
[root@rhvm ~]#
```

   1.2. Use the **rpm** command to verify that the *ovirt-engine-extension-aaa-ldap-setup* package is installed on **rhvm**.

```
[root@rhvm ~]# rpm -q ovirt-engine-extension-aaa-ldap-setup
ovirt-engine-extension-aaa-ldap-setup-1.3.9-1.el7ev.noarch
```

   The *ovirt-engine-extension-aaa-ldap-setup* package is already installed because it is automatically included in a self-hosted engine installation, like the one used in this class.

1.3. To start the interactive setup, run the **ovirt-engine-extension-aaa-ldap-setup** command.

```
[root@rhvm ~]# ovirt-engine-extension-aaa-ldap-setup
[ INFO  ] Stage: Initializing
[ INFO  ] Stage: Environment setup
          Configuration files: ['/etc/ovirt-engine-extension-aaa-ldap-
setup.conf.d/10-packaging.conf']
          Log file: /tmp/ovirt-engine-extension-aaa-ldap-setup-20190702112955-
wwd3ln.log
          Version: otopi-1.8.2 (otopi-1.8.2-1.el7ev)
[ INFO  ] Stage: Environment packages setup
[ INFO  ] Stage: Programs detection
[ INFO  ] Stage: Environment customization
...output omitted...
```

1.4. Type **6** to select **IPA** from the **Available LDAP implementations** list.

```
...output omitted...
           6 - IPA
...output omitted...
          Please select: 6
```

1.5. Press **Enter** to accept the default setting of using DNS to resolve the host name of the Red Hat Enterprise Linux Identity Manager server.

```
...output omitted...
          NOTE:
          It is highly recommended to use DNS resolution for LDAP server.
          If for some reason you intend to use hosts or plain address disable DNS
 usage.

          Use DNS (Yes, No) [Yes]: Enter
```

1.6. Type **1** to select **Single server** from the **Available policy method** list.

```
...output omitted...
          Available policy method:
           1 - Single server
           2 - DNS domain LDAP SRV record
           3 - Round-robin between multiple hosts
           4 - Failover between multiple hosts
          Please select: 1
```

1.7. Type **utility.lab.example.com** to specify the host address of the Red Hat Enterprise Linux Identity Manager server.

```
Please enter host address: utility.lab.example.com
```

1.8. Press **Enter** to accept the default secure connection method (**StartTLS**) for the Red Hat Enterprise Linux Identity Manager server.

```
...output omitted...
          Please select protocol to use (startTLS, ldaps, plain) [startTLS]: Enter
```

1.9. Select the **URL** method to obtain the CA certificate.

```
Please select method to obtain PEM encoded CA certificate (File, URL, Inline,
 System, Insecure): URL
```

1.10. Specify `http://utility.lab.example.com/ipa/config/ca.crt` as the URL to the CA certificate.

```
URL: http://utility.lab.example.com/ipa/config/ca.crt
[ INFO  ] Connecting to LDAP using 'ldap://utility.lab.example.com:389'
[ INFO  ] Executing startTLS
[ INFO  ] Connection succeeded
```

1.11. The Red Hat Enterprise Linux Identity Manager server in the classroom has been configured with a user that the RHV Manager can use to search the LDAP directory for user information. The user DN is **uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com**. The password for this DN is **redhat**.

```
Enter search user DN (for example uid=username,dc=example,dc=com or leave empty
 for anonymous): uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com
Enter search user password: redhat
[ INFO  ] Attempting to bind using
 'uid=rhvadmin,cn=users,cn=accounts,dc=lab,dc=example,dc=com'
```

1.12. Accept **dc=lab,dc=example,dc=com** as the proposed base DN by pressing **Enter**.

```
Please enter base DN (dc=lab,dc=example,dc=com) [dc=lab,dc=example,dc=com]: Enter
```

1.13. Type **No** to indicate that you will not use single sign-on for virtual machines.

```
Are you going to use Single Sign-On for Virtual Machines (Yes, No) [Yes]: No
```

1.14. Specify **lab.example.com** as the name of the profile for the external domain.

```
Please specify profile name that will be visible to users
 [utility.lab.example.com]: lab.example.com
[ INFO  ] Stage: Setup validation
```

1.15. Test the login function to ensure that the Red Hat Enterprise Linux Identity Manager server is connected to the Red Hat Virtualization Manager.

```
NOTE:
It is highly recommended to test drive the configuration before applying it into
 engine.
Login sequence is executed automatically, but it is recommended to also execute
 Search sequence manually after successful Login sequence.
```

```
Please provide credentials to test login flow:
Enter user name: rhvadmin
Enter user password: redhat
[ INFO  ] Executing login sequence...
...output omitted...
[ INFO  ] Login sequence executed successfully
```

1.16. Press **Enter** to use **Done** as the default selection. This completes the configuration.

```
Please make sure that user details are correct and group membership meets
 expectations (search for PrincipalRecord and GroupRecord titles).
Abort if output is incorrect.
Select test sequence to execute (Done, Abort, Login, Search) [Done]: Enter
[ INFO  ] Stage: Transaction setup
[ INFO  ] Stage: Misc configuration (early)
[ INFO  ] Stage: Package installation
[ INFO  ] Stage: Misc configuration
[ INFO  ] Stage: Transaction commit
[ INFO  ] Stage: Closing up
          CONFIGURATION SUMMARY
          Profile name is: lab.example.com
          The following files were created:
              /etc/ovirt-engine/aaa/lab.example.com.jks
              /etc/ovirt-engine/aaa/lab.example.com.properties
              /etc/ovirt-engine/extensions.d/lab.example.com-authz.properties
              /etc/ovirt-engine/extensions.d/lab.example.com-authn.properties
[ INFO  ] Stage: Clean up
          Log file is available at /tmp/ovirt-engine-extension-aaa-ldap-
setup-20190702121518-kusgin.log:
[ INFO  ] Stage: Pre-termination
[ INFO  ] Stage: Termination
```

1.17. Use **systemctl** to restart the **ovirt-engine** service. Wait for the service to finish activating components before accessing the RHV Manager Administration Portal.

```
[root@rhvm ~]# systemctl restart ovirt-engine
```

1.18. Log out from **rhvm**.

```
[root@rhvm ~]# logout
Connection to rhvm closed.
[student@workstation ~]$
```

2. Assign the **SuperUser** role, system-wide, to the **rhvadmin** user in the **lab.example.com** profile.

2.1. On **workstation**, open Firefox and navigate to `https://rhvm.lab.example.com/ovirt-engine`. The RHV-M host's web console service may present a TLS certificate for the HTTPS connection that is signed by an unrecognized Certificate Authority. Either add a security exception for that certificate in your web browser, or configure your web environment to use a trusted Certificate

Authority. Click **Administration Portal** to log in to the web interface as the internal user called **admin** with **redhat** as the password. Select the **internal** profile.

2.2. In the menu, click **Administration**, and then click **Configure**.

2.3. In the **Configure** dialog box, click **System Permissions**.

2.4. Click the **Add** button to add a role to a user.

2.5. In the **Add System Permission to User** dialog box, click the **User** radio button, if not already selected. Click the drop-down list under **Search** to select the **lab.example.com (lab.example.com-authz)** item. This item represents the `lab.example.com` profile you configured in the preceding steps to allow Red Hat Virtualization Manager to use the Red Hat Enterprise Linux Identity Manager as a source for the users.

2.6. Click **GO** to display the users in the Red Hat Enterprise Linux Identity Manager server.

2.7. In the list of users that displays, click the check box for the **rhvadmin** user.

2.8. Click the drop-down list under **Role to Assign**. From the list of available roles, select **SuperUser** role for **rhvadmin**.

2.9. Click **OK** to assign the specified role to the selected user. Notice that the **rhvadmin** user displays in the **System Permissions** list. This list confirms that the **rhvadmin** user has been assigned a role granting administrative access to Red Hat Virtualization.

2.10. In the **Configure** dialog box, click **Close**.

3. Create two new data centers named **datacenter1** and **datacenter2**.

3.1. From the menu, navigate to **Compute** → **Data Centers**.

3.2. From the **Compute >> Data Centers** page, click **New**.

3.3. In the **New Data Center** window, enter **datacenter1** in the **Name** field. Keep the default values for the other fields. Click the **OK** button to create the **datacenter1** data center. The **Data Center - Guide Me** window displays.

3.4. In the **Data Center - Guide Me** window, click **Configure Later**.

3.5. From the **Compute >> Data Centers** page, click **New**.

3.6. In the **New Data Center** window, enter **datacenter2** in the **Name** field. Keep the default values for the other fields. Click the **OK** button to create the **datacenter2** data center. The **Data Center - Guide Me** window displays.

3.7. In the **Data Center - Guide Me** window, click **Configure Later**.

3.8. Confirm that the **Compute >> Data Centers** page lists both **datacenter1** and **datacenter2**.

4. Create the new clusters named **cluster1**, and **cluster2** within the **datacenter1** and **datacenter2** data centers respectfully.

4.1. From the menu, navigate to **Compute** → **Hosts** to access the **Compute >> Hosts** page. From the **Compute >> Hosts** page, click on the name of `hostb.lab.example.com` to access the **Compute >> Hosts >> hostb.lab.example.com** page.

4.2. Use the value of **CPU Type**, under the **Hardware** section of the **General** tab in the **Compute >> Hosts >> hostb.lab.example.com** page, to determine the type of CPU of the **hostb** host. This CPU type is same for **hostc** and **hostd**.

4.3. From the menu, navigate to **Compute** → **Clusters** to access the **Compute >> Clusters** page. In the **Compute >> Clusters** page, click **New**. The **New Cluster** window displays. In the **New Cluster** window, under the **General** section, set the values of the fields according to following table. While selecting the value of **CPU Type** from the available items, ensure that the value matches **CPU Type** of **hostb** you determined previously.

Field	Value
Data Center	**datacenter1**
Name	**cluster1**
Management Network	**ovirtmgmt**
CPU Architecture	**x86_64**
Compatibility Version	**4.3**
Switch Type	**Linux Bridge**

Leave all the other fields with their default values, and click **OK** to create the cluster. The **Cluster - Guide Me** window displays. In the **Cluster - Guide Me** window, click **Configure Later** to continue creating the cluster without configuring the host of the cluster.

4.4. From the **Compute >> Clusters** page, click **New** to create another cluster named **cluster2**. Set the data center for **cluster2** to **datacenter2**. Use the previous table to specify the other properties of **cluster2**.

5. Before you remove the **hostb.lab.example.com**, **hostc.lab.example.com**, and **hostd.lab.example.com** RHV-H hosts from the **Default** cluster, confirm that no virtual machine is running on any of these hosts.

5.1. From the menu, navigate to **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

5.2. From the **Compute >> Virtual Machines** page, confirm that no virtual machine runs on any of the **hostb.lab.example.com**, **hostc.lab.example.com**, **hostd.lab.example.com** RHV-H hosts.

If you see a virtual machine that runs on any of **hostb.lab.example.com**, **hostc.lab.example.com**, **hostd.lab.example.com**, power down that virtual machine.

> **Warning**
> You should see the **HostedEngine** virtual machine that runs on **hosta.lab.example.com** of the **Default** cluster. This virtual machine contains the RHV self-hosted engine. Do not power down this virtual machine.

6. Mark **hostb.lab.example.com**, **hostc.lab.example.com**, and **hostd.lab.example.com** for maintenance.

6.1. From the menu, click on **Compute**, and then click on **Hosts** to access the **Compute >> Hosts** page.

6.2. From the **Compute >> Hosts** page, click on the row for **`hostb.lab.example.com`**.

6.3. From the **Management** drop-down menu, select **Maintenance** to mark **`hostb.lab.example.com`** for maintenance. The **Maintenace Host(s)** window displays. Click **OK** to enable the **Maintenance** status of **`hostb.lab.example.com`**.

6.4. Confirm that the value of the **Status** field for **`hostb.lab.example.com`** is **Maintenance**.

6.5. Use the previous steps to enable the **Maintenance** status for **`hostc.lab.example.com`** and **`hostd.lab.example.com`**.

7. Remove **`hostb.lab.example.com`**, **`hostc.lab.example.com`**, and **`hostd.lab.example.com`** from the **`Default`** cluster.

7.1. From the **Compute >> Hosts** page, click on the row for **`hostb.lab.example.com`**. Click the **Remove** button. The **Remove Host(s)** window displays. In the **Remove Host(s)** window, click **OK** to remove **`hostb.lab.example.com`**.

7.2. Use the previous step to remove **`hostc.lab.example.com`** and **`hostd.lab.example.com`**.

7.3. Confirm that the **`hostb.lab.example.com`**, **`hostc.lab.example.com`**, and **`hostd.lab.example.com`** RHV-H hosts no longer display in the **Compute >> Hosts** page.

8. Add **`hostb.lab.example.com`** and **`hostc.lab.example.com`** to **`cluster1`**.

8.1. From the **Compute >> Hosts** page, click **New**. The **New Host** window displays.

8.2. In the **New Host** window, under the **General** section, set the values of the fields according to the following table.

Field	Value
Host Cluster	**cluster1**
Name	**`hostb.lab.example.com`**
Hostname	**`hostb.lab.example.com`**
SSH Port	**`22`**
Password	**`redhat`**

Ensure that the **Activate host after install** check box is selected. Leave all the other fields with their default values, and click **OK** to register the **`hostb.lab.example.com`** RHV-H host. The **Power Management Configuration** window displays, warning you about the non configuration of Power Management for **`hostb.lab.example.com`**. In the **Power Management Configuration** window, click **OK** to continue adding the host to **`cluster1`** without its Power Management configuration.

8.3. Wait until the value of the **Status** field for **`hostb.lab.example.com`** transitions from **Installing** to **Up**.

8.4. Use the previous steps to add **hostc.lab.example.com** to **cluster1**. While setting the values of different fields for **hostc.lab.example.com** in the **New Host** window, enter **hostc.lab.example.com** as the values of the **Name** and **Hostname** fields. For values of the other fields, refer to the preceding table.

9. Add **hostd.lab.example.com** to **cluster2**.

9.1. From the **Compute >> Hosts** page, click **New**. The **New Host** window displays.

9.2. In the **New Host** window, under the **General** section, set the values of the fields according to the following table.

Field	Value
Host Cluster	**cluster2**
Name	**hostd.lab.example.com**
Hostname	**hostd.lab.example.com**
SSH Port	**22**
Password	**redhat**

Ensure that the **Activate host after install** check box is selected. Leave all the other fields with their default values, and click **OK** to register the **hostd.lab.example.com** RHV-H host. The **Power Management Configuration** window displays, warning you about the non configuration of Power Management for **hostd.lab.example.com**. In the **Power Management Configuration** window, click **OK** to continue adding the host to **cluster2** without its Power Management configuration.

9.3. Wait until the value of the **Status** field for **hostd.lab.example.com** transitions from **Installing** to **Up**.

10. Create a new logical network named **virtual** in the **datacenter1** data center to separate management network traffic from virtual machine network traffic. Specify the **virtual** logical network as a virtual machine network using the VLAN number **10**.

10.1. In the RHV-M **Administration Portal**, navigate to **Network** → **Networks** from the menu to access the **Network >> Networks** page. In the **Network >> Networks** page, click **New** to create a new logical network. The **New Logical Network** window displays. In the **New Logical Network** window, set the values of the different fields according to the following table.

Field	Value
Data Center	**datacenter1**
Name	**virtual**
Description	**Virtual Machine Network**
Comment	**Network for Virtual Machine Traffic**

Select the **Enable VLAN tagging** check box and enter **`10`** as the VLAN number in the text field next to the **Enable VLAN tagging** check box. Ensure that the **VM network** check box is selected, and leave all the other fields with their default values. Click **OK** to create the **`virtual`** logical network.

11. Create the logical network named **`storage`** in both the **`datacenter1`**, and **`datacenter2`** data centers to separate storage traffic from the management network traffic and the virtual machine network traffic. Disable the VLAN tagging for this logical network.

11.1. In the RHV-M **Administration Portal**, navigate to **Network** → **Networks** from the menu to access the **Network >> Networks** page. In the **Network >> Networks** page, click **New** to create a new logical network. The **New Logical Network** window displays. In the **New Logical Network** window, set the values of the different fields according to the following table.

Field	Value
Data Center	**datacenter1**
Name	**`storage`**
Description	**`Storage Network`**
Comment	**`Network for Storage Traffic`**

Ensure that the **Enable VLAN tagging** check box is clear. Clear the **VM network** check box, and leave all the other fields with their default values. Click **OK** to create the **`storage`** logical network.

11.2. Use the previous steps to create a new logical network named **`storage`** in **`datacenter2`**. Disable the VLAN tagging for this logical network. For values of the fields other than **Data center** in the **New Logical Network** window, refer to the preceding table.

12. Assign the **`virtual`**, and **`storage`** logical networks to the **`eth0`**, and **`eth1`** network interfaces respectively of **`hostb.lab.example.com`**. Use DHCP to obtain the IPv4 settings for the network device of the host in the **`virtual`** logical network. Set the **`172.24.0.11`** IP address and the **`255.255.255.0`** netmask statically for the network device in the **`storage`** logical network.

12.1. From the menu, navigate to **Compute** → **Hosts** to access the **Compute >> Hosts** page.

12.2. From the **Compute >> Hosts** page, click on the name of the **`hostb.lab.example.com`** host to access the **Compute >> Hosts >> hostb.lab.example.com** page.

12.3. In the **Compute >> Hosts >> hostb.lab.example.com** page, click the **Network Interfaces** tab.

12.4. Click the **Setup Host Networks** button to change the network configuration of **`hostb.lab.example.com`**.

12.5. In the **Setup Host hostb.lab.example.com Networks** window, click and drag the **virtual (VLAN 10)** box from the right side to the left side of the window. Drop the box next to the **eth0** network interface. After dropping, you should see two logical networks assigned to the **eth0** interface.

12.6. Click and drag the **storage** box from the right side to the left side of the window. Drop the box onto the **no network assigned** field, next to the **eth1** network interface.

12.7. Click on the pencil icon inside the **storage** box. The **Edit Network storage** window displays. In the **Edit Network storage** window, under **Boot Protocol**, select the **Static** radio button.

12.8. In the **IP** field, type **172.24.0.11** as the IP address of **hostb.lab.example.com** in the **storage** network.

12.9. In the **Netmask/Routing Prefix** field, type **255.255.255.0** as the netmask. Leave the **Gateway** field empty.

12.10. Click **OK** to save the settings.

12.11. Ensure that the check boxes for the **Verify connectivity between Host and Engine** and **Save network configuration** options are selected.

12.12. Click **OK** to save the new network configuration for **hostb.lab.example.com**.

> **Warning**
> If you see the value of the **Status** field for the RHV-H hosts as **Non-operational**, click the **Management** drop-down button and select **Activate**.

13. Assign the **virtual**, and **storage** logical networks to the **eth0**, and **eth1** network interfaces respectively of **hostc.lab.example.com**. Use DHCP to obtain the IPv4 settings for the network device of the host in the **virtual** logical network. Set the **172.24.0.12** IP address and the **255.255.255.0** netmask statically for the network device in the **storage** logical network.

13.1. From the menu, navigate to **Compute** → **Hosts** to access the **Compute >> Hosts** page.

13.2. From the **Compute >> Hosts** page, click on the name of the **hostc.lab.example.com** host to access the **Compute >> Hosts >> hostc.lab.example.com** page.

13.3. In the **Compute >> Hosts >> hostc.lab.example.com** page, click the **Network Interfaces** tab.

13.4. Click the **Setup Host Networks** button to change the network configuration of **hostc.lab.example.com**.

13.5. In the **Setup Host hostc.lab.example.com Networks** window, click and drag the **virtual (VLAN 10)** box from the right side to the left side of the window. Drop the box next to the **eth0** network interface. After dropping, you should see two logical networks assigned to the **eth0** interface.

13.6. Click and drag the **storage** box from the right side to the left side of the window. Drop the box onto the **no network assigned** field, next to the **eth1** network interface.

13.7. Click on the pencil icon inside the **storage** box. The **Edit Network storage** window displays. In the **Edit Network storage** window, under **Boot Protocol**, select the **Static** radio button.

13.8. In the **IP** field, type **172.24.0.12** as the IP address of **hostc.lab.example.com** in the **storage** network.

13.9. In the **Netmask/Routing Prefix** field, type **255.255.255.0** as the netmask. Leave the **Gateway** field empty.

13.10. Click **OK** to save the settings.

13.11. Ensure that the check boxes for the **Verify connectivity between Host and Engine** and **Save network configuration** options are selected.

13.12. Click **OK** to save the new network configuration for **hostc.lab.example.com**.

14. Assign the **storage** logical network to the **eth1** network interface of **hostd.lab.example.com**. Set the **172.24.0.13** IP address and the **255.255.255.0** netmask statically for the network device in the **storage** logical network.

14.1. From the menu, navigate to **Compute** → **Hosts** to access the **Compute >> Hosts** page.

14.2. From the **Compute >> Hosts** page, click on the name of the **hostc.lab.example.com** host to access the **Compute >> Hosts >> hostd.lab.example.com** page.

14.3. In the **Compute >> Hosts >> hostd.lab.example.com** page, click the **Network Interfaces** tab.

14.4. Click the **Setup Host Networks** button to change the network configuration of **hostd.lab.example.com**.

14.5. In the Setup Host **hostd.lab.example.com** Networks window, click and drag the **storage** box from the right side to the left side of the window. Drop the box onto the **no network assigned** field, next to the **eth1** network interface.

14.6. Click on the pencil icon inside the **storage** box. The **Edit Network storage** window displays. In the **Edit Network storage** window, under **Boot Protocol**, select the **Static** radio button.

14.7. In the **IP** field, type **172.24.0.13** as the IP address of **hostd.lab.example.com** in the **storage** network.

14.8. In the **Netmask/Routing Prefix** field, type **255.255.255.0** as the netmask. Leave the **Gateway** field empty.

14.9. Click **OK** to save the settings.

14.10. Ensure that the check boxes for the **Verify connectivity between Host and Engine** and **Save network configuration** options are selected.

14.11. Click **OK** to save the new network configuration for **hostd.lab.example.com**.

15. Create an NFS-based storage domain called **datastorage1** to function as the data domain in the **datacenter1** data center. This storage domain should use **172.24.0.8:/exports/data** as the NFS export path in the back end for the **datastorage1** storage domain in the **datacenter1** data center. Use the **hostc.lab.example.com** RHV-H host in **datacenter1** to mount the NFS export. The **172.24.0.8** IP address belongs to the storage network.

15.1. From the menu, navigate to **Storage** → **Domains**.

15.2. Click **New Domain**.

15.3. In the **New Domain** window, set the values of the fields according to the following table.

Field	Value
Data Center	**datacenter1**
Domain Function	**Data**
Storage Type	**NFS**
Host to Use	**hostc.lab.example.com**
Name	**`datastorage1`**
Export Path	**`172.24.0.8:/exports/data`**

15.4. Click **OK** to create the **`datastorage1`** storage domain.

15.5. From the **Storage Domains** page under **Storage**, verify that the **`datastorage1`** storage domain exists, and displays the **Active** status in the **Cross Data Center Status** column. It may take a couple of minutes for the **`datastorage1`** storage domain status to transition from **Locked** to **Active**.

16. Create an iSCSI-based storage domain called **`datastorage2`** to function as the data domain in the **`datacenter2`** data center. Use a LUN from the iSCSI target on the **`172.24.0.8`** address of **`utility`**. The **`172.24.0.8`** IP address belongs to the storage network.

16.1. From the menu, navigate to **Storage** → **Domains**.

16.2. Click **New Domain**.

16.3. In the **New Domain** window, set the values of the fields according to the following table.

Field	Value
Data Center	**datacenter2**
Domain Function	**Data**
Storage Type	**iSCSI**
Host to Use	**hostd.lab.example.com**
Name	**`datastorage2`**

16.4. In the **Discover Targets** section, specify **`172.24.0.8`** in the **Address** field. Set the **Port** field to 3260, if not already set. Click **Discover** to display the available iSCSI target LUNs.

The **`utility`** system uses the **`172.24.0.8`** IP address in the storage network.

16.5. Verify that the **Targets > LUNs** section includes the **`iqn.2019-07.com.example.lab:utility`** target name. Click the right arrow button for the **`iqn.2019-07.com.example.lab:utility`** target name to log in to it.

16.6. Click **+** next to the **`iqn.2019-07.com.example.lab:utility`** target name to expand and display the list of available iSCSI target LUNs. Click **Add** for the displayed iSCSI target LUN. Click **OK** to create the **`datastorage2`** storage domain.

> **Note**
> If you encounter a warning that mentions about the destructive behavior of the operation, select the **Approve operation** check box and click **OK**.

16.7. From the **Storage Domains** page under **Storage**, verify that the **`datastorage2`** storage domain exists with an **Active** status in the **Cross Data Center Status** column. It may take a couple of minutes for the **`datastorage2`** storage domain status to transition from **Locked** to **Active**.

17. Upload the boot image, available at **`http://materials.example.com/rhel-server-7.6-x86_64-boot.iso`**, to the **`datastorage1`** data domain. Use **`rhel-server-7.6-x86_64-boot.iso`** as the name for the image in RHV. This boot image acts as the installation media for the Red Hat Enterprise Linux 7.6 operating system.

17.1. On **`workstation`**, open a terminal and download **`http://materials.example.com/rhel-server-7.6-x86_64-boot.iso`** as **`/home/student/Downloads/rhel-server-7.6-x86_64-boot.iso`**.

```
[student@workstation ~]$ curl -o \
/home/student/Downloads/rhel-server-7.6-x86_64-boot.iso \
http://materials.example.com/rhel-server-7.6-x86_64-boot.iso
```

17.2. From the menu of the RHV-M **Administration Portal**, navigate to **Storage** → **Domains** to access the **Storage >> Storage Domains** page.

17.3. From the **Storage >> Storage Domains** page, click on the name of the **`datastorage1`** data center to access the **Storage >> Storage Domains >> datastorage1** page.

17.4. From the **Storage >> Storage Domains >> datastorage1** page, click on the **disks** tab.

17.5. Click the **Upload** drop-down button and select **Start**. The **Upload Image** window displays.

17.6. In the **Upload Image** window, click **Choose File** to point to **`/home/student/Downloads/rhel-server-7.6-x86_64-boot.iso`**.

17.7. Click the **Test Connection** button to verify this. If clicking the **Test Connection** button returns a green success box, then you are ready to upload. If clicking the **Test Connection** button returns an orange warning box, click the ovirt-engine certificate link within the warning box. Check the box next to Trust this CA to identify websites and then click the **OK** button. After you have done this, click the **Test Connection** button again. It should return a green success box.

> **Important**
>
> If you accidentally forget to check the box next to **Trust this CA to identify websites**, the following procedure will bring up that window again:
>
> 1. Open Preferences for Firefox and then select **Privacy & Security** in the left menu.
> 2. Scroll down to the Security section (at the bottom) and click the **View Certificates...** button.
> 3. In the Certificate Manager window, scroll down to lab.example.com, click rhvm.lab.example.com.34088 so that it is highlighted, and then click the **Delete or Distrust** button.
> 4. Back on the Preferences tab for **Privacy & Security**, scroll up to the Cookies and Site Data section and then click the **Clear Data...** button.
> 5. Accept the default selections and click the **Clear** button. Confirm your choice by clicking the **Clear Now** button in the new window that appears.

17.8. Click **OK** to upload the image.

17.9. Wait until the value of the **Status** field for the image transitions from **Locked** to **OK**. It takes a couple of minutes for the **Status** field to transition to **OK**.

Evaluation

On **`workstation`**, run the **`lab deploy-cr grade`** command to confirm success of this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab deploy-cr grade
```

Finish

On **`workstation`**, run the **`lab deploy-cr finish`** script to complete this lab.

```
[student@workstation ~]$ lab deploy-cr finish
```

This concludes the lab.

Lab

Creating Virtual Machines

In this review, you install a new virtual machine. Once installed, you create and use templates to deploy additional virtual machines.

Outcomes

You should be able to:

- Install a virtual machine manually.
- Create a template.
- Install a virtual machine from a template.

Before You Begin

Log in as the **student** user on **workstation** and run the **lab vms-cr start** command. This command ensures that the RHV environment is setup correctly by configuring the RHV-H hosts, networks, storage, and disk images appropriately.

```
[student@workstation ~]$ lab vms-cr start
```

Instructions

Create two virtual machines and a template according to the following specifications.

- Make sure the *virt-viewer* package is installed on **workstation**. The password for **student** is **student**. Use this password to gain the administrative privileges that have been pre-configured.
- Create a new Red Hat Enterprise Linux virtual machine named **rh1** in **cluster1** using Kickstart. Use the Red Hat Enterprise Linux 7.6 boot ISO image. A Kickstart answer file has been provided at `http://materials.example.com/static/small-7.6.cfg`. Remember that you can use the RHV-M **Administration Portal** to access the console of the virtual machine. The URL for the Administration Portal is `https://rhvm.lab.example.com`.

  The **rh1** virtual machine should use a new disk image of 3 GiB in size. The maximum memory for **rh1** should be 512 MB with the **Instance Type** as **Tiny**. The **rh1** virtual machine should be optimized to run as a server. The first NIC of **rh1** should be associated with the **ovirtmgmt** logical network. The second NIC of **rh1** should be associated with the **virtual** logical network. The Kickstart installation of **rh1** sets the password of the local **root** user to **redhat**. Use the **rhel-server-7.6-x86_64-boot.iso** as the bootable media while invoking the Kickstart installation of **rh1**.
- Ensure that the *qemu-guest-agent* package is installed on **rh1** and that the **qemu-guest-agent** service is enabled. You have been provided with a package repository that contains the guest agent software. The file available at `http://materials.example.com/yum.repos.d/rhel_dvd.repo` gives you repository configuration to enable this package repository in **rh1**.

- Create a template named **rh-template** based on the **rh1** virtual machine. Make sure to seal the template and use **rhel-cr** as the disk alias for the template.
- Using the **rh-template** template, create a new virtual machine named **rh2** in **cluster1** cluster.

Evaluation

On **workstation**, run the **lab vms-cr grade** command to confirm success of this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab vms-cr grade
```

Finish

On **workstation**, run the **lab vms-cr finish** script to complete this lab.

```
[student@workstation ~]$ lab vms-cr finish
```

This concludes the lab.

Solution

Creating Virtual Machines

In this review, you install a new virtual machine. Once installed, you create and use templates to deploy additional virtual machines.

Outcomes

You should be able to:

- Install a virtual machine manually.
- Create a template.
- Install a virtual machine from a template.

Before You Begin

Log in as the **student** user on **workstation** and run the **lab vms-cr start** command. This command ensures that the RHV environment is setup correctly by configuring the RHV-H hosts, networks, storage, and disk images appropriately.

```
[student@workstation ~]$ lab vms-cr start
```

Instructions

Create two virtual machines and a template according to the following specifications.

- Make sure the *virt-viewer* package is installed on **workstation**. The password for **student** is **student**. Use this password to gain the administrative privileges that have been pre-configured.

- Create a new Red Hat Enterprise Linux virtual machine named **rh1** in **cluster1** using Kickstart. Use the Red Hat Enterprise Linux 7.6 boot ISO image. A Kickstart answer file has been provided at `http://materials.example.com/static/small-7.6.cfg`. Remember that you can use the RHV-M **Administration Portal** to access the console of the virtual machine. The URL for the Administration Portal is `https://rhvm.lab.example.com`.

  The **rh1** virtual machine should use a new disk image of 3 GiB in size. The maximum memory for **rh1** should be 512 MB with the **Instance Type** as **Tiny**. The **rh1** virtual machine should be optimized to run as a server. The first NIC of **rh1** should be associated with the **ovirtmgmt** logical network. The second NIC of **rh1** should be associated with the **virtual** logical network. The Kickstart installation of **rh1** sets the password of the local **root** user to **redhat**. Use the **rhel-server-7.6-x86_64-boot.iso** as the bootable media while invoking the Kickstart installation of **rh1**.

- Ensure that the *qemu-guest-agent* package is installed on **rh1** and that the **qemu-guest-agent** service is enabled. You have been provided with a package repository that contains the guest agent software. The file available at `http://materials.example.com/yum.repos.d/rhel_dvd.repo` gives you repository configuration to enable this package repository in **rh1**.

- Create a template named **rh-template** based on the **rh1** virtual machine. Make sure to seal the template and use **rhel-cr** as the disk alias for the template.
- Using the **rh-template** template, create a new virtual machine named **rh2** in **cluster1** cluster.

1. On **workstation**, install *virt-viewer*.

```
[student@workstation ~]$ sudo yum install virt-viewer
[sudo] password for student: student
...output omitted...
Is this ok [y/d/N]: y
...output omitted...
Complete!
```

2. Create a new Red Hat Enterprise Linux virtual machine named **rh1** in **cluster1**. Use the Red Hat Enterprise Linux 7.6 boot ISO image, created at the start of this exercise. A Kickstart answer file has been provided at `http://materials.example.com/static/small-7.6.cfg`. Remember that you can use the Administration Portal to access the console of the virtual machine.

   Create a 3 GiB disk image for this virtual machine. Specify the **Instance Type** as **Tiny**, and set 512 MB as the maximum memory of the virtual machine. Select **Server** as the value of the **Optimized for** field of the **rh1** virtual machine. Associate the virtual machine's first NIC with the **ovirtmgmt** logical network. Associate the virtual machine's second NIC with the **virtual** logical network.

   2.1. Log in to the RHV-M **Administration Portal** as **admin** from the **internal** profile, using **redhat** as the password. The URL for the Administration Portal is `https://rhvm.lab.example.com`.

   2.2. From the menu, navigate to **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

   2.3. In the **Compute >> Virtual Machines** page, click **New** to create a new virtual machine. The window titled **New Virtual Machine** displays. In the **New Virtual Machine** window, under the **General** section, set the values of the fields according to the following table.

Field	Value
Cluster	**cluster1**
Template	**Blank | (0)**
Operating System	**Red Hat Enterprise Linux 7.x x64**
Instance Type	**Tiny**
Optimized for	**Server**
Name	**rh1**

2.4. In the **New Virtual Machine** window, click **Create** near to **Instance Images**. The window titled **New Virtual Disk** displays. In the **New Virtual Disk** window, specify **3** as the value of the **Size (GiB)** field. Leave all the other fields with their default values, and click **OK** to confirm creating the virtual disk of 3 GiB in size for the `rh1` virtual machine.

2.5. In the **New Virtual Machine** window, under the **General** section, click the drop-down menu for **nic1**. From the menu that appears, select **ovirtmgmt (ovirtmgmt)** to associate the first NIC of `rh1` with the `ovirtmgmt` logical network.

2.6. In the **New Virtual Machine** window, under the **General** section, click the **+** button that is displayed in the same row as **nic1**. A second row displays with the **nic2** drop-down menu. Click on the drop-down menu for **nic2**, and select **virtual (virtual)** to associate the second NIC of `rh1` with the `virtual` logical network.

2.7. In the **New Virtual Machine** window, click on **Show Advanced Options** and navigate to the **System** section. Change the value of the **Maximum memory** field to `512 MB`. Click **OK** to confirm your intention to create the virtual machine with the defined properties.

> **Note**
> Selecting an instance type populates memory and CPU values for the virtual machine. These values are visible from the System tab when viewing advanced options. Although choosing either Server or High Performance from the Optimized for menu will switch the Instance Type menu back to Custom, the memory and CPU values for the instance type that you selected will remain the same.

3. Modify the parameters of `rh1` to use the `rhel-server-7.6-x86_64-boot.iso` as the bootable media at its next boot, and invoke the Kickstart installation of `rh1`. Use the answer file available at `http://materials.example.com/static/small-7.6.cfg` for the Kickstart installation.

3.1. Click on the row for the `rh1` virtual machine. Click on the drop-down menu next to the **Run** button. From the menu that displays, select **Run Once**. The window titled **Run Virtual Machine(s)** displays.

3.2. In the **Run Virtual Machine(s)** window, click the **+** sign next to **Boot Options**. Select the check box next to **Attach CD** to use the displayed `rhel-server-7.6-x86_64-boot.iso` file as the bootable media for `rh1`. Selecting this check box inserts the ISO file into the virtual CD-ROM/DVD-ROM drive of `rh1`.

3.3. Under **Predefined Boot Sequence**, select **CD-ROM** and click **Up** to bring **CD-ROM** to the top of the list. Click **OK** to confirm the changes you made, and to boot the `rh1` virtual machine. Wait until the **Console** button becomes active.

3.4. With the `rh1` virtual machine selected, click **Console** to access the console of `rh1`. The **Opening console.vv** window opens and prompts for your confirmation to open the connection file (`console.vv`). Click **OK** to open the `console.vv` file using Remote Viewer. Wait until the installer menu displays in the console of `rh1`.

> **Note**
> If Remote Viewer prompts for your confirmation to inhibit shortcuts, click **Allow**.

3.5. From the installer menu that is displayed in the console of `rh1`, highlight the **Install Red Hat Enterprise Linux 7.6** and press **Tab** to edit the installer options.

The editor opens with the cursor automatically positioned at the end of the existing kernel arguments. At the end of the kernel command line, add a space and the argument **`inst.ks=http://materials.example.com/static/small-7.6.cfg`** to specify the location of the answer file to use for the Kickstart installation. Press **`Enter`** to start the Kickstart installation of the **`rh1`** virtual machine.

3.6. Watch for the installation to complete. After the installation completes, the virtual machine automatically reboots from the CD-ROM drive. Press an arrow key to interrupt the timer for preventing the installer to invoke the installation process.

3.7. Close the console of the **`rh1`** virtual machine.

3.8. Right-click on the row for the **`rh1`** virtual machine, and select **Power Off**. The window titled **Power Off Virtual Machine(s)** displays. In the **Power Off Virtual Machine(s)** window, click **OK** to confirm your intention of powering down the virtual machine.

4. Verify that the *qemu-guest-agent* package is installed on the **`rh1`** virtual machine. Start the **`qemu-guest-agent`** service in **`rh1`**. Use the file located at `http://materials.example.com/yum.repos.d/rhel_dvd.repo` to enable the package repository that provides the *qemu-guest-agent* package.

4.1. In the **Compute >> Virtual Machines** page, select the **`rh1`** virtual machine, and click **Run**. Wait for the value of the **Status** field to transition from **Wait For Launch** to **Up** with **Powering Up** in between.

4.2. After the virtual machine is started, click **Console** to access the console of **`rh1`**. The **Opening console.vv** window opens and prompts for your confirmation to open the connection file (**`console.vv`**). Click **OK** to open the **`console.vv`** file using Remote Viewer.

> **Note**
> If Remote Viewer prompts for your confirmation to inhibit shortcuts, click **Allow**.

4.3. Log in to **`rh1`** as **`root`** with the password **`redhat`**.

4.4. Download the repository configuration file from `http://materials.example.com/yum.repos.d/rhel_dvd.repo` as **`/etc/yum.repos.d/rhel_dvd.repo`**.

```
[root@rh1 ~]# curl http://materials.example.com/yum.repos.d/rhel_dvd.repo \
-o /etc/yum.repos.d/rhel_dvd.repo
```

> **Note**
> The shell prompts that you see may vary from the book.

4.5. Use **`yum`** to verify the package repository on **`rh1`**.

```
[root@rh1 ~]# yum repolist
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use
 subscription-manager to register.
repo id     repo name                          status
rhel_dvd    Remote classroom copy of dvd       5,152
repolist: 5,152
```

4.6. Use **yum** to verify that the *qemu-guest-agent* package is installed on **rh1**.

```
[root@rh1 ~]# yum list installed qemu-guest-agent
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use
 subscription-manager to register.
Installed Packages
qemu-guest-agent.x86_64       10:2.12.0-2.el7         @rhel_dvd
```

4.7. Use **systemctl** to restart the *qemu-guest-agent* service in **rh1**.

```
[root@rh1 ~]# systemctl restart qemu-guest-agent.service
```

4.8. Use **systemctl** to verify that the *qemu-guest-agent* service is active and enabled in **rh1**.

```
[root@rh1 ~]# systemctl status qemu-guest-agent.service
● qemu-guest-agent.service - QEMU Guest Agent
   Loaded: loaded (/usr/lib/systemd/system/qemu-guest-agent.service; enabled;
 vendor preset: enabled)
   Active: active (running) since Tue 2019-09-10 12:04:22 UTC; 4s ago
...output omitted...
```

4.9. Log out from the **rh1** virtual machine.

4.10. Close the console of the **rh1** virtual machine.

5. Create a template named **rh-template** based on the **rh1** virtual machine.

5.1. In the RHV-M **Administration Portal**, navigate to the **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

5.2. In the **Compute >> Virtual Machines** page, right-click on the row for the **rh1** virtual machine and select **Power Off**. The window titled Power Off Virtual Machine(s) displays. In the Power Off Virtual Machine(s) window, click **OK** to power down the virtual machine.

5.3. With **rh1** selected, click on the three vertical dots next to the **Create Snapshot** button. From the menu that displays, select **Make Template**. The window titled **New Template** displays.

5.4. In the **New Template** window, enter **rh-template** as the value of the **Name** field. Select the **Seal Template (Linux only)** check box. Change the value of the **Alias** column under **Disk Allocation** to **rhel-cr**. Leave all the other fields with their default values, and click **OK** to confirm your intention of creating the template.

5.5. Navigate to **Events** and search for the event containing the **`Creation of Template rh-template from VM rh1 has been completed.`** message. It may take a couple of minutes for this message to display. This message confirms that the **`rh-template`** template was successfully created.

6. In the **`cluster1`** cluster, create a new virtual machine named **`rh2`** using the **`rh-template`** template you created previously.

6.1. From the menu, navigate to **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

6.2. In the **Compute >> Virtual Machines** page, click **New** to create a new virtual machine. The window titled **New Virtual Machine** displays. In the **New Virtual Machine** window, under the **General** section, select **cluster1** for the **Cluster** field. Select **rh-template** for the **Template** field. Enter **`rh2`** as the value of the **Name** field. Leave all the other fields with their default values, and click **OK** to confirm creating the new virtual machine based on **`rh-template`**.

6.3. Navigate to **Events** and search for the event containing the **`VM rh2 creation has been completed.`** message. It may take a couple of minutes for this message to display. This message confirms that the **`rh2`** virtual machine was successfully created.

Evaluation

On **`workstation`**, run the **`lab vms-cr grade`** command to confirm success of this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab vms-cr grade
```

Finish

On **`workstation`**, run the **`lab vms-cr finish`** script to complete this lab.

```
[student@workstation ~]$ lab vms-cr finish
```

This concludes the lab.

Lab

Managing Virtual Machines

In this review, you will migrate a virtual machine from one host to another and move a virtual machine from one cluster to another.

Outcomes

You should be able to:

- Migrate virtual machines between hosts.
- Create an NFS export storage domain.
- Export a virtual machine from one data center and import it to another.

Before You Begin

Log in as the **student** user on **workstation** and run the **lab image-cr start** command. This command creates the rh1 and rh2 virtual machines, and ensures that the **hostb.lab.example.com** and **hostc.lab.example.com** RHV-H hosts are active in the **datacenter1** data center. This command also ensures that the **hostd.lab.example.com** RHV-H host is active in the **datacenter2** data center.

```
[student@workstation ~]$ lab image-cr start
```

Instructions

Accomplish the following tasks.

- Ensure that the **rh1** virtual machine successfully live migrates to another host in the cluster. Manually live migrate **rh1** to confirm this behavior.
- Create a new NFS export domain named **movestorage** using **172.24.0.8:/exports/data2** as the NFS export path. Use this NFS export domain to export and import the **rh2** virtual machine from **datacenter1** to **datacenter2** respectively. The imported VM should be named **rh2-dc2** and must have its NIC attached to the **ovirtmgmt** logical network. Set **RHV Managed Virtual Machine** as the custom message of the day in **rh2**.
- The URL for the Administration Portal is `https://rhvm.lab.example.com`. The username is **admin** and the password is **redhat**. Use the **internal** profile.

Evaluation

On **workstation**, run the **lab image-cr grade** command to confirm success of this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab image-cr grade
```

Finish

On **workstation**, run the **lab image-cr finish** script to complete this lab.

```
[student@workstation ~]$ lab image-cr finish
```

This concludes the lab.

Solution

Managing Virtual Machines

In this review, you will migrate a virtual machine from one host to another and move a virtual machine from one cluster to another.

Outcomes

You should be able to:

- Migrate virtual machines between hosts.
- Create an NFS export storage domain.
- Export a virtual machine from one data center and import it to another.

Before You Begin

Log in as the **student** user on **workstation** and run the **lab image-cr start** command. This command creates the rh1 and rh2 virtual machines, and ensures that the **hostb.lab.example.com** and **hostc.lab.example.com** RHV-H hosts are active in the **datacenter1** data center. This command also ensures that the **hostd.lab.example.com** RHV-H host is active in the **datacenter2** data center.

```
[student@workstation ~]$ lab image-cr start
```

Instructions

Accomplish the following tasks.

- Ensure that the **rh1** virtual machine successfully live migrates to another host in the cluster. Manually live migrate **rh1** to confirm this behavior.
- Create a new NFS export domain named **movestorage** using **172.24.0.8:/exports/data2** as the NFS export path. Use this NFS export domain to export and import the **rh2** virtual machine from **datacenter1** to **datacenter2** respectively. The imported VM should be named **rh2-dc2** and must have its NIC attached to the **ovirtmgmt** logical network. Set **RHV Managed Virtual Machine** as the custom message of the day in **rh2**.
- The URL for the Administration Portal is `https://rhvm.lab.example.com`. The username is **admin** and the password is **redhat**. Use the **internal** profile.

1. Start the **rh1** virtual machine. After the **rh1** virtual machine is started, manually live migrate the **rh1** virtual machine to run on another host in the cluster.

   1.1. On **workstation**, use Firefox to log in to RHV-M **Administration Portal** as **admin** from the **internal** profile, using **redhat** as the password. The URL for the Administration Portal is `https://rhvm.lab.example.com`.

   1.2. From the menu, navigate to **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

1.3. From the list of virtual machines, select **rh1** and click **Run** to start the virtual machine. It may take a minute for the virtual machine to start.

1.4. View the **Host** field for **rh1** to determine the host it is currently using.

1.5. From **Compute >> Virtual Machines** page, select **rh1**. Click **Migrate** to migrate **rh1** from the current host to the other available host in the cluster. The window titled **Migrate VM(s)** displays.

In the **Migrate VM(s)** window, leave **Automatically Choose Host** selected for the **Destination Host** field. Click **Migrate** to confirm your intention to migrate **rh1**.

1.6. Wait until the **Status** for the virtual machine transitions from **Migrating From: 0%** to **Up**. Verify that the **Host** field for **rh1** has changed to a different host in the cluster.

2. To support exporting the **rh2** virtual machine from **datacenter1** and importing to **datacenter2**, create a new NFS export domain named **movestorage**. Use the NFS export **172.24.0.8:/exports/data2** as the back-end storage for **movestorage**.

2.1. From the menu, navigate to **Storage** → **Domains** to access the **Storage >> Storage Domains** page.

2.2. Click **New Domain** to create a new storage domain. The window titled **New Domain** displays.

2.3. In the **New Domain** window, set the values of the fields according to the following table, and click **OK** to create the storage domain.

Field	Value
Data Center	**datacenter1**
Domain Function	**Export**
Storage Type	**NFS**
Host to Use	**hostc.lab.example.com**
Name	**movestorage**
Export Path	**172.24.0.8:/exports/data2**

2.4. Verify that the **Cross Data Center Status** is **Active** for the **movestorage** storage domain. It may take a minute for the status to transition from **Locked** to **Active**.

3. Set **RHV Managed Virtual Machine** as the custom message of the day in **rh2**. Use this message of the day to verify that the **rh2** virtual machine was successfully exported and imported.

3.1. When the **movestorage** storage domain becomes active, navigate to **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

3.2. In the **Compute >> Virtual Machines** page, right-click the row for **rh2** and select **Run** to start the virtual machine. After the virtual machine is started, click **Console** to access the console of **rh2**. The **Opening console.vv** window opens and prompts for your confirmation to open the connection file (**console.vv**). Click **OK** to open the **console.vv** file using Remote Viewer.

> **Note**
>
> If Remote Viewer prompts for your confirmation to inhibit shortcuts, click **Allow**.

3.3. Log in to **rh2** as **root** with the password **redhat**.

3.4. Run the **echo "RHV Managed Virtual Machine" >> /etc/motd** command to append the **/etc/motd** file with the given message. Also, run the **sync** command to flush the Linux buffer cache to the disk so that the changes in **/etc/motd** gets successfully written to the disk.

3.5. Log out from **rh2** and close the console of the virtual machine.

4. Export the **rh2** virtual machine from the **datacenter1** data center.

4.1. Right-click the row for the **rh2** virtual machine, and click **Power Off**. The window titled **Power Off Virtual Machine(s)** displays. Click **OK** to confirm your intention of powering down the virtual machine.

4.2. In the **Compute >> Virtual Machines** page, verify that the value of the **Status** field for **rh2** is **Down**.

4.3. Select the **rh2** virtual machine and click the three vertical dots next to the **Create Snapshot** button. From the menu that displays, click **Export to Export Domain**. The window titled **Export Virtual Machine** displays. In the **Export Virtual Machine** window, ignore the two check boxes and click **OK** to confirm exporting the virtual machine.

While **rh2** is being exported, the **Status** field for **rh2** displays **Image Locked**. After **rh2** gets successfully exported, the value of the **Status** field transitions from **Image Locked** to **Down**.

4.4. Navigate to **Events** menu, and search for the log message containing **Vm rh2 was exported successfully to movestorage**. This log message confirms that the **rh2** virtual machine was successfully exported.

5. Detach the **movestorage** export domain from **datacenter1** and attach it to **datacenter2**.

5.1. From the menu, navigate to **Compute** → **Data Centers** to access the **Compute >> Data Centers** page.

5.2. In the **Compute >> Data Centers** page, click on the name of the **datacenter1** data center.

5.3. In the **Storage** tab for **datacenter1**, select **movestorage** and click **Maintenance**. The window titled **Storage Domain maintenance** displays. Click **OK** to confirm your intention of marking **movestorage** for maintenance. Wait for the status of **movestorage** to transition to **Maintenance**. It may take a minute for the status to transition from **Active** to **Maintenance**.

5.4. In the **Compute >> Data Centers >> datacenter1** page, select **movestorage** and click **Detach** to detach it from **datacenter1**. The window titled **Detach Storage** displays. Click **OK** to confirm detaching the export domain.

5.5. From the menu, navigate to **Compute** → **Data Centers** to access the **Compute >> Data Centers** page.

5.6. In the **Compute >> Data Centers** page, click on the name of the **datacenter2** data center.

5.7. In the **Storage** tab for **datacenter2**, click **Attach Export**. The window titled **Attach Export Domain** displays. In the **Attach Export Domain** window, click the radio button next to **movestorage**. Click **OK** to confirm attaching the export domain. Wait for the activation of **movestorage** to be complete. It may take a minute for the status to transition from **Locked** to **Active**.

6. Import the **rh2** virtual machine to the **datacenter2** data center as **rh2-dc2**.

6.1. From the menu, navigate to **Storage** → **Domains** to access the **Storage >> Storage Domains** page.

6.2. In the **Storage >> Storage Domains** page, click on the name of **movestorage**. In the **Storage >> Storage Domains >> movestorage** page that displays, click on the **VM Import** tab.

6.3. In the **VM Import** tab, click the row for **rh2**, and click **Import**. The window titled **Import Virtual Machine(s)** displays. In the **Import Virtual Machine(s)** window, click **OK** to confirm importing the virtual machine.

After you click **OK**, a warning sign is displayed next to the name of the virtual machine. Hover over the warning sign and confirm that the warning is regarding the name of the virtual machine. Click on the warning sign and change the name of the virtual machine from **rh2** to **rh2-dc2** from the **General** tab of the **Import Virtual Machine(s)** window. Click **OK** to confirm the change in name and import the virtual machine.

6.4. The **Import Virtual Machine(s)** window displays a message about the start of the import process, which can be tracked from the **Events** menu. Click **Close** to close the **Import Virtual Machine(s)** window.

7. Verify that the **rh2** virtual machine from **datacenter1** has been successfully imported as **rh2-dc2** to **datacenter2**

7.1. From the menu, navigate to **Events** to search for the log messages related to the import of the virtual machine.

7.2. Search for the log message containing **Vm rh2-dc2 was imported successfully to Data Center datacenter2, Cluster cluster2**. This log message confirms that the virtual machine has been successfully imported. It may take a couple of minutes for the log message to display.

8. Remove any existing NIC from **rh2-dc2**, and create a NIC for **rh2-dc2** from the **ovirtmgmt** logical network.

8.1. From the menu, navigate to **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

8.2. In the **Compute >> Virtual Machines** page, click on the name of the **rh2-dc2** virtual machine to access the **Compute >> Virtual Machines >> rh2-dc2** page. In the **Compute >> Virtual Machines >> rh2-dc2** page, click on the **Network Interfaces** tab.

8.3. Click **Remove** to remove **nic1** that uses **[N/A]** as the value of the **Network Name** field, from the **rh2-dc2** virtual machine. The window titled **Remove Network Interface(s)** displays. In the **Remove Network Interface(s)**, click **OK** to confirm the removal of the NIC.

8.4. Click **New** to add a new NIC to **rh2-dc2** from **ovirtmgmt** logical network. The **New Network Interface** window displays.

8.5. In the **New Network Interface** window, ensure that the **Name** is **nic1**, the **Profile** is **ovirtmgmt/ovirtmgmt**, and the **Type** is **VirtIO**. Leave the other fields to their default values and click **OK** to confirm adding the NIC. Confirm that the **Network Name** for the newly added **nic1** is **ovirtmgmt**.

8.6. From the menu, navigate to **Compute** → **Virtual Machines** to access the **Compute >> Virtual Machines** page.

8.7. Click on the row for the **rh2-dc2** virtual machine, and click **Run** to start **rh2-dc2**.

9. Verify that **rh2-dc2** has an IP address configured in the **172.25.250.0/24** subnet range in its **eth0** network interface.

9.1. Click on the row for the **rh2-dc2** virtual machine, and click **Console** to access the console of **rh2-dc2**. The **Opening console.vv** window opens and prompts for your confirmation to open the connection file (**console.vv**). Click **OK** to open the **console.vv** file using Remote Viewer.

9.2. Log in to **rh2-dc2** as **root** with the password **redhat**. On successful login, you should see the **RHV Managed Virtual Machine** message you configured previously.

9.3. Run the **ip address show** command to view the network configuration of the virtual machine. Confirm that the virtual machine has an IP address in the **172.25.250.0/24** subnet range.

9.4. Log out from **rh2-dc2** and close the console of the virtual machine.

Evaluation

On **workstation**, run the **lab image-cr grade** command to confirm success of this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab image-cr grade
```

Finish

On **workstation**, run the **lab image-cr finish** script to complete this lab.

```
[student@workstation ~]$ lab image-cr finish
```

This concludes the lab.

Lab

Backing Up and Upgrading Red Hat Virtualization

In this review, you back up and restore RHV-M, and update RHV-H hosts.

Outcomes

You should be able to:

- Backup and restore a Red Hat Virtualization Manager installation
- Update Red Hat Virtualization Hosts

Before You Begin

Log in as the **student** user on **workstation** and run the **lab update-cr start** command.

```
[student@workstation ~]$ lab update-cr start
```

Instructions

Perform the following steps:

- Create a full backup without stopping the Red Hat Virtualization infrastructure. The backup file should be created at **/root/rhvm-backup.tgz** and the log file should be created at **/root/backup.log**.
- Clean out the Red Hat Virtualization Manager configuration using **engine-cleanup** and then restore your backup into that clean environment. Your restore command should create **/root/restore.log** and it should restore permissions. Log in to the **Administration Portal** as the **admin** user to confirm that the restoration from backup was successful.
- Apply updates to the hosts running Red Hat Virtualization Host and currently attached to **cluster1** in your environment. Use the provided **rhvh_updates.repo** file from `http://materials.example.com/yum.repos.d/rhvh_updates.repo` to enable Yum repositories containing the necessary software updates.

Evaluation

As the **student** user on **workstation**, run the **lab update-cr** script with the **grade** argument, to confirm success on this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab update-cr grade
```

This concludes the lab.

Solution

Backing Up and Upgrading Red Hat Virtualization

In this review, you back up and restore RHV-M, and update RHV-H hosts.

Outcomes

You should be able to:

- Backup and restore a Red Hat Virtualization Manager installation
- Update Red Hat Virtualization Hosts

Before You Begin

Log in as the **student** user on **workstation** and run the **lab update-cr start** command.

```
[student@workstation ~]$ lab update-cr start
```

Instructions

Perform the following steps:

- Create a full backup without stopping the Red Hat Virtualization infrastructure. The backup file should be created at **/root/rhvm-backup.tgz** and the log file should be created at **/root/backup.log**.
- Clean out the Red Hat Virtualization Manager configuration using **engine-cleanup** and then restore your backup into that clean environment. Your restore command should create **/root/restore.log** and it should restore permissions. Log in to the **Administration Portal** as the **admin** user to confirm that the restoration from backup was successful.
- Apply updates to the hosts running Red Hat Virtualization Host and currently attached to **cluster1** in your environment. Use the provided **rhvh_updates.repo** file from `http://materials.example.com/yum.repos.d/rhvh_updates.repo` to enable Yum repositories containing the necessary software updates.

1. Create a full backup of Red Hat Virtualization Manager without stopping the RHV infrastructure.

    1.1. From **workstation**, open a terminal and use **ssh** to log in to **rhvm.lab.example.com** using the user name **root**. The **student** user on the **workstation** system is configured with the SSH keys for **root** user from **rhvm.lab.example.com** to allow passwordless access.

```
[student@workstation ~]$ ssh root@rhvm.lab.example.com
...output omitted...
```

1.2. To create a full backup without stopping RHV infrastructure, issue the **engine-backup** command, specifying the scope of this backup, the name of the backup file, and the name of the log file:

```
[root@rhvm ~]# engine-backup --scope=all --mode=backup \
 --file=rhvm-backup.tgz --log=backup.log
Backing up:
Notifying engine
- Files
- Engine database 'engine'
- DWH database 'ovirt_engine_history'
Packing into file 'rhvm-backup.tgz'
Notifying engine
Done.
```

1.3. Exit out of **rhvm.lab.example.com**.

2. Clean up the RHV-M configuration using **engine-cleanup** and restore your backup into that clean environment. Log in to the **Administration Portal** as the **admin** user to confirm that the restoration from backup was successful.

2.1. From **workstation**, open a terminal and use **ssh** to log into **hosta.lab.example.com** as **root**. The **student** user on the **workstation** system is configured with the SSH keys needed to log in to **hosta.lab.example.com** as the **root** user.

```
[student@workstation ~]$ ssh root@hosta.lab.example.com
...output omitted...
```

2.2. Set the maintenance mode to **global**.

```
[root@hosta ~]# hosted-engine --set-maintenance --mode=global
```

Exit from hosta.lab.example.com.

```
[root@hosta ~]# exit
...output omitted...
```

2.3. SSH to **rhvm.lab.example.com** as root. Issue the **engine-cleanup** command to completely clean up the environment. The **engine-cleanup** command executes an interactive environment, taking you through a series of questions with default settings displayed in square brackets.

```
[root@rhvm ~]# engine-cleanup
[ INFO  ] Stage: Initializing
(...)
[ INFO  ] Stage: Environment customization
          Do you want to remove all components? (Yes, No) [Yes]: <ENTER>
```

```
(...)
          During execution engine service will be stopped (OK, Cancel)
 [OK]: <ENTER>
          All the installed ovirt components are about to be removed, data will be
 lost (OK, Cancel) [Cancel]: OK
(...)

          --== END OF SUMMARY ==--

[ INFO  ] Stage: Clean up
          Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-
remove-20190827063123-w605h6.log
[ INFO  ] Generating answer file '/var/lib/ovirt-engine/setup/
answers/20190827063653-cleanup.conf'
[ INFO  ] Stage: Pre-termination
[ INFO  ] Stage: Termination
[ INFO  ] Execution of cleanup completed successfully
```

2.4. To restore a full backup of the RHV infrastructure, on **rhvm.lab.example.com** server issue the **engine-backup** command, specifying the scope of this restore, the name of the backup file, and the name of the log file:

```
[root@rhvm ~]# engine-backup --scope=all --mode=restore \
 --file=rhvm-backup.tgz --log=restore.log --restore-permissions
Preparing to restore:
- Unpacking file 'rhvm-backup.tgz'
Restoring:
- Files
- Engine database 'engine'
  - Cleaning up temporary tables in engine database 'engine'
  - Updating DbJustRestored VdcOption in engine database
  - Resetting DwhCurrentlyRunning in dwh_history_timekeeping in engine database
  - Resetting HA VM status
------------------------------------------------------------------------------
Please note:

The engine database was backed up at 2019-08-27 06:17:35.000000000 -0400 .

Objects that were added, removed or changed after this date, such as virtual
machines, disks, etc., are missing in the engine, and will probably require
recovery or recreation.
------------------------------------------------------------------------------
- DWH database 'ovirt_engine_history'
You should now run engine-setup.
Done.
```

2.5. Run the **engine-setup** command with the **--accept-defaults** option to ensure that **ovirt-engine** service is correctly configured. Use the **--offline** mode for this example:

```
[root@rhvm ~]# engine-setup --accept-defaults --offline
[ INFO  ] Stage: Initializing
...output omitted...
         --== END OF SUMMARY ==--
```

```
[ INFO  ] Stage: Clean up
          Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-
setup-20190827064927-pq0980.log
[ INFO  ] Generating answer file '/var/lib/ovirt-engine/setup/
answers/20190827065453-setup.conf'
[ INFO  ] Stage: Pre-termination
[ INFO  ] Stage: Termination
[ INFO  ] Execution of setup completed successfully
```

2.6. Exit from the RHV Manager instance, and then set the maintenance mode back to normal on **hosta**.

```
[root@rhvm ~]$ exit
logout
Connection to rhvm.lab.example.com closed.
[student@workstation ~]$ ssh root@hosta.lab.example.com
```

Using hosted-engine, set the maintenance mode back to **none**.

```
[root@hosta ~]# hosted-engine --set-maintenance --mode=none
```

2.7. Confirm that everything is working again, and that the restoration from backup was successful. On **workstation**, open Firefox and log in to the **Administration Portal** as the **admin** user with the **internal** profile using **redhat** as password.

3. Apply updates to the RHV-H hosts for **cluster1** in your environment. Use the provided **rhvh_updates.repo** file from `http://materials.example.com/yum.repos.d/rhvh_updates.repo` to access the existing Yum update repositories.

3.1. From **workstation**, use a terminal and with **ssh** to log in to **hostb.lab.example.com** using the **root** user and **redhat** as password.

```
[student@workstation ~]$ ssh root@hostb.lab.example.com
...output omitted...
```

3.2. Download the **rhvh_updates.repo** file from `http://materials.example.com/yum.repos.d/rhvh_updates.repo` and place it in the **/etc/yum.repos.d/** directory to enable those repositories.

```
[root@hostb ~]# curl http://materials.example.com/yum.repos.d/rhvh_updates.repo \
-o /etc/yum.repos.d/rhvh_updates.repo
```

3.3. From **workstation**, open a terminal. With the **ssh** command, log in to **hostc.lab.example.com** using the **root** user and **redhat** as password.

```
[student@workstation ~]$ ssh root@hostc.lab.example.com
...output omitted...
```

3.4. Download the **rhvh_updates.repo** file from `http://materials.example.com/yum.repos.d/rhvh_updates.repo` and place it in the **/etc/yum.repos.d/** directory to enable those repositories.

```
[root@hostc ~]# curl http://materials.example.com/yum.repos.d/rhvh_updates.repo \
-o /etc/yum.repos.d/rhvh_updates.repo
```

3.5. On **workstation** open Firefox, go to the RHV-M web interface. Click on the **Administration Portal** link and log in to the web interface as the **admin** user with the **internal** profile using **redhat** as password.

3.6. Click **Compute >> Hosts** menu to access the **Compute** >> **Hosts** page.

3.7. From the list of available RHV-H hosts, highlight the **hostb.lab.example.com** host. Click the **Installation** button, followed by **Check for Upgrade**.

3.8. When the **Upgrade Host** dialog window opens, click **OK** to confirm the upgrade check.

Notice that after a while, a new **Action Item** comes up in the same line as the RHV-H host. This new icon is a notification that an upgrade for this host is available.

3.9. Right-click the **hostb.lab.example.com** host. From the displayed menu, choose **Installation**, followed by **Upgrade**.

3.10. In the **Upgrade Host** dialog window, click the **OK** button to start the upgrade. If you start the upgrade process of **hostc** before the **hostb** status changes to **Up** again, you will change the **datacenter1** from the active state to the nonresponsive state. There needs to be at least one active host for a data domain to be in the active state.

3.11. From the list of available RHV-H hosts, right-click the **hostc.lab.example.com** host. From the displayed menu, choose **Installation** followed by **Check for Upgrade**.

3.12. When the **Upgrade Host** dialog window opens, click **OK** to confirm the upgrade check.

Notice that after a while, a new **Action Item** comes up on the line with the RHV-H host. This new icon is a notification that an upgrade for this host is available.

3.13. Click the **Compute >> Virtual Machines** tab and verify that a virtual machine is running on the **hostc** host. If a virtual machine is running, power that machine off using any of the available methods.

3.14. Click the **Hosts** tab. Right-click the **hostc.lab.example.com** host. From the displayed menu, choose **Installation**, followed by **Upgrade**.

3.15. In the **Upgrade Host** dialog window, click the **OK** button to start the upgrade.

3.16. Wait and watch the upgrade procedure take place.

Evaluation

As the **student** user on **workstation**, run the **lab update-cr** script with the **grade** argument, to confirm success on this exercise. Correct any reported failures and rerun the script until successful.

```
[student@workstation ~]$ lab update-cr grade
```

This concludes the lab.